

Search Manager: A Framework for Hybridizing Different Search Strategies

Yousef Abdi

Department of Computer Engineering
Payame Noor University, Tabriz, Iran
<http://orcid.org/0000-0002-8517-8769>

Yousef Seyfari

Department of Computer Science
University of Tabriz, Tabriz, Iran
<https://orcid.org/0000-0002-2393-8814>

Abstract—In the last decade, many of the metaheuristic search methods have been proposed for solving tough optimization problems. Each of these algorithms uses its own learn-by-example mechanism in terms of “movement strategy” to evolve the candidate solutions. In this paper, a framework, called Search Manager, is proposed for hybridizing different learn-by-example methods in one algorithm, which is inspired by the organizational management system in which managers change their management method by viewing performance reduction in their managerial organization. The proposed framework is verified using standard benchmark functions and real-world optimization problems. Further, it is compared with some well-known heuristic search methods. The obtained results indicate not only the optimization capability of the proposed framework, but also its ability to obtain accurate solutions and to achieve higher convergence precision.

Keywords—Global optimization; metaheuristic; organization management; hybridizing search methods

I. INTRODUCTION

In different areas of science such as industry, engineering, and management there are many complex problems, also known as optimization problems, such that there is no exact algorithm to solve them in polynomial time. Due to this fact, to find a relatively optimal solution for these kinds of problems, in the past decades, a significant number of different optimization algorithms have been introduced by researchers.

Optimization algorithms can be divided into two broad groups, deterministic and stochastic. Deterministic algorithms such as the Nelder–Mead search method [1], the tunnelling method [2], and renormalization group methods [3] perform based on gradients and second-order derivatives. A remarkable advantage of deterministic optimization methods is the fast convergence; however, for high dimensional and multimodal functions they may fall into a local optimum. In stochastic algorithms, although the quality of the obtained solution cannot be guaranteed, they are more efficient and flexible than deterministic approaches. Other advantages are the capability of escaping from a local optimum, good performance, and ease of implementation.

Stochastic optimization algorithms generally are population-based algorithms that begin with a set of randomly generated candidate solutions and by applying some specified rules iteratively, gradually evolve initial solutions. The rules are usually inspired by the behaviors of biological and physical

systems in nature, culture, society or politics. Based on the source of inspiration, they can be classified into three main groups: (1) Evolution-based, (2) Swarm-based, and (3) Human-based algorithms.

Evolution-based algorithms mimic natural biological evolution and selection. The Genetic Algorithm (GA) has been the most popular in this class of optimization algorithms. It uses the Darwinian theory of natural selection, crossover, and mutation [4]. Other algorithms that can be classified in this group are Evolution Strategy (ES) [5], Genetic Programming (GP) [6], Differential Evolution (DE) [7], Evolutionary Programming (EP) [8], and Biogeography-Based Optimizer (BBO) [9].

Swarm-based algorithms get their inspiration from the collaborative conduct of a group of animals, such as ant colonies, honey bees, and bird flocks. They are typically made up of a population of agents (swarm individuals) interacting together to fulfill the main goal of the system. To date, several swarm based optimization algorithms have been proposed in the literature. Particle Swarm Optimization (PSO) is the most popular and significant algorithm that mimics the behavior of a flock of migrating birds heading for an unknown destination [10]. So many variants of PSO algorithm such as MLPSO [11], FST-PSO [12], etc. with the aim of improving its performance in different types of problems have been presented so far. Other examples of this class of algorithms are: Ant Colony Optimization [13], Artificial Bee Colony [14], Bacterial Foraging [15], Cat Swarm Optimization [16], Elephant Herding Optimization (EHO) [17], Bat Algorithm (BA) [18], etc.

Human-based algorithms imitate human social behaviors. For example, Fireworks Algorithm (FA) inspired by observing fireworks explosion [19], Harmony Search algorithm (HS) inspired by harmony improvisation process of musicians [20]. Teaching Learning Based Algorithm (TLBO) imitates the interaction between a teacher and her students [21]. Some of the other popular algorithms in this group are Cultural Algorithm (CA) [22], Imperialist Competitive Algorithm (ICA) [23], Exchange Market Algorithm (EMA) [24], Soccer League Competition (SLC) [25], [26], and Brain Storm Optimization (BSO) [27], World Competitive Contests (WCC) [28].

There are some extra optimization algorithms that cannot be assigned to the above three groups. They imitate some additional rules that exist in the world and universe. For

instance, Optics Inspired Optimization (OIO) uses the law of reflection [29] or Ray Optimization (RO) was developed based on Snell's light refraction law [30]. Some of the other algorithms are: Water Wave Optimization (WWO) [31], Big-Bang Big-Crunch (BBBC) [32], Charged System Search (CSS) [33], Artificial Chemical Reaction Optimization Algorithm (ACROA) [34], Curved Space Optimization (CSO) [35], Central Force Optimization (CFO) [36]. However, we think this kind of optimization algorithms is more complex for understanding than the specified three groups of algorithms.

Regardless of the inspiration's source, still no universal optimization algorithm has emerged; nor does it appear likely that one ever will [37] and the No Free Lunch Theorem of Optimization (NFLT) supports this view [38]. Besides, although many optimization methods have been proposed so far, one or another method presents a better solution on a specific problem. Therefore, finding more efficient algorithms is still in progress and we will see further optimization algorithms with the development of human identity from nature. However, a powerful optimization algorithm can solve many problems.

Considering the aforementioned facts, in this paper we intend to propose a framework, Search Manager, for combining different search abilities of different optimization algorithms in one algorithm to effectively solve the optimization tasks. The proposed framework accomplishes this by imitating the changing management style that is applied by managers in real-world organizations. In any organization when managers cannot obtain better results, they have to change their management style to improve the performance of the organization.

The simplicity and capability of using variable movement strategies are powerful aspects of the proposed framework. Although in this study we use four simple movement strategies, the others can be proposed or obtained from the other optimization algorithms to get better performance for different optimization problems.

The remainder of this paper is organized as follows: Section II gives a general view of the main behavior of the proposed algorithm. Section III reviews some of the main principles about organizations and their management process. In Section IV, the proposed framework is introduced. Section V presents experimental investigation of the proposed method and its comparative study with other metaheuristic algorithms. Finally, some concluding remarks and future works are presented in Section VI.

II. MOTIVATION

Almost all of the previous population-based optimization methods iteratively evolve and optimize a population of individuals (candidate solutions) according to some criteria to reach a final population which has a near-optimal solution to a problem. In fact, they use learn-by-example mechanisms in terms of 'movements' to evolve the candidate solutions, but the main problem of these methods is the lack of alternative movement strategies for different situations.

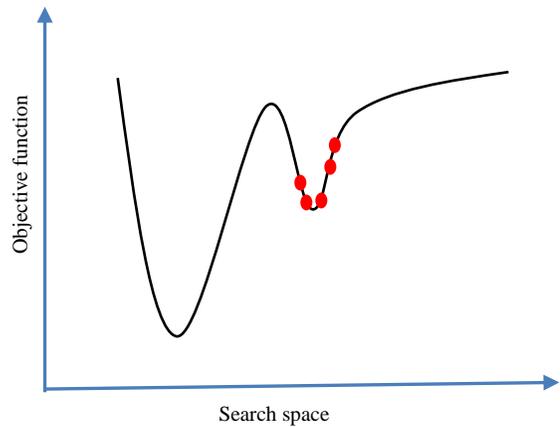


Fig. 1. Search space of an objective function.

The proposed framework overcomes the aforementioned limitation by combining different movement strategies. If the current strategy does not improve solutions within the population, another one will be selected. For example, consider the situation in Fig. 1 that represents the search space of an objective function, in which there exist a local and global minimum at inflection points. In this situation, the population of six individuals has got stuck in the local minimum. If there is not enough diversity in the population like this situation, an optimization algorithm would get trapped in the local optimums due to the lack of an alternative strategy. In the proposed framework, it is more likely that the population with not enough diversity could escape from local minimum by using different movement strategies.

III. ORGANIZATION MANAGEMENT BACKGROUND

Firm infrastructure, considered as organizations' structure and its management process, is a branch of social science, which includes a lot of relative concepts. However, in this section, we only review a few important concepts that are used in the proposed framework.

In the society around us, there are various organizations such as hospitals, schools, social institutions, etc. to achieve certain objectives in social life. In the words of Leavitt, an organization is "a particular pattern of structure, people, tasks and techniques" [39]. In the view of Katz and Kahn, an organization is "a system which is composed of a set of subsystems" [40], put the other way round, an organization is a unified system, whose functionality is divided into different key subsystems. These subsystems are all parts of an organization working together for a common purpose.

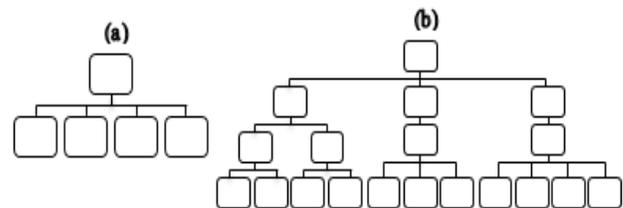


Fig. 2. (a) Flat management structure (b) Hierarchical management structure.

Each organization in the society needs a system of management. Management is defined in different ways by different people. It may be viewed as a process of leading and controlling the activities of organization employees [41], or as a decision-making process in which the manager is a decision-maker that leads the organization to achieve the objectives [42]. In all of the organizations, employees look up to managers for guidance, thus for making appropriate management decisions and guiding organization members, managers adopt a management style or a form of leadership, primarily by their abilities, personalities, and values [43]. However, half of the decisions made by managers within organizations fail [44]. Failure in decisions signals the need for new decisions or changing the management style that decisions were taken under it. Failed decisions can be a source of learning and they can help the manager to take effective and operational decisions in the future [45]. Management style is a form of leadership that specifies how employees should follow organization's policy. There are many types of management styles. Managers move in and out of these various styles as the need arises.

Management structure is another aspect of the management process. It is the manner in which the management of a company or organization is organized. It determines the scope and nature of how leadership is disseminated throughout the organization [42], [43]. Organizations commonly adapt either a flat or hierarchical structure. In the flat structure, there are a few or no levels of middle management between top managers and employees. But in the hierarchical structure, there are a number of hierarchical levels between top managers and employees. Fig. 2 shows these structures. The proposed framework uses the aforementioned concepts for combining different learn-by-example methods in one algorithm for solving optimization problems.

IV. PROPOSED METHOD

In this section, according to the basic principles of organization management outlined in the previous section, a framework is proposed for combining different search methods. The core of the Search Manager is based on a basic principle in the management of today's organizations: "managers need to adopt a new management style when organization's performance goal gets worse". The framework consists of three steps as follows:

- Step 1: Initialization
- Step 2: Movement
- Step 3: Repeat Step 2 until the stop criterion is satisfied.

Fig. 4 shows the flowchart of the Search Manager and details of its steps are represented in the following subsections.

A. Step 1: Initialization

The goal of the optimization task is to select n decision variables x_1, x_2, \dots, x_n (known as candidate solution) from a feasible region in such a way as to optimize a given objective function. The values of the decision variables are represented as floating point numbers and the cost of a candidate solution is obtained by evaluating the objective function f at these variables like the following equation.

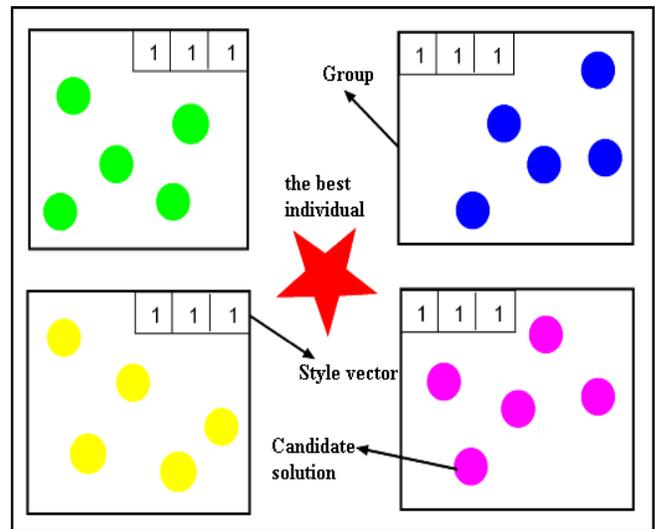


Fig. 3. A sample of population structure in Search Manager.

$$Cost = f(\text{Candidate solution}) = f(x_1, x_2, \dots, x_n)$$

To start the optimization algorithm, a population of size N_{pop} is generated, and then the best individual is selected. The remainder of solutions is divided among m distinct groups. For example, Fig. 3 shows four distinct groups of population, in which the best individual from the population is specified.

Each group will have current and previous average cost. The average cost of a group is defined as

$$GC_n = \text{mean}\{c_1, c_2, \dots, c_i\}$$

Where GC_n is the group cost of the n th group of candidate solutions and c_i is the cost of the i th solution. Furthermore, each group has an t -dimensional style vector, where t is the number of movement styles. The initial values of these vector components are set to 1 and each component holds the score value for the specified movement style. More explanations on this vector are presented in the following subsection.

B. Step 2: Movement

In the movement step, candidate solutions are evolved for the next generation using movement styles. The framework uses four simple movement styles inspired from other optimization algorithms and their mathematical formulas are represented in the following subsections.

As mentioned in subsection 4-A, each group of the candidate solutions has a vector that holds score values for different movement styles. For each group, a style (movement method) is selected by using the values of style vector and the roulette wheel selection method. The selection probability is calculated as (1).

$$P_i = \frac{\text{ScoreValue}_i}{\sum_{j=1}^t \text{ScoreValue}_j} \quad (1)$$

Where, P_i is the probability of being selected i th style from the t styles. It is obvious that the style with a higher score is more probable for selection than the others.

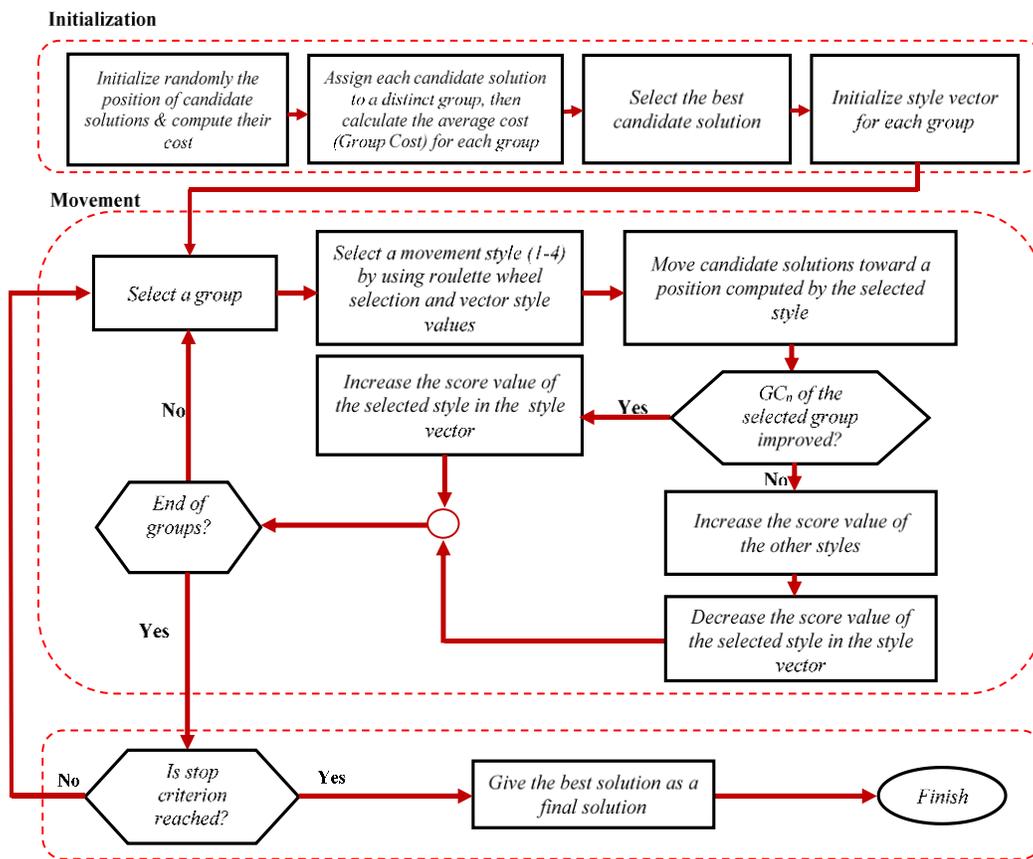


Fig. 4. The flowchart of the proposed framework.

At the beginning of the movement step, for each group one of the styles is selected randomly, then all candidate solutions in the group move toward a position calculated by using the selected movement style, and then finally, the group cost (GC_n) is computed and compared with the previous group cost, if an improvement is seen, the selected style is rewarded by increasing its score value in the style vector. Otherwise, the selected style is penalized by dividing its score value by a random number generated between 2 and the number of population. At the same time, the score values of the other styles are increased by a random number between 0 and 1. For example, Fig. 5 shows a sample of style vector for a group. *Style3* is more probable to be selected for the next iteration. If *Style3* is selected and the group cost is improved, the values of the vector will be changed like Fig. 6. It is observable that the score value of *Style3* has been increased.

Style 1	Style 2	Style 3
8	5	25

Fig. 5. A sample of style vector for holding scores of movement styles.

Style 1	Style 2	Style 3
8	5	25.3

Fig. 6. A sample of style vector after punishing and rewarding styles.

Style 1	Style 2	Style 3

8.5	5.5	2.5
-----	-----	-----

Fig. 7. A sample of style vector after punishing and rewarding styles.

Now, based on the values in Fig. 5, if *Style3* does not improve the group cost or make it worse, the values of the vector style will change like Fig. 7. It can be seen that *Style3* has been decreased and the others have been increased.

The used movement styles are inspired from other optimization algorithms and their equations are formulated as follows:

1) *The First Movement Style*

By the first movement style, solutions accept some of the variable values from the best individual. This style is applied by using (2).

$$\Phi = \text{randomly selected decision variables} \quad (2)$$

$$X_i(\Phi) = X_{\text{the best individual}}(\Phi)$$

Where, X_i is a selected solution from a group, and Φ is a set of randomly chosen decision variables.

2) *The Second Movement Style*

In the second movement style, solutions in the group are updated as

$$\begin{aligned} T &= X_{\text{the best individual}} \\ T &= T + \{\text{rand}\} \times \{T - X_r\} \\ X_i^{\text{new}} &= X_i^{\text{old}} + \{\text{rand}\} \times \{T - X_i^{\text{old}}\} \end{aligned} \quad (3)$$

Where, {rand} is an n-dimensional random vector and its values are in [0, 1]. X_r corresponds to the randomly selected solution from the group and X_i^{old} corresponds to the i th solution in the group.

3) The Third Movement Style

In the third movement style, solutions in the group are updated as

$$\begin{aligned} T &= X_{\text{the best individual}} \\ T &= T + \{\text{rand}\} \times (T - V) \\ X_i^{\text{new}} &= X_i^{\text{old}} + \{\text{rand}\} \times (T - X_i^{\text{old}}) \end{aligned} \quad (4)$$

Where, V is the variance of solutions in the group.

4) The Fourth Movement Style

The fourth movement style uses (5) for updating solutions in the group. Some of the selected variables of the best individual are changed in a temporary vector and then the other solutions in the group move towards this vector.

$$\begin{aligned} \Phi &= \text{randomly selected decision variables} \\ T &= X_{\text{the best individual}} \\ T(\Phi) &= \{\text{rand}\} \\ X_i^{\text{new}} &= X_i^{\text{old}} + \{\text{rand}\} \times (T - X_i^{\text{old}}) \end{aligned} \quad (5)$$

Where, {rand} is a generated random vector from the search space.

After applying movement operation, the best individual is selected from the population.

V. COMPARATIVE STUDY

In this section, the ability of the Search Manager is assessed from two perspectives: one is by applying it to optimization of 44 benchmark functions; another is through 15 real-world problems.

In the first perspective, Search Manager is applied on a wide range of nonlinear benchmark functions, 14 functions from the CEC 2005¹ [46] and all of the 30 functions from the CEC 2014 [47]. These benchmark suites include a diverse set of problem features such as unimodality, multimodality, separability, non-separability, rotation, scalability, etc. for single objective optimization. They are based on classical benchmark functions, such as Rosenbrock's, Rastrigin's, Swefel's, Griewank's, and Ackley's function. It should be noted although these test functions are organized by the community of heuristic algorithms in the framework of a workshop, in this study they are used as standard test functions for comparison. The total functions in CEC 2005 and CEC 2014 can be divided into some groups according to their

characteristics. Table I shows these groups. The other properties of these functions have been defined properly in their corresponding references, and hence they are not repeated here. The experimental results on these benchmark functions may reveal how Search Manager performs on various functions as well as can be compared to those obtained by other algorithms.

In the second perspective, Search Manager is applied to 15 real-world problems, which are derived from CEC 2011 [48].

To evaluate the performance of the proposed method, obtained results using Search Manager is compared with the result of some well-known and recently proposed social and nature-inspired optimization algorithms from the computational viewpoint. The selected algorithms are listed below.

- **CA:** Cultural Algorithm models social evolution and learning in agent-based societies [22].
- **ABC:** Artificial Bee Colony inspired by foraging behavior of honey bee swarm [49].
- **GSA:** Gravitational Search Algorithm uses the law of gravity and the notion of mass interactions based on the laws of gravity and motion [50].
- **FOA:** Forest Optimization Algorithm inspired by few trees in the forests which can survive for several decades, while other trees could live for a limited period [51].
- **WOA:** Whale Optimization Algorithm mimics the social behavior of humpback whales. The algorithm is inspired by the bubble-net hunting [52].
- **DE:** Differential Evolution is a type of standard genetic algorithm [7].

Computation code of all above-mentioned algorithms is taken from web pages dedicated to these algorithms. The MATLAB code for all algorithms is available at: <https://data.mendeley.com/datasets/f5jvxbw8xb/1> (DOI: 10.17632/f5jvxbw8xb.1)

A. Experimental Settings

The experimental environment is a computer of Intel Core i3, 4GB DDR2 memory, and Windows 7 operating system. Tests are performed in 10-, 30-, and 50-dimensional for each test function in the first perspective. All algorithms are executed 30 times for each problem with a total of $D \times 10^4$ and 5×10^4 evaluations of the objective function (Max_FES) in the first and second perspective respectively, where D is the dimension of the problem, and the obtained average results are compared with the other algorithms. The population size $P = 50$ is used for all algorithms. Table II shows the other recommended parameter settings for each algorithm. The values of these parameters are selected based on the recommendation from their original papers or previous related works. Additionally, initial candidate solutions are randomly calculated by uniform distribution between lower and upper limits of benchmark functions.

¹ The first 14 functions are selected from CEC 2005 benchmark set.

TABLE I. FUNCTION TYPES

Test Suite	Unimodal	Simple Multimodal	Hybrid (Multimodal)	Composition (Multimodal)	Expanded (Multimodal)	Hybrid Composition (Multimodal)
CEC 2005	$f_1 - f_5$	$f_6 - f_{12}$	-	-	$f_{13} - f_{14}$	Not used in this study
CEC 2014	$f_1 - f_3$	$f_4 - f_{16}$	$f_{17} - f_{22}$	$f_{23} - f_{30}$	-	-

TABLE II. OPTIMIZATION METHODS PARAMETERS

Optimization Method	Parameters
ABC	Limit = 50D [49]
GSA	$G_0=100, \alpha=20, K_0$ =population size [50]
FOA	Life time = 6, LSC = 2, Area limit = 30, Transfer rate = 10%, GSC = 3 [51]
WOA	$a=2 - 0$ [52]
CA	Acceptance rate = 0.3 [53] Knowledge type = Normative & Situational
DE	$F = 0.9, CR = 0.1$ [54]
Search Manager	Without the need for setting any control parameters

It should be noted that we know better results can be achieved from these algorithms by fine-tuning their control parameters. However, finding the perfect parameter for each problem is expected to be a very time-consuming task. Therefore, fixed parameter settings are adopted for each algorithm and this condition is equal for all algorithms.

B. Results and Discussion on Perspective 1

Experimental results of all algorithms on 14 functions of CEC 2005 and 30 functions of CEC 2014 are presented in Tables III to V and Tables VI to VIII, respectively. In all tables, the mean and standard deviation of 30 runs of the algorithms for each function are adopted to assess the optimization performance of the proposed algorithm. Additionally, a statistical test called Wilcoxon rank-sum test [55], which is a nonparametric statistic test for the independent samples, is conducted on the experimental results at the 5% significance level to judge whether the obtained results from the proposed method are significantly different from the other algorithms and have not occurred by chance [56]. The cases are marked with “+ / \approx / -” when the performance of Search Manager is significantly better than, equal to, and worse than the other test algorithms. For clarity, the best results are marked in bold. Tables IX and X show obtained p -values in the statistical test between the proposed algorithm and each of the remaining algorithms over benchmark functions in 10 and 50 dimensions. Moreover, Fig. 8 and 9 represent the convergence rate of some selected unimodal and multimodal functions for all of the algorithms.

Based on the presented results of CEC 2005 test suite in Tables III to V, Search Manager provided better results over 10 dimensions. Although its performance decreases in 30 and 50 dimensions, it can be seen that its rank is better than the other algorithms overall dimensions. In addition, the proposed algorithm displayed the best performance over expanded functions in all dimensions.

For CEC 2014 test suite, the experimental results in Tables VI to VIII show that the Search Manager presented good results in terms of average rank, but the DE algorithm outperforms Search Manager in most of the functions over 10 and 30 dimensions. Furthermore, Search Manager could not

produce successful results on composition functions as well as unimodal functions as compared to the ABC and DE algorithms. In addition, the GSA showed the best results in 30 and 50 dimensions of hybrid functions. However, Search Manager may get better results by using other types of movement styles.

Based on the convergence curves (Fig. 8 and 9), it can be seen that Search Manager has a good convergence rate and it can reach an optimum solution with less calculation.

To summarize, although Search Manager outperforms most of the compared algorithms in terms of average rank in all benchmark function sets, its performance is lower than the DE and ABC in some complex functions. However, the powerful aspect of the Search Manager is its ability to accept different movement strategies, and hence it may achieve better results for these functions by using other combinations of learn-by-example methods as movement styles. In other words, instead of adjusting parameters to get better results that we have seen in previous optimization algorithms, we can adjust the combination of different learn-by-example methods in Search Manager.

C. Results and Discussion on Perspective 2

Herein, the performance of Search Manager is evaluated through 15 real-world optimization problems. Detailed definitions of these problems can be found in [48].

Considering the experimental results represented in Table XI, Search Manager outperforms all of the compared algorithms in most of the real-world problems. In more details, Search Manager outperforms ABC, FOA, GSA, WOA, CA, and DE on 9, 11, 11, 9, 10, 9, 7 problems respectively. On the contrary, ABC, FOA, GSA, WOA, CA, and DE are better than Search Manager on 2, 2, 2, 2, 2, 1, and 1 problems, respectively. All the compared algorithms obtained similar results in solving the T03BCB and T08TNEP. Therefore, based on the obtained results, it is observable that Search Manager performs better in most of the real-world problems when compared with ABC, FOA, GSA, WOA, CA, and DE.

The conclusion that can be drawn from experiments on real-world problems is that the combination of the used learn-by-example equations as movement styles in Section IV is most suitable for this kind of optimization problems.

VI. CONCLUSION AND FUTURE WORKS

In this paper, we have proposed a framework, called Search Manager that imitates an important part of the management process, which is concerned by managers in any organization. The heart of the Search Manager is applying another movement strategy when the average cost of candidate solutions gets worse. Search Manager has one main operator: Movement. The movement operator implements different learn-by-examples methods as movement styles. Extensive analysis is carried out to reveal the ability of Search Manager

to find the best fitness value over the search space not only in the benchmark functions, but also in the real optimization problems as well. Of course, it should be noted that like the other optimization algorithms, Search Manager is suitable for some sort of problems. However, its important property is that it provides a framework for combining different search methods, so it can be made compatible with any optimization problem by modifying its movement styles.

In the basic Search Manager, we have used four movement strategies as movement styles, but more or other types of movement methods can be formulated to improve its performance.

Search Manager can use the learn-by-example strategies of other optimization algorithms. Therefore, it is capable of combining these strategies in one algorithm and in utilizing their search abilities. From this capability, a possible line for future work would be investigating the behavior of the Search Manager with the combination of different learn-by-example strategies.

Other possible avenues of future research on the Search Manager include: using other mechanisms for rating movement styles because the performance of the proposed framework is highly dependent on this mechanism, applying a mechanism for moving solutions between groups, and finally, management science and its application in organizations have a lot of concepts in the real-world, such as how to fire and hire employees, employee promotion mechanisms, etc. that can be brought into the framework via simulation.

REFERENCES

- [1] A. Nelder, R. Mead, "A simplex method for function minimization," *Comput J.*, vol. 7, no. 4, pp. 308-31, 1965.
- [2] A. V. Levy, A. Montalvo, "The tunneling algorithm for the global minimization of functions," *Siam J Sci Stat Comp*, vol. 6, no. 1, pp. 15-22, 1985.
- [3] D. Shalloway, "Application of renormalization group to deterministic global minimization of molecular conformation energy functions," *J Glob Optim*, vol. 2, pp. 281-311, 1992.
- [4] J. H. Holland, "Genetic algorithms," *Sci Am*, Vol. 267, pp. 66-72, 1992.
- [5] I. Rechenberg, "Evolution strategies," in: B. Schneider, U. Ranft (editors), *Simulationenmethoden in der Medizin und Biologie*, Berlin: Springer Heidelberg, pp. 83-114, 1978.
- [6] J. R. Koza, "Genetic programming: on the programming of computers by means of natural selection," Cambridge, MIT Press, 1992. ISBN: 978-0-262-11170-6.
- [7] R. Storn, K. Price, "Differential Evolution – A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob Optim*, vol. 11, no. 4, pp. 341-59, 1997.
- [8] G. B. Fogel, D. B. Fogel, "Continuous evolutionary programming: Analysis and experiments," *Cybernet and Sys*, vol. 26, pp. 79-90, 1995.
- [9] D. Simon, "Biogeography-based optimization," *IEEE T Evolut Comput*, vol. 12, pp. 702-713, 2008.
- [10] J. Kennedy, R. C. Eberhart, "Particle swarm optimization," in: *IEEE International Conference on Neural Networks*, Perth, Australia, pp.1942-48, 1995.
- [11] P. Liu, J. Liu, "Multi-leader PSO (MLPSO): A new PSO variant for solving global optimization problems," *Appl Soft Comput*, vol. 61, pp. 256-263, 2017.
- [12] M. S. Nobile, P. Cazzaniga, D. Besozzi, R. Colombo, G. Mauri, G. Pasi, "Fuzzy Self-Tuning PSO: A settings-free algorithm for global optimization," *Swarm Evol Comput*, Inpress, 2017.
- [13] M. Dorigo, T. Stützle, "Ant colony optimization. Cambridge," MIT Press, 2004. ISBN: 978-0-262-04219-2.
- [14] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Erciyes University, Engineering Faculty, Computer Engineering Department, Technical Report-TR06, 2005.
- [15] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Contr Sys Mag*, vol. 22, pp. 52-67, 2002.
- [16] S. C. Chu, P. W. Tsai, J. S. Pan, "Cat swarm optimization," in: *9th Pacific Rim International Conference on Artificial Intelligence*, Guilin, China, pp.854-58, 2006.
- [17] G. Wang, S. Deb, L. Coelho, "Elephant herding optimization," in: *3rd International Symposium on Computational and Business Intelligence*, Bali, Indonesia, 2015.
- [18] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in: J. Gonzalez, D. Pelta, C. Cruz, G. Terrazas, N. Krasnogor (editors), *Nature Inspired Cooperative Strategies for Optimization*, vol. 284 of *Studies in Computational Intelligence*, Berlin, Springer Heidelberg, 2010.
- [19] Y. Tan, Y. Zhu, "Fireworks algorithm for optimization," in: Y. Tan, Y. Shi, K. C. Tan (editors), *Advances in swarm intelligence*, Berlin, Springer Heidelberg, pp. 355-64, 2010.
- [20] Z. W. Geem, J. H. Kim, G. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, pp. 60-8, 2001.
- [21] R. V. Rao, V. J. Savsani, D. P. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *Comput Aided Design*, vol. 43, no. 3, pp. 303-5, 2015.
- [22] R. Reynold, "An introduction to cultural algorithms," in: *3rd Annual Conference on Evolutionary Programming*, World Scientific Publishing, pp.131-9, 1994.
- [23] E. Atashpaz-Gargari, C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition," in: *the 2007 IEEE congress on evolutionary computation (CEC)*, pp.4661-7, 2007.
- [24] N. Ghorbani, E. Babaei, "Exchange market algorithm," *App Soft Comp*, vol. 19, pp. 177-87, 2014.
- [25] N. Moosavian, B. K. Roodsari, "Soccer league competition algorithm: a new method for solving systems of nonlinear equations," *Int J Intell Sci*, vol. 4, no. 1, pp. 7-16, 2013.
- [26] N. Moosavian N, B. K. Roodsari, "Soccer league competition algorithm: a novel meta-heuristic algorithm for optimal design of water distribution networks," *Swarm Evol Comp*, vol. 17, pp. 14-24, 2014.
- [27] Y. Shi, "Brain Storm Optimization Algorithm," *Adv Swarm Intell*, vol. 6728, pp. 303-309, 2011.
- [28] Y. Masoudi-Sobhanzadeh, H. Motieghader, "World Competitive Contests (WCC) algorithm: A novel intelligent optimization algorithm for biological and non-biological problems," *Informatics in Medicine Unlocked*, vol. 3, pp. 15-28, 2016.
- [29] A. HusseinzadehKashan, "A new metaheuristic for optimization: Optics inspired optimization (OIO)," *Comput Oper Res*, vol. 55, pp. 99-125, 2015.
- [30] A. Kaveh, M. Khayatizad, "A new meta-heuristic method: ray optimization," *Comput Struct*, vol. 112, pp. 283-94, 2012.
- [31] Y. J. Zheng, "Water wave optimization: A new nature-inspired metaheuristic," *Comput Oper Res*, vol. 55, pp. 1-11, 2015.
- [32] O. K. Erol, I. Eksin, "A new optimization method: big bang-big crunch," *Adv Eng Softw*, vol. 37, no. 2, pp. 106-111, 2006.
- [33] A. Kaveh, S. Talatahari, "A novel heuristic optimization method: charged system search," *Acta Mech*, vol. 213, no. 3, pp. 267-89, 2010.
- [34] B. Alatas, "ACROA: Artificial chemical reaction optimization algorithm for global optimization," *Expert Sys Appl*, vol. 38, no. 10, pp. 13170-80, 2011.
- [35] F. F. Moghaddam, R. F. Moghaddam, M. Cheriet, "Curved space optimization: A random search based on general relativity theory," *arXiv preprint*, arXiv:1208.2214, 2012.

[36] R. A. Formato, "Central force optimization: A new metaheuristic with applications in applied electromagnetics," Prog Electrom Res, vol. 77, pp. 425–91, 2007.

[37] M. Presuss M, "Multimodal Optimization by Means of Evolutionary Algorithms," 1st ed, Switzerland, Springer, 2015. doi:10.1007/978-3-319-07407-8_1

[38] Y. C. Ho, D. L. Pepyne, "Simple Explanation of the No-Free-Lunch Theorem and Its Implications," J. Optimiz Theor Appl, vol. 115, no. 3, pp. 549-70, 2002.

[39] H. J. Leavitt, "Applied organization and readings. Changes in industry: structural, technical and human approach," in: Cooper, WW, et al. New Perspectives in Organization Research, New York, Wiley, 1962.

[40] D. Katz, R. L. Kahn, "The social psychology of organizations," New York, Wiley, 1978. ISBN: 978-0-471-02355-5.

[41] T. Manichander, "Educational management," USA, Lulu Publication, 2016. ISBN: 978-1-329-76327-2.

[42] R. B. Rudani, "Principles of management.," New Delhi, McGraw-Hill Education, 2013. ISBN: 978-1-259-02696-6.

[43] R. Benfari, "Understanding your management style: Beyond the Myers-Briggs Type Indicator," New York, Lexington Books, 1991. ISBN: 978-0-669-24814-2.

[44] M. Benefiel, "Using discernment to make better business decisions," in: G. Flynn (editor), Leadership and Business Ethics, Netherlands, Springer, pp. 31-38, 2008.

[45] H. Gospel, A. Pendleton, "Corporate governance and labour management.," New York, Oxford university press, 2005. doi:10.1093/acprof:oso/9780199263677.001.0001

[46] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," Computational Intelligence Laboratory, Zhengzhou University, China and Nanyang Technological University, Singapore, Technical Report, 2005. http://www.ntu.edu.sg/home/EPNSugan

[47] J. J. Liang, B. Y. Qu, P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization" Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technical Report, Nanyang Technological University, Singapore, 2013.

[48] S. Da, P. Suganthan, "Problem definitions and evaluation criteria for CEC2011 competition on testing evolutionary algorithms on real world optimization problems" Jadavpur University and Nanyang Technological University, Technical report, 2011.

[49] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications", Artif Intell Rev, vol. 42, no. 1, pp. 21-57, 2014.

[50] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, "GSA: a gravitational search algorithm," Inf Sci, vol. 179, pp. 2232-48, 2009.

[51] M. Ghaemi, M. R. Feizi-Derakhshi, "Forest Optimization Algorithm," Expert Sys Appl, vol. 41, no. 15, pp. :6676-6687, 2014.

[52] S. A. Mirjalili, A. Lewis, "The Whale Optimization Algorithm," Adv Eng Softw, 95, pp. 51-67, 2016.

[53] M. Z. Ali, N. H. Award, P. N. Suganthan, R. G. Reynolds, "A modified cultural algorithm with a balanced performance for the differential evolution frameworks," Knowl-Based Syst, vol. 111, pp. 73-86, 2016.

[54] M. El-Abd, "Performance assessment of foraging algorithms vs Evolutionary algorithms," Inf Sci, vol. 182, pp. 243-63, 2012.

[55] F. Wilcoxon, "Individual comparisons by ranking methods," Biometrics, vol. 1, pp. 80-83, 1945.

[56] A. K. Qin, P. N. Suganthan, "Self-adaptive differential evolution algorithm for numerical optimization," in: IEEE Congress on Evolutionary Computation, Edinburgh, Scotland, pp. 1785-1791, 2005.

TABLE III. MINIMIZATION RESULTS FOR 14 FUNCTIONS FROM CEC 2005 OVER 30 RUNS AT 10 DIMENSIONS

CEC05	Search method						
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank
f_1	1.44E-19(6.42E-19) 2	1.43E-12(1.50E-12) +3	2.32E-05(8.99E-06) +4	3.07E+00(7.12E-01) +7	2.92E-02(2.32E-03) +5	6.43E-02(2.67E-01) +6	1.64E-24(1.41E-24) -1
f_2	1.59E-05(2.07E-05) 1	2.57E+00(1.03E+00) +2	1.29E-02(5.28E-02) +2	1.13E+01(2.49E+00) +3	4.80E+03(2.18E+03) +7	2.88E+02(3.34E+02) +5	5.05E+01(2.29E+01) +4
f_3	1.53E+05(9.24E+04) 1	3.11E+05(9.44E+04) +2	1.59E+05(1.04E+05) ≈1	1.67E+05(8.77E+04) ≈1	1.19E+06(1.11E+06) +3	1.72E+06(1.45E+06) ≈1	2.50E+06(1.03E+06) +4
f_4	2.69E-01(4.17E-01) 1	5.03E+01(1.91E+01) +3	6.97E-01(4.23E-01) +2	1.37E+01(4.14E+00) +2	1.18E+04(5.93E+03) +6	8.86E+02(8.87E+02) +5	2.63E+02(8.45E+01) +4
f_5	3.90E+01(4.07E+01) 4	1.36E-01(7.43E-01) -1	1.36E+00(3.50E+00) -3	1.52E+03(1.35E+02) +5	1.72E+03(1.86E+03) +5	1.58E+03(9.25E+02) +5	6.14E-01(8.52E-01) -2
f_6	3.91E+01(1.62E+02) 3	7.38E+00(1.58E+00) -1	2.31E+03(2.98E+03) +4	4.62E+02(2.02E+02) +4	6.90E+03(1.44E+04) +4	4.55E+03(8.41E+03) +4	7.21E+00(5.07E+00) -2
f_7	2.22E+00(1.19E+00) 3	1.52E-01(2.87E-01) -1	6.21E-01(3.71E-01) -2	5.00E-01(8.28E-02) -2	5.92E+00(6.13E+00) +4	2.58E+01(2.28E+01) +5	5.38E-01(1.31E-01) -2
f_8	2.01E+01(7.47E-02) 1	2.04E+01(6.62E-02) +5	2.02E+01(4.50E-02) +2	2.04E+01(6.39E-02) +5	2.02E+01(8.92E-02) +4	2.02E+01(1.24E-01) +3	2.04E+01(7.43E-02) +5
f_9	6.63E-02(2.52E-01) 2	2.52E+01(4.30E+00) +4	4.74E+01(1.52E+01) +6	5.51E+01(7.52E+00) +6	3.28E+01(1.26E+01) +5	2.22E+01(1.08E+01) +3	0.00E+00(0.00E+00) -1
f_{10}	3.08E+01(1.18E+01) 2	3.47E+01(4.35E+00) ≈2	7.86E+01(1.86E+01) +4	7.15E+01(9.07E+00) +4	5.59E+01(2.24E+01) +3	2.46E+01(1.28E+01) -1	2.43E+01(4.39E+00) -1
f_{11}	6.66E+00(1.48E+00) 3	7.16E+00(5.39E-01) ≈3	8.10E+00(1.54E+00) +4	9.10E+00(6.67E-01) +5	8.09E+00(1.22E+00) +4	5.34E+00(1.47E+00) -2	6.64E+00(6.77E-01) ≈3
f_{12}	3.46E+02(1.31E+02) 1	1.79E+03(9.87E+02) +2	6.37E+03(8.86E+03) +3	2.75E+04(5.72E+03) +4	5.03E+03(5.42E+03) +3	3.66E+03(4.77E+03) +3	3.75E+02(1.12E+02) ≈1
f_{13}	5.72E-01(1.72E-01) 1	2.14E+00(2.74E-01) +3	5.01E+00(1.51E+00) +5	8.84E+00(1.02E+00) +6	2.79E+00(1.31E+00) +4	1.51E+00(7.43E-01) +2	5.95E-01(9.99E-02) ≈1

f_{14}	3.53E+00(3.30E-01) 1	3.50E+00(1.34E-01) ≈1	3.84E+00(4.04E-01) +/2	3.60E+00(2.72E-01) ≈1	3.68E+00(3.31E-01) +/2	3.60E+00(4.45E+01) ≈1	3.69E+00(1.21E-01) +/2
+/-	8/3/3	11/1/2	11/2/1	14/0/0	10/2/2	5/3/6	
Avg-rank	1.86	2.36	3.14	3.93	4.21	3.28	2.36

"+", "≈", "-" respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE IV. MINIMIZATION RESULTS FOR 14 FUNCTIONS FROM CEC 2005 OVER 30 RUNS AT 30 DIMENSIONS

CEC05	Search method						
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank
f_1	3.61E-16(5.09E-17) 2	1.25E-15(6.46E-16) +/3	3.39E-04(8.68E-05) +/4	1.48E+01(1.19E+00) +/6	4.12E-01(1.26E-01) +/5	3.74E+02(5.63E+02) +/7	1.26E-18(7.18E-19) -/1
f_2	7.13E-01(3.91E-01) 1	2.94E+03(4.59E+02) +/4	2.84E+01(9.88E+00) +/2	1.91E+02(3.03E+01) +/3	5.25E+04(1.21E+04) +/7	2.06E+04(5.44E+03) +/6	1.48E+04(2.49E+03) +/5
f_3	2.24E+06(1.21E+06) 2	2.87E+07(5.11E+06) +/6	5.40E+06(1.68E+06) +/3	1.07E+06(2.40E+05) -/1	1.90E+07(9.73E+06) +/4	6.22E+07(3.57E+07) +/5	6.04E+07(9.90E+06) +/5
f_4	3.71E+03(1.44E+03) 3	1.13E+04(1.58E+03) +/4	1.02E+02(3.43E+01) -/1	2.14E+02(2.91E+01) -/2	1.48E+05(3.30E+04) +/7	4.76E+04(1.23E+04) +/6	3.06E+04(4.80E+03) +/5
f_5	8.11E+03(1.71E+03) 4	3.31E+03(1.58E+02) -/2	7.45E+03(1.93E+03) ≈/4	4.58E+02(3.38E+01) -/1	1.68E+04(5.53E+03) +/6	1.06E+04(2.69E+03) +/5	6.17E+03(9.55E+02) -/3
f_6	6.77E+02(1.15E+02) 3	3.85E+01(2.19E+01) -/1	2.91E+03(3.36E+03) +/4	2.93E+03(5.19E+02) +/4	1.18E+04(8.20E+03) +/5	1.01E+08(1.29E+08) +/6	4.81E+01(2.48E+01) -/2
f_7	1.35E-02(7.67E-03) 1	4.55E-01(9.83E-02) +/2	1.29E+00(6.21E-02) +/4	9.43E-01(5.24E-02) +/3	6.37E+01(8.40E+01) +/5	8.47E+02(2.76E+02) +/6	8.82E-01(1.25E-01) +/3
f_8	2.03E+01(6.32E-02) 1	2.09E+01(6.21E-02) +/4	2.05E+01(4.42E-02) +/2	2.09E+01(4.42E-02) +/4	2.07E+01(9.64E-02) +/3	2.07E+01(1.38E-01) +/3	2.09E+01(4.21E-02) +/5
f_9	9.86E+00(3.15E+00) 2	1.89E+02(9.20E+00) +/4	2.46E+02(4.04E+01) +/6	2.02E+02(9.41E+00) +/5	2.13E+02(4.66E+01) +/7	1.48E+01(4.16E+01) +/3	1.00E-01(3.06E-01) -/1
f_{10}	2.99E+02(5.70E+01) 3	2.20E+02(1.22E+01) -/2	5.86E+02(6.02E+01) +/5	2.46E+02(1.59E+01) ≈/3	4.29E+02(9.57E+01) +/4	2.18E+02(8.34E+01) -/2	2.04E+02(1.48E+01) -/1
f_{11}	3.22E+01(3.44E+00) 2	3.66E+01(1.22E+00) +/5	3.55E+01(3.05E+00) +/4	3.99E+01(9.91E-01) +/6	3.56E+01(3.38E+00) +/4	2.66E+01(3.25E+00) -/1	3.30E+01(1.24E+00) +/3
f_{12}	1.23E+04(1.05E+04) 2	1.84E+05(2.23E+04) +/5	7.46E+03(5.15E+03) -/1	6.38E+05(8.88E+04) +/6	1.45E+05(9.66E+04) +/4	6.89E+04(2.24E+01) +/3	7.55E+04(1.03E+04) +/3
f_{13}	3.05E+00(9.80E-01) 1	1.63E+01(8.61E-01) +/4	3.04E+01(5.93E+00) +/6	3.96E+02(1.55E+02) +/6	2.02E+01(7.06E+00) +/5	1.28E+01(5.37E+00) +/3	5.49E+00(5.17E-01) +/2
f_{14}	1.31E+01(4.34E-01) 1	1.32E+01(1.84E-01) ≈/1	1.32E+01(4.62E-01) ≈/1	1.32E+01(2.24E-01) ≈/1	1.34E+01(3.96E-01) +/2	1.35E+01(2.87E-01) +/3	1.34E+01(1.87E-01) +/2
+/-	10/1/3	10/2/2	10/2/2	9/2/3	14/0/0	12/0/2	9/0/5
Avg-rank	2.00	3.36	3.35	3.64	4.78	4.00	2.93

"+", "≈", "-" respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE V. MINIMIZATION RESULTS FOR 14 FUNCTIONS FROM CEC 2005 OVER 30 RUNS AT 50 DIMENSIONS

CEC05	Search method						
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank
f_1	4.62E-16(6.12E-16) 1	4.74E-15(3.35E-15) +/2	9.84E-04(2.01E-04) +/4	1.77E+01(1.57E+00) +/5	1.85E+00(9.14E-01) +/5	4.99E+03(3.98E+03) +/6	2.23E-14(1.11E-14) +/3
f_2	2.76E+01(9.03E+00) 1	2.43E+04(1.92E+03) +/4	3.66E+02(5.31E+01) +/2	5.11E+02(8.93E+01) +/3	1.32E+05(2.30E+04) +/7	7.45E+04(1.87E+04) +/6	6.91E+04(8.25E+03) +/5
f_3	2.98E+06(6.56E+05) 2	1.08E+08(1.13E+07) +/5	1.45E+07(3.33E+06) +/3	1.43E+06(2.78E+05) +/1	5.41E+07(1.93E+07) +/4	2.33E+08(9.67E+07) +/6	2.37E+08(4.36E+07) +/6
f_4	2.42E+04(5.10E+03) 3	4.49E+04(5.15E+03) +/4	1.94E+03(1.16E+03) -/2	5.86E+02(9.07E+01) -/1	4.87E+05(1.68E+05) +/7	1.65E+05(5.18E+04) +/6	1.02E+05(1.31E+04) +/5
f_5	2.03E+04(2.48E+03) 4	8.63E+03(5.27E+02) -/2	1.97E+04(3.34E+03) ≈/4	2.52E+03(2.17E+02) -/1	2.93E+04(3.27E+03) +/6	2.36E+04(2.50E+03) +/5	1.64E+04(1.33E+03) -/3
f_6	7.54E+02(1.40E+03) 3	4.72E+01(4.04E+00) -/1	2.64E+03(4.10E+03) +/4	4.43E+03(6.00E+02) +/4	1.62E+04(1.05E+03) +/5	1.25E+09(7.34E+08) +/6	9.68E+01(2.98E+01) -/2

f_7	5.32E-03(1.01E-02) 1	7.88E-01(1.20E-01) +/2	1.71E+00(1.49E-01) +/5	9.53E-01(2.33E-02) +/3	1.35E+02(1.34E+02) +/6	2.18E+03(4.35E+02) +/7	1.07E+00(2.92E-02) +/4
f_8	2.03E+01(5.42E-02) 1	2.11E+01(4.45E-02) +/4	2.09E+01(4.54E-02) +/3	2.11E+01(3.62E-02) +/4	2.09E+01(1.09E-01) +/3	2.08E+01(9.03E-02) +/2	2.12E+01(3.29E-02) +/5
f_9	3.55E+01(6.80E+00) 2	3.96E+02(1.48E+01) +/4	5.19E+02(5.54E+01) +/7	4.65E+02(1.59E+01) +/6	4.12E+02(6.68E+01) +/5	3.42E+02(6.39E+02) +/3	1.29E+00(1.56E+00) -/1
f_{10}	6.50E+02(7.30E+01) 5	4.36E+02(1.95E+01) -/1	1.13E+03(1.18E+02) +/8	5.85E+02(1.65E+01) -/4	8.31E+02(9.10E+01) +/6	5.03E+02(1.11E+02) -/3	4.44E+02(2.14E+01) -/2
f_{11}	6.35E+01(4.92E+00) 2	6.95E+01(1.34E+00) +/4	6.39E+01(4.45E+00) \approx /2	7.29E+01(1.60E+00) +/6	6.80E+01(3.97E+00) +/5	5.16E+02(5.41E+00) +/7	6.62E+01(1.29E+00) +/3
f_{12}	5.03E+04(3.27E+04) 2	8.52E+05(8.47E+04) +/5	2.19E+05(1.39E+05) +/3	3.17E+06(2.50E+05) +/6	5.14E+05(3.85E+05) +/4	4.93E+05(1.52E+05) +/4	4.00E+05(4.49E+04) +/3
f_{13}	7.03E+00(1.74E+00) 1	3.39E+01(1.14E+00) +/3	6.29E+01(7.09E+00) +/5	2.40E+02(3.51E+01) +/6	4.75E+01(1.19E+01) +/4	5.54E+01(2.24E+01) +/4	1.60E+01(7.57E-01) +/2
f_{14}	2.27E+01(3.59E-01) 1	2.30E+01(1.77E-01) +/2	2.29E+01(4.78E-01) \approx/1	2.30E+01(2.31E-01) +/2	2.29E+01(4.94E-01) \approx/1	2.31E+01(4.27E-01) +/3	2.31E+01(1.68E-01) +/3
	+/ \approx -	11/0/3	10/2/1	11/0/3	13/1/0	13/0/1	10/0/4
	Avg-rank 2.07	3.07	3.71	3.71	4.86	4.85	3.21

“+”, “ \approx ”, “-” respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE VI. MINIMIZATION RESULTS FOR 30 FUNCTIONS FROM CEC 2014 OVER 30 RUNS AT 10 DIMENSIONS

CEC14	Search method							
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank	
f_1	3.76E+04(3.61E+04) 1	6.61E+05(2.78E+05) +/4	9.26E+05(5.73E+05) +/5	1.57E+05(9.53E+04) +/3	9.92E+06(3.37E+06) +/6	1.20E+07(1.06E+07) +/7	6.68E+04(3.90E+04) +/2	
f_2	1.20E+03(1.37E+03) 2	4.97E+04(2.35E+04) +/3	3.23E+03(4.36E+03) \approx /2	2.64E+06(6.84E+05) +/5	6.41E+04(4.07E+04) +/4	3.44E+03(3.59E+03) \approx /2	6.98E+01(5.56E+01) -/1	
f_3	1.49E+03(1.81E+03) 3	6.96E+03(2.03E+03) +/5	2.23E+02(1.65E+02) -/2	1.19E+04(4.72E+03) +/6	6.64E+04(2.66E+04) +/7	6.22E+03(6.49E+03) +/4	1.29E+02(1.72E+02) -/1	
f_4	1.93E+01(1.68E+01) 3	1.73E+00(2.70E+00) -/1	2.88E+01(2.01E+01) +/4	3.77E+01(1.00E+01) +/7	3.35E+01(2.00E+01) +/6	3.73E+01(1.88E+01) +/5	4.58E+00(5.72E+00) -/2	
f_5	2.00E+01(3.47E-04) 2	2.03E+01(2.43E-01) +/6	2.00E+01(2.04E-02) +/4	1.92E+01(4.20E+00) -/1	2.03E+01(1.86E-01) +/5	2.00E+01(6.60E-02) +/3	2.00E+01(3.03E-01) +/4	
f_6	4.09E+00(1.57E+00) 3	4.88E+00(7.41E-01) +/4	6.37E+00(1.43E+00) +/5	2.75E+00(1.26E+00) -/2	7.47E+00(1.60E+00) +/6	2.59E+00(1.68E+00) -/2	2.38E+00(5.95E-01) -/1	
f_7	3.77E-01(1.93E-01) 4	2.235E-01(5.52E-02) -/3	1.34E-01(6.63E-02) -/2	8.36E-01(8.71E-02) +/5	9.35E-01(5.15E-01) +/5	2.41E-01(2.18E-01) -/3	6.18E-02(2.17E-02) -/1	
f_8	1.00E-01(3.06E-01) 2	2.37E+01(3.06E+00) +/4	4.25E+01(1.71E+01) +/6	1.35E+01(3.61E+00) +/3	3.74E+01(1.19E+01) +/5	2.38E+01(1.02E+01) +/4	0.00E+00(0.00E+00) -/1	
f_9	2.02E+01(8.68E+00) 2	3.04E+01(4.38E+00) +/3	4.85E+01(1.49E+01) +/4	1.79E+01(4.69E+00) \approx /2	4.76E+01(1.57E+01) +/4	2.45E+01(1.29E+01) \approx /2	5.89E+00(1.49E+00) -/1	
f_{10}	7.12E+01(7.72E+01) 2	1.05E+03(1.23E+02) +/4	1.07E+03(3.12E+02) +/4	5.44E+02(1.34E+02) +/3	5.24E+02(1.73E+02) +/5	5.95E+02(2.64E+02) +/3	9.71E+00(8.23E+00) -/1	
f_{11}	5.78E+02(2.86E+02) 2	1.26E+03(1.42E+02) +/5	1.04E+03(3.23E+02) +/4	5.64E+02(1.39E+02) \approx /2	1.09E+03(3.21E+02) +/4	7.92E+02(3.47E+02) +/3	3.57E+02(1.14E+02) -/1	
f_{12}	1.46E-01(1.10E-01) 1	1.16E+00(2.11E-01) +/6	3.54E-01(1.54E-01) +/3	9.40E-01(1.83E-01) +/5	7.24E-01(3.42E-01) +/4	2.56E-01(1.96E-01) +/2	3.48E-01(6.45E-02) +/3	
f_{13}	3.87E-01(1.53E-01) 3	2.18E-01(3.39E-02) -/2	9.00E-02(2.61E-02) -/1	9.82E-02(1.57E-02) -/1	4.89E-01(1.67E-01) +/4	3.28E-01(1.68E-01) \approx /3	2.09E-01(4.07E-02) -/2	
f_{14}	2.48E-01(6.76E-02) 3	2.17E-01(4.09E-02) \approx /2	5.23E-02(2.70E-02) -/1	4.39E-01(1.66E-02) +/4	3.18E-01(2.18E-01) \approx /3	3.16E-01(9.84E-02) +/3	1.75E-01(2.98E-02) -/2	
f_{15}	2.50E+00(1.12E+00) 5	2.27E+00(2.29E-01) \approx /5	9.12E-01(3.07E-01) -/1	1.97E+00(3.42E-01) +/4	6.26E+00(2.23E+01) +/6	1.75E+00(8.00E-01) -/3	1.09E+00(1.97E-01) -/2	
f_{16}	2.53E+00(5.34E-01) 1	3.31E+00(2.10E-01) +/2	2.87E+00(4.54E-01) +/2	3.24E+00(2.83E-01) +/2	3.26E+00(4.10E-01) +/2	3.21E+00(3.94E-01) +/2	2.50E+00(2.06E-01) \approx/1	
f_{17}	2.84E+03(3.01E+03) 1	8.04E+03(3.92E+03) +/2	4.20E+03(4.00E+03) \approx/1	5.26E+04(8.74E+04) +/3	1.92E+05(3.44E+05) +/6	2.79E+05(3.75E+05) +/6	3.41E+04(1.88E+04) +/4	
f_{18}	7.59E+03(7.23E+03) 3	2.98E+03(1.40E+03) -/2	1.06E+04(1.08E+04) \approx /3	6.67E+03(3.10E+03) +/3	1.18E+04(1.32E+04) \approx /3	8.49E+03(7.50E+03) \approx /3	6.72E+02(5.67E+02) -/1	
f_{19}	1.96E+00(1.07E+00) 2	2.58E+00(2.11E-01) +/3	4.31E+00(1.40E+00) +/4	2.03E+00(5.33E-01) \approx /2	5.01E+00(1.23E+00) +/5	2.77E+00(1.34E+00) +/3	6.29E-01(2.29E-01) -/1	
f_{20}	2.31E+03(3.38E+03) 3	6.81E+02(4.03E+02) \approx /3	9.77E+01(3.15E+01) -/2	4.58E+03(2.83E+03) +/4	3.75E+03(3.47E+03) +/4	1.00E+04(9.68E+03) +/5	4.67E-01(1.77E-01) -/1	

f_{21}	2.02E+03(2.85E+03) 2	2.74E+03(1.10E+03) +/3	4.68E+03(4.37E+03) +/4	5.44E+03(4.74E+03) +/5	1.91E+04(1.71E+04) +/6	5.56E+04(8.41E+04) +/6	1.84E+03(1.81E+03) -/1
f_{22}	7.14E+01(6.55E+01) 2	4.50E+01(8.60E+00) ≈/2	1.57E+02(1.01E+02) +/4	1.55E+02(1.21E+01) +/4	6.68E+01(4.84E+01) ≈/2	1.17E+02(8.49E+01) +/3	2.03E+00(1.73E+00) -/1
f_{23}	3.29E+02(2.40E-09) 2	9.88E+01(1.53E+02) -/1	3.30E+02(6.42E-01) +/4	3.23E+02(2.29E+01) +/5	3.13E+02(4.53E+01) +/4	3.31E+02(3.87E+00) +/4	3.29E+02(7.40E-07) +/3
f_{24}	1.33E+02(9.95E+00) 2	1.34E+02(4.67E+00) ≈/2	1.82E+02(2.44E+01) +/4	1.88E+02(2.77E+01) +/4	1.73E+02(2.62E+01) +/4	1.39E+02(2.19E+01) +/3	1.17E+02(3.00E+00) -/1
f_{25}	1.89E+02(2.35E+01) 3	1.69E+02(6.81E+00) -/2	1.99E+02(7.42E+00) ≈/3	1.99E+02(7.15E-01) ≈/3	1.89E+02(1.44E+01) ≈/3	1.94E+02(1.99E+01) ≈/3	1.49E+02(1.07E+02) -/1
f_{26}	1.00E+02(9.80E-02) 1	1.00E+02(3.36E-02) ≈/1	1.00E+02(2.75E-02) ≈/1	1.01E+02(2.92E+00) ≈/1	1.00E+02(1.87E-01) +/2	1.00E+02(1.21E-01) ≈/1	1.00E+02(4.25E-02) ≈/1
f_{27}	1.41E+02(1.90E+02) 2	6.31E+01(3.45E+01) -/1	4.32E+02(1.89E+02) +/5	3.66E+02(5.02E+01) +/3	3.86E+02(1.38E+02) +/4	3.03E+02(1.69E+02) +/3	1.88E+02(1.76E+02) ≈/2
f_{28}	5.20E+02(1.90E+02) 3	3.07E+02(1.60E-01) -/1	8.78E+02(2.14E+02) +/6	6.53E+02(1.89E+02) +/5	5.45E+02(1.17E+01) ≈/3	5.63E+02(1.30E+02) +/4	3.74E+02(4.80E+00) -/2
f_{29}	1.72E+05(5.26E+05) 4	2.04E+02(4.79E-01) -/1	2.54E+05(8.13E+05) +/5	4.04E+06(1.05E+07) +/6	1.73E+05(5.26E+05) +/5	7.96E+05(1.21E+06) +/5	5.99E+02(7.10E+01) -/2
f_{30}	1.12E+03(4.16E+02) 4	2.54E+02(1.28E+01) -/1	1.92E+03(9.31E+02) +/5	8.52E+02(1.59E+02) -/3	1.67E+03(9.33E+02) +/5	1.67E+03(5.54E+02) +/5	5.33E+02(4.05E+01) -/2
	+/≈/-	14/6/10	19/4/6	20/6/4	25/5/0	21/6/3	5/3/22
	Avg-rank	2.80	3.36	3.53	4.73	3.50	1.63
		2.46					

“+”, “≈”, “-” respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE VII. MINIMIZATION RESULTS FOR 30 FUNCTIONS FROM CEC 2014 OVER 30 RUNS AT 30 DIMENSIONS

CEC14	Search method						
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank
f_1	1.21E+06(5.53E+05) 3	8.57E+07(1.29E+07) +/7	3.56E+07(1.52E+07) +/5	5.94E+05(1.25E+05) -/1	3.08E+07(1.58E+06) +/5	7.47E+07(4.04E+07) +/6	1.76E+07(5.96E+06) +/4
f_2	1.15E+04(8.29E+03) 2	1.55E+06(1.49E+06) +/3	2.24E+08(1.00E+08) +/6	1.94E+07(2.29E+06) +/5	2.99E+06(3.82E+06) +/4	5.52E+08(4.32E+08) +/7	5.84E-9(0.00E+00) -/1
f_3	9.70E+02(1.21E+03) 2	8.60E+04(1.40E+04) +/6	3.56E+03(1.11E+03) +/3	6.42E+03(2.59E+03) +/4	3.51E+04(2.85E+03) +/4	5.03E+04(2.90E+04) +/5	4.30E+01(4.50E+01) -/1
f_4	7.50E+01(4.18E+01) 3	9.44E+00(1.26E+01) -/1	1.55E+02(4.87E+01) +/5	6.00E+01(3.61E+01) -/2	1.82E+02(5.36E+01) +/5	2.45E+02(8.07E+01) +/6	1.03E+02(1.33E+01) +/4
f_5	2.00E+01(1.87E-04) 1	2.09E+01(5.00E-2) +/5	2.00E+01(6.15E-04) +/2	2.09E+01(6.40E-02) +/5	2.04E+01(1.92E-01) +/4	2.02E+01(1.44E-01) +/3	2.04E+01(4.20E-02) +/4
f_6	2.19E+01(3.61E+00) 4	3.43E+01(1.21E+00) +/5	3.11E+01(3.81E+00) +/5	1.63E+01(2.55E+00) -/1	3.50E+01(3.54E+00) +/5	2.03E+01(4.20E+00) -/3	1.72E+01(1.32E+00) -/2
f_7	2.75E-02(3.91E-02) 3	6.37E-04(5.66E-04) -/2	2.77E+00(5.23E-01) +/6	1.18E+00(2.35E-02) +/5	9.90E-01(7.26E-02) +/4	4.52E+00(5.53E+00) +/6	1.50E-04(2.42E-04) -/1
f_8	6.88E+00(2.91E+00) 2	1.81E+02(1.38E+01) +/4	2.00E+02(4.16E+01) +/4	1.17E+02(1.02E+01) +/3	1.88E+02(3.90E+01) +/4	1.21E+02(2.92E+01) +/3	7.14E-01(1.06E+00) -/1
f_9	1.23E+02(2.64E+01) 2	2.05E+02(9.97E+00) +/4	2.36E+02(4.73E+01) +/5	1.34E+02(1.28E+01) +/3	2.31E+02(6.62E+01) +/5	1.52E+02(4.80E+01) +/3	8.78E+01(9.36E+00) -/1
f_{10}	4.63E+02(1.77E+02) 2	6.54E+03(2.63E+02) +/7	4.13E+03(6.33E+02) +/5	2.50E+03(3.59E+02) +/3	4.00E+03(8.70E+02) +/5	2.98E+03(5.72E+02) +/4	2.55E+01(2.87E+01) -/1
f_{11}	3.36E+03(5.77E+02) 2	7.14E+03(2.50E+02) +/4	4.43E+03(8.18E+02) +/3	2.87E+03(4.48E+02) -/1	5.19E+03(8.10E+02) +/3	3.62E+03(5.88E+02) ≈/2	3.44E+03(2.75E+02) ≈/2
f_{12}	2.92E-01(8.74E-02) 1	2.43E+00(3.08E-01) +/7	1.15E+00(2.52E-01) +/5	6.61E-01(1.12E-01) +/4	1.62E+00(3.90E-01) +/6	3.83E-01(1.79E-01) +/2	5.93E-01(8.21E-02) +/3
f_{13}	3.88E-01(1.07E-01) 3	3.48E-01(3.32E-02) ≈/3	2.80E-01(8.63E-02) -/2	1.70E-01(1.97E-02) -/1	5.30E-01(8.99E-02) +/4	4.88E-01(1.09E-01) +/4	4.50E-01(4.26E-02) +/4
f_{14}	2.66E-01(8.99E-02) 2	2.39E-01(1.86E-02) -/1	2.31E-01(1.07E-01) -/1	3.50E-01(5.11E-02) +/5	2.71E-01(5.26E-02) +/3	2.99E-01(1.03E-01) +/3	3.08E-01(4.71E-02) +/4
f_{15}	3.26E+01(1.14E+01) 3	1.65E+01(9.23E-01) -/2	1.24E+01(6.07E+00) -/1	1.30E+01(7.77E-01) -/1	6.68E+01(2.52E+01) +/4	3.66E+02(5.95E+02) +/5	1.01E+01(8.53E-01) -/1
f_{16}	1.17E+01(4.25E-01) 2	1.30E+01(1.74E-01) +/5	1.20E+01(6.09E-01) +/3	1.28E+01(3.03E-01) +/4	1.25E+01(5.20E-01) +/3	1.27E+01(5.58E-01) +/3	1.08E+01(2.81E-01) -/1
f_{17}	2.22E+05(1.71E+05) 3	1.38E+06(3.71E+05) +/4	8.57E+05(5.87E+05) +/4	3.23E+04(2.06E+04) -/1	4.12E+06(2.54E+06) +/6	4.42E+06(3.40E+06) +/6	3.02E+06(1.51E+06) +/5
f_{18}	2.20E+03(2.76E+03) 3	7.07E+02(5.12E+02) -/2	3.97E+03(4.44E+03) +/6	3.13E+02(2.63E+02) -/1	5.52E+03(4.43E+03) +/6	3.04E+03(3.65E+03) +/4	9.99E+03(4.54E+03) +/5

f_{19}	1.03E+01(1.93E+01) 1	1.93E+01(7.68E-01) +/3	2.55E+01(1.64E+01) +/4	1.76E+01(2.15E+01) +/2	5.96E+01(5.42E+01) +/4	5.44E+01(3.08E+01) +/4	1.13E+01(6.47E-01) +/1
f_{20}	4.98E+02(1.14E+02) 1	2.78E+04(8.29E+03) +/4	5.52E+02(4.25E+02) ≈1	1.68E+04(3.35E+03) +/3	2.72E+04(2.11E+04) +/4	3.54E+04(1.57E+04) +/5	7.34E+03(3.04E+03) +/2
f_{21}	1.95E+05(1.33E+05) 3	3.97E+05(1.22E+05) +/4	1.95E+05(9.13E+04) ≈/3	5.52E+04(1.88E+04) -/1	9.09E+05(8.13E+05) +/7	8.16E+05(7.30E+05) +/6	4.97E+05(1.71E+05) +/5
f_{22}	5.38E+02(1.63E+02) 2	6.13E+02(8.92E+01) +/3	6.57E+02(1.90E+02) +/4	9.52E+02(1.94E+02) +/6	8.15E+02(2.20E+02) +/5	5.55E+02(2.56E+02) ≈/2	1.92E+02(7.98E+01) -/1
f_{23}	3.15E+02(3.09E-04) 1	3.14E+02(9.16E-04) ≈/1	3.51E+02(1.48E+01) +/6	3.16E+02(1.76E-01) +/4	3.33E+02(1.08E+01) +/5	3.30E+02(1.03E+01) +/5	3.17E+02(1.86E+00) +/4
f_{24}	2.29E+02(7.53E+00) 4	2.27E+02(5.23E-01) ≈/4	2.64E+02(2.73E+01) +/5	2.08E+02(5.37E-01) -/1	2.07E+02(4.55E+00) -/2	2.49E+02(7.43E+00) +/5	2.25E+02(1.75E+00) -/3
f_{25}	2.21E+02(3.49E+00) 4	2.15E+02(2.78E+00) -/3	2.32E+02(1.83E+01) +/5	2.02E+02(7.92E-02) -/1	2.19E+02(1.91E+01) ≈/4	2.19E+02(6.64E+00) ≈/4	2.07E+02(1.08E+00) -/2
f_{26}	1.24E+02(4.28E+01) 3	1.00E+02(6.05E-02) -/1	1.65E+02(6.64E+01) +/6	1.78E+02(4.12E+01) +/6	1.00E+02(1.24E-01) -/1	1.27E+02(4.52E+01) +/4	1.00E+02(5.41E-02) -/2
f_{27}	5.83E+02(3.42E+02) 2	7.58E+02(1.46E+02) +/4	8.36E+02(4.27E+02) +/4	8.39E+02(3.89E+02) +/5	9.66E+02(4.05E+02) +/5	6.66E+02(2.08E+02) +/3	4.43E+02(2.29E+01) -/1
f_{28}	2.32E+03(3.39E+02) 5	4.25E+02(5.55E+00) -/1	3.95E+03(1.05E+03) +/6	2.40E+03(5.72E+02) ≈/5	2.33E+03(5.56E+02) -/3	2.09E+03(5.65E+02) -/2	8.65E+02(2.64E+01) -/4
f_{29}	2.79E+05(2.19E+06) 4	2.31E+02(3.84E+00) -/1	1.67E+04(1.94E+04) -/3	4.08E+07(5.88E+07) +/7	5.34E+06(4.77E+06) +/6	2.61E+06(3.34E+06) +/5	4.62E+04(1.01E+04) -/2
f_{30}	3.61E+03(8.28E+02) 3	7.60E+02(1.25E+02) -/1	2.66E+04(2.80E+04) +/5	1.97E+03(3.86E+02) -/2	6.86E+04(3.13E+04) +/6	3.08E+04(3.58E+04) +/5	5.60E+03(1.88E+03) +/4
	+/-/≈-	17/3/10	23/3/4	17/1/12	26/1/3	25/3/2	13/1/16
	-						
	Avg-rank 2.53	3.40	4.10	3.06	4.33	4.40	2.56

“+”, “≈”, “-” respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE VIII. MINIMIZATION RESULTS FOR 30 FUNCTIONS FROM CEC 2014 OVER 30 RUNS AT 50 DIMENSIONS

CEC14	Search method							
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank	
f_1	2.36E+06(9.42E+05) 2	4.64E+08(5.76E+07) +/6	1.03E+08(2.55E+07) +/5	1.21E+06(1.87E+05) -/1	3.13E+07(8.67E+06) +/3	7.57E+07(4.18E+07) +/4	8.57E+07(1.29E+07) +/7	
f_2	5.90E+03(5.79E+03) 2	3.06E+01(1.43E+02) -/1	7.11E+08(1.93E+08) +/7	2.75E+07(2.15E+06) +/5	2.16E+07(1.66E+07) +/4	5.04E+08(1.31E+09) +/6	1.53E+04(9.99E+03) +/3	
f_3	1.95E+03(9.38E+02) 1	2.52E+05(3.22E+04) +/7	5.88E+03(1.09E+03) +/2	1.33E+04(2.62E+03) +/4	4.37E+04(1.21E+04) +/5	5.23E+04(2.51E+04) +/6	9.39E+03(3.23E+03) +/3	
f_4	1.04E+02(4.74E+01) 4	4.38E+01(1.26E+00) -/2	4.47E+02(1.17E+02) +/6	6.28E+01(2.87E+01) -/3	2.37E+02(6.19E+01) +/5	2.38E+02(5.97E+01) +/5	1.14E+02(1.02E+01) ≈/4	
f_5	2.00E+01(2.40E-05) 1	2.11E+01(2.45E-02) +/6	2.00E+01(2.38E-04) +/2	2.11E+01(3.30E-02) +/6	2.04E+01(1.66E-01) +/4	2.02E+01(9.20E-02) +/3	2.06E+01(3.65E-02) +/5	
f_6	4.66E+01(5.02E+00) 3	6.61E+01(1.52E+00) +/4	5.94E+01(4.62E+00) +/4	3.44E+01(3.50E+00) -/2	6.51E+01(4.17E+00) +/4	1.93E+01(3.24E+00) -/1	4.53E+01(1.78E+00) +/3	
f_7	1.03E-02(9.58E-03) 2	3.17E-03(3.95E-03) -/1	8.99E+00(2.38E+00) +/6	1.26E+00(2.29E-02) +/4	1.17E+00(1.25E-01) +/3	6.39E+00(4.97E+00) +/5	5.40E-03(4.10E-03) -/1	
f_8	3.12E+01(7.48E+00) 2	3.95E+02(8.76E+00) +/4	3.93E+02(6.16E+01) +/7	2.54E+02(1.54E+01) +/5	3.30E+02(5.27E+01) +/6	1.26E+02(3.78E+01) +/3	1.36E+00(1.33E+00) -/1	
f_9	3.05E+02(5.00E+01) 3	4.16E+02(1.22E+01) +/4	5.34E+02(7.37E+01) +/5	2.56E+02(1.92E+01) -/2	4.52E+02(9.23E+01) +/4	1.39E+02(3.64E+01) -/1	2.88E+02(1.26E+01) -/2	
f_{10}	9.65E+02(2.44E+02) 2	1.28E+04(3.16E+02) +/6	7.70E+03(8.65E+02) +/5	5.29E+03(5.13E+02) +/4	7.45E+03(1.17E+03) +/5	3.07E+03(6.11E+02) +/3	7.71E+01(6.65E+01) -/1	
f_{11}	6.34E+03(9.18E+02) 2	1.37E+04(3.61E+02) +/5	7.60E+03(8.43E+02) +/3	6.22E+03(4.55E+02) ≈/2	8.94E+03(1.55E+03) +/3	3.73E+03(6.42E+02) -/1	9.13E+03(3.88E+02) +/4	
f_{12}	4.07E-01(9.21E-02) 2	3.29E+00(3.21E-01) +/7	1.77E+00(2.73E-01) +/5	6.50E-01(7.85E-02) +/3	2.60E+00(5.28E-01) +/6	3.54E-01(1.77E-01) -/1	1.01E+00(1.32E-01) +/4	
f_{13}	4.97E-01(1.03E-01) 2	5.07E-01(4.33E-02) ≈/2	4.35E-01(8.45E-02) ≈/2	2.29E-01(2.29E-02) -/1	5.88E-01(1.31E-01) +/3	4.50E-01(1.17E-01) ≈/2	5.67E-01(6.11E-02) +/3	
f_{14}	3.08E-01(3.84E-02) 2	2.81E-01(2.31E-02) -/1	3.40E-01(1.45E-01) +/3	3.80E-01(3.72E-02) ≈/2	4.00E-01(1.50E-01) +/4	3.70E-01(3.98E-01) ≈/2	3.38E-01(4.06E-02) +/3	
f_{15}	9.70E+01(2.42E+01) 4	3.45E+01(9.53E-01) -/3	1.02E+02(7.29E+01) ≈/4	2.43E+01(1.15E+00) -/1	2.43E+02(4.89E+01) +/6	2.41E+02(4.15E+02) ≈/4	3.03E+01(1.22E+00) -/2	
f_{16}	2.08E+01(6.99E-01) 3	2.28E+01(1.67E-01) +/5	2.11E+01(7.16E-01) +/4	2.20E+01(6.20E-01) +/4	2.22E+01(5.45E-01) +/4	1.27E+01(6.14E-01) -/1	2.06E+01(2.55E-01) -/2	
f_{17}	3.24E+05(1.82E+05) 3	1.06E+07(2.41E+06) +/5	3.83E+06(1.36E+06) +/4	3.53E+04(1.89E+04) -/1	1.68E+07(9.44E+06) +/6	3.84E+06(2.73E+06) +/4	1.06E+07(3.74E+06) +/5	

f_{18}	1.62E+03(1.30E+03)) 1	2.44E+03(1.43E+03)) +/2	2.62E+05(1.18E+06)) +/4	1.79E+03(5.82E+02)) ≈/1	7.87E+03(7.02E+03)) +/3	1.06E+04(3.94E+04)) +/4	8.13E+03(4.61E+03)) +/3
f_{19}	5.54E+01(3.06E+01)) 3	3.57E+01(6.18E-01)) ≈/3	7.44E+01(2.08E+01)) +/4	2.00E+01(6.14E+00)) -/2	7.65E+01(2.80E+01)) +/4	6.20E+01(3.51E+01)) ≈/3	3.33E+01(6.47E+00)) ≈/3
f_{20}	4.82E+02(2.00E+02)) 1	9.15E+04(2.80E+04)) +/5	1.12E+03(2.56E+02)) +/2	1.96E+04(3.96E+03)) +/3	1.14E+05(1.96E+05)) +/5	3.33E+04(1.88E+04)) +/4	2.16E+04(6.66E+03)) +/3
f_{21}	2.73E+05(1.59E+05)) 3	4.39E+06(9.70E+05)) +/5	1.94E+06(1.15E+06)) +/5	4.15E+04(1.01E+04)) -/1	6.36E+06(3.69E+06)) +/6	9.33E+05(8.40E+05)) +/4	5.14E+06(2.44E+06)) +/5
f_{22}	1.48E+03(3.24E+02)) 3	1.89E+03(1.51E+02)) +/4	1.38E+03(3.65E+02)) ≈/3	1.76E+03(3.76E+02)) +/4	1.97E+03(4.89E+02)) +/5	6.41E+02(1.76E+02)) -/1	8.20E+02(1.58E+02)) -/2
f_{23}	3.44E+02(1.34E-04)) 5	3.37E+02(5.14E-03)) -/2	4.29E+02(2.62E+01)) +/7	3.43E+02(6.01E+00)) -/4	3.73E+02(4.80E+01)) +/6	3.28E+02(7.79E+00)) -/1	3.44E+02(8.89E-01)) ≈/5
f_{24}	2.71E+02(8.22E+00)) 6	2.53E+02(1.99E+01)) -/3	3.30E+02(2.04E+01)) +/7	2.19E+02(6.76E+00)) -/1	2.00E+02(3.56E-01)) -/2	2.49E+02(7.32E+00)) -/4	2.59E+02(2.75E+00)) -/5
f_{25}	2.51E+02(1.19E+01)) 4	2.48E+02(4.58E+00)) ≈/4	2.67E+02(2.25E+01)) +/5	2.03E+02(1.13E-01)) -/1	2.08E+02(1.96E+01)) -/2	2.22E+02(5.24E+00)) -/3	2.22E+02(2.65E+00)) -/3
f_{26}	1.94E+02(2.53E+01)) 5	1.00E+02(4.44E-02)) -/1	2.05E+02(1.98E+01)) +/6	1.78E+02(4.08E+01)) -/6	1.04E+02(1.82E+01)) -/3	1.42E+02(6.15E+01)) ≈/5	1.01E+02(1.13E-01)) -/2
f_{27}	1.56E+03(2.42E+02)) 3	2.04E+03(4.93E+01)) +/4	2.04E+03(1.58E+02)) +/4	3.31E+03(8.88E+02)) +/6	2.07E+03(1.17E+02)) +/4	7.51E+02(2.07E+02)) -/1	1.36E+03(1.32E+02)) -/2
f_{28}	7.02E+03(1.01E+03)) 6	4.37E+02(1.77E+01)) -/2	8.63E+03(1.48E+03)) +/7	6.07E+03(9.00E+02)) -/5	4.89E+03(1.66E+03)) -/4	2.00E+03(6.21E+02)) -/3	2.94E+01(2.49E+01)) -/1
f_{29}	1.90E+03(7.11E+02)) 2	2.35E+02(1.16E+00)) -/1	4.87E+06(2.61E+07)) +/6	1.77E+08(1.18E+08)) +/7	3.58E+07(1.95E+07)) +/5	1.13E+06(2.87E+06)) +/4	3.48E+06(8.29E+06)) +/5
f_{30}	1.33E+04(2.06E+03)) 4	2.02E+03(1.46E+02)) -/1	1.63E+05(9.26E+04)) +/7	3.01E+04(2.38E+03)) +/5	1.09E+05(9.17E+04)) +/6	4.03E+04(4.21E+04)) +/5	1.18E+04(1.17E+03)) -/3
	+/-/≈	16/3/11	27/3/0	13/3/14	26/0/4	14/5/11	14/3/13
	Avg-rank 2.86	3.56	3.07	3.30	4.30	3.13	3.16

“+”, “≈”, “-” respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

TABLE IX. P-VALUES OBTAINED WITH WILCOXON’S RANK SUM TEST OVER 14 FUNCTIONS FROM CEC 2005 BENCHMARK IN 10D

CE C05	Search Manager vs					
	ABC	FOA	GSA	WOA	CA	DE
f_1	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
f_2	3.02E-11	1.01E-08	3.01E-08	3.02E-11	3.02E-11	3.02E-11
f_3	3.80E-07	8.77E-01	9.23E-01	4.68E-08	8.10E-10	3.02E-11
f_4	3.02E-11	7.22E-06	3.02E-11	3.02E-11	3.02E-11	3.02E-11
f_5	3.77E-12	6.51E-09	3.02E-11	8.84E-07	3.34E-11	1.69E-09
f_6	2.07E-02	1.60E-06	5.57E-10	5.57E-10	7.38E-10	7.95E-01
f_7	6.30E-11	8.35E-08	1.20E-08	5.26E-04	1.33E-10	1.70E-08
f_8	7.39E-11	1.44E-02	6.69E-11	5.09E-06	4.71E-04	4.97E-11
f_9	1.94E-11	1.94E-11	1.94E-11	1.94E-11	1.94E-11	6.81E-13
f_{10}	1.30E-01	6.06E-11	4.08E-11	9.51E-06	3.78E-02	5.10E-03
f_{11}	7.01E-02	1.11E-03	1.56E-08	2.39E-04	3.00E-03	8.53E-01
f_{12}	7.66E-05	2.27E-03	3.02E-11	2.15E-06	9.03E-04	1.54E-01
f_{13}	3.02E-11	3.02E-11	3.02E-11	1.61E-10	2.44E-09	2.00E-01
f_{14}	6.10E-01	1.08E-02	2.22E-01	4.84E-02	3.63E-01	3.00E-03

TABLE X. P-VALUES OBTAINED WITH WILCOXON’S RANK SUM TEST OVER CEC 2014 BENCHMARK FUNCTIONS IN 50D

CE C14	Search Manager vs					
	ABC	FOA	GSA	WOA	CA	DE
f_1	3.02E-11	3.02E-11	1.69E-08	3.02E-11	3.02E-11	3.02E-11
f_2	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.77E-10
f_3	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.87E-10
f_4	1.07E-07	3.02E-11	6.55E-04	2.92E-09	1.46E-10	1.41E-01
f_5	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
f_6	3.02E-11	6.06E-11	6.72E-10	3.01E-11	3.02E-11	1.43E-05
f_7	3.03E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	8.00E-03
f_8	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.84E-11
f_9	1.55E-09	7.39E-11	1.63E-05	2.03E-09	3.02E-11	1.25E-04
f_{10}	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.34E-11	3.02E-11
f_{11}	3.02E-11	1.86E-06	5.99E-01	8.48E-09	1.09E-10	1.95E-10
f_{12}	3.02E-11	3.02E-11	2.37E-10	3.01E-11	1.50E-02	3.02E-11
f_{13}	4.20E-01	2.51E-02	3.02E-11	8.68E-03	9.63E-02	3.00E-03

f_{14}	3.03E-03	8.41E-01	4.68E-08	3.83E-05	1.86E-01	5.80E-03
f_{15}	3.02E-11	2.11E-01	3.02E-11	4.07E-11	5.30E-01	3.02E-11
f_{16}	3.01E-11	3.51E-02	2.83E-08	1.28E-09	3.02E-11	3.77E-04
f_{17}	3.02E-11	4.61E-10	3.68E-11	3.02E-11	8.99E-11	3.02E-11
f_{18}	1.50E-02	4.71E-04	1.12E-01	6.73E-06	3.51E-02	3.16E-05
f_{19}	1.85E-01	3.91E-02	6.01E-08	3.50E-03	7.73E-01	1.58E-01
f_{20}	3.02E-11	4.20E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11
f_{21}	3.02E-11	4.50E-11	4.97E-11	3.34E-11	1.86E-06	3.02E-11
f_{22}	3.09E-06	2.40E-01	8.31E-03	4.08E-05	8.15E-11	1.95E-10
f_{23}	3.02E-11	3.02E-11	3.99E-04	8.48E-09	5.57E-10	1.84E-01
f_{24}	1.99E-05	3.02E-11	3.02E-11	3.02E-11	8.48E-09	1.70E-08
f_{25}	2.77E-01	8.12E-04	3.02E-11	2.74E-08	2.87E-10	3.02E-11
f_{26}	2.23E-09	5.07E-10	3.82E-09	5.07E-10	8.89E-10	5.97E-09
f_{27}	3.02E-11	4.97E-11	8.35E-08	4.97E-11	3.02E-11	9.75E-10
f_{28}	3.01E-11	3.37E-05	6.91E-04	3.37E-05	4.80E-07	3.02E-11
f_{29}	3.02E-11	3.02E-11	5.26E-04	3.02E-11	3.02E-11	3.83E-05
f_{30}	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	7.74E-06

TABLE XI. MINIMIZATION RESULTS FOR REAL-WORLD PROBLEMS

Problem	Search method						
	Search Manager Mean(Std.) Rank	ABC Mean(Std.) Comp/Rank	FOA Mean(Std.) Comp/Rank	GSA Mean(Std.) Comp/Rank	WOA Mean(Std.) Comp/Rank	CA Mean(Std.) Comp/Rank	DE Mean(Std.) Comp/Rank
T01 FM	1.79E+01(4.88E+00) 1	1.64E+01(7.05E+00) ≈/1	2.01E+01(3.24E+00) +/2	2.31E+01(1.95E+00) +/3	2.11E+01(3.70E+00) +/2	2.43E+01(1.85E+00) +/4	1.62E+01(3.68E+00) ≈/1
T02L-J	- 1.29E+01(2.97E+00) 2	-4.71E+00(5.15E-01) +/6	- 5.24E+00(1.58E+00) +/6	-5.58E+00(6.45E-01) +/4	- 1.55E+01(5.46E+00) -/1	- 8.41E+00(1.95E+00) +/5	- 1.01E+01(1.41E+00) +/3
T03BCB	1.15E-05(0.00E+00) 1	1.15E-05(0.00E+00) ≈/1	1.15E-05(0.00E+00) ≈/1	1.15E-05(0.00E+00) ≈/1	1.15E-05(0.00E+00) ≈/1	1.15E-05(0.00E+00) ≈/1	1.15E-05(0.00E+00) ≈/1
T04STR	1.42E+01(2.27E-01) 1	1.43E+01(3.51E-01) +/2	1.61E+01(3.17E+00) +/4	2.24E+01(9.64E-01) +/7	2.09E+01(6.88E+00) +/6	1.82E+01(3.33E+00) +/4	2.00E+01(2.31E+00) +/5
T05Si(B)	- 2.88E+01(1.76E+00) 1	- 2.18E+01(1.08E+00) +/4	- 2.81E+01(7.88E+00) +/2	1.13E+01(3.03E+01) +/6	- 2.87E+01(1.76E+00) ≈/1	- 2.33E+01(3.89E+00) +/3	- 2.46E+01(2.51E+00) +/2
T06Si(C)	- 1.83E+01(2.71E+00) 1	- 1.03E+01(1.27E+00) +/5	- 2.20E+01(3.91E+00) -/1	1.36E+01(7.56E+00) +/7	- 1.19E+01(4.44E+00) +/4	6.54E+00(2.06E+00) +/6	- 1.57E+01(2.66E+00) +/2
T07SPR	1.45E+00(2.11E-01) 3	1.67E+00(9.62E-02) +/4	1.35E+00(2.29E-01) -/2	1.90E+00(8.78E-02) +/6	1.86E+00(2.41E-01) +/5	1.30E+00(1.92E-01) -/2	1.57E+00(7.47E-02) +/4
T08TNE	2.20E+02(0.00E+00) 1	2.20E+02(0.00E+00) ≈/1	2.20E+02(0.00E+00) ≈/1	2.20E+02(0.00E+00) ≈/1	2.20E+02(0.00E+00) ≈/1	2.20E+02(0.00E+00) ≈/1	2.20E+02(0.00E+00) ≈/1
T09 LSTP	1.58E+06(3.32E+05) 5	1.25E+06(6.94E+04) -/4	2.52E+06(6.60E+05) +/6	2.07E+07(1.81E+06) +/6	9.05E+05(4.11E+05) -/3	3.25E+07(8.44E+06) +/7	1.03E+05(8.16E+03) -/2
T10CAA	- 1.37E+01(2.78+00) 2	- 1.63E+01(1.36E+00) -/1	- 1.38E+01(2.80E+00) ≈/2	- 1.68E+01(2.99E-01) -/1	-1.05E+01(6.81E-01) +/4	- 1.21E+01(2.25E+00) +/3	- 1.41E+01(1.45E+00) ≈/2
T11.IDE	9.85E+04(5.68E+04) 2	3.22E+06(4.37E+05) +/4	1.85E+06(7.82E+05) +/4	6.41E+04(1.29E+03) -/1	1.31E+06(1.21E+05) +/3	2.96E+06(1.77E+06) +/4	1.03E+08(1.21E+07) +/6
T11.3EL	1.54E+04(3.20E+01) 1	1.54E+04(1.28E+01) ≈/1	3.91E+04(4.80E+04) +/3	9.30E+04(5.87E+04) +/4	1.56E+04(6.07E+01) +/2	1.55E+04(2.20E+01) +/2	1.55E+04(1.37E+01) ≈/1
T11.8HS	1.16E+06(3.25E+05) 1	2.07E+06(3.48E+05) +/3	1.16E+06(3.90E+05) ≈/1	1.72E+06(1.30E+05) +/2	5.21E+06(2.95E+06) +/5	1.38E+06(7.59E+05) ≈/1	2.30E+06(4.27E+05) +/4
T12(me)	2.25E+01(5.64E+00) 1	3.09E+01(3.71E+00) +/3	2.98E+01(5.41E+00) +/3	2.99E+01(4.38E+00) +/3	3.69E+01(7.58E+00) +/4	2.64E+01(6.92E+00) +/2	2.28E+01(2.56E+00) ≈/1
T13(Ca)	2.71E+01(4.11E+00) 2	3.10E+01(2.52E+00) +/3	3.35E+01(6.94E+00) +/3	4.44E+01(4.30E+00) +/4	4.19E+01(5.69E+00) +/4	3.14E+01(7.51E+00) +/3	2.15E+01(1.46E+00) -/1
	+/-	9/4/2	9/4/2	11/2/2	10/3/2	11/3/1	7/6/2
Avg-rank	1.66	2.86	2.73	3.73	3.06	3.20	2.40

“+”, “≈”, “-” respectively denote that the performance of Search Manager is better than, similar to, and worse than the corresponding algorithm.

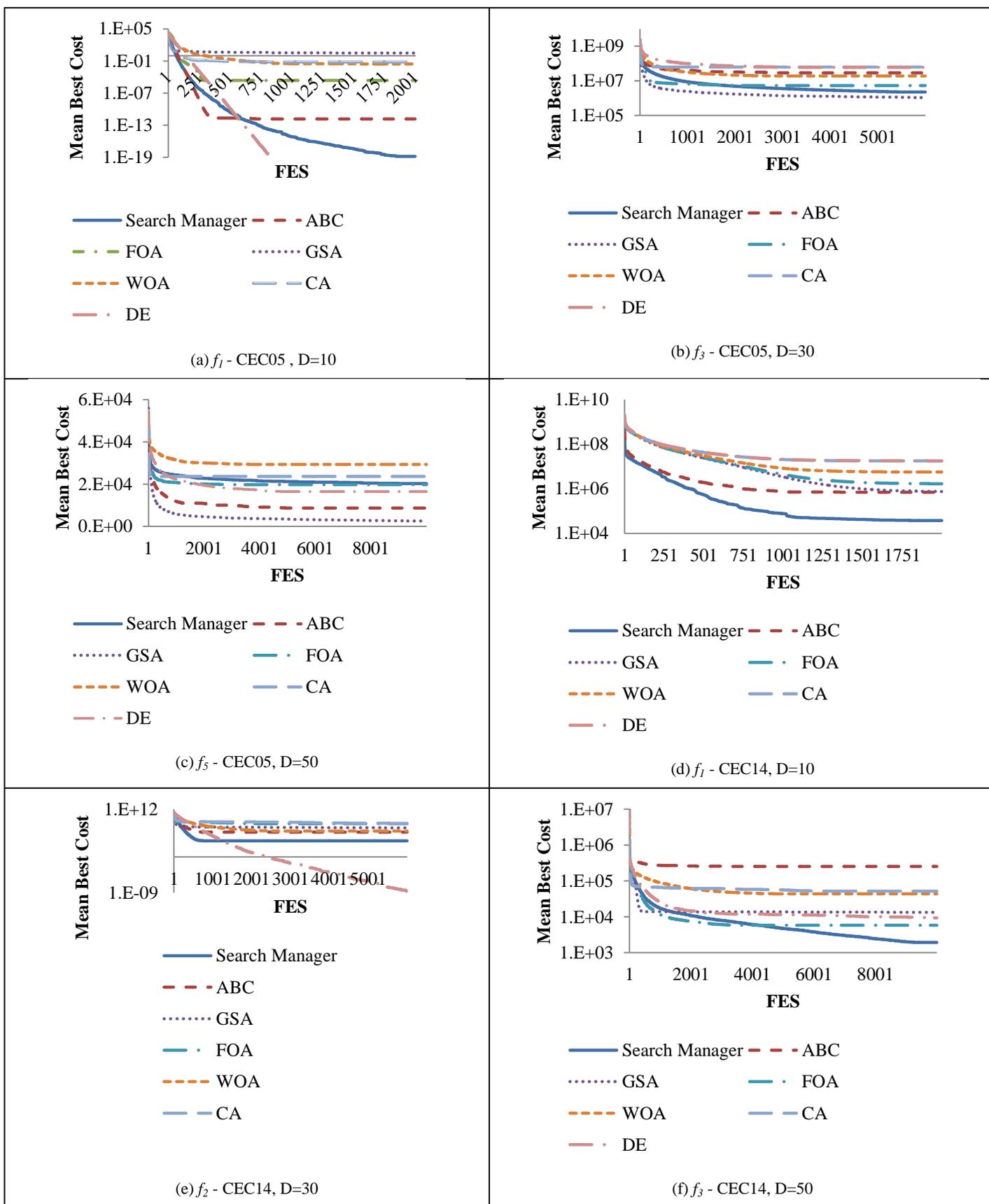


Fig. 8. Convergence curves of the algorithms on unimodal benchmark functions.

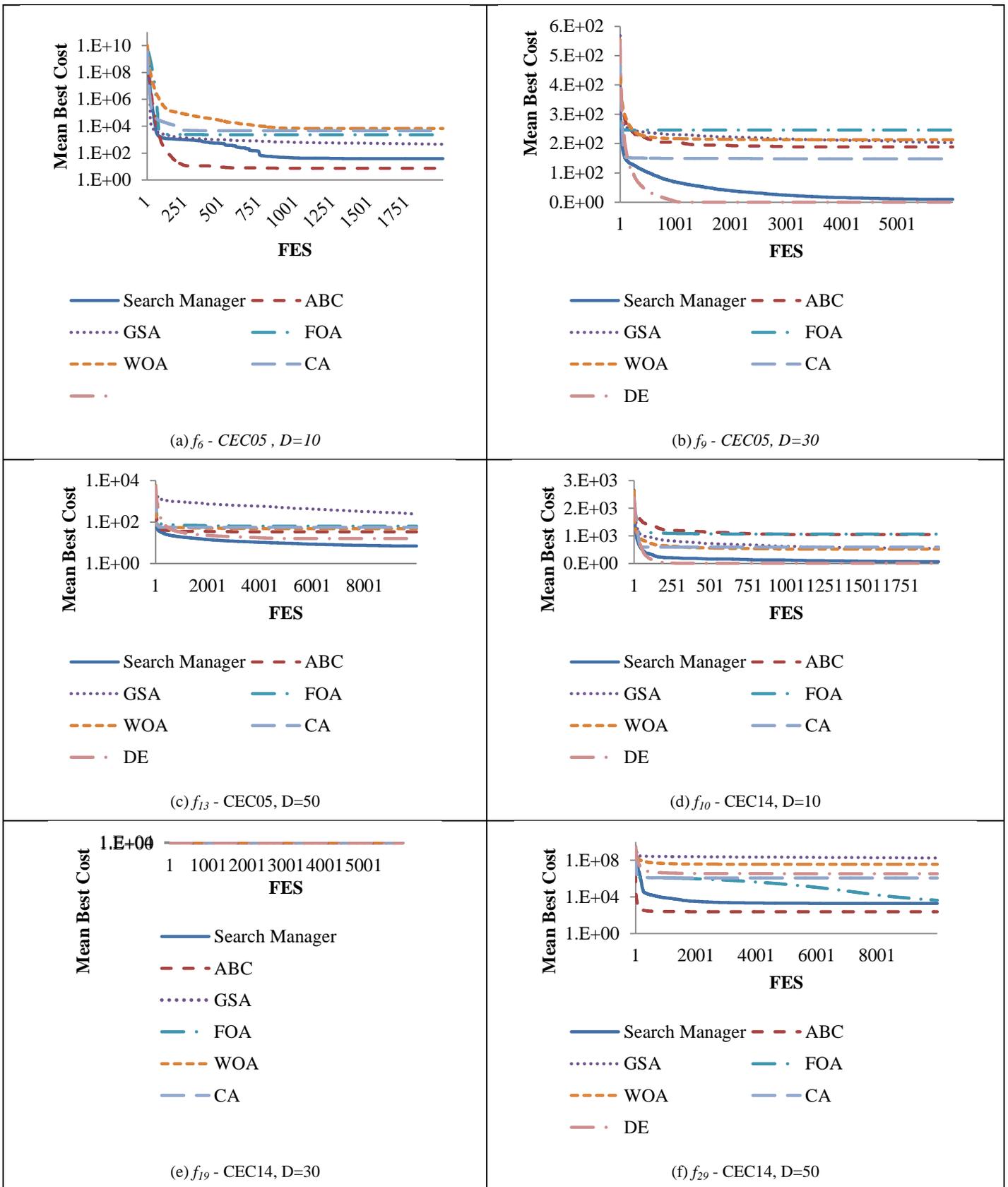


Fig. 9. Convergence curves of the algorithms on multimodal benchmark functions.