Deep Reinforcement Learning for Carrier-borne Aircraft Support Operation Scheduling

Haifeng Feng* and Wei Zeng
Artificial Intelligence and Automation School, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan 430074, China.
*Corresponding Author Email: haifeng_f@hust.edu.cn

***Abstract*—The makespan of support operations of carrier-borne aircraft is a key factor affecting the sortie generation rate. The support operation process involves multiple support resources and operational tasks should satisfy serial and parallel constraint relationships. The effective coordination of these processes can be considered as a multi-resource constrained multi-project scheduling problem (MRCMPSP), which is a complex NP-hard problem. In this paper, a deep reinforcement learning (RL) method is designed to solve the problem, including the image representation of the state, the definition of action mapping, and reward function. Deep convolution neural network and advantage actor-critic algorithm (A2C) are utilized to provide a new solution to the scheduling problem, and experimental results show that the effectiveness of the proposed algorithm.**

I. Introduction

Carrier-borne aircraft is the most important weapon equipment on an aircraft carrier. A series of support operations such as inspection, refueling, inflation, and weapon mounting must be completed before taking off. The makespan of the support operations of carrier-borne aircraft is a key factor affecting the sortie generation rate [1,2].

The carrier-borne aircraft support operation scheduling is a multi-resource constrained multi-project scheduling problem (MRCMPSP) and its main objective is to shorten the makespan of support operations. The research of aircraft support operation scheduling problem mainly focuses on computational simulation and intelligent optimization methods, such as genetic algorithm [3,4,5], particle swarm optimization [6,7], tabu search [8], simulated annealing [9,10], and other algorithms [11,12]. But these algorithms are suitable for specific scheduling problems and do not have the generalization ability.

In this paper, the aircraft support scheduling problem is modeled as Markov decision processes (MDP). To represent the state of resource utilization, an image representation is proposed. Based on this, a convolutional neural network structure is designed, and an aircraft support operation scheduling algorithm based on Advantage Actor-Critic (A2C) [13] is proposed. The algorithm optimizes the makespan of aircraft support operations by focusing on recent resource allocation. The simulation results show that the proposed method can effectively solve the aircraft support operation scheduling problem.

II. Problem Definition and Modeling

A. Problem Description

The scheduling problem of aircraft support operations is a kind of MRCMPSP. Three kinds of resource constraints are considered, including support equipment, support operators, and operating space (station space and cockpit space). Each aircraft needs to complete 10 kinds of support tasks. The task execution process is shown in figure 1.

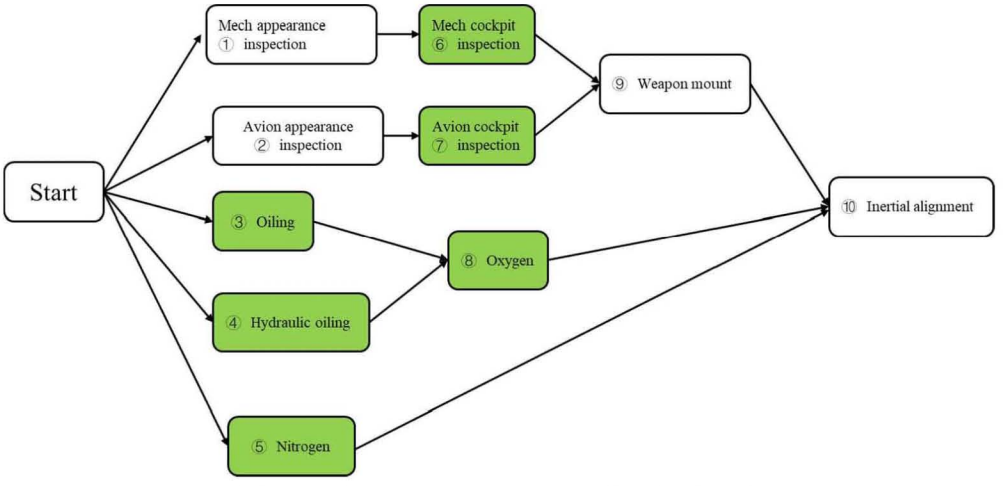

Figure 1. The support task flow chart of each carrier aircraft

978-1-6654-3730-1/21/$31.00 ©2021 IEEE
DOI 10.1109/ICAA53760.2021.00169

2021 International Conference on Intelligent Computing, Automation and Applications (ICAA) | 978-1-6654-3730-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICAA53760.2021.00169

Figure 1 shows the resource requirements for each task. Each task requires support operators and a task with the green color also requires support equipment. In the paper, the number of ①, ②, … is used to refer to a certain task for simplicity.

B. Notation

- Sets

I set of aircraft, $i = 1, \cdots, I$
J_i set of tasks for aircraft i, $j = 0, 1, \cdots, J_i$
J_i^* set of cockpit tasks for aircraft i
Kp set of support operators
Kr set of support equipment
P_{ij} set of pre-constrained tasks for task j of aircraft i
$A_i(t)$ set of running tasks for aircraft i at t moment

- Parameters

d_{ij} duration of task j for aircraft i
S_{ij} start time of task j for aircraft i
c_i finish time of all tasks for aircraft i
C_{max} support operation project duration
T upper bound of the support operation duration
L station space limit of the number of the task executed in parallel
rp_{ij} number of support operator required for task j of aircraft i
re_{ij} number of support equipment required for task j of aircraft i

- Decision variables

X_{pijk}: 1 if operator k is allocated to task j for aircraft i; 0 otherwise
X_{eijl}: 1 if equipment l is allocated to task j for aircraft i; 0 otherwise
x_{ijt}: 1 if task j for aircraft i has been finished at the t moment; 0 otherwise

C. Basic Formulation about Objective Function

Objective Function:

$$\min F = \min(C_{max}) \tag{1}$$

Constrains:

$$E_{ij} = S_{ij} + d_{ij}, \forall i \in I, \forall j \in J_i \tag{2}$$

$$c_i = \max(E_{ij}), \forall i \in I, \forall j \in J_i \tag{3}$$

$$C_{max} = \max(c_i), \forall i \in I \tag{4}$$

$$S_{ij} \geq S_{ih} + d_{ih}, \forall (i,h) \in P_{ij}, \forall i \in I, \forall j \in J_i \tag{5}$$

$$\sum_{t=0}^{T} x_{ijt} = 1, \forall i \in I, \forall j \in J_i \tag{6}$$

$$\sum_{i=1}^{m} \sum_{j \in A_i(t)} rp_{ij} \leq |Kp|, \forall t \in [0,T] \tag{7}$$

$$\sum_{i=1}^{m} \sum_{j \in A_i(t)} re_{ij} \leq |Kr|, \forall t \in [0,T] \tag{8}$$

$$|A_i(t)| \leq L, \forall i \in I, t \in [0,T] \tag{9}$$

$$\sum_{j \in A_i(t) \wedge j \in J_i^*} 1 \leq 1, \forall i \in I, t \in [0,T] \tag{10}$$

$$\sum_{k \in Kp} Xp_{ijk} = rp_{ij}, \forall i \in I, \forall j \in J_i \tag{11}$$

$$\sum_{l \in Ke} Xe_{ijl} = re_{ij}, \forall i \in I, \forall j \in J_i \tag{12}$$

$$Xp_{ijk} + Xp_{i'j'k} \leq 1, \forall i \in I, j \in A_i(t), i' \in I - i, j' \in A_{i'}(t), k \in Kp \quad (13)$$

$$Xe_{ijl} + Xe_{i'j'l} \leq 1, \forall i \in I, j \in A_i(t), i' \in I - i, j' \in A_{i'}(t), l \in Kr \quad (14)$$

$$(Xp_{ijk}, Xe_{ijl}, x_{ijt} \in \{0,1\}, \forall i \in I, \forall j \in J_i, \forall k \in Kp, \forall l \in Ke, \forall t \in [0,T] \quad (15)$$

Constraint (2) defines the relationship between the finish time of each task with the start time; Constraints (3) and (4) define the makespan; Constraint (5) expresses the constraints of aircraft support operation process flow; Constraint (6) expresses that each task is scheduled only once; Constraints (7) and (8) represent the resource constraints of the supply resource capacity; Constraints (9) and (10) represent the station space capacity; Constraints (11) and (12) indicate the allocation matches task required; Constraints (13) and (14) indicate the same resource can only be used for one task at the same time; Constraint (15) expresses the decision variable constraints, all of which are $0-1$ integer variables.

III. SOLUTION DESIGN

A. MDP Formulation

The MDP model is used to describe the problem, including the definition of state S, action A, and reward R.

State-space: This paper designs the image representation to describe the state S. To simplify the representation, the discussion is divided into two parts. The resource features are shown by the resource utilization of support operator Kp and support equipment Kr and the resource requirements of tasks to be assigned. The time window TW is introduced to show the time horizon. The resource representation of the image is shown in figure 2.

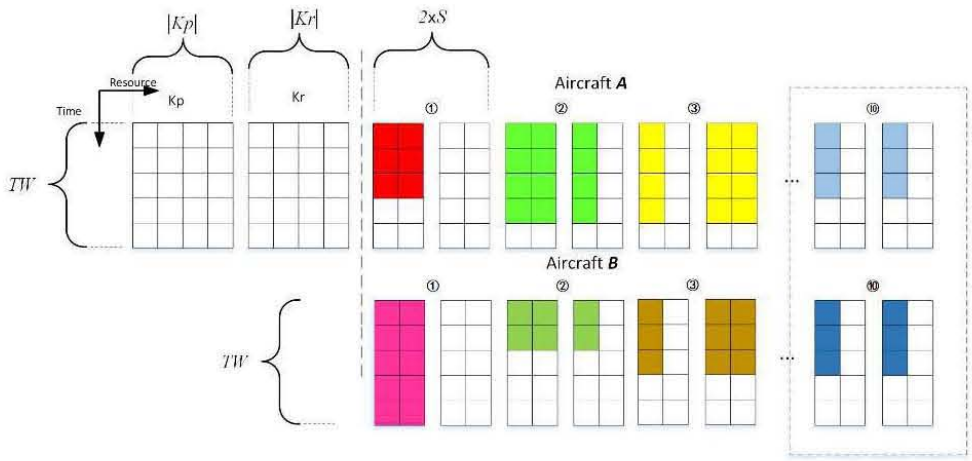

Figure 2. Resource image features

The state when task execution time $t = 0$ is shown in Figure 2, the time window is $t \leq t' \leq t + 5$. The left side of the dotted line indicates the utilization of Kp and Kr, the white square indicates that the resource is idle, each scale represents a unit of resources. The right side of the dotted line shows the resource requirements of the support operations of two aircraft. Different aircraft are stacked vertically sorted by number. Each support operation has two large boxes for two kinds of resource requirements. Different colors are set to distinguish different tasks. For example, the red square in the figure indicates that it needs two units of Kp and takes up three units of time continuously.

There are two kinds of spatial resource constraints: deck station and aircraft cockpit. L is the space constraint of a deck station, L^* is the space constraint of an aircraft cockpit, and S^* is the amount of space occupied during the execution of the task. These two images are stacked by adding channels. The size of the whole image is determined by the larger one. Because $S^* < S$, $L < K$, $L^* < R$, the width of the whole image is $|Kp| + |Kr| + (2 * S) * 10$, and the height is m*TW. To distinguish different tasks, gray value $c = 10 * (i-1) + j$ is used to represent the task of j of aircraft i, where 10 is the number of support tasks of a single aircraft.

Action space: figure 3 illustrates the mapping relationship between tasks and actions.Since the number of support tasks of a single aircraft is 10, $1 \sim 10$ are set to the action number interval of the first aircraft's tasks, and $11 \sim 20$ are set to the action number interval of the next aircraft's tasks in the figure. It is necessary to consider the constraint of execution order between tasks, e.g. action set $\{6,7,\ldots,10\}$, shown in the black color part in the figure, cannot be selected at the current time. The increment of task execution time is the minimum of the remaining time of the currently assigned task,

$$\Delta t = \min\left(d_i^{'}\right)\ \forall i \in A'(t) \qquad (16)$$

Where $A'(t)$ is the set of currently unfinished tasks, $d_i^{'}$ is the remaining execution time of task i. The time window dynamically moves forward with the time increment.

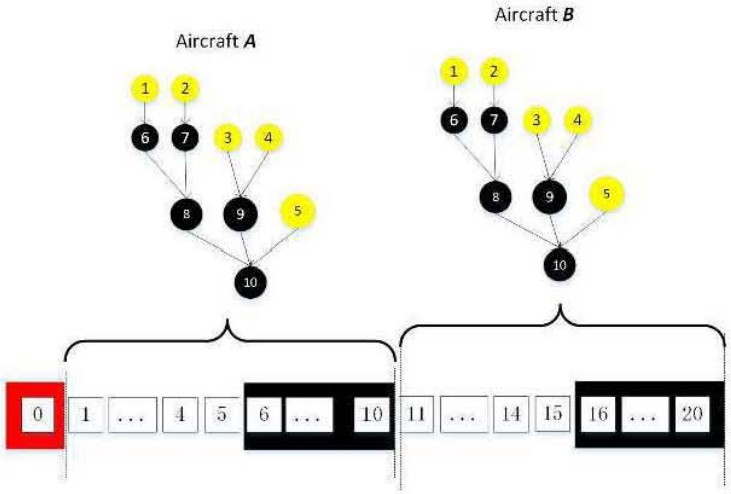

Figure 3. Action mapping relationship

Reward function: The objective of RL is to maximize the cumulative rewards, while the objective of the aircraft support operation scheduling problem is to minimize the makespan of support operations. The reward function is set as a function of action and task execution time. The reward function is as follows:

$$R(s,a) = -\Delta t \qquad (17)$$

When the action will cause the time to increase, return the negative value of the current time increment, else return 0. If the discount rate $\gamma = 1$, the reverse value of the accumulated reward will be consistent with the makespan of support operations.

B. Training Algorithm

In this paper, the reinforcement learning A2C algorithm is used to train the agent. The Actor is the policy network $\pi_\theta(a_t|s_t)$, which estimates the probability distribution of the action by observing the current state s_t. The Critic is the value network $V_{\theta_v}(s_t)$, which is used to guide the update of the policy network.

Policy network update parameters by the expression $\nabla_\theta \log \pi_\theta(a_t|s_t)A(s_t,a_t,\theta,\theta_v)$, where the advantage function $A(s_t,a_t,\theta,\theta_v)$ is computed according to the following expression:

$A(s_t,a_t,\theta,\theta_v) = \sum_{i=0}^{k-1}\gamma^i r(s_{t+i},a_{t+i}) + V_{\theta_v}(s_{t+k}) - V_{\theta_v}(s_t)$,

where k is the length of the learning fragment, which is set to half of the average sequence length. Value network Critic updates its parameters by minimizing the mean square error of the predicted value $V_{\theta_v}(s_t)$ and the real value $\sum_{i=0}^{k-1}\gamma^i r(s_{t+i},a_{t+i}) + V_{\theta_v}(s_{t+k})$.

The inputs of Actor and Critic are state s_t, and the output is a policy $\pi_\theta(a_t|s_t)$ and the value function $V_{\theta_v}(s_t)$. In the initial stage of training, we train Critic first, then train Actor to make Critic efficiently estimate the value function $V_\pi(s_t)$ and update the policy network. This process iterates in the A2C training.

C. Network Architecture

To process the state represented by image features, a convolution neural network is designed. The convolution network is composed of three convolution layers and three full connection layers. Its overall structure is shown in figure 4.

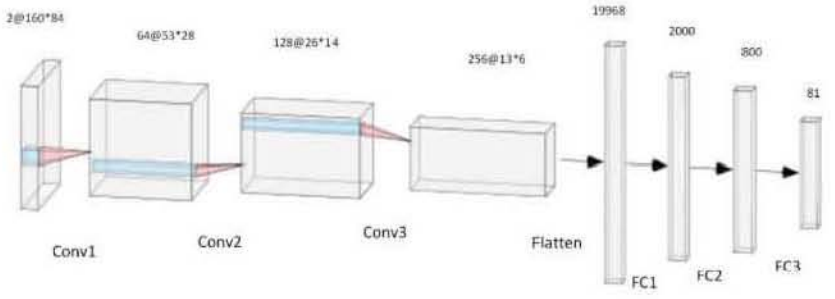

Figure 4. Convolution network structure

The convolution kernel sizes of convolution layers Conv1, Conv2 and Conv3 are 6x6, 4x4, and 4x4, respectively, and the number of channels is 64, 128, and 256, respectively. Three fully connected layers FC1, FC2, and FC3 are connected, and the output is the size of action space. Relu is adopted as the nonlinear activation function of each layer.

The Critic network needs to add an output layer of neurons, and the Actor-network needs to add 81 neurons and the Softmax activation function is adopted.

IV. EXPERIMENT EVALUATION

This section evaluates the effectiveness of the proposed algorithm and compares it with four algorithms, including Random, First-Fit (FF), Best-Fit (BF), and DQN, in a simulation environment. All the algorithms are based on the MDP designed in this paper.

A. Experiment Setup

There are 8 aircraft and 8 support stations in our experiment environment. Each aircraft occupies one support station. There are 12 Kp and 12 Kr, and the maximum number of tasks that each station can perform at the same time is 2. In addition to the verification algorithm, the simulation environment is used for DQN and A2C algorithm training. Different from the test scenario, the training process uses randomly generated data to make the deep neural network have the ability to extract scheduling strategy directly from image features. In the experiment, Python is used as the programming language, and PyTorch 1.6.0 is used as the deep learning framework.

The hyperparameter setting in DQN and A2C are listed as follows. In DQN, the exploration step is set to 900, the minimum exploration rate is 0.01, the total number of training steps is set to $500k$, the exploration rate decays to the minimum at $250k$ steps, the size of experience playback unit D is 10000, the initial learning rate of the network is $\alpha = 0.02$, the learning rate decay is 0.98, the learning batch size is 32, the target network is updated every 200 steps of training, the gradient clipping value is 10, the reward value is reduced to 10 times to prevent the TD error from being too large, and the reward discount is set to 0.99. In A2C, the length of the learning interception segment t_{max} is set to 50, the total number of network updates $step_{total}$ is set to $10k$, each update here uses a multi-step learning method, the reward discount rate is 0.99, the initial learning rate of the network $\alpha = 0.01$, the decay of learning rate is 0.96, the training optimizer is set to Adam, and the gradient clipping value is 20; the reward value is reduced to 10 times. The trained network is used to solve the experimental case.

The resource requirement and execution duration of each type of task are listed in table 1 and 2.

TABLE 1. RESOURCE REQUIREMENT OF EACH TYPE OF TASK

	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
Kp	2	2	2	1	1	1	1	1	3	2
Kr	0	0	2	3	1	1	2	2	0	0

TABLE 2. EXECUTION DURATION OF EACH TASK (MIN)

	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
A	4	2	3	4	5	3	3	4	3	4
B	2	2	3	4	5	3	3	4	3	4
C	2	3	3	4	5	3	3	4	3	4
D	2	2	3	5	5	3	4	4	4	4
E	2	2	5	4	5	4	3	4	3	4
F	2	2	3	4	5	3	3	4	3	4
G	2	2	3	4	2	3	3	4	3	4
H	2	2	4	4	5	3	3	5	3	4

B. *Result Analysis*

According to the aforementioned experimental settings, the following experimental results are obtained.

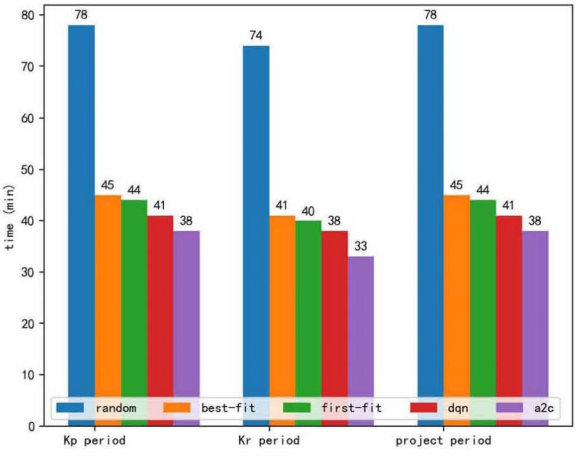

Figure 5. Comparisons of experimental results

Figure 5 shows the comparisons of the resource utilization duration and the makespan of the five algorithms, where the makespan is the maximum of Kp and Kr resource utilization durations. The results show that A2C algorithm is best in the makespan of support operations.

V. CONCLUSION AND FUTURE WORK

In this paper, a carrier-bourne aircraft support scheduling algorithm based on deep learning is proposed, which aims to optimize the makespan of support operations considering the resource constraints in the scheduling process. To represent the state of resource utilization, an image representation is proposed. To avoid excessive image size, a dynamic time window method is proposed. The advantage of the method is that the image features only focus on the recent resource allocation. Compared with Random, First-Fit, Best-Fit, and DQN, the scheduling algorithm proposed in the paper is best in terms of makespan and resource utilization efficiency.

There are some limitations to the present study that may require future research. The image features designed in this article cannot express the execution order of tasks. In the future, we may consider embedding the representation method of graph structure to solve the scheduling problem with a complex network structure between tasks.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant 71871100.

REFERENCE

[1] Binheng Y, Yuquan B, Biao Z et al. 2016 Ship Electron Eng. 36 8

[2] RYAN J C, CUMMINGS M L, ROY N. 2011 Designing an interactive local and global decision support system for aircraft carrier deck scheduling. Proceedings of AIAA Information Technology.

[3] Yuan P, Han W, Su X, et al. 2018 Appl Sci Technol. 8 9

[4] Elena P, Marta P, Antolín L. 2015 Soft Computing. 20 8

[5] Yuyang L. 2017 Research on Genetic Algorithm of the Security Personnel Allocation Optimization. Dissertation for Ph.D. Degree. (Harbin: Harbin Institute of Technology) p 12

[6] Weichao S Wei H, Weiwei S. 2012 Acta Aeronautica et Astronautica Sinica. 33 11

[7] Weichao S, Wei H, Yan S et al. 2013 Application Research of Computers. 30 2

[8] Menglong L, Minhui Y. 2018 Chin J Ship Res. 13 5

[9] Weichao S, Wei H, Wei X et al. 2016 J Syst Eng Electron. 38 10

[10] Dapeng B, Tiantian L, Qinchao M et al. 2014 Appl Sci Technol. 33 09

[11] Wei H, Xichao S, Junfeng C. 2015 J Syst Eng Electron. 37 4

[12] Wei H, Rongwei C, Xichao S. 2021 Control and Decision. 3 13

[13] Volodymyr M, Adri`a P, Mehdi M et al. 2016 Asynchronous methods for deep reinforcement learning. CoRR, abs/1602.01783