Supplementary Material for ‘Action-Priority Driven Policy Optimization for Flexible Job Shop Scheduling Problem’

Wenjun Zheng[1], Weilin Cai[2], Wei Qu[3], Jianfeng Mao*
[1,2,3]:{wenjunzheng,weilincai,weiqu}@link.cuhk.edu.cn
*: jfmao@cuhk.edu.cn; corresponding author

I. Important Notations

This paper introduces various notations for problem descriptions, property definitions, formal proofs, and related purposes. In this subsection, we highlight the key notations that are frequently used throughout the chapters:

$\mathcal{J}$:	The set for all jobs.	$\mathcal{M}$:	The set for all machines.
$\mathbb{n}$:	The number of jobs.	$\mathbb{m}$:	The number of machines.
J_j:	The j-th job.	M_m:	The m-th machine.
$\mathbb{k}_j$:	# operations of job j.	$O_{j,i}$:	i-th operation of job j.
$\mathcal{M}_{j,i}$:	Machines that can process $O_{j,i}$.	$(\mathcal{P})_{m,j,i}$:	Processing time of M_m on $O_{j,i}$.
$p_{m,j,i}$:	Remaining processing time of M_m on $O_{j,i}$.	$C_{\max}$:	The makespan of a schedule.
t:	Stage index.	s_t:	State at stage t.
π:	Stochastic policy.	μ:	Deterministic policy.
$\mathbf{J}_t$:	Job progress matrix	$\mathbf{M}_t$:	Machine availability vector
$\mathbf{W}_t$:	Work set.	$\mathbb{T}_t$:	System clock
$\mathcal{S}$:	Set of all states.	$\mathcal{O}$:	Set of all observations.
$\mathcal{A}$:	All feasible actions	$\mathcal{A}(s)$:	Feasible actions given state s.
$A_m(s)$:	Feasible action set with the machine fixed as M_m	$A_{(j,i))}(s)$	Feasible action set with the operation fixed as $O_{j,i}$
τ:	Trajectory	$G(\tau)$:	Return of a trajectory
$V^{\pi}(s)$:	Value function under policy π.	P_{τ}^{π}:	The probability of trajectory τ under policy π.
$\mathbb{I}(s,a)$:	Indicator function.	δ:	Step size of policy update
t_{τ}^{+}:	First system clock change in trajectory τ.	$\tau\|s$:	Truncated trajectory of τ after s
$\mathcal{A}_t(\tau)$:	The set of actions in τ before stage t.	Φ:	Action-priority vector.

Please note that these notations are explained in detail in their respective sections. Not all variables are included here, as **some intermediate variables are intentionally omitted**. This subsection serves as a centralized reference for quick access to commonly used notations.

II. Model Explanation

A. Detailed Version of System Dynamics

In this subsection, we introduce the system dynamics in details. Since the FJSP problem is deterministic, the next state $s_{t+1} = (\mathbf{J}_{t+1}, \mathbf{M}_{t+1}, \mathbf{W}_{t+1}, \mathbb{T}_t)$ is uniquely determined given the current state $s_t = (\mathbf{J}_t, \mathbf{M}_t, \mathbf{W}_t)$ and action $a_t = (m, j, i)$. The state update follows three steps: Step (1) is an **Intermediate Information Storage**: In this step, intermediate job, machine, and working information are created to store the state after applying action a. Here Step (2) is for **Checking Available Actions**: Since, in our model, a stage represents a decision point when an available machine can be assigned to an operation, we need to check whether any available actions remain after executing a. If no actions are available, we proceed to Step (3), i.e. **Processing Progression**: The system advances the processing stage, during which some operations are completed, and some machines are released. After this update, we again check for available actions, leading back to Step (2) if any exist. The details for each step are shown below:

(1) First, obtain intermediate job information $\mathbf{J}$, machine information $\mathbf{M}$, working information $\mathbf{W}$ and current system clock $\mathbb{T}$. The newly assigned machine M_m for operation $O_{j,i}$ and its remaining processing time are added to the work information:

$$\mathbf{W} = \mathbf{W}_t \cup \{(m, j, i, (\mathcal{P})_{m,j,i})\}$$

The first column of $\mathbf{J}_t$ remains unchanged:

$$(\mathbf{J})_{*,1} = (\mathbf{J}_t)_{*,1}$$

The status of job J_j is updated from idle to in progress:

$$(\mathbf{J})_{j',2} = \begin{cases} 1, & \text{if } j' = j, \\ (\mathbf{J}_t)_{j',2}, & \text{otherwise.} \end{cases}$$

The status of machine M_m is changed from idle to busy:

$$(\mathbf{M})_{m'} = \begin{cases} 1, & \text{if } m' = m, \\ (\mathbf{M}_t)_{m'}, & \text{otherwise.} \end{cases}$$

The system clock remains the same: $\mathbb{T} = \mathbb{T}_t$.

(2) If based on $\mathbf{J}$, $\mathbf{M}$ and $\mathcal{P}$, there is no available action, then goes to Step (3), otherwise let $\mathbf{J}_{t+1} = \mathbf{J}, \mathbf{M}_{t+1} = \mathbf{M}, \mathbf{W}_{t+1} = \mathbf{W}, \mathbb{T}_{t+1} = \mathbb{T}$.

(3) Since there is no action available, the system should precede the processing progress. Let $\hat{p}$ represent the minimum remaining processing time, defined as:

$$\hat{p} = \min \{p_{m,j,i} \mid (m, j, i, p_{m,j,i}) \in \mathbf{W}\} .$$

Then the system clock should updated as $\hat{\mathbb{T}} = \mathbb{T} + \hat{p}$. The set $\hat{\mathbf{W}}$ is then updated as follows:

$$\hat{\mathbf{W}} = \{(m, j, i, p_{m,j,i} - \hat{p}_t) \mid p_{m,j,i} > \hat{p}_t\} .$$

This update process removes machine-operation pairs where the remaining processing time is exactly $\hat{p}$, while reducing the remaining time for all other pairs accordingly. We define the set of machines that are released as:

$$\mathcal{M}^{\text{released}} = \{m \mid (m, j, i, p_{m,j,i}) \in \mathbf{W}, p_{m,j,i} = \hat{p}\} .$$

Similarly, the set of jobs that need to be updated is given by:

$$\mathcal{J}^{\text{released}} = \{j \mid (m, j, i, p_{m,j,i}) \in \mathbf{W}, p_{m,j,i} = \hat{p}\} .$$

Additionally, let

$$\mathcal{J}^{\text{done}} = \{j \mid (\mathbf{J}_t)_{j,1} = \Bbbk_j\}$$

denote the set of jobs that have already completed their final operation. The job status is updated as follows for the first column:

$$(\hat{\mathbf{J}})_{j,1} = \begin{cases} 0, & \text{if } j \in \mathcal{J}^{\text{released}} \cap \mathcal{J}^{\text{done}}, \\ (\mathbf{J})_{j,1} + 1, & \text{if } j \in \mathcal{J}^{\text{released}} \setminus \mathcal{J}^{\text{done}}, \\ (\mathbf{J})_{j,1}, & \text{if } j \notin \mathcal{J}^{\text{released}}. \end{cases}$$

- If $j \in \mathcal{J}^{\text{released}} \cap \mathcal{J}^{\text{done}}$, the job has completed its final operation.
- If $j \in \mathcal{J}^{\text{released}} \setminus \mathcal{J}^{\text{done}}$, the job has finished its current operation but still has remaining steps.

For the second column, the update rule is:

$$(\hat{\mathbf{J}})_{j,2} = \begin{cases} 0, & \text{if } j \in \mathcal{J}^{\text{released}}, \\ (\mathbf{J})_{j,2}, & \text{if } j \notin \mathcal{J}^{\text{released}}. \end{cases}$$

The machine status is updated as follows:

$$(\mathbf{M}_{t+1})_m = \begin{cases} 0, & \text{if } m \in \mathcal{M}^{\text{released}}, \\ (\mathbf{M})_m, & \text{if } m \notin \mathcal{M}^{\text{released}}. \end{cases}$$

This ensures that machines in $\mathcal{M}^{\text{released}}$ are marked as available, while others retain their previous status. Let $\mathbf{J} = \hat{\mathbf{J}}, \mathbf{M} = \hat{\mathbf{M}}, \mathbf{W} = \hat{\mathbf{W}}, \mathbb{T} = \hat{\mathbb{T}}$, then goes to Step (2).

B. Model Illustration

To illustrate our model design more effectively, we present a small example with a dynamic visualization, as shown in Figure 1. The first part, enclosed within the top rectangle, introduces the problem setting: the example consists of two jobs and two machines, with each job containing two operations. The processing options for each operation are as follows:

- $O_{1,1}$ can be processed by M_1 in 2 time units or by M_2 in 1 time unit, as indicated by the dotted line.
- $O_{1,2}$ can only be processed by M_2 in 2 time units.
- $O_{2,1}$ can only be processed by M_2 in 3 time units.
- $O_{2,2}$ can only be processed by M_1 in 2 time units.

To visually distinguish operations, we use border colors: dark red for Job 1 and green for Job 2. Machines are differentiated by their fill colors, with light orange representing Machine 1 and light red representing Machine 2.

Following the problem setup, we present a sequence of cut-scenes illustrating the model's progression from Stage 0 (initial stage) to the terminal stage, i.e., Stage 4. The cut-scenes are arranged vertically and successively, each within a rectangle border. Each cut-scene consists of:

1) **Gantt Charts (left)**: Rectangles are used to represent operations: ⟨1⟩ Current operations (i.e., the latest ones that are not yet started or the earliest ones that are not yet completed) of each job are wrapped with thick borders; ⟨2⟩ Operations that are available but not yet assigned are depicted with solid borders and no fill; ⟨3⟩ Operations that have not yet started are shown with dashed borders; ⟨4⟩ Potential scheduling actions for an operation are represented by blocks with dashed borders and filled with color, with block lengths proportional to the processing times of the corresponding operations; ⟨5⟩ The remaining portion of each assigned operation is displayed with solid borders and filled with the color of the assigned machine—either light orange or light red, with some transparency; ⟨6⟩ The assigned action can also be inferred

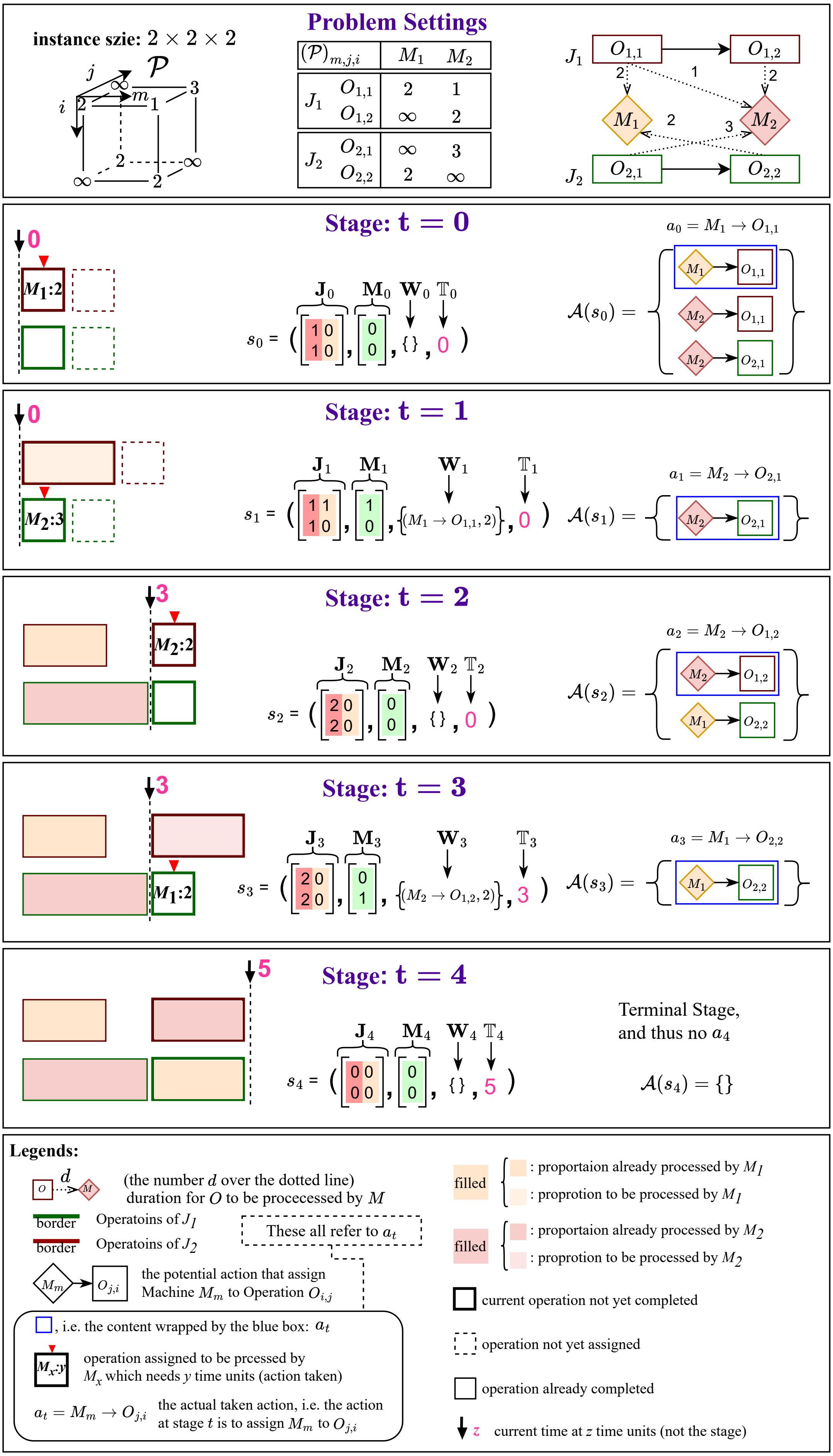

Fig. 1. Model Illustration

from this Gantt chart: a red downward-pointing triangle indicates the operation assigned to a machine, and the machine is explicitly written inside the current operation rectangle, followed by the expected processing time after a colon.

2) **Timeline Marker (top-left)**: Arrows indicate the current and next time steps. The line for the current time is black, while the line for the next time is light gray. The value corresponds to $\mathbb{T}_t$, the fourth element in the state.
3) **Stage Marker (top)**: The current stage is marked with a number denoted by t. Note that this is distinct from the current time.
4) **State Representation (middle)**: This section presents the current system state as a 4-tuple, $(\mathbf{J}, \mathbf{M}, \mathbf{W}, \mathbb{T})$. Note that when $\mathbf{W}$ is empty, we represent it as "{}" rather than "∅" for consistency.
5) **Feasible Action Set (right)**: Feasible actions—i.e., proper operation-machine assignments—are enclosed by a pair of large brackets. Each action is represented ideographically, with the machine inside a rhombus and the operation inside a rectangle. An arrowed line connects the two shapes. The selected action is highlighted with a blue outer border and is further emphasized above the action set.

At the bottom, relevant legends are provided and explained. Note that there are three ways to represent the current action, all of which are detailed in the legend. Readers may follow the visualization from top to bottom in general, and from left to right within each cut-scene. The cut-scenes are designed to be self-contained, enabling readers to understand the model design without referring to the main text.

III. Proofs in Details

A. Proof of Lemma 1 (with the original lemma provided)

Lemma 1. *Given a policy π and its corresponding action-priority vector Φ, suppose $G(\tau') > V^\pi(s_0')$, then for the new policy π' and its corresponding Φ' constructed according to (8), we have following relations:*

- $\pi(a|s) = \pi'(a|s), \forall a \neq a_0', a \in \mathcal{A}(s)$, *if a_0' is invalid for state s.*
- $\pi'(a|s) < \pi(a|s), \forall a \neq a_0', a \in \mathcal{A}(s)$, *if a_0' is valid for state s. More specifically, if s is the state that satisfies $\mathcal{A}(s) \subseteq \mathcal{A}(s_0')$, then $\pi'(a|s) \leq \delta\pi(a|s)$ with equality holds when $\mathcal{A}(s) = \mathcal{A}(s_0')$.*

For the case $G(\tau') < V^\pi(s_0')$, the first result still holds, while the second result changes the inequality $\leq$ to $\geq$.

Proof. Here, we provide the proof for the case $G(\tau') > V^\pi(s_0')$. The proof for the converse case is similar.

- If $\tilde{a}_0$ is invalid for state s, then $\mathbb{I}(s, \tilde{a}_0) = 0$.

$$
\begin{aligned}
\pi'(a|s) &= \frac{\phi_a'}{\sum_j \mathbb{I}(s,j)\phi_j'} = \frac{\phi_a'}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j'} \\
&= \frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\delta\phi_j} = \frac{\delta\phi_a}{\delta\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j} \\
&= \frac{\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j} = \frac{\phi_a}{\sum_j \mathbb{I}(s,j)\phi_j} \\
&= \pi(a|s)
\end{aligned}
$$

- If $\tilde{a}_0$ is valid for state s, then $\mathbb{I}(s, \tilde{a}_0) = 1$.

$$
\begin{aligned}
\pi'(a|s) &= \frac{\phi_a'}{\sum_j \mathbb{I}(s,j)\phi_j'} = \frac{\phi_a'}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j' + \phi_{\tilde{a}_0}'} \\
&= \frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\delta\phi_j + \phi_{\tilde{a}_0} + (1-\delta)\sum_{j\neq\tilde{a}_0} \mathbb{I}(\tilde{s}_0,j)\phi_j} < \frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\delta\phi_j + \delta\phi_{\tilde{a}_0}} \\
&= \frac{\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j + \phi_{\tilde{a}_0}} = \frac{\phi_a}{\sum_j \mathbb{I}(s,j)\phi_j} \\
&= \pi(a|s)
\end{aligned}
$$

Now, if $\mathcal{A}(s) \subseteq \mathcal{A}(\tilde{s}_0)$, we have $\sum_{j\neq\tilde{a}_0} \mathbb{I}(\tilde{s}_0,j)\phi_j \geq \sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j$ since $\phi_i > 0$. Therefore,

$$
\begin{aligned}
\frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\delta\phi_j + \phi_{\tilde{a}_0} + (1-\delta)\sum_{j\neq\tilde{a}_0} \mathbb{I}(\tilde{s}_0,j)\phi_j} &\leq \frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\delta\phi_j + \phi_{\tilde{a}_0} + (1-\delta)\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j} \\
&= \frac{\delta\phi_a}{\sum_{j\neq\tilde{a}_0} \mathbb{I}(s,j)\phi_j + \phi_{\tilde{a}_0}} = \delta\pi(a|s)
\end{aligned}
$$

That is $\pi'(a|s) \leq \pi(a|s)$. ■

B. Proof of Lemma 2 (with the original lemma provided)

Lemma 2. *Suppose $\mathcal{A}_m(s_0')$ satisfying **CD3** and $\tau \in \mathcal{T}$, then $t_\tau^+ \geq 2$.*

Proof. Since $\tau \in \mathcal{T}$, by the definition of $\mathcal{T}$, we have $\tau = (s_0', a_0, \cdots)$, where either $a_0 \in \mathcal{A}_m(s_0') \cup \mathcal{A}_{(j,i)}(s_0')$ or $a_0 \notin \mathcal{A}_m(s_0') \cup \mathcal{A}_{(j,i)}(s_0')$. If $a_0 \in \mathcal{A}_m(s_0') \cup \mathcal{A}_{(j,i)}(s_0')$, then by **CD3**, we directly obtain $t_\tau^+ \geq 2$. Otherwise, if $a_0 \notin \mathcal{A}_m(s_0')$, suppose $a_0 = (m, j, i)$ and $a_0' = (m', j', i')$. Since $m \neq m'$, and $j \neq j'$, $i \neq i'$, it follows that after executing action a_0, action a_0' is still available. Hence, we conclude that $t_\tau^+ \geq 2$. ■

C. Proof of Lemma 3 (with the original lemma provided)

Lemma 3. *Suppose $\tau \in \mathcal{T}_1$, then $a_0' \notin \mathcal{A}_{t_\tau^+}(\tau)$. If π' is updated according to updating rule (8), then*

$$\mathrm{P}_\tau^{\pi'} < \frac{\delta^2(1-\delta\eta)\sum_j \mathbb{I}_j(s)\phi_j}{(1-\eta)(\delta\sum_j \mathbb{I}_j(s)\phi_j + (1-\delta)\sum_j \mathbb{I}(s_0', j)\phi_j)}\mathrm{P}_\tau^{\pi}, s \in \tau$$

where $\mathrm{P}_\tau^{\pi'}$ represents the probability of τ given policy π'. π' is the new policy and π is the old one.

Proof. Suppose $\tau = (s_0', a_0, \ldots, a_{t_\tau^+ - 1}, s_{t_\tau^+}, \ldots)$. By the definition of $\mathcal{T}_1$, we have $a_t \neq a_0', \forall 0 \le t \le t_\tau^+ - 1$. Then, by Lemma 1, it follows that $\pi'(a_0|s_0') = \delta\pi(a_0|s_0')$ and $\pi'(a_t|s_t) < \delta\pi(a_t|s_t)$ for $1 \le t \le t_\tau^+ - 1$.

In the worst case, if action a_0' is selected at some state s in τ, then

$$\pi(a_0'|s) = \frac{\phi_{a_0'}}{\sum_{j\neq a_0'} \mathbb{I}_j(s)\phi_j + \phi_{a_0'}}$$

and

$$\pi'(a_0'|s) = \frac{\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j}{\delta\sum_{j\neq a_0'} \mathbb{I}_j(s)\phi_j + \phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j}.$$

Again, by Lemma 1, once action a_0' is selected, we have $\pi'(a|s) = \pi(a|s)$. Thus, for some $s \in \tau$, it follows that

$$\begin{aligned} p_\tau^{\pi'} &< \frac{\delta^{t_\tau^+}[\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j]}{\delta\sum_{j\neq a_0'} \mathbb{I}_j(s)\phi_j + \phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j} \cdot \frac{\sum_{j\neq a_0'} \mathbb{I}_j(s)\phi_j + \phi_{a_0'}}{\phi_{a_0'}} p_\tau^\pi, \quad \text{for some } s \in \tau \\ &= \frac{\delta^{t_\tau^+}(1-\delta\eta)\sum_j \mathbb{I}_j(s)\phi_j}{(1-\eta)(\delta\sum_j \mathbb{I}_j(s)\phi_j + (1-\delta)\sum_j \mathbb{I}_j(s_0')\phi_j)} p_\tau^\pi. \end{aligned}$$

Since, by Lemma 2, we have $t_\tau^+ \ge 2$, it follows that

$$p_\tau^{\pi'} < \frac{\delta^2(1-\delta\eta)\sum_j \mathbb{I}_j(s)\phi_j}{(1-\eta)(\delta\sum_j \mathbb{I}_j(s)\phi_j + (1-\delta)\sum_j \mathbb{I}_j(s_0')\phi_j)} p_\tau^\pi.$$

■

D. Proof of Lemma 4 (with the original lemma provided)

Lemma 4. *Suppose $\tau \in \mathcal{T}$, $a_0' \in \mathcal{A}_{t_\tau^+}(\tau)$ and a_0' is executed at stage t, we know $0 \le t < t_\tau^+$. Then,*

$$\mathrm{P}_\tau^{\pi'} < \frac{\delta^t(1-\delta\eta)}{1-\eta}\mathrm{P}_\tau^\pi$$

Proof. By definition, the set $\mathcal{A}_{t_\tau^+}(\tau)$ consists of the actions taken in trajectory τ before stage t_τ^+. Since $a_0' \in \mathcal{A}_{t_\tau^+}(\tau)$, we can express τ as

$$\tau = (s_0', a_0, \ldots, s_t, a_0', \ldots, a_{t_\tau^+ - 1}, s_{t_\tau^+}, \ldots).$$

Then, by Lemma 1, it follows that $\pi'(a_0|s_0') = \delta\pi(a_0|s_0')$ and $\pi'(a_{t'}|s_{t'}) < \delta\pi(a_{t'}|s_{t'})$ for $1 \le t' \le t-1$. At state s_t, we have

$$\pi(a_0'|s_t) = \frac{\phi_{a_0'}}{\sum_j \mathbb{I}_j(s_t)\phi_j}$$

and

$$\pi'(a_0'|s_t) = \frac{\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j}{\delta\sum_{j\neq a_0'} \mathbb{I}_j(s_t)\phi_j + \phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j}.$$

From Lemma 1, once action a_0' is taken, we have $\pi'(a|s_{t'}) = \pi(a|s_{t'})$ for $t' > t$. Thus, we obtain

$$\begin{aligned} p_\tau^{\pi'} &< \frac{\delta^t(\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j)}{\delta\sum_{j\neq a_0'} \mathbb{I}_j(s_t)\phi_j + \phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}_j(s_0')\phi_j} \cdot \frac{\sum_j \mathbb{I}_j(s_t)\phi_j}{\phi_{a_0'}} p_\tau^\pi \\ &< \frac{\delta^t(\delta\phi_{a_0'} + (1-\delta)\sum_j \mathbb{I}_j(s_0')\phi_j)}{\phi_{a_0'}} p_\tau^\pi \\ &= \frac{\delta^t(1-\delta\eta)}{1-\eta} p_\tau^\pi. \end{aligned}$$

Such relaxation exactly match the required inequality in Lemma 4. ■

E. Proof of Lemma 5 (with the original lemma provided)

Lemma 5. *Suppose that $\tilde{\tau} := (s'_0, a'_0, \tilde{s}_1, \tilde{a}_1, \ldots, \tilde{a}_{t^+_{\tilde{\tau}}-1}, \tilde{s}_{t^+_{\tilde{\tau}}}, \ldots) \in \mathcal{T}_3$, and the set $\{a'_0, \tilde{a}_1, \ldots, a_{t^+_{\tilde{\tau}}-1}\}$ contains all the actions taken before the first change in system clock within trajectory $\tilde{\tau}$. Next, define $\tilde{\tau}^{(t)}$ as the trajectory whose first $t^+_{\tilde{\tau}}$ actions match the set $\{a'_0, \tilde{a}_1, \ldots, a_{t^+_{\tilde{\tau}}-1}\}$, and satisfy $\tilde{\tau}^{(t)}|\tilde{s}_{t^+_{\tilde{\tau}}} = \tilde{\tau}|\tilde{s}_{t^+_{\tilde{\tau}}}$. By the existence of the **Permutation Irrelevance** Property (Proposition 3), the equality $\tilde{\tau}^{(t)}|\tilde{s}_{t^+_{\tilde{\tau}}} = \tilde{\tau}|\tilde{s}_{t^+_{\tilde{\tau}}}$ is well-defined.*

The order of the first $t^+_{\tilde{\tau}}$ actions in $\tilde{\tau}^{(t)}$ is given by

$$a^{(t)}_{t'} = \begin{cases} a_1, & \text{if } t' = 0, \\ a'_0, & \text{if } t' = t, \\ a_{t'+1}, & \text{if } 0 < t' < t, \\ a_{t'}, & \text{if } t' > t, \end{cases} \tag{1}$$

where $0 \le t' \le t^+_{\tilde{\tau}} - 1$.

Then, an equivalent expression for the set $\mathcal{T}_2$ is given by

$$\mathcal{T}_2 = \bigcup_{\tilde{\tau} \in \mathcal{T}_3} \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}.$$

Similarly, the set $\mathcal{T}_4$ can be expressed in the same form. In other words, $\mathcal{T}_4$ is a special case of this relation, since $\mathcal{T}_5$ contains only a single element, τ'. Specifically, we have

$$\mathcal{T}_4 = \{\tau'^{(0)}, \ldots, \tau'^{(t^+_{\tau'}-1)}\}.$$

where $\tau'^{(t)}$ for $0 \le t \le t^+_{\tau'} - 1$ is defined similarly, satisfying $\tau'^{(t)}|s'_{t^+_{\tau'}} = \tau'|s'_{t^+_{\tau'}}$.

Proof. We first derive an equivalent expression for the set $\mathcal{T}_2$. Recall its definition:

$$\mathcal{T}_2 = \left\{\tau \mid \exists \tilde{\tau} \in \mathcal{T}_3, \text{ s.t. } \mathcal{A}_{t^+_{\tilde{\tau}}}(\tau) = \mathcal{A}_{t^+_{\tilde{\tau}}}(\tilde{\tau}), \tau \neq \tilde{\tau}, \tau \notin \mathcal{T}_3, \tau \in \mathcal{T}\right\}.$$

For any $\tau \in \mathcal{T}_2$, there exists some $\tilde{\tau} \in \mathcal{T}_3$ such that $\mathcal{A}_{t^+_{\tilde{\tau}}}(\tau) = \mathcal{A}_{t^+_{\tilde{\tau}}}(\tilde{\tau})$. This implies that the action set before the first system change is identical for both trajectories. Since $\tau \neq \tilde{\tau}$, the sequence of actions before stage $t^+_{\tilde{\tau}}$ must be a permutation of the set $\mathcal{A}_{t^+_{\tilde{\tau}}}$. Consequently, $\tau \in \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}$. Thus, we conclude that

$$\mathcal{T}_2 \subseteq \bigcup_{\tilde{\tau} \in \mathcal{T}_3} \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}.$$

Conversely, suppose $\tau \in \bigcup_{\tilde{\tau} \in \mathcal{T}_3} \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}$. Then, there must exist some $\tilde{\tau} \in \mathcal{T}_3$ such that $\tau \in \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}$. By the definition of the set $\{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}$ (see equation (1)), we have

$$\mathcal{A}_{t^+_{\tilde{\tau}}}(\tau) = \mathcal{A}_{t^+_{\tilde{\tau}}}(\tilde{\tau}) \quad \text{and} \quad \tau \neq \tilde{\tau}.$$

This implies $\tau \in \mathcal{T}_2$. Thus, $\bigcup_{\tilde{\tau} \in \mathcal{T}_3} \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\} \subseteq \mathcal{T}_2$. Combining both directions, we can then conclude that $\mathcal{T}_2 = \bigcup_{\tilde{\tau} \in \mathcal{T}_3} \{\tilde{\tau}^{(0)}, \ldots, \tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}$. The proof of the equivalence relation for the set $\mathcal{T}_4$ is similar and follows the same structure as the proof above. ■

F. Proof of Proposition 1 (with the original proposition provided)

Proposition 1. *For any $\delta \in (0, 1)$, condition **CD1** is satisfied, i.e.,*

$$\frac{\delta^2(1-\delta\eta)}{(1-\eta)\left(\delta + (1-\delta)\frac{\sum_j \mathbb{I}(s'_0, j)\phi_j}{\sum_j \mathbb{I}_j(s)\phi_j}\right)} < \frac{1-(1-\delta\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}, \quad \forall s \neq s'_0.$$

Proof. Denote $\sigma = \frac{\sum_j \mathbb{I}(s'_0, j)\phi_j}{\sum_j \mathbb{I}_j(s)\phi_j} > 0$, then

$$\begin{aligned}
&\frac{\delta^2(1-\delta\eta)}{(1-\eta)(\delta + (1-\delta)\frac{\sum_j \mathbb{I}(s'_0, j)\phi_j}{\sum_j \mathbb{I}_j(s)\phi_j})} < \frac{1-(1-\delta\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}} \\
\Leftrightarrow &\frac{\delta^2(1-\delta\eta)}{(1-\eta)((1-\sigma)\delta+\sigma)} < \frac{1-(1-\delta\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}} \\
\Leftrightarrow &\delta^2(1-\delta\eta)\frac{1-(1-\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}{1-\eta} < ((1-\sigma)\delta+\sigma)(1-(1-\delta\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}) \\
\Leftrightarrow &\delta^2(1-\delta\eta)\frac{1-(1-\eta)\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}}{1-\eta} < (1-\sigma)\eta\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}\delta^2 + [(1-\sigma)(1-\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}) + \sigma\eta\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)}]\delta + \sigma(1-\mathrm{P}^\pi_{\tau'|(s'_0,a'_0)})
\end{aligned}$$

$$\Leftrightarrow \left[\frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} - (1-\sigma)\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}\right]\delta^2 - [(1-\sigma)(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}) + \sigma\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}]\delta - \sigma(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}) < \frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta}\eta\delta^3$$

Let $f_1(\delta) = \left[\frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} - (1-\sigma)\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}\right]\delta^2 - [(1-\sigma)(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}) + \sigma\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}]\delta - \sigma(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')})$ and $f_2(\delta) = \frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta}\eta\delta^3$. We find that $f_1(1) = f_2(1)$ and $f_1(0) = -\sigma(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}) < 0$. Since

$$\begin{aligned}
\frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} - (1-\sigma)\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} &< \frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} - \eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} \\
&= \frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - \eta(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} \\
&= \frac{1-(1-\eta^2)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta} \\
&> \frac{1-(1-\eta^2)}{1-\eta} = \frac{\eta^2}{1-\eta} > 0
\end{aligned}$$

Hence, $f_1(\delta)$ is a convex quadratic function. Since

$$\frac{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\eta}\eta > 0,$$

it follows that, as shown in Figure 2, we have $f_1(\delta) < f_2(\delta), \quad \forall\delta \in (0,1)$. ∎

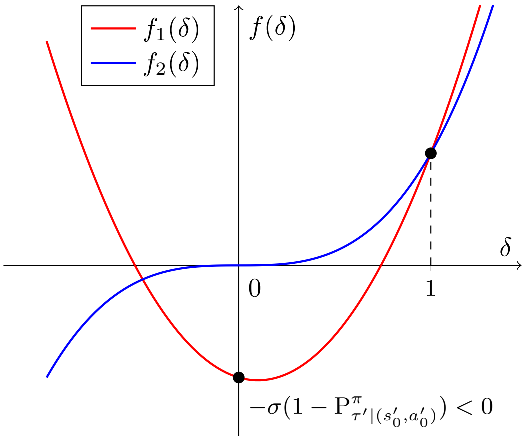

Fig. 2. $f_1(\delta), f_2(\delta)$

G. Proof of Proposition 2 (with the original proposition provided)

Proposition 2. *If* $1 - \frac{\eta}{1-\eta} > \alpha + \beta$, *where*

$$\alpha = \frac{\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}, \quad \beta = \frac{1}{1-\eta}\frac{\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\frac{\sum_j \mathbb{I}(s_0',j)\phi_j}{\sum_j \mathbb{I}(s_1',j)\phi_j},$$

then there exist a $\epsilon > 0$, *such that when* $\delta \in (\epsilon, 1)$, ***CD2*** *is satisfied.*

Remark 1. *By the definition of*

$$\eta = \frac{\sum_{j\neq a_0'} \mathbb{I}(s_0',j)\phi_j}{\sum_j \mathbb{I}(s_0',j)\phi_j},$$

as well as the trajectory τ', *states* s_0' *and* s_1', *the values of* α *and* β *can be easily computed. Thus, the condition*

$$1 - \frac{\eta}{1-\eta} > \alpha + \beta$$

can be efficiently verified.

Proof. Given $\tau \in \mathcal{T}_3, \mathcal{T}_5$, let $g_1(\delta) = \sum_{t=1}^{t_\tau^+-1} \delta^t \frac{1-\delta\eta}{1-\eta}\mathrm{P}^{\pi}_{\tau(t)}$ and

$$g_2(\delta) = \delta\left(\sum_{t=1}^{t_\tau^+-1} \frac{\eta\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}\mathrm{P}^{\pi}_{\tau(t)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} + \frac{\eta\mathrm{P}^{\pi}_{\tau|(s_0',a_0')}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right)\sum_{t=1}^{t_\tau^+-1}\frac{(1-\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')})\mathrm{P}^{\pi}_{\tau(t)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} - \frac{\eta\mathrm{P}^{\pi}_{\tau|(s_0',a_0')}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}$$

The following theorem is established in Spivak, M.'s book [1], and we will use it directly:

Theorem (for Proposition 2)**.** *Let $g_1(\delta)$ and $g_2(\delta)$ be continuously differentiable functions on $(0,1]$. If*

$$g_1(1) = g_2(1) \quad \textit{and} \quad g_1'(1) > g_2'(1),$$

then there exists $\epsilon > 0$ such that

$$g_1(\delta) < g_2(\delta), \quad \forall \delta \in (\epsilon, 1).$$

Since $g_1(1) = g_2(1)$ and both functions are continuously differentiable, it suffices to show that $g_1'(1) > g_2'(1)$. If this holds, then for δ close to 1, we have $g_1(\delta) < g_2(\delta)$.

The derivatives at $\delta = 1$ are given by

$$g_1'(1) = \sum_{t=1}^{t_\tau^+ - 1} \left(t - \frac{\eta}{1-\eta} \right) \mathrm{P}^\pi_{\tau^{(t)}}$$

and

$$g_2'(1) = \sum_{t=1}^{t_\tau^+ - 1} \frac{\eta \mathrm{P}^\pi_{\tau'|(s_0', a_0')} \mathrm{P}^\pi_{\tau^{(t)}}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}} + \frac{\eta \mathrm{P}^\pi_{\tau|(s_0', a_0')}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}}.$$

Observing the dependency on t_τ^+, we see that when t_τ^+ increases by 1,

$$g_1'(1) \quad \text{increases by} \quad (t_\tau^+ - \frac{\eta}{1-\eta}) \mathrm{P}^\pi_{\tau^{(t_\tau^+)}},$$

whereas

$$g_2'(1) \quad \text{increases by} \quad \frac{\eta \mathrm{P}^\pi_{\tau'|(s_0', a_0')} \mathrm{P}^\pi_{\tau^{(t_\tau^+)}}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}}.$$

By Lemma 2, we know that $t_\tau^+ \geq 2$ and since the increase in $g_1'(1)$ is greater than that of $g_2'(1)$, therefore, we only need to consider the base case $t_\tau^+ = 2$, where

$$g_1'(1) = \left(1 - \frac{\eta}{1-\eta} \right) \mathrm{P}^\pi_{\tau^{(1)}}$$

and

$$g_2'(1) = \frac{\eta \mathrm{P}^\pi_{\tau'|(s_0', a_0')} \mathrm{P}^\pi_{\tau^{(1)}}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}} + \frac{\eta \mathrm{P}^\pi_{\tau|(s_0', a_0')}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}}.$$

Now, given that $t_\tau^+ = 2$, suppose $\tau = (s_0', a_0', s_1, a_1, s_2, \dots)$, then we have $\tau^{(1)} = (s_0', a_1, s_1^{(1)}, a_0', s_2, \dots)$. Thus,

$$\begin{aligned}
\mathrm{P}^\pi_{\tau|(s_0', a_0')} = \pi(a_1|s_1)\mathrm{P}^\pi_{\tau|(s_1, a_1)} &= \frac{\pi(a_1|s_1)\pi(a_1|s_0')\pi(a_0'|s_1^{(1)})\mathrm{P}^\pi_{\tau|(s_1, a_1)}}{\pi(a_1|s_0')\pi(a_0'|s_1^{(1)})} && (2) \\
&= \frac{\pi(a_1|s_1)\mathrm{P}^\pi_{\tau^{(1)}}}{\pi(a_1|s_0')\pi(a_0'|s_1^{(1)})} && (3) \\
&= \mathrm{P}^\pi_{\tau^{(1)}} \frac{\phi_{a_1}}{\sum_j \mathbb{I}(s_1, j)\phi_j} \frac{\sum_j \mathbb{I}(s_0', j)\phi_j}{\phi_{a_1}} \frac{\sum_j \mathbb{I}(s_1^{(1)}, j)\phi_j}{\phi_{a_0'}} && (4) \\
&= \mathrm{P}^\pi_{\tau^{(1)}} \frac{\sum_j \mathbb{I}(s_0', j)\phi_j}{\phi_{a_0'}} \frac{\sum_j \mathbb{I}(s_1^{(1)}, j)\phi_j}{\sum_j \mathbb{I}(s_1, j)\phi_j} && (5) \\
&= \mathrm{P}^\pi_{\tau^{(1)}} \frac{1}{1-\eta} \frac{\sum_j \mathbb{I}(s_1^{(1)}, j)\phi_j}{\sum_j \mathbb{I}(s_1, j)\phi_j} && (6) \\
&< \mathrm{P}^\pi_{\tau^{(1)}} \frac{1}{1-\eta} \frac{\sum_j \mathbb{I}(s_0', j)\phi_j}{\sum_j \mathbb{I}(s_1, j)\phi_j} && (7)
\end{aligned}$$

The last inequality follows from the fact that $\mathcal{A}(s_1^{(1)}) \subset \mathcal{A}(s_0')$. Thus, we obtain

$$\begin{aligned}
g_2'(1) &= \frac{\eta \mathrm{P}^\pi_{\tau'|(s_0', a_0')} \mathrm{P}^\pi_{\tau^{(1)}}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}} + \frac{\eta \mathrm{P}^\pi_{\tau|(s_0', a_0')}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}} \\
&< \left(\frac{\eta \mathrm{P}^\pi_{\tau'|(s_0', a_0')}}{1 - (1-\eta)\mathrm{P}^\pi_{\tau'|(s_0', a_0')}} + \frac{1}{1-\eta} \frac{\sum_j \mathbb{I}(s_0', j)\phi_j}{\sum_j \mathbb{I}(s_1, j)\phi_j} \right) \mathrm{P}^\pi_{\tau^{(1)}} \\
&= (\alpha + \beta)\, \mathrm{P}^\pi_{\tau^{(1)}}
\end{aligned}$$

Thus, if $1 - \frac{\eta}{1-\eta} > \alpha + \beta$, then

$$g_2'(1) < (\alpha+\beta)\mathrm{P}^{\pi}_{\tau^{(1)}} < \left(1 - \frac{\eta}{1-\eta}\right)\mathrm{P}^{\pi}_{\tau^{(1)}} = g_1'(1).$$

This establishes the case for $t_\tau^+ = 2$. For $t_\tau^+ > 2$, the result follows from the previous discussion. Finally, by applying Theorem for Proposition 2, we complete the proof. ■

H. Proof of Proposition 3 (with the original proposition provided)

Proposition 3 (Permutation Irrelevant). *Suppose $(s_0, a_0, \cdots, s_t, a_t, s_{t+1})$ is a state-action sequence with $\mathbb{T}_0 = \mathbb{T}_t < \mathbb{T}_{t+1}$, meaning that the system clocks of states s_0 and s_t are the same and both are less than that of s_{t+1}. Then, consider any permutation of the action set $\{a_0, \dots, a_t\}$, which forms a new ordered action sequence. If we execute these actions in the new given order, the resulting state will be exactly s_{t+1}.*

Proof. It's equivalent to prove the existence of some function (or more generally, some mapping) h that maps the input $(s_0, \{a_1, a_2, \cdots, a_t\})$ to s_{t+1}, i.e. we shall have $s_{t+1} = h(s_0, \{a_1, a_2, \cdots, a_t\})$.

As we write the action as $a = (m, j, i)$ and the calculation of $\mathbf{W}$ actually adds the term $(m, j, i, (\mathcal{P})_{m,j,i})$, we may rewrite such term as $(a, (\mathcal{P})_a)$. Denote the set $\underline{\mathbf{W}} \doteq \mathbf{W}_0 \cup \{(a_0, (\mathcal{P})_{a_0}), (a_1, (\mathcal{P})_{a_1}), \cdots, (a_t, (\mathcal{P})_{a_t})\}$, the processing time $\tilde{p} \doteq \min\{(\mathcal{P})_a | (a, (\mathcal{P})_a) \in \underline{\mathbf{W}}\}$, the action set $\tilde{\mathcal{A}} \doteq \{a \mid (\mathcal{P})_a = \tilde{p}\}$, and the sets $\overline{\mathbf{W}} \doteq \underline{\mathbf{W}} \setminus \left\{(a, (\mathcal{P})_a) \middle| a \in \tilde{\mathcal{A}}\right\}$, $\tilde{\mathbf{W}} \doteq \overline{\mathbf{W}} \setminus \mathbf{W}_0$. We also define the following auxiliary sets:

- $\tilde{\mathcal{J}} \doteq \left\{j \middle| (m, j, i) \in \tilde{\mathcal{A}}\right\}$
- $\hat{\mathcal{J}} \doteq \{j | (m, j, i, (\mathcal{P})_{m,j,i}) \in \overline{\mathbf{W}}\}$
- $\bar{\mathcal{J}} \doteq \{j \mid (\mathbf{J}_0)_{j,1} = \Bbbk_j\}$
- $\tilde{\mathcal{M}} \doteq \left\{m \middle| (m, j, i, (\mathcal{P})_{m,j,i}) \in \tilde{\mathbf{W}}\right\}$
- $\hat{\mathcal{M}} \doteq \left\{m \middle| (m, j, i) \in \tilde{\mathcal{A}}\right\} \cap \{m | (m, j, i, (\mathcal{P})_{m,j,i}) \in \mathbf{W}_0\}$

Now, we can calculate s_{t+1} in as follows according to the system dynamics:

- $(\mathbf{J}_{t+1})_{j,1} = \begin{cases} 0, & \text{if } j \in \tilde{\mathcal{J}} \cap \bar{\mathcal{J}}; \\ (\mathbf{J}_0)_{j,1} + 1, & \text{if } j \in \tilde{\mathcal{J}} \setminus \bar{\mathcal{J}}; \\ (\mathbf{J}_0)_{j,1}, & \text{otherwise.} \end{cases}$
- $(\mathbf{J}_{t+1})_{j,1} = \begin{cases} 0, & \text{if } j \in \tilde{\mathcal{J}}; \\ 1, & \text{if } j \in \hat{\mathcal{J}}. \end{cases}$
- $(\mathbf{M}_{t+1})_m = \begin{cases} 0, & \text{if } m \in \hat{\mathcal{M}}; \\ 1, & \text{if } j \in \tilde{\mathcal{M}}. \end{cases}$
- $\mathbf{W}_{t+1} = \left\{(a, (\mathcal{P})_a - \tilde{p}) \middle| \left(a, (\mathcal{P}_a \in \overline{\mathbf{W}}\right)\right\};$
- $\mathbb{T}_{t+1} = \mathbb{T}_0 + \tilde{p}$.

Here the second part of the inputs of h is some set $\{a_0, a_1, \dots, a_t\}$ rather than some sequence. Since a set is identical no matter how the elements are permuted, with $s_{t+1} = h(s_0, \{a_0, a_1, \dots, a_t\})$, we can tell that the permutation irrelevant property exists.

A potential issue is that when the sequence $(a_0, a_1, \cdots, a_t)$ is reordered, the proof above requires that these actions are still feasible at each time step. This is true: for $t_x, t_y \in \{0, 1, \cdots, t\}$, suppose $a_{t_x} = (m_{t_x}, j_{t_x}, i_{t_x})$ and $a_{t_y} = (m_{t_y}, j_{t_y}, i_{t_y})$, as long as $t_x \neq t_y$, we shall have $m_{t_x} \neq m_{t_y}$ and $(j_{t_x} \neq j_{t_y})$. Since the assignments at each time step are mutually exclusive, the feasibility can be guaranteed even if the action sequence is permuted.

For a better understanding of the permutation property, we present a simple example, as shown in Figure 3, where the legends follow the tradition in 1 explained in Subsection II-A. The figure consists of seven rows: the first row describes the problem setting, while each of the remaining rows represents a state-action sequence. Each row is divided into four parts, with each part containing state information.

As illustrated in the first row, we consider three jobs, each consisting of three operations. There are three machines, and the capability of each machine is indicated by the dashed line connecting the operations. The exact processing time of a machine for a given operation is provided in a table (left-middle) and can be represented as a tensor $\mathcal{P}$ (left). If a machine is not capable of processing an operation, its corresponding value in the table is set to ∞. A detailed version for capability and processing time information is shown with three sub-figures (right).

Below are the (partial) potential trajectories with permutation of actions at the same time point. The stages of different trajectories are of superscript indicating the index of the corresponding trajectory: for example, $t^{(1)} = 4$ indicates that it's the fourth stage of the first trajectory. The stages are listed until the next stage that the current time changes, i.e. Stage 7 ($\mathbb{T}_t = 3 \rightarrow \mathbb{T}_t = 4$. We can see that given the states at Stage 4 are the same, the states at Stage 7 are also the same even though the actions are permuted among these trajectories. Thus, this also show the correctness of Proposition 3.

■

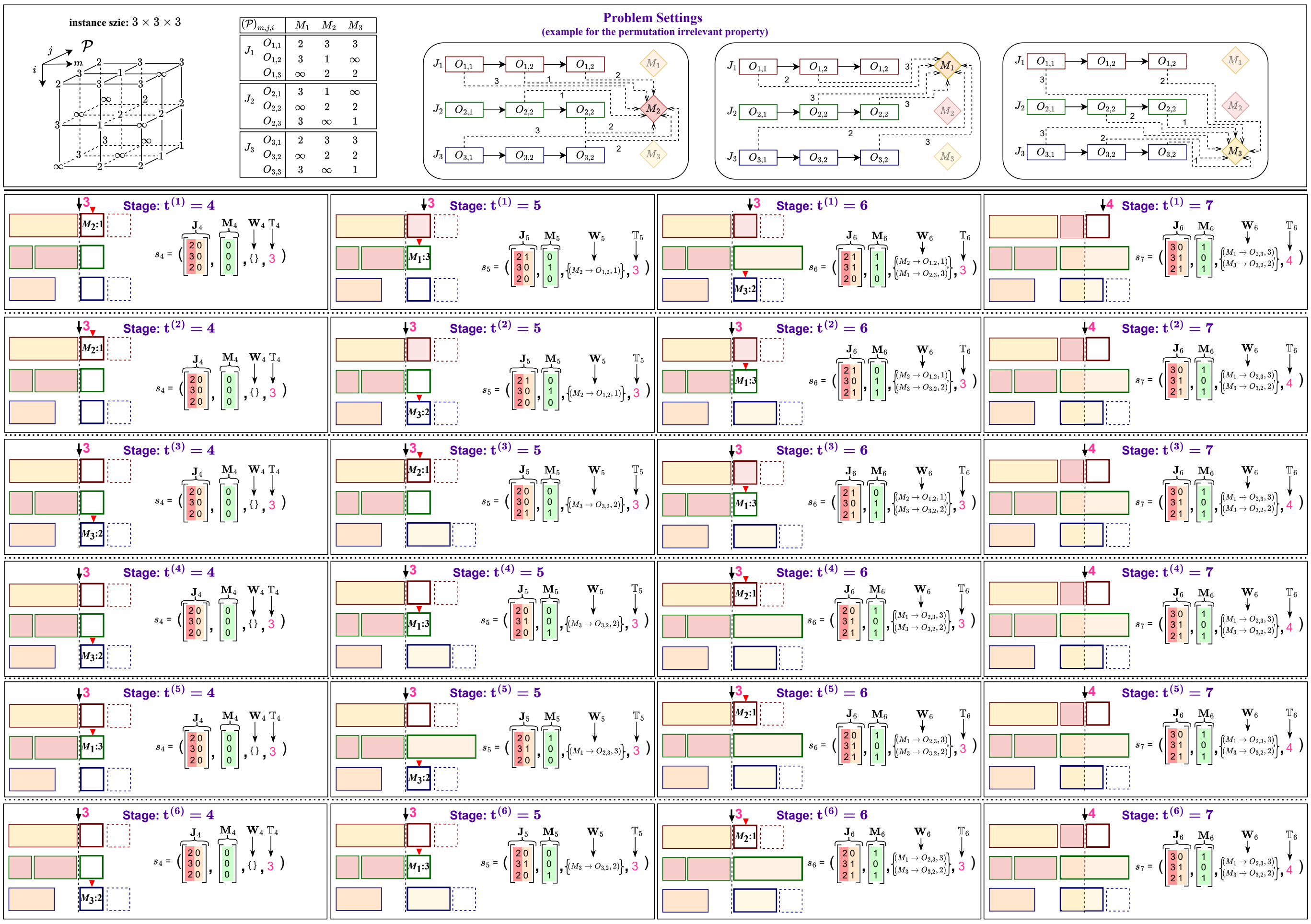

Fig. 3. Permutation Irrelevant Property Explanation

I. Proof of Theorem 1 (with the original theorem provided)

Theorem 1 (Policy Improvement-Version I). *Given a policy π, corresponding action-priority vector Φ and a trajectory $\tau' := (s'_0, a'_0, s'_1, \ldots)$. Assume $a'_0 = (m, j, i)$, let $\mathcal{A}_m(s'_0) \subseteq \mathcal{A}(s'_0)$ denote the set of feasible actions where the machine is fixed as M_m. Similarly, let $\mathcal{A}_{(j,i)}(s'_0) \subseteq \mathcal{A}(s'_0)$ denote the set of feasible actions where the job and operation are fixed as J_j and $O_{j,i}$, respectively. Suppose $G(\tau') > V^{\pi}(s'_0)$, then we have new policy π' and its corresponding Φ' where*

$$\phi'_a = \begin{cases} \phi_a + (1-\delta) \sum_{a' \neq a} \mathbb{I}(s'_0, a')\phi_{a'}, & \text{if } a = a'_0 \\ \delta\phi_a, & \text{otherwise} \end{cases} \tag{8}$$

*and $\delta \in (0, 1)$. Here the policy π' and its corresponding Φ' updated according to (8), will perform better, that is $V^{\pi'}(s'_0) > V^{\pi}(s'_0)$ if δ satisfies **CD1**, **CD2** and $\mathcal{A}_m(s'_0)$ satisfies **CD3**:*

CD1 $\frac{\delta^2(1-\delta\eta)}{(1-\eta)(\delta+(1-\delta)\frac{\sum_j \mathbb{I}(s'_0,j)\phi_j}{\sum_j \mathbb{I}_j(s)\phi_j})} < \frac{1-(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}, \forall s \neq s'_0$

CD2 $\sum_{t=1}^{t^{+}_{\tau}-1} \delta^t \frac{1-\delta\eta}{1-\eta} \mathrm{P}^{\pi}_{\tau(t)} < \delta(\sum_{t=1}^{t^{+}_{\tau}-1} \frac{\eta \mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)} \mathrm{P}^{\pi}_{\tau(t)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}} + \frac{\eta \mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}) + \sum_{t=1}^{t^{+}_{\tau}-1} \frac{(1-\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)})\mathrm{P}^{\pi}_{\tau(t)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}} - \frac{\eta \mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}, \forall \tau \in \mathcal{T}_3, \mathcal{T}_5$

CD3 *For any $a \in \mathcal{A}_m(s'_0) \cup \mathcal{A}_{(j,i)}(s'_0)$, after executing action a, the system clock of the resulting state remains the same as $\mathbb{T}'_0$, which is the system clock of s'_0.*

Proof. First, the value function $V^{\pi}(s'_0)$ can be expressed as

$$V^{\pi}(s'_0) = \sum_{\tau \in \mathcal{T}_1} \mathrm{P}^{\pi}_{\tau} G(\tau) + \sum_{\tau \in \mathcal{T}_2} \mathrm{P}^{\pi}_{\tau} G(\tau) + \sum_{\tau \in \mathcal{T}_3} \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)} G(\tau) + \sum_{\tau \in \mathcal{T}_4} \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)} G(\tau) + \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)} G(\tau')$$

Since $G(\tau') > V^{\pi}(s'_0)$, we have

$$G(\tau') > \frac{\sum_{\tau \in \mathcal{T}_1} \mathrm{P}^{\pi}_{\tau} G(\tau) + \sum_{\tau \in \mathcal{T}_2} \mathrm{P}^{\pi}_{\tau} G(\tau) + \sum_{\tau \in \mathcal{T}_3} \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)} G(\tau) + \sum_{\tau \in \mathcal{T}_4} \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)} G(\tau)}{1 - \pi(a'_0|s'_0)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}} \tag{9}$$

Since in our modeling, once an action is executed, it will never be valid again, and together with Lemma 1, then

$$\mathrm{P}^{\pi'}_{\tau|(s'_0,a'_0)} = \mathrm{P}^{\pi}_{\tau|(s'_0,a'_0)} \tag{10}$$

Since

$$\begin{aligned}
\pi'(a_0'|s_0') &= \frac{\phi'_{a_0'}}{\sum_j \mathbb{I}(s_0', j)\phi'_j} \\
&= \frac{\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j}{\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\delta\phi_j + \phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j} \\
&= \frac{\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j}{\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j + \phi_{a_0'}} = \frac{\phi_{a_0'} + (1-\delta)\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j}{\sum_j \mathbb{I}(s_0', j)\phi_j} \\
&= \pi(a_0'|s_0') + \frac{(1-\delta)\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j}{\sum_j \mathbb{I}(s_0', j)\phi_j}
\end{aligned}$$

Since $\eta = \frac{\sum_{j\neq a_0'} \mathbb{I}(s_0', j)\phi_j}{\sum_j \mathbb{I}(s_0', j)\phi_j}$, then

$$\pi'(a_0'|s_0') = \pi(a_0'|s_0') + (1-\delta)\eta \tag{11}$$

and

$$\pi(a_0'|s_0') = \frac{\phi_{a_0'}}{\sum_j \mathbb{I}(s_0', j)\phi_j} = 1 - \eta \tag{12}$$

By (9), (10), (11) and (12),

$$\begin{aligned}
V^{\pi'}(s_0') - V^{\pi}(s_0') &= \sum_{\tau\in\mathcal{T}_1} (\mathrm{P}_\tau^{\pi'} - \mathrm{P}_\tau^{\pi})G(\tau) + \sum_{\tau\in\mathcal{T}_2} (\mathrm{P}_\tau^{\pi'} - \mathrm{P}_\tau^{\pi})G(\tau) + \sum_{\tau\in\mathcal{T}_3} (1-\delta)\eta \mathrm{P}^{\pi}_{\tau|(s_0',a_0')} G(\tau) \\
&+ \sum_{\tau\in\mathcal{T}_4} (1-\delta)\eta \mathrm{P}^{\pi}_{\tau|(s_0',a_0')} G(\tau) + (1-\delta)\eta \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} G(\tau') \\
&> \sum_{\tau\in\mathcal{T}_1} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta)\eta \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\pi(a_0'|s_0')\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi} - \mathrm{P}_\tau^{\pi}\right] G(\tau) + \sum_{\tau\in\mathcal{T}_2} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta)\eta \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\pi(a_0'|s_0')\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi} - \mathrm{P}_\tau^{\pi}\right] G(\tau) \\
&+ \sum_{\tau\in\mathcal{T}_3} \left[(1-\delta)\eta + \frac{(1-\delta)\eta(1-\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\pi(a_0'|s_0')\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}^{\pi}_{\tau|(s_0',a_0')} G(\tau) + \sum_{\tau\in\mathcal{T}_4} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta)\eta \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}{1-\pi(a_0'|s_0')\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi} - \mathrm{P}_\tau^{\pi}\right] G(\tau) \\
&= \sum_{\tau\in\mathcal{T}_1} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi}\right] G(\tau) + \sum_{\tau\in\mathcal{T}_2} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi}\right] G(\tau) \\
&+ \sum_{\tau\in\mathcal{T}_3} \left[\frac{(1-\delta)\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}^{\pi}_{\tau|(s_0',a_0')} G(\tau) + \sum_{\tau\in\mathcal{T}_4} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi}\right] G(\tau)
\end{aligned}$$

Since $G(\tau) < 0$, **CD1** and Lemma 3, we have

$$\begin{aligned}
\sum_{\tau\in\mathcal{T}_1} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi}\right] G(\tau) &> \sum_{\tau\in\mathcal{T}_1} \left[\frac{\delta^2(1-\delta\eta)\sum_j \mathbb{I}_j(s)\phi_j}{(1-\eta)(\delta\sum_j \mathbb{I}_j(s)\phi_j + (1-\delta)\sum_j \mathbb{I}(s_0',j)\phi_j)} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}_\tau^{\pi} G(\tau) \\
&= \sum_{\tau\in\mathcal{T}_1} \left[\frac{\delta^2(1-\delta\eta)}{(1-\eta)(\delta + (1-\delta)\frac{\sum_j \mathbb{I}(s_0',j)\phi_j}{\sum_j \mathbb{I}_j(s)\phi_j})} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}_\tau^{\pi} G(\tau) \\
&\geq 0
\end{aligned}$$

By Lemma 5, we have

$$\mathcal{T}_2 = \bigcup_{\tilde\tau\in\mathcal{T}_3} \{\tilde\tau^{(0)}, \ldots, \tilde\tau^{(t^+_{\tilde\tau}-1)}\}$$

and

$$G(\tilde\tau^{(0)}) = \cdots = G(\tilde\tau^{(t^+_{\tilde\tau}-1)}) = G(\tilde\tau).$$

Furthermore, for any $\tilde\tau^{(t)} \in \{\tilde\tau^{(0)}, \ldots, \tilde\tau^{(t^+_{\tilde\tau}-1)}\}$ with $1 \leq t \leq t^+_{\tilde\tau} - 1$, it follows from (1) that the action a_0' is taken at stage t. Then, by Lemma 4, we obtain

$$\mathrm{P}_\tau^{\pi'} G(\tilde\tau^{(t)}) = \mathrm{P}_\tau^{\pi'} G(\tilde\tau) < \frac{\delta^t(1-\delta\eta)}{1-\eta} \mathrm{P}_\tau^{\pi} G(\tilde\tau).$$

Hence, we have

$$\begin{aligned}
&\sum_{\tau\in\mathcal{T}_2} \left[\mathrm{P}_\tau^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_\tau^{\pi}\right] G(\tau) + \sum_{\tau\in\mathcal{T}_3} \left[\frac{(1-\delta)\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}^{\pi}_{\tau|(s_0',a_0')} G(\tau) & (13) \\
&= \sum_{\tilde\tau\in\mathcal{T}_3} \left(\sum_{\tilde\tau^{(t)}\in\{\tilde\tau^{(0)},\ldots,\tilde\tau^{(t^+_{\tilde\tau}-1)}\}} \left[\mathrm{P}_{\tilde\tau^{(t)}}^{\pi'} + \frac{(1-\delta\eta) \mathrm{P}^{\pi}_{\tau'|(s_0',a_0')} - 1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}} \mathrm{P}_{\tilde\tau^{(t)}}^{\pi}\right] G(\tilde\tau^{(t)}) + \left[\frac{(1-\delta)\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s_0',a_0')}}\right] \mathrm{P}^{\pi}_{\tilde\tau|(s_0',a_0')} G(\tilde\tau) \right) & (14)
\end{aligned}$$

$$= \sum_{\tilde{\tau}\in\mathcal{T}_3}\left(\sum_{\tilde{\tau}^{(t)}\in\{\tilde{\tau}^{(0)},\ldots,\tilde{\tau}^{(t^+_{\tilde{\tau}}-1)}\}}\left[\mathrm{P}^{\pi'}_{\tilde{\tau}^{(t)}}+\frac{(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}-1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tilde{\tau}^{(t)}}\right]+\left[\frac{(1-\delta)\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\right]\mathrm{P}^{\pi}_{\tilde{\tau}|(s'_0,a'_0)}\right)G(\tilde{\tau}) \tag{15}$$

$$> \sum_{\tilde{\tau}\in\mathcal{T}_3}\left(\sum_{t=1}^{t^+_{\tilde{\tau}}-1}\left[\frac{\delta^t(1-\delta\eta)}{1-\eta}\mathrm{P}^{\pi}_{\tilde{\tau}^{(t)}}-\frac{1-(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tilde{\tau}^{(t)}}\right]+\left[\frac{(1-\delta)\eta}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\right]\mathrm{P}^{\pi}_{\tilde{\tau}|(s'_0,a'_0)}\right)G(\tilde{\tau}) \tag{16}$$

$$> 0 \tag{17}$$

The last inequality follows from **CD2**. Similarly, for the set $\mathcal{T}_4$, applying Lemma 4 and Lemma 5, we obtain

$$\begin{aligned}\sum_{\tau\in\mathcal{T}_4}\left[\mathrm{P}^{\pi'}_{\tau}+\frac{(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}-1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tau}\right]G(\tau) &= \sum_{\tau'^{(t)}\in\{\tau'^{(0)},\ldots,\tau'^{(t^+_{\tau'}-1)}\}}\left[\mathrm{P}^{\pi'}_{\tau'^{(t)}}+\frac{(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}-1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tau'^{(t)}}\right]G(\tau'^{(t)})\\ &= \sum_{t=1}^{t^+_{\tau'}-1}\left[\mathrm{P}^{\pi'}_{\tau'^{(t)}}+\frac{(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}-1}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tau'^{(t)}}\right]G(\tau')\\ &> \sum_{t=1}^{t^+_{\tau'}-1}\left[\frac{\delta^t(1-\delta\eta)}{1-\eta}\mathrm{P}^{\pi}_{\tau'^{(t)}}-\frac{1-(1-\delta\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}{1-(1-\eta)\mathrm{P}^{\pi}_{\tau'|(s'_0,a'_0)}}\mathrm{P}^{\pi}_{\tau'^{(t)}}\right]G(\tau')\\ &> 0\end{aligned}$$

The last inequality follows from **CD2**. Consequently, we have proven that $V^{\pi'}(s'_0) > V^{\pi}(s'_0)$. ■

IV. Numerical Results

In this section, we present a comprehensive evaluation of the proposed scheduling framework through extensive testing on established FJSP benchmarks. The experiments are conducted on Hurink's instances, which include three variants—edata, rdata, and vdata—each representing different levels of machine assignment flexibility. Our experimental design examines two primary benchmark instances representing different scheduling complexities: Brandimarte's mk instances featuring medium flexibility, and Hurink's modified la instances that systematically test performance across varying flexibility levels. The performance is assessed using absolute and relative gaps between the proposed method (APDPO) and the baseline (OR-Tools), providing insights into the algorithm's effectiveness under varying flexibility conditions.

A. Description of Evaluation Cases

Brandimarte's mk instances represent medium-scale industrial scenarios with partial machine flexibility, featuring 10-20 jobs and 5-15 machines, where each operation can be assigned to 3-6 eligible machines. To further evaluate flexibility adaptation, we conduct the experiments on Hurink's la instances which is adapted from classical JSSP formulations, provides particularly valuable insights through its three distinct flexibility variants. The **edata** configuration represents minimal flexibility conditions where operations typically have only one or two eligible machines. In contrast, **rdata**, representing semi-automated workshops where 60-70% of operations can be processed on 2-3 machines. Furthermore, **vdata** pushes flexibility to its limits by allowing all operations to be processed on multiple machines, averaging half the total machines $\frac{1}{2}m$ per operation. These instances span problem scales from 10×5 to 30×15 job-machine combinations, thereby providing a structured framework for evaluating algorithm performance across varying degrees of flexibility.

B. Experimental Results

The performance evaluation benchmarks our method against three established standards: Google OR-Tools solutions, theoretically computed lower bounds (LB) following [2], and the best known upper bounds (UB) from the literature. While OR-Tools is capable of attaining the true optimal solution at a high computational cost, and LBs represent theoretical limits that are not always practically attainable, our approach consistently achieves solutions within approximately 6% of the LB values. Furthermore, we provide detailed results for each individual instance. The results demonstrate consistent performance across all flexibility regimes, with particular strength in high-flexibility vdata configurations where average gaps of just 7.65 (absolute) and 0.89% (relative) were achieved. Notable exact matches occur in several instances, including la5 (edata) and la16 (vdata), where our solutions attain the theoretical lower bounds of 503 and 717 respectively. These optimal results coincide with cases where UB, LB and OR-Tools values converge, confirming the method's ability to reach theoretical limits when conditions permit.

TABLE I: Performance on Public Benchmarks, mk series

mk	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
mk01	41	36	39	40	5	13.89%
mk02	27	24	26	26	3	12.50%
mk03	**204**	204	204	204	0	0.00%
mk04	62	48	60	60	14	29.17%

(Continued on next page)

(Continued from previous page)

mk	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
mk05	174	168	172	173	6	3.57%
mk06	60	33	58	58	27	81.82%
mk07	144	133	139	140	11	8.27%
mk08	**523**	523	523	523	0	0.00%
mk09	320	299	307	307	21	7.02%
mk10	226	165	197	208	61	36.97%
avg.	961.53	923.33	934.28	933.53	38.20	4.16%

TABLE II: Performance on Public Benchmarks, la series (rdata)

rdata	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
la1	588	570	571	570	18	3.16%
la2	556	529	530	529	27	5.10%
la3	484	477	478	477	7	1.47%
la4	521	502	502	502	19	3.78%
la5	467	457	457	457	10	2.19%
la6	819	799	799	799	20	2.50%
la7	766	749	750	749	17	2.27%
la8	782	765	765	765	17	2.22%
la9	864	853	853	853	11	1.29%
la10	823	804	804	804	19	2.36%
la11	1081	1071	1071	1071	10	0.93%
la12	955	936	936	936	19	2.03%
la13	1051	1038	1038	1038	13	1.25%
la14	1090	1070	1070	1070	20	1.87%
la15	1104	1089	1090	1089	15	1.38%
la16	752	717	717	717	35	4.88%
la17	670	646	646	646	24	3.72%
la18	689	666	666	666	23	3.45%
la19	713	647	700	700	66	10.20%
la20	770	756	756	756	14	1.85%
la21	850	808	835	842	42	5.20%
la22	773	737	760	766	36	4.88%
la23	866	816	842	849	50	6.13%
la24	843	775	808	805	68	8.77%
la25	808	752	791	785	56	7.45%
la26	1154	1056	1061	1071	98	9.28%
la27	1112	1085	1091	1097	27	2.49%
la28	1101	1075	1080	1084	26	2.42%
la29	1022	993	998	997	29	2.92%
la30	1087	1068	1078	1080	19	1.78%
la31	1594	1520	1521	1523	74	4.87%
la32	1680	1657	1659	1659	23	1.39%
la33	1595	1497	1499	1498	98	6.55%
la34	1613	1535	1536	1536	78	5.08%
la35	1590	1549	1550	1550	41	2.65%
la36	1087	1016	1030	1023	71	6.99%
la37	1101	989	1077	1062	112	11.32%
la38	976	943	962	954	33	3.50%
la39	1087	966	1024	1011	121	12.53%
la40	977	955	970	955	22	2.30%
avg.	961.53	923.33	934.28	933.53	38.20	4.16%

TABLE III: Performance on Public Benchmarks, la series (edata)

edata	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
la1	621	609	609	609	12	1.97%
la2	692	655	655	655	37	5.65%
la3	578	550	550	550	28	5.09%
la4	596	568	568	568	28	4.93%
la5	**503**	503	503	503	0	0.00%
la6	**833**	833	833	833	0	0.00%
la7	788	762	762	762	26	3.41%
la8	860	845	845	845	15	1.78%
la9	912	878	878	878	34	3.87%
la10	**866**	866	866	866	0	0.00%
la11	1115	1087	1103	1103	28	2.58%
la12	968	960	960	960	8	0.83%
la13	**1053**	1053	1053	1053	0	0.00%
la14	1124	1123	1123	1123	1	0.09%
la15	1151	1111	1111	1111	40	3.60%
la16	931	892	892	892	39	4.37%
la17	739	707	707	707	32	4.53%
la18	862	842	842	842	20	2.38%
la19	843	796	796	796	47	5.90%
la20	904	857	857	857	47	5.48%
la21	1056	895	1017	1009	161	17.99%
la22	904	832	882	880	72	8.65%
la23	980	950	950	950	30	3.16%
la24	954	881	909	908	73	8.29%
la25	1007	894	941	936	113	12.64%
la26	1167	1089	1125	1106	78	7.16%
la27	1219	1181	1186	1181	38	3.22%
la28	1187	1116	1149	1144	71	6.36%
la29	1169	1058	1118	1113	111	10.49%
la30	1370	1147	1204	1194	223	19.44%
la31	1560	1523	1539	1532	37	2.43%
la32	1717	1698	1698	1698	19	1.12%
la33	1581	1547	1547	1547	34	2.20%
la34	1646	1592	1604	1599	54	3.39%
la35	1780	1736	1736	1736	44	2.53%
la36	1187	1006	1162	1160	181	17.99%
la37	1402	1355	1397	1397	47	3.47%
la38	1171	1019	1144	1141	152	14.92%
la39	1209	1151	1184	1184	58	5.04%
la40	1157	1034	1174	1144	123	11.90%
avg.	1059.05	1005.03	1029.48	1026.80	54.03	5.47%

TABLE IV: Performance on Public Benchmarks, la series (vdata)

vdata	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
la1	573	570	570	570	3	0.53%
la2	536	529	529	529	7	1.32%
la3	484	477	477	477	7	1.47%
la4	506	502	502	502	4	0.80%
la5	469	457	457	457	12	2.63%
la6	808	799	799	799	9	1.13%
la7	760	749	749	749	11	1.47%
la8	770	765	765	765	5	0.65%
la9	861	853	853	853	8	0.94%
la10	808	804	804	804	4	0.50%
la11	1081	1071	1071	1071	10	0.93%

vdata	APDPO	LB	UB	OR-Tools	gap-abs	gap-ratio
la12	938	936	936	936	2	0.21%
la13	1046	1038	1038	1038	8	0.77%
la14	1075	1070	1070	1070	5	0.47%
la15	1100	1089	1089	1089	11	1.01%
la16	**717**	717	717	717	0	0.00%
la17	**646**	646	646	646	0	0.00%
la18	**663**	663	663	663	0	0.00%
la19	618	617	617	617	1	0.16%
la20	**756**	756	756	756	0	0.00%
la21	815	800	806	805	15	1.88%
la22	753	733	739	742	20	2.73%
la23	823	809	815	819	14	1.73%
la24	794	773	777	780	21	2.72%
la25	765	751	756	759	14	1.86%
la26	1068	1052	1054	1055	16	1.52%
la27	1101	1084	1085	1088	17	1.57%
la28	1080	1069	1070	1072	11	1.03%
la29	999	993	994	994	6	0.60%
la30	1082	1068	1069	1074	14	1.31%
la31	1528	1520	1520	1521	8	0.53%
la32	1663	1657	1658	1659	6	0.36%
la33	1504	1497	1497	1499	7	0.47%
la34	1540	1535	1535	1536	5	0.33%
la35	1565	1549	1549	1551	16	1.03%
la36	**948**	948	948	948	0	0.00%
la37	987	986	986	986	1	0.10%
la38	943	943	943	943	0	0.00%
la39	930	922	922	922	8	0.87%
la40	955	955	955	955	0	0.00%
avg.	926.45	918.80	919.65	920.40	7.65	0.89%

References

[1] M. Spivak, *Calculus*. Cambridge University Press, 2006.

[2] N. B. Ho and J. C. Tay, "Solving multiple-objective flexible job shop problems by evolution and local search," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 5, pp. 674–685, 2008.