Independent double DQN-based multi-agent reinforcement learning approach for online two-stage hybrid flow shop scheduling with batch machines

Ming Wang , Jie Zhang , Peng Zhang , Li Cui

Journal of Manufacturing Systems (2022)

IIE8557-01 고등강화학습

경영과학연구실 김지원

online two-stage hybrid flow shop scheduling with batch machines

- most of the existing methods for solving the BHFSP have assumed a static manufacturing environment
- For the OBHFSP, different kinds of jobs with due date constraint arrive dynamically bringing the demand of rapid response
- It is important to handle online batch forming tasks and online scheduling tasks

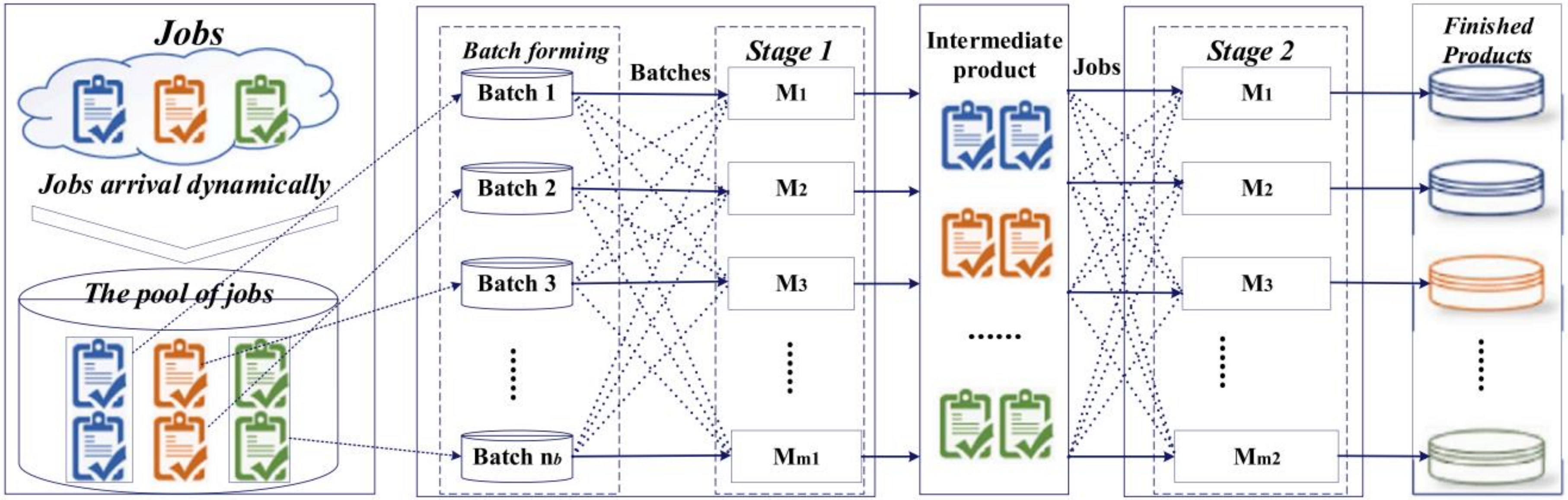


Fig. 1. The abridged general view of the OBHFSP.

Two-stage hybrid flow shop scheduling with batch machines and jobs

Key Idea

- Independent double deep-q-network-based multi-agent reinforcement learning (MA-IDDQN)

  1) Two-stage hybrid flow shop scheduling with batch machines and jobs arriving over time is formulated as a MDP
  2) Multi-agent reinforcement learning approach is proposed to produce an adaptive rule for batch forming and scheduling

Problem formulation

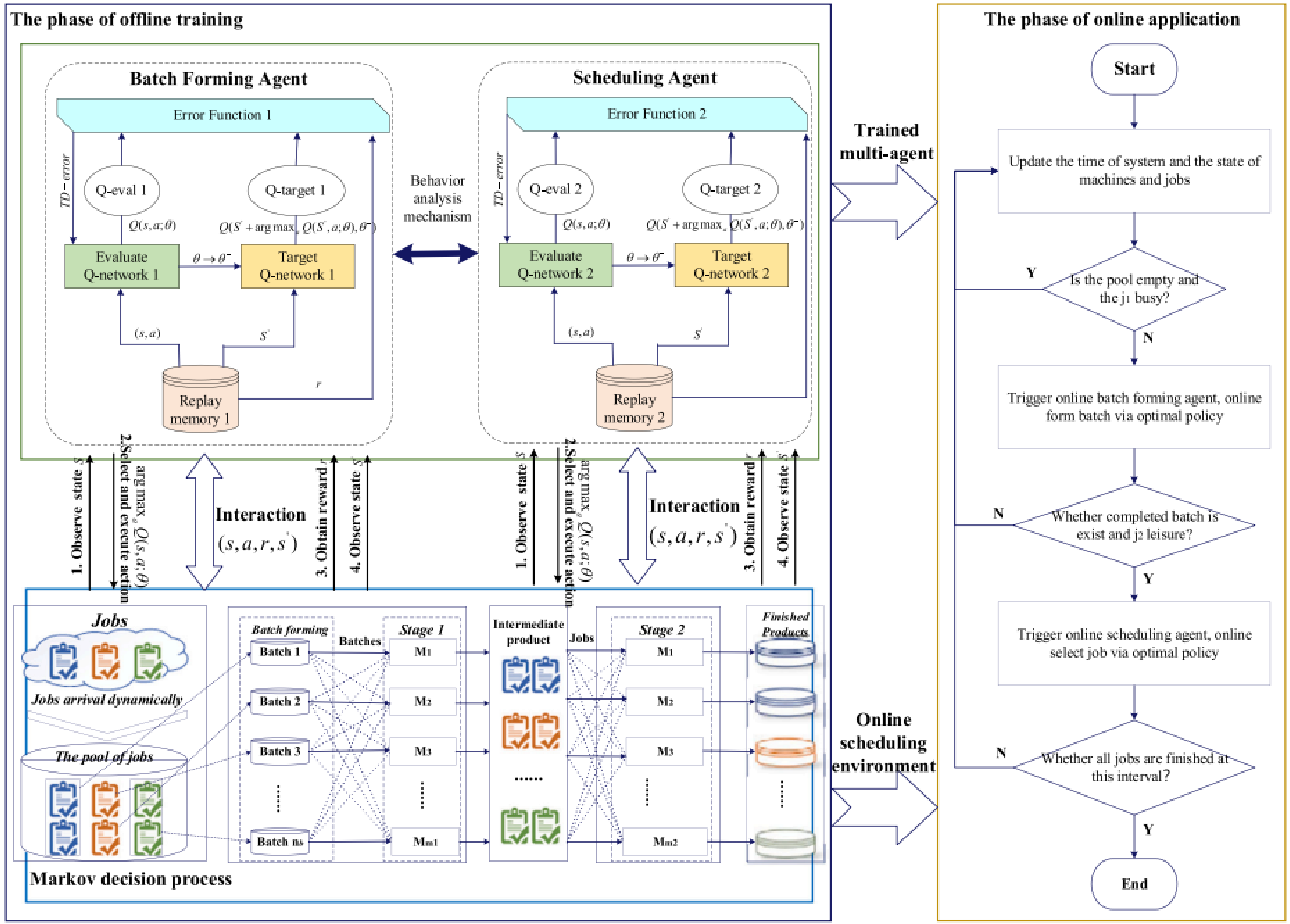
- OBHFSP can be formulated as a mixed integer linear programming model

$$TT_{min} = \sum_{i=1}^{n_{total}} \max(0, (e_{ij_2} - d_i))$$

Constraint 1) one job can only be grouped into one batch
Constraint 2)one batch can only be processed on one batch machine at stage 1
Constraint 3) one job can only be processed on one parallel machine at stage 2
Constraint 4) all machines are available during all periods.
Constraint 5)every batch should be processed on the batch machines, followed by the processing on parallel machines at stage 2
Constraint 6) one batch must be grouped by jobs with the same type and meet the capacity constraint of the batch machine
Constraint 7) sequence-independent setup time is required between batches with different types
Constraint 8) jobs cannot be added or removed while one batch is grouped
Constraint 9) there is unlimited buffer capacity between the two stages
Constraint 10) due date of each job

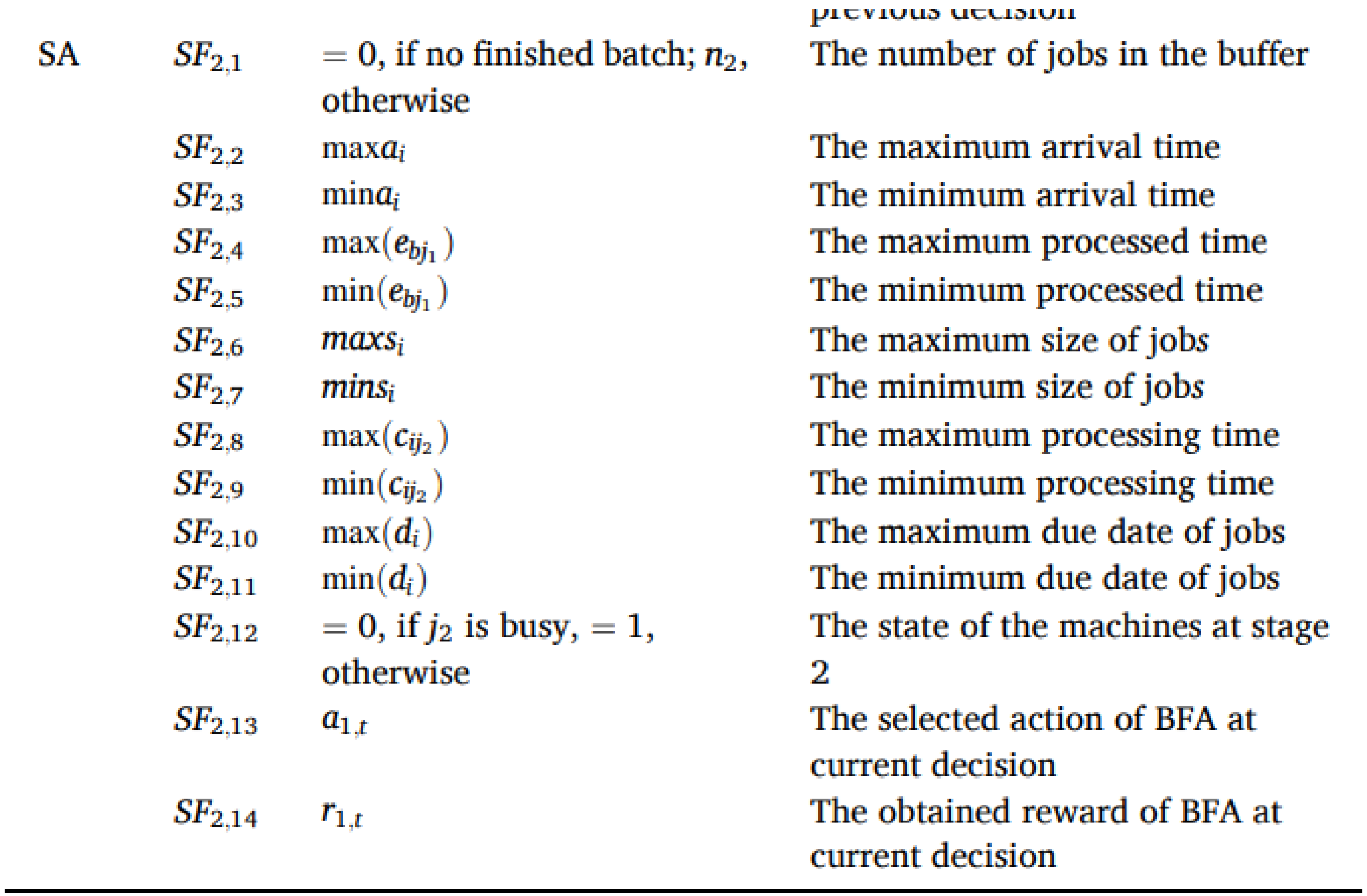
MA-IDDQN for OBHFSP

- During the offline training phase, the OBHFSP is modeled as a sequential decision-making problem
- Two agents: batch forming /scheduling



The scheduling environment

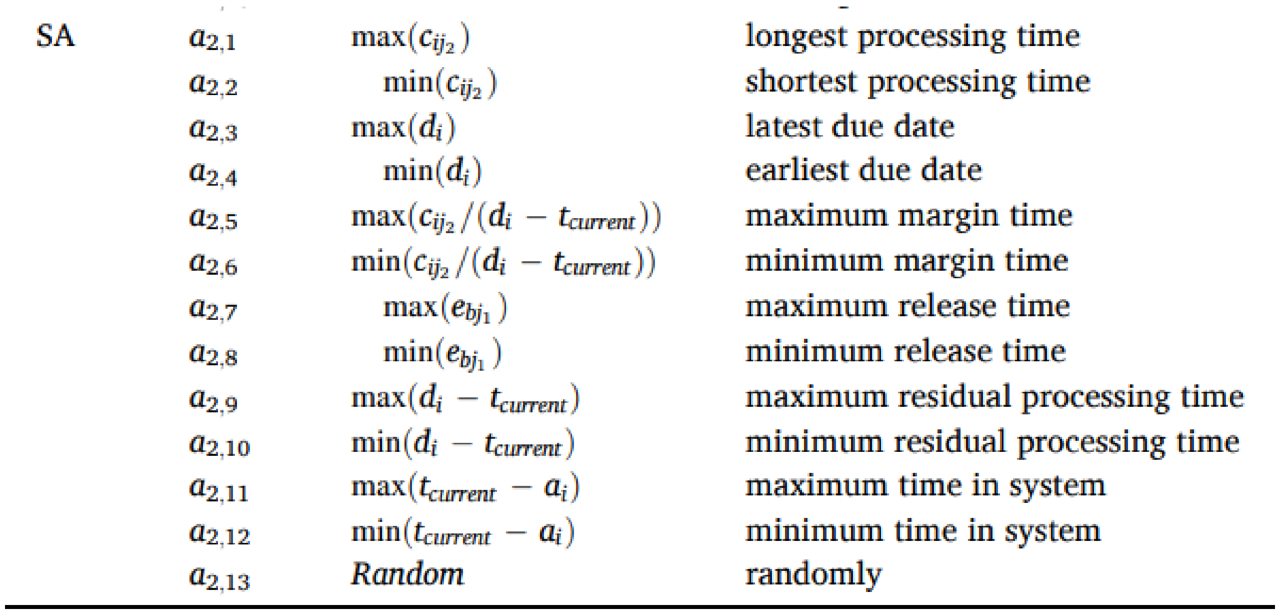
1) State spaces



			previous decision
SA	$SF_{2,1}$	$= 0$, if no finished batch; n_2, otherwise	The number of jobs in the buffer
	$SF_{2,2}$	$\max a_i$	The maximum arrival time
	$SF_{2,3}$	$\min a_i$	The minimum arrival time
	$SF_{2,4}$	$\max(e_{bj_1})$	The maximum processed time
	$SF_{2,5}$	$\min(e_{bj_1})$	The minimum processed time
	$SF_{2,6}$	$maxs_i$	The maximum size of jobs
	$SF_{2,7}$	$mins_i$	The minimum size of jobs
	$SF_{2,8}$	$\max(c_{ij_2})$	The maximum processing time
	$SF_{2,9}$	$\min(c_{ij_2})$	The minimum processing time
	$SF_{2,10}$	$\max(d_i)$	The maximum due date of jobs
	$SF_{2,11}$	$\min(d_i)$	The minimum due date of jobs
	$SF_{2,12}$	$= 0$, if j_2 is busy, $= 1$, otherwise	The state of the machines at stage 2
	$SF_{2,13}$	$a_{1,t}$	The selected action of BFA at current decision
	$SF_{2,14}$	$r_{1,t}$	The obtained reward of BFA at current decision

Scheduling Agent

2) Action spaces
: to select heuristic rule adaptively



SA	$a_{2,1}$	$\max(c_{ij_2})$	longest processing time
	$a_{2,2}$	$\min(c_{ij_2})$	shortest processing time
	$a_{2,3}$	$\max(d_i)$	latest due date
	$a_{2,4}$	$\min(d_i)$	earliest due date
	$a_{2,5}$	$\max(c_{ij_2}/(d_i - t_{current}))$	maximum margin time
	$a_{2,6}$	$\min(c_{ij_2}/(d_i - t_{current}))$	minimum margin time
	$a_{2,7}$	$\max(e_{bj_1})$	maximum release time
	$a_{2,8}$	$\min(e_{bj_1})$	minimum release time
	$a_{2,9}$	$\max(d_i - t_{current})$	maximum residual processing time
	$a_{2,10}$	$\min(d_i - t_{current})$	minimum residual processing time
	$a_{2,11}$	$\max(t_{current} - a_i)$	maximum time in system
	$a_{2,12}$	$\min(t_{current} - a_i)$	minimum time in system
	$a_{2,13}$	*Random*	randomly

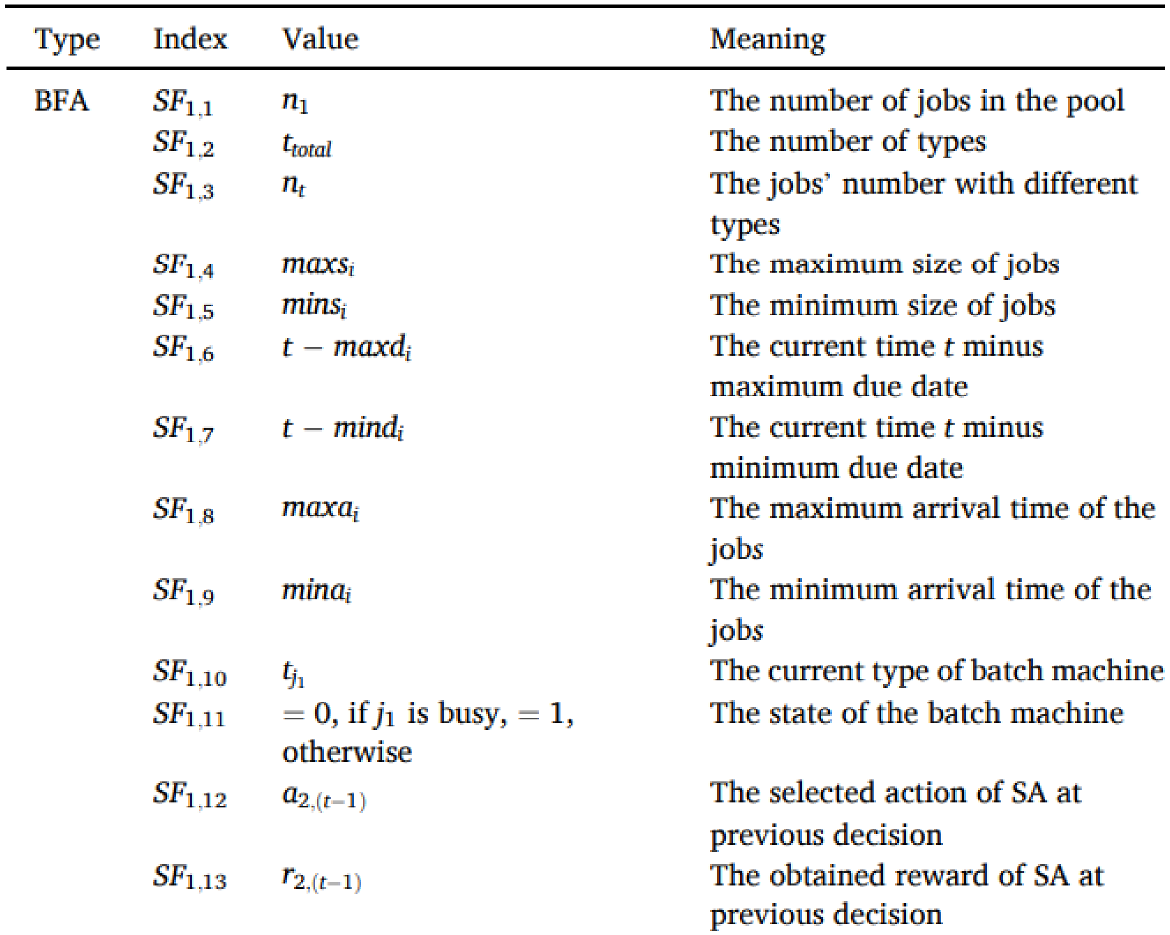
3) Reward Function

$$r_{2,t_s} = -\left(\sum_{i=1}^{n_{jr}} x_i[t_{s'} - \max(t_s, d_i)] + \sum_{i=1}^{n_{jp}} x_i[t'_{s'} - \max(t_s, d_i)]\right)$$

ts :current time, ts ' : the next decision time
xi = 1, if di < ts '; = 0, otherwise

Batch Forming Agent

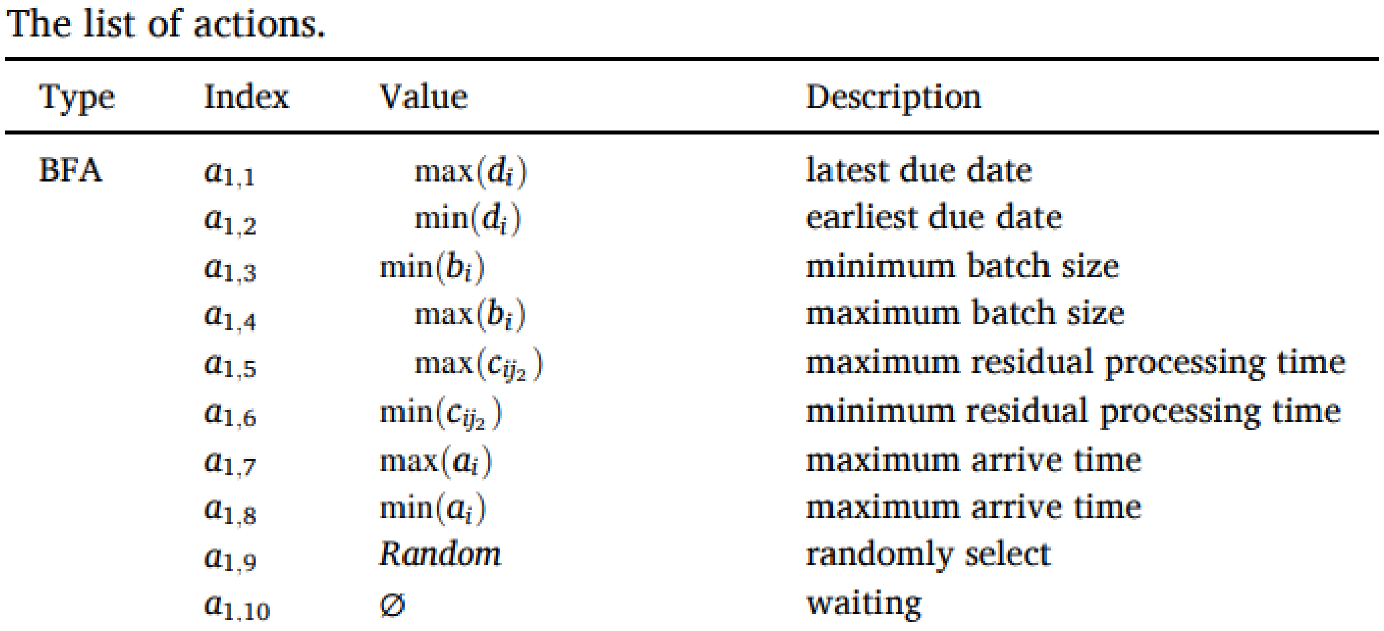
1) State spaces



Type	Index	Value	Meaning
BFA	$SF_{1,1}$	n_1	The number of jobs in the pool
	$SF_{1,2}$	t_{total}	The number of types
	$SF_{1,3}$	n_t	The jobs' number with different types
	$SF_{1,4}$	$maxs_i$	The maximum size of jobs
	$SF_{1,5}$	$mins_i$	The minimum size of jobs
	$SF_{1,6}$	$t - maxd_i$	The current time t minus maximum due date
	$SF_{1,7}$	$t - mind_i$	The current time t minus minimum due date
	$SF_{1,8}$	$maxa_i$	The maximum arrival time of the jobs
	$SF_{1,9}$	$mina_i$	The minimum arrival time of the jobs
	$SF_{1,10}$	t_{j_1}	The current type of batch machine
	$SF_{1,11}$	$= 0$, if j_1 is busy, $= 1$, otherwise	The state of the batch machine
	$SF_{1,12}$	$a_{2,(t-1)}$	The selected action of SA at previous decision
	$SF_{1,13}$	$r_{2,(t-1)}$	The obtained reward of SA at previous decision

Batch Forming Agent

2) Action spaces
: to select heuristic rule adaptively



The list of actions.

Type	Index	Value	Description
BFA	$a_{1,1}$	$\max(d_i)$	latest due date
	$a_{1,2}$	$\min(d_i)$	earliest due date
	$a_{1,3}$	$\min(b_i)$	minimum batch size
	$a_{1,4}$	$\max(b_i)$	maximum batch size
	$a_{1,5}$	$\max(c_{ij_2})$	maximum residual processing time
	$a_{1,6}$	$\min(c_{ij_2})$	minimum residual processing time
	$a_{1,7}$	$\max(a_i)$	maximum arrive time
	$a_{1,8}$	$\min(a_i)$	maximum arrive time
	$a_{1,9}$	*Random*	randomly select
	$a_{1,10}$	$\varnothing$	waiting

3) Reward Function

$$r_{1,t_s} = -(\sum_{i=1}^{n_{jp}} x_i[t_{s'} - \max(t_s, d_i)] + \sum_{i=1}^{n_{jg}} x_i[t'_{s'} - \max(t_s, d_i)])$$

ε-greedy policy considering waiting in BFA

- This paper designs a special action selection method including waiting
- When the number of waiting performed by the agent reaches the maximum number (w_tmax), the agent chooses the action with the second largest Q value or randomly selects

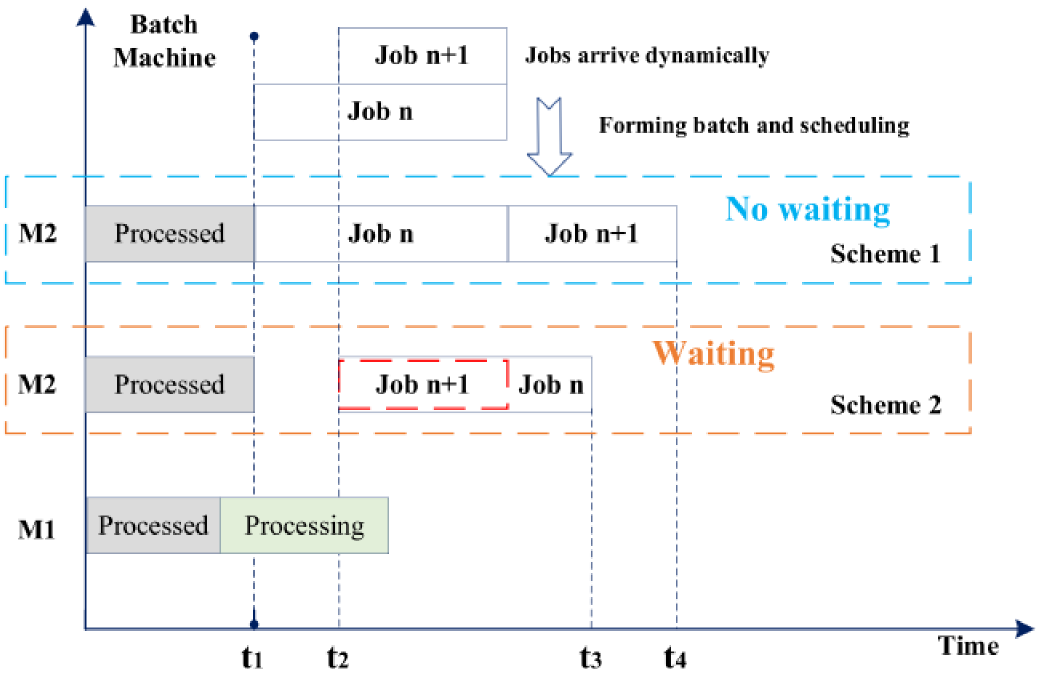
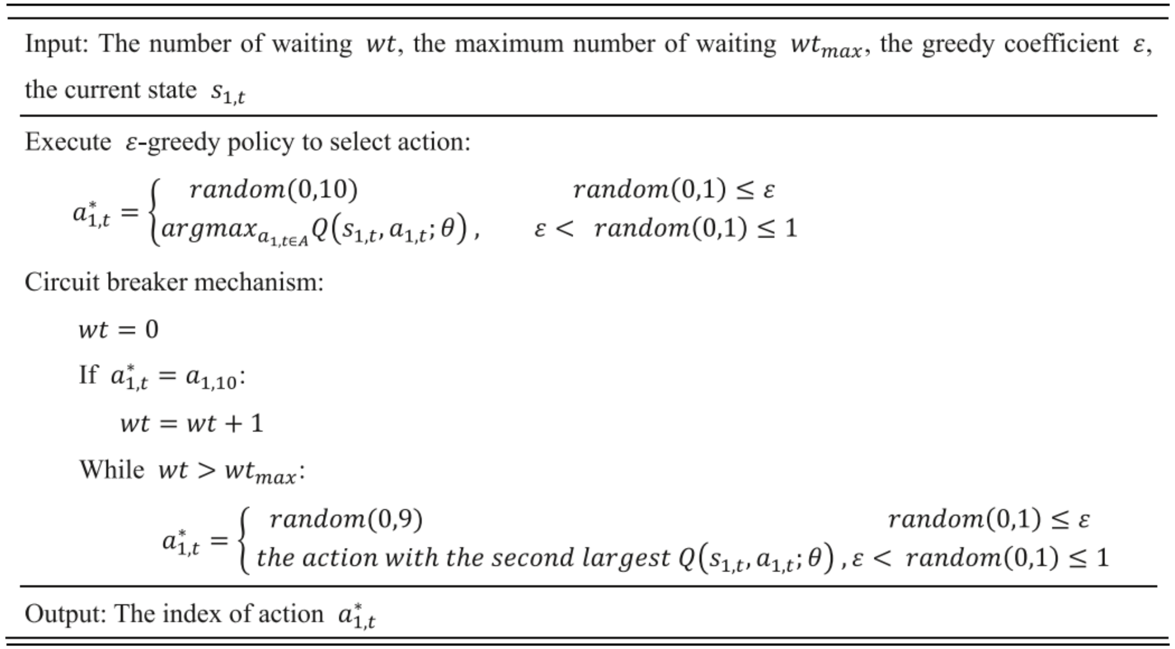
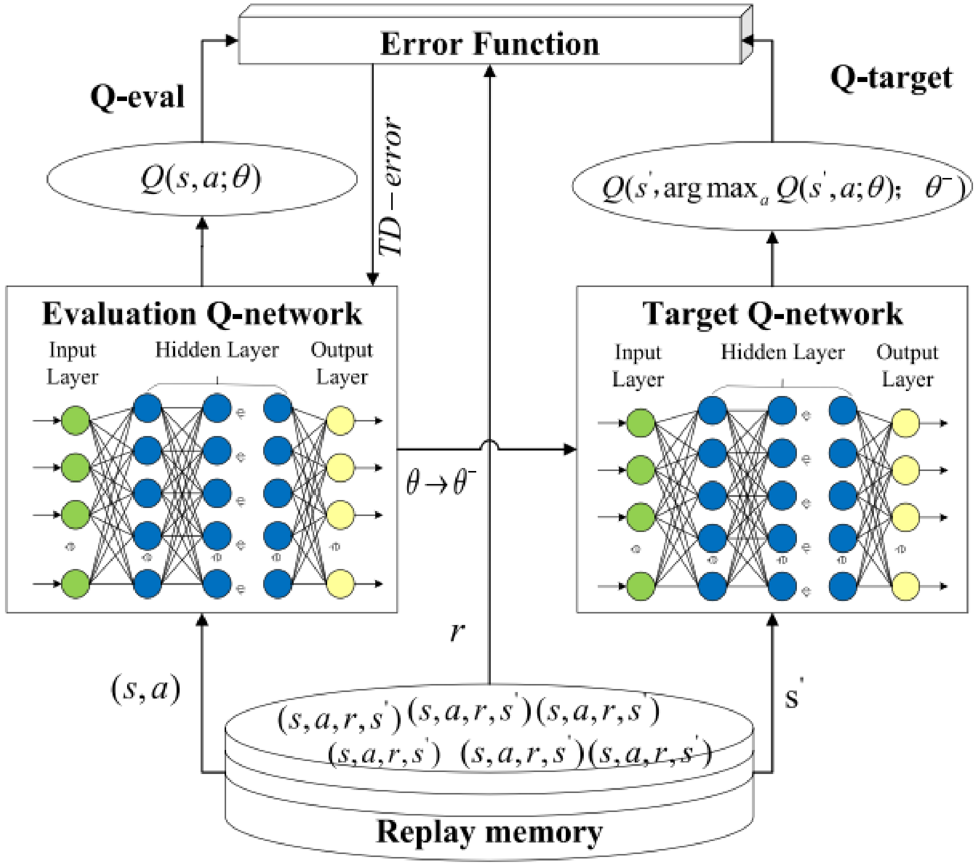


Fig. 3. The comparison between waiting and no waiting.



```
Input: The number of waiting $wt$, the maximum number of waiting $wt_{max}$, the greedy coefficient $\varepsilon$, the current state $s_{1,t}$
Execute $\varepsilon$-greedy policy to select action:
    $a^*_{1,t} = \begin{cases} random(0,10) & random(0,1) \le \varepsilon \\ argmax_{a_{1,t\epsilon A}} Q(s_{1,t}, a_{1,t}; \theta), & \varepsilon < random(0,1) \le 1 \end{cases}$
Circuit breaker mechanism:
    $wt = 0$
    If $a^*_{1,t} = a_{1,10}$:
        $wt = wt + 1$
    While $wt > wt_{max}$:
        $a^*_{1,t} = \begin{cases} random(0,9) & random(0,1) \le \varepsilon \\ \text{the action with the second largest } Q(s_{1,t}, a_{1,t}; \theta), & \varepsilon < random(0,1) \le 1 \end{cases}$
Output: The index of action $a^*_{1,t}$
```

Framework of the BFA and Scheduling agent



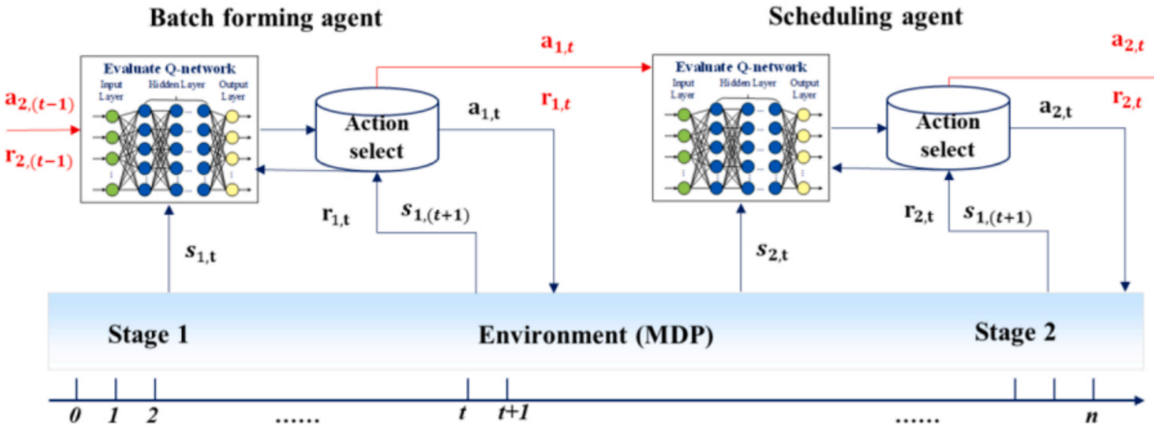
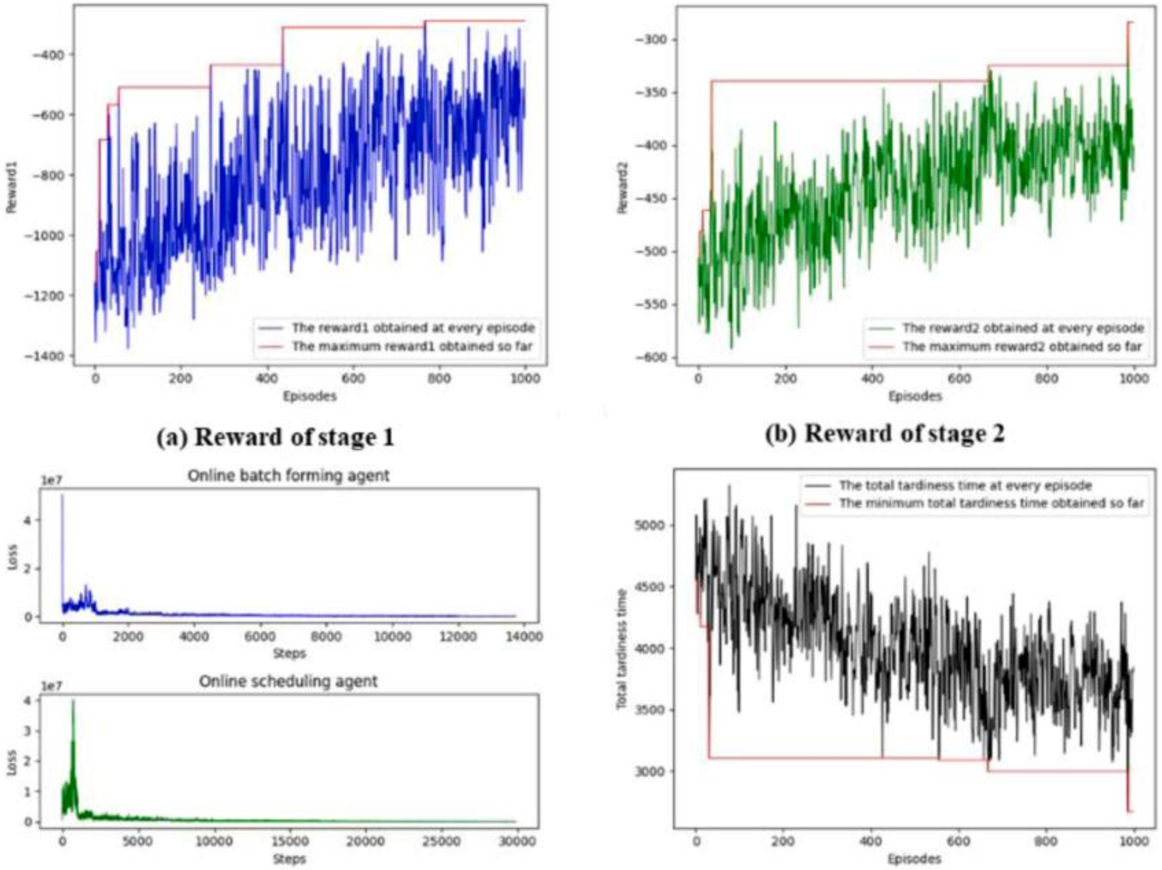


Fig. 5. The mechanism of behavior analysis.

The training process of multi-agent

- 27 instances with different settings



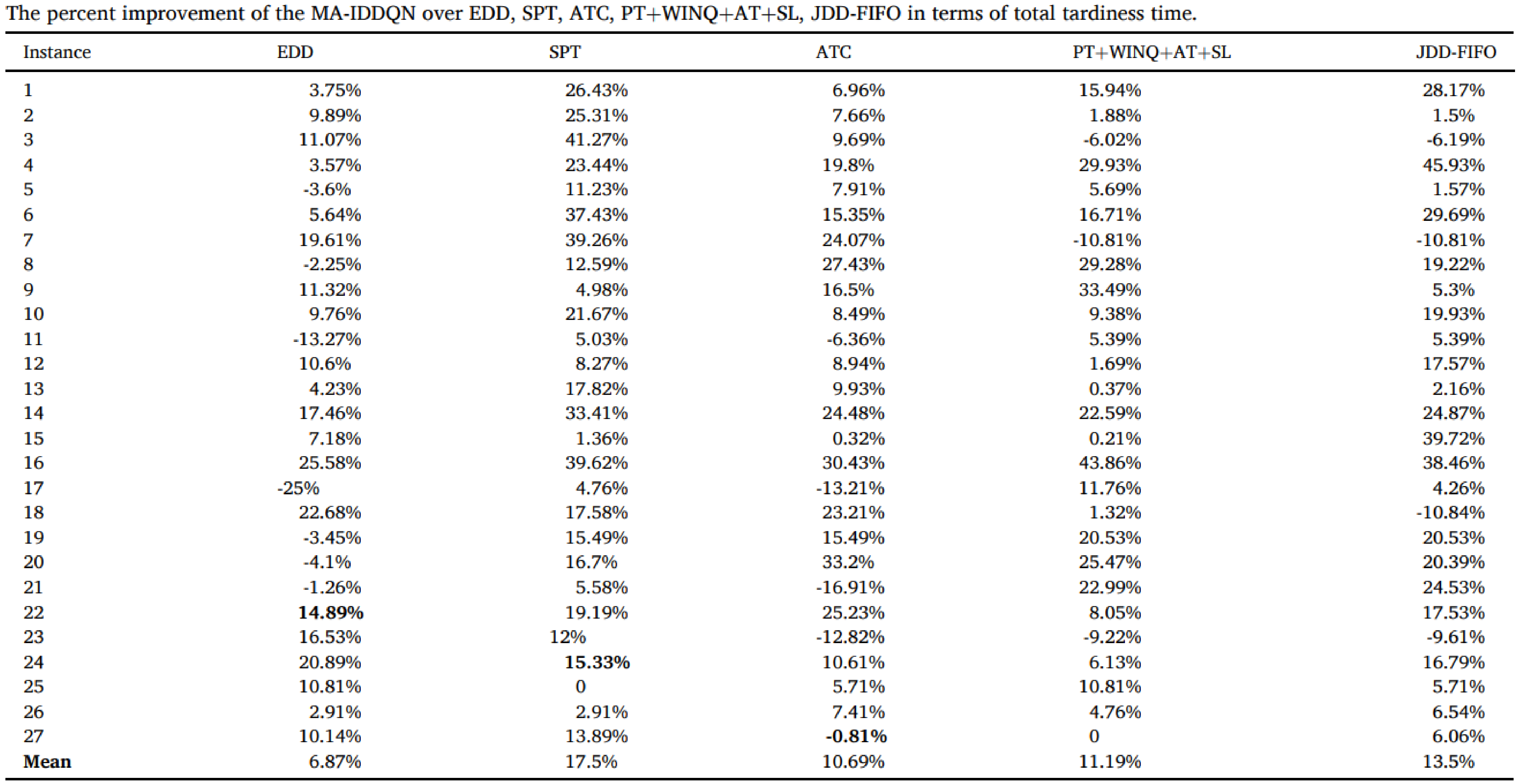
(a) Reward of stage 1

(b) Reward of stage 2

(c) The loss of agents

(d) Total tardiness time

Comparisons with frequently-used heuristic rules



The percent improvement of the MA-IDDQN over EDD, SPT, ATC, PT+WINQ+AT+SL, JDD-FIFO in terms of total tardiness time.

Instance	EDD	SPT	ATC	PT+WINQ+AT+SL	JDD-FIFO
1	3.75%	26.43%	6.96%	15.94%	28.17%
2	9.89%	25.31%	7.66%	1.88%	1.5%
3	11.07%	41.27%	9.69%	-6.02%	-6.19%
4	3.57%	23.44%	19.8%	29.93%	45.93%
5	-3.6%	11.23%	7.91%	5.69%	1.57%
6	5.64%	37.43%	15.35%	16.71%	29.69%
7	19.61%	39.26%	24.07%	-10.81%	-10.81%
8	-2.25%	12.59%	27.43%	29.28%	19.22%
9	11.32%	4.98%	16.5%	33.49%	5.3%
10	9.76%	21.67%	8.49%	9.38%	19.93%
11	-13.27%	5.03%	-6.36%	5.39%	5.39%
12	10.6%	8.27%	8.94%	1.69%	17.57%
13	4.23%	17.82%	9.93%	0.37%	2.16%
14	17.46%	33.41%	24.48%	22.59%	24.87%
15	7.18%	1.36%	0.32%	0.21%	39.72%
16	25.58%	39.62%	30.43%	43.86%	38.46%
17	-25%	4.76%	-13.21%	11.76%	4.26%
18	22.68%	17.58%	23.21%	1.32%	-10.84%
19	-3.45%	15.49%	15.49%	20.53%	20.53%
20	-4.1%	16.7%	33.2%	25.47%	20.39%
21	-1.26%	5.58%	-16.91%	22.99%	24.53%
22	**14.89%**	19.19%	25.23%	8.05%	17.53%
23	16.53%	12%	-12.82%	-9.22%	-9.61%
24	20.89%	**15.33%**	10.61%	6.13%	16.79%
25	10.81%	0	5.71%	10.81%	5.71%
26	2.91%	2.91%	7.41%	4.76%	6.54%
27	10.14%	13.89%	**-0.81%**	0	6.06%
Mean	6.87%	17.5%	10.69%	11.19%	13.5%

Conclusions

- MA-IDDQN is proposed to address the OBHFSP via forming batch and scheduling to minimize the total tardiness
- The average improvement rate of performance is between 6.87%– 17.5%
- In further work, more uncertain disturbances such as machine breakdowns and raw material shortage will be considered

Q & A