IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 July 2022, accepted 5 August 2022, date of publication 8 August 2022, date of current version 16 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3197637

RESEARCH ARTICLE

Evolutionary Multiobjective Satellite Range Scheduling With Learning-Guided Population Generation

MINGHUI XIONG, WEI XIONG, AND ZHENG LIU
Science and Technology on Complex Electronic System Simulation Laboratory, Space Engineering University, Beijing 101416, China
Corresponding author: Minghui Xiong (xtkxxmh@163.com)

This work was supported in part by the Science and Technology on Complex Electronic System Simulation Laboratory Funding under Grant XM2020XT1023.

ABSTRACT Most existing evolutionary approaches to satellite range scheduling seek optimal solution in terms of the request satisfaction. The scheduling demand of managing ground station resource is seldom considered, which restricts their real-world applications. To effectively generate a set of more rational satellite range schedules, this paper establishes the multi-objective satellite range scheduling mathematical model. Unlike existing approaches, we propose a general population generation approach to solve the problem without relying on any specific kind of evolutionary algorithm, so different types of evolutionary algorithms can be extended to satellite range scheduling without modifying the original framework and search strategy. The idea is to utilize the request satisfaction and resource utilization knowledge learnt from parent schedules to guide the generation and updating of new solutions. Furthermore, an iterative rewriting operator is designed to guide a biased faster convergence towards the low request failure region in objective space. The proposed approach has been applied to five different types of classical and state-of-the-art evolutionary algorithms and examined on benchmark problems. Experimental results illustrate the search efficiency enhancement and good adaptability to different evolutionary algorithms, which show the broad application prospect for satellite range scheduling.

INDEX TERMS Satellite range scheduling, multi-objective optimization, evolutionary algorithm, learning-guided.

I. INTRODUCTION

Satellite system plays an important role in military and civilian fields, such as military reconnaissance, marine surveillance, disaster forecasting, precise navigation, telecommunications, etc. To support the normal operation and application of onboard satellites, the ground stations perform the tasks of satellite tracking, telemetry, data transmission and command betting. All these satellite management activities rely on the communications between satellites and ground stations. The scheduling process of allocating specific ground station resource to establish satellite-ground communication is named satellite range scheduling problem (SRSP). The main goal of SRSP is to maximize the communication requests satisfaction under time and resource constraints. But the expanding satellite system types and scales intensifies the contradiction between the increasing satellite requests and limited ground station resource. Therefore, rationally allocating the ground station resource to satisfy more requests and improve the resource management efficiency is of great signficance.

In SRSP, the satellite measurement, control and data transmission resource should be appropriately assigned to the requests. The different types of ground station resources can only serve the request if the requested satellite is visible, and this time interval is named the visible time window (VTW). In this paper, we uniformly describe different types of resources as satellite-ground visible time window resources. The SRSP need to determine the time window and execution time to support the requests. During the scheduling process,

The associate editor coordinating the review of this manuscript and approving it for publication was Arturo Conde.

the situations where several satellites visible simultaneously and several ground station resource available contemporaneously could occur, making the VTW highly overlapped. The SPSP has been proven a NP-complete problem with combinational characteristic [1], [2].

Previous literatures mainly focus on the oversubscribed characteristic of SRSP, where not all the requests can be served due to the available time window conflicts. The earliest studies on satellite range scheduling are performed by researchers at the Air Force Institute of Technology to address the insufficient measurement and control resources in the 1990s [3], [4]. Over the years, a large number of algorithms have been proposed to explore the highly constrained search space and find the optimal solution. Based on the exactness, the algorithms can be classified into two groups: the deterministic algorithm and random search algorithm [5].

The deterministic algorithms explore the entire solution space by mathematical programming approaches, such as branch and cut [6], exact polynomial time algorithm [7], dynamic programming [8]. The deterministic algorithm can obtain the optimal or nearly-optimal solution with a polynomial time. But the search space of the problem grows exponentially as the problem size increases, leading to an inacceptable computational burden. To decrease the complexity of the problem and improve the search efficiency in large-scale request scenarios, several problem decomposition methods, for example the column generation [9], Lagrange relaxation [10] and cut plane [11] are adopted to find the tight bounds of the problem.

Different from the deterministic algorithms, the random search algorithms only search a limited part of the solution space, thus achieving more efficiency in finding a feasible solution. The random search approaches can be further divided into local search and evolutionary search. The local search algorithm searches from an initial solution and is capable to construct a high-quality solution in a short time. In local search algorithms, the feature knowledge of the feasible solution and problem is utilized to guide the initial schedule construction and search process. Based on the conflict information between requests, Luo *et al.* [12] first construct an elite initial solution by prescheduling and then iteratively improve the solution in a subspace of feasible solutions. The greedy algorithm adopts a simple heuristic which always replaces the current solution if a better solution is found. The results in [2] illustrate that the simple greedy heuristic failed to show competitive performance on large scale instances. Although the greedy heuristic suffers from premature convergence in finding optimal solution, it is still a simple but efficient method to construct elite initial schedule [13]. To avoid being trapped in the local optima, Han [14] and Sarkheyli [15] utilize simulated annealing and tabu search to improve the possibilities to accept worse solution. Based on heuristic rules, Chen *et al.* [16] restart the search process from the current solution.

Due to the excellent global search capability and broad applicability, the evolutionary computation has been extensively used in scheduling domains [17], [18], [19]. The evolutionary algorithms that are commonly adopted to solve SRSP include genetic algorithm (GA) [20], ant colony optimization (ACO) [21], [22], [23]. Some researchers also hybrid the evolutionary algorithms with different search operators to improve optimization efficiency, such as GA-ACO [24], GA-PSO [25] and ACO-Tabu [26].

Most of the previous literatures make efforts to schedule as much requests as possible, which can be classified as single-objective optimization approach. In recent years, multi-objective satellite scheduling problem is derived in order to consider multiple scheduling criterions simultaneously. In earth observation satellite scheduling problem, Yang *et al.* [27] optimize the number of observed targets and overall image quality. Xiong *et al.* [28] proposed a cooperative coevolutionary multi-objective algorithm to produce high-quality solutions in Chinese Navigation Satellite System Project. In [29], [30], and [31], the researchers formulate the SPSP as a many-objective problem which contains windows fitness, clashes fitness, time requirement fitness, and resource usage fitness. Petelin *et al.* [32] utilized six different evolutionary algorithms to obtain the trade-off solutions of the problem. The comparison with weighted-sum approach indicates that the evolutionary algorithms always manage to obtain as good solution as the weighted-sum, while the other non-dominated solutions give the decision makers more additional alternatives. Song *et al.* [33] optimized the overall profit and task failure rate with an improved NSGA-II. Most of these optimization objectives are still designed from a satellite user view, i.e., to serve more requests.

However, from the perspective of resource management department, constructing a schedule with better ground station resource management is another important task during the scheduling process. For example, the better antenna load balance degree represents a relatively balanced antenna working time, which can prolong the overall service life of the equipment and improve the response capability to emergency tasks. Therefore, it is necessary to solve SRSP considering the scheduling needs of both satellite users and resource managers, which is one of the motivations of our proposed approach.

Over the past two decades, a great deal of researches on evolutionary algorithms have been proposed to solve multi-objective optimization problems, proven effective on benchmark instances and real-world applications [34], [35], [36]. Compared to traditional mathematical programming approaches, the multi-objective evolutionary algorithms (MOEAs) can obtain a set of non-dominated solutions in a single run. However, randomly generated population in original MOEAs suffers from poor efficiency due to the combinational and highly-constrained characteristics of SRSP.

Apart from the population-based evolution algorithms, the memetic algorithm (MA), which combines the evolutionary operators with local search, has shown promising performance for solving scheduling problems. For flow-shop scheduling problem, Wang *et al.* [37] explore the solution space with two collaborative populations, and several problem-specific operators are incorporated to intensify the local search capability. For job-shop scheduling problem, Gong *et al.* [38] designed a similarity-based crossover operator to reduce the ineffective crossover and search space in late optimization phase. Du *et al.* [39] proposed a general memetic algorithm frame work for satellite range scheduling problem, in which two heuristic operators are designed and tabu search is utilized as local operators. Memetic algorithms enhance the search capability through the combination of global and local search, where the local search operators start the optimization from the place that the evolutionary operators have found promising in previous iterations.

Motivated by the scheduling need of both satellite user and ground station resource manager, we focus on the multi-objective satellite range scheduling problem (MOSRSP). Inspired by the successful application of MOEAs and the hybrid idea of memetic algorithms, in this paper, a learning-guided population generation approach is proposed to enhance the performance of MOEAs and solve the problem more efficiently. The main contributions of this paper can be enumerated as follows:

- A multi-objective SRSP mathematical model that allows the decision makers to find a set of potential satellite range schedules. These Pareto-optimal schedules should satisfy the requests of satellite users with more rational ground station resource utilization.
- A learning-guided population generation approach, which learn the request satisfaction and antenna load balance knowledge from parent schedules, is proposed to enhance the performance of MOEAs on solving the MOSRSP without modifying the original algorithm frameworks and search strategies.
- An iterative rewriting operator that guides the population further exploit the low request failure region in objective space and accelerate the convergence speed.
- Experiments that investigate feasibility and efficiency the proposed approach as well as the adaptability to different types of evolutionary algorithms.

The remainder of this paper is organized as follows. In Section II, we describe the MOSRSP and formulate the mathematical model. The proposed approach is described in detail in Section III, Section IV and Section V shows the experimental design and result comparison. The conclusions of the study would be discussed in the last part.

II. PROBLEM FORMULATION

A. PRELIMINARIES

The ground stations monitor the satellite current status, upload command and receive data to support the satellite daily and emergency requests. The multi-satellite range scheduling problem is to obtain a rational dispatch of the ground station resource to satisfy the satellite requests within the time range.

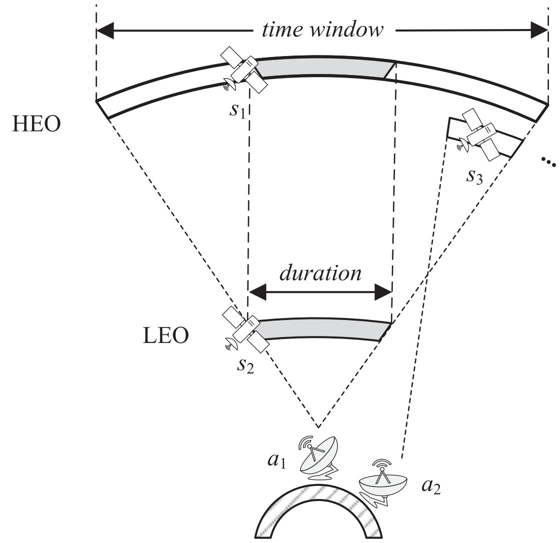


FIGURE 1. Satellite range scheduling problem.

In essence, it can be described as a visible time window allocation problem under the time and resource constraints. During scheduling process, the decision makers are concerned with the request satisfaction and resource utilization to determine a final schedule. Considering the limits of equipment capacity and satellite-ground station visibility, these problems of optimizing a schedule with more than two objectives can be depicted by a multi-objective optimization problem under equality and inequality constraints.

As shown in Fig. 1, the problem involves multiple satellites and multiple ground stations. When the satellites pass over the ground station, the visible time window resource could be used to establish the satellite-ground communication. For low-earth-orbit (LEO) satellites, the requested communication time may last for the entire time window due to the short time window duration. If the entire time window is utilized for communication, scheduling LEO satellite requests is similar to the job scheduling problem [40]. For high-earth-orbit (HEO) satellites, the available time window in longer than the requested duration, which means both the time window and the exact execution time should be determined. The crucial point to obtain an ideal schedule is to handle the *conflict* between satellite requests (multi-satellite simultaneously pass, e.g., s_1 and s_2 in Fig. 1) and *competition* between ground station antennas (multi-antenna simultaneously available, e.g., a_1 and a_2 in Fig. 1) during the scheduling process.

With reference to previous researches and engineering experiences, we make the following assumptions to convert the SRSP problem into a scientific problem [41]:

- *Precedence:* there is no predefined execution sequence relationship between requests.
- *Preemption:* preemption is not allowed. The ground station antenna cannot be released until the current communication finished. Once an execution starts, it should be finished without interruption.

- *Duration:* the requested duration time of each request is fixed.
- *Uncertainty:* the uncertain factors are not considered. All the resources and requests are known before scheduling. The equipment is reliable, and no operation accident would occur during the scheduling.
- *Availability:* each request has at least one available time window to be served.
- *Energy:* the ground stations have abundant power, storage and other resources.

B. NOTATIONS

1) INPUT

In general, the MOSRSP can be described by:

$$MOSRSP = \{R, A, S, TW\} \tag{1}$$

where,

- $R = \{1, 2, \ldots, |R|\}$ denotes the set of $|R|$ requests. Each request $r = \{s_{ID}, erst, duet, dur, p\}$ is specified by its requested satellite identification, earliest possible start time $erst$, due time $duet$, requested duration dur, and the priority weight p measuring its importance.
- $A = \{1, 2, \ldots, |A|\}$ denotes $|A|$ ground station antennas. Each antenna $a = \{a_{ID}, swi\}$ is determined by the antenna identification and the switch time swi to serve next reques.
- $S = \{1, 2, \ldots, |S|\}$ is the set of $|A|$ satellites.
- $TW = \{1, 2, \ldots, |TW|\}$ is the set of $|TW|$ time windows. More specifically, for each request r, the ground station antennas have K time windows $tw_r = \{tw^k_{r,a} \mid k = 1, 2, \ldots K\}$. $\forall tw^k_{r,a} \in tw_r$, $tw^k_{r,a} = \{r, a, k, st, et\}$. It indicates $tw^k_{r,a}$ is the k-th time window that antenna a can support request r from start time st to end time et. Noted that a in $tw^k_{r,a}$ epresents the specific antenna that provides the k-th time window, rather than the k-th visible time window on a.

2) OUTPUT

The output corresponding to request r in final schedule should include:

- $x^k_{r,a}$: a binary variable determines whether to execute request r on k-th feasible time window $tw^k_{r,a}$. $x^k_{r,a} = 1$ if selected, or $x^k_{r,a} = 0$ otherwise. $R = \{1, 2, \ldots, |R|\}$ denotes the set of $|R|$ requests. Each request $r = \{s_{ID}, erst, duet, dur, p\}$ is specified by its requested satellite identification, earliest possible start time $erst$, due time $duet$, requested duration dur, and the priority weight p measuring its importance.
- est_r: the start time to execute request r. Once the exact start time is determined, the end time ent_r can be obtained by $ent_r = est_r + dur_r$.

C. MATHEMATICAL MODEL

During the scheduling process, the satellite users focus on the request satisfaction, while the resource manage departments pay more attention to resource utilization. From the scheduling demand of both satellite user and resource manager, we formulate the satellite range scheduling problem as a multi-objective minimization problem under resource and time constraints:

$$min \quad f_1 = 1 - \left(\sum_{r=1}^{|R|} x^k_{r,a} \cdot p_r \right) \Bigg/ \sum_{r=1}^{|R|} p_r \tag{2}$$

$$f_2 = \left\{ \frac{\sum_{a=1}^{|A|} (L(a) - \overline{L(A)})^2}{N_a - 1} \right\}^{1/2} \Bigg/ \overline{L(A)} \tag{3}$$

$$s.t. \quad \sum_{k=1}^{K} x^k_{r,a} \le 1, \quad r \in R \tag{4}$$

$$\sum_{t=1}^{|T|} \sum_{s=1}^{|S|} x^t_{s,a} \le 1, \quad t \in T, \ s \in S, \ a \in A \tag{5}$$

$$\sum_{t=1}^{|T|} \sum_{a=1}^{|A|} x^t_{s,a} \le 1, \quad t \in T, \ s \in S, \ a \in A \tag{6}$$

$$st_i \le est_i < ent_i \le et_i, \quad i \in R \tag{7}$$

$$[st_r, et_r + swi_a] \cap [st_{r'}, et_{r'} + swi_a] = \emptyset, \quad r \ne r', \ a \in A \tag{8}$$

where (2) and (3) represent the optimization objectives:

- *Weighted request failure rate:* Weighted request failure rate of a schedule presents the punishment from unscheduled requests, which intuitively reflects the dissatisfaction degree from satellite users;
- *Antenna load imbalance degree:* Equation (3) evaluates the load imbalance degree between each ground station antenna, which reflects the demand for ground station resource management. $L(a)$ denotes the total working time of antenna a, and $\overline{L(A)}$ is the mean working load of ground station antennas.

And a feasible schedule should satisfy constraint (4)-(8):

- *Execution uniqueness:* Constraint (4) ensures each request is to be served at most once;
- *Satellite uniqueness:* Constraint (5) describes one satellite can interact with at most one antenna simultaneously, where $x^t_{s,a} = 1$ denotes satellite s communicates with antenna a at time t, or $x^t_{s,a} = 0$ otherwise;
- *Antenna uniqueness:* Similar to constraint (5), constraint (6) ensures one-to-one interactions between the satellites and antennas, the main difference lies in the aggregation method of $x^t_{s,a}$;
- *Execution feasibility:* Constraint (7) indicates the execution time should be located within the time window.
- *Switch time:* For the requests served on the same antenna, constraint (8) restricts the execution time interval to be longer than the switch time.

III. PROPOSED METHOD

The goal of the above-formulated two-objective satellite range scheduling problem is to find the optimal schedule

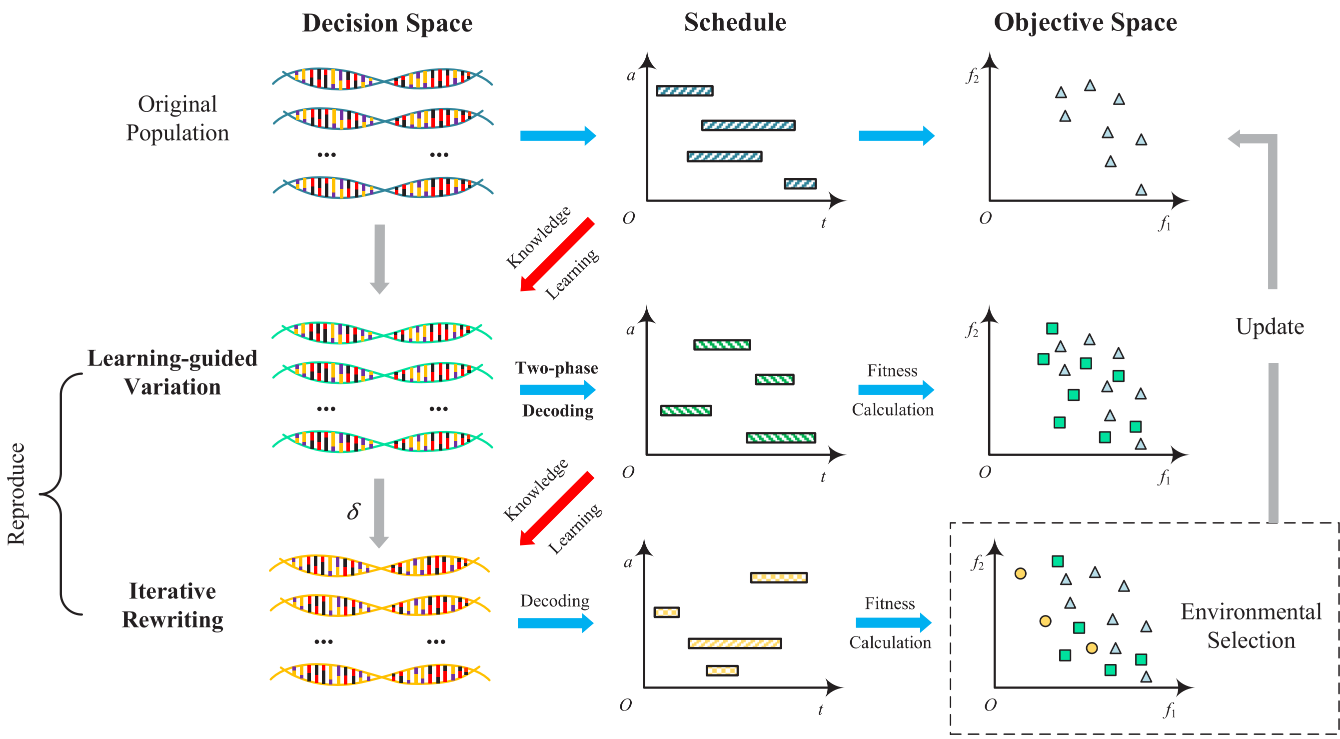


FIGURE 2. Relationships among the learning-guided variation, iterative rewriting and two-phase decoding.

which meets the demand of satellite user and resource management department. To optimize the request failure rate and load imbalance degree resulted from request conflict and antenna competition, multi-objective evolutionary algorithms are commonly used approaches. However, conventional randomly-generated population suffers from low efficiency due to the combinational and highly-constrained characteristic of MOSRSPs. To improve the overall performance of MOEAs on handling MOSRSPs, we propose a learning-guided population generation approach, which generate the final offspring solution by two-phase decoding and rewriting (2D&R). In this section, an overview of the framework is first given, and then, the key components of the proposed method are illustrated in the following parts.

Algorithm 1 General Framework

Require: Request size $|R|$, Population size N, Rewriting probability δ

Ensure: Final population P_{tmax}

1: $t \Leftarrow 1$
2: $P_0 \Leftarrow$ Initialize a random population
3: **while** termination criterion not met **do**
4: $\quad S_t \Leftarrow$ **Learning-guidedVariation**(P_t)
5: $\quad Q_t \Leftarrow$ **Two-phaseDecoding**(P_t, S_t)
6: $\quad Q_t^{'} \Leftarrow$ **IterativeRewriting**(Q_t, δ)
7: $\quad P_t \Leftarrow P_t \cup Q_t^{'}$
8: $\quad P_t + 1 \Leftarrow$ Environmentalselection(P_t, N)
9: $\quad t \Leftarrow t + 1$
10: **end while**
11: **Return** P_{tmax}

A. FRAMEWORK

Algorithm 1 presents the proposed evolutionary algorithm framework based on learning-guided population generation. Firstly, an initial population P_0 consisting of N solutions is randomly generated, the length of each solution is $|R|$ (line 2). In each generation, the request satisfaction and resource utilization knowledge learnt from the parent schedule is introduced to guide the evolutionary variation operation (line 4). Due to the existence of time and resource constraints, the initial offspring solution could be an infeasible solution. A two-phase decoding strategy is adopted in Line 5 to repair the infeasible solution and improve population diversity. Then, an iterative rewriting strategy is utilized to rewrite the solution gene components while ensuring optimizing the request failure rate, where δ is the probability of performing rewriting (line 6). A larger δ indicates a more biased search behavior towards the low request failure rate objective space. Once the parent population is updated (line 7), the environmental selection would determine the parent population for next generation based on the fitness value (line 8). Fig. 2 shows the relationships between the learning-guided variation, two-phase decoding, iterative rewriting and environmental selection for better understanding.

B. SOLUTION ENCODING

A schedule corresponding to the input request set should decide two variables: (1) $x_{r,a}^k$, whether to serve request r on its k-th feasible time window $tw_{r,a}^k$ and (2) st_r, the exact

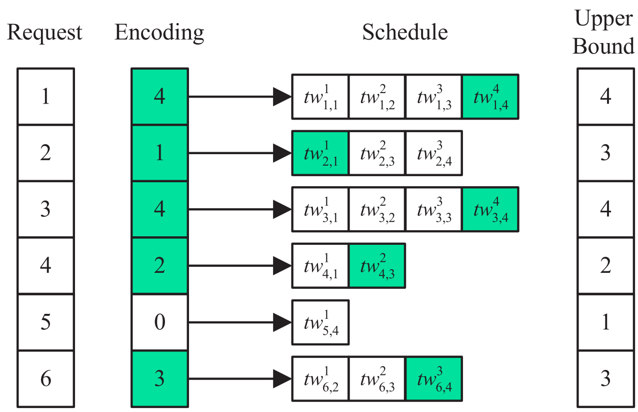


FIGURE 3. Solution encoding.

start time to execute request r. As discussed in Section II, a longer the time window indicates more execution opportunities and potential conflicts, leading to a rapid growth in the search space and complexity of the problem. To reduce the search space, we build a dual decision-making model of satellite range scheduling, where the time window allocation is encoded in each solution and the execution start time is decided in decoding phase.

A solution x is encoded by the following integer array:

$$x = \{x_1, x_2, \ldots, x_{|R|}\} \tag{9}$$

$$x_r = \begin{cases} k & \arg(x^k_{r,a} = 1 \ \wedge \ \sum\limits_{k \in K} x^k_{r,a} = 1) \\ 0 & \sum\limits_{k \in K} x^k_{r,a} = 0 \end{cases} \tag{10}$$

Each gene site corresponds to a request, and each request is sorted by earliest possible start time. The length of the solution is the size of the request set. The 0 value of x_r indicates request r would be cancelled in the schedule (request 5 in Fig. 3). Otherwise, x_r indicates the selected time window.

Fig. 3 gives an example of a solution encoding. There are six requests to be scheduled in total. Request 1-4 and 6 is assigned the 4th, 1st, 4th, 2nd and 3rd time window, request 5 would not be served. Noted that the a of $tw^k_{r,a}$ is utilized to specify the exact antenna providing the time window resource, which is useful in conflict detecting procedure. The upper bound for each gene is the number of feasible time windows for request r. Equation (10) indicates each request corresponds no more than one time window, hence the execution and satellite uniqueness constraint (constraint (4)-(5)) would be naturally satisfied.

C. LEARNING-GUIDED VARIATION

1) MUTATION OPERATOR

The mutation updates the value of multiple gene site within the lower and upper bound. The mutation probability for each gene site is pm/D, where pm and D is the input mutation probability and number of decision variables, respectively. A random number set of size $[1, D]$ is compared with mutation probability to determine whether to perform mutation. The polynomial mutation [42] is employed to obtain the updated value of selected mutation site. The mutation

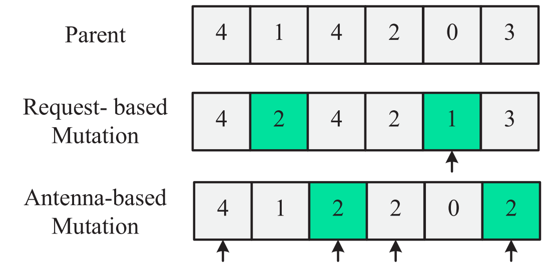


FIGURE 4. The mutation operation.

operator includes two different methods, which utilize different types of knowledge learnt from parent schedule:

1) Request-based mutation

The mutation probability for the unscheduled request ur in parent schedule is pm;

2) Antenna-based mutation

We give priority to mutate the requests which are served on the antenna with high load imbalance degree. First, the load imbalance degree of antenna a is calculated by:

$$lid(a) = \frac{|L(a) - \overline{L(A)}|}{\sum\limits_{a \in A} |L(a) - \overline{L(A)}|} \tag{11}$$

where $L(a)$ is the load of antenna a and $L(A)$ denotes the mean working load of antenna set A. And $lid(a)$ implies the load imbalance degree of each antenna in parent schedule.

Next, the two antennas with the largest lid value, namely, the antenna with the highest and lowest working load are selected. Then, the gene sites of the requests which can be supported by the selected antennas are attached a higher mutation probability of pm.

Fig. 4 give a demonstration of the mutating the solution in Fig. 3 under different rules. Assume all the requests are successfully executed on the assigned time windows and the requested durations are of the same length. The gene site marked by the arrow represent the request given higher mutation probability, and the green square shows the final mutated gene. In request-based mutation, the unscheduled request 5 has more tendency to be served in next iteration, while other requests (e.g., request 2) are still likely to be mutated under influence of randomness factor.

Suppose all the requested durations are 10 minutes. The working load for each antenna is [10, 0, 10, 30]. Then, the load imbalance degree for the five antennas is [0.07, 0.36, 0.07, 0.5]. Therefore, the requests which can be served on antenna 2 and 4 would be mutated with a higher possibility.

The mutation operation would be conducted before crossover, and the mutated solution are also preserved in offspring population, which can be regarded as a local search from parent solution.

2) CROSSOVER OPERATOR

The crossover operation is performed to exchange the selected genes between the two parent solutions. For each gene site in generation t, the crossover probability

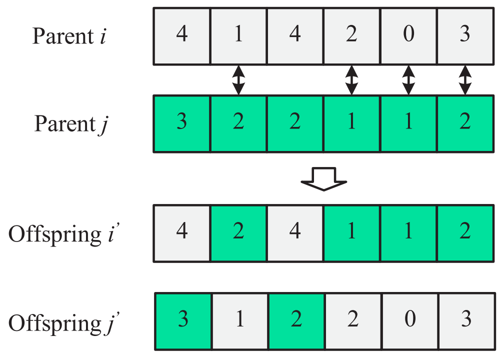


FIGURE 5. **The crossover operation.**

can be obtained by:

$$pc = pc_l + (pc_u - pc_l) \cdot \frac{Gen - t}{Gen} \tag{12}$$

where pc_l and pc_u are predefined lower and upper bound of crossover probability; Gen denotes the maximum number of generations. Similar to the mutation operator, the corresponding genes of unscheduled requests have a higher crossover probability of $2pc$. As shown in Fig. 5, the multiple genes of the parent solutions would be exchanged in the randomly selected sites. The crossover operation is an effective mechanism to prevent the population from being trapped in local optimum, thus to improve the global search capability.

The learning-guided variation operators incorporate the objective-specified knowledge, i.e., the request satisfaction and antenna working load information of parent solution for improving the search efficiency, while the randomness inherited from the original evolutionary variation operators is another factor to improve the population capability of jumping out of the local optimum. The contributions of each operator during the optimization process would be further discussed in Section V.

D. TWO-PHASE DECODING STRATEGY

The encoding of each individual in the population represents the allocation of time window resource. Having generated the new time window schedule in variation operation, the exact start time to serve the request would be determined in decoding phase.

The encoded genes of each solution can be regarded as an initial time window allocation schedule. Due to the antenna uniqueness and switch time constraint, only one request might be successfully scheduled between conflicting requests. For preemption is not allowed, sequential decoding the solution based on the request ID would always cancel the requests that arrives later, indicating the loss of potential high-quality solutions. Observing the limitations, we propose a two-phase decoding strategy to repair the infeasible solution and improve the population diversity. The variated gene sites in variation procedure, denoted by vx, are adopted to decide the decoding sequence. The main steps are as follows:

(1) Split solution x into x_1 and x_2 based on the variated flag vx:

$$x^1 = \bigcup_{r \in R} {x_r}^1 \tag{13}$$

$$x^2 = \bigcup_{r \in R} {x_r}^2 \tag{14}$$

where x_1 and x_2 denotes the decision variables that would be decoded in the first and second phase, respectively. And the decision variable for each gene site in the sub-solution is decided by:

$${x_r}^1 = \begin{cases} x_r & vx_r = 1 \\ 0 & vx_r = 0 \end{cases} \tag{15}$$

$${x_r}^2 = \begin{cases} x_r & vx_r = 0 \\ 0 & vx_r = 1 \end{cases} \tag{16}$$

where $vx_r = 1$ represents the variation occurred in the gene site corresponding to request r, or $vx_r = 0$ otherwise.

(2) Initialize the solution final schedule $p = \{sr, fr, usr\}$ and occupied antenna resource $OAR = \{oar_a \mid a \in A\}$, where sr, fr and usr denotes the set of scheduled, failed and unscheduled requests; oar_a is the occupied resource on antenna a.

(3) Next, decide the exact start time to serve the requests scheduled in x^1 based on the task-time window selection algorithm (TTSA) with top position arrangement strategy in [33].Within each time window, the earliest executable time window position is selected as the service start time.

(4) Find the requests that are successfully served in original schedule and remain unvaried in offspring solution, detect the conflict between these requests and current OAR. If not conflicting, add these requests to sr, update OAR and set the corresponding gene of x^2 to 0.

(5) Based on the current schedule, and antenna resource occupation status, we decode x^2 to obtain the final schedule.

Fig. 6 gives a schematic diagram of the two-phase decoding. Suppose the requests scheduled on the same antenna are conflicting. Decoding the same individual, the proposed decoding approach would construct different final schedule under different variated flag, thus improving the population diversity. In addition, the requests, which are unscheduled or assigned to the high load imbalance degree antenna in parent schedule, are attached higher possibility to be variated in variation procedure. Hence, prioritizing the variated requests could lead to a better population performance in objective space.

E. ITERATIVE REWRITING STRATEGY

In multi-objective optimization, each objective is mathematically equivalent. As shown in Fig. 7, the solutions are roughly divided into three groups based on the performance in objective space. Both the solutions with low request failure rate and low load imbalance degree are necessary to describe the entire Pareto front (PF). Whereas in context of satellite range scheduling, a solution located in the low request failure region

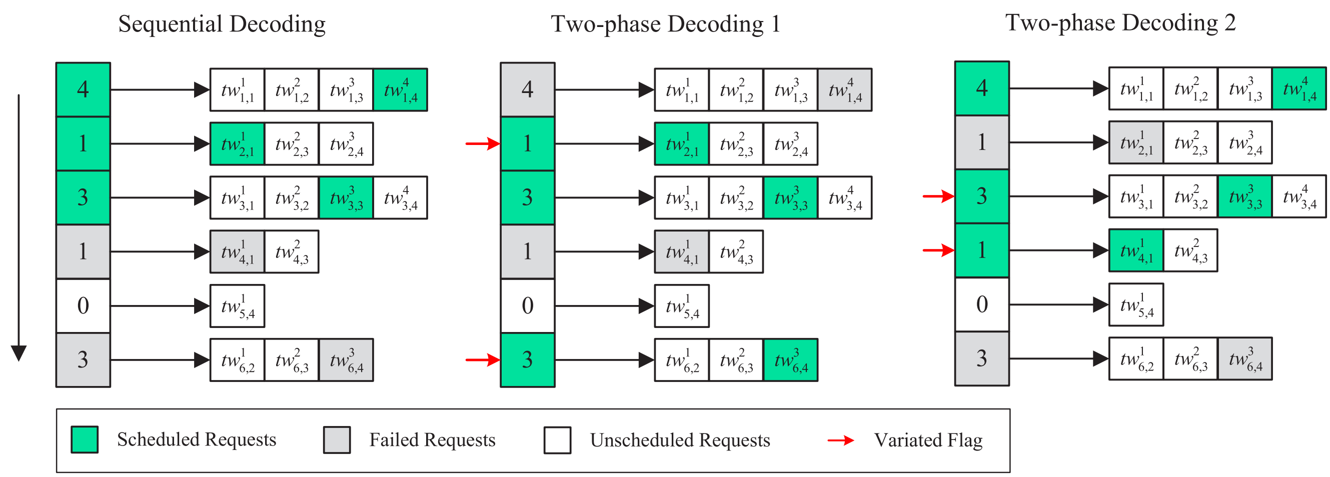


FIGURE 6. The comparison between sequential decoding and two-phase decoding under different variated flag.

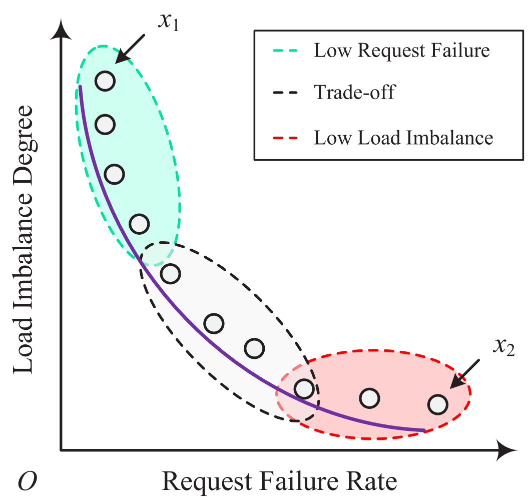


FIGURE 7. Distribution of solutions to approximate the PF of MOSRSP.

(e.g., x_1) is preferred than that in low load imbalance region (e.g., x_2).

Hence, we perform iterative rewriting on the selected offspring solutions for a faster convergence to the low request failure region. The rewriting strategy iteratively modifies genes corresponding to the unscheduled requests in initial offspring schedule to generate a neighboring solution. Based on the knowledge learnt from the initial offspring solution, the rewriting path is defined by:

(1) For each unscheduled request, update the current feasible time window tw'_r according to constraint (7). Obtain the current load $L(A)$ of the antenna set.

(2) Calculate the rewriting priority. Considering the antenna load, request weight and flexibility, we define the rewriting priority indicator as:

$$pl_r = \frac{w_r}{\min(L'_{r,a}) \cdot fl_r} \tag{17}$$

where $L'_{r,a}$ is the current load of the remaining antennas that are still capable to serve request r; w_r is the input weight of request r; fl_r denotes the schedule flexibility of r:

$$fl_r = \frac{\sum_{k=1}^{K'} |tw^k_{r,a}|}{dur_r} \tag{18}$$

where $|tw^k_{r,a}|$ and dur_r denotes the length of each available time window and requested duration, respectively.

In our priority indicator, both the load, weight and flexibility are considered. A higher fl_r means more freedom in choosing execution time window and start time. The initial schedule of the request with lower available antenna load, higher weight and less flexibility would be rewrite first. Noted that all the different scaled factors are normalized into [0,1] before calculation.

(3) Using roulette to select a request r according to the rewriting priorities.

(4) Select the time window resource and execution start time. If the request is successfully scheduled, add r to scheduled request set sr, delete r in unscheduled request set ur, rewrite the gene value corresponding to request r with the selected time window number k and update the occupied antenna resource set OAR. If failed, delete r in ur and rewrite the corresponding gene to be 0.

(5) Repeat (1)-(4) until ur is empty or no feasible time window left.

IV. EXPERIMENTAL DESIGN

This section is devoted to the experimental design for investigating the performance of the proposed heuristic population generation approach. First, the test problem, performance indicators and algorithms employed to incorporate 2D&R are briefly introduced; then, the investigation into parameter settings is conducted; finally, the main tasks of the experimental design is described.

A. TEST PROBLEM AND PERFORMANCE INDICATORS

The Air Force Institute of Technology (AFIT) benchmark suit (http://www.cs.colostate.edu/sched/data.html) consisting of seven problem instances is utilized for comparative study. Each AFIT problem represents actual 24 hours request data and visible time windows from AFSCN. The ground station antennas are globally distributed and the requests involves both high-orbit requests and low orbit requests. The total number of the requests to be scheduled in the seven problems are 322, 302, 300, 316, 305, 298, and 297 respectively. To test the ability of each algorithm in scheduling more requests, we set all the request weight to be 1.

The performance indicators are needed to quantitively analysis the performance of different algorithms. In this paper, the widely used IGD [43], DM [44] and HV [45] indicators are adopted. The metric IGD and DM measures the convergence and diversity of the final non-dominated solution set respectively, while HV evaluate the both in a sense. The knowledge of the true PF is not required for the calculation of HV, which is hard to obtain in practice. Hence, HV is adopt as the primary comparison criterion. The reference point is set, as recommended in [46], to be [1.1, 1.1]. A larger value of HV indicates better performance. In addition, the final non-dominated solution set that all the algorithms obtained is utilized as an approximation to the true PF to evaluate IGD and DM.

B. ALGORITHM SELECTION

To verify the proposed 2D&R, the following five different types of algorithms are selected to incorporate 2D&R: 1) MaOEADPP [47], a novel approach using a repulsive point process to identify high-quality solutions based on the decomposition of the kernel matrix; 2) MOEA/D-DU [48], a decomposition-based algorithm that update the K nearest parent solutions based on the perpendicular distance from a solution to the weight vector for a better trade-off between convergence and diversity; 3) NSGA-II [49], a classical Pareto dominance-based approach with elite mechanism and maintain diversity through crowding distance sorting; 4) NSGA-III [50], a reference point-based algorithm; 5) 1by1EA [51], a niche-based MOEA which selects the offspring individuals one-by-one based on the convergence indicator.

C. EXPERIMENTAL SETTINGS

The general settings and parameter settings are listed as follows:

- *Termination criterion:* The termination criterion is set in the form of maximum number of evaluations, and the specific number is assigned to 30000 after the pre-experiment.
- *Number of runs:* Each algorithm is run 10 times independently for each test.
- *Rewriting probability δ:* The threshold δ to determine whether to perform rewriting is set to 0.3, and the influence of different δ on the algorithm performance would be further investigated in Section V.
- *Platform:* All the experiments are carried out in PlatEMO [52] using Matlab R2019b.

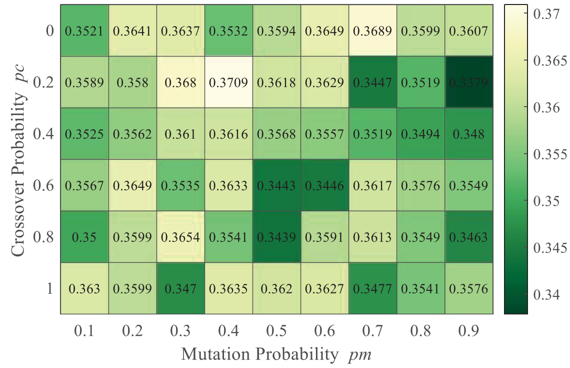


FIGURE 8. Mean performance of HV values under different crossover and mutation probability.

To determine the appropriate parameters for crossover and mutation, we conduct a preliminary experiment on AFIT3 using MaOEADPP-2D&R.

The upper bound of mutation probability is set to 0.9. The mean algorithm performance in 5 independent runs in shown in Fig. 8. The result indicates that the influence of different variation probability on the algorithm performance is not fixed. According to the HV value, we set the mutation and crossover probability to 0.2 and 0.4 in the following experiments.

D. RESEARCH TASKS AND QUESTIONS

To validate the 2D&R performance, the main tasks of the experimental design is to answer the following four questions:

- *Feasibility:* Is the 2D&R-incorporated algorithms capable to obtain a final non-dominated solution set with good convergence and diversity?
- *Effectiveness:* Can the 2D&R improve the convergence speed and degree of the existing evolutionary algorithms, especially to the low request failure region?
- *Sensitivity:* How does the rewriting probability δ influence the algorithm performance?
- *Contribution:* Which heuristic operator contribute the most to the optimization process?

V. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the research questions raised in the previous section is to be validated. Correspondingly, our experiment can be divided into four parts. We compare different types of MOEA-2D&Rs with the original algorithms on benchmark instances, analysis the convergence speed with respect to the performance indicators, and discuss the influence of different rewriting probability. Furthermore, the contributions of each operator are evaluated based on the total amount of generated novel non-dominated solutions.

A. FEASIBILITY VERIFICATION

1) OVERVIEW

An overall analysis of MaOEADPP, MOEA/D-DU, NSGA-II, NSGA-III, 1by1EA, and the corresponding 2D&R version,

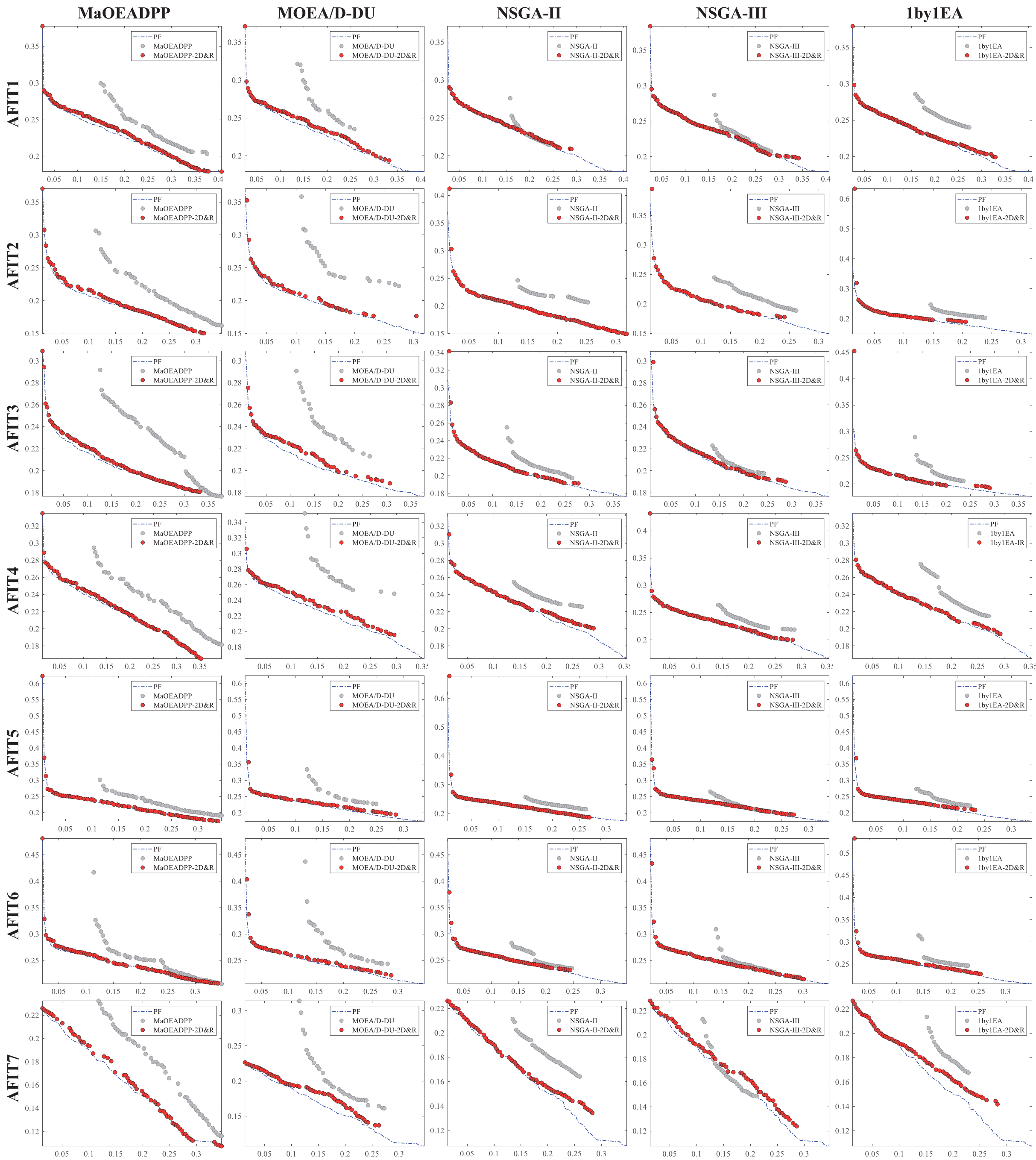


FIGURE 9. Non-dominated solutions found by different algorithms in 10 runs on AFIT1-AFIT7.

on solving multi-objective satellite range scheduling problem is described in this section. Fig. 9 shows the distribution of final non-dominated solutions that different algorithms obtained in 10 independent runs. Each row presents different algorithms to solve the same problem, and each column plots the final solution set that each algorithms obtained. In each subplot, the x-axis and y-axis denote the request failure rate (f_1) and antenna load imbalance degree (f_2), respectively. The blue dotted line is all the final non-dominated solutions that the ten algorithms obtained on solving the same problem. It is adopted as an approximation to the true PF.

From the result comparisons, we can obtain the following observations for the proposed 2D&R:

1) It is evident from Fig. 9 that the MOEA-2D&Rs are capable to obtain a final solution set with better distribution. The incorporation of 2D&R improves the population distribution on all instances, which indicates that 2D&R could generate solutions with more diversified performance for the algorithm to select. For decision-makers, this means more

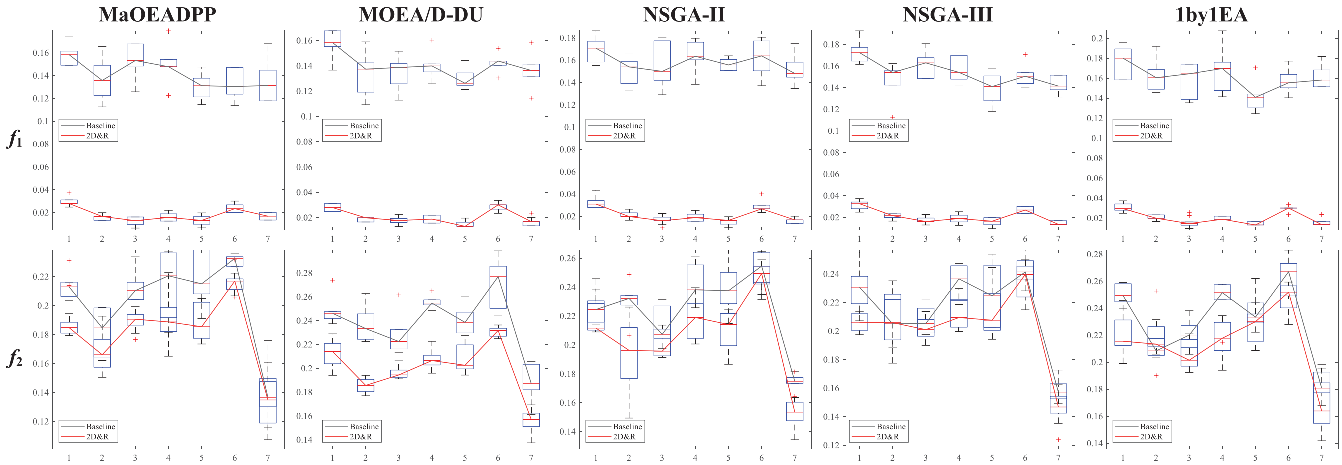


FIGURE 10. The comparisons of the best values for different algorithms on AFIT1-AFIT7.

diversified schedules with different request failure rate and load imbalance degree to choose from.

2) In terms of convergence, the 2D&R-incorporated algorithm could obtain the non-dominated solutions which are closer to the PF on most instances. The results imply that the introduction of heuristic information could effectively enhance the efficiency of generating the solutions with better convergence. In some instances, the original algorithms find the solutions with lower load in some areas (for example NSGA-III on AFIT7), which shows that the original algorithms could find the non-dominated solution using random operators, but with lower efficiency.

3) Among the algorithms, MaOEADPP-2D&R outperforms other algorithms on the tested instances. In the final non-dominated solution set on each instance, most of the solutions with the lowest load are obtained by MaOEADPP-2D&R, which shows that the algorithm has good ability of exploring the objective space. Moreover, though in some instances, the final non-dominated solutions of other algorithms cannot approximate the PF as well as the MaOEADPP-2D&R does, all these solutions are located at the left part of the PF. In satellite range scheduling, the above biased population distribution means assigning more computational resource to explore the low request failure rate region in objective space.

4) In some instances, the original algorithms show similar performance to the 2D&R-incorported algorithms in terms of the load imbalance degree (e.g., MaOEADPP, NSGA-II, NSGA-III on AFIT6). However, these solutions are located on the high request failure rate part of the PF. In the low request failure region, the 2D&Rs always perform better than the original algorithms.

To further analyze search ability of 2D&R of generating extreme solutions, the comparisons of the best values for different algorithms on AFIT1-AFIT7 are shown in Fig. 10.

A smaller value of each objective function indicates better performance. All the MOEA-2D&Rs manage to find the solution whose request failure rate is less than 4% in each run on the seven instances, while the best load imbalance degree ranges from 0.11 to 0.23. In term request failure rate, the introduction of 2D&R can significantly improve the efficiency of generation solutions with lower request failure. For antenna load balancing performance, the 2D&R can still find the solution with lower load imbalance degree. Whether the decision maker requires a schedule with lower request failure rate or lower antenna load imbalance degree, 2D&R could generate the solution which satisfy the preference.

2) NUMERICAL ANALYSIS

In this section, the IGD [43], DM [44] and HV [45]are utilized to quantitively analysis both the algorithm performance and 2D&R performance. Five different types of algorithms are selected for comparison, including the classical reference point-based, decomposition-based, niche-based, dominance-based algorithms and a novel proposed determinantal point process-based algorithm. The mean results of the performance indicators for each algorithm in ten runs are listed in Table 1. The '/' denotes the original algorithms and the '2D&R' denotes the improved algorithms. The best performance is highlighted in bold and the better performance between the MOEA and MOEA-2D&R is marked in underline.

As the results in Table 1 illustrate, the 2D&R shows a promising result in most benchmark instances. Moreover, the 2D&R results show good adaptation capability, where the performance of all incorporated algorithms have been significantly improved. In terms of mean IGD value, the convergence degree of all algorithms to the PF is effectively improved.

The 2D&R performs much more stable than random population generation since the latter obtains quite many solutions distant to the PF. With regard to DM performance, it can be observed that the MOEA-2D&R is clearly better than the original algorithms in exploring the objective space. The overall performance comparison results prove the scheduling capability enhancement for each algorithm in terms of the HV value.

TABLE 1. Comparison of mean IGD, DM and HV values for different algorithms on AFIT1-AFIT7 problems.Best and better performance are shown in bold and underline respectively.

Problem	Metric	MaOEADPP		MOEA/D-DU		NSGA-II		NSGA-III		1by1EA	
		2D&R	/	2D&R	/	2D&R	/	2D&R	/	2D&R	/
AFIT1	IGD	<u>**1.09E-02**</u>	5.87E-02	<u>1.89E-02</u>	7.45E-02	<u>2.83E-02</u>	6.10E-02	<u>1.55E-02</u>	6.32E-02	<u>3.07E-02</u>	7.61E-02
	DM	<u>**0.825**</u>	0.571	<u>0.722</u>	0.456	<u>0.662</u>	0.347	<u>0.749</u>	0.362	<u>0.597</u>	0.327
	HV	<u>**0.414**</u>	0.264	<u>0.392</u>	0.236	<u>0.395</u>	0.269	<u>0.406</u>	0.262	<u>0.393</u>	0.228
AFIT2	IGD	<u>**1.60E-02**</u>	7.07E-02	<u>2.34E-02</u>	9.00E-02	<u>4.33E-02</u>	8.06E-02	<u>5.67E-02</u>	6.49E-02	7.87E-02	<u>7.64E-02</u>
	DM	<u>**0.751**</u>	0.53	<u>0.678</u>	0.503	<u>0.558</u>	0.378	<u>0.491</u>	0.403	<u>0.368</u>	0.342
	HV	<u>**0.477**</u>	0.267	<u>0.459</u>	0.231	<u>0.459</u>	0.241	<u>0.439</u>	0.279	<u>0.429</u>	0.246
AFIT3	IGD	<u>**1.97E-02**</u>	6.85E-02	<u>2.05E-02</u>	7.50E-02	<u>4.28E-02</u>	6.56E-02	<u>4.01E-02</u>	6.68E-02	<u>5.19E-02</u>	7.33E-02
	DM	<u>**0.716**</u>	0.524	<u>0.696</u>	0.469	<u>0.565</u>	0.358	<u>0.556</u>	0.316	<u>0.496</u>	0.329
	HV	<u>**0.376**</u>	0.199	<u>0.361</u>	0.201	<u>0.364</u>	0.227	<u>0.363</u>	0.235	<u>0.355</u>	0.209
AFIT4	IGD	<u>**1.54E-02**</u>	7.17E-02	<u>2.02E-02</u>	8.12E-02	<u>3.71E-02</u>	7.59E-02	<u>2.68E-02</u>	6.99E-02	<u>3.49E-02</u>	8.17E-02
	DM	<u>**0.749**</u>	0.541	<u>0.722</u>	0.506	<u>0.606</u>	0.334	<u>0.651</u>	0.347	<u>0.58</u>	0.349
	HV	<u>**0.368**</u>	0.192	<u>0.346</u>	0.176	<u>0.344</u>	0.193	<u>0.356</u>	0.207	<u>0.346</u>	0.181
AFIT5	IGD	<u>**1.37E-02**</u>	6.73E-02	<u>2.34E-02</u>	7.24E-02	<u>2.63E-02</u>	7.48E-02	<u>3.39E-02</u>	6.17E-02	<u>4.95E-02</u>	6.97E-02
	DM	<u>**0.778**</u>	0.585	<u>0.726</u>	0.477	<u>0.666</u>	0.396	<u>0.64</u>	0.371	<u>0.496</u>	0.358
	HV	<u>**0.656**</u>	0.412	<u>0.628</u>	0.41	<u>0.641</u>	0.37	<u>0.644</u>	0.414	<u>0.638</u>	0.399
AFIT6	IGD	<u>**1.01E-02**</u>	5.20E-02	<u>1.82E-02</u>	7.73E-02	<u>5.33E-02</u>	6.59E-02	<u>2.99E-02</u>	5.52E-02	<u>5.44E-02</u>	7.19E-02
	DM	<u>**0.781**</u>	0.606	<u>0.716</u>	0.528	<u>0.491</u>	0.354	<u>0.632</u>	0.402	<u>0.477</u>	0.363
	HV	<u>**0.502**</u>	0.331	<u>0.486</u>	0.288	<u>0.478</u>	0.297	<u>0.491</u>	0.319	<u>0.475</u>	0.291
AFIT7	IGD	<u>**1.65E-02**</u>	6.38E-02	<u>2.40E-02</u>	7.81E-02	<u>2.63E-02</u>	6.60E-02	<u>1.92E-02</u>	5.33E-02	<u>4.15E-02</u>	7.48E-02
	DM	<u>**0.801**</u>	0.533	<u>0.733</u>	0.536	<u>0.688</u>	0.443	<u>0.771</u>	0.448	<u>0.603</u>	0.374
	HV	<u>**0.301**</u>	0.166	<u>0.262</u>	0.118	<u>0.282</u>	0.161	<u>0.287</u>	0.205	<u>0.251</u>	0.14

Based on the above results, we introduce a performance indicator to give an intuitive comparison on the algorithm performance [53]. Suppose there are N_A algorithms $Alg_1, Alg_2, \ldots, Alg_N$, where $Alg_i, Alg_j \in Alg \wedge i \neq j$. The performance indicator $\partial_{i,j}$ is determined as:

$$\partial_{i,j} = \begin{cases} 1 & Alg_j \text{ significantly outperforms } Alg_i \\ 0 & otherwise \end{cases} \tag{19}$$

Then, the performance score of algorithm Alg_i can be defined by:

$$p(Alg_i) = \sum_{j=1}^{N_A} \partial_{i,j} \tag{20}$$

The $p(Alg_i)$ means the number of the algorithms which are significantly better than Alg_i. A lower value of this indicator indicates better performance. The Wilcoxon signed-rank test [54] at a 5% signficance level is conducted to measure the statistical signficance. Fig. 11 gives a summary of the average performance score comparison in terms of IGD, DM and HV values.

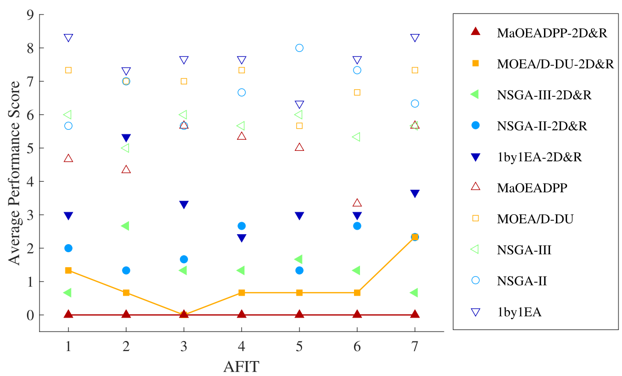


FIGURE 11. Mean performance score comparison over AFIT1-AFIT7.

From Fig. 11, MaOEADPP shows best overall performance on all the considered instances. Moreover, it also ranks the best among the original algorithms. The outstanding results indicate the repulsive point process adapted in MaOEADPP works well on diversity maintenance and objective space exploration. MOEA/D-DU is the second-best algorithm, and the performance rank is improved from the fourth to the second. Guided by the knowledge of parent solution, the 2D&R can generate diversified offspring solution for MOEA/D-DU to update the population, thus effectively improves the efficiency of the decomposition-based algorithm.

The overall performance of NSGA-II and NSGA-III is close. On AFIT1 and AFIT7, NSGA-III manages to obtain better results. 1by1EA could not obtain satisfying results in several instances. The overall performance of 1by1EA-2D&R on solving AFIT2 is not as good as that of original MaOEADPP and NSGA-III, where 1by1EA could only cover a limited proportion of the PF for 4 times out of 10 runs. In 1by1EA, part of the solutions is pre-selected based on an updating penalty threshold. Pre-selecting these

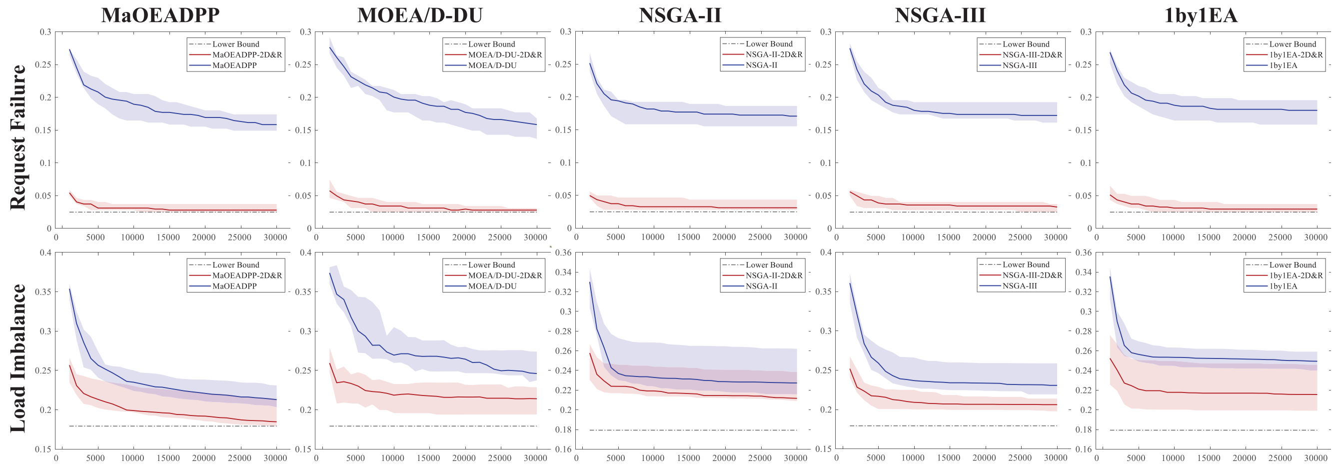


FIGURE 12. The convergence performance comparison on AFIT1. The x-axis denotes the number of the evaluations, y-axis represents the request failure (f_1) and load imbalance (f_2) respectively.

solutions could accelerate approaching to the PF, while the possibility of diversity lost would also increase.

B. EFFICIENCY ANALYSIS

In this section, we compare the 2D&R with original algorithms to analysis the optimization efficiency improvement. The best objective values that each algorithm obtained during the iteration is adopted to represent the convergence efficiency. The convergence performance comparison on AFIT1 is shown in Fig. 12. Each column represents different algorithm, and each row denotes the optimization objective. The gray dotted line represents the best value obtained by each algorithm in 10 runs.

In terms of the request failure, it can be observed that the convergence speed of each algorithm has been significantly improved, which proves the efficiency of the proposed 2D&R to generate the solution with low request failure rate. For antenna load imbalance degree, the convergence speed of 2D&R is similar to that of the original algorithm, but the best fitness value of 2D&R in each run is lower. In summary, the 2D&R can accelerate the population converging to the low request failure rate region, and search for the schedule with low load imbalance degree within this region.

Moreover, the mean computational time consumed by ten algorithms on AFIT1-AFIT7 in 10 runs is summarized in Fig. 13. It can be observed that the MOEA-2D&Rs use less computational time than the original MOEAs. The reason is that in the implement of 2D&R, we construct the offspring schedule on the basis of parent schedule in decoding procedure, rather than decode the decision variables one-by-one as the original algorithms does. More details of the offspring decoding are described in Section III-D.

C. INFLUENCE OF PARAMETER δ

In this section, the influence of parameter δ on the algorithm performance would be investigated. The rewriting strategy is designed to guide a biased population generation towards the low request failure rate region. And the introduction of

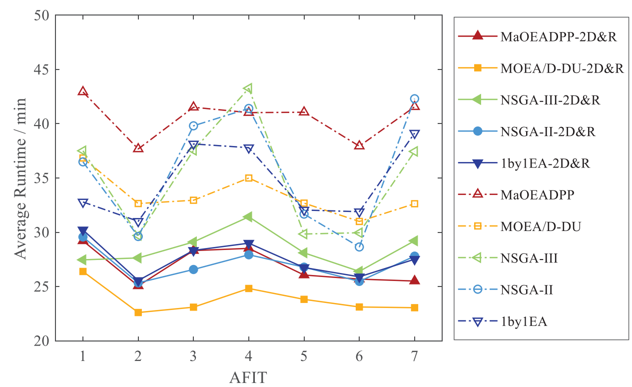


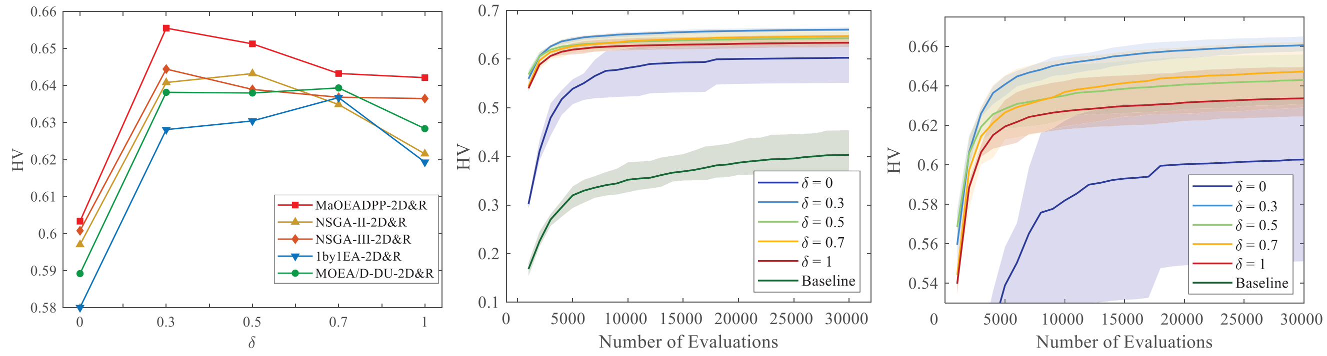
FIGURE 13. Mean computational time (in minutes) consumed by different algorithms on AFIT1-AFIT7.

parameter δ aims to maintain a balance between biased search and global search. As discussed in Section IV-A, the HV indicator is utilized to measure the overall performance.

Fig. 14 shows how the algorithms performance varies with the change of δ. We vary δ between [0,0.3,0.5,0.7,1], where $\delta = 0$ and $\delta = 1$ means the entire population are generated with two-phase decoding and rewriting respectively. It can be observed that:

1) Compared with $\delta = 0$, the introduction of rewriting would improve the algorithm performance.
2) $\delta = 1$ would be a better choice than 0, but the overemphasized rewriting may lead to premature convergence to the local optima.
3) The best δ for different algorithms varies. $\delta = 0.3$ is best MaOEADPP and NSGA-III, and 1by1EA get better performance with $\delta = 0$. NSGA-II and MOEA/D-DU are more robust over different δ values.

Then, the convergence performance with varied δ values is further discussed. Fig. 14(b) shows the comparison result of MaOEADPP-2D&R. The original MaOEADPP is adopted as the baseline. It can be observed that, the heuristic information introduced in the two-phase decoding strategy ($\delta = 0$) would enhance the convergence performance, while the rewriting



(a) Mean HV comparison (b) Convergence comparison (c) Detailed comparison of (b)

FIGURE 14. **Representative examination of the influence of δ on HV for AFIT5. The figures show the average HV of 10 independent runs each.**

TABLE 2. **Operator contributions to generating novel non-dominated solutions.**

Operators		MaOEADPP		MOEA/D-DU		NSGA-II		NSGA-III		1by1EA	
2D	Mutation-r	38.2	12.5	34.5	8.7	38.6	15.6	38.1	15.2	40.0	15.7
	Mutation-a		12.2		6.3		15.7		15.2		16.0
	Crossover		13.5		19.5		7.3		7.7		8.3
R	Mutation-r	**61.8**	18.5	**65.5**	17.6	**61.4**	21.8	**61.9**	<u>23.1</u>	**60.0**	22.0
	Mutation-a		17.6		12		<u>22.8</u>		21.9		<u>23.0</u>
	Crossover		<u>25.7</u>		<u>35.9</u>		16.8		16.9		15.0

strategy would further accelerate the convergence ($\delta > 0$). More detailed comparison with different δ values is depicted in Fig. 14(c). In this paper, we set the δ value to be 0.3.

D. OPERATOR CONTRIBUTION ANALYSIS

In this section, the contribution that each operator devotes to the population evolution are analyzed, which is indicated by the number of novel non-dominated solutions found by each operator. The *novel* means the non-dominated offspring solution is different to the current parent population either in objective space or decision space.

Table 2 presents the comparison results, where all the values are normalized into [0,1] and expressed by percentage. The strategy and operator that contributes the most is highlighted in bold and underline respectively. As can be seen, the rewriting strategy (R) contributes over 50% in all five types of the algorithms, revealing that rewriting can update the population more efficiently. The request-based mutation operator (Mutation-r) and antenna-based mutation operator (Mutation-a) make similar contributions. The overall contributions of the mutation operators are greater than the crossover operator. This result indicates that the heuristic information introduced in the mutation process could effectively enhance the quality of the generated offspring solution. Based on the above analysis, it can be concluded that each operator designed in 2D&R managed make contribution to the population evolution.

E. DISCUSSION

In summary, the experimental results show a promising performance both in accelerating the convergence and maintaining diversity by using the 2D&R. The improvement should be attributed to the following characteristics of the 2D&R:

1) Combination: the 2D&R combines the idea of global optimization and local search into the population generation, which is similar to the idea of memetic algorithm. The original evolutionary algorithms explore the objective space based on the randomly generated solutions, which would lead to poor efficiency in MOSRSPs. We combine the solutions generated both by the mutation and crossover operation. The variatied gene sites in mutation operation are fewer than that in crossover, which can be considered as a local search from current solution. Furthermore, part of the population would be rewrite for a biased further exploitation in the low request failure objective space. The main difference between 2D&R and memetic algorithm is the local search strategy. In memetic algorithms, the local search is conducted after the evolutionary operations with a fixed neighborhood size. Then replace the current solution with neighboring solution if the replace criterion is satisfied. In 2D&R, the parent solution is mutated with a certain probability, and all the new neighbor solutions would participate in environmental selection.
2) Iterative: iteration is a key mechanism in evolutionary operations, where offspring solutions are generated from the areas found promising by the parent solutions. Based on the 2D&R, the population generation would be guided by the heuristic information learnt from parent solution (variation and two-phase decoding) and original offspring solution (rewriting). The request satisfaction and load imbalance are different for different solutions, hence the learnt knowledge is diversified. The 2D&R is not concerned to find an optimal solution in a single search as the local search does, but to generate possible better solutions during each iteration, and the optimization occurs as the search and selection progresses.
3) Probabilistic: basic iterative improvement from current solution could lead to local optima, not necessarily converging to the global Pareto front. To avoid being trapped in a low-quality local optimum, several advanced local search algorithms, such as the simulated annealing and tabu search, assign probability to

choose a neighbor solution even if its performance is worse than the current solution. In 2D&R, we introduce the heuristic information in a probabilistic manner. Moreover, the randomness of evolutionary operations is another factor for the population to explore new areas. The randomly generated variation sites for other genes can be regarded as the points to restart the search from the current solution.

F. LIMITATIONS OF OUR APPROACH

There are two main limitations to our approach:

(1) The request failure of the solution found by our algorithm is lower than the known best value found by single-objective optimization [12]. The reason is that we set the start time of each request as early as possible to simplify the problem, which may sometimes prune too much search space;

(2) Some real-world constraints are not taken into account. For example, the request types are not considered, which may influence the request priority and restrict the request execution order. These limitations would be solved in our future work.

VI. CONCLUSION

In this paper, we address the problem of multi-objective satellite range scheduling, which optimizes the request failure and antenna load imbalance. The goal is to find a set of satellite range schedules which can satisfy the input requests with more rational resource utilization. Instead of relying on a specific evolutionary algorithm, a general learning-guided population generation approach, named 2D&R, is proposed to overcome the low convergence rate of evolutionary algorithms for solving MOSRSPs. In variation phase, the request satisfaction and resource utilization knowledge learnt from the parent solutions is used to attach different variation probabilities to each gene site. The variated gene sites can then be used as the flags to start the two-phase decoding, intending to improve the population diversity. Finally, the rewriting process is conducted with probability to guide the population further exploit within the low-request failure region in objective space.

We have applied our approach on five different types of the MOEAs, including the DPP-based, Pareto dominance-based, reference point-based, niche-based and decomposition-based MOEAs. Experimental results show the good adaptability to different evolutionary algorithms. The comparison results on real-word problem instances reveal that the proposed 2D&R could enhance the performance of original MOEAs in terms of exploration capability and convergence efficiency.

In future studies, we will further analysis the population generation behavior of 2D&R, for example, the performance on instances of different scales. Furthermore, more constraints reflecting the real-world situations would be taken into account, such as request types and uncertainty. This paper focuses on analyzing and comparing the performance enhancement to different evolutionary algorithms. The comparison with existing multi-objective satellite range scheduling approaches would be conducted in future work.

REFERENCES

[1] A. J. Vazquez and R. S. Erwin, "On the tractability of satellite range scheduling," *Optim. Lett.*, vol. 9, no. 2, pp. 311–327, Feb. 2015.
[2] L. Barbulescu, J.-P. Watson, L. D. Whitley, and A. E. Howe, "Scheduling space–ground communications for the air force satellite control network," *J. Scheduling*, vol. 7, no. 1, pp. 7–34, 2004.
[3] T. D. Gooley, "Automating the satellite range scheduling process," M.S. thesis, Wright-Patterson Air Force Base, Wright-Patterson, OH, USA, Mar. 1993. [Online]. Available: https://ui.adsabs.harvard.edu/abs/1993MsT..........7G/exportcitation
[4] D. A. Parish, "A genetic algorithm approach to automating satellite range scheduling," M.S. thesis, Wright-Patterson Air Force Base, Wright-Patterson, OH, USA, Mar. 1994. [Online]. Available: https://ui.adsabs.harvard.edu/abs/1994MsT.........13P/exportcitation
[5] F. Xhafa and A. W. H. Ip, "Optimisation problems and resolution methods in satellite scheduling and space-craft operation: A survey," *Enterprise Inf. Syst.*, vol. 15, no. 8, pp. 1022–1045, Sep. 2021.
[6] T. D. Gooley, J. J. Borsi, and J. T. Moore, "Automating air force satellite control network (AFSCN) scheduling," *Math. Comput. Model.*, vol. 24, no. 2, pp. 91–101, Jul. 1996.
[7] A. J. Vazquez and R. S. Erwin, "Optimal fixed interval satellite range scheduling," in *Proc. ICORES*, 2014, pp. 401–408.
[8] Z. Liu, Z. Feng, and Z. Ren, "Route-reduction-based dynamic programming for large-scale satellite range scheduling problem," *Eng. Optim.*, vol. 51, no. 11, pp. 1944–1964, Nov. 2019.
[9] C. A. Rigo, L. O. Seman, E. Camponogara, E. M. Filho, E. A. Bezerra, and P. Munari, "A branch-and-price algorithm for nanosatellite task scheduling to improve mission quality-of-service," *Eur. J. Oper. Res.*, vol. 303, no. 1, pp. 168–183, Nov. 2022.
[10] N. Brown, B. Arguello, L. Nozick, and N. Xu, "A heuristic approach to satellite range scheduling with bounds using Lagrangian relaxation," *IEEE Syst. J.*, vol. 12, no. 4, pp. 3828–3836, Dec. 2018.
[11] S. Spangelo, J. Cutler, K. Gilson, and A. Cohn, "Optimization-based scheduling for the single-satellite, multi-ground station communication problem," *Comput. Oper. Res.*, vol. 57, pp. 1–16, May 2015.
[12] K. Luo, H. Wang, Y. Li, and Q. Li, "High-performance technique for satellite range scheduling," *Comput. Oper. Res.*, vol. 85, pp. 12–21, Sep. 2017.
[13] X. Chen, G. Reinelt, G. Dai, and M. Wang, "Priority-based and conflict-avoidance heuristics for multi-satellite scheduling," *Appl. Soft Comput.*, vol. 69, pp. 177–191, Aug. 2018.
[14] C. Han, Y. Gu, G. Wu, and X. Wang, "Simulated annealing based heuristic for multiple agile satellites scheduling under cloud coverage uncertainty," 2020, *arXiv:2003.08363*.
[15] A. Sarkheyli, A. Bagheri, B. Ghorbani-Vaghei, and R. Askari-Moghadam, "Using an effective Tabu search in interactive resources scheduling problem for LEO satellites missions," *Aerosp. Sci. Technol.*, vol. 29, no. 1, pp. 287–295, Aug. 2013.
[16] Y. Chen, Y. Song, Y. Du, M. Wang, R. Zong, and C. Gong, "A knowledge-based scheduling method for multi-satellite range system," in *Proc. Int. Conf. Knowl. Sci., Eng. Manag.* Cham, Switzerland: Springer, 2020, pp. 388–396.
[17] F. Zhang, Y. Mei, S. Nguyen, M. Zhang, and K. C. Tan, "Surrogate-assisted evolutionary multitask genetic programming for dynamic flexible job shop scheduling," *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 651–665, Aug. 2021.
[18] E.-D. Jiang, L. Wang, and Z.-P. Peng, "Solving energy-efficient distributed job shop scheduling via multi-objective evolutionary algorithm with decomposition," *Swarm Evol. Comput.*, vol. 58, Nov. 2020, Art. no. 100745.
[19] Z. Liu, Z. Feng, and L. Ke, "Fireworks algorithm for the multi-satellite control resource scheduling problem," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2015, pp. 1280–1286.
[20] M. Chen, J. Wen, Y.-J. Song, L.-N. Xing, and Y.-W. Chen, "A population perturbation and elimination strategy based genetic algorithm for multi-satellite TT&C scheduling problem," *Swarm Evol. Comput.*, vol. 65, Aug. 2021, Art. no. 100912.
[21] Z. Zhang, N. Zhang, and Z. Feng, "Multi-satellite control resource scheduling based on ant colony optimization," *Exp. Syst. Appl.*, vol. 41, no. 6, pp. 2816–2823, 2014.
[22] P. Gao, Y. J. Tan, J. F. Li, and R. J. He, "An ant colony algorithm for remote satellite and ground integration scheduling problem in parallel environment," in *Advanced Materials Research*, vol. 791. Baech, Switzerland: Trans Tech, 2013, pp. 1341–1346.

[23] Z. Zhang, F. Hu, and N. Zhang, "Ant colony algorithm for satellite control resource scheduling problem," *Appl. Intell.*, vol. 48, no. 10, pp. 3295–3305, 2018.
[24] T. J. Zhang, J. S. Li, and J. Li, "Space-ground integrated scheduling based on the hybrid ant colony optimization," *Syst. Eng. Electron.*, vol. 38, no. 7, pp. 1555–1562, 2016.
[25] Y. Chen, D. Zhang, M. Zhou, and H. Zou, "Multi-satellite observation scheduling algorithm based on hybrid genetic particle swarm optimization," in *Advances in Information Technology and Industry Applications* (Lecture Notes in Electrical Engineering). Berlin, Germany: Springer, 2012, pp. 441–448.
[26] K. Cui, J. Xiang, and Y. Zhang, "Mission planning optimization of video satellite for ground multi-object staring imaging," *Adv. Space Res.*, vol. 61, no. 6, pp. 1476–1489, Mar. 2018.
[27] W. Yang, Y. Chen, R. He, Z. Chang, and Y. Chen, "The bi-objective active-scan agile earth observation satellite scheduling problem: Modeling and solution approach," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2018, pp. 1–6.
[28] J. Xiong, R. Leus, Z. Yang, and H. A. Abbass, "Evolutionary multi-objective resource allocation and scheduling in the Chinese navigation satellite system project," *Eur. J. Oper. Res.*, vol. 251, no. 2, pp. 662–675, Jun. 2016.
[29] F. Xhafa, X. Herrero, A. Barolli, L. Barolli, and M. Takizawa, "Evaluation of struggle strategy in genetic algorithms for ground stations scheduling problem," *J. Comput. Syst. Sci.*, vol. 79, no. 7, pp. 1086–1100, Nov. 2013.
[30] M. Antoniou, G. Petelin, and G. Papa, "Preferred solutions of the ground station scheduling problem using NSGA-III with weighted reference points selection," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2021, pp. 1840–1847.
[31] M. Antoniou, G. Petelin, and G. Papa, "On formulating the ground scheduling problem as a multi-objective bilevel problem," in *Proc. Int. Conf. Bioinspired Methods Their Appl.* Cham, Switzerland: Springer, 2020, pp. 177–188.
[32] G. Petelin, M. Antoniou, and G. Papa, "Multi-objective approaches to ground station scheduling for optimization of communication with satellites," *Optim. Eng.*, 2021. [Online]. Available: https://link.springer.com/article/10.1007/s11081-021-09617-z
[33] Y.-J. Song, X. Ma, X.-J. Li, L.-N. Xing, and P. Wang, "Learning-guided nondominated sorting genetic algorithm II for multi-objective satellite range scheduling problem," *Swarm Evol. Comput.*, vol. 49, pp. 194–205, Sep. 2019.
[34] Y. Liu, G. G. Yen, and D. Gong, "A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies," *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 660–674, Aug. 2019.
[35] N. Saini and S. Saha, "Multi-objective optimization techniques: A survey of the state-of-the-art and applications," *Eur. Phys. J. Special Topics*, vol. 230, no. 10, pp. 2319–2335, Sep. 2021.
[36] Y.-N. Guo, X. Zhang, D.-W. Gong, Z. Zhang, and J.-J. Yang, "Novel interactive preference-based multiobjective evolutionary optimization for bolt supporting networks," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 750–764, Aug. 2020.
[37] J.-J. Wang and L. Wang, "A bi-population cooperative memetic algorithm for distributed hybrid flow-shop scheduling," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 6, pp. 947–961, Dec. 2021.
[38] G. Gong, Q. Deng, R. Chiong, X. Gong, and H. Huang, "An effective memetic algorithm for multi-objective job-shop scheduling," *Knowl.-Based Syst.*, vol. 182, Oct. 2019, Art. no. 104840.
[39] Y. Du, L. Xing, J. Zhang, Y. Chen, and Y. He, "MOEA based memetic algorithms for multi-objective satellite range scheduling problem," *Swarm Evol. Comput.*, vol. 50, Nov. 2019, Art. no. 100576. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S2210650219302299
[40] E. Bahel and C. Trudeau, "Stability and fairness in the job scheduling problem," *Games Econ. Behav.*, vol. 117, pp. 1–14, Sep. 2019.
[41] A. J. V. Álvarez and R. S. Erwin, *An Introduction to Optimal Satellite Range Scheduling*. Berlin, Germany: Springer, 2015, vol. 106.
[42] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, pp. 115–148, Apr. 1995.
[43] C. A. C. Coello and N. C. Cortés, "Solving multiobjective optimization problems using an artificial immune system," *Genetic Program. Evolvable Mach.*, vol. 6, no. 2, pp. 163–190, 2005.
[44] K. Deb and S. Jain, "Running performance metrics for evolutionary multi-objective optimization," in *Proc. 4th Asia–Pacific Conf. Simulated Evol. Learn.*, 2002, pp. 13–20.
[45] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.
[46] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, and Y. Nojima, "Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space," in *Proc. Int. Conf. Parallel Problem Solving Nature*. Berlin, Germany: Springer, 2010, pp. 91–100.
[47] P. Zhang, J. Li, T. Li, and H. Chen, "A new many-objective evolutionary algorithm based on determinantal point processes," *IEEE Trans. Evol. Comput.*, vol. 25, no. 2, pp. 334–345, Apr. 2021.
[48] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 180–198, Apr. 2016.
[49] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2000.
[50] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Apr. 2013.
[51] Y. Liu, D. Gong, J. Sun, and Y. Jin, "A many-objective evolutionary algorithm using a one-by-one selection strategy," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2689–2702, Sep. 2017.
[52] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.
[53] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, Mar. 2011.
[54] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in Statistics*. Berlin, Germany: Springer, 1992, pp. 196–202.

MINGHUI XIONG was born in Biyang, Henan, China, in 1995. He received the M.S. degree from Space Engineering University, in 2019, where he is currently pursuing the Doctor's degree. His research interests include evolutionary multi-objective optimization and satellite range scheduling.

WEI XIONG was born in 1971. He received the Ph.D. degree from Beihang University, in 2005. He is currently a Professor with the Science and Technology on Complex Electronic System Simulation Laboratory. His research interests include the complex networks, systems engineering, and theory.

ZHENG LIU was born in Zhangjiagang, Suzhou, China, in 1990. He received the B.S. degree from Space Engineering University, in 2018, where he is currently pursuing the M.S. degree. His research interests include satellite scheduling, combinatorial optimization, and reinforcement learning.