

Research Article

A Knowledge-Based Simulated Annealing Algorithm to Multiple Satellites Mission Planning Problems

Da-Wei Jin¹ and Li-Ning Xing²

¹ Information and Safety School, Zhongnan University of Economic and Law, Wuhan 430073, China

² College of Information System and Management, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Li-Ning Xing; xinglining@gmail.com

Received 21 September 2013; Accepted 5 November 2013

Academic Editor: Zhongxiao Jia

Copyright © 2013 D.-W. Jin and L.-N. Xing. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The multiple satellites mission planning is a complex combination optimization problem. A knowledge-based simulated annealing algorithm is proposed to the multiple satellites mission planning problems. The experimental results suggest that the proposed algorithm is effective to the given problem. The knowledge-based simulated annealing method will provide a useful reference for the improvement of existing optimization approaches.

1. Introduction

The imaging satellite plays a significant role in various fields such as the disaster prevention and environmental protection and attracts great attention of many scholars in the world [1, 2]. Although the number of satellites in orbit is increasing, the imaging satellite resources are still limited compared with the rapid growing imaging requirements [3, 4]. To improve the performance of satellite resources, the appropriate mathematical models and software tools must be employed to realize a better management and allocation of satellite resources [5, 6].

With the development of aerospace industry, the number and classification of imaging satellites are increasing. Also, both the number and the category of imaging requirements are largely increased [7]. It is very necessary to make a comprehensive scheduling of multiple imaging satellites [8, 9]. Therefore, this paper proposes a novel method to deal with the multiple satellites mission planning problem [10, 11].

The motivation of this work can be summarized as follows. At first, the multiple satellites mission planning is a classical combination optimization problem and it is very urgent to develop some new methods to deal with this problem. At second, the interaction between evolution and learning is more popular than before, so a knowledge-based

simulated annealing algorithm is designed and implemented according to this fashion.

2. Multiple Satellites Mission Planning Problems

The multiple satellites mission planning problem can be summarized as follows. Under the conditions of satisfying the resource constraints and imaging requirements, make a comprehensive scheduling of spot targets and regional targets, allocate satellite resources and working time for each target, and develop an optimized observation plan to maximize the summation of priority of completed targets. There are two subproblems to be solved in this problem. One is how to make a comprehensive scheduling of spot targets and regional targets, based on which the other one is how to optimize the observation plan of satellite resources. The multiple satellites mission planning is a complex combination optimization problem.

2.1. Hypothesis

- (1) There is only one sensor in each satellite. In fact, if there are multiple sensors in one satellite, then this

satellite can be divided into multiple satellites with one sensor. For this reason, both the sensor s_i and the satellite s_i have the same meaning.

- (2) The regional target can be divided into multiple atomic tasks. In fact, a spot target just is one atomic task. In this paper, the atomic task o_{ijkv} means the v^{th} subtask of task t_j which can be observed in the k^{th} time window of satellite s_i . The regional target t_j is divided into the following sets:

$$O_j = \bigcup_{i=1}^{N_s} \bigcup_{k=1}^{N_{ij}} \bigcup_{v=1}^{N_{ijk}} o_{ijkv}, \quad (1)$$

where N_s denotes the number of satellites, N_{ij} denotes the number of visible time windows about sensor s_i and target t_j , and N_{ijk} denotes the number of atomic tasks which can be observed in the k^{th} time window of satellite s_i .

- (3) In order to improve the performance of imaging resources, multiple atomic tasks are integrated into one composite task, which can be observed at one time.

2.2. Inputs

- (1) The number of satellites is N_s . Span_i denotes the longest working time of satellite s_i within one single booting.
- (2) M_i and P_i denote the maximum of storage capacity and energy of satellite s_i , m_i , and p_i refer to the storage and energy consumption of satellite s_i for observation in unit time.
- (3) R_i denotes the maximum number of side-view imaging in each circle, ϑ_i is the swaying speed of satellite s_i (degree per second), ρ_i refers to the energy consumption of swaying in each degree, d_i is the settling time of swaying of satellite s_i .
- (4) The number of tasks is N_T . Suppose the first h targets are spot ones, and the last $N_T - h$ targets are regional ones. LP_i means the priority of target i .
- (5) $VW_{ij} = \{vw_{ij}^1, vw_{ij}^2, \dots, vw_{ij}^{N_{ij}}\}$ denotes the set of time windows about sensor s_i and target t_j , and N_{ij} is the total number of visible time windows about sensor s_i and target t_j .
- (6) ws_{ijkv} , we_{ijkv} , and g_{ijkv} refer to the starting time, ending time, and observation angle of atomic task o_{ijkv} , respectively.
- (7) b_{il} denotes the l^{th} composite task of satellite s_i , while ws_{il} , we_{il} , and g_{il} refer to the starting time, ending time, and observation angle of composite task b_{il} .
- (8) $B_i = \{b_{i1}, b_{i2}, \dots, b_{iN_{B_i}}\}$ denotes the set of composite tasks of satellite s_i , in which N_{B_i} is the number of composite tasks.

2.3. *Outputs.* In the multiple satellites mission planning problem, the outputs (decision variables) can be summarized as follows:

$$x_{ijkv}^l = \begin{cases} 1, & \text{The atomic task } o_{ijkv} \text{ was executed} \\ & \text{in the } l^{\text{th}} \text{ composite tasks of satellite } s_i \\ 0, & \text{Otherwise.} \end{cases} \quad (2)$$

2.4. *Objectives.* The objective of this problem is to maximize the priority summation of fulfilled tasks. That is,

$$\max J = C_{\text{Spot}} + C_{\text{Polygon}}, \quad (3)$$

where C_{Spot} denotes the priority summation of spot targets, and it is

$$\begin{aligned} C_{\text{Spot}} &= \sum_{j=1}^h C_{\text{Spot}}(t_j) \\ &= \sum_{j=1}^h \sum_{i=1}^{N_s} \sum_{k=1}^{N_{ij}} \sum_{l=1}^{N_{B_j}} x_{ijk1}^l \times \text{LP}_j. \end{aligned} \quad (4)$$

C_{Polygon} means the priority summation of regional targets. This computation process is a little complex. At first, we should compute the overlap area between the multiple observed atomic tasks and regional target t_j . That is,

$$S_{\text{Polygon}} = \left(\bigcup_{i=1}^{N_s} \bigcup_{k=1}^{N_{ij}} \bigcup_{v=1}^{N_{ijk}} \bigcup_{l=1}^{N_{B_j}} x_{ijkv}^l \psi(o_{ijkv}) \right) \cap \psi(t_j). \quad (5)$$

Here, $\psi(o_{ijkv})$ denotes the polygon area of atomic task o_{ijkv} , \cap denotes the intersection operation of multiple sets, $\psi(t_j)$ denotes the polygon area of regional target t_j , and \cup denotes the union operation of different sets. At second, we should compute the coverage rate of each regional target. It is

$$\text{Cover}(t_j) = \frac{f(S_{\text{Polygon}})}{f(t_j)}, \quad (6)$$

where $f(S_{\text{Polygon}})$ denotes the proportion of polygon area S_{Polygon} , and $f(t_j)$ denotes the proportion of regional target t_j . In total, the priority summation of regional targets can be computed as follows:

$$C_{\text{Polygon}} = \sum_{j=h+1}^{N_T} [\text{Cover}(t_j) \times \text{LP}_j]. \quad (7)$$

2.5. Constraints

- (1) Uniqueness constraint on spot targets; that is, each spot target can only be observed once:

$$\sum_{i=1}^{N_s} \sum_{k=1}^{N_{ij}} \sum_{l=1}^{N_{B_j}} x_{ijk1}^l \leq 1, \quad \forall j = 1, \dots, h. \quad (8)$$

- (2) Conversion time constraints between composite tasks, it is

$$we_{il} + \frac{|g_{il} - g_{i(l+1)}|}{\vartheta_i} + d_i \leq ws_{i(l+1)}, \quad (9)$$

$$\forall i = 1, \dots, N_S; l = 1, \dots, N_{B_i} - 1.$$

Each composite task corresponds to an observation activity of the satellite, and every two composite tasks must satisfy the corresponding conversion time constraint. The conversion time include the rotation time and settling time of the sway.

- (3) Storage constraints of the satellite:

$$\sum_{l=1}^{N_{B_i}} (we_{il} - ws_{il}) m_i \leq M_i, \quad \forall i = 1, \dots, N_S. \quad (10)$$

- (4) Energy constraints of the satellite:

$$\sum_{l=1}^{N_{B_i}} (we_{il} - ws_{il}) p_i + \sum_{l=1}^{N_{B_i}-1} |g_{i(l+1)} - g_{il}| \rho_i \leq P_i, \quad (11)$$

$$\forall i = 1, \dots, N_S.$$

The satellites usually consume some energy during the imaging and swaying process, so the corresponding energy consumption can be represented in a function about the observation time of satellites and sensor's swaying. The energy consumption should not exceed the largest energy limitation.

- (5) Constraint on the maximum number of side-view imaging in a single circle. We can obtain the identifier of orbit circles according to the starting time and ending time of observation activities and then verify this constraint. Suppose the orbit set of satellite s_i is

$$\text{Orbit}_i = \left\{ \text{orbit}_{i1}, \text{orbit}_{i2}, \dots, \text{orbit}_{iN_{O_i}} \right\} \quad (12)$$

in which N_{O_i} is the number of orbits, and orbit_{ir} refers to orbit r of satellite s_i . Given that orbit_{ir} contains N_{ir} composite tasks $\{b_{ir_1}, b_{ir_2}, \dots, b_{ir_{N_{ir}}}\}$,

$$N_{ir} \leq R_i, \quad \forall i = 1, \dots, N_S, r = 1, \dots, N_{O_i}. \quad (13)$$

3. The Proposed Approach

In the existing intelligent optimization methods, they have not involved the effective management (i.e., learning, storage, and utilization) to domain knowledge of practical problems, and thus they cannot effectively obtain the optimal solution of optimization problems. Based on our knowledge, the complex optimization problem can be solved through learning some related knowledge from the optimization process and then employing it to guide the subsequent optimization process. In this paper, we mainly focus on the explicit

knowledge, namely, the knowledge that can be definitely expressed. All the explicit knowledge can be expressed and stored through computer language and updated and adopted by other models.

In this paper, the knowledge-based simulated annealing algorithm is defined as a hybrid approach which effectively combines the simulated annealing model and knowledge model, in which the simulated annealing model searches the solution space of optimization problems by "neighborhood search" strategy, and the knowledge model learns useful knowledge from the optimization process and then uses it to guide the subsequent optimization process. The knowledge-based simulated annealing algorithm employs the modeling idea of integrating the knowledge model and the simulated annealing model. Its basic framework is shown in Figure 1.

The working mechanism of knowledge-based simulated annealing algorithm is shown in Figure 2. The top illustrates the optimization process of simulated annealing algorithm: search the feasible space of optimization problems through the "neighborhood search" strategy and then converge to the near-optimal solution through continually iterations. And the bottom shows the effect of knowledge model: learn the useful knowledge from the early optimization process and then apply it to guide the subsequent optimization process.

During the early operation process of knowledge-based simulated annealing algorithm, due to that the available samples are not sufficient, so the reliability of learned knowledge is not high, and the guidance to optimization process is not distinct. However, with the advance of iterations, the reliability of learned knowledge is higher than before, and the guidance to optimization process is more distinct. Compared with traditional simulated annealing algorithm, with the help of knowledge model, the knowledge-based simulated annealing algorithm can either converge to a satisfactory solution more quickly or converge to a satisfactory solution with higher quality.

3.1. Knowledge Definition, Learning, and Application

(1) *Operator Knowledge*. Usually, different operators have different application scopes. When solving practical problems, a single operator may be more effective to some problems than to others. Therefore, it is very difficult to find out a general operator that can solve all kinds of practical problems effectively. In order to improve the performance of simulated annealing algorithm, we can adopt several kinds of operators for different operations in simulated annealing algorithm, with the hope of trying to mine some operators that can effectively solve the present problem.

When an operator is employed to one operation, suppose the original individual set is C_B and the newly generated individual set is C_A , and if the optimal individual in C_A is better than that in C_B , then the present operation is considered to be successful. The optimization performance of one operator can be defined as the number of successful operations realized by the given operator. Operator knowledge mainly

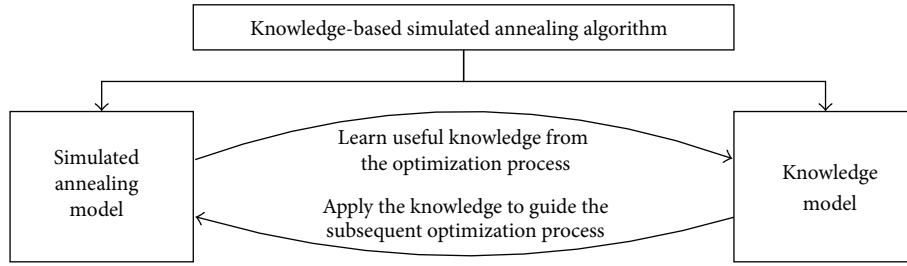


FIGURE 1: The basic framework of knowledge-based simulated annealing algorithm.

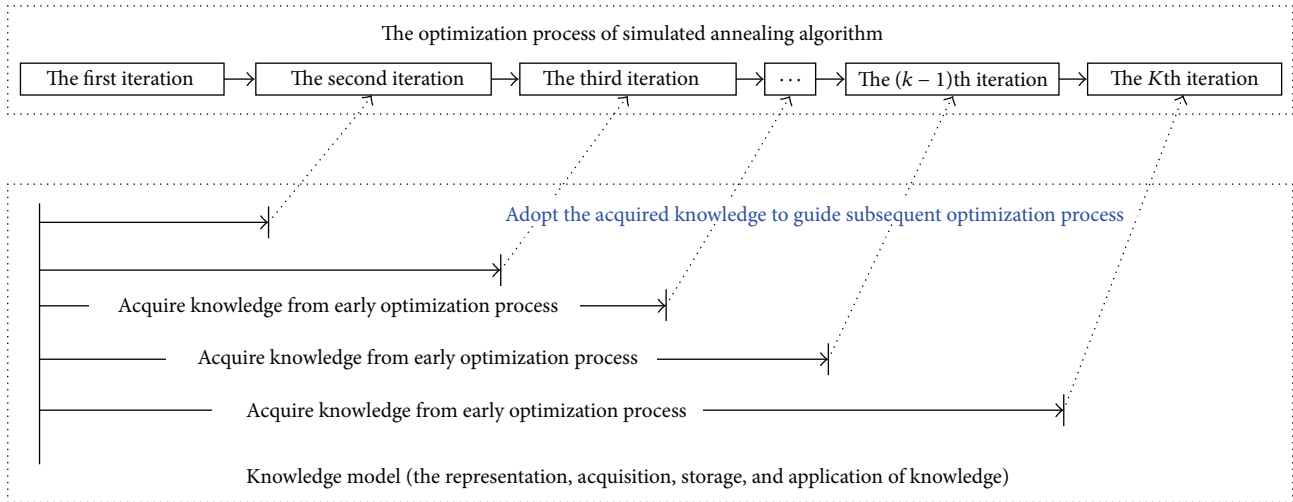


FIGURE 2: The operation mechanism of knowledge-based simulated annealing algorithm.

refers to the accumulated knowledge for the optimization performance of various operators.

The simple case for extraction and application of operator knowledge is displayed as follows. Suppose there are three operators in the simulated annealing algorithm, if the current operation implemented by the k^{th} operator is successful, then the optimization performance $N(k)$ of the k^{th} operator will be added by one. During the next operation, the selection probability for each operator will be calculated according to the following formula:

$$P(i) = \frac{N(i)}{\sum_{i=1}^3 N(i)}, \quad (14)$$

where $P(i)$ denotes the probability that the i^{th} operator is selected for the operation.

(2) *Parameter Knowledge.* One simple and easy method to improve the optimization performance of simulated annealing algorithm is to adjust the parameter values dynamically during the optimization process. To lower the sensitivity of parameters to experimental results, we employ several different parameter combinations to develop the evolution process and decide the parameter combination selected for the next iteration according to their optimization performance. After a single iteration, if the global optimal solution has been improved, then it is a successful iteration. In the process

of solving current cases, the number of successful iterations obtained by the given parameter combination is regarded as its optimization performance. Parameter knowledge mainly refers to the accumulated knowledge of the optimization performance for various parameter combinations.

During the initialization stage, the orthogonal method can be adopted to generate different parameter combinations and all their optimization performance is initialized to 1. Before any iteration of simulated annealing algorithm, a parameter combination will be selected stochastically from several parameter combinations through roulette selection (based on the optimization performance of parameter combinations) as the parameter of current iteration. If the global optimal solution has been improved in this iteration, then the optimization performance value of the parameter combination used in this iteration will be increased.

3.2. *The Available Neighborhood Structures.* The following neighborhood structures have been designed based on the basic ones and passed the constraint checking of the basic neighborhood structure, thus greatly simplifying the optimization calculation process.

(1) *Arrange.* The scheduling of imaging satellites for composing tasks usually involves the task arrangement which is performed through composing or inserting the contained

atomic tasks. Neighborhood composition is preferred in the arrangement of atomic tasks, and the insertion operation is adopted only when atomic tasks cannot be composed or the task has not been finished, which contributes to the observation composition of much more tasks and helps to arrange more tasks.

(2) *Relocate*. This kind of neighborhood can reallocate the satellite resource and time windows of tasks. There are several time windows about satellites and tasks, so we can reallocate time windows for the task as to the same satellite. Two kinds of operations are involved here: one is to reallocate the task to different time windows about the same satellite and the other is to reallocate the task to another satellite. The relocating aims to increase the chance of arranging other tasks through adjusting the satellite resource and time windows of tasks.

(3) *Swap*. This neighborhood aims to exchange two tasks. About parallel scheduling problems, three kinds of neighborhood structures for task exchange are mentioned: the exchange between two tasks with the same resource (exchanging the execution order); the exchange between two tasks with different resources; the exchange between the scheduled tasks and unscheduled ones. To the multiple satellites mission planning problem, the relevant experiments indicate that the first two exchanges are unsuccessful. Therefore, this paper adopts the third kind of neighborhood structure for task exchange, namely, to exchange tasks under the real satellite resource and the virtual one.

(4) *Sample*. Random sampling neighborhood is a special measure adopted to handle large-scale neighborhood effectively. It is not a new neighborhood structure itself, but a method to generate new neighbourhood structures through sampling the existing ones randomly. To be specific, before each move, random sampling neighbourhood will sample all neighbors generated by the given neighbourhood structure, so as to limit the move to part of the sampled neighbourhood, and the proportion of sampling is usually specified in advance upon requirements.

Among the designs of neighbourhood structure, the arrangement of neighbourhood is designed to implement the insertion and composition operations among atomic tasks in which the task composition is preferred, and it is a kind of improved neighbourhood; the neighbourhood relocation and swap disturb the solutions, respectively, through adjusting the resources and time windows of the task as well as exchanging the scheduled tasks with unscheduled ones, and they belong to a regulative neighbourhood which can increase the chance to arrange other tasks after the relocation and swap of tasks, so they should be tried on all the unfinished tasks.

4. Experimental Results

Six sets of even-distributed targets are generated, respectively, with the number of 100, 200, 300, 400, 500, and 600. Based on a different number of satellites, several test cases are then constructed for each set of targets. Each case will be

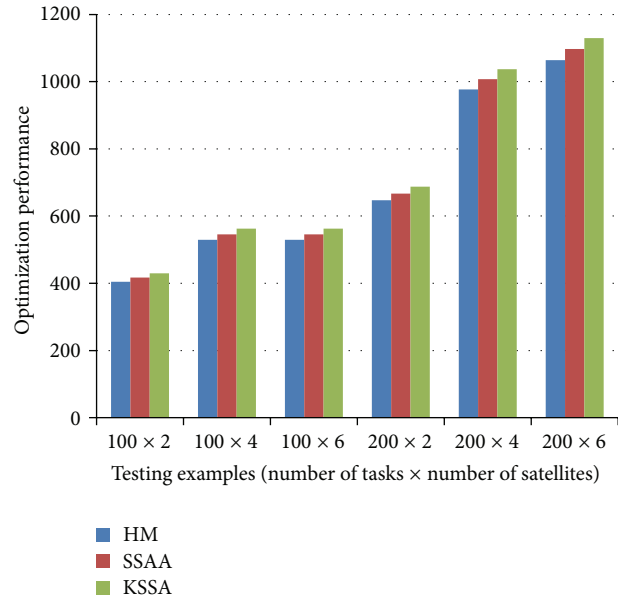


FIGURE 3: Experimental results in optimization performance of small scale testing examples.

calculated for fifty times and the comparison will be made on the averages of the results to avoid the influence of randomness. To validate the performance of this algorithm, the Heuristic Method (HM) and the Standard Simulated Annealing Algorithm (SSAA) are employed to be compared with the Knowledge-Based Simulated Annealing Algorithm (KSAA).

4.1. *Experimental Results of Small Scale Testing Examples*. Six small scale testing examples (100×2 , 100×4 , 100×6 , 200×2 , 200×4 , 200×6) are applied to test the performance of different algorithms. Experimental results of small scale testing examples are displayed in Figures 3 and 4. Experimental results in Figure 3 suggest that there is no remarkable difference in optimization performance among HM, SSAA, and KSSA. Experimental results in Figure 4 suggest that the computational time of SSAA and KSSA is little larger than that of HM. Also, there is no distinct difference in computational time between SSAA and KSSA. Since SSAA and KSSA are all intelligent optimization approaches, so the computational time of SSAA and KSSA is larger than that of HM.

4.2. *Experimental Results of Medium Scale Testing Examples*. Six medium scale testing examples (300×2 , 300×4 , 300×6 , 400×2 , 400×4 , 400×6) are applied to test the performance of different algorithms. Experimental results of medium scale testing examples are displayed in Figures 5 and 6. Experimental results in Figure 5 suggest that there is no significant difference in optimization performance among SSAA and KSSA, but the optimization performance of SSAA and KSSA is better than that of HM. Since SSAA and KSSA are all intelligent optimization approaches, so the optimization performance of SSAA and KSSA should be

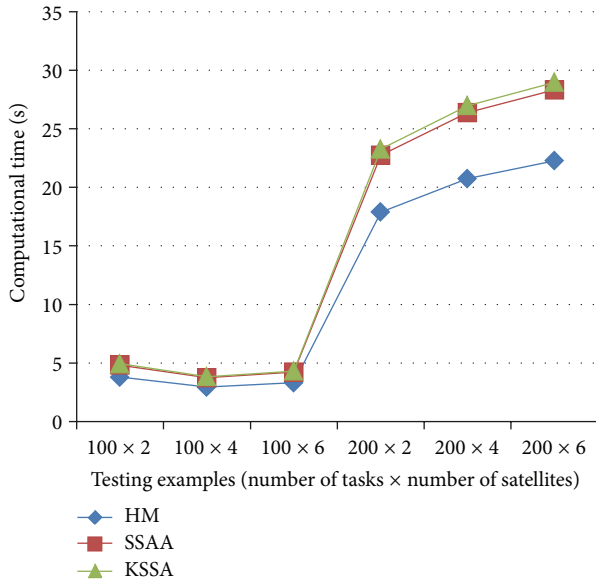


FIGURE 4: Experimental results in computational time of small scale testing examples.

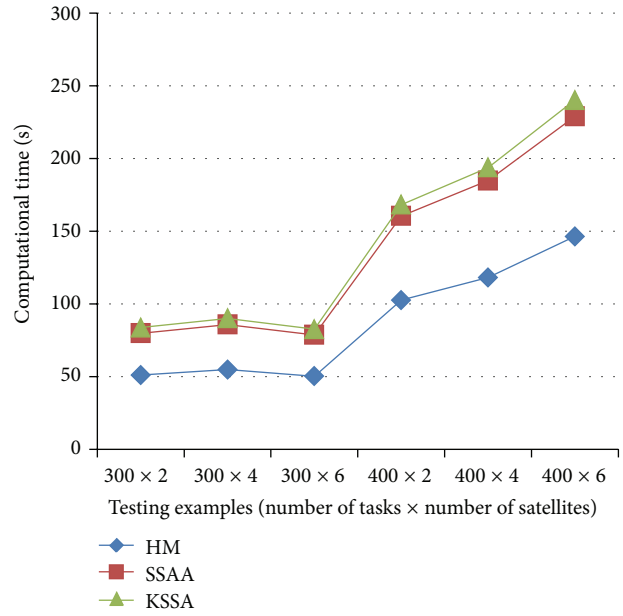


FIGURE 6: Experimental results in computational time of medium scale testing examples.

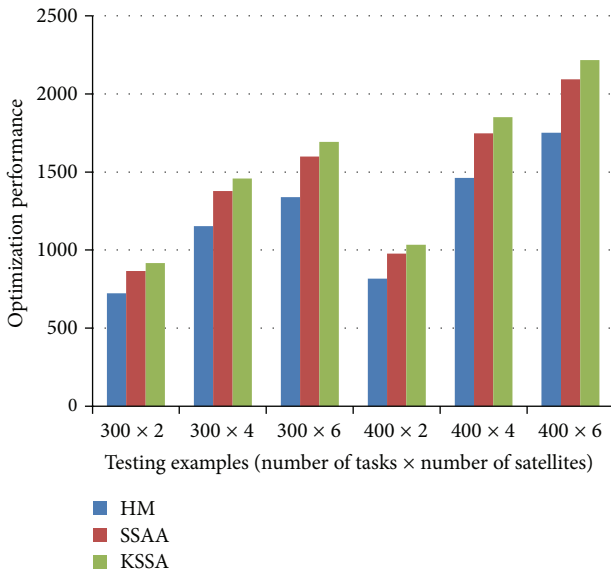


FIGURE 5: Experimental results in optimization performance of medium scale testing examples.

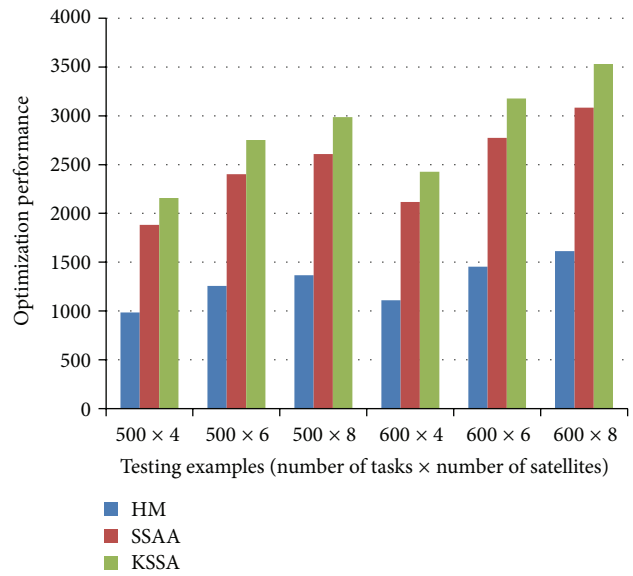


FIGURE 7: Experimental results in optimization performance of large scale testing examples.

better than that of HM, and the experimental result indicates this viewpoint. Experimental results in Figure 6 suggest that the computational time of SSAA and KSSA is little larger than that of HM. Also, there is no distinct difference in computational time between SSAA and KSSA.

4.3. *Experimental Results of Large Scale Testing Examples.* Six large scale testing examples ($500 \times 4, 500 \times 6, 500 \times 8, 600 \times 4, 600 \times 6, 600 \times 8$) are applied to test the performance of different algorithms. The experimental results of large scale testing examples are displayed in Figures 7 and 8.

Experimental results in Figure 7 suggest that the optimization performance of SSAA is better than that of HM, and the optimization performance of KSSA is better than that of SSAA. Since SSAA is an intelligent optimization approach, so the optimization performance of SSAA should be better than that of HM, and experimental results indicate this viewpoint. At the same time, KSSA extended the learning and guidance function based on the SSAA, so its performance should be better than that of SSAA, and experimental results validate this opinion. Experimental results in Figure 8 suggest that

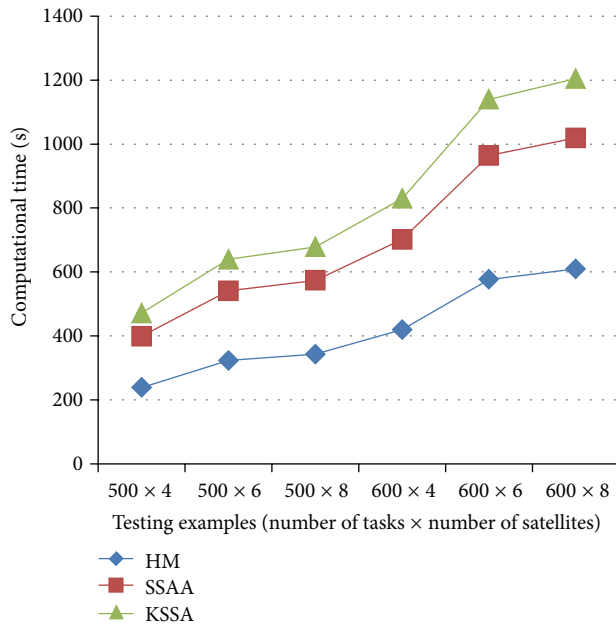


FIGURE 8: Experimental results in computational time of large scale testing examples.

the computational time of SSAA is larger than that of HM, and the computational time of KSSA is larger than that of SSAA. Since SSAA is an intelligent optimization approach, so it needs more computation cost than that of HM to deal with large scale testing examples. Also, the number of modules in KSSA is more than that in SSAA, so it needs more computation cost than that of SSAA to deal with large scale testing examples.

5. Conclusions

The innovation of this paper lies in that it has established a multisatellite mission planning model based on the constraint satisfaction and proposed a knowledge-based simulated annealing algorithm to solve multisatellite mission planning problems, which has been proved effective in the relevant experiment.

The future research directions can be summarized as follows. (1) Extend the types of knowledge. Extracting the experiential knowledge of experts to multiple satellites mission planning problems, we could employ this knowledge to guide the optimization process effectively so as to enhance optimization efficiency as much as possible. (2) Adopt new knowledge mining techniques. We should consider employing advanced knowledge mining through machine learning or data mining to obtain useful knowledge from the optimization process.

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgments

The authors would like to thank the reviewers for the valuable comments and constructive suggestions. they also would like to thank the editor-in-chief, associate editor and editors for helpful suggestions. This work is supported by the National Natural Science Foundation of China (nos. 71101150, 71071156, and 71331008), Program for New Century Excellent Talents in University, the Ministry of Education in China (MOE) Project of Humanities and Social Sciences (no. 12YJC630078).

References

- [1] V. Gerard and L. Michel, "Planning activities for Earth watching and observing satellites and constellations," in *Proceedings of the 16th International Conference on Automated Planning and Scheduling (ICAPS '06)*, Cumbria, UK, 2006.
- [2] F. N. Kucinskis and M. G. V. Ferreira, "Planning on-board satellites for the goal-based operations for space missions," *IEEE Latin America Transactions*, vol. 11, no. 4, pp. 1110–1120, 2013.
- [3] A. Manzak and H. Goksu, "Application of Very Fast Simulated Reannealing (VFSR) to low power design," in *Proceedings of the 5th International Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '05)*, pp. 308–313, Springer, Berlin, Germany, July 2005.
- [4] Z. Y. Lian, Y. Wang, Y. J. Tan et al., "Integration solving framework for global navigation satellite system planning and scheduling problem," *Disaster Advanced*, vol. 6, no. 1, pp. 330–336, 2013.
- [5] S. Rojanasoonthon and J. Bard, "A GRASP for parallel machine scheduling with time windows," *INFORMS Journal on Computing*, vol. 17, no. 1, pp. 32–51, 2005.
- [6] R. J. He and L. N. Xing, "A learnable ant colony optimization to the mission planning of multiple satellites," *Research Journal of Chemistry and Environment*, vol. 16, no. S2, pp. 18–26, 2012.
- [7] B. Suman and P. Kumar, "A survey of simulated annealing as a tool for single and multiobjective optimization," *Journal of the Operational Research Society*, vol. 57, no. 10, pp. 1143–1160, 2006.
- [8] A. Globus, J. Crawford, J. Lohn, and A. Pryor, "A comparison of techniques for scheduling earth observing satellites," in *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-2004): Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI '04)*, pp. 836–843, July 2004.
- [9] D. A. Knapp, D. S. Roffman, and W. J. Cooper, "Growth of a pharmacy school through planning, cooperation, and establishment of a satellite campus," *American Journal of Pharmaceutical Education*, vol. 73, no. 6, article 102, 2009.
- [10] B. Sudarshan and D. R. Nikil, "Very fast simulated annealing for HW-SW partitioning," Tech. Rep. CECS-TR-04-17. UC, Irvine, 2004.
- [11] C. Wang, J. Li, N. Jing, J. Wang, and H. Chen, "A distributed cooperative dynamic task planning algorithm for multiple satellites based on multi-agent hybrid learning," *Chinese Journal of Aeronautics*, vol. 24, no. 4, pp. 493–505, 2011.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

