Improving the Performance and Scalability of Differential Evolution

Antony W. Iorio and Xiaodong Li

School of Computer Science and Information Technology,
RMIT University,
Melbourne Australia
iantony@gmail.com,
xiaodong@goanna.cs.rmit.edu.au
http://goanna.cs.rmit.edu.au/~xiaodong/ecml

Abstract. Differential Evolution (DE) is a powerful optimization procedure that self-adapts to the search space, although DE lacks diversity and sufficient bias in the mutation step to make efficient progress on non-separable problems. We present an enhancement to Differential Evolution that introduces greater diversity. The new DE approach demonstrates fast convergence towards the global optimum and is highly scalable in the decision space.

Keywords: Differential Evolution, Optimization, Rotational Invariance.

1 Introduction

Despite the power of many population-based stochastic optimization algorithms, they can meet with difficulties on optimization problems which are non-separable. Traditional Genetic Algorithms fail to optimize these problems efficiently because they typically perform independent perturbations of decision variables. Unfortunately, many real-world problems are not linearly separable. On problems which are not aligned with the principle coordinate axes, the small mutation rates frequently used in Genetic Algorithms are known to be even less efficient than a random search [1]. One approach for optimizing such problems is to use a vector-based scheme such as Differential Evolution.

The Differential Evolution (DE) optimization algorithm works by generating difference vectors between points in the search space, and using the resulting scaled difference vector to perturb existing points in the population [2]. For example offspring can be generated using the DE scheme $\mathbf{x}^{(i)} + F(\mathbf{x}^{(r1)} - \mathbf{x}^{(r2)})$ where $r1$ and $r2$ refer to the indices of two distinct randomly selected individuals from a population, and i is the index of the current individual in the population. In this approach, individuals are selected for a DE operation such that the resulting difference vector, $(\mathbf{x}^{(r1)} - \mathbf{x}^{(r2)})$, has a magnitude greater than 0. The addition of $(\mathbf{x}^{(r1)} - \mathbf{x}^{(r2)})$ to $\mathbf{x}^{(i)}$ is considered to be a mutation operation which perturbs $\mathbf{x}^{(i)}$. F is a scaling factor for the difference vector.

X. Li et al. (Eds.): SEAL 2008, LNCS 5361, pp. 131–140, 2008.

Differential Evolution has a number of attractive features; difference vectors can be correlated with the search space, it uses only $\mathcal{O}(Np)$ processes (where Np is the population size), it doesn't need a predefined probability distribution for generating offspring, the objective functions do not need to be differentiable, it can provide multiple solutions from a single run of the algorithm, it is very simple to implement, and is a parallel optimization procedure like many other population based schemes.

Of course DE has some limitations, which we attempt to address in this paper. It has been reported that DE performs poorly on problems that are not linearly separable because of inefficient exploitation during the differential mutation phase [3]. Two hypotheses were explored by Sutton; when the crossover rate (Cr) is low, DE can exploit the separability of a function. When DE has a Cr of 1.0, DE becomes rotationally invariant and depends entirely on the differential mutation step. In order to efficiently solve non-separable problems, DE typically must depend more on mutation than crossover, although DE lacks selection pressure in the differential mutation step to make efficient progress. Furthermore, making Cr equal to 1.0 is not recommended as it reduces the number of trial vectors and can result in stagnation [4]. Finally, DE becomes highly dependent on population size in order to avoid stagnation when no crossover is employed.

2 Problem Statement

The use of crossover in DE introduces diversity to the population, far more than mutation alone. As the decision space dimension scales, the importance of having a diverse population from which to sample becomes significantly important in order to make efficient progress towards more optimal solutions in the search space. Unfortunately, because the offspring that crossover can generate are dependent on the principle coordinate axes, crossover provides little benefit to the optimization of non-separable problems.

For an algorithm to be rotationally invariant in the context of optimization algorithms it should produce offspring in the same relative location, irrespective of the orientation of the fitness landscape. Although the rotationally invariant DE/rand/1/bin approach provides vector wise samples which are not biased with respect to any particular coordinate axes it also lowers the number of potential offspring dramatically because it does not use crossover [4]. Our contention is that in order for an optimization algorithm to perform efficiently on a non-separable problem it must not exhibit an extreme dependency on the principle coordinate axes. In addition, we contend that it is unnecessary for it to be strictly rotationally invariant, as long as it is capable of generating sufficient diversity.

It is important to elucidate further on one of the reasons for why the rotationally invariant DE/rand/1/bin used in this study performs poorly on non-separable problems. In Figure 1 the offspring and parents are represented for a population size of 4. The number of potential unique offspring that can be sampled for a single base-vector by such a scheme is determined by Equation 1,

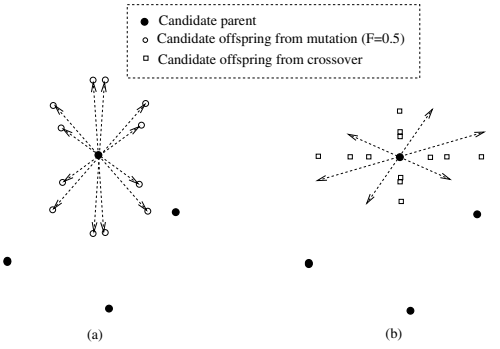

Fig. 1. (a) The offspring generated from a population of 4 using the CSDE scheme. (b) The offspring generated from a population of 4 using the basic DE/rand/1/bin scheme *with* and *without* crossover.

$$(Np-1)(Np-2)(2^D-1) \tag{1}$$

where D is the decision space dimension. The term 2^D represents the number of possible offspring that can be generated from binomial crossover. The term $(Np-1)(Np-2)$ is the number of possible offspring that can be generated from vector-wise mutation. Also, crossover can produce duplicate individuals that were already sampled. In order to not count these individuals, we subtract the duplicates. It deserves to be noted that Equation 1 is equivalent to previous results which reported upon the number of samples possible in an entire population [4].

In accordance with this equation and the offspring distributions in Figure 1(b), an enumeration of all 18 possible offspring for a single target vector is provided in Table 1. The first column of this table details all possible differentials from Figure 1(b). The second contains the location of offspring produced by a mutation operation with F=0.5. The third column contains the coordinates of unique offspring resulting from crossover, which do not overlap with offspring resulting from a mutation operation or any existing parents. The total number of offspring possible from a population of 4 individuals where $0 < CR < 1.0$ is 18.

It is apparent from Equation 1 that as the decision space dimension scales, crossover is responsible for the majority of the offspring individuals that the algorithm can generate through the 2^D term [4]. It is also clear from this figure that crossover samples along the principle coordinate axes, so although it generates many offspring, it also constrains them to this region. It is only capable of independent sampling in each decision space dimension.

If we consider Figure 1(b), where crossover is absent, but rotational invariance is maintained, significantly fewer offspring can be sampled for a single base-vector. The number of potential offspring that can be sampled is equal to $(Np-1)(Np-2)$. The implication here is that a rotationally invariant DE scheme is highly dependent on the population size in order to maintain sample diversity. Although it samples offspring independent of any particular coordinate axes, it does not scale in the decision space as well as a scheme incorporating crossover.

Table 1. An enumeration of all 18 possible unique offspring for a single target vector resulting from crossover and mutation operations in Figure 1(b). The base-vector, $\mathbf{x}^{(i)}$ is located at (2.0,2.5). In addition, the coordinates of A, B, and C are respectively (0.0,0.0), (2.3,-1.0) and (4.0,1.0).

$(\mathbf{x}^{(r1)} - \mathbf{x}^{(r2)})$	$\mathbf{x}^{(i)}$+F$(\mathbf{x}^{(r1)} - \mathbf{x}^{(r2)})$	Crossed with $\mathbf{x}^{(i)}$
A-B = (-2.3,1.0)	(1.15,3.0)	{(1.15,2.5),(2.0,3.0)}
B-A = (2.3,1.0)	(3.15,2.0)	{(3.15,2.5),(2.0,2.0)}
A-C = (-4.0,-1.0)	(0.0,2.0)	{(0.0,2.5),(2.0,2.0)}
C-A = (4.0,1.0)	(4.0,3.0)	{(4.0,2.5),(2.0,3.0)}
B-C = (-1.7,-2.0)	(1.85,1.5)	{(1.85,2.5),(2.0,1.5)}
C-B = (1.7,2.0)	(2.85,3.5)	{(2.85,2.5),(2.0,3.5)}

Ideally, we would like a scheme which is biased in order to accelerate convergence, is capable of generating a diverse variety of offspring solutions in a manner which minimizes distribution bias, and is capable of optimizing non-separable and separable problems equally well. Furthermore, it should be simple to implement and computationally efficient.

As we mentioned earlier, traditional crossover offers sampling diversity, but is really only effective on separable problems because of the way it generates points. It would be desirable for DE to have the capability of producing a large number of samples, while still remaining effective on non-separable problems in high decision space dimensions.

3 An Improved Sample-Based DE Algorithm

In this section we describe the Combinatorial Sampling Differential Evolution (CSDE) algorithm which uses a 'target' best individual and maintains diversity using the sampling of difference vectors from two parent vectors. For the purpose of simplicity we describe the behavior of the algorithm in a two dimensional decision space, although the process easily generalizes to an arbitrary number of decision space dimensions. We also discuss some of the advantages and characteristics of the approach which are different from the typical DE.

Firstly, the algorithm loops over all individuals in a population such that each individual $\mathbf{x}^{(i)}$ has an opportunity to participate in the DE calculation. A second individual $\mathbf{x}^{(r)}$ is chosen for a difference vector calculation such that the population index r is not equal to i, and r is an index randomly chosen from the population.

Two types of samples are performed in this algorithm around an individual that is deemed to be better than another. The first type of sample we call a C-sample (*correlated sample*) such that the vector difference and perturbation are in the same direction, around a better individual (In Figure 2, $\mathbf{x}^{(i)}$ is better than $\mathbf{x}^{(r)}$, for the purpose of explaining the operation of the algorithm. Of course, if the opposite was true, then sampling would occur around individual $\mathbf{x}^{(r)}$). The point labeled by ①② corresponds to the point specified by Equation (2) and (3).

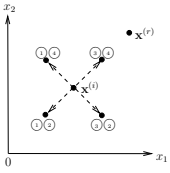

Fig. 2. In a 2-dimensional decision space vectors are sampled around a 'better' individual

In these equations, $u_1^{(i)}$ represents the offspring parameter from the DE mutation equation for the first parameter in the decision vector, and $u_2^{(i)}$ represents the offspring for the second parameter in the decision vector. Similarly, the point labeled by ③④ corresponds to the point specified by Equation (4) and (5). Both points ①② and ③④ are correlated because they are in the same direction as the difference vector. The points at ①② and ③④ are sampled with the same probability.

$$u_1{}^{(i)} = x_1{}^{(i)} + F(x_1{}^{(i)} - x_1{}^{(r)}) \tag{2}$$

$$u_2{}^{(i)} = x_2{}^{(i)} + F(x_2{}^{(i)} - x_2{}^{(r)}) \tag{3}$$

$$u_1{}^{(i)} = x_1{}^{(i)} + F(x_1{}^{(r)} - x_1{}^{(i)}) \tag{4}$$

$$u_2{}^{(i)} = x_2{}^{(i)} + F(x_2{}^{(r)} - x_2{}^{(i)}) \tag{5}$$

The second type of sample is labeled in Figure 2 by ①④ and ③② which respectively correspond to the points generated by Equation 2 and 5 and Equation 4 and 3. Both of these samples are uncorrelated and not rotationally invariant because the magnitudes of the difference vectors for these samples is the result of the difference between $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(r)}$ and they vary depending on the orientation of $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(r)}$. It is this second type of sampling that contributes diversity to the search. In traditional DE, only a single difference vector can result from two points. Our approach dramatically increases the number of possible samples at the expense of always generating rotationally invariant correlated samples. We call these sample points UC-samples (*uncorrelated and correlated samples*), and if the algorithm samples such points, it does so with equal probability for each possible point, including the rotationally invariant correlated sample points ①② and ③④ and the uncorrelated points ①④ and ③②. As the decision space dimension scales, the number of such samples increases in proportion to 2^D, where D is the decision space dimension. In two decision space dimensions, there are four equations that can specify the possible sample points. In three dimensions, there will be eight equations. This can easily be implemented programmatically by specifying an equal probability for $u_j{}^{(i)} = x_j{}^{(i)} + F(x_j{}^{(i)} - x_j{}^{(r)})$ and $u_j{}^{(i)} = x_j{}^{(i)} + F(x_j{}^{(r)} - x_j{}^{(i)})$ to be used for each decision space parameter $j = 1$ to D, so that all possible samples have an equal chance of occurring.

Whether a C-sample or UC-sample occurs is determined probabilistically by a control parameter κ. This parameter is responsible for controlling the balance between C-sampling and UC-sampling in the generation of offspring.

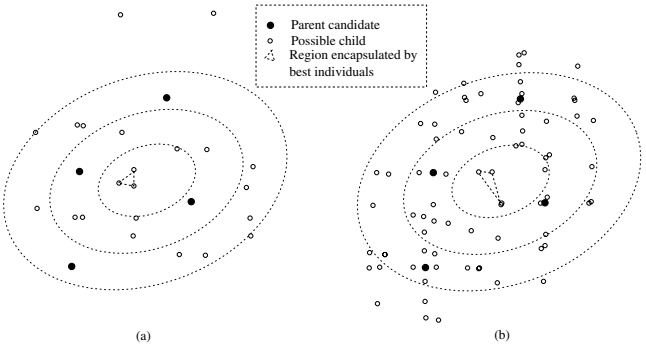

Fig. 3. Distribution of all possible offspring from 3 parent individuals and a mutation scaling factor F=0.5. (a) CSDE and (b) DE/rand/1/bin with crossover.

3.1 Characteristics and Advantages

In CSDE there are two pressures in the generation of offspring; exploitation results from the highly correlated rotationally invariant samples (C-samples) being generated, which rapidly drives the algorithm towards better solutions, and exploration occurs from the UC-sampling, which attempts to discover new and diverse points around the better individual. The UC-sampling method sacrifices emphasis on correlated rotationally invariant points for a dramatic increase in diversity as the decision space dimension scales to higher dimensions. The general idea of this approach is to increase the diversity that DE is capable of generating using a relatively small population size. A critical point to consider here is that although crossover is also not a rotationally invariant scheme, it only generates points which are aligned with the target parent. The UC-samples are not biased in such a fashion, and although they do not result in rotational invariance, they do produce offspring sample vectors distributed around the target vector.

An attractive feature of CSDE is that the number of potential offspring that can be sampled is bounded by 2^D, as in the crossover based DE described in Figure 1(b). Figure 1(a) shows how offspring are sampled using our approach. The number of candidates that can potentially be sampled around a base-vector in the sampling based approach is in proportion to $(Np-1)2^D$. It is apparent from Figure 1(a) that CSDE is superior to standard DE with crossover because it can generate points that are not solely sampled along the principle coordinate axes, unlike DE with crossover in Figure 1(b). As a result, CSDE can be highly effective on problems which are non-separable compared with an algorithm which only produces biased samples along the principle coordinate axes aligned with a parent target vector. Furthermore, it bears mentioning that the smallest population size that CSDE can work with is 2, unlike standard DE which requires 4 individuals.

The difference in efficiency between the CSDE approach and DE/rand/1/bin is detailed further in Figure 3 where the CSDE is capable of generating offspring in near optimal regions of the search space with far fewer samples than the

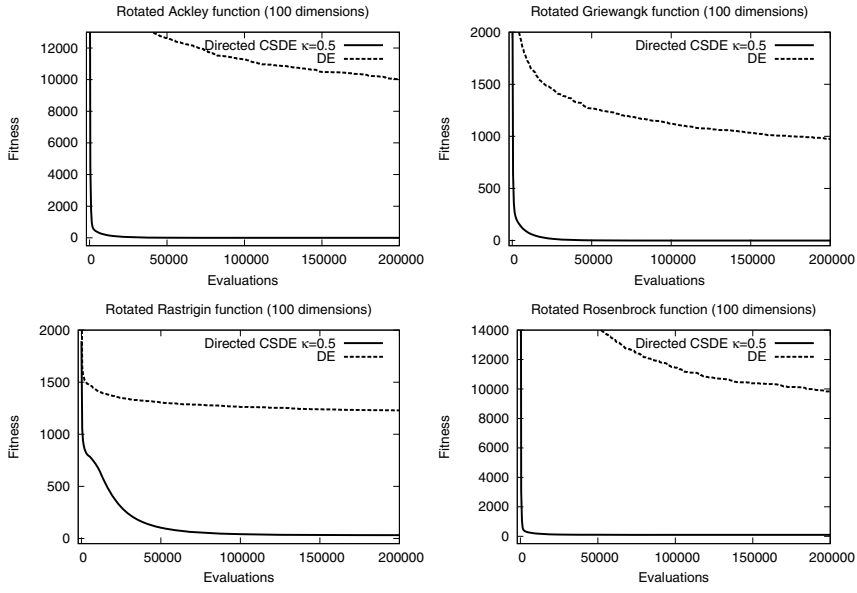

Fig. 4. Fitness over 200,000 evaluations on rotated problems in 100 dimensions of the decision space for the rotated Ackley, Griewangk, Rastrigin and Rosenbrock functions

DE/rand/1/bin approach. Although crossover based DE generates more points, it is apparent from this figure that the sampling based CSDE scheme is superior because of the greater focus it produces on more promising regions of the search space. In contrast, it is apparent that crossover based DE/rand/1/bin samples many more offspring but such offspring may not be advantageous towards finding the optima efficiently.

The proposed sampling based approach can be efficiently directed towards more optimal regions using appropriate vector selection. The CSDE approach is also capable of generating more points than standard rotationally invariant DE and like standard DE, only $\mathcal{O}(Np)$ processes are required.

4 Experiments and Methodology

In the CSDE approach described in this paper, an individual is deemed to be better than another individual with respect to fitness in order to determine an appropriate direction for the vector difference. If a better direction is not apparent from the measures of fitness associated with both individuals, then a direction is chosen randomly. In addition, if an individual has better fitness than another individual it replaces the inferior individual in the population.

Four DE variants are evaluated in this study. Firstly, a baseline DE technique incorporating three vectors is employed. This baseline approach was briefly discussed in the introductory section. For our purposes the baseline DE algorithm

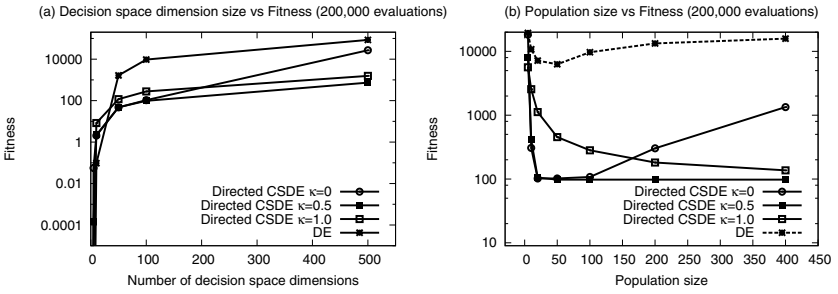

Fig. 5. Problem dimensionality vs. fitness and population size vs fitness after 200,000 evaluations on the rotated Rosenbrock function

used here for benchmarking is equivalent to the DE/rand/1/bin approach [4] because we use it with $CR = 1.0$ in this study.

Secondly, the CSDE algorithm is evaluated with κ set to 1.0, 0.5 and 0. When κ is set to 0.5, half the time the algorithm favors C-samples that are highly directed towards better solutions, otherwise it performs UC-sampling. When κ is set to 0 there is no bias, and vectors are sampled using UC-sampling only. In addition, when κ is set to 1.0, the algorithm solely performs C-samples.

A population size of 100 individuals is used for each of the algorithms on each of the test problems for the performance evaluation of the variants over time, with fitness on the vertical axis. For the DE variants, F is set to 0.5.

The problems that are used in the evaluation of each of the algorithms evaluated in this paper are the rotated Rosenbrock, Griewangk, Ackley and Rastrigin functions. In addition, the Rosenbrock function is evaluated with the algorithm variants in order to determine sensitivity to population size and scalability in the decision space.

A rotation matrix is used to introduced parameter interactions between decision variables, thereby making the problem non-separable. Rotations for each of these test problems are performed in the decision space, on each plane, using a random uniform rotation matrix, which introduces parameter interactions between all parameters [7]. Each algorithm is run 50 times on each test problem, for a total of 200,000 problem evaluations for each run. A new random uniform rotation matrix is generated for each run of each algorithm for the purpose of an unbiased assessment.

5 Scalability in the Decision Space and Sensitivity to Population Size

From Figure 5(a) it is apparent that CSDE with $\kappa = 0.5$ is insensitive to an increase in decision space size on the 100 dimensional rotated Rosenbrock function. In addition, it is able to find highly competitive solutions which are far superior to a canonical rotationally invariant DE/rand/1/bin approach, which performed poorly. In addition, in Figure 5(b) it is apparent that the CSDE

approach with $\kappa = 0.5$ is highly insensitive to a change in population size and is capable of finding similarly good solutions after 200,000 evaluations because of the order of magnitude higher degree of sampling that is possible compared with the canonical rotationally invariant DE/rand/1/bin. In contrast, the performance of CSDE with $\kappa = 0$ where only U-sampling is performed, peaks in performance between a population size of 20 to 100 individuals on the rotated Rosenbrock function in 100 dimensions. This indicates that a large population size detracts from the performance of CSDE when U-sampling is used. The reasons for this are that as the number of individuals in the population increases, the probability of sampling highly directed correlated samples reduces when U-sampling is employed. It is apparent that rotationally invariant correlated sampling is beneficial to the performance of the CSDE approach, in order to make it more insensitive to the population size. In contrast, the CSDE approach with $\kappa = 1.0$ is highly dependent on population size for sampling diversity, and the performance only begins to approach CSDE with $\kappa = 0.5$ as the population size approaches 500 individuals on this problem. It is also clear from Figure 5 that rotationally invariant DE/rand/1/bin performs extremely poorly on the 100 dimensional Rosenbrock problem in the presence of parameter interactions.

From these results it is clearly apparent that when $\kappa = 0.5$, the CSDE algorithm has superior performance over the UC-sampling approach which uses $\kappa = 0$. This indicates that sampling of highly directed rotationally invariant correlated vectors is critically important in order for the algorithm to remain insensitive to population size variations as well as discover highly fit solutions in extremely large decision spaces.

In order to test the performance of the variants over time, the rotated Ackley, Rastrigin, Griewangk and Rosenbrock functions were employed with 100 dimensions. The results in Figure 4 indicate that the performance of the CSDE approach with $\kappa = 0.5$ are dramatically superior to the rotationally invariant DE/rand/1/bin algorithm which does not employ sampling.

6 Implications and Conclusion

In this work we have addressed the stagnation issue discussed by [4] with the CSDE approach. Until now, in order to overcome stagnation in DE a very large population size had to be employed, or crossover was used in order to add more sampling diversity even though crossover is typically ineffective when optimization problems have many parameter interactions. Furthermore, rotationally invariant DE applied to non-separable problems is limited to rather low decision space dimensions and is highly dependent on population size. In contrast, the CSDE approach is insensitive to population size on the test problems used, even though it does not employ crossover in the traditional sense. It can also handle problems with parameter interactions in high dimensional spaces very well even though it is not a strict rotationally invariant algorithm. The results presented in this paper are significantly important to practitioners who are interested in

optimizing non-separable problems. Until now, previous work in this area focussed on computationally expensive Evolutionary Strategy techniques.

We have presented a computationally efficient, simple optimization algorithm for dramatically improving optimization performance on non-separable problems in high dimensional spaces. Although the comparative evaluation presented here was rather limited and only showed a comparison with the DE/rand/1/bin algorithm, the results are very promising. We intend to perform a more comprehensive study with a variety of DE algorithms and test problems in a future study.

References

1. Salomon, R.: Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions: A survey of some theoretical and practical aspects of genetic algorithms. Bio. Systems 39(3), 263–278 (1996)
2. Price, K.: Differential evolution: a fast and simple numerical optimizer. In: Biennial Conference of the North American Fuzzy Information Processing Society, New York, vol. 3339, pp. 524–527 (1996)
3. Sutton, A.M., Lunacek, M., Whitley, L.D.: Differential evolution and non-separability: using selective pressure to focus search. In: GECCO 2007: Proceedings of the 9th annual conference on Genetic and evolutionary computation, pp. 1428–1435 (2007)
4. Lampinen, J., Zelinka, I.: On stagnation of the differential evolution algorithm. In: Proceedings of MENDEL 2000, 6th International Mendel Conference on Soft Computing, pp. 76–83 (2000)
5. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6(2), 182–197 (2002)
6. Deb, K., Goyal, M.: A combined genetic adaptive search (GENEAS) for engineering design. Computer Science and Informatics 26(4), 30–45 (1995)
7. Iorio, A., Li, X.: Rotated test problems for assessing the performance of multi-objective optimization algorithms. In: GECCO 2006: Proceedings of the 8th annual conference on Genetic and evolutionary computation, pp. 683–690 (2006)
8. Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms - a comparative case study. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) PPSN 1998. LNCS, vol. 1498, pp. 292–304. Springer, Heidelberg (1998)