Blended Ranking to Cross Infeasible Regions in Constrained Multiobjective Problems

Nicholas Young
Central Queensland University
Rockhampton, QLD 4701 Australia
n.young@cqu.edu.au

Abstract

We present a multiobjective evolutionary algorithm designed to reliably cross infeasible regions of objective space and find the true constrained Pareto front, which may lie across multiple disconnected feasible regions. By blending an individual's rank in objective space with its rank in constraint space, some infeasible solutions may be selected over some feasible solutions, allowing the population to traverse infeasible regions smoothly. Results from artificial benchmark problems qualitatively illustrate this behaviour, in contrast to NSGA-II which must cross infeasible regions in a single generation.

1. Introduction

Many real world problems can be formulated as constrained multiobjective optimisation problems:

$$\text{minimise } f_i(x), \qquad i = 1,\ldots,M \tag{1}$$

$$\begin{aligned}\text{subject to } g_j(x) \le 0, &\quad j = 1,\ldots,N \\ \text{and } h_k(x) = 0, &\quad k = 1,\ldots,L\end{aligned} \tag{2}$$

It is common to reformulate equality constraints as inequality constraints (eg. as $g_i(x) \equiv |h_i(x)| \le 0$), and so we consider only inequality constraints in this paper.

Unconstrained multiobjective evolutionary algorithms (EAs), which attempt to solve just (1), have been extensively studied; see for example the survey [1].

Most multiobjective EAs use the notion of Pareto dominance. One solution, a, *dominates* another solution, b, written as:

$$a \succ b \quad \text{iff} \quad \forall i \in \{1,\ldots,M\}, f_i(a) \le f_i(b) \\ \wedge\ \exists i \in \{1,\ldots,M\} \mid f_i(a) < f_i(b)$$

that is, a is at least as good as b on all objectives, and is better on at least one. If neither $a \succ b$ nor $b \succ a$ then $a \sim b$, that is a and b are *non-dominated*.

There may be many solutions to (1) such that it is impossible to improve the solution's performance on one objective, f_i, without degrading its performance on another objective, f_j. These solutions are non-dominated with respect to each other. The set of all such solutions is called the Pareto front.

Most multiobjective EAs are designed to find a set of non-dominated solutions that approximate the true Pareto front. A good algorithm finds a set that is close to the Pareto front, covers all extremes of the front, and has an even distribution of points across the front.

Constrained multiobjective EAs that solve the constrained problem (1) subject to (2) have received growing attention recently; see for example the survey [2], and specific multiobjective EAs [3-5].

2. Background

A constrained multiobjective EA can operate in multiple spaces. A solution x is a single point in decision space (plotted by decision variables on the chromosome); objective space (plotted by objective function scores $f_{1\ldots M}(x)$); and constraint space (plotted by constraint scores $g_{1\ldots N}(x)$).

Pareto dominance ranking and diversity can be measured in objective space. Dominance ranking [6] is one way of assigning fitness to a solution in a multiobjective problem. It is an iterative process: those solutions in the population that are non-dominated are assigned to the first rank. They are then removed from consideration, and those solutions that previously were dominated only by non-dominated solutions become the new non-dominated solutions and are assigned to the next rank. They are then removed, and the process continues until all solutions in the population are ranked. A solution's fitness is simply its rank.

Dominance ranking provides the selection pressure to move through objective space towards the Pareto front, but provides no pressure to spread out along the Pareto front. It is combined with some measure of diversity that is usually internal to each rank. (For example, NSGA-II [5] measures the average distance to a solution's nearest neighbours over all objective dimensions.) Either the diversity measure modifies the

Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Confe
Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05)
0-7695-2504-0/05 $20.00 © 2005 IEEE

fitness such that the modification by diversity is less than the difference between two ranks, or the EA's selection operators are modified to compare first on rank and second on diversity. Assuming a tournament selection operator, the end effect is the same.

Constrained EAs can handle constraints in many ways: the EA's variation operators (crossover and mutation) can be written to avoid creating infeasible solutions; the EA can introduce an additional operator to repair infeasible solutions; infeasible solutions can be discarded; or infeasible solutions can be retained in the population and used in the search process.

Of EAs that retain infeasible solutions, some algorithms do not treat constraint scores as a multidimensional space and simply sum or min-max constraint violations (collapsing the space into one dimension) or apply penalties back onto objective scores (distorting the objective space). Other algorithms do treat constraint space as a space in its own right, in which Pareto dominance ranking can be performed.

Constrained multiobjective EAs need to work with both constraint space and objective space. There are multiple ways for the spaces to interact.

Many algorithms use an extended dominance relation [7] that bridges together objective space and constraint space to form an extended space in which to perform one ranking. The extended dominance relation is:

1. If both solutions are feasible, dominance is measured in objective space.
2. If one solution is feasible and the other is infeasible, the feasible solution dominates.
3. If both solutions are infeasible, dominance is measured in constraint space.

The effect of this dominance relation is to ensure that a non-dominated rank is either all feasible or all infeasible, and that all feasible ranks are better than all infeasible ranks. NSGA-II [5] uses this extended dominance relation, but uses a sum of normalised constraint violations rather than Pareto dominance in constraint space.

The approach used in this paper is blended space, as proposed by Angantyr, Andersson, & Aidanpaa in [8], though their unnamed algorithm was single-objective. In an algorithm using blended space, every solution is assigned two Pareto dominance ranks: one calculated in objective space, R_O, and the other in constraint space, R_C. The two ranks are blended together to give one blended rank which can be used to compare solutions for reproductive selection and survival. The blended rank, R_B, is defined as:

$$R_B = \alpha R_O + (1-\alpha) R_C \tag{3}$$

$$\text{where } \alpha = \frac{\text{num feasible solutions}}{\text{population size}} \tag{4}$$

α acts as a weight to place emphasis on either progress in objective space or progress in constraint space, but it is not set by the experimenter. α varies dynamically so that if the population is entirely infeasible (α = 0) all emphasis is placed on minimising the constraint rank – searching for a feasible solution – thereby increasing the number of feasible solutions and increasing α. Conversely, if the population is entirely feasible (α = 1) all emphasis is placed on minimising the objective rank – searching for the optimal solution – likely increasing the number of infeasible solutions and decreasing α. α varies dynamically somewhere in between during the run of the algorithm. The effect is that the population will straddle the constraint boundaries that separate feasible region from the unconstrained optimum. One drawback in a multiobjective problem is that only part of the final population will be feasible and useful in identifying the Pareto front.

3. Motivations for Research

NSGA-II, which uses the extended dominance relation, has proved to be very effective, even when the population is initially infeasible. If the population is initially infeasible, the extended dominance relation will cause the population to first move towards feasible regions of objective space, then find the constrained Pareto front, and then spread out along it. The population will consistently form up into one non-dominated rank spread evenly across the Pareto front.

One potential weakness of elitist extended dominance algorithms such as NSGA-II is the greedy preference for feasible solutions over infeasible solutions. As soon as the algorithm finds a feasible region of space, the population will quickly converge to that area abandoning all infeasible solutions. This is a problem when the constrained Pareto front lies across multiple discrete feasible regions; the algorithm may converge to the first feasible region it finds. Any new solutions that are subsequently generated anywhere in infeasible regions are discarded because they are dominated by the existing feasible solutions that fill the population. The only mechanism to escape a discrete feasible region is to generate a new solution by just one crossover and mutation operation that "jumps" into another feasible region and is non-dominated by the existing solutions. However, depending on the crossover and mutation rates and, more importantly, the mutation magnitude, it may be highly unlikely to generate a new solution in a non-dominated feasible region in a single step.

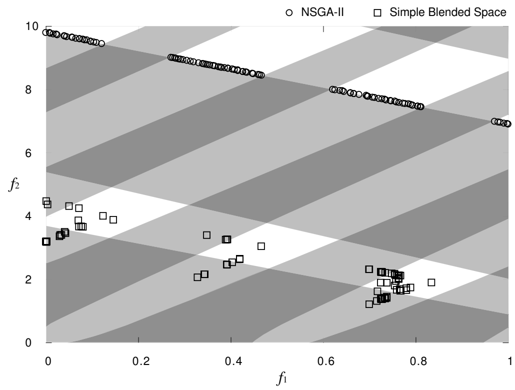

Fig. 1. A two-objective, two-constraint problem based on CTP in [9]. Infeasible regions are shaded and are darker where constraints overlap. Feasible regions are white. a) NSGA-II stuck behind an infeasible region, with little chance of mutation allowing it to jump down to the true Pareto front in a single generation. However, NSGA-II does exhibit excellent coverage. b) A simple multi-objective implementation of [8]. Selection based on blended space enables it to cross the infeasible region that blocked NSGA-II, but coverage of the Pareto front is poor.

This behaviour was observed in both NSGA-II and a version of the algorithm presented in this paper that used the extended dominance relation.

Blended space algorithms promise the ability to cross large infeasible regions in small, evolutionary steps. An infeasible solution can trade off a bad rank in constraint space for a good rank in objective space, and the blending parameter α ensures that there is a balance of feasible and infeasible solutions in the population.

The single-objective blended space algorithm proposed in [8] has several limitations when applied to a multiobjective problem. Fig. 1 shows the typical behaviour of a straightforward multiobjective implementation of [8]. It successfully finds the feasible regions in which the true Pareto front lies, but coverage is poor. The poor coverage is attributable to multiple interacting causes: 1) The blended space algorithm actively avoids collapsing to a single non-dominated rank because the dynamic α parameter will keep a mix of feasible and infeasible solutions in the population. 2) The diversity measure always influences the stochastic (weak) reproductive selection, but it can only influence the elitist (strong) survival selection when the elite rank fills the entire population, which will never happen due to α. 3) Ranking in constraint space differentiates feasible solutions as well as infeasible solutions into multiple ranks (in effect, it continues to try to optimise constraints even after they are all satisfied). This moves solutions away from the constraint boundary into the middle of the feasible regions, as well as creating even more ranks that reduce the effectiveness of the diversity measure.

This paper's Blended Space Evolutionary Algorithm (BSEA) is a robust constrained multiobjective algorithm designed to cross infeasible regions to find the true and entire (possibly disconnected) constrained Pareto front.

4. BSEA

The following is a description of this paper's Blended Space Evolutionary Algorithm (BSEA).

4.1. Overview

The EA uses a population of fixed size P from generation to generation. At each generation:

1. Selection for reproduction: $P/2$ pairs of parents are selected, with replacement, using binary tournament selection. Tournament selection compares blended rank first, and if solutions tie on blended rank, it compares diversity measure second.
2. Each pair of parents generates a pair of children using crossover and mutation suitable for the decision variables (in this study, 1-point crossover and Gaussian mutation). This process generates P children, which are evaluated and added to the population. Duplicate solutions are removed from the population. After this stage, parents and children in the expanded population compete equally for survival.
3. Every solution is ranked in objective space (R_O) and constraint space (R_C); the value of α is calculated; and the final blended rank is calculated for every solution.
4. Every solution is given a diversity measure relative to other solutions in the same objective rank. Diversity is only measured in objective space.
5. Selection for survival to the next generation:
   i. Some number of the best (in objective space) feasible solutions are preserved.
   ii. The remainder of the population is sorted by blended rank first and diversity second, and the best are preserved to make up the P solutions.

4.2. Ranking

R_O and R_C are both calculated using Pareto dominance and the ranking procedure in [6]. The only difference is that R_C is calculated using Pareto dominance in constraint space instead of in objective space.

The ranking process in constraint space ranks feasible solutions in exactly the same way as infeasible solutions. The constraint functions $g(x)$ are allowed to be negative, with any constraint $g(x) \leq 0$ indicating the constraint is satisfied. Two feasible solutions may be placed in different constraint ranks, and solutions around the edge of feasible regions are likely to have a worse rank than solutions around the centre of feasible regions. Therefore, R_O and R_C are often in conflict.

BSEA takes the approach of treating constraint space and objective space on equal terms, relying solely on α to mediate the two ranks.

The blended rank is calculated similarly to [8] as (3) and (4). However, in [8], R_O and R_C are integers directly corresponding to the rank number. This creates bias in the blended rank when the number of objective dimensions is different to the number of constraint dimensions. In general the number of ranks in a space is inversely proportional to the number of dimensions, because more dimensions create more ways solutions can be non-dominated relative to one another. To address this, in BSEA the rank values R_O and R_C are normalised to the interval [0,1), causing the blended rank R_B to also fall in [0, 1).

4.3. Diversity

The diversity measure is calculated for each solution, grouped by objective space ranks. The diversity of a solution is the volume of objective space dominated by that solution but not dominated by any other solution in its rank. Solutions that have extreme positions in objective space are assigned an infinite volume to preserve them (and any measure of hypervolume on the extreme points in space would be arbitrary anyway). This is the hypervolume measure proposed in [10] as the basis for admitting solutions to an external archive. The advantage of the hypervolume measure is an insensitivity to scale. In practice hypervolume and NSGA-II's distance-to-neighbours worked equally well on the problems tested.

4.4. Feasible non-dominated Solutions

A limitation of the blended space algorithm, when applied to a multiobjective problem, is that the population actively avoids collapsing into a single non-dominated rank. This behaviour is unfortunate, since a significant part (in practice, most) of the population is useless as a final solution set: many solutions are either infeasible or dominated.

To address this BSEA recognises that, ultimately, extended-dominance-based algorithms produce a solution set with desirable properties: an entire population of feasible, non-dominated solutions, distributed evenly by diversity. Therefore, BSEA reserves up to a certain fraction, r, of the population for feasible non-dominated solutions. If there are too many feasible non-dominated solutions, then those with the highest diversity are preserved, up to the reserved fraction. The remainder of the population is selected as normal.

$$r = \frac{\text{current generation}}{\text{total generations}} \tag{5}$$

The fraction of the population reserved for feasible non-dominated solutions increases linearly over the run of the algorithm, until when the algorithm terminates the entire population is allowed to be feasible and non-dominated. This reservation overrides α, which merely influences the blended rank; however the preserved feasible solutions still count towards α, causing α to rise steadily with r. This relationship was found to be complementary: as the number of infeasible solutions forcibly decreases, α will shift R_B more towards R_O, allowing progressively more unconstrained exploration with the fewer remaining infeasible solutions.

Alternatively, if the number of generations is not known, r should be dependent on whatever other stopping criteria is used.

5. Performance of BSEA

We are primarily interested in the ability of BSEA to cross infeasible regions too large to be reliably crossed by extended space algorithms, such as NSGA-II. However, this must be accomplished without sacrificing performance on problems with different characteristics. The first experiment tests BSEA and NSGA-II on the harder Constrained Test Problems, CTP-6, CTP-7, and CTP-8 found in [9]. The second experiment modifies CTP-8 to ensure that the population must cross infeasible regions to reach the true Pareto front.

In all experiments, for both algorithms, operators and parameters are kept identical (parameters are taken from NSGA-II settings for CTP-6,7,8).

Table 1. Algorithm Parameters

Population size	100
Total generations	1000
Crossover rate	0.9
Mutation rate	0.5
Distribution index for crossover	10
Distribution index for mutation	20

CTP-6, 7, and 8 are all of the form:

Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Confe
Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05)
0-7695-2504-0/05 $20.00 © 2005 IEEE

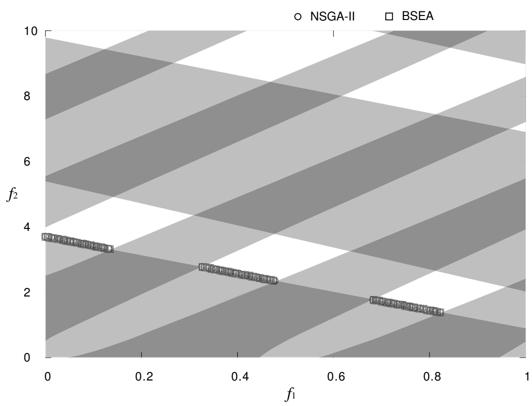

Fig. 2. Final population of NSGA-II and BSEA on CTP-8.

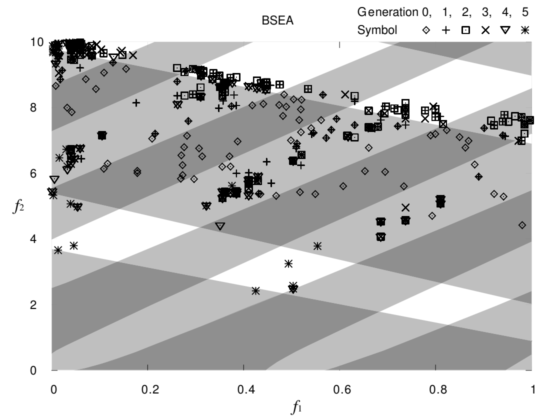

Fig. 4. The first 5 generations of BSEA on the restricted CTP-8.

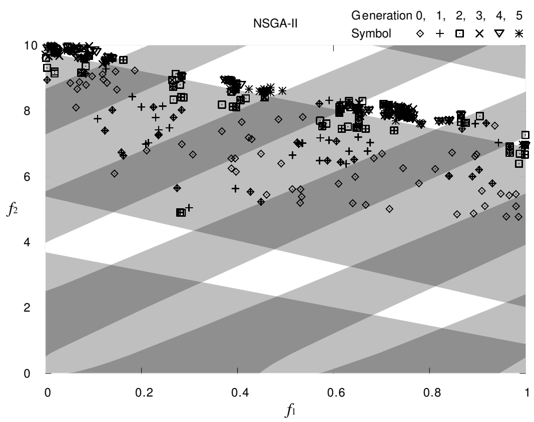

Fig. 3. The first 5 generations of NSGA-II on the restricted CTP-8.

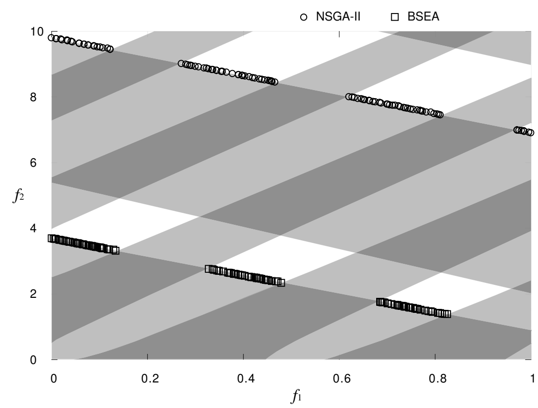

Fig. 5. The final populations after 1000 generations of NSGA-II and BSEA, on the restricted CTP-8.

$$\text{minimise} \quad f_1(x) = x_1$$

$$\text{minimise} \quad f_2(x) = (1.0 + x_2)\left(1.0 - \sqrt{\frac{x_1}{1.0 + x_2}}\right)$$

$$0 < x_1 \le 1 \,;\; 0 < x_2 \le 10$$

$$\text{subject to} \quad g(x) \equiv \cos(\theta)\big(f_2(x) - e\big) - \sin(\theta) f_1 \quad \ge$$

$$a\left|\sin\left(b\pi\big(\sin(\theta)\big(f_2(x) - e\big) + \cos(\theta) f_1(x)\big)^c\right)\right|^d$$

where the variables θ, a, b, c, d, e are defined for CTP-6, 7, and 8 to give different shaped constraints.

Table 2. CTP Parameters

	θ	a	b	c	d	e
CTP-6	0.1π	40.0	0.5	1.0	2.0	-2.0
CTP-7	-0.05π	40.0	5.0	1.0	6.0	0.0
CTP-8 g_1	0.1π	40.0	0.5	1.0	2.0	-2.0
CTP-8 g_2	-0.05π	40.0	2.0	1.0	6.0	0.0

Both BSEA and NSGA-II perform excellently on CTP-6,7,8; both evenly cover the constrained Pareto front, and there is no appreciable difference between them. The results shown in Fig. 2 were visually indistinguishable for 10 out of 10 runs for each algorithm, and over many algorithm parameter values in addition to the stated parameter values. Due to space constraints, only results for CTP-8 are shown.

However, CTP-6,7,8 do not test the algorithms' ability to *cross* infeasible regions: the feasible regions are large enough that the random initial population finds them all. The algorithms merely need to abandon infeasible solutions in preference for the non-dominated feasible solutions.

To test the algorithms' ability to cross infeasible regions, we simply restrict the *random initial* value of x_2 to $6 < x_2 \le 10$. (As soon as evolution begins, the usual $0 < x_2 \le 10$ range applies.) This prevents any *initial* solutions from being generated in the lower feasible regions. To reach the true constrained Pareto

Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Confe
Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05)
0-7695-2504-0/05 $20.00 © 2005 IEEE

front, the population must migrate across the middle infeasible band.

Figures 3 and 4 show the behaviour of NSGA-II and BSEA in detail over the first 5 generations, to directly illustrate the difference in behaviour.

In both cases the random initial population is distributed over the same upper region of objective space, with most solutions in the infeasible region and some in the upper feasible regions. The entire NSGA-II population quickly pulls back to the few feasible solutions over the first three generations, abandoning some infeasible solutions that were close to the lower feasible regions. As soon as the entire population is feasible, it becomes impossible for any infeasible solution to survive in the population leaving NSGA-II stuck behind the infeasible region after 1000 generations.

In contrast, part of the BSEA population builds around the promising infeasible solutions while part withdraws to keep feasible solutions in the population. This balance is maintained by α dynamically adjusting R_B in favour of objective optimisation R_O or constraint satisfaction R_C. However as the run progresses r steadily increases, allowing the population to collapse into a single non-dominated rank by the end of the run. After 1000 generations, the population has correctly found the constrained Pareto front.

The described behaviours, with the same population distributions after 1000 generations (seen in Fig. 5), were observed in 10 out of 10 runs for each algorithm.

Further experiments confirmed that BSEA is still able to cross the infeasible region when the mutation standard deviation set to as little as 1/1000th of the decision variable range. In this case the algorithm was allowed to run for 10,000 generations to give it time to evolve gradually across the infeasible region.

6. Conclusion

The concept of blended space proposed in [8] is used to build a constrained multiobjective EA. BSEA is designed to operate robustly in highly constrained spaces. Its strength is its ability to cross infeasible regions by gradual evolution rather than by one chance mutation.

BSEA is very robust to parameter settings; it simply requires more generations to creep across the infeasible region when smaller mutation/crossover rates and magnitudes are used.

BSEA does not require any additional parameters; both α and r are self-adjusting. The computational overhead of blended ranking is $O(MP^2+NP^2)$ where M is the number of objectives, N is the number of constraints, and P is the population size. This is higher than an extended-dominance algorithm which is $O(MP^2)$ [5]. However, this linear increase in overhead is mitigated by blended ranking using a simpler dominance ranking than extended dominance ranking, and is expected to be masked by real evaluation functions.

The experiments in this paper have illustrated BSEA's behaviour on one small set of problems. Further experiments, including application to real problems rather than artificial benchmarks, need to be performed.

References

[1] C. Coello, "An Updated Survey of GA-Based Multiobjective Optimization Techniques," *ACM Computing Surveys*, vol. 32, pp. 109 - 143, 2000.

[2] C. Coello, "Theoretical and Numerical Constraint-Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 1245--1287, 2002.

[3] T. Ray, K. Tai, and K. C. Seow, "Multiobjective Design Optimization By An Evolutionary Algorithm," *Engineering Optimization*, vol. 33, pp. 399-424, 2001.

[4] S. R. Ranjithan, S. K. Chetan, and D. K. Harish, "Constraint Method-Based Evolutionary Algorithm (CMEA) for Multiobjective Optimization," in *Lecture Notes in Computer Science*, vol. 1993: Springer-Verlag, 2001.

[5] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGAII," Indian Institute of Technology, Kanpur KanGAL Report No. 200001, 2000.

[6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Sydney: Addison-Wesley, 1989.

[7] F. Jiménez and J. L. Verdegay, "Evolutionary techniques for constrained optimization problems," presented at 7th European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany, 1999.

[8] A. Angantyr, J. Andersson, and J.-O. Aidanpaa, "Constrained Optimization based on a Multiobjective Evolutionary Algorithm," presented at Congress on Evolutionary Computation, Canberra, Australia, 2003.

[9] K. Deb, A. Pratap, and T. Meyarivan, "Constrained Test Problems for Multi-objective Evolutionary Optimization," presented at First International Conference on Evolutionary Multi-Criterion Optimization, Zurich, Switzerland, 2001.

[10] J. D. Knowles, D. W. Corne, and M. Fleischer, "Bounded Archiving using the Lebesgue Measure," presented at Congress on Evolutionary Computation 2003, Canberra, Australia, 2003.

0-7695-2504-0/05 $20.00 © 2005 IEEE