

Design and Validation of a Hybrid Interactive Reference Point Method for Multi-Objective Optimization

Madan Sathe, Günter Rudolph, Kalyanmoy Deb

Abstract—This paper offers a classification of the main representatives of *interactive* classical and evolutionary methods. After a crossfertilization of these two fields a new hybrid interactive reference point method is designed and implemented with a graphical user interface. Finally, it is validated on two well-known real-world test problems.

I. INTRODUCTION

Many real world problems have conflicting multiple objectives which a decision maker would like to optimize simultaneously. In high-dimensional multi-objective optimization problems it is a time consuming task to calculate the whole Pareto front. Interactive methods - methods which include the decision maker in the algorithm - try to solve the problem by calculating only solutions which are interesting for the user. From the classical side interactive algorithms are mainly developed as a decision support since the 1960s. Mainly, since 2002 [26] there is a huge interest to combine the interactive classical methods with ideas of Computational Intelligence to solve problems in a computational faster manner. In section II some basic definitions are introduced, followed by a classification of interactive classical and evolutionary methods in section III. The new hybrid interactive reference point method and the corresponding graphical user interface are presented in section IV. We validate our approach by some test problems in section V. Finally, a summary and an outlook are provided in section VI.

II. PRELIMINARIES

A multi-objective optimization problem is given by

$$\begin{aligned} \min \quad & \vec{f}(\vec{x}), \\ \text{subject to} \quad & \vec{x} \in S \end{aligned} \quad (1)$$

where $S \subseteq \mathbb{R}^n$ is the feasible region and $\vec{f}: S \rightarrow \mathbb{R}^M$ the vector-valued objective function. Since there is no natural total order on vectors the concept of dominance is used for comparison.

Definition 1 (Pareto Domination): A vector \vec{x} is *Pareto dominating* a vector \vec{y} (denoted by $\vec{x} \preceq \vec{y}$) iff

- 1) $x_i \leq y_i \quad \forall i = 1, \dots, M$ and
- 2) $x_i < y_i \quad \exists i \in \{1, \dots, M\}$.

With the help of the domination operator it can be theoretically determined if a solution x^* is optimal, i.e. there exists no solution which dominates x^* .

Madan Sathe (madan.sathe@unibas.ch) is with the Department of Computer Science at the University of Basel (Switzerland). Günter Rudolph (guenter.rudolph@tu-dortmund.de) is with the Department of Computer Science at the Technical University of Dortmund (Germany) and Kalyanmoy Deb (deb@iitk.ac.in) is with the Indian Institute of Technology at Kanpur (India).

Definition 2 (Pareto Optimal): A point $x^* \in S$ is *Pareto optimal*, if for every $x \in S$ it holds:

$$\nexists x \in S : f(x) \preceq f(x^*). \quad (2)$$

Very useful are the ideal and nadir objective vectors. If each objective function is individually optimized the constitution of each solution builds the ideal vector. The nadir vector gives an upper bound of the objective functions according to the Pareto optimal set (see Figure 3). For all these definitions see e.g. [9].

III. INTERACTIVE CLASSICAL AND EVOLUTIONARY METHODS

The idea of including the decision maker (DM) in the decision process is the best method for satisfying the DM—the main goal of the analyst. For that purpose there are a lot of interactive classical methods which are traditionally classified into two classes: information required from the user and the inner resolution strategy [19]. Some possible criteria to compare classical methods are e.g. the ease of use of the method or the number of iterations to find the compromised solution [1]. In general an interactive approach can be characterized by following procedure:

- 1) Find a preferably feasible and efficient solution.
- 2) Interact with the DM to obtain his response to the solution.
- 3) Repeat steps 1 and 2 until satisfaction is reached or until some other termination criterion is met.

Classifying the classical methods according to the DM's preferences is one usual method. Hence, the DM is the person deciding about his most preferred solution and about his way to find one satisfactory solution. Subsequently, we present a user-based classification where the DM gets an overview about his involving in each method and what kind of information will be required by the chosen method. Figures 1 and 2 are providing fifteen methods [36] including the reference point method. There exist some more classical methods as listed above but the list is limited to the traditional ones. The former figure displays at which step each method reaches at least one feasible solution. The second figure gives an insight into the procedure handling of each method regarding to the feasible solution. Both figures are modelling only the main aspect of each method to give a quick overview about the interactive method landscape. The user is the central aspect. He (assuming the DM is a male individual) can give some new information or modification at a solution after each calculation step. The circled and rectangled boxes present the user-sided

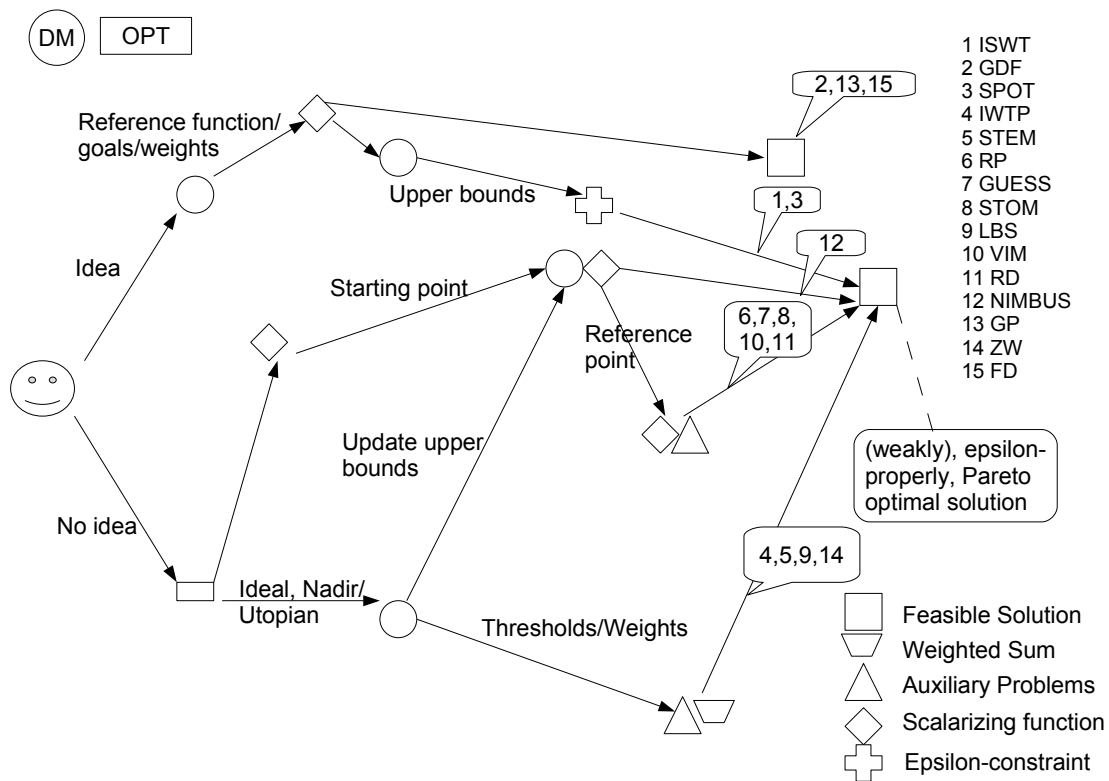


Fig. 1. Landscape of the classical methods until the user gets a feasible (often Pareto optimal) solution. For a detailed description see [36].

and analyst-sided interaction in each method path. The other boxes represent the next problem to solve to obtain the squared box, a feasible solution. The arrows indicate the output of a chosen method or the requirements for the next method/box.

Random search heuristics, especially evolutionary algorithms (EAs), are adapted to these methods to hopefully improve the computational time especially in high-dimensional problems. The existing interactive evolutionary multi-objective (I-EMO) methods can be classified based on:

- Reference Point [7], [24], [45], [40], [14], [36]
- Reference Direction [11]
- Light Beam Search [10]
- Tools [31], [6], [36]
- Modified EMO [4], [34], [30]
- Neuronal Networks [29], [42], [38]
- Simulated Annealing / Tabu Search [2], [20], [41]
- Utility Function
 - Fuzzy Logic [25], [17], [27], [35], [32], [8]
 - Comparison [23], [5], [33], [39], [3], [37]

Definitely, there is not a strong separation between each class but the presented classification is one possible way to categorize each method.

The new hybrid interactive reference point method is integrated in the class of reference point methods as well as in tools. Many I-EMO methods try to combine the idea of a classical method with an existing EMO. Some of the approaches try to determine the utility function of the user in

an explicit form or using fuzzy logic to transform the user's thoughts and assessments to numbers.

Most of the implemented classical algorithms are based on the reference point idea [43]. This observation is most likely caused by the fact that the idea is very simple and the used concept of scalarizing functions controls the success of the algorithm. The problem on the classical side with the reference point method is that it sometimes takes a lot of computational time for solving non-linear problems, whereas this idea combined with evolutionary algorithms results in a more satisfying computational time.

Furthermore, most of the implemented interactive EAs only have an alphanumeric interface which might not be as user-friendly as a graphical user interface.

The developed algorithm combines the classical idea of the reference point with the $(1 + 1)$ - EA. Hence, the idea of using a scalarizing function as a distance measure is a good selection operator and fitness function in the EA. The user decides about some reference points and the hybrid algorithm optimizes the desirable goals due to the direction to the Pareto front. The DM can interact through the GUI, for instance by changing weights or altering reference points before the hybrid algorithm optimizes taking into account these new adjustments. The implemented algorithm is restricted to two- and three-objective problems and is at the moment restricted to nonlinear optimization problems.

Next, the hybrid interactive reference point method is explained, followed by an application of the implemented

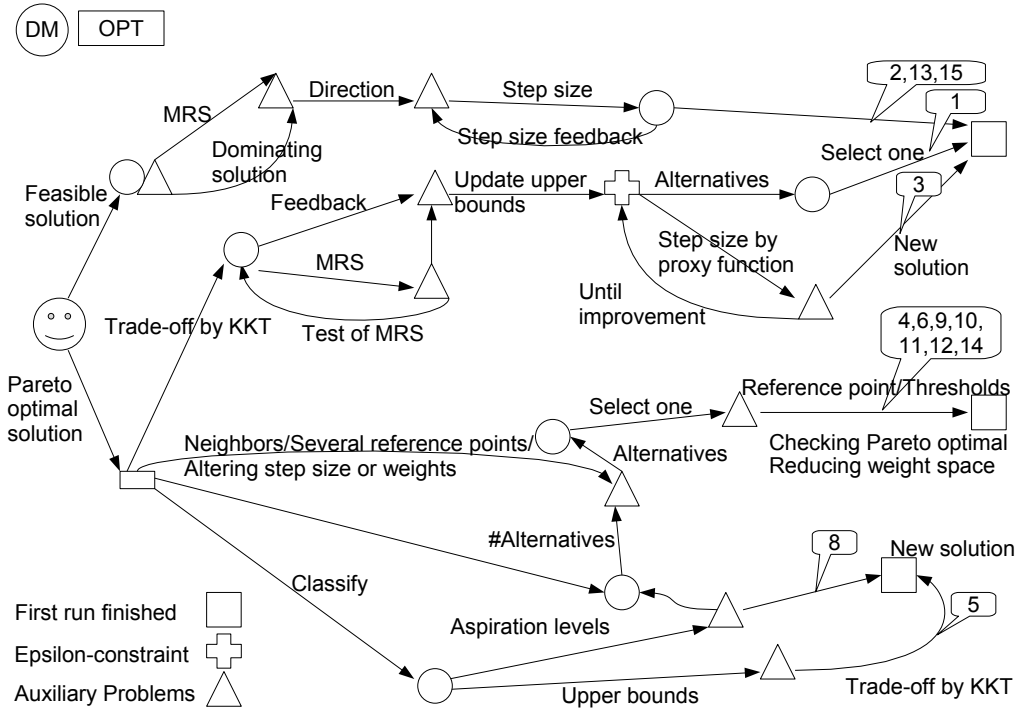


Fig. 2. Landscape of the classical methods until the user gets a new solution by interaction whereas KKT is the short form for the Karush-Kuhn-Tucker conditions and MRS for marginal rates of substitution. For a detailed description see [36].

method on two engineering design problems.

IV. HYBRID INTERACTIVE REFERENCE POINT METHOD

A. Algorithm

The algorithm is a combination of a $(1+1)$ - EA and an achievement scalarizing function. For obtaining a (weakly, ϵ -properly [9] or) Pareto optimal solution a special distance measure, called scalarizing function, is often used (Figure 3). As the proposed achievement scalarizing function the distance between the achievement solution (e.g. the reference point) and the current solution are weighted and maximized over each component. Additionally, a small term is added to the maximum to guarantee the finding of Pareto optimal solutions instead of weakly Pareto optimal solutions (see equation 3). In each step the best parent solution is mutated with the polynomial mutation operator [9] to create an offspring solution and the best of them, the minimum of both solutions, is the new parent solution in the next step. If the DM is satisfied with the assessed final solution, the neighborhood of this solution can be also displayed. For this purpose the DM determines the search distance to the final solutions and a $(1+1)$ - EA with a scalarizing function is started with slightly altered originally weights adapting the preferred distance from the user. Finally, the user should choose the most compromised solution.

- 1) Ask the DM to select n of reference points z_i^h with $i \in \{1, \dots, n\}$ for the optimization problem under consideration. Set $h = 1$.
- 2) Present n feasible starting points z_i^h to the DM.

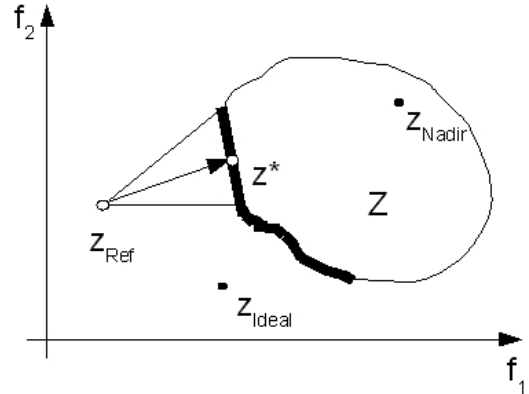


Fig. 3. In this figure the concept of the achievement scalarizing function is given at a two-dimensional problem whereas Z is the objective space, z_{nadir} is the nadir vector, z_{ideal} is the ideal vector, z^* is the obtained Pareto optimal solution and z_{Ref} is the chosen reference point. The bold line is the Pareto optimal front. The arrow symbolizes the direction to the Pareto optimal solution determined by the weighting vector and the lines of z_{Ref} to the Pareto front marked the bounds for the weighting vector.

- 3) Ask the DM to specify an aspiration level for each objective at each starting point z_i^h , resulting in a reference point z_i^h .
- 4) Calculate the next solution with a $(1+1)$ - EA where
 - the mutation operator is the polynomial mutation with $p = \frac{1}{\sqrt{n}}$,
 - the fitness function is the achievement scalarizing

function $s(z_i^h, (w, \hat{z}_i^h))$:

$$\text{maximize}_{j=1, \dots, m} [w_j \cdot (z_{ij}^h - \hat{z}_{ij}^h)] + \rho \cdot \sum_{j=1}^m w_j \cdot (z_{ij}^h - \hat{z}_{ij}^h) \quad (3)$$

where $\rho > 0$ is a small augmentation scalar (here, $\rho = 0.000001$).

- 5) The selection operator prefers the minimum of the fitness function.
- 6) Present each assessed solution and the relating reference point to the DM.
- 7) If the DM is satisfied with this direction but not with the solution, go to step 4 with $z_i^{h+1} = z_i^h$. Set $h = h + 1$. If the DM is also satisfied with the calculated solution, the DM can start a local multi-gradient search algorithm, a modified Pareto descent method [22], [21], to improve the solution values if there exist the partial derivatives. The direction calculation from the Pareto descent method is combined with the step size control presented in [18]. Present the final solution to the DM and determine some solutions which have a user-defined distance to the final solution by altering slightly the weights of the scalarizing function. Calculate some neighborhood solutions with the $(1 + 1)$ - EA and the scalarizing function, again.
Otherwise go to step 3.

B. Graphical User Interface

First, the user has to choose his problem in the configuration window and determine the number of mutation generations, the number of gradient calculations and the number of starting solutions which fixed the number of reference points (see Figure 4). If he presses the button “OK” the main program will start. Within the GUI (see Figure 5) the user is working

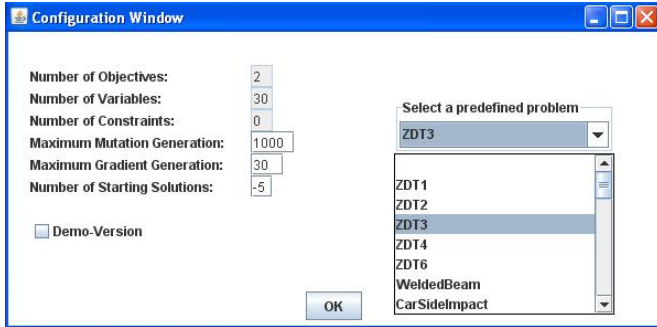


Fig. 4. The starting point in the program will be a configuration window. Therein the user can choose his optimization problem and some maximum boundaries as the number of starting solutions.

in the objective space. He can choose the number of solutions to work with and can give several aspiration levels for each objective - summarized to a reference point with the support of sliders and adjust different weights for each objective according to his preferences. In the textboxes at the bottom of the GUI the DM can see the current assessed values at each step as the detailed variable vector values. The buttons on the

top right-hand side are giving to the user some possible events such as the button “Show all” to display all calculated solutions (see Figure 5). Underneath the button area there are some user installed parameter and the above mentioned slider. As in this figure it is possible to create an approximated Pareto front by the calculated solutions and the neighborhood solutions as seen in the graphical view. The second implemented GUI displays a demonstrator where the DM can see the assessed Pareto front of the NSGA-II [13] at the top right-hand side and can interact inside the window. The third dimension is represented by colors as in [28].

V. APPLICATIONS

Our pragmatic approach to describe “How does the application work?” applies our method to two real-world problems such as the biobjective Welded Beam Problem [15] and the three-objective Car Side Impact Problem [12].

A. Real-World Problem Welded Beam

The welded beam design problem has to minimize the two objectives cost of fabrication (f_1) and end deflection of the welded beam (f_2) and as variables the variable vector $z = (b, h, l, t)$. Figure 7 gives an insight into the calculated approximated Pareto Front of the problem. Formally, the

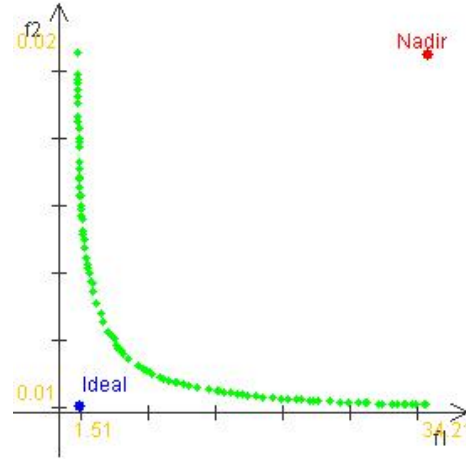


Fig. 7. The approximated Pareto front of the welded beam design problem with the NSGA-II with the support of the Demonstrator version.

problem looks like

$$\begin{aligned} &\text{minimize} && f_1(z) = 1.10471h^2l + 0.04811tb(14.0 + l), \\ &\text{minimize} && f_2(z) = \frac{2.1952}{t^3b}, \\ &\text{subject to} && g_1(z) = 13,600 - \tau(z) \geq 0, \\ &&& g_2(z) = 30,000 - \sigma(z) \geq 0, \\ &&& g_3(z) = b - h \geq 0, \\ &&& g_4(z) = P_c(z) - 6,000 \geq 0, \\ &&& 0.125 \leq b \leq 5.0, 0.125 \leq h \leq 5.0, \\ &&& 0.1 \leq l \leq 10.0, 0.1 \leq t \leq 10.0. \end{aligned}$$

The first constraint g_1 ensures that the shear stress developed at the support location of the beam is smaller than the allowable shear strength of the material (13,600 psi). The second

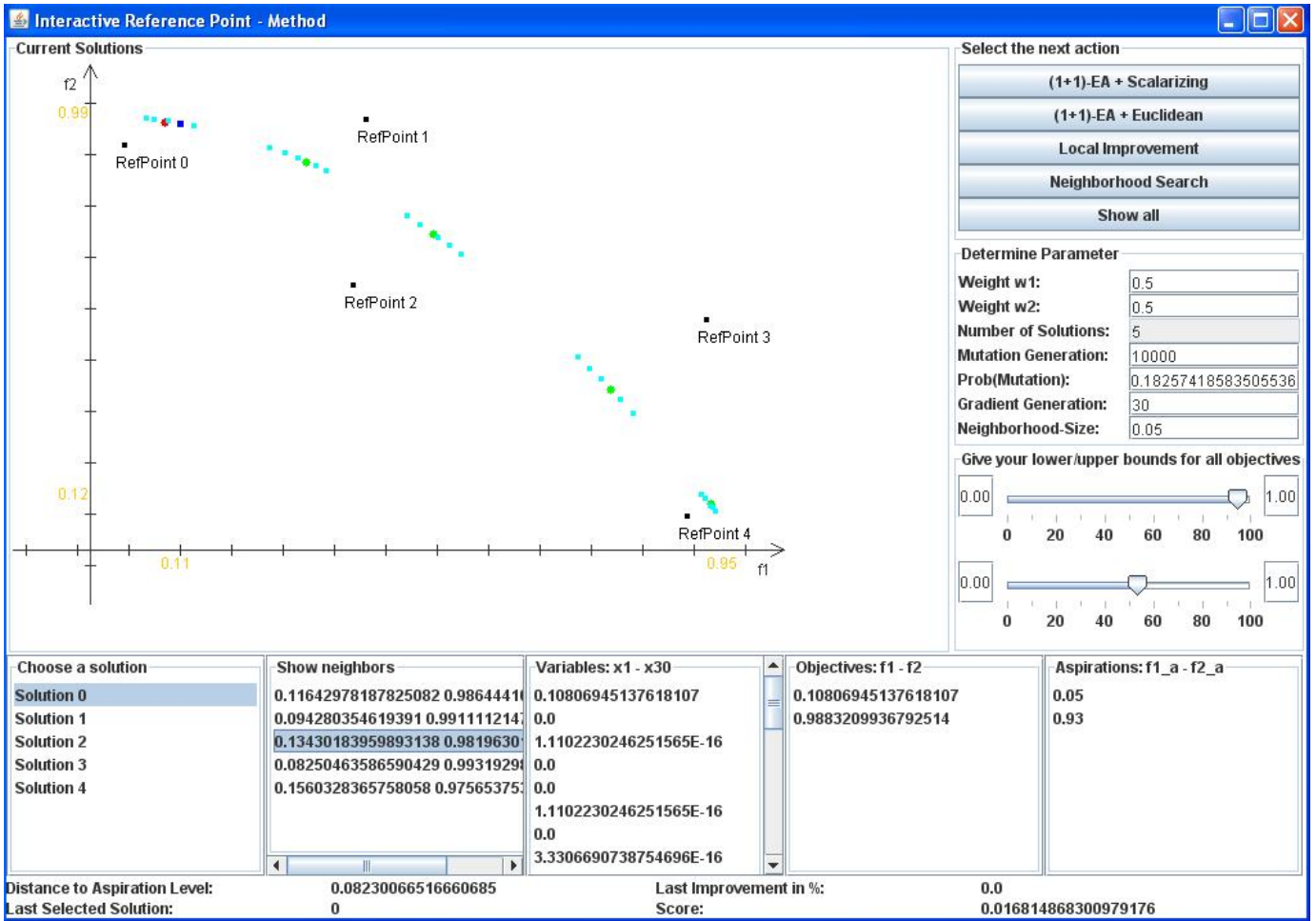


Fig. 5. The test problem ZDT2 [44] has been solved by five reference points. The resulted solutions are creating an approximated Pareto front.

constraint g_2 makes sure that normal stress developed at the support location of the beam is smaller than the allowable yield strength of the material (30,000 psi). The third constraint g_3 makes sure that thickness of the beam is not smaller than the weld thickness from a practical view. The fourth constraint g_4 makes sure that the allowable buckling load (along t direction) of the beam is more than the applied load $F = 6,000$ lbs. A violation of any of the above four constraints will make the design unacceptable. The stress and buckling terms are non-linear to design variables and are given as follows

$$\tau(z) = \frac{\sqrt{(\tau')^2 + (\tau'')^2 + (l\tau'\tau'')/\sqrt{0.25(l^2 + (h+t)^2)}}}{\sqrt{0.25(l^2 + (h+t)^2)}},$$

$$\tau' = \frac{6,000}{\sqrt{2hl}},$$

$$\tau'' = \frac{6,000(14+0.5l)\sqrt{0.25(l^2 + (h+t)^2)}}{2\{0.707hl(l^2/12+0.25(h+t)^2)\}},$$

and

$$\sigma(z) = \frac{504,000}{l^2b},$$

$$P_c(z) = 64,746.022(1 - 0.0282346t)tb^3.$$

The starting solutions have the values as shown in Table I. In the Interactive Reference Point - Display the DM chooses every solution seeing the minimum objective values of each

Objective Values (f_1, f_2)	Variable Vector z	Aspiration Levels
(46.623, $4.7232 \cdot 10^{-4}$)	(4.787, 0.8419, 4.7972, 9.902)	(14.94, $4.2 \cdot 10^{-4}$)
(18.0252, 16.87997)	(3.988, 1.3922, 7.7935, 0.3195)	(8.28, 0.0015)
(231.935, 0.00183)	(4.5843, 4.6061, 8.54027, 6.39103)	(11.52, $8.4 \cdot 10^{-4}$)
(82.4304, 3.6095)	(0.15529, 4.3055, 4.0149, 1.57626)	(0, 0)
(189.8575, 0.00335)	(2.3469, 4.9748, 6.3939, 6.5339)	(18, 0.002)

TABLE I

THE TABLE SHOWS THE STARTING SOLUTIONS OF THE DESIGN PROBLEM WELDED BEAM. EACH LINE REPRESENT A SOLUTION FROM 0 TO 4.

starting solution. Consequently, he adjusts the upper bounds of the slider. The lower bounds of zero is an ideal point for each objective. The new upper bound vector is $B'' = (f_{1Bu}, f_{1Bu}) = (18, 0.002)$. For the DM the cost factor is more important than the end deflection of the welded beam, so that he changes the weights to $w_{new} = (0.7, 0.3)$. The aspiration levels are chosen in respect to the new bounds. For solution three he chooses the ideal point and for solution four the current worst objective value. The resulted solutions are given in Table II. The DM sees the minimum bound 0.26 units for fabrication cost at solution three. So he decides to get only solutions which have maximum costs of five units and alters the relating bound of objective one of the slider. According to this change, the DM

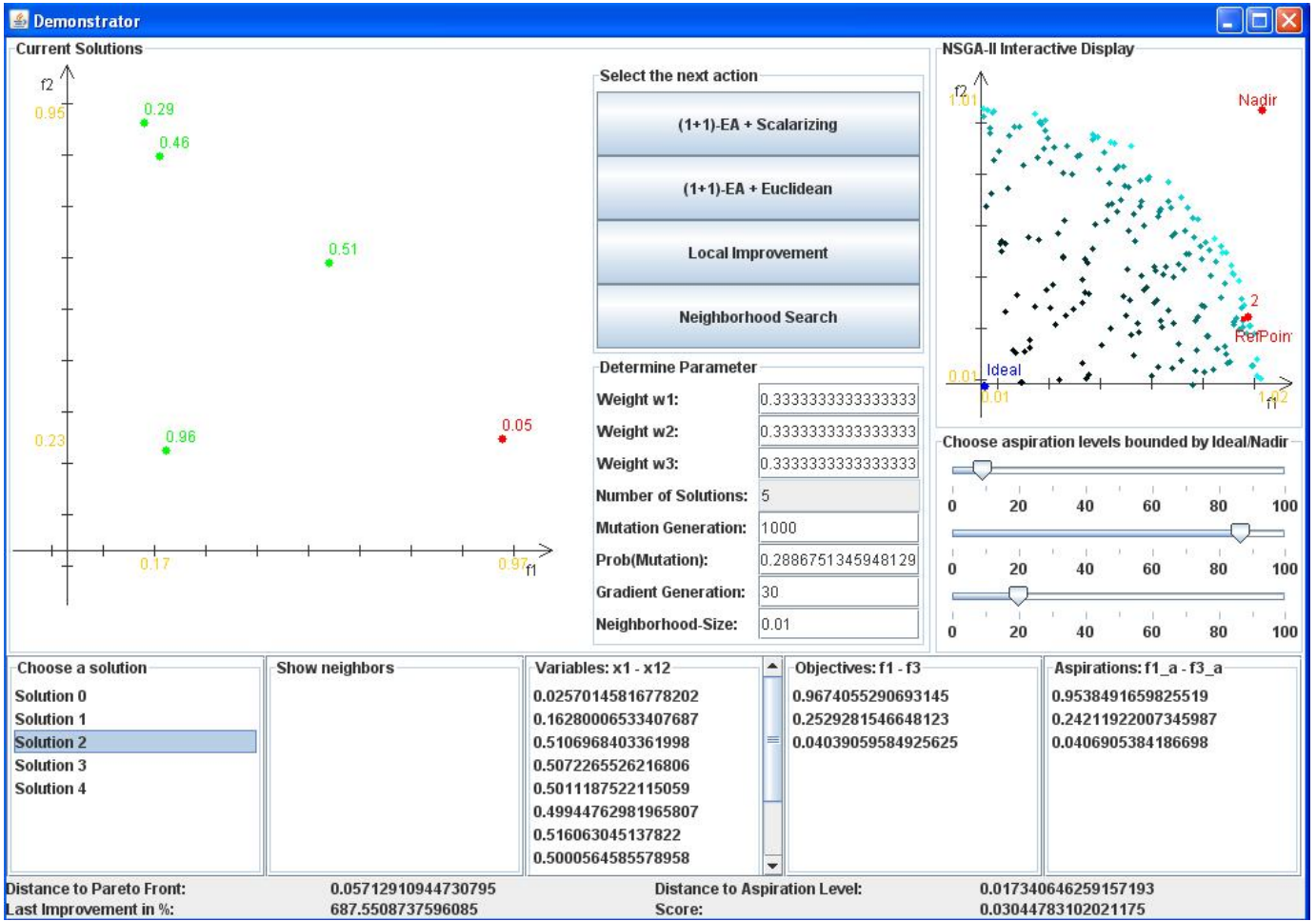


Fig. 6. The well-known three dimensional testproblem DTLZ2 [16] at the demonstrator version. The third color represents the third dimension in the small window on the right-hand side.

Solution	Objective Values (f_1, f_2)	Variable Vector z
0	(14.939, $9.969 \cdot 10^{-4}$)	(2.202, 0.125, 0.1, 10.0)
1	(8.226, 0.0018)	(1.212, 0.126, 0.10, 9.99)
2	(11.5202, 0.0013)	(1.7014, 0.125, 0.1, 9.98)
3	(0.2615, 0.6105)	(0.125, 0.125, 0.1, 3.0641)
4	(17.993, $8.277 \cdot 10^{-4}$)	(2.6523, 0.125, 0.1, 10.0)

TABLE II

THE TABLE SHOWS THE CURRENT SOLUTIONS OF THE DESIGN PROBLEM WELDED BEAM AFTER A CALCULATION STEP.

Objective Values (f_1, f_2)	Variable Vector z	Aspiration Level
(4.3506, 0.0034)	(0.6411, 0.125, 0.10, 10.0)	(4.35, $4.2 \cdot 10^{-4}$)
(2.701, 0.0055)	(0.3979, 0.125, 0.1, 10.0)	(2.7, 0.0015)
(1.105, 0.00135)	(0.1625, 0.125, 0.1, 10.0)	(1.1, $8.4 \cdot 10^{-4}$)
(1.503, 0.0099)	(0.2214, 0.125, 0.1, 10.0)	(1.5, 0.0017)
(4.997, 0.00298)	(0.7367, 0.125, 0.10, 10.0)	(5.0, 0.002)

TABLE III

THE TABLE SHOWS THE FINAL SOLUTIONS OF THE DESIGN PROBLEM WELDED BEAM. EACH LINE REPRESENT A SOLUTION FROM 0 TO 4.

gives five new reference points by choosing new aspiration levels for objective one and increases the number of mutation generation to 10000 to find final solutions (see Table III). The user prefers solution one with $s_1 = (f_1, f_2) = (2.701, 0.0055)$ and $z = (b, h, l, t) = (0.397, 0.125, 0.1, 10.0)$. He attempts to optimize the solution with the local improvement approach and generates some solutions close to the chosen one. As final action the DM starts the Demonstrator with the same adjustments to obtain the position of the final solution on the approximated Pareto front.

B. Real-World Problem Car-Side Impact

The car-side impact problem is a real world problem whereas a car is subjected to a side-impact based on European Enhanced Vehicle-Safety Committee (EEVC) procedures. The effect of the side-impact on a dummy in terms of head injury (HIC), load in abdomen, pubic symphysis force, viscous criterion ($V * C$), and rib deflections at the upper, middle and lower rib locations are considered. The effect on the car is considered in terms of the velocity of the B-Pillar at the middle point and the velocity of the front door at the B-Pillar. An increase in dimension of the car parameters may improve the performance on the dummy but the increased weight of the car

may have an adverse effect on the fuel economy. Thus, there is a need to come up with a design having a balance between the weight and the safety performance. The three-dimensional car-side impact problem has seven design variables, ten constraints and is dealing with the best trade-off solution between the car parameter (f_1 and f_3) and the personal safety (f_2) (Figure 8).

First, the DM determined his reference function to adjust

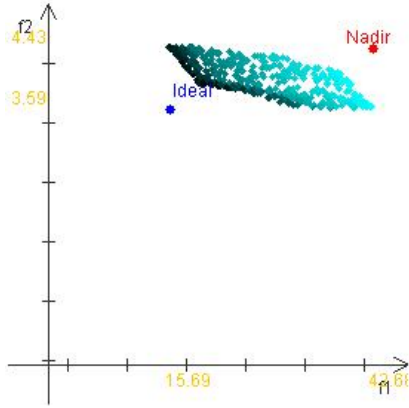


Fig. 8. The approximated Pareto front of the car-side impact design problem with the NSGA-II with the support of the Demonstrator version. The third dimension represents the color gradient from black to white (colored as turquoise).

some weights. His reference function is objective two, implying personal safety is more important than the weight of the car and the velocity of the V-Pillar, so that the weights are chosen as $w = (w_1, w_2, w_3) = (0.25, 0.5, 0.25)$. The range for each objective can be seen from the current solutions and the DM provides approximate bounds for each objective. For objective one the values are between 20 and 38 units. For the pubic symphysis force the values are between 3.7 and 4.5 kN, and the average of the velocities lies in $[11.5, 13]$ per mm/ms. Accordingly to the bounds he sets the reference points (see Table IV) and starts the optimization process. The

Solution	Objective Values (f_1, f_2)	Aspiration Level
0	(37.40, 3.7538, 11.7182)	(23.6, 4.396, 12.475)
1	(28.714, 4.3268, 11.782)	(30.98, 3.952, 11.964)
2	(21.519, 4.212, 12.769)	(38.0, 4.388, 12.115)

TABLE IV
STARTING SOLUTIONS OF THE DESIGN PROBLEM CAR-SIDE IMPACT.

resulting solutions are checked by the user in relation to his value function. Now, he starts the local improvement to obtain better results. Afterwards, he increases the mutation generation to 10000. At the end, he generates the neighborhood solution for the most compromising solution one, because the value has the smallest value for all solutions in the second objective (see Table V).

VI. SUMMARY

In the first part of the paper there is a classification approach of the existing classical and evolutionary interactive

Objective Values (f_1, f_2)	Aspiration Level	Neighborhood One
(22.926, 4.059, 11.802)	(23.6, 4.396, 12.475)	(30.583, 3.754, 11.5678)
(30.580, 3.75316, 11.5662)	(30.98, 3.952, 11.964)	(30.58168, 3.7528, 11.5657)
(36.7748, 3.7759, 10.8909)	(38.0, 4.388, 12.115)	(30.5847, 3.75437, 11.5687)

TABLE V
FINAL SOLUTIONS OF THE DESIGN PROBLEM CAR-SIDE IMPACT. EACH LINE REPRESENT A SOLUTION FROM 0 TO 2.

methods. Presently, there are only few GUI-based interactive implemented algorithms available. This study applies the reference point idea to a GUI-based application as a new approach. It is validated on several well-known test problems as well as real world problems. The implemented hybrid interactive reference point method presents the current solutions to the DM in each step. He determines the direction by aspiration levels for each objective and an approximate step size by the number of mutation generation. The direction can be influenced by a weighting vector which alters the found solutions by the used achievement scalarizing function. In the applications shown, the computation time of this algorithm is very small even though we have used the default values. As a consequence, the DM obtains results very quickly. This is important, since the response times of interactive methods must be as short as possible.

In the future it should be possible to check the final solution for Pareto optimality and to integrate more than one decision maker in the algorithm and in the GUI, respectively. This will target the group decision methods. At the moment the method is restricted to the objective space. Maybe it could be possible to extend the idea to the decision space, too. Also, more than three-dimensional problems could be displayed by the idea of using different colors in the GUI as in the three-dimensional case shown. Finally, adding several hybrid methods as a combination of the GDF method with a $(1 + 1)$ - EA or a neuronal network to determine the approximated utility function could be a more efficient support for the DM. Hence, he is able to compare the solutions after each calculation step. For all these extension ideas some more detailed investigations are required.

REFERENCES

- [1] Y. Aksoy, T. W. Butler, and E. D. Minor. Comparative studies in interactive multiple objective mathematical programming. *European Journal of Operational Research*, 89:408–422, 1996.
- [2] M. João Alves and João C. N. Clímaco. An interactive method for 0-1 multiobjective problems using simulated annealing and tabu search. *J. Heuristics*, 6(3):385–403, 2000.
- [3] Meghna Babbar, Barbara Minsker, and Hideyuki Takagi. Interactive genetic algorithm framework for long term groundwater monitoring design, 2004.
- [4] Jürgen Branke and Kalyanmoy Deb. Integrating User Preferences into Evolutionary Multi-Objective Optimization. In Yaochu Jin, editor, *Knowledge Incorporation in Evolutionary Computation*, pages 461–477. Springer, Berlin Heidelberg, 2005.
- [5] Jürgen Branke, Thomas Kaufler, and Harmut Schneck. Guidance in Evolutionary Multi-Objective Optimization. *Advances in Engineering Software*, 32:499–507, 2001.
- [6] Shamik Chaudhuri. *An Interactive Multi-Objective Optimization and Decision-making Procedure Using Evolutionary Methods (I-MODE)*. PhD thesis, Indian Institute of Technology, Kanpur, India, 2006.

- [7] Carlos Coello, Luis V. Santana, Alfredo Hernandez-Dyaz, Rafael Caballero, and Julian Molina. g-dominance: Reference point based dominance. In Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen, and Rafael Slowinski, editors, *Practical Approaches to Multi-Objective Optimization*. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, see <http://kathrin.dagstuhl.de/files/Materials/06/06501/>, checked at 04-06-2007, 2006.
- [8] D. Cvetkovic and I. C. Parmee. Preferences and their application in evolutionary multiobjective optimization. *IEEE-TEC*, 6:42–57, 2002.
- [9] Kalyanmoy Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, Chichester, UK, 2001.
- [10] Kalyanmoy Deb and Abhay Kumar. Light beam search based multi-objective optimization using evolutionary algorithms. KanGAL report 2007005, Indian Institute of Technology, Kanpur, India, 2007.
- [11] Kalyanmoy Deb and Abishek Kumar. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. KanGAL report 2007001, Indian Institute of Technology, Kanpur, India, 2007.
- [12] Kalyanmoy Deb, Dhanesh Padmanabhan, Sulabh Gupta, and Abhishek Kumar Mall. Reliability-based multi-objective optimization using evolutionary algorithms. In Shigeru Obayashi, Kalyanmoy Deb, Carlo Poloni, Tomoyuki Hiroyasu, and Tadahiko Murata, editors, *EMO*, volume 4403 of *Lecture Notes in Computer Science*, pages 66–80. Springer, 2006.
- [13] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [14] Kalyanmoy Deb and J. Sundar. Reference point based multi-objective optimization using evolutionary algorithms. In Mike Cattolico, editor, *Genetic and Evolutionary Computation Conference, GECCO 2006, Proceedings, Seattle, Washington, USA, July 8-12, 2006*, pages 635–642. ACM, 2006.
- [15] Kalyanmoy Deb and J. Sundar. Reference point based multi-objective optimization using evolutionary algorithms. In Mike Cattolico, editor, *Genetic and Evolutionary Computation Conference, GECCO 2006, Proceedings, Seattle, Washington, USA, July 8-12, 2006*, pages 635–642. ACM, 2006.
- [16] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable Multi-Objective Optimization Test Problems. In *Congress on Evolutionary Computation (CEC'2002)*, volume 1, pages 825–830. Piscataway, New Jersey, 2002b. IEEE Service Center.
- [17] Eduardo Fernández and Juan Carlos Leyva. A method based on multiobjective optimization for deriving a ranking from a fuzzy preference relation. *European Journal of Operational Research*, 154(1):110–124, 2004.
- [18] J. Fliege and B. F. Svaiter. Steepest descent methods for multicriteria optimization. *Math. Methods Oper. Res.*, 51(3):479–494, 2000.
- [19] L. R. Gardiner and R. E. Steuer. Unified interactive multiple objective programming. *European Journal of Operational Research*, 74(3):391–406, 1994.
- [20] Maciej Hapke, Andrzej Jaskiewicz, and Roman Slowinski. Pareto Simulated Annealing for Fuzzy Multi-Objective Combinatorial Optimization. *Journal of Heuristics*, 6(3):329–345, 2000.
- [21] Ken Harada, Kokoro Ikeda, and Shigenobu Kobayashi. Hybridization of genetic algorithm and local search in multiobjective function optimization: Recommendation of ga then ls. In *GECCO*, pages 667–674, 2006b.
- [22] Ken Harada, Jun Sakuma, and Shigenobu Kobayashi. Local search for multiobjective function optimization: Pareto descent method. In *GECCO 2006: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, volume 1, pages 659–666, Seattle, Washington, USA, 2006a. ACM Press.
- [23] F. C. Hsu and J. S. Chen. A study on multi criteria decision-making model: Interactive genetic algorithms approach. *Proceedings of the IEEE Int. Conf. on System, Man, and Cybernetics (SMC'99)*, 3:634–639, 1999.
- [24] Hisao Ishibuchi and Kaname Narukawa. Comparison of evolutionary multiobjective optimization with reference solution-based single-objective approach. In Hans-Georg Beyer and Una-May O'Reilly, editors, *Genetic and Evolutionary Computation Conference, GECCO 2005, Proceedings, Washington DC, USA, June 25-29, 2005*, pages 787–794. ACM, 2005.
- [25] Yaochu Jin and Bernhard Sendhoff. Incorporation of Fuzzy Preferences into Evolutionary Multiobjective Optimization. In W. B. Langdon and E. Cantú-Paz et al., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*, page 683, San Francisco, California, 2002. Morgan Kaufmann Publishers.
- [26] D. F. Jones, S. K. Mirrazavi, and M. Tamiz. Multi-objective metaheuristics: an overview of the current state-of-the-art. *European Journal of Operational Research*, 137:1–9, 2002.
- [27] T. Kiyota, Y. Tsuji, and E. Kondo. Unsatisfying functions and multiobjective fuzzy satisficing design using genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics Part B-Cybernetics*, 33(6):889–897, 2003.
- [28] Alexander V. Lotov, Vladimir A. Bushenkov, and Georgy K Kamenev. *Interactive Decision Maps. Approximation and Visualization of Pareto Frontier*. Kluwer Academic Publishers, Boston, Massachusetts, 2004.
- [29] Behnam Malakooti and Vishnu Raman. Clustering and group selection of multiple criteria alternatives with application to space-based networks. *Journal of Intelligent Manufacturing*, 11:435–451, 2000.
- [30] Rafael Caballero Mercedes, Gonzalez Flor Guerrero, Alfredo G. Hernandez-Diaz, Julian Molina, and Concepcion Paralara. i-mom: An interactive multi-objective metaheuristic method. application to a real problem. *7th Int. Conf. on Multi-Objective Programming and Goal Programming (MOPGP06)*, xx:xx–xx, 2006.
- [31] K. Miettinen and M. Mäkelä. Interactive multiobjective optimization system WWW-NIMBUS on the internet. *Computers and Operations Research*, 27(7-8):709–723, 2000.
- [32] S. Phelps and M. Köksalan. An interactive evolutionary metaheuristic for multiobjective combinatorial optimization. *Management Science*, 49(12):1726–1738, 2003.
- [33] Gang Quan, Garrison W. Greenwood, Donglin Liu, and Sharon Hu. Searching for multiobjective preventive maintenance schedules: Combining preferences with evolutionary algorithms. *European Journal of Operational Research*, 177(3):1969–1984, 2007.
- [34] Atul Rangarajan, A. Ravi Ravindran, and Patrick Reed. An interactive multi-objective evolutionary optimization algorithm. In *In Proceedings of the 34th International Conference on Computers & Industrial Engineering*, pages 277–282, 2004.
- [35] Masatoshi Sakawa, Kosuke Kato, and Toshihiro Shibano. An interactive fuzzy satisficing method for multiobjective multidimensional 0-1 knapsack problems through genetic algorithms. In *Proceedings of the 1996 International Conference on Evolutionary Computation (ICEC'96)*, pages 243–246, 1996.
- [36] Madan Sathe. Interactive evolutionary algorithms for multi-objective optimization. Master's thesis, Technical University of Dortmund, 2007.
- [37] Abhishek Singh, Barbara Minsker, and Hideyuki Takagi. Interactive genetic algorithms for inverse groundwater modeling: Issues with human fatigue and prediction models, 2005.
- [38] Minghe Sun, Antonie Stam, and Ralph E. Steuer. Interactive multiple objective programming using tchebycheff programs and artificial neural networks. *Computers & OR*, 27(7-8):601–620, 2000.
- [39] Ye Tao, Hong-Zhong Huang, and Bo Yang. An interactive preference-weight genetic algorithm for multi-criterion satisficing optimization. In *ICNC (1)*, pages 643–652, 2006.
- [40] Lothar Thiele, Kaisa Miettinen, Pekka J Korhonen, and Julian Molina. A preference-based interactive evolutionary algorithm for multiobjective optimization. Working Papers W-412, Helsinki School of Economics, 2007.
- [41] E. L. Ulungu, J. Teghem, and Ch. Ost. Efficiency of interactive multi-objective simulated annealing through a case study. *Journal of the Operational Research Society*, 49:1044–1050, 1998.
- [42] Ping Wang, Hong-Hong Wuang, and Xu Hang. Multi-objective satisfactory optimization method. *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems*, 2:695–699, 2004.
- [43] A. P. Wierzbicki. The use of reference objectives in multiobjective optimization. In G Fandel and T Gal, editors, *Multiple Criteria Decision Making Theory and Applications*, LNEMS 177, pages 468–486. Springer, 1980.
- [44] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2):173–195, 2000.
- [45] Eckart Zitzler and Simon Künzli. Indicator-based Selection in Multiobjective Search. In Xin Yao et al., editor, *Parallel Problem Solving from Nature - PPSN VIII*, pages 832–842, Birmingham, UK, 2004. Springer-Verlag. Lecture Notes in Computer Science Vol. 3242.