Multi-Objective Evolutionary Algorithms Embedded with Machine Learning - A Survey

Zhun Fan
Guangdong Provincial Key Laboratory of Digital Signal and Image Processing
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

Kaiwen Hu
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

Fang Li
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

Yibiao Rong
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

Wenji Li
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

Huibiao Lin
Department of Electronic Engineering
Shantou University
Guangdong, Shantou 515063

***Abstract*—Multi-objective evolutionary algorithms (MOEAs) have been widely used in solving multi-objective optimization problems. A great number of the-state-of-art MOEAs have been proposed. These MOEAs can be classified into the following categories: decomposition-based, domination-based, indicator-based, and probability-based methods. Among them, the first four categories belong to non-model based methods, while the fifth one is considered to be model-based method, in which machine learning techniques are often used to build the models. Recently, embedding machine learning mechanisms into MOEAs is becoming popular and promising. In this paper, a relatively thorough review on both traditional MOEAs and those equipped with machine learning mechanisms are made, with the aim of shedding light on the future development of this emerging research field.**

***Index Terms*—Evolutionary Algorithm, Machine Learning**

I. Introduction

Many real-world problems can be formulated as optimization problems. These problems often include two- or three-objectives to be optimized, and are normally called multi-objective optimization problems (MOPs). In particular, when the number of objectives to be optimized is greater than three, they are called many-objective optimization problems. In general, a MOP can be defined as follows:

$$\begin{aligned} &minimize && F(X) = (f_1(X), ..., f_m(X))^T \\ &subject\ to && X \in \Omega \end{aligned} \tag{1}$$

where $X = (x_1, x_2, ..., x_n)^T \in \Omega$ constitutes decision variable(vector), Ω is the decision space, R^m is the objective space, and $F : \Omega \to R^m$ is a mapping from the decision space to the objective space composing of m real-value evaluation functions. In general, no single point in the decision space can reach the optimum in each objective simultaneously. Instead, a series of points which cannot be dominated by any other point form a Pareto Set (PS) in the decision space, are mapped to the objective space forming the Pareto Front(PF).

Let $F = (f_1, f_2, ..., f_m)$, $F' = (f'_1, f'_2, ..., f'_m) \in R^m$ be two decision vectors. F is said to dominate F' if $x_i \leq y_i$ for all $i = 1, 2, ...m$ and $F \neq F'$. A point $X^* \in \Omega$ is called Pareto optimal if there is no $X \in \Omega$ such that $F(X)$ dominates $F(X^*)$. In applying MOEAs to solve MOPs, convergence and diversity are two major indexes to monitor the performance of MOEAs. These two goals are often conflicting with each other, so a main issue of designing MOEAs is how to achieve a proper balance of them. Being able to obtain a set of solutions in single running and optimize multiple objectives at the same time, MOEAs are often used to solve non-linear, complex and large-scale optimization problems. In the past decade, many MOEAs have been proposed, which can be divided into the following categories: decomposition-based (such as MOEA/D [1]), domination-based (such as NSGA-II [2]), indicator-based (such as IBEA [3]) and probability-based (such as MOEA/D-EGO [4]) methods. In this paper, some very recently works in MOEA are also included, with some hybrid methods discussed in detail.

The rest of paper is organized as follows. Section II has a brief survey on existing state-of-the-art MOEAs based on non-model method. In section III several traditional techniques of machine learning are presented and some representative model-based MOEAs are reviewed. Finally, the paper concludes in section IV with some potential directions for future research discussed.

II. Non-Model-Based MOEAs

The research work on MOEAs, which are based on non-model method, is surveyed in this section from different aspects. The framework of MOEAs in general contains the following three operations - reproduction, evaluation and selection, which form an iterative process. The more crucial operations, which to a larger extent decide the performance and characteristics of the algorithm, is reproduction and reproduction.Due to this reason, the non-model-based MOEAs

978-1-5090-0623-6/16/$31.00

can be divided into the following three classes according to the operation of selection, which are dominance-based, decomposition-based and indicator-based ones. In dominance-based selection methods, a solution is selected to enter into next generation based on the non-dominated ranking. Typical methods of this type include NSGA-II [2], SPEA-II [5], and PAES-II [6]. In the Indicator-based selection category, a solution is selected into next generation according to its contribution to the performance metrics such as IGD [7] and HV [8]. Representative methods include IBEA [3], R2IBEA [9], SMS-EMOA [10] and HypE [11]. In decomposition-based selection mechanism, a solution is kept to the next generation based on certain decomposition functions. Representative methods include IMMOGLS [12], UGA [13], cMOGA [14], MOEA/D [1] and MOEA/D-M2M [15].

Currently, MOEA/D is a very popular decomposition-based Method. It decomposes a multi-objective optimization problem into a number of scalar objective optimization subproblems and optimizes them in a collaborative way. The diversity can be obtained in an implicit manner by setting the weight vectors in MOEA/D. A number of MOEA/D variants have been proposed and studied (e.g., [16], [17], [15], [18], [19])), with their main efforts also focusing on balancing the performance of convergence and diversity. MOEA/D-DRA [18] emphasizes on improving the performance of convergence by allocating the computational resource to each subproblems dynamically based on their utility functions. MOEA/D-M2M [15] concentrates on enhancing the performance of diversity by decomposing a multi-objective optimization problem into a set of simpler multi-objective optimization subproblems. MOEA/D-STM [17] uses a stable matching(STM) model to trade off the performance of convergence and diversity. EAG-MOEA/D [19] uses an external archive to guide the search direction, and the performance of convergence and diversity can be balanced effectively.

III. Model-based MOEAs

In practice, a MOP is hard to be solved by analytical method. The model-based method supplies an approximation of the true global optimal solutions from probabilistic perspective.

The reason to build a model is to extract information hidden in data which contains e.g. the position distribution of population individuals in the searching space, landscape with feasible area and relationship between PF and PS. In the following sections, we briefly introduce some typical machine learning techniques and their applications in some model-based MOEAs.

A. Review of Machine Learning

Machine learning, which evolved from the study of pattern recognition and computational learning theory in artificial intelligence, is a subfield of computer science and a very useful tool for information extraction. It explores the study and design of algorithms that can learn from and make predictions based on data [20]. Generally, the methods of machine learning can be divided into two categories: supervised methods and unsupervised methods. Supervised methods [21], [22], [23], [24], [25] infer a model from labeled data, then the inferred model is used to make a prediction for unseen instances. Unsupervised methods [26], [27] on the other hands infer a function to describe hidden structure from unlabeled data.

1) Supervised Method:

- The K-Nearest Neighbors (KNN): KNN method is a non-parametric method [21]. It classifies an object by a majority vote of its neighbors, with the object being assigned to the most common class among its k nearest neighbors. There are several key elements of this approach [28]: (i) the set of labeled objects to be used for evaluating a test object class, (ii) a distance or similarity metric that can be used to compute the closeness of objects, (iii) the number of nearest neighbors k, and (iv) the method used to determine the class of the target object based on the classes and distances of the k nearest neighbors.
- Boosting algorithm: Boosting is a family of ensemble algorithms which try to construct a set of weak learners and combine them to form a strong learner. The general boosting procedure is illustrated in [28]. It adjusts the distribution of the training set to generate the weak learners. One of the most influential ensemble methods is the AdaBoost algorithm formulated by Yoav Freund and Robert Schapire who won the Godel Prize, which is one of the most prestigious awards in theoretical computer science, in the year 2003 for their AdaBoost paper [22].
- Support vector machines (SVMs) algorithm: SVMs were developed by Vapnik in the 1990s [23], [29], which have a strong theoretical foundation rooted in statistical learning theory and are often insensitive to the number of dimensions. SVMs is based on the Structural Risk Minimization principle from computational theory [23]. The idea of SVM is to find a hypothesis to guarantee the lowest true error. The kernel trick [30] is a commonly used technique in SVM to solve linearly inseparable problems [28]. The key is to define an appropriate kernel function based on the inner product between the given data, as a nonlinear transformation of data from the input space with higher dimension in order to make the problems linearly separable. The underlying justification of kernel trick is a complex pattern classification problem nonlinearly cast in a high-dimensional space is more likely to be linearly separable than in a low-dimensional space [28].
- A decision tree is a tree where each split node stores a split function to be applied to the incoming data. Each leaf node stores a final predictor [24]. One of the great works on decision tree is the Classification and Regression Tree [31]. However, training optimal decision trees from data has been a long standing problem [24], for which one of the most popular algorithms is C4.5 of Quinlan [32]. In this early work [31], [32], trees are treated as individual entities. A random decision forest is instead an ensemble of randomly trained decision trees.

It has been shown to yield superior generalization to both boosting and pruned C4.5-trained trees on certain tasks [20]. Generally, there are two ways to train the trees for the random forest: randomly subsampling the data used to train each tree or randomly subsampling the features used to train each node [33]. In testing, an unseen data is given, starting at the root, each split node applies its associated split function to the given data. Depending on the result of the split function, the data is sent to left or right child. This process is repeated until the data point reaches a leaf node and the predictor is stored in the leaf node. The predicted result of the random forest takes the average or the majority vote over outputs of all the trees.

- Naive Bayes algorithm: Naive Bayes, also called independence Bayes [27] is a probabilistic classifier based on applying Bayes Theorem. This method is easy to implement, not needing any complicated iterative parameter estimation schemes. In addition, it is so easy to interpret that users unfamiliar with classifier technology can understand why it is making the classification it makes [28]. In Naive Bayes algorithm, the assumption is that the features is independent, meaning that the order of features is irrelevant and that the presence of one feature does not affect other features in classification tasks. This assumption makes the computation of Bayesian classification more efficient, but the applicability of Naive Bayes is limited by this assumption of feature independence [34].

2) Unsupervised methods:

- K-means: The term k-means was first used by J. MacQueen in 1967 [26]. It is a straightforward and widely used clustering algorithm. Given a set of objects, the goal of clustering is to divide these objects into groups such that objects within a group tend to be more similar to one another as compared to objects belonging to different groups. The K-means algorithm is a simple iterative clustering algorithm that partitions a given dataset into a user-specified number of clusters so as to minimize the sum of distance of each point in the cluster to the K centers [28].
- Expectation-Maximization (EM) algorithm: The EM algorithm was explained and given its name by A. Dempster et al. [27] in 1977. It is a broadly applicable approach to the iterative estimates of maximum likelihood (ML), useful in a variety of incomplete data problems [28]. It is an iterative method for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables. There are two steps in the EM algorithm. In the expectation (E) step, a function for the expectation of the log-likelihood evaluated using the current estimate for the parameter is created. In the maximization (M) step, parameters maximizing the expected log-likelihood found on the E step are computed. The EM iteration alternates between performing these two steps until the stop condition is reached [27].

Since the conventional machine learning techniques were limited in their ability to process natural data in their raw form, namely we have to design certain appropriate features to classify raw data. To design appropriate features is a time-consuming work and needs a great number of priori knowledge. The most important thing is that machine learning techniques can not work well while appropriate features are not extracted. In recent years, because deep learning has the capability to learn appropriate features of raw data automatically, it becomes a hot research topic [35]. Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction [35]. Deep learning has dramatically improved the state-of-the-art in many applications, such as speech recognition, visual object recognition.

From the brief reviews, machine learning technologies can dig problem specific information from seemingly disordered data. Most data generated in process of optimization with MOEA looks seemingly disorder. However, certain regularities, e.g. functional relationship between objective value and decision variables[36], [37], [38], [39], [40], exist among them. And these regularities can not be found by intuition easily. Thus, we can employ machine learning technologies to dig these regularities to guide MOEAs to search global optimal solutions more efficiently and make MOEAs more effective in solving optimization problems.

B. Review of Model-based MOEAs

As mentioned in the previous section, machine learning technique can provide several ways to extract valuable information from data. In this section, a multi-objective evolutionary algorithm, which uses machine learning to obtain information from data, can be defined as model-based multi-objective evolutionary algorithm. For instance, model-based MOEAs, which integrate with clustering method such as k-mean [41], estimation of distribution(EDA) [42], MMEA [43], RM-MEDA [38], EDA-NDE [44], clustering method CLUMOEA [45], modeling based on Gaussian model MOEA/D-MG [46], IM-MOEA[47], etc. Most existing MOEAs aim at finding or approximating the PF at the objective space. In many real-world applications, a good approximation to the PS is needed. A clustering method can also be applied in MOEA, called as clustering based multi-objective evolutionary algorithm (CLUMOEA) [45], which uses K-means clustering method to divide the population individuals into several classes in each generation, and only the solutions in the same class are permitted to reproduce with each other. In [43], an algorithm which classifies a population into a number of subpopulations based on their distribution in the decision space determined by clustering method, is proposed and named as model based multi-objective evolutionary algorithm(MMEA). In [38], an algorithm, which establishes an estimation of distributed in decision space by regularity model, the PS of a continuous MOP can be divided into a number of piecewise continuous (m-1)-dimensional manifolds under

Karush-Kuhn-Tucker(KKT) condition [48], is proposed and named as regularity model based multi-objective estimation of distribution algorithm(RM-MEDA)[38]. The method of estimation of distribution, which is based on a regularity model and aims to build a probabilistic model for predicting promising solutions in the search space based on statistical information extracted from the previous populations, is used to generate new solutions by sampling from the distribution model. The method has been integrated in the framework of MOEA for building a model of distribution of population in the decision space. Gaussian model was widely used for modeling [46], [**?**]. However, in practice, the assumption of Gaussian distribution is hardly satisfied in all cases. More complicated models entail more parameters to be described [46]. An algorithm called nEDA [44], based on Monte Carlo method [49] and non-parameter model has also been proposed. Another important research is based on particle filter (PF) [50], which builds a probabilistic model by a number of particles utilizing Monte Carlo method. PF has been combined with evolutionary algorithm, resulting in a new algorithm called as evolutionary particle filter (EPF)[51].

IV. Conclusion and Future Direction

In this paper, research work on non-model-based MOEAs and model-based MOEAs has been surveyed. In non-model based MOEAs, very limited rigorous analysis is needed, and information on population and individuals has not been fully extracted and utilized in the process of evolving. The model-based MOEA is more capable of extracting useful information from that process, equipped with mathematical models based on probability theories. Machine learning techniques have great potential to be integrated in the framework of MOEAs thought this integration is still in its early stage. The following issues may help to shape the future research of MOEAs.

- Combine machine learning and MOEAs: The current MOEAs rarely explicitly utilize statistical information of the original data, which contains e.g. distribution of population, clustering of population, etc. Machine learning can extract useful information from the data generated in the evolutionary process, which can in return benefit the evolutionary process if utilized properly. More research should be done to investigate the integration of machine learning and evolutionary computation, in particular in the framework of MOEAs.
- Design novel operators for MOEAs: In [46], authors have generalized the traditional operators originally designed for single objective optimization, which lose their efficacy in multi-objective scenarios. Many other researchers have done research in this line and propsed new operators, such as [52], [53], [54], [55], [46], [56]. Thus, new operators designed for multi-objective and/or many-objective optimization problems are another key issue to the development of MOEAs.

References

[1] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 6, pp. 712–731, 2007.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, 2002.

[3] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving from Nature-PPSN VIII*. Springer, 2004, pp. 832–842.

[4] Q. Zhang, W. Liu, E. Tsang, and B. Virginas, "Expensive multiobjective optimization by MOEA/D with gaussian process model," *Evolutionary Computation, IEEE Transactions on*, vol. 14, no. 3, pp. 456–474, 2010.

[5] E. Zitzler, M. Laumanns, L. Thiele, E. Zitzler, E. Zitzler, L. Thiele, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," 2001.

[6] D. W. Corne, N. R. Jerram, J. D. Knowles, M. J. Oates *et al.*, "Pesa-ii: Region-based selection in evolutionary multiobjective optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO2001*. Citeseer, 2001.

[7] P. A. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *Evolutionary Computation, IEEE Transactions on*, vol. 7, no. 2, pp. 174–188, 2003.

[8] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *evolutionary computation, IEEE transactions on*, vol. 3, no. 4, pp. 257–271, 1999.

[9] D. H. Phan and J. Suzuki, "R2-IBEA: R2 indicator based evolutionary algorithm for multiobjective optimization," in *Evolutionary Computation (CEC), 2013 IEEE Congress on*. IEEE, 2013, pp. 1836–1845.

[10] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[11] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary computation*, vol. 19, no. 1, pp. 45–76, 2011.

[12] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 28, no. 3, pp. 392–403, 1998.

[13] Y.-W. Leung and Y. Wang, "Multiobjective programming using uniform design and genetic algorithm," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 30, no. 3, pp. 293–304, 2000.

[14] T. Murata, H. Ishibuchi, and M. Gen, "Specification of genetic search directions in cellular multi-objective genetic algorithms," in *Evolutionary multi-criterion optimization*. Springer, 2001, pp. 82–95.

[15] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 3, pp. 450–455, 2014.

[16] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 2, pp. 284–302, 2009.

[17] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 6, pp. 909–923, 2014.

[18] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of moea/d on cec09 unconstrained mop test instances." in *IEEE Congress on Evolutionary Computation*, vol. 1, 2009, pp. 203–208.

[19] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 19, no. 4, pp. 508–523, Aug 2015.

[20] R. Kohavi and F. Provost, "Glossary of terms," *Machine Learning*, vol. 30, no. 2-3, pp. 271–274, 1998.

[21] E. Fix and J. L. Hodges Jr, "Discriminatory analysis-nonparametric discrimination: consistency properties," DTIC Document, Tech. Rep., 1951.

[22] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[23] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2013.
[24] A. Criminisi, J. Shotton, and E. Konukoglu, *Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning*. Now, 2012.
[25] D. J. Hand and K. Yu, "Idiot's bayesnot so stupid after all?" *International statistical review*, vol. 69, no. 3, pp. 385–398, 2001.
[26] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA., 1967, pp. 281–297.
[27] A. D. SI, N. Laird, and D. Rubin, "Maximum likelihood estimation from incomplete data via the em algorithm," *J. Royal Statist. Soc. B*, 1977.
[28] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.
[29] V. N. Vapnik and V. Vapnik, *Statistical learning theory*. Wiley New York, 1998, vol. 1.
[30] B. Schiilkopf, "The kernel trick for distances," in *Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference*, vol. 13. MIT Press, 2001, p. 301.
[31] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.
[32] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.
[33] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," 2014.
[34] B. Baharudin, L. H. Lee, and K. Khan, "A review of machine learning algorithms for text-documents classification," *Journal of advances in information technology*, vol. 1, no. 1, pp. 4–20, 2010.
[35] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning." *Nature*, vol. 521, no. 7553, pp. 436–44, 2015.
[36] K. Deb, A. Sinha, and S. Kukkonen, "Multi-objective test problems, linkages, and evolutionary methodologies," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 1141–1148.
[37] H. Li and Q. Zhang, "A multiobjective differential evolution based on decomposition for multiobjective optimization with variable linkages," in *Parallel problem solving from nature-PPSN IX*. Springer, 2006, pp. 583–592.
[38] Q. Zhang, A. Zhou, and Y. Jin, "RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm," *Evolutionary Computation, IEEE Transactions on*, vol. 12, no. 1, pp. 41–63, 2008.
[39] K. Deb and A. Srinivasan, "Innovization: Innovating design principles through optimization," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 1629–1636.
[40] K. Deb, "Unveiling innovative design principles by means of multiple conflicting objectives," *Engineering Optimization*, vol. 35, no. 5, pp. 445–470, 2003.
[41] S. Rogers and M. Girolami, *A first course in machine learning*. CRC Press, 2011.
[42] P. Larranaga and J. A. Lozano, *Estimation of distribution algorithms: A new tool for evolutionary computation*. Springer Science & Business Media, 2002, vol. 2.
[43] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 5, pp. 1167–1189, 2009.
[44] L. Zhou, A. Zhou, G. Zhang, and C. Shi, "An estimation of distribution algorithm based on nonparametric density estimation," in *Evolutionary Computation (CEC), 2011 IEEE Congress on*. IEEE, 2011, pp. 1597–1604.
[45] H. Zhang, S. Song, A. Zhou, and X.-Z. Gao, "A clustering based multiobjective evolutionary algorithm," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, 2014, pp. 723–730.
[46] A. Zhou, Q. Zhang, and G. Zhang, "Multiobjective evolutionary algorithm based on mixture gaussian models," *J Softw*, vol. 25, no. 5, pp. 913–928, 2014.
[47] R. Cheng, Y. Jin, K. Narukawa, and B. Sendhoff, "A multiobjective evolutionary algorithm using gaussian process-based inverse modeling," *Evolutionary Computation, IEEE Transactions on*, vol. 19, no. 6, pp. 838–856, Dec 2015.
[48] G. Feng, Z. Lin, and B. Yu, "Existence of an interior pathway to a karush-kuhn-tucker point of a nonconvex programming problem," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 32, no. 6, pp. 761–768, 1998.
[49] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.
[50] N. Chopin, "A sequential particle filter method for static models," *Biometrika*, vol. 89, no. 3, pp. 539–552, 2002.
[51] S. Park, J. P. Hwang, E. Kim, and H.-J. Kang, "A new evolutionary particle filter for the prevention of sample impoverishment," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 4, pp. 801–809, 2009.
[52] S. Tsutsui, M. Yamamura, and T. Higuchi, "Multi-parent recombination with simplex crossover in real coded genetic algorithms," in *Proceedings of the genetic and evolutionary computation conference*, vol. 1, 1999, pp. 657–664.
[53] Y. Mei, K. Tang, and X. Yao, "A global repair operator for capacitated arc routing problem," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 3, pp. 723–734, 2009.
[54] Z. Fan, H. Huang, W. Li, S. Xie, X. Cai, and E. Goodman, "An opposition-based repair operator for multi-objective evolutionary algorithm in constrained optimization problems," in *Natural Computation (ICNC), 2015 11th International Conference on*, Aug 2015, pp. 330–336.
[55] K. Harada, J. Sakuma, I. Ono, and S. Kobayashi, "Constraint-handling method for multi-objective function optimization: Pareto descent repair operator," in *Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 156–170.
[56] A. Zhou, Q. Zhang, and G. Zhang, "A multiobjective evolutionary algorithm based on decomposition and probability model," in *Evolutionary Computation (CEC), 2012 IEEE Congress on*. IEEE, 2012, pp. 1–8.