SCIENCE CHINA
Information Sciences

• **Supplementary File** •

A multi-mode multi-skill project scheduling reformulation for reconnaissance mission planning

Junqi CAI[1], Zhihong PENG[1*], Sijian LIAO[1] & Shuxin DING[2]

[1] *School of Automation, Beijing Institute of Technology, Beijing,* 100081*, China;*
[2] *Signal and Communication Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing,* 100081*, China*

Appendix A Background information

Figure A1 shows an example of a reconnaissance mission. To the best of the author's knowledge, there are currently no papers modeling the reconnaissance mission planning problem with the following characteristics at the same time: (1) The mission which consists of several tasks with precedence relations, the precedence relations between tasks must be respected. (2) Each reconnaissance agent has multiple type of skills, for a specific type of skill, the skill units masted by different agents are usually different. (3) The processing time of each task in a reconnaissance mission is a problem specific nonlinear function of the agents assigned to that task and the type of skill each agent performs in that task. (4) Considering the transfer time between different tasks. Based on the requirement of sensors combinations and worker skills, there are three types of reconnaissance capacities: (1) Terrain data collection (TDC): collecting the terrain feature to reconstruct it, the data is typically collected by using lidar, sonar, and photogrammetric sources. (2) Urban area search (UAS): search for required features in the urban environment, which can be people, animals or machinery. (3) Woodland area search (WAS): search for required features in the woodland environment. Compared to urban area, the woodland area is covered with more trees and bushes, some additional sensors like infrared equipment and synthetic-aperture radar is required to work together. In this paper, TDC, UAS and WAS are abstracted as three types of skill required to perform reconnaissance missions. A number is usually specified for each area to limit the number of agents working in this area, the longest processing time for reconnaissance task is also specified.

Figure A2 is an example of the decomposition of reconnaissance tasks based on reconnaissance skill requirements. The blue area represents a reconnaissance task in a reconnaissance mission. The light yellow area is the area that need to conduct terrain data collection, which is the same area as the blue area. The red and green part represents the urban search area and the woodland search area respectively. Apparently, the area of TDC may overlap with UAS and WAS, but there can't be any overlap between UAS and WAS. Figure A3 shows a early phase of a reconnaissance task execution. The green point is the start point, where all the reconnaissance resources enter the reconnaissance area together as the bold green line shows. When all the resources reach the start point, instead of performing reconnaissance task immediately, the reconnaissance resources are moved to different locations across the area, and then the sensors is adjusted to the corresponding working status to start the reconnaissance task. The time for reconnaissance resources transferred between different tasks are defined as "transfer time", the time used for the this kind of spread out and sensors adjustment inside an reconnaissance task area is defined as "setup time". Using the indices and symbols presented below, the information used to describe a reconnaissance mission includes: (1) The precedence relations and transfer times between tasks. (2) The area of TDC,UAS and WAS in each task (denoted as A_j^l). (3) The setup time for each task (denoted as Γ_j). (4) maximum number of agents searing in the task area (denoted as MR_j^l). (5) L denotes the set of skill types, $l \in L$ is the elements of L, L_j denotes the set of skill required to perform task j. (6) The units of skill l mastered by reconnaissance agent k is denoted as u_{kl}, the physical meaning of u_{kl} is the size of the corresponding area searched by agent k using skill l per unit time (time step). (7) R denotes the set of agents. For a specific task j, RA_j^l denotes the resources which are allocated to perform ability l in task j. The processing time of a reconnaissance task j (denoted as p_j) can be calculated as :

$$\begin{aligned} p_j &= \max_{l \in L_j} p_j(l) \\ p_j(l) &= \Gamma_j + \frac{A_j^l}{\sum_{k \in RA_j^l} u_{kl}} \\ \textbf{s.t.} \quad & |RA_j^l| \leqslant MR_j^l, l \in L. \end{aligned} \tag{A1}$$

* Corresponding author (email: peng@bit.edu.cn)

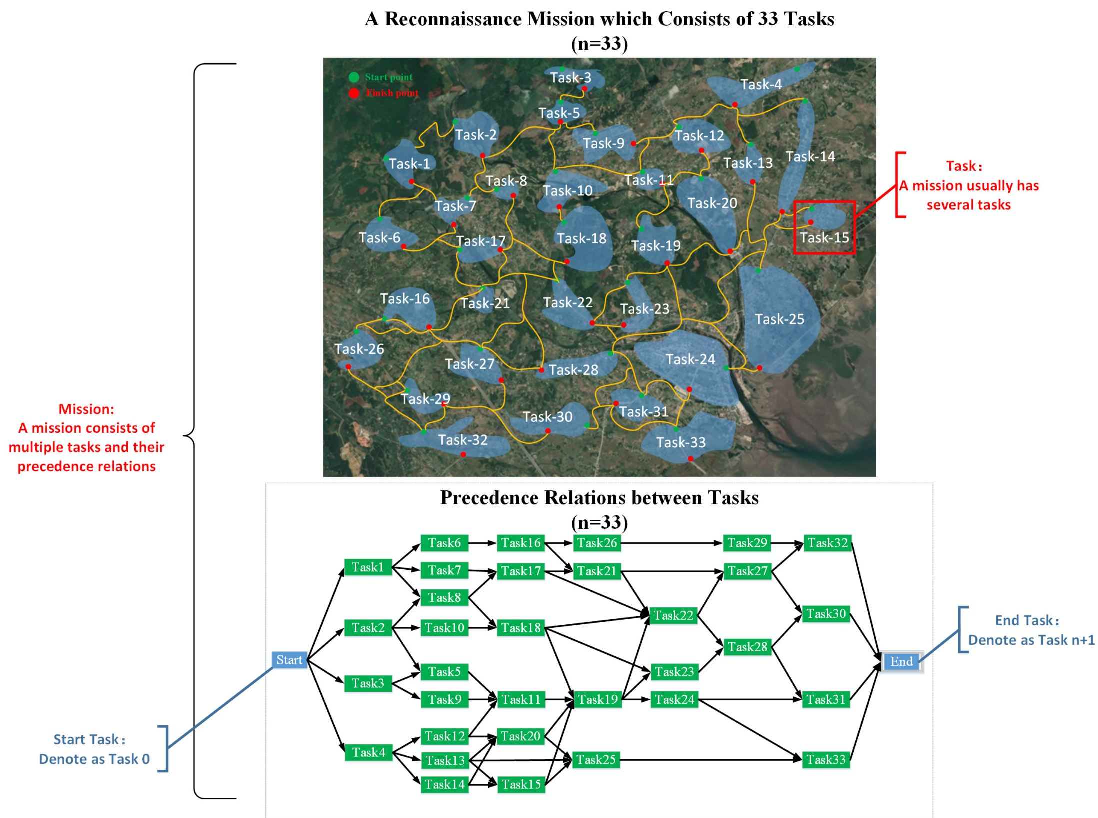

Figure A1 An example of reconnaissance mission

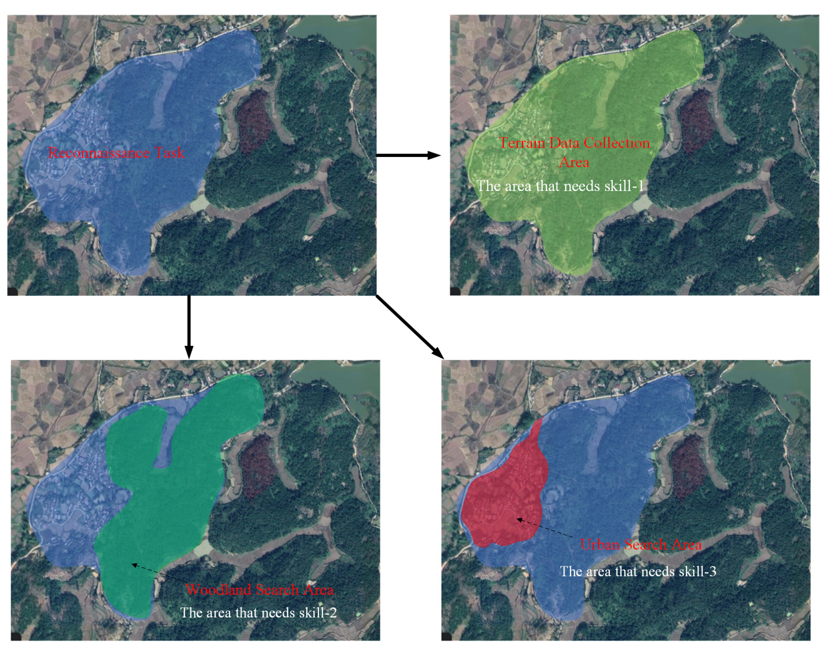

Figure A2 Reconnaissance task division by different skill requirement

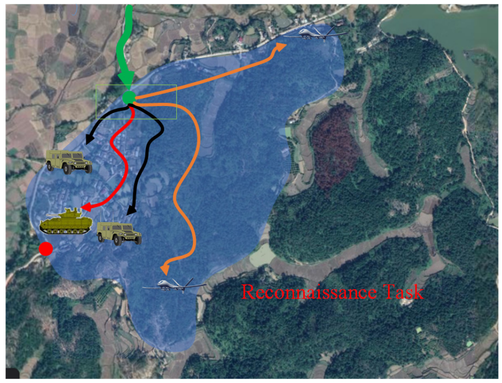

Figure A3 Example of setup time

Appendix B Detail of the problem formulation

The objective of reconnaissance mission planning problem is to schedule all tasks satisfying the precedence and resource constraints in such a way that the makespan of the mission is minimized. The precedence relationship between tasks is modeled as a task on node network $G = (V, E)$, where V represents a set of tasks, E indicates the precedence relationship between tasks.

Appendix B.1 Assumptions

The following assumptions are made for reconnaissance mission planning problem:

- Preemption is not allowed, that is, if an task is being processed it must be processed to the end of that task.
- Each agent can only contribute one skill to a specific task.
- The dummy tasks have no resource requirement and a zero processing time.
- An task can start to be processed only if all the resources assigned to it have been transferred to the start point of that task.

Appendix B.2 Indices

- i, j: index of tasks, $i, j = 0, 1, 2, ..., n, n + 1$.
- l: index of skills, $l = 1, 2, ..., LN$.
- k: index of reconnaissance agent (resource), $k = 1, 2, ..., K$.
- t: index of discrete time step, $t = 0, 1, 2, ..., UB$, where UB denotes the upper bound of discrete time step.

Appendix B.3 Notations

The notations used in problem formulation is listed in Table B1. Noted that n represent the number of non-dummy tasks. The dummy task includes start task and end task, start task is denoted as task 0, end task is denoted as task $n + 1$.

Appendix B.4 Decision Variable

- s_{jt}: 1 if task j to be processed at time t, 0 otherwise.
- x_{jklt}: 1 if resource k processes task j with skill l at time t and t is the start time of task j, 0 otherwise. Noted that each agent can only perform one skill to a specific task, which means each agent can be assigned to a task at most once.
- z_{ijk}: 1 if resource k is transferred for i to j, 0 otherwise.

Appendix B.5 Mathematical Formulation

We reformulate the model presented by [1]. The mathematical formulation of reconnaissance mission planning problem can be formulated as follows:

$$\min \sum_{t=ES_{n+1}}^{LS_{n+1}} ts_{(n+1)t} \tag{B1}$$

$$\textbf{s.t.} \quad p_j = \max_{l \in L_j} \{\Gamma_j + \frac{A_j^l}{\sum_{k \in R} \sum_{t \in T} u_{kl} \times x_{jklt}}\}, \qquad j \in V, \tag{B2}$$

$$\sum_{k \in R} \sum_{t \in T} x_{jklt} \leqslant MR_j^l, \qquad j \in V, l \in L, \tag{B3}$$

Table B1 The notations

Symbols	Description
n	the number of non-dummy tasks.
$V = \{0, \ldots, i, \ldots, j, \ldots, n+1\}$	set of tasks, task 0 and n+1 are dummy tasks.
$R = \{1, \ldots, k, \ldots K\}$	set of resources, in reconnaissance scheduling problem, resources represents reconnaissance agent.
$L = \{1, \ldots, l, \ldots LN\}$	set of skills which corresponding to reconnaissance capacity.
P_j, P_j^I	tasks which are the direct and indirect predecessor of task j respectively.
S_j, S_j^I	tasks which are the direct and indirect predecessor of task j respectively.
L_j	skills which are required by task j.
L^k	skills which are mastered by resource k.
V_k	tasks requiring skills mastered by resource k.
R_j	resources which can contribute at least one skill required by task j.
A_j^l	the size of area which requires skill l to perform in task j.
RA_j^l	resources which are allocated to perform skill l in task j.
MR_j^l	the maximum number of resources working in area which require skill l in task j.
r_{jl}	the units of skill l required to process task j.
u_{kl}	the units of skill l that resource k masters.
p_j	the processing time of task j.
p_j^{max}	the longest processing time of task j.
Γ_j	the setup time of task j.
Δ_{ij}	transfer time between task i to task j.
$T = \{0, \ldots, t, \ldots, UB\}$	set of discrete time steps.
$ES_j, LS_j,$	earliest and latest start time of task j.

$$\sum_{t=ES_j}^{LS_j} s_{jt} = 1, \qquad j \in V, \quad \text{(B4)}$$

$$\sum_{t=ES_j}^{LS_j} ts_{jt} - \sum_{t=ES_i}^{LS_i} (t+p_i)s_{it} \geqslant 0, \qquad i \in V \backslash \{n+1\}, j \in V \backslash P_i^I, \quad \text{(B5)}$$

$$\sum_{t=ES_j}^{LS_j} ts_{jt} - \sum_{t=ES_i}^{LS_i} (t+p_i)s_{it} - (UB + \Delta_{ij}) \cdot z_{ijk} \geqslant -UB,$$

$$i \in V \backslash \{n+1\}, j \in V \backslash P_i^I, k \in R, \quad \text{(B6)}$$

$$\sum_{i \in V \backslash S_j^I} z_{ijk} \geqslant \sum_{e \in V \backslash P_j^I} z_{jek}, \qquad j \in V \backslash \{0\}, k \in R_j, \quad \text{(B7)}$$

$$z_{ijk} = 0, \qquad i \in V,\ j \in P_j^I \cup \{i\}, \quad \text{(B8)}$$

$$\sum_{j \in V} z_{ijk} \leqslant 1, \qquad i \in V, k \in R_i \cap R_j, \quad \text{(B9)}$$

$$\sum_{l \in L} \sum_{t=ES_j}^{LS_j} x_{jklt} = \sum_{i \in V \backslash \{n+1\}} z_{ijk}, \qquad j \in V \backslash \{0, n+1\}, k \in R_i \cap R_j, \text{(B10)}$$

$$\sum_{l \in L} x_{jklt} \leqslant s_{jt}, \qquad j \in V, k \in R, t \in \{ES_j, \ldots, LS_j\}, \text{(B11)}$$

$$\sum_{j \in V} \sum_{\tau=\max\{ES_j, t-p_j+1\}}^{\min\{LS_j, t\}} \sum_{l \in L} x_{jkl\tau} \leqslant 1, \qquad k \in R, t \in T, \text{(B12)}$$

$$\sum_{t=ES_j}^{LS_j} \sum_{k \in R} x_{jklt} \cdot u_{kl} \geqslant \frac{A_j^l}{p_j^{max} - \Gamma_j}, \qquad j \in V \backslash \{0, n+1\}, l \in L, \text{(B13)}$$

$$s_{jt} \in \{0, 1\}, \qquad j \in V, t \in \{ES_j, \ldots, LS_j\}, \text{(B14)}$$

$$x_{jklt} \in \{0, 1\}, \qquad j \in V, k \in R, l \in L, t \in \{ES_j, \ldots, LS_j\}, \text{(B15)}$$

$$z_{ijk} \in \{0, 1\}, \qquad i \in V \backslash \{0\}, j \backslash \{S_j^I\}, k \in R_i \cap R_j. \text{(B16)}$$

In this model, the objective functidon (B1) is to minimize the makespan of the project, where t represents time step and $s_{(n+1)t}$ is a decision variable which describe the start time of the dummy end task $n+1$, $s_{(n+1)t}$ equal 1 when the dummy end task $n+1$ starts at time t. The value of function (B1) is the start time of the dummy end task, which equals to the makespan of the reconnaissance mission, the objective function (B1) in this paper refers to the equation (12) in [1]. Constraints (B2) explain how to compute the actual execution time of a reconnaissance task, where A_j^l denotes the the size of area which requires skill l to perform in task j, x_{jklt} is a decision variable of agent k, x_{jklt} equals to 1 if agent k is assigned to perform skill l to task j at time step t and t equals to the start time of task j, u_{kl} denotes the units of skill l mastered by agent k. $\sum_{k\in R}\sum_{t\in T} u_{kl}\times x_{jklt}$ in constraints (B2) denotes the total units of skill l assigned to perform task j, the physical meaning of u_{kl} is the size of the corresponding area searched by agent k using skill l per time step, therefore the $\sum_{k\in R}\sum_{t\in T} u_{kl}\times x_{jklt}$ in constraints (B2) is also equals to the total size of area per time step which search by agents assigned to task j and assigned to perform skill l. Constraints (B3) restricts the maximum number of resources working in task area. Constraints (B4) ensure that every task starts exactly once. Constraints (B5) make sure that the precedence relations must be satisfied, which means one task can start to be processed only if all its predecessors have been finished. Constraints (B6) means that the transfer time between reconnaissance tasks is taken into consideration. Constraints (B7)(B8)(B9) are resource flow constraints. Constraints (B7) and (B8) ensure that if an agent is transferred to a task, the current position of the agent must be the predecessor or indirect predecessor of that task, which avoids inefficient assignments for agents that be assigned to transfer from a successor task to a predecessor task. Constraints (B9) means that an agent can only be assigned to a task at most once. Constraints (B10) shows the relationship between variable x and variable z, which means that if an agent is assigned to perform a task, it must be transferred from another task. Constraints (B11) shows a reconnaissance resource can only provide one type of skill to a task. Constraints (B12) restrict that an agent can not perform more than one task at the same time. Constraints (B13) ensure that enough resources are assigned to the reconnaissance task to avoid the execution time exceeding the maximum execution time. Constraints (B14)(B15)(B16) define the domain of the decision variables. This model is build in an easily understandable way for readers, but constraints (B5)(B6) include nonlinear part $p_i\times s_{it}$ and p_i is also coupled with decision variable x_{jklt}, which makes this model infeasible for commercial optimizer.

Appendix C Proof of the Proposition

Proposition 1. By adding a new decision variable $w_{it}=p_i\times s_{it},\ i\in V,\ t\in T$, the nonlinear component $p_i\times s_{it}$ in (B5)(B6) can be converted to linear convex expression by replacing it with w_{it} and its constraints: (1) $w_{it}\leqslant p_i^{max}\cdot s_{it}$; (2) $w_{it}\leqslant p_i^{min}s_{it}+p_i+p_i^{min}$; (3) $w_{it}\geqslant p_i^{min}\cdot s_{it}$; (4) $w_{it}\geqslant p_i^{max}\cdot s_{it}+p_i-p_i^{max}$. ($p_i^{max}$, p_i^{min} is a upper bound and a lower bound of p_i respectively, p_i^{max} is specified in task information, p_i^{min} can be obtained by using Eq.(A1))

Proof. Proposition 1 is the implementation result of McCormick Envelopes method. Let $w_{it}=p_i\cdot s_{it}$, obviously $p_i^{min}\leqslant w_{it}\leqslant p_i^{max}$, $0\leqslant s_{it}\leqslant 1$.

1. Let $a=p_i^{max}-p_i, b=s_{it}$, then $a\times b\geqslant 0$. Therefore $(p_i^{max}-p_i)\cdot s_{it}=p_i^{max}s_{it}-p_is_{it}=p_i^{max}s_{it}-w_{it}\geqslant 0$, $w_{it}\leqslant p_i^{max}\cdot s_{it}$
2. Let $a=p_i-p_i^{min}, b=1-s_{it}$, then $a\times b\geqslant 0$. Therefore $(p_i-p_i^{min})\cdot(1-s_{it})=p_i-p_i^{min}-s_{it}p_i+p_i^{min}s_{it}=p_i-p_i^{min}-w_{it}+p_i^{min}s_{it}\geqslant 0$, $w_{it}\leqslant p_i^{min}s_{it}+p_i+p_i^{min}$.
3. Let $a=p_i-p_i^{min}, b=s_{it}$, then $a\times b\geqslant 0$. Therefore $(p_i-p_i^{min})\cdot s_{it}=p_is_{it}-p_i^{min}s_{it}=w_{it}-p_i^{min}s_{it}\geqslant 0$, $w_{it}\geqslant p_i^{min}\cdot s_{it}$.
4. Let $a=p_i^{max}-p_i, b=1-s_{it}$, then $a\times b\geqslant 0$. Therefore $(p_i^{max}-p_i)(1-s_{it})=p_i^{max}-p_i-p_i^{max}s_{it}+p_is_{it}=p_i^{max}-p_i-p_i^{max}s_{it}+w_{it}\geqslant 0$, $w_{it}\geqslant p_i^{max}\cdot s_{it}+p_i-p_i^{max}$.

Appendix D NP-hardness of the problem

In this part, the reconnaissance mission planning problem is proven to be NP-hard.

Theorem 1. The proposed reconnaissance mission planning problem is NP-hard.

Proof. For a reconnaissance mission planning problem investigated in this paper, it can be simplified to RCPSP with the following settings: (1) The transfer time between tasks are all set to 0; (2) Only one type of skill is involved; (3) The processing time for each task is fixed, which means each task has only one mode. Obviously, RCPSP is a special simplified case of reconnaissance mission planning problem. RCPSP is proven to be the class of strongly NP-hard problem in [**?**]. Therefore, the reconnaissance mission planning problem investigated in this paper is NP-hard.

Appendix E Detail of the experiment

In this part, the computational experiment is conducted to assess the quality of proposed algorithm. The heuristic algorithm is coded in Python programming language. All experiments are conducted on a machine running an Intel Core i5-7400 3.4GHz CPU with 16GB of RAM. We first present some parameters used to describe the computational instance and the way to generate test instances, then we present the detail of the test data, at last we show and analysis the result of computational experiment.

Appendix E.1 Instance Generation

As the reconnaissance mission planning problem is modeled as multi-mode multi-skill RCPSP, some common parameters usually used in traditional project scheduling problem are introduced to describe the difference between test instances: (1) $maxSkill$: the maximum types of skill a resource can master. (2) $nStart$: the maximum number of tasks which are the direct successor of start task. (3) $nFinish$: the maximum number of tasks which are direct predecessor of end task. (4) $MaxPred$: maximum number of predecessor for each task. (5) $MaxSucc$: maximum number of successors of each task. (6) NC: the network complexity, which is defined by the average number of successor of each task in the project (reconnaissance mission); (7) $nAct$: the number of tasks in a project (reconnaissance mission). (8) K: The number of resource (reconnaissance agent).

Considering the multi-skill nature of reconnaissance agent, some new parameters are proposed to describe the feature of resource skills. MSU_l: The maximum skill unit of skill l that a resource can master. TS_l: The total skill supply of skill l.

Some parameters are used in other papers, but they are modified in this paper to fit the reconnaissance mission planning problem. Skill factor (SF) is used to describe the relationship between the number of skill types required to perform a task ($|L_j|$) and total number of skill types ($|L|$), $|L_j| = \lceil |L| \times SF \rceil$. By introducing parameter Resource Strength $RSS_l = \frac{TS_l}{\sum_{j\in V} SR_l^{min}(j)}$ for each skill $l \in L$, we can control the total skill requirement of the reconnaissance mission.

As there is no benchmark for reconnaissance mission planning problem, we make some extensions for the instance generator proposed by [3] to generate the test instance. Firstly, use the resource generation algorithm in [3] to generate the reconnaissance resources. Due to the resource in reconnaissance mission is reconnaissance agent, to fit the character of reconnaissance agent, the skill level of each resource is re-sampling by a Gaussian Distribution. For each skill re-sampling process, the mean equals to the value generated by resource generation algorithm in [3], the standard deviation is set to 0.9. Then executing "activity generation" algorithm in [3], the precedence network ($G = (V, E)$), the longest processing time (p_j^{max}) for each task and the minimum skill requirement ($SR_l^{min}(j)$) are obtained. According to statistics, setup time of a reconnaissance is usually between 0.12 to 0.18 times the total execution time. So the setup of a task is generate by $\beta \times p_j^{max}$, β is randomly selected from $[0.1, 0.2]$. An interval defined by $[NCR, MCR]$ is introduced to specified the maximum number of resources working in task area A_j^l, α is randomly selected from $[NCR, MCR]$, $MR_j^l = \frac{A_j^l}{max_{k\in R} u_{kl}} \times \alpha$.

Appendix E.2 Test Data

To test performance by using exact method, a tiny instance set with 8 small size instances is used, the details of the instance set is presented in Appendix E.3.1. To test the performance of heuristic algorithms, we performed the computational experiments using three different sets of instances that we name Set 1, Set 2 and Set 3. Each set corresponds to the 320 instances (totally 960 instances) generated by using the method in Appendix E.1. According to statistics, the number of tasks in a reconnaissance mission changes from 38 to 101, the number of reconnaissance agents is usually between 33 to 81, the value of α is usually between 0.1 to 0.25. Based on the statics, some basic parameters to generated instances Set 1, Set 2 and Set 3 are listed in Table E1.

Table E1 The basic parameters for instance set

Parameter	Set 1	Set 2	Set 3
$nAct$	40	80	120
$nStart$	4	5	6
$nFinish$	4	5	6
$MaxSucc$	6	8	10
$MaxPred$	6	8	10
$maxSkill$	3	3	3
K	30	60	90
L	3	3	3
MSU	40	40	40
TS	{3000,5200,7720}	{6220, 10400, 16020}	{10480, 14280, 23640}
NCR	0.1	0.1	0.05
MCR	0.2	0.25	0.2

Five main features which may have relatively more influence on the complexity of reconnaissance mission instance are considered, which are: (1) NC: The network complexity. (2) $nAct$: The number of tasks in a reconnaissance mission. (3) K: The number of resources, in other words, the number of reconnaissance agents. (4) SF: Skill factor. (5) RSS: The resource strength. To make the information more clear for different set of instances, we briefly introduce the main characteristics of the instances of Set 1, Set 2 and Set 3 in Table E2. Each combination of the main parameters is associated with 40 different instances.

Table E2 Main parameters of instance set

Instance	Set 1					Set 2					Set 3				
	NC	$nAct$	K	SF	RSS	NC	$nAct$	K	SF	RSS	NC	$nAct$	K	SF	RSS
1	1.5	40	30	0.50	{0.12, 0.20, 0.18}	1.8	80	60	1.00	{0.11, 0.15, 0.14}	1.8	120	90	1.00	{0.13, 0.22, 0.15}
2	1.5	40	30	0.50	{0.18, 0.25, 0.24}	1.8	80	60	1.00	{0.19, 0.21, 0.18}	1.8	120	90	1.00	{0.18, 0.23, 0.18}
3	1.5	40	30	0.75	{0.12, 0.20, 0.18}	1.8	80	60	0.75	{0.11, 0.15, 0.14}	1.8	120	90	0.75	{0.13, 0.22, 0.15}
4	1.5	40	30	0.75	{0.18, 0.25, 0.24}	1.8	80	60	0.75	{0.19, 0.21, 0.18}	1.8	120	90	0.75	{0.18, 0.23, 0.18}
5	1.7	40	30	0.50	{0.12, 0.20, 0.18}	2.1	80	60	1.00	{0.11, 0.15, 0.14}	2.1	120	90	1.00	{0.13, 0.22, 0.15}
6	1.7	40	30	0.50	{0.18, 0.25, 0.24}	2.1	80	60	1.00	{0.19, 0.21, 0.18}	2.1	120	90	1.00	{0.18, 0.23, 0.18}
7	1.7	40	30	0.75	{0.12, 0.20, 0.18}	2.1	80	60	0.75	{0.11, 0.15, 0.14}	2.1	120	90	0.75	{0.13, 0.22, 0.15}
8	1.7	40	30	0.75	{0.18, 0.25, 0.24}	2.1	80	60	0.75	{0.19, 0.21, 0.18}	2.1	120	90	0.75	{0.18, 0.23, 0.18}

Appendix E.3 Test results

Appendix E.3.1 *Small Size Instance with Exact Method*

In this part, 8 small size instances are solved with exact method. Using commercial optimizer (Gurobi 8.1.1) with the linear model to solve reconnaissance mission planning problem. Since this part is a small size instance experiment which is not included in the statistics mentioned in Appendix E.2, in order to make the small size instance do not violate the constraints in real-world reconnaissance mission, the parameters of small size instance are inconsistent with the statistics. The computation time limit for commercial optimizer is set to 7200s. Some parameters of experiment result are shown in Table E3: "Makespan" represent the finish time of the reconnaissance mission, "CPU time" denotes the computation time, "Optimal" shows whether optimal solution is obtained within the time limit.

Table E3 Solving small scale instances with exact method

Instance	$nAct$	K	L	NC	SF	RSS	NCR	MCR	Optimal	Makespan	CPU time (s)
1	8	10	2	1.1	1	{0.8,0.6}	0.2	0.4	Yes	920	24
2	10	10	2	1.2	1	{0.8,0.6}	0.2	0.4	Yes	1046	41
3	12	15	2	1.4	1	{0.6,0.7}	0.2	0.4	Yes	1103	332
4	15	20	2	1.5	1	{0.5,0.6}	0.2	0.4	Yes	1341	2324
5	18	22	2	1.5	1	{0.4,0.5}	0.2	0.4	Yes	1894	6972
6	23	22	2	1.6	1	{0.4,0.6}	0.2	0.3	No	-	7200
7	25	20	3	1.8	1	{0.6,0.7,0.8}	0.2	0.3	No	-	7200
8	25	25	3	1.8	1	{0.6,0.7,0.8}	0.2	0.3	No	-	7200

The experiment result in Table E3 shows that despite the scale of instances is relatively small, only 5 of 8 instances obtain the optimal value within the optimizer time limit, but none of them meet the computation time constraints for reconnaissance mission planning. Due to complexity of reconnaissance mission planning problem, with the increase of instance size, the computation time increases sharply. Obviously, even in the simplest case, the exact algorithm cannot meet the computation time limit of the reconnaissance mission planning (less than 1s). The most complex instance in Table E3 is instance-8, but it is sill of smaller size than the simplest instance in Set-1, and Set-1 is the simplest with respect to complexity among 3 sets of instance. This experiment shows the optimal solution of reconnaissance scheduling problem is very hard to obtain. Hence we need to design a method to evaluate the distance between the solution obtained by the proposed algorithm and the optimal value. In this paper, we proposed to used the lower bound (LB) of the reconnaissance mission's makespan to show the maximum gap (denoted as $Mgap$) between makespan obtained by proposed algorithm and the optimal value. The lower bound makespan of a reconnaissance mission equals to the longest path weight from its start task to its end task, as shown (E1). If m is makespan obtained by proposed algorithm, the $Mgap$ between m and the corresponding optimal value can be evaluate using a LB as (E1)(E2) shows.

$$LB = \mu(0, n+1) \tag{E1}$$

$$Mgap = \frac{m - LB}{LB} \tag{E2}$$

Appendix E.3.2 *Comparison Algorithm Introduction*

In this section, we compare the proposed algorithm (MRFM-based heuristic algorithm) and other classic algorithms. Some parameters of experiment result that are reported in this paper: (1) Makespan gap: the gap between the makespan of reconnaissance mission and the lower bound. (2) CPU time: the processing time of the algorithm. (3) Workload: the sum of the working time of all reconnaissance agent. The mode information is not known in advance for reconnaissance mission planning, but the time to calculate the mode information has exceeded the time limit for reconnaissance mission planning. The MRFM-based heuristic algorithm proposed in this paper do not need to calculate the exact mode information while considering the multi-mode feature. For ME algorithm and the classic algorithm in Table E4, in practice, two boundary modes are usually specified for these single mode algorithms: (1) minimum requirement mode (Min), for a specific task, this mode requires the least resources, but corresponds to the longest processing time; (2) maximum requirement mode (Max), which requires the most resources, but corresponds to the smallest processing time for a specific task.

Table E4 Classic heuristics

Abbreviate	Rule	Tasks sorted by:
ME	mixed	Network flowed based optimization [2]
LFT	Latest Finish Time	Non-decreasing order of their latest finish times
LPT	Longest Processing	Time Non-increasing order of their processing times
LST	Latest Start Time	Non-decreasing order of their latest start times
SPT	Shortest Processing Time	Non-decreasing order of their processing times
MIS	Most Immediate Successors	Non-increasing order of their number of immediate successors
MTS	Most Total Successors	Non-increasing order of their total number of successors (immediate and transitive)
GRPW	Greatest Rank Positional Weight	Non-increasing order of the sum of their processing times with the processing times of their immediate successors
GRPW*	Greatest Rank Positional Weight*	Non-increasing order of the sum of their processing times with the processing times of all their successors

Appendix E.3.3 *Makespan Comparision*

We start by analyzing separately the behavior of different algorithms with respect to their makespan gap to the lower bound. The test result for each set of instances in shown in Table E5, the best results in each row are shown in bold. The average gap statistic result for Max-mode and Min-mode is shown in Figure E1. The statistics shows, on average, the MRFM-based heuristic algorithm has the best performance among the 10 algorithms, LFT has the second best performance, ME algorithm has only average level performance. By observing Figure E1, we can conclude that: expect for the MREM-based algorithm, there is no algorithm performing always better than any other. For algorithms in Table E4, the gap of Max mode is smaller than of Min mode. This is mainly because the processing time of a task can be shorten by assigning more resources to it, Max mode is with more resource allocation therefore it tends to get a shorter makespan. Although the max mode is of the shortest task processing time, the maximum resource requirement means that it is possible to delay the start of subsequent tasks. The classic single mode algorithms in Table E4 do not consider the above factors, the proposed algorithm comprehensively considers the relationship between the decrease in task execution time and the delay in the start time of subsequent tasks, so it can get better makespan results than others.

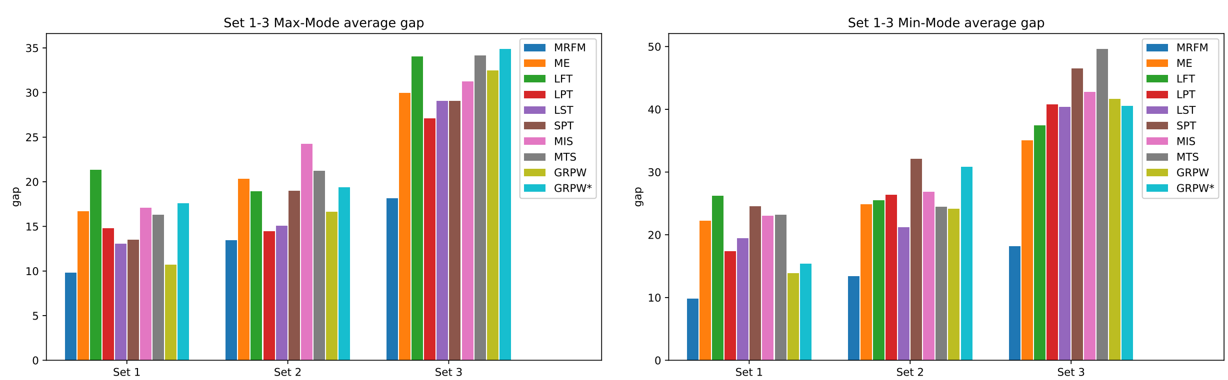

Figure E1 Makespan gap analysis (a) Max-mode (b) Min-mode

Appendix E.3.4 *CPU time Comparision*

Then, we analyze the CPU time of each algorithm. The CPU time for each set of instance in shown in Table E6, the best results in each row are shown in bold. The average CPU time statistic result for Max-mode and Min-mode is shown in Figure E2. Except for MRFM-based heuristic and ME algorithm, the CPU time of the other algorithms is basically at the same level, and the CPU time for a specific algorithm between two modes have no significant changes. Due to the importing of Minimum Resource Feasible Match problem, the MRFM-based algorithm takes more CPU time to obtain a feasible solution. Similarly, the modified part of network flow problem in ME algorithm takes some time to be processed,

Table E5 Makespan gap (%)

	Instance	MRFM	ME		LFT		LPT		LST		MTS		MIS		SPT		GPRW		GPRW*	
			Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
Set 1	1	**8.09**	16.05	23.61	23.24	21.14	12.10	13.75	9.95	21.53	9.37	26.41	17.48	26.82	17.59	21.61	9.16	17.78	21.14	14.50
	2	11.21	16.32	19.47	23.54	24.47	13.64	12.47	11.79	24.19	15.07	20.17	19.05	18.32	11.44	26.21	8.42	**7.57**	15.85	16.62
	3	11.80	18.47	21.99	15.98	22.17	18.98	13.74	19.75	17.85	18.00	28.21	13.79	21.22	16.87	22.02	**5.13**	13.54	25.65	10.38
	4	12.97	15.15	20.76	22.83	28.36	**8.66**	13.12	14.54	18.66	18.54	26.96	12.94	25.84	15.33	20.80	11.26	12.22	15.63	19.87
	5	**10.45**	16.84	22.63	21.29	29.64	20.09	25.07	16.00	16.56	11.49	28.43	19.19	26.06	17.24	22.55	15.28	15.27	19.83	14.06
	6	**7.59**	16.12	23.07	22.06	29.61	15.74	22.97	10.74	12.83	9.02	21.11	21.83	21.28	20.36	24.43	10.06	8.90	16.35	14.77
	7	**8.67**	16.32	22.18	20.99	28.66	12.27	12.72	8.83	27.15	12.36	24.33	20.29	22.87	17.12	21.06	10.87	23.23	12.80	13.16
	8	**7.99**	18.11	23.98	20.81	26.03	17.01	25.01	13.03	17.17	14.02	20.72	**12.20**	21.78	14.50	26.68	20.40	12.34	13.37	20.00
	Overall	**9.85**	16.67	22.21	21.34	26.26	14.81	17.36	13.08	19.49	13.48	24.54	17.10	23.02	16.31	23.17	11.33	13.86	17.58	15.42
Set 2	1	**14.68**	19.90	23.03	19.45	25.04	15.73	29.45	20.36	24.26	16.24	29.21	26.09	25.54	24.98	22.60	18.83	29.24	14.58	29.24
	2	12.00	19.78	25.91	19.12	22.48	11.97	29.14	11.57	26.56	20.92	30.38	21.15	25.51	23.24	21.15	**8.21**	20.66	16.19	32.91
	3	15.44	21.69	23.86	13.21	22.66	21.26	33.36	**10.85**	18.29	18.19	35.34	24.35	23.75	18.20	21.06	20.91	20.45	14.25	34.32
	4	**13.41**	21.78	26.09	22.20	25.06	17.53	26.53	19.07	17.63	20.08	31.62	26.73	23.88	20.02	26.03	19.12	31.80	19.56	23.53
	5	**11.51**	19.32	21.89	19.62	29.99	17.07	18.61	11.86	19.63	15.77	34.00	25.60	26.08	23.73	28.04	12.10	19.58	25.04	37.53
	6	15.82	21.00	26.45	19.74	23.29	**7.78**	23.69	19.41	16.90	22.16	31.21	27.65	31.52	20.76	26.05	19.43	26.57	25.98	31.59
	7	**10.77**	20.37	26.70	21.61	27.74	14.33	21.41	23.02	28.42	22.12	29.70	19.75	28.42	22.84	21.28	25.31	28.22	16.77	27.14
	8	**13.82**	18.84	25.06	16.35	28.05	15.01	28.93	12.69	17.72	16.32	35.55	22.85	30.30	15.88	29.30	12.16	16.78	22.78	30.49
	Overall	**13.43**	20.34	24.87	18.91	25.54	15.09	26.39	16.10	21.18	18.98	32.13	24.27	26.87	21.21	24.44	17.01	24.16	19.39	30.84
Set 3	1	**18.20**	28.25	35.85	33.73	42.21	26.89	47.75	35.18	39.77	26.42	46.95	28.05	45.51	35.25	50.53	24.95	54.49	40.42	39.49
	2	**18.95**	30.46	34.64	35.95	33.64	27.43	39.74	23.41	45.64	26.89	41.73	34.23	43.88	32.87	50.69	31.26	36.57	37.78	35.33
	3	**18.61**	31.39	35.25	37.54	42.69	34.57	36.86	36.78	40.63	28.36	47.71	29.35	37.73	35.30	54.40	28.32	37.34	29.21	39.27
	4	**20.21**	29.96	32.97	29.41	36.37	25.39	46.64	28.43	44.00	33.29	48.78	33.82	46.04	31.38	48.15	36.01	37.31	35.31	43.72
	5	**14.56**	28.42	37.05	33.08	40.66	24.40	36.33	25.50	44.81	31.61	49.04	31.55	45.43	31.06	48.31	36.91	50.16	37.52	40.12
	6	21.71	29.63	37.71	35.57	34.41	**20.79**	45.08	35.32	35.85	27.75	41.52	29.89	38.95	33.75	48.08	26.70	44.31	32.77	45.51
	7	**17.80**	31.23	33.27	32.21	35.53	32.19	35.69	25.08	34.71	27.58	47.22	32.43	46.87	40.23	48.84	34.24	36.16	28.24	38.35
	8	**15.29**	30.13	33.42	34.94	33.98	25.32	38.39	22.68	37.92	30.47	49.40	30.58	38.11	33.34	47.75	41.33	36.70	37.65	42.42
	Overall	**18.16**	29.93	35.02	34.05	37.43	27.12	40.81	29.05	40.42	29.05	46.54	31.24	42.82	34.15	49.59	32.47	41.63	34.86	40.53

so the CPU time of ME algorithm is longer than the classic algorithm in Table E4. The parameters of 3 instance sets are designed according to the statistics of reconnaissance mission, the size of Set 1 to Set 3 gradually increases. Set 1 has the smallest size of 3 instance sets, but it is larger then the reconnaissance mission of the smallest size in statistics. Set 3 has the largest size of 3 instance sets, it is larger then the reconnaissance mission of the largest size in statistics. Therefore, the 3 sets of instances can cover all situations of reconnaissance mission. The maximum computation time for reconnaissance mission planning problem is 1s. The average CPU time of MRFM-based algorithm in Set 1 to 3 is 0.065s, 0.384s and 0.354s respectively. Therefore, the MRFM-based algorithm can meet the computation time requirement of reconnaissance mission planning (less than 1s).

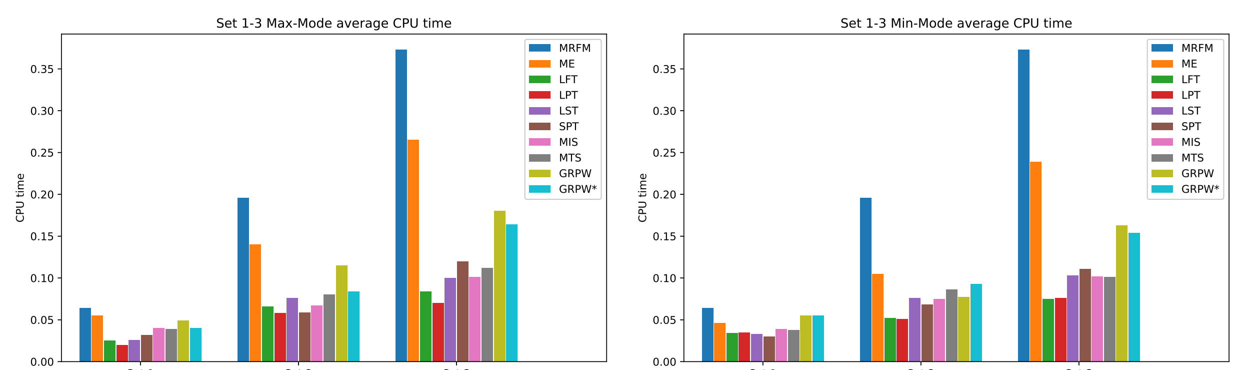

Figure E2 CPU time analysis (a) Max-mode (b) Min-mode

Appendix E.3.5 *Workload Comparision*

Finally, the workload of each algorithm is analyzed. The workload for each set of instances is shown in Table E7, the best results in each row are shown in bold. The average workload statistic result for Max-mode and Min-mode is shown in Figure E3. The MRFM-based algorithm has the minimal workload among all 10 algorithms in Max-mode.

Table E6 CPU time (s)

	Instance	MRFM	ME		LFT		LPT		LST		MTS		MIS		SPT		GPRW		GPRW*	
			Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
Set 1	1	0.052	0.059	0.041	0.030	0.033	**0.016**	0.038	0.029	0.032	0.032	0.021	0.041	0.037	0.037	0.044	0.050	0.056	0.040	0.048
	2	0.050	0.063	0.046	**0.016**	0.026	0.022	0.032	0.033	0.026	0.036	0.037	0.044	0.029	0.031	0.041	0.041	0.044	0.051	0.052
	3	0.085	0.060	0.030	0.022	0.039	**0.021**	0.038	0.026	0.034	0.027	0.033	0.050	0.045	0.031	0.044	0.053	0.050	0.044	0.048
	4	0.075	0.061	0.057	0.032	0.039	0.027	0.029	0.026	0.030	0.034	**0.025**	0.030	0.043	0.047	0.026	0.046	0.056	0.034	0.064
	5	0.067	0.059	0.046	0.029	0.024	**0.014**	0.035	0.018	0.035	0.036	0.040	0.046	0.040	0.044	0.038	0.050	0.044	0.042	0.063
	6	0.055	0.055	0.045	**0.017**	0.032	**0.017**	0.040	0.029	0.035	0.030	0.026	0.041	0.047	0.044	0.046	0.053	0.068	0.028	0.063
	7	0.050	0.044	0.056	0.028	0.037	**0.020**	0.037	0.031	0.033	0.027	0.027	0.040	0.040	0.032	0.032	0.050	0.064	0.029	0.054
	8	0.082	0.038	0.049	0.026	0.040	0.024	0.032	**0.017**	0.036	0.033	0.031	0.028	0.033	0.047	0.034	0.050	0.062	0.048	0.047
	Overall	0.064	0.055	0.046	0.025	0.034	**0.020**	0.035	0.026	0.033	0.032	0.030	0.040	0.039	0.039	0.038	0.049	0.055	0.040	0.055
Set 2	1	0.198	0.135	0.083	**0.046**	0.048	0.066	0.064	0.055	0.086	0.051	0.067	0.087	0.080	0.064	0.074	0.125	0.078	0.078	0.074
	2	0.229	0.113	0.095	0.076	0.042	0.046	**0.036**	0.093	0.061	0.060	0.080	0.050	0.054	0.061	0.066	0.105	0.064	0.077	0.106
	3	0.123	0.160	0.116	0.071	0.047	0.058	**0.041**	0.089	0.085	0.054	0.068	0.074	0.077	0.060	0.099	0.093	0.079	0.117	0.107
	4	0.229	0.149	0.102	0.073	0.057	0.080	**0.048**	0.090	0.062	0.056	0.055	0.075	0.070	0.090	0.108	0.129	0.064	0.085	0.112
	5	0.147	0.105	0.103	0.064	0.052	0.056	**0.045**	0.079	0.093	0.066	0.075	0.061	0.070	0.097	0.075	0.114	0.058	0.070	0.120
	6	0.222	0.146	0.086	0.070	0.063	0.064	0.061	**0.056**	**0.056**	0.067	0.080	0.061	0.099	0.099	0.079	0.111	0.107	0.072	0.069
	7	0.201	0.136	0.137	0.066	0.070	0.045	0.068	0.056	0.071	**0.042**	0.064	0.064	0.054	0.071	0.093	0.125	0.089	0.101	0.062
	8	0.222	0.175	0.122	0.060	0.051	**0.046**	0.056	0.089	0.091	0.077	0.057	0.061	0.095	0.100	0.093	0.120	0.080	0.071	0.096
	Overall	0.196	0.140	0.105	0.066	0.054	0.058	**0.052**	0.076	0.076	0.059	0.068	0.067	0.075	0.080	0.086	0.115	0.077	0.084	0.093
Set 3	1	0.369	0.229	0.295	0.067	**0.057**	0.072	0.083	0.089	0.080	0.136	0.140	0.103	0.082	0.095	0.095	0.177	0.168	0.173	0.133
	2	0.315	0.273	0.274	0.064	0.089	0.075	**0.067**	0.090	0.104	0.126	0.097	0.102	0.117	0.110	0.142	0.178	0.174	0.186	0.128
	3	0.369	0.281	0.190	0.094	**0.054**	0.073	0.055	0.130	0.083	0.083	0.086	0.078	0.106	0.121	0.082	0.160	0.156	0.117	0.135
	4	0.393	0.303	0.190	0.065	**0.070**	0.074	0.063	0.118	0.104	0.079	0.094	0.144	0.090	0.096	0.085	0.129	0.138	0.154	0.139
	5	0.496	0.281	0.239	0.106	0.104	**0.060**	0.101	0.092	0.084	0.147	0.127	0.084	0.131	0.146	0.122	0.212	0.167	0.204	0.129
	6	0.393	0.308	0.270	0.106	**0.059**	0.082	0.074	0.102	0.145	0.094	0.095	0.140	0.112	0.117	0.101	0.169	0.204	0.130	0.219
	7	0.347	0.186	0.181	0.060	0.084	**0.057**	0.061	0.076	0.122	0.143	0.147	0.082	0.087	0.119	0.104	0.219	0.124	0.165	0.202
	8	0.299	0.259	0.271	0.106	0.083	**0.068**	0.103	0.100	0.101	0.150	0.100	0.080	0.093	0.088	0.077	0.194	0.171	0.179	0.150
	Overall	0.373	0.265	0.239	0.084	0.075	**0.070**	0.076	0.100	0.103	0.120	0.111	0.101	0.102	0.112	0.101	0.180	0.163	0.164	0.154

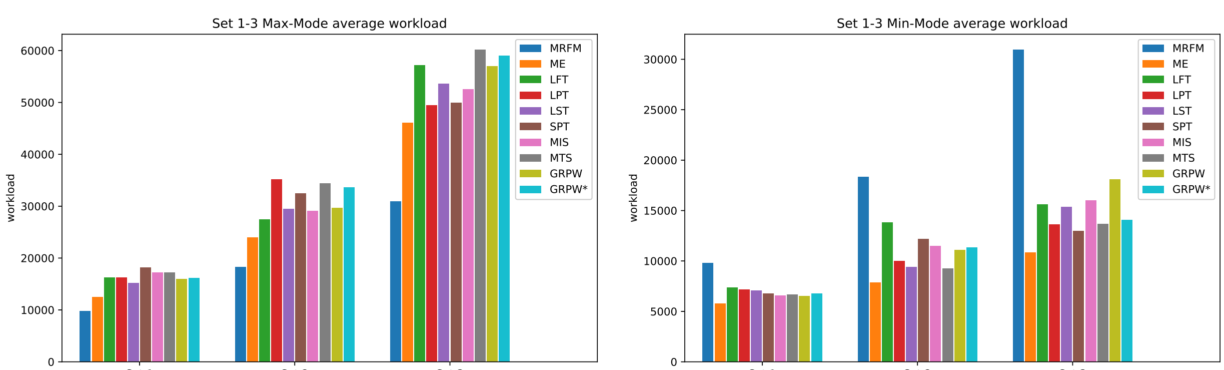

Figure E3 Workload analysis (a) Max-mode (b) Min-mode.

Table E7 Workload

	Instance	MRFM	ME		LFT		LPT		LST		MTS		MIS		SPT		GPRW		GPRW*	
			Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
Set 1	1	9792	11841	**6036**	15408	8454	16382	7554	15693	6838	19467	6100	18590	6985	17469	6360	16814	6105	15148	6763
	2	9609	13612	6327	15927	7942	17730	7946	15152	6182	17204	**6068**	17302	7473	17943	6185	16573	7217	16641	8105
	3	9990	13091	**5758**	17370	7282	15633	6324	15670	8131	17386	7505	17219	7142	16758	6845	14837	5762	15560	6729
	4	9503	12035	**5594**	15989	6660	15648	7693	15500	7771	17725	7166	16317	5862	17799	6294	16744	5586	16900	5999
	5	10369	13200	6115	15381	6124	16486	6673	15316	7384	19416	6998	17379	**5922**	16848	7021	14900	6205	17475	6136
	6	9420	12153	**4802**	15828	8339	16522	7832	15094	6791	18037	5767	17045	6670	16853	6578	15204	5674	15235	7066
	7	10194	12126	6022	16777	7101	15300	5674	14557	6715	16909	7169	17008	**5671**	16650	7742	15851	8107	16238	6351
	8	9502	11603	**5426**	17079	7266	16746	7817	14639	6643	19316	7412	16731	6793	17770	6414	16791	7774	16467	7041
	Overall	9797	12458	**5760**	16220	7396	16306	7189	15203	7057	18182	6773	17199	6565	17261	6680	15964	6554	16208	6774
Set 2	1	16771	23292	8028	28260	15785	35730	**7408**	29620	9116	34328	11737	30400	9835	32217	6966	31604	9820	33747	13470
	2	17322	23236	**7377**	26694	14827	32879	12109	30911	11181	33450	12560	28561	11154	37214	9120	29371	12117	33841	8924
	3	17386	23746	9784	26232	14622	32467	12272	28091	10712	33528	11216	27426	11551	36771	**7891**	28127	8829	34498	10095
	4	19397	25199	**6659**	29700	11820	36718	9094	27643	7673	31727	12950	28332	11974	32427	9579	28747	12888	34746	14164
	5	19583	24062	9634	27163	13439	35163	8412	31601	**6744**	30587	12614	27772	10769	32989	11472	31485	9060	32389	8845
	6	18277	23279	**6678**	25921	12854	35668	9273	28009	8271	34266	12708	30959	13149	33965	8411	29303	11278	33152	12444
	7	19042	23081	**7380**	28469	15540	35890	11675	28756	10644	30659	12886	29859	11229	36951	10993	29140	12888	33097	10789
	8	18887	26373	**7376**	27498	11876	36742	9740	31175	10850	31164	11008	29124	12125	32556	9591	30031	12080	33900	12218
	Overall	18333	24033	**7864**	27492	13845	35157	9998	29476	9399	32464	12210	29054	11473	34386	9253	29726	11120	33671	11369
Set 3	1	32644	43776	**9154**	54107	16590	47105	11037	55796	12997	47629	11436	54587	19265	60036	10153	52468	14077	57106	16895
	2	29330	44811	**9057**	59053	19855	47648	11150	55187	14223	48612	10469	50528	14138	63780	16259	57914	14780	60453	18397
	3	29916	47870	10281	56942	18794	46584	15381	57291	12272	48191	15662	49551	17156	57245	**9618**	56880	17703	60166	16653
	4	32867	47006	**12762**	60394	12832	54035	13445	51938	20735	52823	14987	50966	13700	63916	14811	55458	17671	59959	11309
	5	31176	48484	12839	53540	**11733**	52660	15519	50844	16574	49578	14626	52673	13011	58433	14519	60242	19640	60299	15245
	6	31390	45144	14169	54078	13216	50735	15030	50002	18347	50300	11427	50291	19209	54959	15086	58283	16051	58121	**11371**
	7	29714	45901	**9640**	57392	18600	49103	15178	57594	13975	53477	10867	57838	18606	61524	11462	60489	22650	57733	11543
	8	30477	45766	**9073**	61812	13390	48153	12262	50344	13902	49031	14517	53790	13149	61235	17473	53918	22190	58637	11013
	Overall	30939	46095	**10872**	57165	15626	49503	13625	53625	15378	49955	12999	52528	16029	60141	13673	56957	18095	59059	14053

References

1 Almeida B.F, Correia I, Saldanha-da-Gama F. Modeling frameworks for the multi-skill resource-constrained project scheduling problem: A theoretical and empirical comparison. International Transactions in Operational Research, 2019, 26(3), 946-967.

2 Almeida B.F, Correia I, Saldanha-da-Gama F. Priority-based heuristics for the multi-skill resource constrained project scheduling problem. Expert Systems with Applications, 2016, 57, 91-103.

3 Almeida B.F, Correia I, & Saldanha-da-Gama, F. An instance generator for the multi-skill resource-constrained project scheduling problem. University of Lisbon School of Science-Center for Mathematics, Basic Applications and Operations Research. 2015.