Dynamic Uniform Scaling for Multiobjective Genetic Algorithms

Gerulf K. M. Pedersen[1] and David E. Goldberg[2]

[1] Aalborg University, Department of Control Engineering, Fredrik Bajers Vej 7, DK-9220 Aalborg Ø, Denmark
[2] University of Illinois at Urbana-Champaign, Department of General Engineering, 104 S. Mathews Ave., Urbana IL 61801, USA

Abstract. Before Multiobjective Evolutionary Algorithms (MOEAs) can be used as a widespread tool for solving arbitrary real world problems there are some salient issues which require further investigation. One of these issues is how a uniform distribution of solutions along the Pareto non-dominated front can be obtained for badly scaled objective functions. This is especially a problem if the bounds for the objective functions are unknown, which may result in the non-dominated solutions found by the MOEA to be biased towards one objective, thus resulting in a less diverse set of tradeoffs. In this paper, the issue of obtaining a diverse set of solutions for badly scaled objective functions will be investigated and the proposed solutions will be implemented using the NSGA-II algorithm.

1 Introduction

Multiobjective EAs (MOEAs) have been applied in a variety of areas. From design of airframes [1] to economic load dispatch problems in power systems [2] and over evolutionary path planners [3], MOEAs are becoming an important tool in practical optimization and decision making. Moreover, MOEAs themselves have received empirical and theoretical study which is well summarized in two monographs devoted to MOEAs [4, 5].

Despite the rise in application, implementation, and theoretical interest, an important consideration in developing broadly capable MOEAs appear to have received scant attention. In particular, many MOEAs use a distance metric in an attempt to ensure a uniform distribution of individuals along the Pareto front, but the individual objective functions may or may not operate over a comparable scale. As a result, it is important to explicitly consider and adapt to widely disparate scalings among different objectives. Here, we will examine the performance of MOEAs when objective functions are badly scaled and consider dynamic uniform scaling procedures to solve such difficulty.

First we consider how badly scaled problems might pose a problem for MOEAs. Then the crowding mechanism of NSGA-II [6] will be investigated with regard to badly scaled problems, and some alterations for calculating the crowding distance will be suggested. The proposed changes will then be tested and the result will be presented, followed by some concluding remarks.

2 Background

Most MOEAs use a distance metric in objective space in order to maintain diversity for the non-dominated solutions on the Pareto optimal front. By ensuring diversity among the non-dominated solutions, it is possible to choose from a variety of solutions when attempting to solve a specific problem at hand.

Suppose we have two objective functions $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$. In this case we can define the distance metric as the Euclidean distance in objective space between two neighboring individuals and we thus obtain a distance given by

$$d_f^2(\boldsymbol{x_1}, \boldsymbol{x_2}) = [f_1(\boldsymbol{x_1}) - f_1(\boldsymbol{x_2})]^2 + [f_2(\boldsymbol{x_1}) - f_2(\boldsymbol{x_2})]^2 \quad . \tag{1}$$

where x_1 and x_2 are two distinct individuals that are neighboring in objective space. If the functions are badly scaled, e.g. $[\Delta f_1(\boldsymbol{x})]^2 \gg [\Delta f_2(\boldsymbol{x})]^2$, the distance metric can be approximated to

$$d_f^2(\boldsymbol{x_1}, \boldsymbol{x_2}) \approx [f_1(\boldsymbol{x_1}) - f_1(\boldsymbol{x_2})]^2 \quad . \tag{2}$$

In some cases this approximation will result in an acceptable spread of solutions along the Pareto front, especially for small gradual slope changes as shown in the illustrated example in Fig. 1.

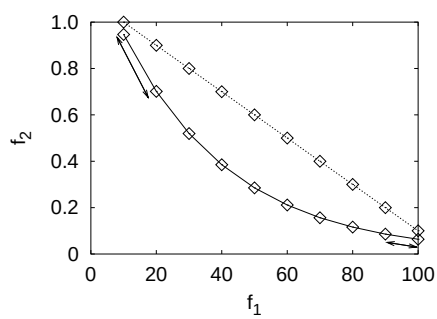
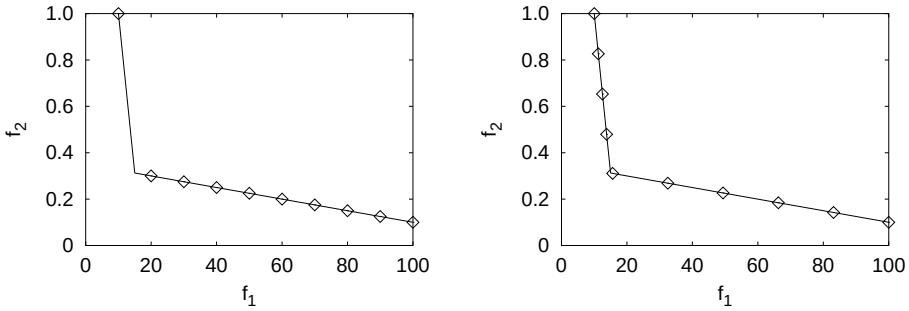


Fig. 1. For fronts with small gradual slope changes an acceptable distribution can be obtained even if one of the objectives (in this case f_2) is neglected from the distance calculations.

As can be seen in the figure, the distances marked by the arrows are not equal, but the solutions can still be seen to cover the front relatively well.

In other cases, however, the result is that a significant portion of the Pareto front is ignored, especially for large sudden changes of slope. This can be seen clearly in the illustration given in Fig. 2a, where only one objective function is used for calculation of the crowding distance thereby ignoring a large portion of the Pareto front along $f_2(\boldsymbol{x})$.

The method proposed in this paper will try to deal with this issue such that the bad distribution illustrated in Fig. 2a can be avoided and instead be replaced by one similar to the one shown in Fig. 2b. First, however, it is necessary to get a better understanding of the distributions themselves.



(a) Neglecting one of the objectives for distance calculation results in a bad distribution.

(b) An ideal uniform distribution.

Fig. 2. Illustration of distributions on a piecewise linear front with bad scaling.

3 Distributions

When finding the Pareto non-dominated front for a multiobjective problem, the desired distribution of solution points on the Pareto front might differ depending on the problem being solved. In this paper it is desired for the solutions on the Pareto front to have a uniform distribution along the entire front. Such a uniform distribution is desirable, because it would provide the user with a good estimate of the tradeoffs between the different objectives. In case a non-uniform distribution is desired, it can be obtained by applying a transformation to the uniformly distributed points, but that will not be discussed further since it is out of the scope of this paper.

3.1 Uniform Distribution

The uniform distribution aims at generating a uniform spacing between each solution point on the Pareto non-dominated front. However, there are two different aspects of uniform distributions which must be considered separately, those with known bounds, which includes those problems with known scaling factors, and those with unknown bounds.

Known Bounds Throughout the last decade, a lot of research has gone into finding crowding mechanisms, or other methods, that would result in a uniform distribution of non-dominated solutions along the Pareto front when the bounds of the individual objective functions are known or can be approximated [7, 8]. For the case of a problem with known bounds, the bounds can be used to normalize the objective functions, resulting in some very efficient crowding methods that ensure a good uniform distribution of points.

These crowding methods with known scaling factors do have some limitations becoming apparent later in this paper. They will give unsatisfactory result for badly scaled problems with unknown bounds. If a MOEA is used to find tradeoffs for a real-world design problem, the probability of obtaining this unsatisfactory result is quite high, since that is the situation in which the unknown bounds are most likely to be encountered.

In the public available version of NSGA-II [6], the crowding distance, d_j, for an individual, j, is calculated using hypercubes. The calculation of the hypercubes is given by sorting the individuals in ascending order, according to the objective value of the ith objective, f_i. The crowding distance for individual j is then calculated using

$$d_j = d_j + \frac{f_{i,j+1} - f_{i,j-1}}{f_{i,max} - f_{i,min}} \quad . \tag{3}$$

For the case where the objective value for an individual is either the highest or the lowest amongst the population, the crowding distance is set to ∞. Such a policy ensures that an individual at the edge of the Pareto front will be preferred to an individual fully contained within the edges of the Pareto front. The values for $f_{i,min}$ and $f_{i,max}$ are in the literature recommended to be set equal to the maximum obtainable values for the objective function f_i [5].

If the normalization by $f_{i,min}$ and $f_{i,max}$ was not done, the crowding distance would be dominated by the objective functions with the largest values. Hence the distribution of the solutions on the Pareto front would be biased towards those objectives. As long as the bounds of the objective functions are known, then, as earlier mentioned, this normalization can be used to ensure a good uniform distribution of solution points. However, when the bounds are unknown it is necessary to find another way to normalize the crowding distance calculations.

Unknown Bounds In order to obtain a good distribution we desire to avoid the problem of bad scaling when encountering unknown bounds. By ensuring that each objective can be normalized even when the bounds are unknown, an equal number of significant tradeoffs can be obtained such that no objectives will be able to dominate the others. The crowding distance for unknown bounded scaled hypercubes can still be calculated using (3). However, the values for $f_{i,min}$ and $f_{i,max}$, since the bounds are unknown, need to be either fixed at a certain value or changed dynamically.

It would be unwise to use a fixed value for $f_{i,min}$ and $f_{i,max}$ when the bounds are unknown, since the lack of information about the objective functions would impair the effort to choose good values. Changing the values dynamically will, however, require that a scheme for updating the values are derived. Two methods for updating the values will be presented here.

The normalization can be done either locally for each front or globally for all fronts. When done locally the maximum and minimum values for each objective in each front are used to normalize the crowding distances for that front. When done globally for all fronts, the maximum and minimum values for each objective in the entire population are used for normalization. Therefore, by using one of those normalization techniques, it should thus be possible to obtain a good uniform distribution for the solutions along the different fronts when the bounds are unknown. With the two normalization schemes in place it is now time to take a closer look at the setup of the experiments.

4 Experimental Setup

The emphasis in this paper is focused on investigating how to successfully normalize the objective functions for the problems with unknown bounds when used in the crowding distance calculations. The goal is to obtain a uniform distribution along the Pareto non-dominated front for these problems.

Objective Functions For simplicity this paper will only consider cases consisting of two objectives. The results are also valid for higher dimension cases, but the use of only two objective functions is chosen to ensure a straightforward visualization of the results obtained.

The objective functions are deliberately chosen such that the optimal Pareto-front is known, and such that bad scaling can be easily obtained. This allows for a direct comparison of the optimal solutions with those found by the algorithm, which will provide a help in fully understanding how the points are distributed along the Pareto-front and may also shed light on possible issues that could be further improved.

To illustrate the problems with badly scaled objective functions, the objective functions for this paper are chosen such that the relative scaling of the objective functions can be easily varied. To achieve this, the objective functions are given by

$$f_1 = x_1^k + |x_2|, \quad k > 0 \qquad \text{and} \qquad f_2 = x_1^{-l} + |x_2|, \qquad x_1 > 0, \;\; l > 0 \;\; , \tag{4}$$

where $x, y \in \mathbb{R}$ are the optimization variables, and k and l determine the extent of scaling between the functions. The optimal Pareto-front is obtained for

$$f_2 = f_1^{-\gamma} \qquad x_2 = 0 \;\; , \tag{5}$$

where $\gamma = \frac{l}{k}$. Thus, for small values of γ ($\gamma < 1$), f_1 will have bigger scaling than f_2 and vice versa. Optimal scaling between objective functions will be obtained for $\gamma = 1$.

Variables The number of variables is two, as shown in (4), and the encoding of both variables is binary. A real-valued encoding should be able to produce similar results for this simple problem. However, that is out of the scope of this paper, since it depends on the operators and their implementation.

The representation for variable x_1 is chosen to be 16 bit long belonging to $[0.1, 10]$, so x_1 complies with the constraints given in (4) and assures that the resulting non-dominated fronts found using NSGA-II will not have excessive extreme values. The variable x_2 is represented using 16 bits and, since the range can be chosen arbitrarily, we set it to $[-100, 100]$. The specified ranges allows for the calculation of the maximum and minimum values of both objective functions. It is necessary to emphasize that this assumption of known bounds does not in any way affect the conclusion obtained in this paper, since the paper is meant to illuminate potential problems, showing measures of how they might be avoided for real world problems.

Algorithmic Setup The algorithm chosen for the investigations in this paper is the NSGA-II developed by Kalyanmoy Deb. NSGA-II was chosen since it is an overall good MOEA with respect to several different classes of problems [9, 10].

The selection operator is not expected to have a major influence on the results derived in this paper. The only requirements to the selection operator are that solutions belonging to a lower ranked front should be preferred over solutions belonging to higher ranked fronts, and in case the solutions belong to the same front the one with the largest crowding distance should be preferred. Since the selection method used in NSGA-II is tournament selection with size 2, which meets the specified requirements, that setting is used without modification.

The problem defined in (4) is very simple and it is thus not expected that using different types of crossover will influence the performance of the algorithm too much. Thus, the crossover operator is chosen to be uniform. The crossover probability p_c is chosen to be 0.9 which should ensure good convergence for this simple problem [11].

With the binary representation, the use of bitwise mutation is straightforward to implement and is expected to produce results that will be similar to those obtained if using other more complex mutation methods. The mutation probability p_m is set to 0.01.

Using the previously defined parameters it can be determined that the search space is comprised of 2^{32} possible solutions, whereas, when the known optimal Pareto front is taken into account, the number of Pareto non-dominated solutions consists of 2^{16} points. With a population size of 200 individuals it will be possible to cover 0.3% of the non-dominated front if the entire population belongs to that front. A coverage that loose will give ample possibility to fully investigate how the crowding mechanisms will perform while still providing a good estimate of the Pareto non-dominated front, which in this case is known to be continuous and smooth.

The algorithm is run for 200 generations, which should be sufficient for finding solutions on the true Pareto non-dominated front, and also to apply the crowding measures which will spread the solutions uniformly along the front.

30 independent trials, with differing random seeds, will be conducted for each experiment. This is done to ensure that no individual run will be able to overly influence the results of an experiment and the conclusions drawn from those results.

The parameters used to produce the results of this paper are summarized in table 1, and with the experimental setup now in place we can move forward to the experiments.

5 Results

In this section the results of the proposed method are compared to the original NSGA-II algorithm. For all of the tests performed, the algorithm succeeded in finding the actual Pareto fronts with the entire population, all 200 solution points, located on said front. Also, the algorithm always succeeded in finding the outermost solution points for the Pareto front, and for this reason the calculation of the spread on the Pareto front will not include any terms penalizing the lack in finding the outermost solutions.

Table 1. Parameters used for NSGA-II.

Parameter description	Value	Designation
Number of objectives	2	f_1, f_2
Format of variables	Binary	-
Number of variables	2	x_1, x_2
No. of bits for x_1 variable	16	l_{x_1}
No. of bits for x_2 variable	16	l_{x_2}
Range for x_1 variable	[0.1,10]	$[x_{1_{min}}, x_{1_{max}}]$
Range for x_2 variable	[-100,100]	$[x_{2_{min}}, x_{2_{max}}]$
Selection operator	Tournament	-
Tournament size	2	s
Crossover type	Uniform	-
Crossover probability	0.9	p_c
Mutation type	Bitwise	-
Mutation probability	0.01	p_m
Population size	200	n_{pop}
Maximum generations	200	n_{max}

5.1 Original NSGA-II

The algorithm was first run using 3 distinct γ-values (0.2, 1, and 5) with the original crowding distance calculations. It was then possible to illustrate the effects of a bad scaling for each of the cases where f_1 and f_2 were overly emphasized. It was also possible to show what resulted for the optimal situation, with equal scaling of the objectives. The results for the 3 different γ-values can be seen in Fig. 3, 4a and 4b, where the true fronts are also shown[3].

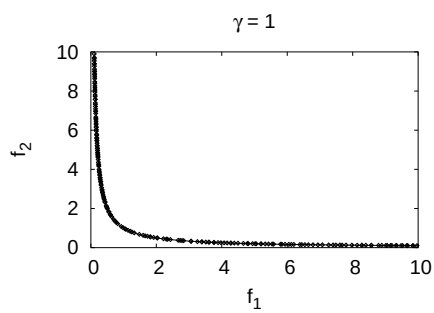


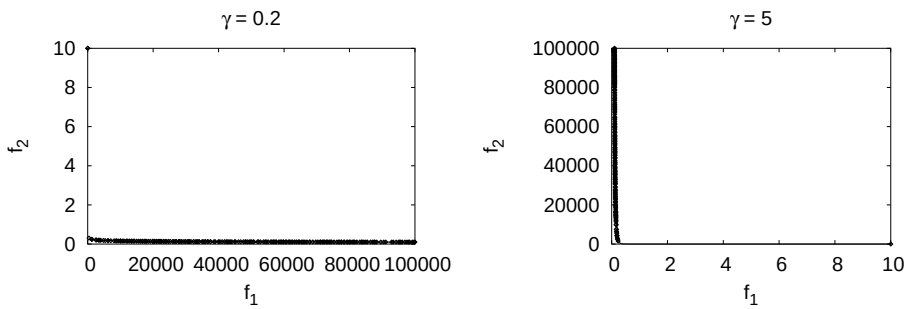
Fig. 3. Pareto non-dominated front for unity γ value plotted on the known optimal front.

From figure 3 it can be seen that when the objective functions have equal scaling ($\gamma = 1$), then NSGA-II is able to find the true Pareto optimal front with a uniform distribution of points which preserves the tradeoffs for both objective functions. The spread of the distribution for a single run can be calculated using

[3] All of the result graphs presented in this paper are based on a run with a random seed of 0.1234.

$$spr = \sum_{j=1}^{n_{pop}} \left(d_j - \bar{d}\right)^2 \quad , \tag{6}$$

where $\bar{d}$ is the mean of the normalized Euclidean distances between the solution points on the graph. This spread metric is a modified version of the one used in [5], the main difference being the normalization, which is not used in [5]. Due to the simplicity of the problem it has been deemed unnecessary to include the distance between the outermost solutions and the known extremes of the Pareto front, since those in all experiments coincided, thus giving an added value of zero. The normalization in (6) is done according to $f_{i,max} - f_{i,min}$ for both objectives, thus the distance between the minimum and maximum values for an objective after normalization is equal to 1. The spread is then averaged over 30 runs and the resulting mean spread for running NSGA-II with $\gamma = 1$ is $6.373 \cdot 10^{-3}$.



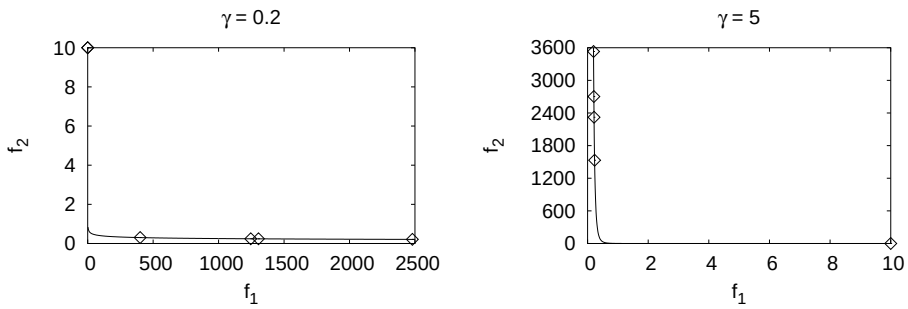
(a) Small γ value.

(b) Large γ value.

Fig. 4. Pareto non-dominated fronts for large and small γ values plotted on the known optimal front.

As seen in Fig. 4 it is clear that even though the Pareto optimal front is found the distribution of points are not uniform for both objectives. In other words, in Fig. 4a the tradeoffs for objective f_2 is almost non-existent, since all of the solution points are located in the direction of objective function f_1. This can also be seen from the fact that the spread for this case is $924.8 \cdot 10^{-3}$, which is much higher than that for $\gamma = 1$. Similarly the opposite can be seen in Fig. 4b, where the solution points are distributed towards objective function f_2 and the spread is found to be $942.7 \cdot 10^{-3}$.

The problem for both of the cases $(\gamma = 0.2, \gamma = 5)$ lies in the fact that the bounds are unknown which causes the crowding distance, calculated using 3, to overly emphasize the objective function with the highest values. The details of this can be seen more clearly in Fig. 5a and 5b, which gives a detailed view of the lower portion of the objective space for γ-values of 0.2 and 5. In these figures it is clear that the crowding distance is dominated by the objective with the largest scale, which is equivalent to the situation described in section 2.



(a) Small γ value with f_1 values less than 2500.

(b) Large γ value with f_2 values less than 3600.

Fig. 5. Lower region of the Pareto non-dominated front for small and large γ values.

It is now clear that when the objective functions with unknown bounds are badly scaled the resulting non-dominated Pareto front does not have a distribution of points which allow for determining proper tradeoffs between the objective functions, since the tradeoffs for one objective totally overshadows the other. This can be concluded since no actual tradeoffs are shown on the Pareto front. As such, it is now time to take a look at the proposed approaches.

5.2 Global Scaling

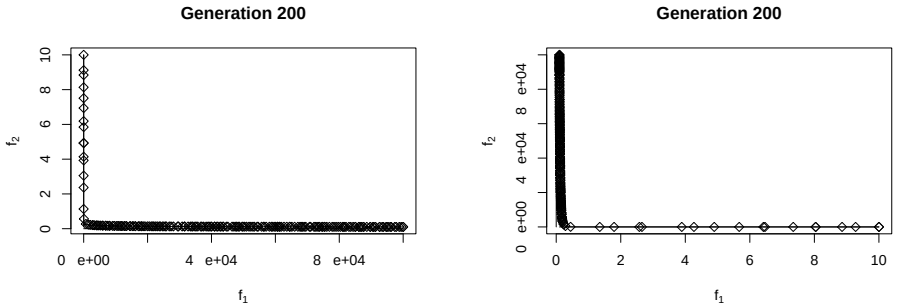
The modified crowding measure proposed, where the normalization values of $f_{i,min}$ and $f_{i,max}$ were set according to either global or local minimum and maximum values, was then implemented and tested on the same problem.

The results obtained using the global normalization scheme can be seen in Fig. 6a and 6b. As seen on the figures, the global normalization is capable of maintaining a small number of Pareto non-dominated solutions for the objectives with the smaller objective values. This means that the global normalization did help in producing a better distributed set of solutions, nevertheless it is also evident that the problem was only partly solved, since the majority of solutions is still concentrated on the objective with the highest fitness values. This is further emphasized by the calculated spreads which are found to be $71.00 \cdot 10^{-3}$ and $70.29 \cdot 10^{-3}$ for γ-values 0.2 and 5 respectively.

Results for $\gamma = 1$ were also obtained but they did not differ significantly from the results obtained using the original crowding calculations of NSGA-II, with a spread of $6.833 \cdot 10^{-3}$.

5.3 Local Scaling

The algorithm was then run using the other proposed method, the locally based normalization, which resulted in Fig. 7a and 7b.



(a) Small γ value.

(b) Large γ value.

Fig. 6. Pareto non-dominated fronts for small and large γ values when using globally based normalization.

It is very clear from the figures that the locally based normalization resulted in a distribution of solution points which is almost uniformly distributed for both objective functions. This is further emphasized by the fact that the spreads are found to be $8.116 \cdot 10^{-3}$ and $7.912 \cdot 10^{-3}$ for γ-values of 0.2 and 5 respectively.

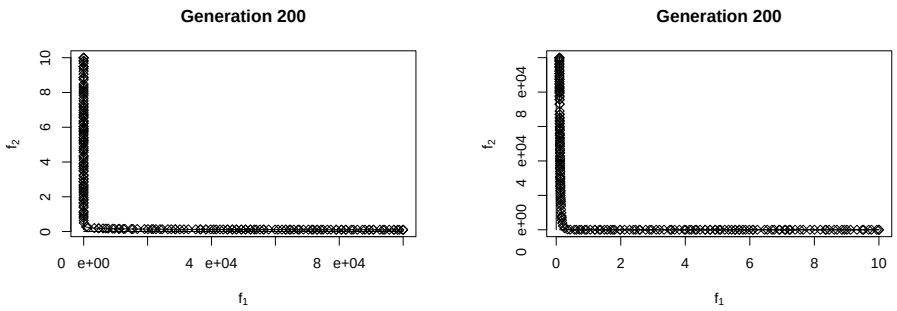
Thus, by using a locally based normalization, when calculating the crowding distances, it was possible to ensure a nearly uniform distribution for a badly scaled problem with unknown bounds. Results were also obtained for $\gamma = 1$ but once again they did not display results that differed significantly from those obtained when using the original crowding calculations and the spread for this case was $6.439 \cdot 10^{-3}$.

5.4 Discussion

The results obtained are summarized in table 2. A salient issue remaining to be explained is how the globally based normalization of the crowding distances were able to partially solve the problem of the non-uniform distribution of solutions on the Pareto front. In order to get a full understanding of that issue it is necessary take a look at how the MOEA itself is implemented.

Table 2. Spreads obtained using different γ-values for original crowding distance calculations and for globally and locally based normalization of crowding distances.

Crowding method	$\gamma = 0.2$	$\gamma = 1$	$\gamma = 5$
Original	$924.8 \cdot 10^{-3}$	$6.373 \cdot 10^{-3}$	$942.7 \cdot 10^{-3}$
Global	$71.00 \cdot 10^{-3}$	$6.833 \cdot 10^{-3}$	$70.29 \cdot 10^{-3}$
Local	$8.116 \cdot 10^{-3}$	$6.439 \cdot 10^{-3}$	$7.912 \cdot 10^{-3}$



(a) Small γ value.

(b) Large γ value.

Fig. 7. Pareto non-dominated front for small and large γ values when using locally based normalization.

The vital part of the explanation lies in the fact that the selection in NSGA-II is based on Pareto ranking. In many cases the solution space with the Pareto non-dominated solutions might only be a subset of the objective space. Due to the ranking method, the subsequent fronts will always have higher values for at least one objective compared to the solutions on the fronts that dominate the one currently considered. This means that the subsequent fronts might span a set that is bigger than that spanned by the fronts dominating the current ones. As a result, when the normalization parameter was calculated globally, it was found to be the maximum value of the set given by the union of all fronts. Thus, if the application of crossover and mutation had resulted in solutions that belonged to subsequent dominated fronts, which is highly likely, especially for the Pareto non-dominated front, then the normalization values was calculated based on values that most likely would be too large. The normalization parameter would thus also be too large and the distribution would be biased away from that objective accordingly.

For the locally based dynamic scaling, the scaling was calculated using only values already present in the front under consideration. Thus, higher values in subsequent front would have no effect on the current front and no biased scaling would occur. The crowding distances calculated for those fronts would then be independent of the relative scaling of the objective functions and a uniform distribution can then be obtained for the badly scaled problems.

6 Conclusion

In this paper we investigated the effect of badly scaled objective functions on the distribution of solution points along the Pareto front with unknown objective function bounds. To correct the problem, two different dynamic scaling methods were implemented in NSGA-II and the results showed that one of the methods was able to successfully achieve a near uniform distribution of solution points along the Pareto front.

It is clear from the obtained results that when dealing with a badly scaled problem with unknown bounds then problems can arise. The results also show that, in order to

obtain a uniform distribution of solution points for both objectives, dynamic scaling based on locally normalization will give the desired result. As a result, the proposed dynamic scaling proposed in this paper should be remembered when designing new MOEAs or if an existing MOEA is used for a badly scaled optimization problem. This will also help in bridging the gap between MOEA theory and MOEAs applied for real world problems, since the amount of knowledge required to implement the MOEA for a real world problem is reduced.

7 Acknowledgments

The authors would like to thank Kumara Sastry and Dr. Xavier Llorà for helpful comments and suggestions. Additionally we would like to thank the reviewers.

References

1. Parmee, I.C., Watson, A.H.: Preliminary airframe design using co-evolutionary multiobjective genetic algorithms. Proceedings of the Genetic and Evolutionary Computation Conference 1999 **2** (1999) 1657–1665
2. Das, D.B., Patvardhan, C.: New multi-objective stochastic search technique for economic load dispatch. IEEE Proceedings of Generation, Transmission and Distribution **145** (1998) 747–752
3. Dozier, G., McCullough, S., Homaifar, A., Tunstel, E., Moore, L.: Multiobjective evolutionary path planning via fuzzy tournament selection. Proceedings of 1998 IEEE International Conference on Evolutionary Computation (1998) 684–689
4. Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multi-Objective Problems. Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers, New York, NY (2002)
5. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. 1st edn. John Wiley & Sons, Ltd., West Sussex, England (2001)
6. Deb, K., Pratap, A., Moitra, S.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. Parallel Problem Solving from Nature - PPSN VI (2000) 849–858 NSGA-II code available at KanGAL website: 'http://www.iitk.ac.in/kangal/'.
7. Obayashi, S.: Pareto genetic algorithm for aerodynamic design using the Navier-Stokes equations. In Quagliarella, D., Périaux, J., Poloni, C., Winter, G., eds.: Genetic Algorithms and Evolution Strategies in Engineering and Computer Science. John Wiley & Sons, Ltd., Trieste, Italy (1997) 245–266
8. Deb, K., Mohan, M., Mishra, S.: A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions. Technical Report 2003002, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, Kanpur, PIN 208016, India (2003)
9. Deb, K., Jain, S.: Multi-speed gearbox design using multi-objective evolutionary algorithms. Technical Report 2002001, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, Kanpur, PIN 208016, India (2002)
10. Deb, K.: Unveiling innovative design principles by means of multiple conflicting objectives. Technical Report 2002007, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, Kanpur, PIN 208016, India (2002)
11. Goldberg, D.E.: The Design of Innovation: Lessons from and for Competent Genetic Algorithms. Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers, Norwell, MA (2002)