This document is published in:

Di Chio, Cecilia, et al. (eds.) (2010) *Applications of Evolutionary Computation: EvoApplicatons 2010: EvoCOMPLEX, EvoGAMES, EvoIASP, EvoINTELLIGENCE, EvoNUM, and EvoSTOC, Istanbul, Turkey, April 7-9, 2010, Proceedings, Part I.* (Lecture Notes in Computer Science, 6024). Springer, pp. 512- 521.
Doi: http://dx.doi.org/10.1007/978-3-642-12239-2_53

Advancing Model–Building for Many–Objective Optimization Estimation of Distribution Algorithms

Luis Martí, Jesús García, Antonio Berlanga, and José M. Molina

Universidad Carlos III de Madrid, Group of Applied Artificial Intelligence
Av. de la Universidad Carlos III, 22. Colmenarejo, Madrid 28270, Spain
{lmarti,jgherrer}@inf.uc3m.es, {aberlan,molina}@ia.uc3m.es

Abstract. In order to achieve a substantial improvement of MOEDAs regarding MOEAs it is necessary to adapt their model–building algorithms. Most current model–building schemes used so far off–the–shelf machine learning methods. These methods are mostly error–based learning algorithms. However, the model–building problem has specific requirements that those methods do not meet and even avoid.

In this work we dissect this issue and propose a set of algorithms that can be used to bridge the gap of MOEDA application. A set of experiments are carried out in order to sustain our assertions.

1 Introduction

The multi–objective optimization problem (MOP) can be expressed as the problem in which a set of objective functions should be jointly optimized. In this class of problems the optimizer must find one or more feasible solutions that jointly minimizes (or maximizes) the objective functions. Therefore, the solution to this type of problem is a set of trade–off points.

Evolutionary algorithms (EAs) have proven themselves as a valid and competent approach from theoretical and practical points of view. These multi–objective evolutionary algorithms (MOEAs) [4] have succeeded when dealing with these problems because of their theoretical properties and real–world application performance.

There is a class of MOPs that are particularly appealing because of their inherent complexity: the so–called many–objective problems [18]. These are problems with a relatively large number of objectives.

The results of works that have ventured into these problems called for the search of other approaches that could handle many–objective problems with a reasonable performance. Among such approaches we can find estimation of distribution algorithms (EDAs) [13]. However, although multi–objective EDAs (MOEDAs) have yielded some encouraging results, their introduction has not lived up to their a priori expectations. This fact can be attributed to different causes, some of them, although already existing in single–objective EDAs, are better exposed in MOEDAs, while others are derived from the elements taken from MOEAs. An analysis on this issue led us to distinguish a number of inconveniences, in particular, the drawbacks derived from the incorrect treatment of population outliers; the loss of population diversity, and; the dedication of an excessive computational effort to finding an optimal population model.

There have been some works that have dealt with those three issues, in particular with the loss of diversity. Nevertheless, the community has failed to acknowledge that, perhaps, the underlying cause for those problems can be traced back to the algorithms used for model–building in EDAs.

In this work we examine the model–building issue of EDAs in order to show that some its characteristics, which have been ignored so far, render most current approaches inviable. We hypothesize that the problems of current EDAs can be traced back to the error–based machine learning algorithms used for mode–building and, that new classes of algorithms must be applied to properly deal with the problem. With that idea in mind we carried out a set of experiments that compare some algorithms typically used for model–building with other that, according to our hypothesis should perform well in this class of problems.

Reaching a rigorous understanding of the state–of–the–art in MOEDAs' model–building is hard since each model builder is embedded in a different MOEDA framework. Therefore, in order to comprehend the advantages and shortcomings of each algorithm, they should be tested under similar conditions and isolated from the MOEDA it is part of. That is why, in this work we assess some of the main machine learning algorithms currently used or suitable for model–building in a controlled environment and under identical conditions. This framework guarantees the direct comparison of the algorithms and allows for valid tests.

The rest of this contribution proceeds as we introduce the theoretical aspects that support our discussions. We then deal with the model–building problem, its properties and how it has been approached by the main MOEDAs. Subsequently, a set of experiments, using community–accepted, complex and scalable test problems with a progressive increase in the number of objective functions. Finally some concluding remarks and lines for future work are put forward.

2 Theoretical Background

The concept of multi–objective optimization refers to the process of finding one or more feasible solutions of a problem that corresponds to the extreme values (either maximum or minimum) of two or more functions subject to a set of restrictions.

More formally, a multi–objective optimization problem (MOP) can be defined as:

Definition 1 (Multi–objective Optimization Problem)

$$\left.\begin{array}{c}\text{minimize } \boldsymbol{F}(\boldsymbol{x}) = \langle f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})\rangle\,, \\ \text{with } \boldsymbol{x} \in \mathcal{D}\,,\end{array}\right\} \tag{1}$$

where $\mathcal{D}$ is known as the decision space*. The functions $f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})$ are the* objective functions*. The image set, $\mathcal{O}$, product of the projection of $\mathcal{D}$ through $f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})$ is called* objective space *($\boldsymbol{F} : \mathcal{D} \rightarrow \mathcal{O}$).*

In this class of problems the optimizer must find one or more feasible solutions that jointly minimizes (or maximizes) the objective functions. Therefore, the solution to this type of problem is a set of trade–off points. The adequacy of a solution can be expressed in terms of the Pareto dominance relation. The solution of (1) is the Pareto–optimal set, $\mathcal{D}^*$; which is the subset of $\mathcal{D}$ that contains elements that are not dominated by other elements of $\mathcal{D}$. Its image in objective space is called Pareto–optimal front, $\mathcal{O}^*$.

MOPs have been addressed with a broad range of approaches. Among them, evolutionary algorithms (EAs) have proven themselves as a valid and competent approach from theoretical and practical points of view. These multi–objective evolutionary algorithms (MOEAs) have succeeded when dealing with these problems because of their theoretical properties and real–world application performance.

2.1 Estimation of Distribution Algorithms

Estimation of distribution algorithms (EDAs) have been claimed as a paradigm shift in the field of evolutionary computation. Like EAs, EDAs are population based optimization algorithms. However in EDAs the step where the evolutionary operators are applied to the population is substituted by construction of a statistical model of the most promising subset of the population. This model is then sampled to produce new individuals that are merged with the original population following a given substitution policy. Because of this model–building feature EDAs have also been called probabilistic model–building genetic algorithms (PMBGAs).

The introduction of machine learning techniques implies that these new algorithms lose the biological plausibility of its predecessors. In spite of this, they gain the capacity of scalably solve many challenging problems, significantly outperforming standard EAs and other optimization techniques.

Probably because of their success in single–objective optimization, EDAs have been extended to the multi–objective optimization problem domain, leading to multi–objective EDAs (MOEDAs) [17].

3 Error–Based Learning in Model–Building Algorithms

One topic that remains not properly studied inside the MOEDA scope is the scalability of the algorithms. The most critical issue is the dimension of the objective space. It has been experimentally shown to have an exponential relation with the optimal size of the population. This fact implies that, with the increase of the number of objective functions an optimization algorithm needs an exponential amount of resources made available to it.

Notwithstanding the diverse efforts dedicated to providing usable model–building methods for EDAs the nature of the problem itself has received relatively low attention. In spite of the progressively improving succession of results of EDAs, one question arises when looking for ways to further improve them. Would current statistically sound and robust approaches be valid for the problem being addressed? or, in other terms, does the model–building problem have particular demands that require custom–made algorithms to meet them? Machine learning and statistical algorithms, although suitable for their original purpose, might not be that effective in the particular case of model–building.

Generally, those algorithms are off–the-shelf machine learning methods that were originally intended for other classes of problems. On the other hand, the model–building problem has particular requirements that those methods do not meet and even have conflicts with. Furthermore, the consequences of this misunderstanding would be more dramatic when scaling up in the amount of objectives since the situation is aggravated by the implications of the curse of dimensionality.

An analysis of the results yielded by current multi–objective EDAs and their scalability with regard to the number of objective leads to the identification of certain issues that might be hampering the obtention of substantially better results with regard to other evolutionary approaches. Among those issues we can distinguish the following: incorrect treatment of data outliers, and; loss of population diversity.

This behavior, in our opinion, can be attributed the error–based learning approaches that take place in the underachieving MOEDAs. Error–based learning is rather common in most machine learning algorithms. It implies that model topology and parameters are tuned in order to minimize a global error measured across the learning data set. This type of learning isolated data is not taken into account because of their little contribution to the overall error and therefore they do not take an active part of learning process. In the context of many problems this behavior makes sense, as isolated data can be interpreted as spurious, noisy or invalid data.

That is not the case of model–building. In model–building all data is equally important and, furthermore, isolated data might have a bigger significance as they represent unexplored zones of the current optimal search space. This assessment is supported by the fact that most the approaches that had a better performance do not follow the error–based scheme. That is why, perhaps another class of learning, like instance–based learning [11] or match–based learning [7] would yield a sizable advantage. Therefore, it can be presumed that, in order to obtain a substantial improvement on this matter, algorithms that conform those types of learning should be applied.

3.1 Randomized Leader Algorithm

The randomized leader algorithm [8] is a fast and simple partitioning instance–based algorithm that was first used in the EDA context as part of the IDEA framework [3]. Its use is particularly indicated in situations when the overhead introduced by the clustering algorithm must remain as low as possible. Besides its small computational footprint, this algorithm has the additional advantage of not having to explicitly specify in advance how many partitions should be discovered. On the other hand, the drawbacks of the leader algorithm are that it is very sensitive to the ordering of the samples and that the values of its thresholds must be guessed a priori and are problem dependent.

The algorithm goes over the data set exactly once. For each sample drawn it finds the cluster whose leader is the closest, given threshold the ρ_{Ld}. If such partition can not be found, a new partition is created containing only this single sample. Once the amount of samples in a cluster have exceeded the amount ρ_{Lc}, the leader is substituted by the mean of the cluster members. The mean of a partition changes whenever a sample is added to that partition. After obtaining the clustering a Gaussian mixture is constructed relying on it, as described for the naïve MIDEA algorithm [3]. This allows the sampling of the model in order to produce new elements.

3.2 Model–Building Growing Neural Gas

The model–building growing neural gas network (MB–GNG) [15] has been proposed as a form of dealing with the model–building issue. It has been devised with to deal with the model–building issue. The multi–objective neural EDA (MONEDA) [14], that incorporates MB–GNG, has yielded relevant results [14, 16].

MB–GNG is a modified growing neural gas (GNG) network [6]. GNG networks have been chosen previously presented as good candidates for dealing with the model–building issue because of their known sensibility to outliers [19].

The network grows to adapt itself automatically to the complexity of the dataset being modelled. It has a fast convergence to low distortion errors and incorporates a cluster repulsion term to the original adaptation rule that promotes search and diversity.

3.3 Gaussian Adaptive Resonance Theory Network

Adaptive Resonance Theory (ART) neural networks are capable of fast, stable, on-line, unsupervised or supervised, incremental learning, classification, and prediction following a match–based learning scheme [7]. During training, ART networks adjust previously–learned categories in response to familiar inputs, and creates new categories dynamically in response to inputs different enough from those previously seen. A vigilance test allows to regulate the maximum tolerable difference between any two input patterns in a same category. It has been pointed out that ART networks are not suitable for some classes of classical machine–learning applications [20], however, what is an inconvenience in that area is a feature in our case.

There are many variations of ART networks. Among them, the Gaussian ART [21] is most suitable for model–building since it capable of handling continuous data. The result of applying Gaussian ART is a set of nodes each representing a local Gaussian density. These nodes can be combined as a Gaussian mixture that can be used to synthesize new individuals.

4 Experimental Analysis

To identify the model–building issue and its relation to error–based learning it is helpful to devise a comparative experiment that casts light on the different performance of a selected set of model–building algorithms subject to the same conditions when dealing with a group of problems of scaling complexity. In particular, we deal with two of the problems of the Walking Fish Group (WFG) continuous and scalable problem set [9], in particular the WFG4 and WFG9.

WFG4 is a separable and strongly multi–modal problem while WFG9 is non–separable, multi–modal and have deceptive local–optima. Both problems have a concave Pareto–optimal front that lies in the first orthant of a unit hypersphere located at the coordinates origin. This feature make them suitable for high–dimensional experiments where assessing the progress of algorithms is expensive for other shapes of fronts.

A MOEDA framework is shared by the model–building algorithms involved in the tests in order to ensure the comparison and reproducibility of the results.

The model–building algorithms involved in the tests were: (i) expectation maximization algorithm, as described for MIDEA [3]; (ii) Bayesian networks, as used in MrBOA [1]; (iii) $(1+\lambda)$–CMA–ES as described in [10]; (iv) randomized leader algorithm, (v) MB–GNG, and; (vi) Gaussian ART.

4.1 Shared EDA Framework

To test MB–GNG is it essential to insert it in an EDA framework. This framework should be simple enough to be easily understandable but should also have a sufficient problem solving capacity. It should be scalable and preserve the diversity of the population.

Our EDA employs the fitness assignment used by the NSGA–II algorithm [5] and constructs the population model by applying MB–GNG. The NSGA–II fitness assignment was chosen because of its proven effectiveness and its relative low computational cost.

It maintains a population of individuals, P_t, where t is the current iteration. It starts from a random initial population P_0 of z individuals. It then proceeds to sort the individuals using the NSGA–II fitness assignment function. A set $\hat{P}_t$ containing the best $\lfloor \alpha \left| P_t \right| \rfloor$ elements is extracted from the sorted version of P_t,

$$\left| \hat{P}_t \right| = \alpha \left| P_t \right| . \quad (2)$$

The population model is then built using $\hat{P}_t$. The model is then used to create $\lfloor \omega \left| P_t \right| \rfloor$ new individuals is synthesized. Each one of these individuals substitute a randomly selected ones from the section of the population not used for model–building $P_t \setminus \hat{P}_t$. The set obtained is then united with best elements, $\hat{P}_t$, to form the population of the next iteration P_t.

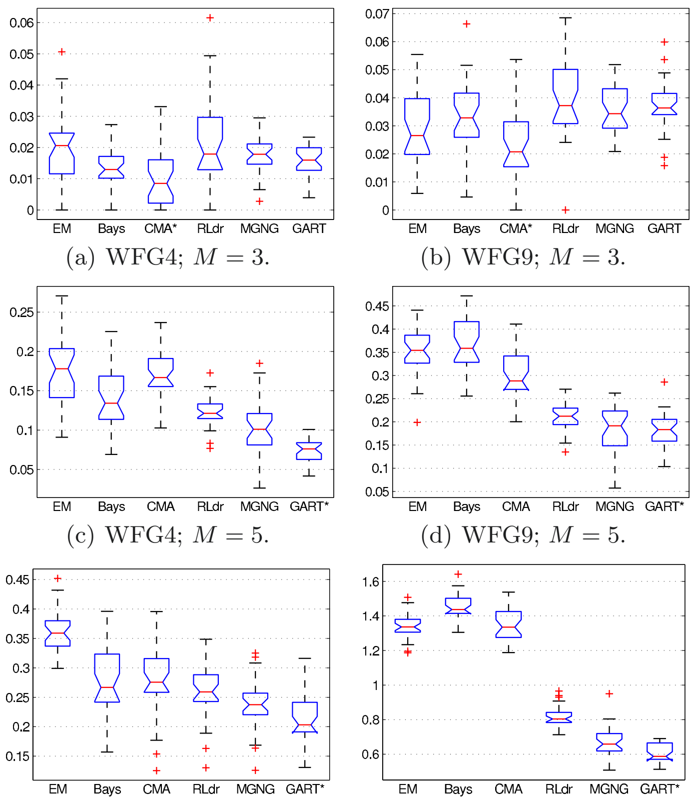
Iterations are repeated until a given stopping criterion is met. The output of the algorithm is the set of non–dominated individuals of P_t.

4.2 Results

The WFG4 and WFG6 problems were configured with 3, 5 and 7 objective functions. The dimension of the decision space was set to 10. Tests were carried out under the PISA experimental framework [2]. The binary additive epsilon indicator [12] was used to assess the performance. Although many other suitable indicators exist we have limited to this one because its low computational footprint and the space constraints imposed to this paper.

Figure 1 shows the box plots obtained after 30 runs of each algorithm for solving the different the problem/dimension configuration.

In the three dimensional problems our approach performed similarly to the rest of the algorithms. This was an expected outcome. However, in the case of

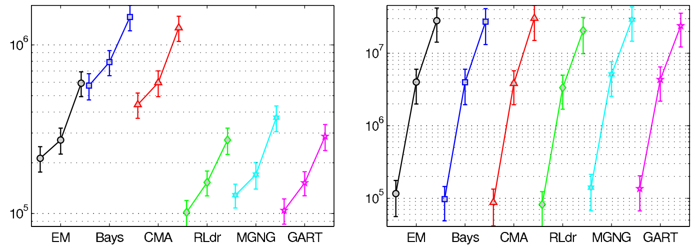


(a) WFG4; $M = 3$. (b) WFG9; $M = 3$.

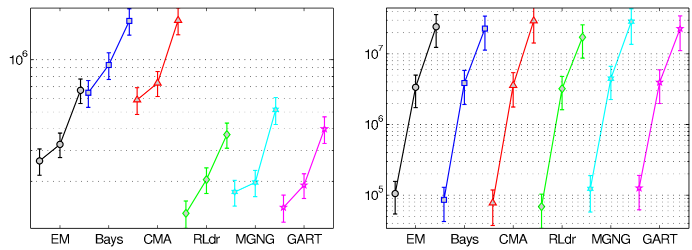
(c) WFG4; $M = 5$. (d) WFG9; $M = 5$.

(e) WFG4; $M = 7$. (f) WFG9; $M = 7$.

Fig. 1. Boxplots of the binary additive epsilon indicator values obtained when dealing with the WFG4 and WFG9 problems with EDAs using expectation–maximization (EM), Bayesian networks (Bays), covariance matrix adaptation ES (CMA), randomized leader algorithm (RLdr), modified growing neural gas networks (MGNG) and Gaussian adaptive resonance theory neural networks (GART) for model–building. The result of each algorithm is measured against a sampled version of the Pareto–optimal front.



(a) WFG4; M–B CPU ops. (b) WFG4; Obj. func. evals.



(c) WFG9; M–B CPU ops. (d) WFG9; Obj. func. evals.

Fig. 2. Analysis of the computational cost of dealing with WFG4 and WFG9. The number of CPU ops dedicated for model–building and the number of objective functions evaluations are measured in each case (see Fig. 1 for algorithms' acronyms).

five and seven the three non–error–based learning algorithms outperform the rest of the optimizers applied.

One can hypothesize that, in this problem, the model–building algorithm induces the exploration of the search space and therefore it manages to discover as much as possible of the Pareto–optimal front. It is most interesting that our proposal exhibits rather small standard deviations. This means that it performed consistently well across the different runs. These results must be investigated further to understand if the low dispersion of the error indicators can only be obtained in the problems solved or if can be extrapolated to other problems.

These results are further confirmed by inspecting figure 2. Here the advantages of using error–based learning in approximation quality terms are supplemented by the low computational costs of those algorithms. It can be perceived that, while all algorithms used similar numbers of objective function evaluations, the three non–error-based ones required far less computational resources to build the population models.

It is important to underline the performance of Gaussian ART that had never been used before in this application context. Gaussian ART outperformed the rest of the approaches in 5 and 7 objectives in WFG4 and WFG6 in terms of solution quality and computational cost.

5 Conclusions

In this paper we have discussed an important issue in current evolutionary multi–objective optimization: how to build algorithms that have better scalability with regard to the number of objectives. In particular, we have focused on one promising set of approaches, the estimation of distribution algorithms.

We have argued that most of the current approaches do not take into account the particularities of the model–building problem they are addressing and, because of that they fail to yield results of substantial quality.

In any case, it seems obvious after the previous discussions and experiments that the model–building problem deserves a different approach that takes into account the particularities of the problem. Perhaps the ultimate solution to this issue is to create custom–made algorithms that meet the specific requirement of this problem.

Acknowledgements

This work was supported by projects CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC, SINPROB, CAM CONTEXTS S2009/TIC-1485 and DPS2008-07029-C02-0.

References

1. Ahn, C.W.: Advances in Evolutionary Algorithms. Theory, Design and Practice. Springer, Heidelberg (2006)
2. Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E.: PISA—A Platform and Programming Language Independent Interface for Search Algorithms. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L. (eds.) EMO 2003. LNCS, vol. 2632, pp. 494–508. Springer, Heidelberg (2003)
3. Bosman, P.A.N., Thierens, D.: The naïve MIDEA: A baseline multi–objective EA. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 428–442. Springer, Heidelberg (2005)
4. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: Evolutionary Algorithms for Solving Multi-Objective Problems. In: Genetic and Evolutionary Computation, 2nd edn. Springer, New York (2007), `http://www.springer.com/west/home/computer/foundations?SGWID=4-156-22-173660344-0`
5. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
6. Fritzke, B.: A growing neural gas network learns topologies. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Advances in Neural Information Processing Systems, vol. 7, pp. 625–632. 625–632. MIT Press, Cambridge (1995)
7. Grossberg, S.: Studies of Mind and Brain: Neural Principles of Learning, Perception, Development, Cognition, and Motor Control. Reidel, Boston (1982)
8. Hartigan, J.A.: Clustering Algorithms. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New York (1975)

9. Huband, S., Hingston, P., Barone, L., While, L.: A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Transactions on Evolutionary Computation 10(5), 477–506 (2006)
10. Igel, C., Hansen, N., Roth, S.: Covariance matrix adaptation for multi-objective optimization. Evolutionary Computation 15(1), 1–28 (2007)
11. Kibler, D., Aha, D.W., Albert, M.K.: Instance–based prediction of real–valued attributes. Computational Intelligence 5(2), 51–57 (1989)
12. Knowles, J., Thiele, L., Zitzler, E.: A tutorial on the performance assessment of stochastic multiobjective optimizers. TIK Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich (2006)
13. Larrañaga, P., Lozano, J.A. (eds.): Estimation of Distribution Algorithms. A new tool for Evolutionary Computation. In: Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers, Dordrecht (2002)
14. Martí, L., García, J., Berlanga, A., Molina, J.M.: Introducing MONEDA: Scalable multiobjective optimization with a neural estimation of distribution algorithm. In: Thierens, D., Deb, K., Pelikan, M., Beyer, H.G., Doerr, B., Poli, R., Bittari, M. (eds.) GECCO 2008: 10th Annual Conference on Genetic and Evolutionary Computation, pp. 689–696. ACM Press, New York (2008); EMO Track "Best Paper" Nominee
15. Martí, L., García, J., Berlanga, A., Molina, J.M.: Scalable continuous multiobjective optimization with a neural network–based estimation of distribution algorithm. In: Giacobini, M., Brabazon, A., Cagnoni, S., Di Caro, G.A., Drechsler, R., Ekárt, A., Esparcia-Alcázar, A.I., Farooq, M., Fink, A., McCormack, J., O'Neill, M., Romero, J., Rothlauf, F., Squillero, G., Uyar, A.Ş., Yang, S. (eds.) EvoWorkshops 2008. LNCS, vol. 4974, pp. 535–544. Springer, Heidelberg (2008), http://dx.doi.org/10.1007/978-3-540-78761-7_59
16. Martí, L., García, J., Berlanga, A., Molina, J.M.: Solving complex high–dimensional problems with the multi–objective neural estimation of distribution algorithm. In: Thierens, D., Deb, K., Pelikan, M., Beyer, H.G., Doerr, B., Poli, R., Bittari, M. (eds.) GECCO 2009: 11th Annual Conference on Genetic and Evolutionary Computation. ACM Press, New York (2009) (to appear)
17. Pelikan, M., Sastry, K., Goldberg, D.E.: Multiobjective estimation of distribution algorithms. In: Pelikan, M., Sastry, K., Cantú-Paz, E. (eds.) Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications. Studies in Computational Intelligence, pp. 223–248. Springer, Heidelberg (2006)
18. Purshouse, R.C., Fleming, P.J.: On the evolutionary optimization of many conflicting objectives. IEEE Transactions on Evolutionary Computation 11(6), 770–784 (2007), http://dx.doi.org/10.1109/TEVC.2007.910138
19. Qin, A.K., Suganthan, P.N.: Robust growing neural gas algorithm with application in cluster analysis. Neural Networks 17(8–9), 1135–1148 (2004), http://dx.doi.org/10.1016/j.neunet.2004.06.013
20. Sarle, W.S.: Why statisticians should not FART. Tech. rep., SAS Institute, Cary, NC (1995)
21. Williamson, J.R.: Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps. Neural Networks 9, 881–897 (1996)