PAPER • OPEN ACCESS

Survey on Multi-Objective Evolutionary Algorithms

To cite this article: Wenlan Huang *et al* 2019 *J. Phys.: Conf. Ser.* **1288** 012057

IOP Conf. Series: Journal of Physics: Conf. Series **1288** (2019) 012057 doi:10.1088/1742-6596/1288/1/012057

Survey on Multi-Objective Evolutionary Algorithms

Wenlan Huang[1, a], Yu Zhang[1, 2, b] and Lan Li[1, c]

[1] School of Computer and Information Engineering, Harbin University of Commerce Harbin, 150028, China
[2] Heilongjiang Provincial Key Laboratory of Electronic Commerce and Information Processing, Harbin University of Commerce Harbin, 150028, China
E-mail: [a] hwl0601@163.com; [b] zhy_0423@163.com; [c] lilan9339cn@sina.com

Abstract. Multi-objective evolutionary algorithm (MOEA) is the main method to solve multi-objective optimization problem (MOP), which has become one of the hottest research areas of evolutionary computation. This paper surveys the development of MOEA and its research status, classifies it into four categories, analyzes the advantages and disadvantages of these algorithms, and summarizes the main application fields of MOEA. Finally several viewpoints for the future research of MOEA are presented.

1. Introduction

Most practical optimization problems need to optimize multiple mutually conflicting objectives simultaneously, take the problem of minimizing all objectives as an example, a multi-objective optimization problem can be represented as follows[1].

$$\begin{aligned} &\text{minimize} \quad \mathrm{F}(\boldsymbol{x}) = \left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_m(\boldsymbol{x})\right)^T \\ &\text{subject to:} \quad \boldsymbol{x} \in \Omega \end{aligned} \tag{1}$$

where Ω is the decision space and $\boldsymbol{x} \in \Omega$ is a decision vector,$\Omega \subseteq \mathrm{R}^n$. R^m is the objective space and $\mathbf{y} \in \mathrm{R}^m$ is a objective vector, $\mathbf{y} = \mathrm{F}(\boldsymbol{x})$: $\Omega \rightarrow \mathrm{R}^m$, where m is the number of objective functions.

Due to the conflicts among objectives, a MOP usually does not have a single optimal solution for all objectives but tradeoff optimal solutions known as Pareto-optimal (P-O) solutions [1]. Early work focused on a priori methods [2][3]. Classical methods transform the MOP to a single-objective optimization problem by constructing aggregation functions and obtaining one P-O solution at a time. The most common methods are weighted sum approach, Tchebycheff approach, goal programming approach and Min-Max approach. The disadvantages are that the weights corresponding to the objective functions are difficult to give properly, it is possible to find different P-O solutions after running multiple times, the Pareto-front (PF) concaves cannot be processed well. On the one hand, due to the discontinuity of the PF, and the complexity of the objective, it is difficult to obtain exact solutions in finite time. On the other hand, it is easier to get approximate solutions that meet the practical needs. So people focus more on getting a set of approximate P-O solutions. Evolutionary algorithm (EA) is very suitable for solving such problem, and provide an equilibrium solution set.

EA is a heuristic search algorithm, which has been successfully applied in the field of multi-objective optimization [4], and these EAs are called MOEAs. Population-based search and information exchange among individuals are the two characteristics of EA. It can obtain multiple P-O solutions in a single simulation run and do not have to know the derivative information or aggregate different properties, which effectively overcomes the limitations of the classical method. The three goals [5] of an MOEA are: (1) Convergence: to find a set of solutions as close as possible to the PF; (2) Diversity: to find a well distributed set of solutions; and (3) Coverage: to cover the entire PF.

The remainder of this paper is organized as follows. Section 2 summarizes the development and research status of MOEA. Section 3 discusses the main applications of MOEA. Section 4 presents the directions for future research. Section 5 makes the conclusions.

2. Multi-objective Evolutionary Algorithm

In recent decades, multi-objective evolutionary algorithms have developed rapidly and are roughly divided into four phases in table 1.

Table 1. The development phases of MOEAs

Phase	Typical Algorithm
First phase	VEGA, etc.
Second phase	MOGA, NSGA, NPGA, etc.
Third phase	SPEA, PAES, PESA, PESA-II , NPGA, NPGA2, Micro-GA , SPEA2 , NSGA-II, etc.
Fourth phase	MOEA/D, NSGA-III , MOPSO, RMEDA, IBEA, SMS-EMOA, etc.

The first phase in table1 is before the 1990s, In 1967, Rosenberg [6] proposed solving MOP with the genetic search, but he did not implement it. In 1985, Schaffer [7] proposed VEGA, which pioneered the use of EA to deal with MOPs. In 1989, Goldberg proposed combining Pareto theory with EAs [8].

The second phase in table1 is in the early 1990s, the concept of Pareto dominance was incorporated into these algorithms. Fonseca et al. proposed MOGA [9], Srinivas and Deb proposed NSGA [10], Horn and Nafpliotis proposed NPGA [11]. What needs to be improved during this period is how to reduce computational complexity and find better diversity preservation strategy.

The third phase in table1 is from 1999 to 2002, algorithms in this period adopted the elite preservation concept and better diversity preservation strategies. Some of the most prominent studies are SPEA [12] proposed by Zitzler and Thiele, PAES [13], PESA [14] and PESA-II [15] proposed by Knowles and Corne, Erichson et al. proposed NPGA, NPGA2 [16], Micro-GA [17] proposed by Coello and Pulido. SPEA2 [18] proposed by Zitzler et al. and NSGA-II [19] proposed by Deb et al. are the most representative algorithms. Many algorithms during this period dominated the field of MOEA for many years, but are still not suitable for many-objective optimization problems.

The fourth phase in table1 started from 2003, it has presented a more diversified development: (1) the indicator-based and decomposition-based MOEA have been widely used in many-objective optimization problems. NSGA-III [20] proposed by Deb et al. has drawn wide attention. (2) Introduce some new evolutionary strategies, such as MOPSO [21]. (3) Research on environmental selection operators, such as ε-domination principle [22]and μ-domination principle [23]. (4) Many scholars have successively designed more efficient benchmark problems, such as ZDT [24], DTLZ [25] proposed by Deb et al., Huband et al. [26][27] proposed WFG. (5) The research on MOEAs combined with machine learning technology is also vigorously carried out. For example, Zhang Q et al. proposed a regularity model-based multi-objective estimation of distribution algorithm (RMEDA) [28], MOEAs based on opposition-based learning [29].

Recently, some research has been carried out in two directions: (1) to establish a more confident termination of a MOEA in guaranteeing the convergence of solutions close to their Pareto optimality [30]. (2) Interactive MOEA, which integrates decision making procedures into MOEA and directly produces preferred optimal solutions.

According to the characteristics of current MOEAs, they can be roughly divided into four categories:

(1) Domination-based MOEAs. Individuals are assigned fitness based on Pareto-dominance principle to achieve good convergence, and an explicit diversity preservation scheme is provided to maintain the diversity of solutions.

NSGA-II [19] uses fast non-dominated sorting to classify individuals and reduces time complexity. The crowded-comparison operator without specifying sharing parameter replaces the fitness sharing mechanism. The elite preservation strategy of (μ+λ)[31] is introduced, which parents and good genes are competing to produce the next generation. But most of the comparisons between solutions in non-dominated sorting are redundant or unnecessary. Researchers have proposed a variety of faster methods based on it, such as deductive sort [32], and efficient non-dominated sort [33], etc. Density information will work to maintain diversity when different individuals have the same nondomination rank. Density estimation techniques include niche technology [11], crowding distance [19], K-nearest neighbor [18], and hyperbox [15]. SPEA2 [18] incorporates an individual's density value which is the density estimate determined by the individual's distance to its kth nearest neighbor into the fitness assignment. SPEA2 simplifies the cluster-based external population updating method in SPEA. Although the time complexity is still $O(M^3)$, the distribution uniformity of the obtained solution is better.

There are still some disadvantages in domination-based MOEAs. The slow convergence speed to PF, the poor performance on MOPs with complex PF, and the true convergence behavior of each non-dominated solutions from the Pareto optimal set is unknown. Recently, Deb et al. [34] developed a KKT Proximity Measure (KKTPM) for estimating proximity of a solution from Pareto optimal set. The Pareto-dominated MOEAs are not suitable for many-objective optimization. Because as the number of objective increases, the proportion of non-dominated solutions increases exponentially, thereby reducing the selection pressure and slowing down the evolutionary process, diversity and convergence are all impaired. In recent years, a variety of new domination principles have been proposed, the main purpose is to relax the comparison criteria, so that feasible solutions can be compared with each other, such as: ε-domination principle [22], μ-domination principle [23], etc. The disadvantage is that they are too dependent on parameters, and although the diversity and convergence are improved, the final non-dominated solutions are not guaranteed to cover the entire PF uniformly.

NSGA-III [20] is an effective algorithm to deal with many-objective optimization problems._Its' basic framework remains similar to the original NSGA-II algorithm. The biggest change is using well-distributed reference points to maintain diversity. In addition to emphasizing the dominance relationship, the number of individuals associated with each reference point is also emphasized. When the reference points are evenly distributed over the entire hyper-plane, the resulting solutions may be evenly distributed and close to the PF. However, due to the low selection pressure of Pareto dominance, its convergence still needs to be improved.

(2) Indicator-based MOEAs, The performance indicators of solution quality measurement are integrated into MOEA as the selection criteria of environmental selection, guide the search and continually optimize the expected attributes of the entire population.

The first well known indicator-based MOEA is IBEA [35], proposed by Zitzler and Kunzl in 2004, which defines the decision maker's preference as a binary indicator to compare the relative quality of two approximate solution sets. This indicator is subject to the Pareto rule, so the fitness calculation can be performed as the Pareto-based fitness assignment strategy. It provides a general framework for indicator-based MOEAs, allowing future studies to go along this direction.

SMS-EMOA [36] uses the HV indicator which measures how much objective space is dominated by the approximate solution set as the selection criterion, discarding solutions which have the smallest HV on the worst front, avoiding the large computational cost caused by the comparisons based on Pareto dominance. Bader and Zitzler [37] proposed a fast HV-based evolutionary algorithm named HypE to reduce excessive computational cost. HV indicator is very attractive in practical problems because it does not need to know PF and contains approximation, diversity and correlation informations, and finally evaluates the quality of the approximate solution set [38]. Of course, there is still a lot of research work to be done on solving the computational cost of HV.

Some studies have shown that replacing other indicators with HV indicators has less computational cost and equally good theoretical properties. For example, Trautmann et al. proposed the R2 indicator [39] to solve MOPs. Ye Tian et al. proposed an enhanced generation distance indicator named IGD-NS to distinguish solutions that do not contribute to IGD indicator [40].

The indicator-based algorithm has many advantages, but its higher selection pressure than Pareto domination makes solutions prefer some specific regions of PF, causing some optimal solutions to disappear during evolutionary process, then final solutions may not be evenly distributed along the PF.

(3) Decomposition-based MOEAs. Although the decomposition method is widely used in traditional mathematical programming methods to solve MOPs. Most MOEAs only regard MOP as a whole and rely on domination to measure the quality of the solution, and then these solutions may not be uniformly distributed over PF. The fitness evaluation based on scalar function has scalability to the number of objectives and strong search ability, its' computational complexity does not increase exponentially with the increase of objectives. Therefore, many decomposition-based MOEAs have been proposed.

The most representative decomposition-based MOEA is MOEA/D proposed by Zhang and Li in 2007 [41]. Various traditional single-objective optimization methods and local search methods can be applied to MOEA/D framework. The basic idea is using the decomposition strategy to decompose a MOP into a number of scalar optimization subproblems. Solutions of each subproblem are optimized by performing evolutionary operations among its several neighboring subproblems, and these subproblems are organized organically and solved simultaneously. Neighbor structures between subproblems are defined based on the distance between their weight vectors. The computational complexity of MOEA/D is O (MNT), T is the number of weight vectors, which is lower than the computational complexity O (MN^2) of NSGA-II. Commonly used decomposition methods are Weighted Sum Approach [42], Tchebycheff Approach [42], and Penalty-Based Boundary Intersection Approach [43].

MOEA/D still has many limitations. For example, the weight vectors are generated using the simplex lattice design method [43]. The disadvantages are that the population size increases nonlinearly with the increase of the number of objectives and cannot be set arbitrarily, and weight vector distribution is not very uniform for three or more objectives. In addition, when the PF is unknown or highly irregular, even a uniformly distributed weight vector cannot guarantee that the solutions are uniformly distributed on the PF. MOEA/D is difficult to choose a suitable decomposition method for different problems.

In recent years, There are many researches and application works in the framework of MOEA/D: as some new weight vector generation methods have been proposed, the MOEA/D-AWA [44] proposed by Qi et al. uses an adaptive weight vector adjustment (AWA) strategy for MOPs with complex PFs. Developing a new decomposition strategy, Ishibuchi et al. [45] proposed an approach based on combining different scalarizing functions within the MOEA/D framework; A more reasonable computational resource allocation, Zhang et al. proposed MOEA/D-DRA [46].

(4) Hybrid MOEAs: It is natural to combine the characteristics of different algorithms and utilize their advantages to deal with complicated MOPs, thus proposing a series of hybrid MOEAs. The two most major problems in hybrid algorithms are: What techniques to use and how to hybridize them [47].

Hybridizing different search methods. For example, combining global search and local search methods, such as introducing a local search method into EAs to locally improve the individuals generated by the common EA operators, which is good for obtaining better non-dominated solutions.

Dividing the search process into different phases and using different search strategies. For example, Yang [48] et al. use different search strategies in different phases to emphasize dominated solutions, to balance dominated and non-dominated solutions, and to focus on non-dominated solutions, respectively.

Combining hybrid MOEAs design idea with the new evolutionary model, Xie et al. proposed an enhanced multi-objective fireworks optimization algorithm [49].

Most of the current research focuses on how to combine different techniques to generate offspring population. But research in other areas is not too much.

3. Applications of MOEAs

With the rapid development of MOEAs, researchers in different scientific and engineering fields have applied MOEAs to solve optimization problems in their fields. The main applications are as follows.

(1) Application in scheduling strategy and manufacturing. Production scheduling can lead to complex combinatorial optimization problems and most of them are NP-hard. Multi-objective evolutionary algorithms are used to solve real time event in flexible job shop scheduling problem [50]. In traffic engineering and transportation, Vehicle scheduling of an urban bus line [51].

(2) Application in machine learning or deep learning. Such as classification of financial data [52]. Gong et al. presents a MO-SFL model for deep neural networks. [53].

(3) Application in pattern recognition and image processing. Such as multi-objective evolutionary algorithms for unsupervised change detection in multispectral landsat images [54].

(4) Application in assignment and management. Such as solving multi-objective water management problems [55].

(5) Application in circuit and communication. Such as wireless sensing network, circuit design etc., for example, using MOEA to optimize the topology and coverage control of wireless sensor networks [56].

(6) Application in bioinformatics. Such as molecular docking, DNA sequence design, gene networks, etc., For example, applying NSGA-II to the problem of identifying appropriate locations in cytochrome oxidase I (COI) gene for species- or group-specific primer design[57].

(7) Application in control systems and robotics. Greenhouse control, robot motion planning, control scheme design, etc., for example, using multi-objective evolutionary algorithms to solve problems in navigation of humanoid robots [58].

(8) Application in the financial field. For example, Sun proposed to optimizes the Markowitz extension multi-objective optimization model using the NSGA-II algorithm [59].

(9) Applications in other fields. Such as life sciences, fault diagnosis, etc.

4. Future Directions

Each generation of algorithms is limited by the level of research at that time. The current multi-objective evolutionary algorithms are still in the early stage, and there are still many problems that need to be studied.

(1) Many-objective optimization. It is necessary to study which problems are feasible by classifying many-objective optimization problems, this is crucial for any practical progress in the field. Researchers can also consider how to extend the original excellent algorithms to many-objective situation, or to develop new algorithm framework.

(2) A unified algorithm framework should be designed for different dimensional objectives. Now, most algorithms are only effective for optimization problems with specific objective number. Researchers should propose more unified evolutionary optimization algorithm.

(3) Interactive MOEA, which interacts with the decision maker's preferences during evolutionary process, is a very important research avenue.

(4) The design of each stage algorithm has certain limitations, and many seemingly perfect theories still need to be improved. Researchers should conduct in-depth theoretical analysis of some popular frameworks and study their advantages and disadvantages to improve them or develop some new algorithm frameworks. The Pareto-dominance principle is very popular, but it may not be suitable for all problems, researchers should consider introducing some new dominance principles.

(5) Some shortcomings of MOEAs are related to robustness and parameterization issues. They usually require a large number of function estimates and multiple runs to find an approximate optimal solution. In addition, parameters of these algorithms need to be well adjusted. For large and complex design problems, researchers should study how to further reduce the computational cost and a better adaptive control method for parameter adjustment.

(6) Studying the characteristics of solutions, such as similarities among them, and which design changes will affect the trade-off between objectives. The data mining method can be used to discover the knowledge of the solution obtained by multi-objective optimization, so as to have a deeper understanding of the problem. For example, the degree of approximation of the solution relative to the P-O solution can be studied to improve the convergence of the algorithm.

(7) Dynamic and noisy multi-objective optimization, although some work has been done on this, some fundamental issues have not been studied well yet.

(8) Quantum evolutionary algorithms should be studied in the field of multi-objective and constrained optimization [60]. Researchers should design effective quantum evolutionary algorithm and carry out corresponding theoretical research.

(9) An important issue in the study of multi-objective evolutionary algorithms is how to define the common characteristics of practical multi-objective problems, and to turn these common characteristics into special problems. Therefore, it is necessary to study and design some benchmark problems that better reflect the basic characteristics of practical multi-objective problem.

5. Conclusion

This paper presents a comprehensive survey of research works on MOEA. It introduces MOP and its mathematical model, describes the research status of MOEA and its four categories, analyzes the advantages and disadvantages of various algorithms, and summarizes the application fields of MOEA. And finally, the further research trends of multi-objective evolutionary algorithms are given to better carry out research on MOP.

6. Acknowledgement

This research was financially supported by the Youth Reserve Talents Foundation of Harbin Science and Technology Bureau under Grant No.2017RAQXJ120 and the Education Reform of Heilongjiang Province under Grant No. HSDJY2015025.

7. Reference

[1] Deb K 2001 *Multi-Objective Optimization Using Evolutionary Algorithms* (John Wiley & Sons) pp 124-124
[2] Nagar A, Heragu S S. and Haddock J 1995 *J. Oper. Res. Soc* **46** 721-734
[3] Lee C E and Chou F D 1998 *Int. J. Ind. Eng-Theory* **5** 128-139
[4] Gong M, et al. 2009 *J. Sw* **20** 271-289
[5] Trivedi A, et al. 2017 *IEEE. T. Evolut. Comput* **21** 440-462
[6] Rosenberg R S 1967 *Simulation of Genetic Populations with Biochemical Properties* (Ann Harbor, Michigan, USA: University of Michigan)
[7] Schaffer J D 1985 *Proc. 1st. Int. Conf. on Genetic Algorithms* (L. Erlbaum Associates, Hillsdale, New Jersey, USA) pp 93–100
[8] Goldberg D E 1989 *Genetic Algorithm: in Search, Optimization, and Machine Learning* (Addison-Wesley Publishing Company, Menlo Park, California, USA)
[9] Fonseca C M 1993 *Proc. Fifth. Int. Conf on Genetic Algorithms* (Morgan Kaufmann Publishers, San Francisco, California, USA) pp 416–423
[10] Srinivas N and Deb K 1994 *Evol. Comput* **2** 221-248
[11] Horn J, Nafpliotis N and Goldberg D E. 1994 *Proc. 1st. IEEE. Conf. on Evolutionary Computation* IEEE (New York USA) pp 82–87
[12] Zitzler E and Thiele L 1999 *IEEE. T. Evolut. Comput* **3** 257-271
[13] Knowles J D and Corne D W 2000 *Evol. Comput* **8** 149-172
[14] Corne D W, Knowles J D and Oates M J 2000, *Proc. 6th. Int. Conf. on Parallel Problem Solving from Nature* (Springer-Verlag Heidelberg Berlin Germany) pp 839-848
[15] Corne D W, Jerram N R and Knowles J D, et al. 2001 *Proc. 2nd. Annual Genetic and Evolutionary Computation Conf.* (Morgan Kaufmann Publishers San Francisco California USA) pp 283-290
[16] Erickson M, Mayer A and Horn J 2001 *Proc. 1st. Int. Conf. on Evolutionary Multi-Criterion Optimization* (Springer Heidelberg Berlin Germany) pp 681-695
[17] Coello C A C and Pulido G T 2001 *Proc. 1st. Int. Conf. on Evolutionary Multi-Criterion Optimization* (Springer, Heidelberg, Berlin, Germany) pp 126-140.
[18] Zitzler E, Laumanns M and Thiele L 2001 *TIK-repor*t 103.
[19] Deb K, Pratap A, Agarwal S, et al. 2002 *IEEE T. Evolut. Comput* **6** 182-197.
[20] Deb K and Jain H 2014 *IEEE T. Evolut. Comput* **18** 577-601.

IOP Conf. Series: Journal of Physics: Conf. Series **1288** (2019) 012057 doi:10.1088/1742-6596/1288/1/012057

[21] Sierra M R and Coello C A C 2005 *Proc. 3rd. Int. Conf. on Evolutionary Multi-Criterion Optimization* (Springer, Heidelberg, Berlin, Germany) pp 505-519.
[22] Laumanns M, Thiele L, Deb K, et al. 2002 *Evol. Comput* **10** 263-282.
[23] Koduru P, Das S and Welch S M 2007 *Proc. 9th. Annual Genetic and Evolutionary Computation Conf.*, ACM (New York, USA) pp 853–860.
[24] Zitzler E, Deb K, Thiele L 2000 *Evol. Comput* **8** 173-195.
[25] Deb K, Thiele L, Laumanns M, et al. *Proc. of the 2002 Cong. on Evolutionary Computation. IEEE* pp 825-830.
[26] Huband S, Barone L, While L, et al. 2005 *Lec Notes Comput Sc* **3410** 280-295.
[27] Huband S, Hingston P, Barone L, and While L 2006 *IEEE Trans. Evol. Comput* **10** 4 77–506.
[28] Zhang Q., Zhou A., Jin Y. 2008 *IEEE Trans. Evol. Comput* **12** 41-63.
[29] X Ma, F Liu, Y Qi, M Gong, et al. 2014 *Neurocomputing* **146** 48-64.
[30] Abouhawwash M, Seada H, Deb K 2017 *Comput. Oper. Res* **79** 331-346.
[31] Schwefel H P, et al. 1995 *Evolution and optimum seeking* (Wiley, New York)
[32] Mcclymont K and Keedwell E 2012 *Evol. Comput* **20** 1–26.
[33] Zhang X, Tian Y, Cheng R, et al. 2015 *IEEE Trans. Evol. Comput* **19** 201-213.
[34] Deb K and Abouhawwash M 2016 *IEEE Trans. Evol. Comput* **20** 515-528.
[35] Zitzler E, and Künzli S 2004 *Proc. 8th. Int. Conf. on Parallel Problem Solving from Nature* vol 3242 (Springer, Heidelberg, Berlin, Germany) pp 832-842.
[36] Beume N, Naujoks B and Emmerich M *Eur. J. Oper. Res* **181** 1653-1669.
[37] Bader J and Zitzler E 2014 *Evol. Comput* **19** 45-76.
[38] Helbig M and Engelbrecht A P 2013 *Inform Sciences* **250** 61–81.
[39] Trautmann H, Wagner T, Brockhoff D 2013 *Int. Conf. on Learning and Intelligent Optimization* (Springer, Heidelberg, Berlin, Germany) pp 70-74.
[40] Tian Y, Cheng R, Zhang X and Cheng F 2018 *IEEE Trans. Evol. Comput* **22** 609-622
[41] Zhang Q and Li H 2007 *IEEE Trans. Evol. Comput* **11** 712-731.
[42] Miettinen K 1999 *J. Oper. Res. Soc* **51** 246.
[43] Das I, Dennis J E 1998 *Siam. J. Optimiz* **8** 631-657.
[44] Qi Y, Ma X, Liu F, et al. 2014 *Evol. Comput* **22** 231-264.
[45] Ishibuchi H, Sakane Y, Tsukamoto N and Nojima Y 2010 Proc. *12th. annual conf. on Genetic and evolutionary computation*. ACM pp 519-526.
[46] Zhang Q, Liu W, and Li H 2009 *Proc. IEEE Congr. Evol. Comput. (CEC)*, (Trondheim, Norway) pp 203–208.
[47] Zhou A, Qu B, Li H, et al. 2011 *Evol. Comput* **1** 32-49.
[48] Yang D and Jiao L, Gong M 2009 *Comput. Intell* **25** 84–108.
[49] Xie C and Xu L, et al. 2017 *Acta electronica sinica.* **45** 2323-2331.
[50] Sreekara Reddy M B S, et al. 2018 *Measurement.* **114** 78-90.
[51] Zuo C, Chen C, Tan W, et al. 2015 *IEEE T. Intell Transp* **16** 1030-1041.
[52] Antonelli M, Bernardo D, Hagras H et al. 2017 *IEEE T. Fuzzy. Syst* **25** 249-264.
[53] Gong M, Liu J, Li H, et al. 2017 *IEEE T. Neur. Net. Lear* **26** 3263-3277.
[54] Yavariabdi A and Kusetogullari H 2017 *IEEE Geosci. Remote. S* **14** 414-418.
[55] Lewis A and Randall M 2017 *J. Environ. Manage* **204** 179-188.
[56] Jameii S M, Faez K and Dehghan M 2015 *Int. J. Distrib. Sens. N* **2015** 1-11.
[57] Jeong K S, Kim M H, Jo H, et al. 2015 *Ecol. Inform* **29** 214-220.
[58] Lee K B, Myung H and Kim J H 2015 *IEEE T. Ind. Electron.* **62** 5586-5597.
[59] Sun X 2013 *Computer Engineering & Applications* **49** 267-270.
[60] Wang L 2008 *Control and Decision.* **23** 1321-1326.