

Evolutionary Computation and Big Data: Key Challenges and Future Directions

Shi Cheng¹(✉), Bin Liu², Yuhui Shi³, Yaochu Jin⁴, and Bin Li⁵

¹ School of Computer Science, Shaanxi Normal University, Xi'an, China
cheng@snnu.edu.cn

² School of Computer Science and Technology,
Nanjing University of Posts and Telecommunications, Nanjing, China
bins@ieee.org

³ Department of Electrical and Electronic Engineering,
Xi'an Jiaotong-Liverpool University, Suzhou, China
yuhui.shi@xjtlu.edu.cn

⁴ Department of Computing, University of Surrey, Guildford, Surrey, UK
yaochu.jin@surrey.ac.uk

⁵ University of Science and Technology of China, Hefei, China
binli@ustc.edu.cn

Abstract. Over the past few years, big data analytics has received increasing attention in all most all scientific research fields. This paper discusses the synergies between big data and evolutionary computation (EC) algorithms, including swarm intelligence and evolutionary algorithms. We will discuss the combination of big data analytics and EC algorithms, such as the application of EC algorithms to solving big data analysis problems and the use of data analysis methods for designing new EC algorithms or improving the performance of EC algorithms. Based on the combination of EC algorithms and data mining techniques, we understand better the insights of data analytics, and design more efficient algorithms to solve real-world big data analytics problems. Also, the weakness and strength of EC algorithms could be analyzed via the data analytics along the optimization process, a crucial entity in EC algorithms. Key challenges and future directions in combining big data and EC algorithms are discussed.

Keywords: Big data analytics · Data science · Evolutionary algorithms · Evolutionary computation · Swarm intelligence

1 Introduction

Nowadays, big data has been attracting increasing attention from academia, industry and government [2, 20, 36]. Big data is defined as the dataset whose size is beyond the processing ability of traditional databases or computers. Four elements are emphasized in the definition, which are capture, storage, management, and analysis [26]. The focus of the four elements is the last stage, the big

data analytics, which is about automatic extraction of knowledge from a large amount of data.

Big data analysis can be seen as mining or processing of massive data, thereby retrieving “useful” information from large dataset [29]. Big data analytics can be characterized by several properties, such as large volume, variety of different sources, and fast increasing speed (velocity) [26]. It is of great interest to investigate the role of evolutionary computation (EC) techniques, including evolutionary algorithms and swarm intelligence for the optimization and learning involving big data, in particular, the ability of EC techniques to solve large scale, dynamic, and sometimes multiobjective big data analytics problems.

Traditional methods for data analysis are based mainly on mathematical models and data is then collected to fit the models. With the growth of the variety of temporal data, these mathematical models may become ineffective in solving problems. The paradigm should shift from the model-driven to the data-driven approach. The data-driven approach not only focuses on predicting what is going to happen, but also concentrates on what is happening right now and how to be prepared for future events.

With the amount of data growing constantly and exponentially, the current data processing tasks are beyond the computing ability of traditional computational models. The data science, or more specifically, the big data analytics, has received more and more attention from researchers. The data are easily generated and gathered, while the volume of data is increasing very quickly. It exceeds the computational capacity of current systems to validate, analyze, visualize, store, and extract information. To analyze these massive data, there are several kinds of difficulties, such as the large volume of data, dynamical changes of data, data noise, *etc.* New and efficient algorithms should be designed to handle massive data analytics problems.

Evolutionary computation (EC) algorithms, which include swarm intelligence and evolutionary algorithms, are a set of search and optimization techniques [13, 14, 22]. To search a problem domain, an EC algorithm processes a population of individuals. Different from traditional single-point based algorithms such as hill-climbing algorithms, each EC algorithm is a population-based algorithm, which consists of a set of points (population of individuals). Each individual represents a potential solution to the problem being optimized. The population of individuals is expected to have high tendency to move towards better and better solution areas over iterations through cooperation and competition among themselves.

In this paper, we present the analysis of the relationship from data science to evolutionary computation algorithms, which include swarm intelligence and evolutionary algorithms. EC algorithms could be applied to optimize the data mining problems or to handle data directly. In evolutionary computation algorithms, individuals move through a solution space and search for solution(s) for the data mining task. The algorithm could be utilized to optimize the data mining problem, e.g., the parameter tuning. The EC algorithms could be directly applied to the data samples, e.g., subset data extraction. With the EC

algorithms, more effective methods can be designed and utilized in massive data analytics.

In evolutionary computation algorithms, solutions are spread in the search space. Each solution can also be considered as a data point in the search space; the distribution of solutions can be utilized to reveal the landscape of a problem. Data analysis techniques have been exploited to design new swarm intelligence/evolutionary algorithms, such as brain storm optimization algorithm [30,31] and estimation of distribution algorithms [17,28].

The aim of this paper is to analyze the association between big data analytics and evolutionary computation (EC) algorithms. The possible directions of utilizing evolutionary computation algorithms on data science and applying data science methods for EC algorithms will be discussed. The remaining of the paper is organized as follows. Section 2 reviews the basic concepts of big data analytics methods. Section 3 discusses the key challenges of the combination of EC algorithms and big data analytics. Section 4 analysis the future directions of EC algorithms utilized to optimize data science methods and data analysis methods utilized to analyze EC algorithms, followed by conclusions in Sect. 5.

2 Big Data Analytics

Currently, data science or big data analytics is a popular topic in computer science and statistics. It concerns with a wide variety of data processing tasks, such as data collection, data management, data analysis, data visualization, and real-world applications.

The data science is a fusion of computer science and statistics. The statistics is the study of the collection, analysis, interpretation, presentation, and organization of data [12]. From the perspective of statistics research, the data science has the same objectives as the statistics, except that the data science emphasizes more on volume, and the variety of data. The data science is more like a synonym of big data research. From the perspective of statistics, there are two aims in data analyses [12]:

- Prediction: To predict the response/output of future input variables;
- Inference: To deduce the association among response variables and input variables.

From the perspective of computer science research, the data science is more practical. The phrase “data mining” is often used to indicate the data science tasks. The process of converting raw data into useful information, termed as knowledge discovery in databases. Data mining, which is data analysis process of knowledge discovery, attempts to discover useful information (or patterns) in large data repositories [15].

The statistics and data science both focus on the study of the extraction of knowledge from data. The main difference is that the data in data science are increasingly heterogeneous and unstructured [10].

EC algorithms, which include evolutionary algorithms and swarm intelligence, are a set of search and optimization techniques [14, 22]. To search a problem domain, a swarm intelligence algorithm processes a population of individuals. Each swarm intelligence algorithm is a population-based algorithm, which consists of a set of points (individuals). Each individual represents a potential solution to the problem being optimized. The population of individuals is expected to have high tendency to move towards better and better solution areas over iterations through cooperation and competition among themselves.

3 Key Challenges

The aim of evolutionary computation and big data analytics is to combine the strengths of EC algorithms and data science techniques. It has two meanings: potential applications of evolutionary computation algorithms in the big data analytics and big data analytics techniques in enhancing evolutionary computation algorithms. With the data analytics during the optimization process, the relationship between the algorithm and problems could be revealed. The EC algorithms could be utilized to solve real-world big data analytics problems [6, 7].

3.1 Data-Driven Evolutionary Computation Algorithms

In general, most of the EC algorithms share a similar framework and usually involve the following two phases [18]:

1. Generate candidate solutions (e.g., random initialization) according to a specified probabilistic distribution.
2. Update the explicit or implicit model, on the basis of the information (solutions and their fitness values) collected in the previous/current step, to guide the future search toward “better” solutions.

This framework could obtain “good enough” solutions for problems with low dimensions or simple landscape. However, with the developments of evolutionary computation techniques, problems with more complex structures and large-scale data are arising in real-world applications. To handle the new challenges of optimization problem, more efficient and adaptive algorithms should be designed. For the traditional algorithms, there are several obstacles need to overcome for this framework:

1. Not much problem specific information is used: algorithms with the same parameters and structure are used to solve different kinds of problems. The problem’s information is not taken advantage of during the search process.
2. The algorithms need a balance between short and long time memory. There are two kinds of memories which are used in EC algorithms: the short time memory, e.g., the previous solutions (parent generation in genetic algorithm, previous position in particle swarm optimization algorithm) and the long time memory, e.g., the personal best position.

3. Only the fitness of objective function is used to guide the search. Normally, the individuals with the better fitness values have more possibility to reserve in next iteration, and other individuals are more likely to be abandoned during the search.

With the development of big data analytics techniques, more data could be stored and more unstructured information could be analyzed. The EC algorithms could be understood better and improved through the information analyses during the search. The paradigm in data-driven EC algorithms seems to be: collect a (massive) dataset, design an explicit or implicit model such as evolutionary algorithm or particle swarm optimization algorithm that can propagate search information from one individual to another, and finally converge to some good enough solutions until time runs out.

The meta-heuristics algorithms could be roughly divided into two categories: instance-based search and model-based search [18,37]. Most of the traditional search methods, such as simulated annealing and iterated local search, could be classified into instance-based search, which the new candidate solutions are generated using solely the current solution or the current group of solutions. For the recently meta-heuristics algorithms, such as ant colony optimization and estimation of distribution algorithms, could be classified as model-based search. In model-based search, candidate solutions are generated using an explicit or implicit model, which is updated using the information of previously solutions. The search is guided to concentrate on the regions containing high quality solutions iteration over iteration.

Figure 1 gives a framework of data-driven EC algorithms. Each candidate solution is a data sample from the search space. The model could be designed or adjusted via the data analysis on the previous solutions. The landscape or the difficulty of a problem could be obtained during the search, i.e., the problem could be understood better. With the learning process, more suitable algorithms could be designed to solve different problems, thus, the performance of optimization could be improved.

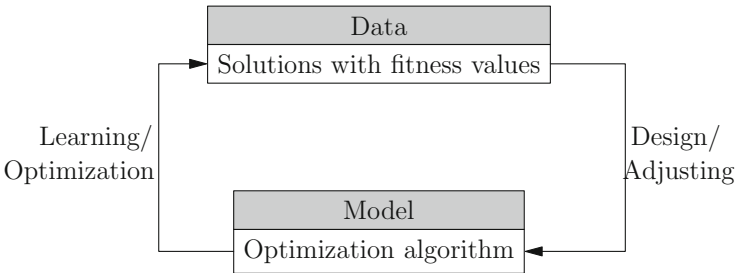


Fig. 1. A framework for data-driven evolutionary computation algorithms.

Massive information exists during the search process. For EC algorithms, there are several individuals existed at the same time, and each individual has a

corresponding fitness value. The individuals are created iteration over iteration. There is also massive volume of information on the “origin” of an individual, such as that an individual was created by applying which strategy and parameters to which former individual(s). The data-driven EC algorithm is a new approach to analyze and guide the search in evolutionary algorithms/swarm intelligence. These strategies could be divided into off-line methods and online methods. An off-line method is based on the analysis of previous storage search history, such as history based topological speciation for multimodal optimization [25] or maintaining and processing submodels (MAPS) based estimation of distribution algorithm on multimodal problems [32]. While for an online method, the parameters could be adaptively changed during the different search states.

3.2 EC on Solving Data Analytics Problems

The big data analytics is a new research area of information processing, however, the problems of big data analytics have been studied in other research fields for decades under a different title. The rough association between big data analytics and evolutionary computation algorithms can be established and shown in Table 1.

Table 1. The rough association between big data analytics and evolutionary computation algorithms.

Big data analytics	Evolutionary computation algorithms
Volume	Large scale/high dimension
Variety	
Velocity	Dynamic environment
Veracity	Noise/uncertain/surrogates
Value	Fitness/objective

The characteristics of the big data analytics are summarized into several words, which are volume, variety, velocity, veracity, and value. These complexities are a collection of different research problems that existed for decades. Corresponding to the EC algorithms, the volume and the variety mean large-scale and high dimensional data; the velocity means data is rapidly changing, like an optimization problem in dynamic environment; the veracity means data is inconsistent and/or incomplete, like an optimization problem with noise or approximation; and the value is the objective of the big data analytics, like the fitness or objective function in an optimization problem.

The big data analytics is an extension of data mining techniques on a large amount of data. Data mining has been a popular academic topic in computer science and statistics for decades. The swarm intelligence and evolutionary algorithms are subfields of evolutionary computation techniques which study the

collective intelligence in a group of simple individuals. Like data mining, in the evolutionary computation algorithms, useful information can be obtained from the competition and cooperation of individuals.

The key challenges of EC solving big data analytics problems could be divided into four elements: handling a large amount of data, handling high dimensional data, handling dynamical data, and multiobjective optimization. Most real world big data problems can be modeled as a large scale, dynamical, and multiobjective problems.

Handling Large Amount of Data. The big data analysis requires a fast mining on a large scale dataset, i.e., the immense amount of data should be processed in a limited time to reveal useful information. As the computing power improves, more volume of data can be processed. The more data are retrieved and processed, the better understanding of problems can be obtained.

The analytic problem can be modeled as an optimization problem. An evolutionary computation algorithm is a search process based on the previous experiences. To reveal knowledge from a large volume of data within the big data context, the search ranges of the solved problem have to be widened and even extended to the extreme.

A quick scan is critical to solve the problem with massive data sets. Evolutionary computation algorithms are techniques based on the sampling of the search space. Through the meta-heuristics rules, data samples are chosen from the massive data space. From these representative data samples, the problem structure could be obtained. Based on the evolutionary computation algorithms, we could find a “good enough” solution with a high search speed to solve the problem with a large volume of data.

A large amount of data does not necessarily mean high dimensional data, and a high volume of data can accumulate in single dimension such as high frequency data sampled by sensors with higher resolutions.

Handling High Dimensional Problems. In general, the optimization problem concerns with finding the best available solution(s) for a given problem within allowable time, and the problem may have several or numerous optimal solutions, of which many are local optimal solutions. Normally, the problem will become more difficult with the growth of the number of variables and objectives. Specially, problems with a large number of variables, e.g., more than a thousand variables, are termed as large scale problems.

Many optimization methods suffer from the “curse of dimensionality”, which implies that their performance deteriorates quickly as the dimension of the search space increases [3, 11, 16, 24]. There are several reasons that cause this phenomenon.

The solution space of a problem often increases exponentially with the problem dimension and thus more efficient search strategies are required to explore all promising regions within a given time budget. An evolutionary computation algorithm is based on the interaction of a group of solutions. The promising

regions or the landscape of problems are very difficult to reveal by small solution samples (compared with the number of all feasible solutions).

The characteristics of a problem may also change with the scale. The problem will become more difficult and complex when the dimension increases. Rosenbrock's function, for instance, is unimodal for two dimensional problems but becomes multimodal for higher dimensional problems. Because of such a worsening of the features of an optimization problem resulting from an increase in scale, a previously successful search strategy may no longer be capable of finding an optimal solution. Fortunately, an approximate result with a high speed may be better than an accurate result with a tardy speed. Evolutionary computation algorithms can find a good-enough solution rapidly, which is the strength of the EC algorithms in solving the big data analytics problems.

A data mining problem can be modeled as an optimization problem, and the research results of the large scale optimization problems can also be transferred to data mining problems. In evolutionary computation algorithms, many effective strategies are proposed for high dimensional optimization problems, such as problem decomposition and subcomponents cooperation [34], parameter adaptation [35], and surrogate-based fitness evaluations [21]. Especially, the particle swarm optimization or ant colony optimization algorithms can be used in the data mining to solve single objective [1] and multiobjective problems [9].

In the EC algorithms, the problem of handing a large amount of data and/or high dimensional data can be represented as large scale problems, i.e., problems with massive variables to be optimized. Based on the EC algorithms, an effective method could find good solutions for large scale problems, in terms of both the time complexity and the result accuracy.

Handling Dynamical Problems. The big data, such as the web usage data of the Internet and real time traffic information, rapidly changes over time. The analytical algorithms need to process these data swiftly. The dynamic problems are sometimes also termed as non-stationary environment [27] or uncertain environment [19] problems. The EC algorithms have been widely applied to solve both stationary and dynamical optimization problems [33].

The EC algorithms often have to deal with the optimization problems in the presence of a wide range of uncertainties. Generally, uncertainties in the problems can be divided into the following categories.

1. The fitness function or the processed data is noisy.
2. The design variables and/or the environmental parameters may change over the optimization process, and the quality of the obtained optimal solution should be robust against environmental changes or deviations from the optimal point.
3. The fitness function is approximated, such as surrogate-based fitness evaluations. The fitness function suffers from the approximation errors.
4. The optimum in the problem space may change over time. The algorithm should be able to track the optimum continuously.

5. The optimization target may change over time. The computing demands need to adjust to the dynamical environment. For example, there should be a balance between the computing efficiency and the power consumption for different computing loads.

In all these cases, additional measures must be taken so that the EC algorithms are still able to solve the dynamic problems satisfactorily [4, 19].

Handling Multiobjective Problems. A general multiobjective optimization problem (MOP) or a many objective optimization problem (MaOP) can be described as a vector function \mathbf{f} that maps a tuple of n parameters (decision variables) to a tuple of k objectives.

Different sources of data are integrated in the big data research, and for the majority of the big data analytics problems, more than one objective need to be satisfied at the same time. In a multiobjective optimization problem, we aim to find the set of optimal trade-off solutions known as the Pareto optimal set. Pareto optimality is defined with respect to the concept of nondominated points in the objective space. EC algorithms are particularly suitable to solve multiobjective optimization problems because they deal simultaneously with a set of possible solutions. This allows us to find an entire set of Pareto optimal set in a single run of the algorithm, instead of having to perform a series of separate runs as in the case of the traditional mathematical programming techniques [8, 23]. Additionally, EC algorithms are less susceptible to the shape or continuity of the Pareto front.

4 Future Directions

The future direction is combining the strengths of EC algorithms and big data analytics to design new algorithms on the optimization or data analytics.

4.1 EC Algorithms for Big Data Problems

The big data is created in many areas in our everyday life. The big data analytics problem not only occurs in Internet data mining, but also in complex engineering or design problems [5]. The big data problem could be analyzed from the perspective of computational intelligence and meta-heuristic global optimization [36]. A real-world application could be modeled as a multiobjective, dynamic, large scale optimization problem. It is recognized that the EC algorithms are good ways to handle this kind of problems. Based on the utilization of EC algorithms, the real-world system will be more efficient and effective [6, 7].

4.2 Big Data Analytics for EC Algorithms

A population of individuals in EC algorithms is utilized to evolve the optimized functions or goals by cooperative and competitive interaction among individuals.

Massive information exists during the search process, such as the distribution of individuals and the fitness of each solution. To improve the search efficiency or to recognize the search state, the data generated in the optimization process should be analyzed.

The following list gives some directions on the combination of big data analytics and evolutionary computation:

1. High-dimensional and many-objective evolutionary optimization;
2. Big data driven optimization of complex engineering systems;
3. Integrative analytics of diverse, structured and unstructured data;
4. Extracting new understanding from real-time, distributed, diverse and large-scale data resources;
5. Big data visualization and visual data analytics;
6. Scalable, incremental learning and understanding of big data;
7. Scalable learning techniques for big data;
8. Big data driven optimization of complex systems;
9. Human-computer interaction and collaboration in big data;
10. Big data and cloud computing;
11. Cross-connections of big data analysis and hardware;
12. GPU-based EC algorithms;
13. Big data techniques for business intelligence, finance, healthcare, bioinformatics, intelligent transportation, smart city, smart sensor networks, cyber security and other critical application areas;
14. MapReduce implementations combined with evolutionary computation algorithms approaches.

5 Conclusions

In evolutionary computation (EC) algorithms, a population of individuals is utilized to evolve the optimized functions or goals by cooperative and competitive interaction among individuals. Massive information exists during the search process, such as the distribution of individuals and the fitness of each solution. To improve the search efficiency or to recognize the search state, the data generated in the optimization process should be analyzed.

With the amount of data growing constantly and exponentially, the data processing tasks have been beyond the computing ability of traditional computational models. To handle these massive data, i.e., deal with the big data analytics problem, more effective and efficient methods should be designed. There is no complex mathematical model in evolutionary computation algorithms. The algorithm is updated based on few iterative rules and the evaluation of solution samples. The massive data analytics may be benefited from these properties because massive data are difficult or impossible to be represented by mathematical models.

In this paper, the connection between big data analytics and evolutionary computation algorithms was discussed. The potential applications of the EC algorithms in the big data analytics and the big data analytics techniques in EC

algorithms were analyzed. The big data analytics involves prediction or inference on a large amount of data. Most real world big data problems can be modeled as a large scale, dynamical, and multiobjective problems. EC algorithms study the collective behaviors in a group of individuals. With the combination of big data analytics and evolutionary computation algorithms, more rapid and effective methods can be designed to solve optimization and data analytics problem.

Acknowledgments. This work is partially supported by National Natural Science Foundation of China under Grant Numbers 60975080, 61273367, 61571238, and 61302158.

References

1. Abraham, A., Grosan, C., Ramos, V. (eds.): *Swarm Intelligence in Data Mining, Studies in Computational Intelligence*, vol. 34. Springer, Heidelberg (2006)
2. Alexander, F.J., Hoisie, A., Szalay, A.: Big data. *Comput. Sci. Eng.* **13**(6), 10–13 (2011)
3. Bellman, R.: *Adaptive Control Processes: A guided Tour*. Princeton University Press, Princeton (1961)
4. Bui, L.T., Michalewicz, Z., Parkinson, E., Abello, M.B.: Adaptation in dynamic environments: a case study in mission planning. *IEEE Trans. Evol. Comput.* **16**(2), 190–209 (2012)
5. Chai, T., Jin, Y., Sendhoff, B.: Evolutionary complex engineering optimization: opportunities and challenges. *IEEE Comput. Intell. Mag.* **8**(3), 12–15 (2013)
6. Cheng, S., Shi, Y., Qin, Q., Bai, R.: Swarm intelligence in big data analytics. In: Yin, H., Tang, K., Gao, Y., Klawonn, F., Lee, M., Weise, T., Li, B., Yao, X. (eds.) *IDEAL 2013. LNCS*, vol. 8206, pp. 417–426. Springer, Heidelberg (2013)
7. Cheng, S., Zhang, Q., Qin, Q.: Big data analytic with swarm intelligence. *Ind. Manag. Data Syst.* **116**(4) (2016, in press)
8. Coello, C.A.C., Lamont, G.B., Veldhuizen, D.A.V.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. Genetic and Evolutionary Computation Series, 2nd edn. Springer, New York (2007)
9. Coello, C.A.C., Dehuri, S., Ghosh, S. (eds.): *Swarm Intelligence for Multi-objective Problems in Data Mining, Studies in Computational Intelligence*, vol. 242. Springer, Heidelberg (2009)
10. Dhar, V.: Data science and prediction. *Commun. ACM* **56**(12), 64–73 (2013)
11. Domingos, P.: A few useful things to know about machine learning. *Commun. ACM* **55**(10), 78–87 (2012)
12. Donoho, D.L.: 50 years of data science. Technical report, Stanford University September 2015
13. Dorigo, M., Stützle, T.: *Ant Colony Optimization*. MIT Press, Cambridge (2004)
14. Eberhart, R., Shi, Y.: *Computational Intelligence: Concepts to Implementations*. Morgan Kaufmann Publisher, San Francisco (2007)
15. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery in databases. *AI Mag.* **17**(3), 37–54 (1996)
16. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2nd edn. Springer, New York (2009)

17. Hauschild, M., Pelikan, M.: An introduction and survey of estimation of distribution algorithms. *Swarm Evol. Comput.* **1**(3), 111–128 (2011)
18. Hu, J., Fu, M.C., Marcus, S.I.: A model reference adaptive search method for global optimization. *Oper. Res.* **55**(3), 549–568 (2007)
19. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments - a survey. *IEEE Trans. Evol. Comput.* **9**(3), 303–317 (2005)
20. Jin, Y., Hammer, B.: Computational intelligence in big data. *IEEE Comput. Intell. Mag.* **9**(3), 12–13 (2014)
21. Jin, Y., Sendhoff, B.: A systems approach to evolutionary multiobjective structural optimization and beyond. *IEEE Comput. Intell. Mag.* **4**(3), 62–76 (2009)
22. Kennedy, J., Eberhart, R., Shi, Y.: *Swarm Intelligence*. Morgan Kaufmann Publisher, San Francisco (2001)
23. Kim, Y.S.: Multi-objective clustering with data- and human-driven metrics. *J. Comput. Inf. Syst.* **51**(4), 64–73 (2011)
24. Lee, J.A., Verleysen, M.: *Nonlinear Dimensionality Reduction*. Information Science and Statistics. Springer, New York (2007)
25. Li, L., Tang, K.: History-based topological speciation for multimodal optimization. *IEEE Trans. Evol. Comput.* **19**(1), 136–150 (2015)
26. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H.: *Big data: the next frontier for innovation, competition, and productivity*. Technical report, McKinsey Global Institute, May 2011
27. Morrison, R.W., De Jong, K.A.: A test problem generator for non-stationary environments. In: *Proceedings of the 1999 Congress on Evolutionary Computation (CEC 1999)*, vol. 3, pp. 2047–2053, July 1999
28. Pelikan, M., Goldberg, D.E., Lobo, F.G.: A survey of optimization by building and using probabilistic models. *Comput. Optim. Appl.* **21**(1), 5–20 (2002)
29. Rajaraman, A., Leskovec, J., Ullman, J.D.: *Mining of Massive Datasets*. Cambridge University Press, Cambridge (2012)
30. Shi, Y.: Brain storm optimization algorithm. In: Tan, Y., Shi, Y., Chai, Y., Wang, G. (eds.) *ICSI 2011, Part I. LNCS*, vol. 6728, pp. 303–309. Springer, Heidelberg (2011)
31. Shi, Y.: An optimization algorithm based on brainstorming process. *Int. J. Swarm Intell. Res. (IJSIR)* **2**(4), 35–62 (2011)
32. Yang, P., Tang, K., Lu, X.: Improving estimation of distribution algorithm on multimodal problems by detecting promising areas. *IEEE Trans. Cybern.* **45**(8), 1438–1449 (2015)
33. Yang, S., Li, C.: A clustering particle swarm optimizer for locating and tracking multiple optima in dynamic environments. *IEEE Trans. Evol. Comput.* **14**(6), 959–974 (2010)
34. Yang, Z., Tang, K., Yao, X.: Differential evolution for high-dimensional function optimization. In: *Proceedings of 2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, pp. 35231–3530. IEEE (2007)
35. Yang, Z., Tang, K., Yao, X.: Scalability of generalized adaptive differential evolution for large-scale continuous optimization. *Soft. Comput.* **15**(11), 2141–2155 (2011)
36. Zhou, Z.H., Chawla, N.V., Jin, Y., Williams, G.J.: Big data opportunities and challenges: discussions from data analytics perspectives. *IEEE Comput. Intell. Mag.* **9**(4), 62–74 (2014)
37. Zlochin, M., Birattari, M., Meuleau, N., Dorigo, M.: Model-based search for combinatorial optimization: a critical survey. *Ann. Oper. Res.* **131**, 373–395 (2004)