Improved MOMVO algorithm to solve the multi-objective HFSP considering machine shutdown

Hui Huang1,2,*, Bin Luo1,2, Zhongning Wang1,2, Hao Yu1,2

1College of Mechanical Engineering, Chongqing University, Chongqing, 400044.
2China.b State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing, 400044, China.
*Corresponding author’s E-mail: 201807021057@cqu.edu.cn

Abstract. Machine shutdown is a common phenomenon in production workshops, and it is also an important issue that many workshops should consider when scheduling. Based on the multi-variety characteristics of hybrid flow shop scheduling problem (HFSP), this paper fully considers the machine shutdown constraint, establishes an HFSP model with dual goals of maximum completion time and machine load balance, and improves the multi-objective multi-verse optimizer (MOMVO) to calculate an example based on actual production data assumptions. The research results showed that the scheduling scheme obtained by this model and algorithm meets the actual needs and can provide an effective scheduling scheme for the actual production of the enterprise.

1. Introduction
HFSP refers to the rational allocation of the sorting of processing workpieces and the assignment of machinery and equipment within a certain period so that certain goals can reach the optimal value [1]. HFSP is widely used in chemical, textile, steel, semiconductor, tobacco processing, Chinese herbal medicine production, and other industries [2-3]. HFSP combines the characteristics of flow shop scheduling and parallel machine scheduling, which makes HFSP more complex and more difficult to solve, and it has been proven to be an NP-Hard problem.

In recent years, the literature on HFSP issues has gradually increased. Zhang, C.J. et al. studied the HFSP problem with a limited buffer and two process routes and proposed a discrete whale swarm algorithm to find the approximate optimal solution of the scheduling problem [4]. Sun, Y.J. et al. considered the diversity of customer needs and aimed to reduce completion time and delay costs, and established a scheduling model. At the same time, they proposed a hybrid meta-heuristic algorithm including differential evolution and local search [5]. Ebrahimi, M. et al. studied the scheduling problem that relies on the sequence setting time in the hybrid flow shop, taking into account the randomness and uncertainty of the delivery date, establishing a model, and proposing two meta-heuristic algorithms based on genetic algorithms to solve it [6]. Han, W. et al. described and attributed the HFSP problem to the Markov decision process, and studied the reinforcement learning method of HFSP [7]. Zhao, Z. et al. proposed an improved genetic algorithm based on parallel sequential movement and variable mutation rate for the multi-objective HFSP problem [8].

In the existing research work on HFSP, there is almost no research on scheduling problems with machine shutdown constraints, and most of the literature is studying single-objectives. However, in actual hybrid flow production, machine shutdown and multiple scheduling objectives are very common.
problems, and most production systems have these two problems at the same time. If we ignore these two problems to study production scheduling, the result will have large deviations from actual production, which will greatly reduce the value and significance of the research. This paper fully considers the two problems of machine shutdown and multi-objective scheduling, establishes the corresponding mathematical model, and uses the improved MOMVO algorithm to solve it. The primary contributions of this paper include:

• The machine shutdown constraint is added to the classic HFSP, making the scheduling process closer to reality.
• A dual-objective model that balances the maximum completion time and machine load is established, so that the scheduling plan can achieve the comprehensive optimal.
• An improved MVMVO algorithm is proposed to achieve the best results through the two synchronous processes of the exploration cycle and the development cycle.

2. Establishment of HFSP model considering machine shutdown

2.1. Description of HFSP considering machine shutdown

HFSP considering machine shutdown can be described as follows: n workpieces are processed continuously in S (S≥2) processing stages, and stage s has m_s (m_s≥1; s=1,2,…, S) parallel machines. Among all machines, some machines need to be shut down for a certain period during the entire processing process. Figure 1 is a schematic diagram of the HFSP considering machine shutdown. The machine in the dashed frame in the figure represents a shutdown during processing, while all machines in the traditional HFSP do not have a shutdown. When the machine shuts down, the workpiece needs to be processed on other machines at the same stage or wait for the machine to resume work before continuing to process on this machine. The common scenarios of machine shutdown are: the machine shuts down for maintenance according to the inspection table, or it is predicted that the life of the tool (or other parts) on the machine will be exhausted and needs to shut the machine down for replacement. The machine's shutdown information and the processing time of the workpiece on the machine are known.

2.2. Parameter settings

i: Workpiece number, i ∈ {1,2,3…n};
s: Processing stage (operation) number, s ∈ {1,2,3…S};
p: Statistical number of the number of parallel machines in stage s, p ∈ {1,2,3…m_s};
m_min: The smallest number of parallel machines in stage s;
m_max: The biggest number of parallel machines in stage s;
k_s: Number of parallel machines in stage s, k_s ∈ [m_min, m_max], k_s and p point to the same machine;
N_s,p: The total number of workpieces processed by machine P in stage s;
t_i,p: Processing time of workpiece i on machine p in processing stage s;
T(i,s): Processing time of workpiece i in processing stage s;
C(i,s,k_s): Processing time of workpiece i on machine k_s in stage s;
S(i,s,ks): Start processing time of workpiece i on Machine ks in stage s;
q: The position index of the workpiece processed on each machine in each stage,q∈{1,2,3…Ns,p};
U_{i,s,p,q}: if workpiece i is processed at the q-th position on the machine p of stage s, U_{i,s,p,q} =1, otherwise
\quad U_{i,s,p,q} =0;
S_{q,p}^{s}: The starting time of the q-th workpiece on the machine p of stage s;
C_{q,p}^{s}: The completion time of the q-th workpiece on the machine p of stage s.

2.3. mathematical model
Select makespan as the objective function \(f(1)\) and the expression of \(f(1)\) is shown in formula (1). The smaller makespan is, the smaller the total processing time is. For enterprises, it is of great significance to save production time, shorten the production cycle and improve production capacity.

\[ f(1) = \text{makespan} = \max \{ C(i,S,k_S) \} \quad (1) \]

Select the machine load balance index, that is, the average value \(T_{lb}\) of the standard deviation of the processing time of each machine and equipment in each processing stage, as the objective function \(f(2)\) the expression of \(f(2)\) is shown in formula (2) and formula (3). The smaller the \(T_{lb}\), the smaller the long-term difference and the fewer jumps during the processing of each section on the same machine. This is beneficial to reduce machine loss, extend the service life of the machine, and further reduce the production cost of the enterprise.

\[ f(2) = T_{lb} = \frac{1}{S} \sum_{s=1}^{S} \frac{\sum_{p=1}^{N_s} \sum_{q=1}^{N_{i,p}} \left( C_{i,p}^{s} - S_{i,p}^{s} \right) - \overline{T_{W_s}}}{m_s} \]  

\[ \overline{T_{W_s}} = \frac{\sum_{p=1}^{N_s} \sum_{q=1}^{N_{i,p}} \left( C_{i,p}^{s} - S_{i,p}^{s} \right)}{m_s}, s \in \{1,2,3,\ldots,S\} \quad (2) \]

The mathematical function with the dual objectives of minimizing the maximum completion time and minimizing the machine load balance is:

\[ f = \min \{ f(1), f(2) \} \quad (4) \]

\[ \sum_{p=1}^{N_s} \sum_{q=1}^{N_{i,p}} U_{i,s,p,q} = 1 \]  

\[ \sum_{p=1}^{N_s} N_{i,p} = n \]  

\[ C(i,s,k_s) = S(i,s,k_s) + U_{i,s,p,q} \times t_{i,s,p} \]  

(5)

Formula (a) indicates that any workpiece can only be processed on one machine once in any processing stage; Formula (b) indicates that any workpiece must be processed through each processing stage; Formula (c) indicates that the completion time of the workpiece is equal to the start time plus the processing time of the workpiece. After the model is established, the model needs to be solved. The heuristic swarm intelligence search algorithm shows good performance in solving this kind of problem. According to similar research, it is found that the MOMVO algorithm is very suitable for solving the specific problems studied in this paper. This paper improves the original MOMVO to get IMOMVO and then uses IMOMVO to solve the model calculation examples.

3. Algorithm design of IMOMVO
MOMVO is a multi-objective version of Multi-Verse Optimizer (MVO). MVO is a swarm intelligence optimization algorithm proposed by Mirjalili, S., Mirjalili, S.M. and Hatamlou, A. in 2015 based on the
theory of multiverse in physics [9]. This algorithm has been successfully applied to function optimization and engineering design. In MOMVO, a universe is regarded as a solution to the optimization problem, and each object in a single universe is regarded as a component of the corresponding solution, and the expansion rate of a single universe is proportional to the value of the objective function of the corresponding solution. Regardless of the size of the expansion rate, the individual universe will stimulate internal objects to move to the current optimal universe to achieve local changes and improve its expansion rate. This process is executed according to formula (6).

$$x_j' = \begin{cases} X_j + TDR \times \left\{ \left( ub_j - lb_j \right) \times r3 + lb_j \right\} & r2 < 0.5 \quad r1 < WEP \\ X_j - TDR \times \left\{ \left( ub_j - lb_j \right) \times r3 + lb_j \right\} & r2 \geq 0.5 \quad r1 < WEP \\ x_j' & r1 \geq WEP \end{cases}$$

(6)

Among them, $x_j'$ is the j-th object of the i-th universe, which is equivalent to the j-th component of the i-th solution. $X_j$ represents the j-th object in the current optimal universe, $ub_j$ and $lb_j$ are the upper and lower bounds of the object $x_j'$, and $r1$, $r2$, and $r3$ are random numbers in the range of $[0,1]$. WEP and TDR are two important parameters. WEP represents the probability of the existence of wormholes in the multiverse space, and TDR represents the step length of the object moving toward the current optimal universe. They are updated according to formula (7) and formula (8) respectively.

$$WEP = WEP_{min} + \frac{time}{max\ time} \times (WEP_{max} - WEP_{min})$$

(7)

$$TDR = 1 - (time / max\ time)^{1/6}$$

(8)

Where time is the current number of iterations, and maxtime is the maximum number of iterations. When maxtime=500, the TDR value decreases concavely from 0.6 to 0. $WEP_{min}=0.2$, $WEP_{max}=1$, WEP increases linearly from 0.2 to 1.

3.1. Improvement of population initialization

Population initialization usually uses corresponding functions to randomly generate chromosomes according to the designed chromosome codes. These chromosomes are all feasible solutions to the optimization problem, and they together constitute the initial population as initial individuals. When generating the initial population, in addition to ensuring random probability, to improve the performance of the algorithm, the following processing is often needed: First, make the initial population as evenly distributed in the solution space as possible to avoid falling into local optimum; Some simple methods are used to preliminarily screen the generated initial solutions and remove the poorest part of the solution. This paper draws lessons from the opposition-based learning strategy proposed in the literature [10] and proposes a method to generate a uniformly distributed initial solution. First, generate an initial population of randomly distributed N solutions according to formula (9). Each solution is a D-dimensional vector, that is $x_j^i=(x_1^j, x_2^j, \ldots, x_D^j), i \in \{1,2,\ldots,N\}$. Then according to formula (10), N opposite learning solutions $(x_j')$ corresponding to the initial solutions are generated. In this way, 2N initial solutions are generated. According to their descending fitness, the first N solutions are selected as the initial solutions of the algorithm.

$$x_j = lb_j + rand(0,1)(ub_j-lb_j)$$

(9)

$$x_j' = lb_j + ub_j - x_j$$

(10)

Where $i \in \{1,2,\ldots,N\}, j \in \{1,2,\ldots,D\}$, $ub_j$ and $lb_j$ are the upper and lower bounds of the j-th dimension of the vector, respectively.
3.2. Improvement of TDR
In the MOMVO algorithm, the algorithm optimization mainly relies on the black hole to traverse by the wormhole and travel around the optimal universe. The travel distance rate TDR is an important variable to coordinate the algorithm exploration and development capabilities, and it is the main parameter that affects the basic MOMVO algorithm optimization. But the TDR value in the MOMVO algorithm is reduced concavely from 0.6 to 0, and the change range is too narrow. It can be seen from equation (6) that a larger TDR value is conducive to global exploration capabilities, and a smaller TDR value is conducive to local in-depth development. Here, the faster iterative trend should be maintained in the early stage of the iteration for global exploration, and a slower iterative trend should be maintained in the latter stage of the iteration for local development. Therefore, this paper proposes the following nonlinear convergence factor:

$$TDR = \left[\log_{2}(\frac{-\text{time}}{\text{max time}} + 2)\right]^{6}$$

3.3. Improvement of Elite Leading Strategy
The elite retention strategy in the original MOVMO algorithm can retain the best individuals in the population of each generation and select the best-ranked individual as the leader, which greatly improves the performance and efficiency of the algorithm. However, as the evolutionary generation increases, a large number of high-ranking level individuals that are the same as the parent population will gradually appear in the offspring population. This will lead to a rapid decline in the diversity of the population, which will cause the algorithm to appear premature or fall into a local optimum. To make up for the shortcomings of the elite selection strategy and ensure the diversity of the population to ensure that the search direction develops towards the true Pareto front, this article selects the individuals with the top ranks as the elites according to a certain probability when implementing the elite-leading strategy, instead of selecting only the individual with the best rank as the elite. This not only improves the diversity of the population but also improves the search efficiency of the algorithm to a certain extent.

3.4. Improvement of Elite Leading Strategy
Step 1: Initialize the number N of universes, WEP, TDR, maxtime, etc.
Step 2: Generate all universes in the first-generation population according to formulas (9) and (10).
Step 3: Calculate the fitness value of each universe population.
Step 4: Obtain the current optimal fitness value and the optimal universe and record them.
Step 5: Enter the main loop. Update the values of WEP and TDR according to formulas (7) and (11).
Step 6: Implement the roulette mechanism.
Step 7: Update the universes according to formula (6) to obtain the updated universe population.
Step 8: Calculate the optimal universe of the new universe population. If it is better than the current optimal universe, replace it, otherwise, the current optimal universe is still retained.
Step 9: If the current number of iterations reaches the maximum number of iterations, exit the main loop and output the optimal fitness and the optimal universe; otherwise, return to step 3.

4. Solution on the scheduling example and standard test functions by IMOMVO
The algorithm is programmed in Matlab 2016b, and the simulation experiment is performed on CPU: Intel(R) Core(TM)15-8500 CPU @ 3.00GHZ; RAM: 8.00GB; Windows 10. The commonly used MOEAD, NSGA-II, MODA and MOMVO are selected as the comparison algorithms of IMOMVO. The number of individuals in the population is 100, and the maximum number of iterations is 500.

4.1. Solution on the scheduling example by IMOMVO
In this example, there are 20 workpieces, and each workpiece needs to go through 4 processing stages. The number of machines in each stage is 2, 1, 2, 3; there are 8 machines in total. The machine's shutdown information is as follows: With the start of processing as zero time, machine 3 needs to be shut down during the 30th min to 50th minute period, and machine 6 needs to be shut down during the 80th min to 100th minute period.
PF represents the Pareto front obtained by the algorithm, and TPF represents the true Pareto front of the problem. The generation distance (GD) and the inverse generation distance (IGD) are selected as indicators to evaluate the performance of the algorithm. The smaller the GD value, the better the convergence of the PF and the closer it is to the TPF. The smaller the IGD distance, the better the overall performance of the algorithm. The TPF of this scheduling example is the approximate front obtained by summarizing the solution sets obtained by running each of the five algorithms 20 times. Then each algorithm is run independently again 20 times to obtain the average Ave and standard deviation Std of GD, and the average Ave and standard deviation Std of GD, as shown in Table 1.

| Algorithm | MODA | NSGA2 | MOEA/D | MOMVO | IMOMVO |
|-----------|------|-------|--------|--------|--------|
| GD        | 0.141740 | 0.187309 | 0.088942 | 0.123024 | 0.097824 | 0.058391 |
| IGD       | 0.071342 | 0.089420 | 0.106513 | 0.097824 | 0.058391 | 0.063040 |
| Ave       | 0.052972 | 0.012193 | 0.014219 | 0.01463 | 0.009277 | 0.058059 |
| Std       | 0.052972 | 0.012193 | 0.014219 | 0.01463 | 0.009277 | 0.058059 |

Table 1 The GD and IGD values obtained by the five algorithms to solve the scheduling example.

Figure 2 shows the PF obtained by the five algorithms to solve the scheduling example.

It can be seen from Table 2 that when solving the scheduling example, from the average Ave of GD, the convergence performance of IMOMVO is the best among the five algorithms. From the average Ave of IGD, the overall performance of IMOMVO is also the best among the five algorithms. Besides, from the standard deviation Std of GD and IGD, among the five algorithms, the solution of IMOMVO has the smallest Std value, which means that the solution is more stable. From figure 2, it can be found that the PF obtained by IMOMVO is very close to the TPF of the scheduling example, and is uniformly distributed along with the TPF, and is better than PF obtained by the other four algorithms. In summary, the improvements made to MOMVO are effective, and the obtained IMOMVO has better performance than other algorithms. IMOMVO has the good solving ability, and the obtained solution has higher quality and stability.

The solution (404, 9.770) in which the values of target makespan and target Tlb in the solution set PF of IMOMVO are both in the middle is selected as the display, and the corresponding scheduling Gantt chart is shown in Figure 3.
In Figure 3, the abscissa represents the time, the ordinate represents the machine number of each processing stage, and the number in the rectangle is the number of the workpiece. It can be seen from Fig. 3 that machine 3 is in a shutdown state during the period from 30 minutes to 50 minutes after the start of processing. During the period, although the workpieces 7, 20, 3, 13, and 5 have completed the processing of stage 1 before the 30th minute, due to the overlap between the processing period of them in the processing stage 2 and the shutdown period of the machine 3, and there is only one machine in the processing stage 2, the workpieces 7, 20, 3, 13, and 5 cannot enter stage 2 to be processed during the 30th to 50th minutes, they can only wait in the buffer zone. After the 50th minute, machine 3 stops shutting down and resumes normal operation. At this time, according to the code, workpiece 3 starts to process stage 2 on machine 3 first. Similarly, machine 6 is in a shutdown state during the 80th min to 100th min after the start of processing. At this time, although workpiece 3 has completed processing stage 3 before the 80th minute, its processing period of processing stage 4 overlaps with the shutdown period of machine 6, so workpiece 3 cannot select machine 6 for processing in stage 4. In processing stage 4, machine 8 is selected for its processing. Finally, when all the workpieces are processed, the time is 404 minutes.

Analysis of the Gantt chart in Figure 3 shows that when there is a machine shutdown, for the processing of each workpiece in each processing stage, on the one hand, the machine selected by the workpiece is constrained by the machine shutdown. When a machine has a shutdown, the processing task on the machine needs to reselect the non-shutdown machine for processing to make the scheduling target as small as possible; on the other hand, the start time and end time of processing are also affected by the machine shut down, and there is waiting time caused by machine shutdown. When there is no machine shutdown, these phenomena will not appear in the scheduling scheme. From the perspective of practical significance, in actual manufacturing, machine shutdown is a common and normal phenomenon. If machine shutdown is not taken into consideration, then a scheduling plan is generated, and during processing according to the scheduling plan, once the machine shuts down, on the one hand, a certain workpiece during processing may be scrapped due to the sudden shutdown of the machine, causing economic losses and potential safety hazards. On the other hand, unfinished processing tasks need to be rescheduled due to machine shutdown, resulting in production interruptions and more cost of time and logistics. While the scheduling plan is generated under the condition of machine shut down in advance, and the actual processing process is consistent with the scheduling plan, which can effectively avoid the adverse effects caused by the machine shutdown described above. It can be seen that the

Figure 3 The scheduling Gantt chart for the solution (404, 9.770) of the scheduling example
consideration of machine shutdown during scheduling has great practical significance for guiding the production of enterprises.

4.2. The algorithm's solution on the standard test functions

To further verify the effectiveness of IMOMVO, the commonly used multi-objective standard test functions are selected to further test the algorithm. For a multi-objective test function, different dimensions will lead to different difficulty and accuracy of the solution. The higher the dimension, the greater the difficulty of the solution and the worse the accuracy of the solution. In some literature, different dimensions of the test function are selected for testing. This article selects the commonly used ZDT1, ZDT2, ZDT3, ZDT6 multi-objective test functions, where the search domain of ZDT1-ZDT3 has 30 dimensions, and the search domain of ZDT6 has 10 dimensions.

For each test function, the five algorithms are run independently 20 times to obtain the average Ave and standard deviation Std of GD, and the average Ave and standard deviation Std of GD. As shown in Table 2.

| function | MODA | NSGA2 | MOEAD | MOMVO | IMOMVO |
|----------|------|-------|-------|-------|--------|
| ZDT1 Ave | 0.30855 | 0.04503 | 0.12470 | 0.02975 | 0.08655 |
| Std      | 0.14234 | 0.00584 | 0.03352 | 0.00795 | 0.01302 |
| ZDT2 Ave | 0.47238 | 0.07952 | 0.23798 | 0.06464 | 0.07049 |
| Std      | 0.16454 | 0.00970 | 0.09586 | 0.01636 | 0.01509 |
| ZDT3 Ave | 0.14092 | 0.02474 | 0.06514 | 0.01291 | 0.01912 |
| Std      | 0.04212 | 0.00239 | 0.01550 | 0.00432 | 0.01454 |
| ZDT6 Ave | 0.76728 | 0.06397 | 0.45747 | 0.10384 | 0.62759 |
| Std      | 1.02153 | 0.07647 | 0.18506 | 0.02435 | 0.08105 |

![Figure 4: The PF diagram of ZDT1](image)

![Figure 5: The PF diagram of ZDT2](image)

![Figure 6: The PF diagram of ZDT3](image)

![Figure 7: The PF diagram of ZDT6](image)
It can be analyzed from Table 5 that when the 4 test functions are solved, from the Ave of GD, the convergence performance of IMOMVO on three functions is the best among the 5 algorithms. From the Ave of IGD, the comprehensive performance of IMOMVO on all functions is the best among the five algorithms. Besides, from the standard deviation Std of GD and IGD, among the five algorithms, IMOMVO has a smaller Std value for solving most functions, which means that the solutions are more stable. From the analysis of Figure 4-7, it can be found that the PF obtained by IMOMVO is very close to the TPF of the test function, and is uniformly distributed along with the TPF, and is better than the PF obtained by the other four algorithms. In summary, the improvements made to MOMVO are effective, and the obtained IMOMVO has better performance than other algorithms. IMOMVO has the good solving ability, and the obtained solution has higher quality and stability.

5. Conclusion
Based on the consideration of machine shutdown constraints, this paper establishes an HFSP model with dual goals of maximum completion time and machine load balance and improves the MOMVO algorithm to solve the model. The good performance of the IMOMVO algorithm on scheduling examples and benchmark functions shows that the improvement of the algorithm is feasible and superior. Through the simulation solution of the scheduling example based on production data, the effectiveness of the scheduling model considering machine constraints is verified. The scheduling scheme obtained by this model is more in line with actual production requirements and has great reference value and significance for guiding the actual production scheduling of HFSP.

Funding
The presented work was supported by the National Science and Technology Innovation 2030 of China Next-Generation Artificial Intelligence Major Project, (no. 2018AAA0101804).

References
[1] Golneshini, F.P., Fazlollahtabar, H. (2019) Meta-heuristic algorithms for a clustering-based fuzzy bi-criteria hybrid flow shop scheduling problem. Soft Comput, 23: 12103–12122.
[2] Ribas, I., Leisten, R., Framiñan, J.M. (2010) Review and classification of hybrid flow shop scheduling problems from a production system and a solutions procedure perspective. Computers & Operations Research, 37(8):1439-1454.
[3] Ruiz, R., Vázquez Rodríguez, J.A. (2009) The hybrid flow shop scheduling problem. European Journal of Operational Research, 205(1):1-18.
[4] Zhang, C.J., Tan, J.W., Peng, K.K., et al. (2021) A discrete whale swarm algorithm for hybrid flow-shop scheduling problem with limited buffers. Robotics and Computer-Integrated Manufacturing, 68:102081.
[5] Sun, Y.J., Qi, X., Zhen, L. (2020) A DE-LS Metaheuristic Algorithm for Hybrid Flow-Shop Scheduling Problem considering Multiple Requirements of Customers. Scientific Programming, 2020:1-14.
[6] Ebrahimi, M., Ghomi, S.M.T.F., Karimi, B. (2014) Hybrid flow shop scheduling with sequence dependent family setup time and uncertain due dates. Applied Mathematical Modelling, 38(9-10):2490-2504.
[7] Han, W., Guo, F., Su, X.C. (2019) A Reinforcement Learning Method for a Hybrid Flow-Shop Scheduling Problem. Algorithms, 12(11):222.
[8] Zhao, Z., Wang, A.M., Ge, Y., et al. (2019) An Improved Genetic Algorithms for Multi-objective Hybrid Flow-shop Scheduling Problem. In: Proceedings of the 3rd International Conference on Power, Energy and Mechanical Engineering. Prague. pp. 4.
[9] Mirjalili, S., Mirjalili, S.M., Hatamlou, A. (2016) Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications, 27(2):495-513.
[10] Gao, W.F., Liu, S.Y. (2012) A modified artificial bee colony algorithm. Computers & Operations Research, 39(3):687-697.