APEM journal
Advances in Production Engineering & Management
Volume 18 | Number 3 | September 2023 | pp 303–316
https://doi.org/10.14743/apem2023.3.474
ISSN 1854-6250
Journal home: apem-journal.org
Original scientific paper

An improved deep reinforcement learning approach: A case study for optimisation of berth and yard scheduling for bulk cargo terminal

Ai, T.[a], Huang, L.[a,*], Song, R.J.[b], Huang, H.F.[c], Jiao, F.[d], Ma, W.G.[a]

[a]School of Economics and Management, Beijing Jiaotong University, Beijing, P.R. China
[b]Experimental Center of Data Science and Intelligent Decision Making, School of Management, Hangzhou Dianzi University, Hangzhou, P.R. China
[c]CRRC Information Technology CO., LTD, P.R. China
[d]Research and Development Center, Agricultural Bank of China, Beijing, P.R. China

ABSTRACT

The cornerstone of port production operations is ship handling, necessitating judicious allocation of diverse production resources to enhance the efficiency of loading and unloading operations. This paper introduces an optimisation method based on deep reinforcement learning to schedule berths and yards at a bulk cargo terminal. A Markov Decision Process model is formulated by analysing scheduling processes and unloading operations in bulk port imports business. The study presents an enhanced reinforcement learning algorithm called PS-D3QN (Prioritised Experience Replay and Softmax strategy-based Dueling Double Deep Q-Network), amalgamating the strengths of the Double DQN and Dueling DQN algorithms. The proposed solution is evaluated using actual port data and benchmarked against the other two algorithms mentioned in this paper. The numerical experiments and comparative analysis substantiate that the PS-D3QN algorithm significantly enhances the efficiency of berth and yard scheduling in bulk terminals, reduces the cost of port operation, and eliminates errors associated with manual scheduling. The algorithm presented in this paper can be tailored to address scheduling issues in the fields of production and manufacturing with suitable adjustments, including problems like the job shop scheduling problem and its extensions.

ARTICLE INFO

Keywords:
Bulk cargo terminal;
Scheduling;
Optimisation;
Markov decision process (MDP) model;
Deep reinforcement learning;
Prioritised experience replay and softmax strategy-based dueling;
Double deep Q-network (PS-D3QN)

**Corresponding author:*
lhuang@bjtu.edu.cn
(Huang L.)

Article history:
Received 22 August 2023
Revised 5 November 2023
Accepted 7 November 2023

1. Introduction

The significance of maritime logistics has been underscored by the "2022 Maritime Review" report published by the United Nations Conference on Trade and Development (UNCTAD) [1]. Maritime shipping constitutes over 80 % of global trade and is increasingly pivotal in the global economy. Ports act as crucial intermediaries, facilitating the transfer of a substantial volume of goods through loading and unloading operations. With the continuous growth of global trade and the increasing complexity of logistics, efficient port operation scheduling is vital for optimis-

ing resource utilisation, enhancing loading and unloading efficiency, and minimising operational costs.

Maritime cargo encompasses various types, including containers, dry bulk commodities (such as coal, steel, and grains), and liquids. Consequently, ports can be categorised into container terminals and bulk cargo terminals. Unlike container terminals, bulk cargo terminals involve a more intricate range of goods, each necessitating distinct loading and unloading processes. Currently, most bulk cargo terminals rely on manual expertise for scheduling, which may compromise the efficiency and rationality of scheduling plans. Furthermore, berth and yard scheduling are often treated separately in practical scheduling processes, lacking a unified approach. Thus, an intelligent berth and yard scheduling method in bulk cargo terminals is imperative to enhance logistical efficiency, on-time delivery rates, and port operational cost reduction, thereby contributing to sustainable economic development.

While a substantial body of literature exists on port resource scheduling and operational optimisation, research on optimisation for bulk cargo terminals is relatively limited compared to container terminals. The primary distinctions between container and bulk cargo terminals lie in the layout of port resources, loading and unloading procedures, handling machinery, and cargo types [2]. Given the more intricate variety of goods involved in bulk cargo terminals, it is imperative to delve into the optimisation of their loading and unloading procedures. Within bulk cargo terminals, berths and yards are the most critical and scarce resources during operational processes, making berth and yard allocation the focal points.

Berth Allocation Problem (BAP), as highlighted by Bierwirth *et al*. [3, 4] in 2010 and 2015, has been a subject of study, with subsequent research largely building upon their classification scheme. This classification is equally applicable to bulk cargo terminals. BAP can be categorised into four classes based on space, time, processing time, and performance metrics [4]. Spatially, terminals may exhibit discrete, continuous, or hybrid layouts. Temporally, BAP can be classified as static arrival, dynamic arrival, periodic arrival, and stochastic arrival. Considering the influence of ship loading and unloading times, BAP can be classified as "fixed" or "pos", depending on the position. Performance metrics categorise BAP models based on optimisation goals, with most research in BAP prioritising minimising total ship dwell time.

Berth optimisation in bulk cargo terminals aims to improve berth operational efficiency and optimise berth allocation. In a study by Đelović *et al*. [5], the authors analysed the berth productivity of multifunctional bulk terminals using mathematical and statistical methods. Their goal was to systematically identify the factors influencing berth productivity and categorise 14 groups of influential factors. The study confirmed a substantial disparity between gross and net berth productivity, primarily attributable to the substantial portion of non-operational time during vessel dwell periods.

For the BAP in bulk cargo terminals, Barros *et al*. [6] addressed the problem in tidal bulk cargo ports, formulating an integer linear programming model for discrete terminal layout and dynamic arrival scenarios while accounting for inventory constraints. Umang *et al*. [7] explored mixed terminal layouts and dynamic arrivals, proposing exact methods and a heuristic approach for minimising total service time, considering various cargo types aboard ships. Ribeiro *et al*. [8] tackled the discrete BAP in a dynamic arrival scenario for ore terminals, aiming to minimise delay and scheduling costs by employing mixed-integer linear programming and adaptive large neighbourhood search. Ernst *et al.* [9] researched continuous BAP with dynamic ship arrivals under tidal constraints, proposing two mixed-integer linear formulations and testing them on different instances. Cheimanoff *et al*. [10] studied multiple continuous terminals with dynamic arrivals, considering tidal restrictions and terminal-specific limitations for each ship, using mixed-integer linear models and iterative local search for small- and large-scale instances.

Yard management is a cornerstone of port terminal operations, as intelligent management of storage and transportation within yards can optimise space utilisation and decrease ship loading and unloading times, thereby enhancing port operational efficiency. Research focusing on yard allocation optimisation is limited compared to BAP studies. Tang *et al*. [11] examined joint storage space allocation and ship scheduling, establishing a mixed-integer programming model solved via the Benders decomposition algorithm. In their work, yard storage areas were divided

into slots, each dedicated to a single product, with the possibility of extending product stacks across multiple slots. Rocha de Paula *et al*. [12] devised a genetic algorithm to maximise coal terminal throughput by arranging coal arrivals, determining stack and recovery cycles, and scheduling ship arrival and departure times.

In bulk cargo terminals, BAP is often coupled with yard allocation problems. Robenek *et al*. [13] extended the work of Umang *et al*. [7], expanding BAP to allocate yard positions to incoming ships based on specific cargo types, aiming to minimise ship service time. Unsal *et al.* [14] integrated berth allocation, stacker scheduling, and yard allocation in the context of exporting coal terminals. This problem entailed operational challenges and constraints concerning tidal windows, multiple stocking pads, non-crossing stackers, ships and berth size, addressed through integer programming formulations.

For integrated scheduling in bulk cargo terminals, some studies have combined berth and yard allocation, albeit primarily focusing on ship operation time as an optimisation goal and overlooking transport costs. Others have considered coal and ore terminals, both specialised cases, and may not be easily extended to highly diversified bulk cargo terminals. Therefore, this study focuses on a comprehensive bulk cargo terminal with diverse goods and aims to optimise berth and yard allocation in a dynamic discrete environment, minimising ship stay time and total transport costs.

Presently, research on bulk cargo port scheduling employs mathematical programming methods and intelligent algorithms, including heuristics, simulation, and genetic algorithms. These conventional mathematical and intelligent optimisation algorithms can yield favourable results when aptly modelled for certain issues. However, the berth and yard scheduling issues addressed in this study are characterised by dynamic and uncertain environments with a large scope. Traditional mathematical programming methods, heuristic algorithms, and similar optimisation techniques may lack the flexibility to address real-time changes in complex production scheduling scenarios. In contrast, recent advancements in artificial intelligence methods, such as deep learning and reinforcement learning, offer promising solutions to such challenges. For instance, Tian *et al*. [15] proposed a data prediction model for the dynamic job-shop scheduling problem (DJSP) using the Long Short-Term Memory Network (LSTM). They improved the model by integrating Dropout technology and other techniques, subsequently assessing its performance. Moreover, they devised a scheduling model with objective functions encompassing maximum makespan, total device load and key device load. Ultimately, an enhanced Multi-Objective Genetic Algorithm (MOGA) was formulated to tackle this challenge.

The scheduling problem under study falls under the category of sequential decision-making in a finite state space. The environment's state at the next time step is solely influenced by the current environment state and the actions taken by port resources. It follows the Markov property and can be formulated as a Markov Decision Process (MDP). Reinforcement Learning [16] is an artificial intelligence technique designed to address MDPs, making it well-suited for solving the scheduling problem presented in this study. Reinforcement learning algorithms have matured over recent years and span multiple branches. Depending on action selection methods, reinforcement learning can be classified into value-based methods and policy gradient-based methods [16]. Among the most common value-based algorithms are the Deep Q-Network (DQN) [17] and its variants. The Double DQN algorithm proposed by Van Hasselt *et al*. [18] and the Dueling DQN algorithm by Wang *et al*. [19] optimise target network Q-value computation and neural network architecture, respectively. Additionally, the DDPG algorithm is a popular policy gradient-based method [20]. Given that this scheduling problem involves discrete action spaces and unknown state transition probabilities, requiring the agent to continually interact with the environment for learning, value-based model-free DQN algorithms and their variants are better suited for solving this problem.

Deep learning and reinforcement learning have found applications across various fields [21-23]. However, its application to port resource scheduling remains relatively limited. Li *et al*. [24] developed a MILP mathematical model to minimise total ship stay time and employed a genetic algorithm as a fundamental optimisation method. They introduced a Q-learning approach with dynamic parameter selection for crossbreeding and mutation, along with a simulated annealing

operation to address ship scheduling. Dai *et al*. [25] examined BAP and QCSP for container terminals. They created a Markov Decision Process model accounting for terminal loading capacity, cargo types, and switch setup time. Their research involved greedy insertion algorithms and DDQN reinforcement learning algorithms for offline and online scenarios. Li *et al*. [26] proposed an improved Double DQN algorithm for scheduling bulk cargo loading at a coal terminal. Their approach enhanced the ε-greedy policy and introduced a random policy for illegal actions, increasing algorithm convergence.

This study presents a deep reinforcement learning approach called the Prioritised Experience Replay and Softmax strategy-based Dueling Double Deep Q-Network (PS-D3QN). The effectiveness of this method is validated through a case study on import operations at a port in southern China. By considering the scheduling processes and unloading operations, the dynamic discrete environment of bulk terminal berth and yard scheduling is modelled as a Markov Decision Process (MDP). The PS-D3QN algorithm is subsequently employed to solve this model using actual port conditions and collected data.

The main contributions of this study are summarised as follows:

- By analysing import business scheduling processes and ship unloading operations at the port, along with incorporating real port conditions and related data, this study formulated the problem of berth and yard scheduling in bulk cargo terminals as a Markov Decision Process (MDP) model. The model's state space and action space were designed, aiming to minimise total ship stay time at the port and total transport costs. A linearly weighted reward function was devised, and legal action validity was defined, providing the basis for the subsequent introduction of deep reinforcement learning algorithms.
- This study introduced a berth and yard real-time scheduling method (PS-D3QN) based on an improved DQN algorithm. This method combined the advantages of the Double DQN and Dueling DQN algorithms, optimising the algorithm by introducing a well-designed Prioritised Experience Replay (PER) mechanism and a softmax action selection strategy. This optimisation enhanced the algorithm's convergence and stability.
- The proposed PS-D3QN algorithm was validated using actual port data from a bulk cargo terminal case study. Comparative analyses were conducted with the Double DQN and Dueling DQN algorithms, as well as real-world scheduling plans. The experimental results demonstrated the effectiveness and reliability of the proposed algorithm in addressing the bulk cargo berth and yard scheduling problem.

The rest of this paper is organised as follows. Section 2 introduces the model construction and optimisation algorithm design. Section 3 presents numerical experiments and discussions using actual port data. Finally, Section 4 summarises the study's achievements and contributions and suggests avenues for further improvement.

2. Models and algorithms

2.1 Problem statement and MDP modelling

This study focuses on the ship unloading operations within the import business of a bulk cargo terminal located in southern China. The research commences by investigating the specific operational environment and business procedures of the port. A detailed analysis of the port's actual scheduling process follows. The simplified scheduling process is illustrated in Fig. 1.

Upon receiving ship forecast information, the planning department formulates day and night plans considering current berth utilisation. Subsequently, the warehouse department develops yard operation plans based on these day and night plans and the cargo transportation mode. After the ship's arrival, the scheduling department arranges berthing, while the warehouse department coordinates ship unloading in accordance with yard operation plans and the day-night scheme. The scheduling department concludes the ship's operations by orchestrating its departure from the port.

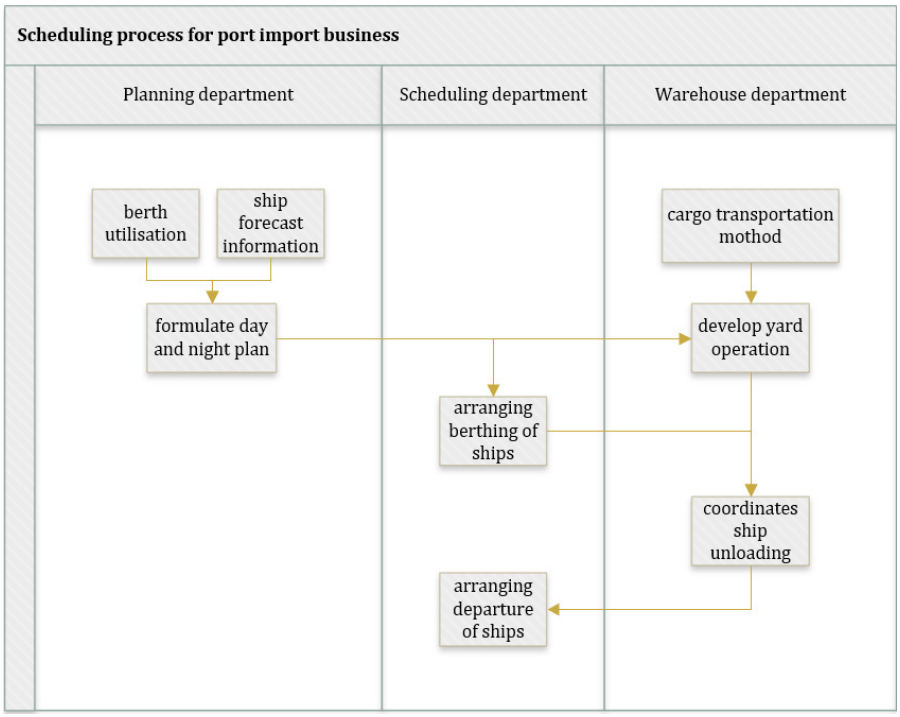

Fig. 1 Scheduling process for port import business

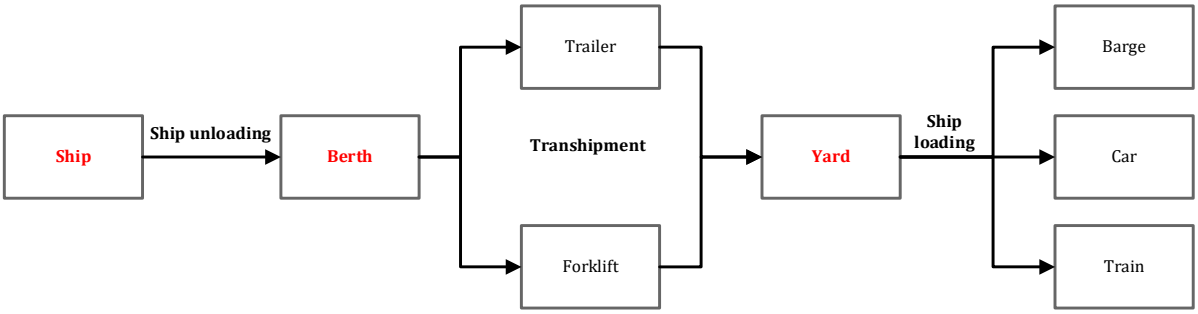

Fig. 2 Production operation process of port import business

The production operations of port import businesses encompass two main facets: ship unloading and transportation operations. Fig. 2 provides an overview of these processes. Ship unloading involves berthing, cargo unloading, transport, and storage. Conversely, transportation operations encompass the relocation of cargo within the yard and its transportation outside the port via railways, highways, coastlines, and other channels. Throughout the port's operational procedures, berths and yards, as precious resources, serve as pivotal nodes bridging ship unloading and transportation operations.

Analysing the port import business's scheduling process and its production operations highlights the current practice of separately planning berth and yard scheduling within bulk cargo terminals. This approach neglects the interdependencies and mutual constraints between yard and berth resources. Furthermore, the scheduling process relies heavily on manual experience, leading to subjective influences. Communication and coordination among departments demand considerable time, ultimately resulting in suboptimal scheduling efficiency and a lack of scientifically driven optimisation goals and decision-making criteria. Consequently, the challenge of port berth and yard scheduling necessitates a holistic resource allocation optimisation supported by intelligent methodologies to enhance scheduling efficiency, reduce manual scheduling costs, and elevate port production and operation efficiency.

This paper's scheduling problem can be defined as follows: Given the planned arrival time and essential information about ships, berths, and yards, the objective is to address the unloading operations in the port's import business. Specifically, the study encompasses all ships antici-

pated to arrive and depart within a fixed planning period. The focus lies on ship docking, loading/unloading operation timing, and the unloading location. The optimisation aims to minimise the total dwell time for all arriving ships and the aggregate cargo transportation costs, resulting in berth and yard allocation plans for each ship.

This study establishes an MDP model by integrating the operational processes, layout environment, and actual data of ships, berths, and yards in the port to address this scheduling problem. The pertinent design elements of the model are detailed as follows:

(1) State space

The state space encompasses the operational and usage states of ships, berths, and yards. This information is summarised in Table 1.

Table 1 State space of berth and yard scheduling in bulk terminals

State space	Data structure	Dimension	Category	Description
C_s	Arrays	10	Int	Type of cargo loaded on the ship (0: empty cargo ship, 1: steel, 2: coal, 3: grain, 4: ore)
W_s	Arrays	10	Int	Weight of cargo loaded on the ship
B_s	Arrays	10	Int	Ship docking position
S_s	Arrays	10	Int	Ship operational status (0: waiting, 1: unloading, 2: completed)
Y_s	Arrays	10	Int	Location of storage of cargo loaded on the ship
B	Arrays	6	Boolean	Whether the berth is occupied
Y	Arrays	20	Boolean	Whether the yard is occupied
C_y	Arrays	20	Int	Type of cargo stored in yard space
W_s	Arrays	20	Int	Weight of cargo stored in yard space

(2) Action Space

To tackle the scheduling problem, the action space incorporates ship berthing and cargo storage yard allocations. Each action comprises two components: berth and yard. The former denotes the berth number representing the ship's docking location, while the latter indicates the yard number for cargo storage. The action space encompasses 10 ships, 6 berths, and up to 20 stacks, amounting to a total of 1200 possible actions. However, due to varying ship arrival times and the presence of docking and storage constraints, certain actions will be infeasible. Hence, a filtering process is necessary when designing the action space.

(3) Reward function

This paper belongs to the Multi-Objective Reinforcement Learning (MORL) problem, aiming to minimise the total dwell time of all ships in port and the aggregate cargo transportation cost. Each objective corresponds to a distinct reward function. Consequently, the overall reward consists of a collection of individual objective vectors. When objectives are directly correlated (e.g., minimised time or cost), MORL can be transformed into a single-objective RL problem through linear weighting. In reality, objectives frequently feature conflicts or constraints, requiring selective optimisation or trade-offs between conflicting objectives [27]. Both objectives in this study pertain to the minimisation of total time or cost. Thus, a linear weighted approach is adopted to formulate the reward function.

The designed reward function is expressed by Eq. 1:

$$R = -k(T + S) - l(D \times W) + C \tag{1}$$

Here, k and l denote the weights of the two objectives. After empirical investigation, k is set to 0.7 and l to 0.3. T represents the ship's operation time, calculated by dividing the weight of loaded cargo by the average operation speed of the berth corresponding to the cargo type. S signifies the ship's waiting time, determined by subtracting the arrival time from the commencement of operations. D captures the total cargo distance from berth to yard. W stands for cargo weight, and C is an adjustment value. Positive rewards are bestowed on reasonable actions, whereas penalties are applied through negative reward functions for suboptimal choices. To encourage intelligent agents to select legitimate actions, penalties for illegitimate actions slightly exceed positive rewards, thus fostering effective learning.

By crafting a well-designed reward function, intelligent agents are incentivised to choose appropriate actions while avoiding improper selections, leading to enhanced learning outcomes.

2.2 Berth and yard scheduling approach based on PS-D3QN

Reinforcement learning models for berth and yard scheduling in bulk ports entail managing substantial state variables and action decisions. Moreover, their state and action spaces exhibit considerable complexity, necessitating the employment of the DQN algorithm for approximating high-dimensional states. Derived from the DQN algorithm, Double DQN and Dueling DQN are advanced techniques that address its limitations. Double DQN overcomes overestimation issues by estimating target network Q-values using the action selected based on current evaluation network Q-values. On the other hand, Dueling DQN enhances stability by decoupling action-value functions through modifying neural network structure and achieving more accurate Q-value estimation.

This paper introduces a real-time scheduling approach, termed PS-D3QN, for berths and yards, based on an enhanced DQN algorithm. PS-D3QN integrates the Q-value estimation methodology of Double DQN and the concept of action-value function separation from Dueling DQN, synergising their strengths to enhance algorithm performance. Additionally, algorithm performance is further improved through the refinement of Prioritised Experience Replay (PER) and Softmax strategies.

In the PS-D3QN framework proposed in this study, the action value function is decomposed into a combination of state value V and action advantage function A, enabling a more valuable assessment of actions. The state value reflects the current state, while the action advantage function measures the disparity between current action performance and average performance. Actions that outperform the average yield a positive advantage function, while others yield a negative advantage function. Given a fixed Q, countless combinations of V and A can generate Q. Consequently, restrictions are imposed on A; typically, the average of the advantage function A for the same state is constrained to 0. Thus, the action value function is calculated as shown in Eq. 2:

$$Q(s_t, a_t) = V(s_t) + \left(A(s_t, a_t) - \frac{1}{|A|} \sum_{a_t} A(s_t, a_t) \right) \tag{2}$$

In PS-D3QN, the maximum action value from the evaluation network is used to calculate Q-values in the target network. The target network's value function is derived according to Eq. 3, where θ_t and θ_t^- denote the parameters of the evaluation and target networks, respectively.

$$Y_t^{PS-D3QN} = R_{t+1} + \gamma Q' \left(S_{t+1}, \underset{a}{argmax}\, Q(S_{t+1}, a; \theta_t) ; \theta_t' \right) \tag{3}$$

The primary objective of PS-D3QN is to train parameters that minimise the loss function, formulated in Eq. 4.

$$L^{PS-D3QN}(\theta_t) = E\left(\left(Y_t^{PS-D3QN} - Q_t(S_t, a; \theta_t) \right)^2 \right) \tag{4}$$

Following loss function computation, PS-D3QN employs stochastic gradient descent to update training parameters, transferring them to the target network parameters as illustrated in Eq. 5.

$$\theta_{t+1} = \theta_t + \alpha E \left(Y_t^{PS-D3QN} - Q_t(S_t, a; \theta_t) \frac{\partial Q_t(S_t, a; \theta_t)}{\partial \theta_t} \right) \tag{5}$$

The neural network architecture of the PS-D3QN algorithm, presented in Fig. 3, is constructed based on the MDP model established in Section 2.1.

PS-D3QN employs two networks: the evaluation network and the target network. Their structures are identical, featuring an input layer, two fully connected hidden layers, and an output layer. The ReLU function serves as the neuron activation function. Input consists of the state set, with output comprising the action set. The second fully connected layer separately outputs state values and action advantage functions, combining to present individual actions and their respective Q-values.

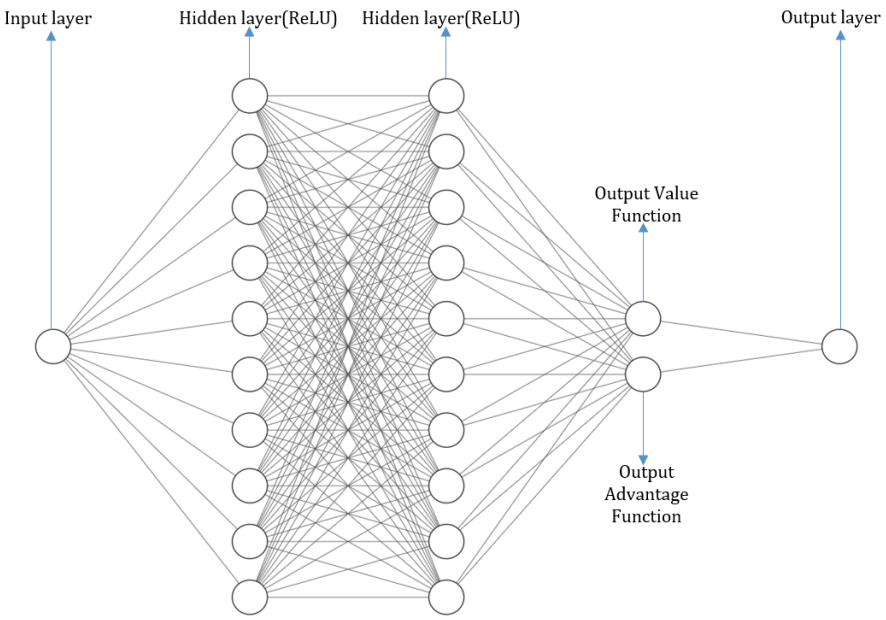

Fig. 3 Neural network structure of PS-D3QN

The PS-D3QN algorithm proposed in this study optimises its performance using the Prioritised Experience Replay (PER) mechanism, assigning priority to each experience based on Time Difference Error (TD-error). By favouring experiences with higher priority, the algorithm is inclined to select them for training. To avoid an excessive focus on high-priority experiences, this study introduces priority sampling weights for experience extraction, enhancing experience utilisation and promoting effective training and convergence. The experience priority in the algorithm, shown in Eq. 6, is influenced by TD-error and the number of completed tasks. This approach facilitates selecting valuable data for training, accelerating learning, and improving performance.

$$P = |Q_t - Q_c| + \frac{N_t}{N_t + 1/\sigma} \tag{6}$$

In Eq. 6, P represents the experience priority, Q_t stands for the target Q value, Q_c corresponds to the current Q value, N_t denotes the current number of completed tasks, σ signifies the weight, which progressively increases with the number of iterations, eventually reaching a final value of 0.01. Within this paper, experience priority is influenced by TD-error and the number of completed tasks. Consequently, the algorithm tends to favour more valuable data during training, enhancing the reuse of pivotal experiences. This approach accelerates the learning process and enhances the algorithm's overall performance.

The PS-D3QN method proposed here replaces the ε-greedy strategy with the Softmax strategy from the DQN algorithm. Softmax is a common technique for balancing exploration and exploitation in reinforcement learning, selecting actions based on a probability distribution derived from each action's estimated average reward. This strategy encourages frequent selection of actions with higher average rewards while still exploring other actions through non-zero probabilities. Softmax, governed by the Boltzmann distribution, allocates probabilities to action selection based on the estimated average reward. As depicted in Eq. 7:

$$P(k) = \frac{e^{\frac{Q(k)}{\tau}}}{\sum_{i=1}^{K} e^{\frac{Q(i)}{\tau}}} \tag{7}$$

In Eq. 7, $P(k)$ denotes the probability of selecting action k, $Q(k)$ represents the estimated average reward value for action k based on historical data, and $Q(i)$ records the average reward value after the current action's completion. τ, referred to as the "temperature", influences the

trade-off between exploration and exploitation. Lower τ values emphasise exploitation, while higher values encourage exploration. In this study, τ is set using hyperbolic annealing. Initially, a higher temperature is employed to promote exploration, gradually reducing as experience accumulates to encourage utilisation of well-performing actions. The temperature update process is governed by Eq. 8, where τ_0 is the initial temperature and τ_k controls the annealing rate.

$$\tau(i) = \frac{\tau_0}{1 + \tau_k i} \tag{8}$$

The overall flow of the PS-D3QN algorithm is presented in Fig. 4.

During PS-D3QN algorithm training, the environment is initialised with yard storage information, berth occupancy data, ship states, and cargo details. Based on the current state, the Softmax strategy is used to select scheduling actions for berths and yards. Throughout the training, the algorithm sequentially stores experiences derived from interacting with the environment in the experience replay pool. Once sufficient experiences are collected, the PS-D3QN algorithm conducts random and priority sampling from the experience replay pool based on priority sampling weights.

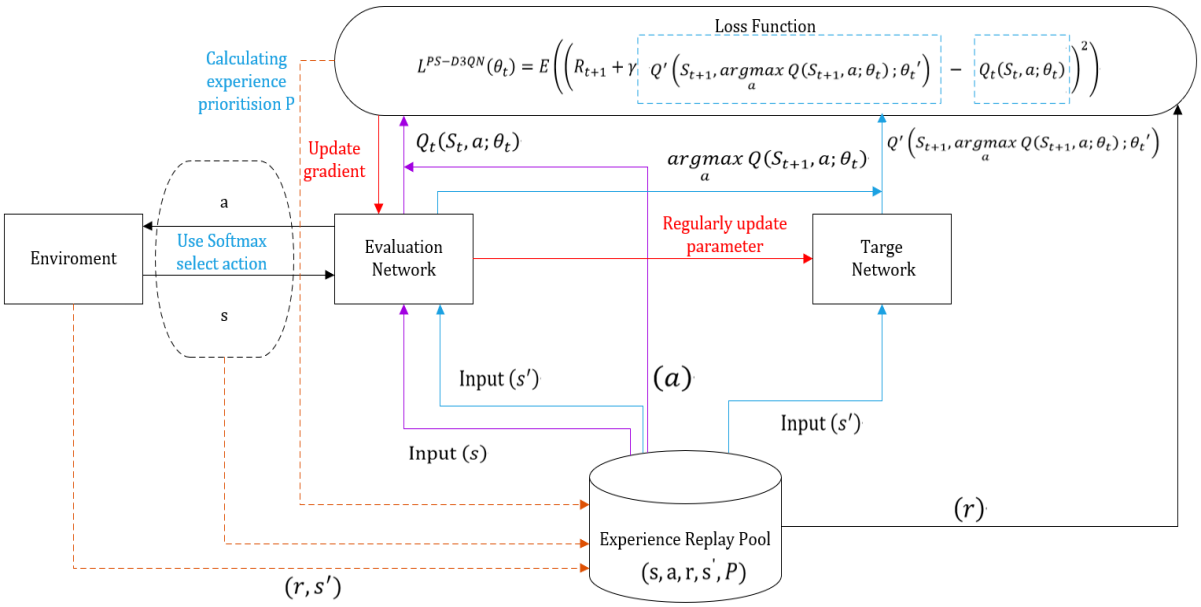

Fig. 4 PS-D3QN algorithm flowchart

3. Experimental results and discussion: A case study for bulk cargo terminal

The PS-D3QN algorithm proposed in this paper is deployed on a cloud server. The relevant environment configuration includes Windows Server 2022, Python 3.8, and PyTorch 1.12. The processor is a 16-core Intel Xeon (Ice Lake) Platinum 8369B, the GPU is NVIDIA Tesla A100 (80GB video memory), and the memory is 125GB.

This paper presents a case study on ship unloading operations in the import business of a port in southern China. In this section, we validate the proposed PS-D3QN algorithm using dynamic ship arrival data for a single planning period. The selected ship arrival data comprehensively covers most cargo types, ensuring good representativeness and generalisation. Based on this data, the PS-D3QN algorithm is trained and its results are compared and analysed against the Double DQN algorithm, Dueling DQN algorithm, and the actual scheduling scheme. The algorithm's relevant parameters are presented in Table 2.

The arriving ship data, berth data, and yard data used in this study are provided in Tables 3, 4, and 5. After conducting numerical tests, the simulated reward function curve and simulated loss function curve of the PS-D3QN algorithm, Double DQN algorithm, and Dueling DQN algorithm are illustrated in Fig. 5 and Fig. 6.

Table 2 Parameter settings

Parameter	Description	value
α	Learning Rate	0.001
γ	Discount Factor	0.99
Replay buffer size	Experience replay pool capacity	10000
Batch size	Batch size of samples per training	32
Tau	Rate at which the target network copies weights from the evaluation network	0.001
episode	Maximum number of steps per training round	500
Alpha_Prioritise	Weights for prioritised sampling	0.6
τ_0	Softmax initial temperature	200
τ_k	Softmax annealing speed	0.01

Table 3 Arriving ship data

Serial number	Ship name	Length	Depth	Cargo type	Cargo name	Cargo weight	Estimated arrival time(data prepro-cessing)
1	Ship 1	112.21	9	grain	cassava	11531	0d 13h
2	Ship 2	158.8	10	coal	coal	24376	0d 7h
…							
9	Ship 9	166.31	9	steel	steel	13015	0d 16h
10	Ship 10	149.18	10	grain	soya	21825	1d 23h

Table 4 Berth data

Serial number	Berth	Length	Depth	Operational speed of ore (tonnes per hour)	Operational speed of steel (tonnes per hour)	Operational speed of coal (tonnes per hour)	Operational speed of grain (tonnes per hour)
1	Berth 1	181	9	300	450	0	0
2	Berth 2	192	9	350	600	0	0
…							
5	Berth 5	201	10.5	240	0	110	180
6	Berth 6	202	10.5	360	0	500	210

Table 5 Yard data

Serial number	Yard	Cargo type	Yard type	Yard capacity	Horizontal relative position
1	Position 1 in district 1	Coal	Outside	25000	2
2	Position 2 in district 1	Coal	Outside	25000	2
3	Position 3 in district 1	Steel	Outside	20000	2
…					
19	Position 1 in district 7	Grain	Warehouse	20000	8
20	Position 2 in district 7	Grain	Warehouse	20000	8

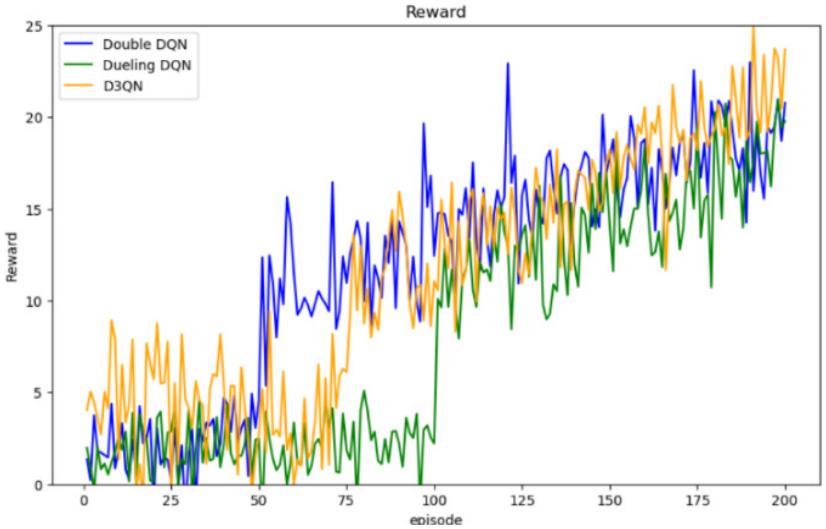

Fig. 5 Simulation reward function curve

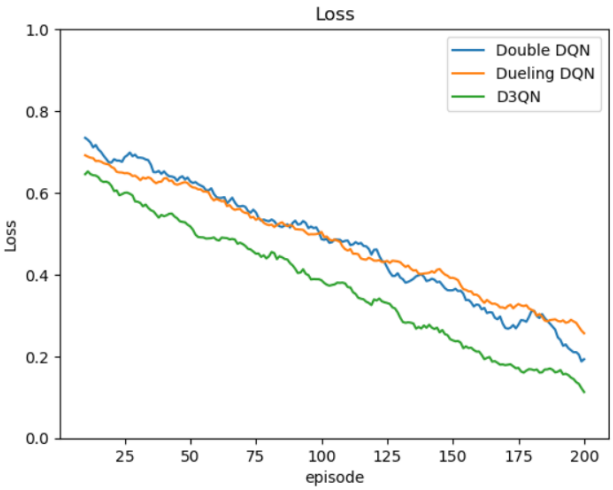

Fig. 6 Simulation reward loss curve

As shown in Fig. 5, the Double DQN algorithm quickly identifies a relatively high reward value in the initial iterations due to mitigating the overestimation problem. However, as training progresses, it might get trapped in a local optimum. Dueling DQN exhibits slower convergence because of its complex network structure, yet its decoupled action-value function ensures more accurate final computation results. The PS-D3QN algorithm proposed in this paper maintains faster convergence, stability, and superior scheduling outcomes by combining the strengths of both algorithms.

Fig. 6 portrays the decreasing trend in the simulated loss function curves for the three algorithms. The PS-D3QN algorithm's curve exhibits a smoother decrease compared to the other two algorithms. From the start to the end of iterations, its loss function gradually decreases, converging towards 0 with a relatively swift convergence rate.

To summarise, Double DQN's quick convergence is attributed to alleviating the overestimation problem. However, it can get trapped in a local optimum, leading to convergence on a locally optimal strategy. Dueling DQN's slower learning process due to increased network complexity, is associated with greater fluctuations and inadequate stability despite eventual convergence. The PS-D3QN algorithm, as proposed, excels in convergence speed and stability. It efficiently discovers maximum reward values, leveraging the combination of the other two algorithms to address overestimation issues while enhancing Q-value estimation through action value function splitting, thereby streamlining the training process. Incorporating the Prioritised Experience Replay mechanism and the Softmax action selection strategy optimises computational efficiency and stability. These enhancements facilitate faster convergence while overcoming local optima issues.

Following numerical experiments, the final scheduling results of the three algorithms and the outcomes of the actual scheduling scheme are presented in Table 6.

The PS-D3QN algorithm, Double DQN algorithm, and Dueling DQN algorithm each improve scheduling scheme efficiency by 12.85 %, 9.18 %, and 8.93 %, respectively, compared to the actual scheduling scheme. The scheduling scheme derived from the PS-D3QN algorithm effectively reduces ships' port dwell time, thereby laying the groundwork for subsequent ship arrivals. Furthermore, the scheme contributes to significant reductions in total cargo transportation costs, enhancing ship loading and unloading efficiency and subsequently reducing overall port operating costs.

Table 6 Scheduling results

Scheduling scheme	Total ship dwell time (hours)	Total costs of cargo transport (ten thousand tonnes multiplied by metres)
PS-D3QN	437	27.6
Double DQN	449	30
Dueling DQN	441	31.9
Actual scheduling scheme	464	39

The complex nature of bulk berth and yard scheduling, characterised by extensive state and action spaces, is effectively addressed by the PS-D3QN algorithm proposed in this study. The algorithm leverages deep neural networks for nonlinear modelling and approximation. By integrating the strengths of the Double DQN and Dueling DQN algorithms and optimising via the Prioritised Experience Replay mechanism and the Softmax action selection strategy, the PS-D3QN algorithm not only maintains swift convergence but also exhibits stability, effectively navigating potential local convergence issues while demonstrating notable learning and generalisation capabilities within the context of the current problem.

The PS-D3QN algorithm proposed in this study can also be applied to dynamic scheduling challenges in other fields, such as production and manufacturing, after reconfiguring the MDP model for specific problems with extensive discrete action and state spaces. It demonstrates commendable learning and generalisation capabilities in such scenarios.

4. Conclusion

This study presents a novel real-time scheduling approach called PS-D3QN based on the improved DQN algorithm. This method amalgamates the meritorious aspects of the Double DQN and Dueling DQN algorithms and employs two dueling neural networks. It adeptly gauges the Q-value of the target network by virtue of the action elected through the Q-value of the currently evaluating network. Moreover, the optimisation has been further finetuned by the ingenious design of a rational Prioritised Experience Replay (PER) mechanism and the integration of a Softmax action selection strategy. Additionally, this study examined the berth and yard scheduling predicaments prevalent in bulk cargo terminals. It has crafted an MDP model specifically for the scheduling issue, with the overarching goal of minimising both the cumulative time ships remain in the port and the total cost associated with cargo transportation. This was achieved by ingeniously amalgamating the authentic port milieu and pertinent data to configure the MDP model's state space, action space, and reward function.

Employing the PS-D3QN algorithm to address the scheduling conundrums based on actual ship, berth, and yard data yielded commendable optimization outcomes. In contrast with existing scheduling strategies and two alternative deep reinforcement learning algorithms, the PS-D3QN algorithm, as proposed in this study, has exhibited a substantial enhancement in the efficacy of berth and yard scheduling in port operations. Furthermore, it has contributed to the reduction of operational costs for ports while simultaneously mitigating the inherent empirical bias that arises from manual scheduling.

In the realm of future research endeavours, there exists the potential to elevate the algorithm's performance and stability through the refinement of the neural network architecture and the strategic selection of fitting optimisers. It is noteworthy that the intricacies of loading and unloading processes within bulk cargo ports involve a broader spectrum of resources. This study's scope was delimited to the berth and yard allocation facets of scheduling optimisation. The algorithm in this study holds the promise of expansion and applicability, potentially extending to more intricate challenges. By extending the model, PS-D3QN can address other resource scheduling problems at bulk cargo terminals, such as machinery and equipment scheduling. Furthermore, the algorithm can be applied to scheduling problems in production and manufacturing fields by re-establishing the MDP model, such as the job shop scheduling problem and its extensions.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant No. 52172311) and China State Railway Group Co., Ltd. (Grant No. L2021X001).

References

[1] United Nations Conference on Trade and Development (2022). *Review of Maritime Transport 2022*, United Nations Publications, New York, USA, doi: 10.18356/9789210021470.

[2] Bouzekri, H., Alpan, G., Giard, V. (2023). Integrated laycan and berth allocation problem with ship stability and conveyor routing constraints in bulk ports, *Computers & Industrial Engineering*, Vol. 181, Article No. 109341, doi: 10.1016/j.cie.2023.109341.

[3] Bierwirth, C., Meisel, F. (2010). A survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research*, Vol. 202, No. 3, 615-627, doi: 10.1016/j.ejor.2009.05.031.

[4] Bierwirth, C., Meisel, F. (2015). A follow-up survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research*, Vol. 244, No. 3, 675-689, doi: 10.1016/j.ejor.2014.12.030.

[5] Đelović, D., Medenica Mitrović, D. (2017). Some considerations on berth productivity referred on dry bulk cargoes in a multipurpose seaport, *Tehnički Vjesnik – Technical Gazette,* Vol. 24, Supplement 2, 511-519, doi: 10.17559/TV-20150226074034.

[6] Barros, V.H., Costa, T.S., Oliveira, A.C.M., Lorena, L.A.N. (2011). Model and heuristic for berth allocation in tidal bulk ports with stock level constraints, *Computers & Industrial Engineering*, Vol. 60, No. 4, 606-613, doi: 10.1016/j.cie.2010.12.018.

[7] Umang, N., Bierlaire, M., Vacca, I. (2013). Exact and heuristic methods to solve the berth allocation problem in bulk ports, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 54, 14-31, doi: 10.1016/j.tre.2013.03.003.

[8] Ribeiro, G.M., Mauri, G.R., Beluco, S.d.C., Lorena, L.A.N., Laporte, G. (2016). Berth allocation in an ore terminal with demurrage, despatch and maintenance, *Computers & Industrial Engineering*, Vol. 96, 8-15, doi: 10.1016/j.cie.2016.03.005.

[9] Ernst, A.T., Oğuz, C., Singh, G., Taherkhani, G. (2017). Mathematical models for the berth allocation problem in dry bulk terminals, *Journal of Scheduling*, Vol. 20, No. 5, 459-473, doi: 10.1007/s10951-017-0510-8.

[10] Cheimanoff, N., Fontane, F., Kitri, M.N., Tchernev, N. (2021). A reduced VNS based approach for the dynamic continuous berth allocation problem in bulk terminals with tidal constraints, *Expert Systems with Applications*, Vol. 168, Article No. 114215, doi: 10.1016/j.eswa.2020.114215.

[11] Tang, L., Sun, D., Liu, J. (2016). Integrated storage space allocation and ship scheduling problem in bulk cargo terminals, *IIE Transactions*, Vol. 48, No. 5, 428-439, doi: 10.1080/0740817X.2015.1063791.

[12] Rocha de Paula, M., Boland, N., Ernst, A.T., Mendes, A., Savelsbergh, M. (2019). Throughput optimisation in a coal export system with multiple terminals and shared resources, *Computers & Industrial Engineering*, Vol. 134, 37-51, doi: 10.1016/j.cie.2019.05.021.

[13] Robenek, T., Umang, N., Bierlaire, M., Ropke, S. (2014). A branch-and-price algorithm to solve the integrated berth allocation and yard assignment problem in bulk ports, *European Journal of Operational Research*, Vol. 235, No. 2, 399-411, doi: 10.1016/j.ejor.2013.08.015.

[14] Unsal, O., Oguz, C. (2019). An exact algorithm for integrated planning of operations in dry bulk terminals, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 126, 103-121, doi: 10.1016/j.tre.2019.03.018.

[15] Tian, W., Zhang, H.P. (2021). A dynamic job-shop scheduling model based on deep learning, *Advances in Production Engineering & Management*, Vol. 16, No. 1, 23-36, doi: 10.14743/apem2021.1.382.

[16] François-Lavet, V., Henderson, P., Islam, R., Bellemare, M.G., Pineau, J. (2018). An introduction to deep reinforcement learning, *Foundations and Trends® in Machine Learning*, Vol. 11, No. 3-4, 219-354, doi: 10.1561/2200000071.

[17] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D. (2015). Human-level control through deep reinforcement learning, *Nature*, Vol. 518, 529-533, doi: 10.1038/nature14236.

[18] Van Hasselt, H., Guez, A., Silver, D. (2016). Deep reinforcement learning with double q-learning, In: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence,* Palo Alto, California, USA, 12-17, doi: 10.1609/aaai.v30i1.10295.

[19] Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., Lanctot, M., de Freitas, N. (2016). Dueling network architectures for deep reinforcement learning, In: *Proceedings of the 33rd International Conference on Machine Learning*, Brookline, USA, 1995-2003, doi: 10.48550/arXiv.1511.06581.

[20] Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D. (2015). Continuous control with deep reinforcement learning, *Machine Learning,* doi: 10.48550/arXiv.1509.02971.

[21] Sivamayil, K., Rajasekar, E., Aljafari, B., Nikolovski, S., Vairavasundaram, S., Vairavasundaram, I. (2023). A systematic study on reinforcement learning based applications, *Energies*, Vol. 16, No. 3, Article No. 1512, doi: 10.3390/en16031512.

[22] Lee, R.-Y., Chai, T.-Y., Chua, S.-Y., Lai, Y.-L., Wai, S.Y., Haw, S.-C. (2022). Cashierless checkout vision system for smart retail using deep learning, *Journal of System and Management Sciences*, Vol. 12, No. 4, 232-250, doi: 10.33168/JSMS.2022.0415.

[23] Kim, H.-J., Madhavi, S. (2022). A reinforcement learning model for quantum network data aggregation and analysis, *Journal of System and Management Sciences*, Vol. 12, No. 1, 283-293, doi: 10.33168/JSMS.2022.0120.

[24] Li, R., Zhang, X, Jiang, L, Yang, Z, Guo, W. (2022). An adaptive heuristic algorithm based on reinforcement learning for ship scheduling optimization problem, *Ocean & Coastal Management*, Vol. 230, Article No. 106375, doi: 10.1016/j.ocecoaman.2022.106375.
[25] Dai, Y., Li, Z., Wang, B. (2023). Optimizing berth allocation in maritime transportation with quay crane setup times using reinforcement learning, *Journal of Marine Science and Engineering*, Vol. 11, No. 5, Article No. 1025, doi: 10.3390/jmse11051025.
[26] Li, C., Wu, S., Li, Z., Zhang, Y., Zhang, L., Gomes, L. (2022). Intelligent scheduling method for bulk cargo terminal loading process based on deep reinforcement learning, *Electronics*, Vol. 11, No. 9, Article No. 1390, doi: 10.3390/electronics11091390.
[27] Hayes, C.F., Rădulescu, R., Bargiacchi, E., Källström, J., Macfarlane, M., Reymond, M., Verstraeten, T., Zintgraf, L.M., Dazeley, R., Heintz, F., Howley, E., Irissappane, A.A., Mannion, P., Nowé, A., Ramos, G., Restelli, M., Vamplew, P., Roijers, D.M. (2022). A practical guide to multi-objective reinforcement learning and planning, *Autonomous Agents and Multi-Agent Systems*, Vol. 36, No. 26, 1-59, doi: 10.1007/s10458-022-09552-y.