

I-EMO: An Interactive Evolutionary Multi-Objective Optimization Tool

Kalyanmoy Deb and Shamik Chaudhuri

Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology, Kanpur
Kanpur, PIN 208016, India
{deb,shamik}@iitk.ac.in,
WWW home page: <http://www.iitk.ac.in/kangal>

KanGAL Report Number 2005003

Abstract. With the advent of efficient techniques for multi-objective evolutionary optimization (EMO), real-world search and optimization problems are being increasingly solved for multiple conflicting objectives. During the past decade of research and application, most emphasis has been spent on finding the complete Pareto-optimal set, although EMO researchers were always aware of the importance of procedures which would help choose one particular solution from the Pareto-optimal set for implementation. This is also one of the main issues on which the classical and EMO philosophies are divided on. In this paper, we address this long-standing issue and suggest an interactive EMO procedure which, for the first time, will involve a decision-maker in the evolutionary optimization process and help choose a single solution at the end. This study is the culmination of many years of research on EMO and would hopefully encourage both practitioners and researchers to pay more attention in viewing the multi-objective optimization as an aggregate task of optimization and decision-making.

1 Introduction

In the modern-day design and decision-making process, optimization plays a major role. Till now, most studies concentrated in finding the global optimum in a problem. To do so, the concerned algorithm searches the whole search space and at the end produces the best solution or a set of best solutions. However, with the advent of proper statistical tools designers and decision-makers got interested in introducing uncertainties and reliability-based concepts in the optimization process. These concepts are introduced simply because the real-world problem is full of such uncertainties and often the global optimum of an idealized deterministic optimization problem is not quite reliable.

In the context of multi-objective optimization and decision-making, there are additional problems of finding a set of optimal solutions and then making a scientific decision in choosing a particular solution. Often, such decision making

needs are subjective and only when adequate information about the nature of the optimal solutions is provided to the DM, a proper and reliable decision-making can be achieved. In this direction, the classical multi-objective literature provides a number of studies in which interactive methods were suggested. The sheer number of different methods also tells that none of these methods is widely accepted as a generic procedure. One common matter with all these methods is that they demand a lot of problem information from the DM without providing the DM any idea of the nature of the resulting Pareto-optimal front. Thus, in most cases, the DM, while using these methods, rely on guess-work.

On the other hand, the research focus in evolutionary multi-objective optimization (EMO), was to find the nature and shape of the Pareto-optimal front by using a number of well-distributed solutions. In their history of about 10 years, the researchers did not seem to address the issue of selecting one solution from the front. In this paper, for the first time, we suggest an interactive procedure (I-EMO), which takes the positive aspects of each field and attempts to suggest a procedure which can iteratively and interactively with a DM find the complete front and help focus to a single preferred solution. Based on the idea, we are currently developing a software. In this paper, we also provide some snapshots from this software to illustrate the usefulness and importance of the interactive procedure.

2 Related Works

There exists different interactive multi-objective optimization methods in the literature. But most of them are developed based on the classical optimization methods. Some of the most popular ones are Interactive Surrogate Worth Trade-off (ISWT) method [3], Reference Point method [12], the NIMBUS approach [10] etc. Each method is different from each other, but uses a single solution in each iteration. Based on particular multi-objective optimization procedure, a guess solution is modified to another solution iteratively and by gathering some information from a DM. Since a single solution is used in an iteration, the DM often has a local information (such as a local trade-off or search direction) and cannot make a decision using a more global picture of the true Pareto-optimal front. However, in the context of EMO, there does not exist many interactive studies. Tan et al. developed a GUI-based MOEA toolbox for multi-objective optimization [11]. The toolbox is designed with some classical decision-making aides, such as goal and priority settings. But a clear procedure of arriving at a single preferred solution is not present in the toolbox. Fonseca and Fleming [8] devised a GUI-based procedure which allowed some target values to be set for each objective and the effect on the trade-off of different objectives found using an EMO procedure was demonstrated. The procedure can be used to choose a preferred solution by visually comparing the solutions with the target objective values, the procedure clearly does not provide any indication of the location of chosen solutions on the Pareto-optimal front and also lacks quantitative statistical analysis of the solutions.

3 Interactive Evolutionary Multi-objective Optimization (I-EMO)

In the proposed interactive EMO procedure, we attempt to put together some recent salient research results of EMO (described below) to constitute an interactive multi-criterion decision-making procedure:

1. An EMO is capable of finding the *entire* or a *partial* Pareto-optimal set, as desired [4, 2].
2. An EMO is capable of finding the *knee* solutions from the Pareto-optimal set [1].
3. An EMO is capable of finding the *robust* solutions [6].
4. A local search starting from the EMO-obtained solutions has better convergence properties [7].
5. An EMO is capable of handling nonlinear constraints with extra additional parameter [4].

Further, in order to make a reliable decision, we ensure a verification procedure based on available multi-objective optimization techniques so that the decision-maker is satisfied and confident in deciding every step. The following classical principles can be used for this purpose and also for assisting in choosing a specific solution:

1. The extreme Pareto-optimal solutions can be verified by finding the individual optimum of each objective function (with constraints) [10].
2. Multiple applications of ϵ -constraint method can generate a set of Pareto-optimal solutions [10].
3. Interactive multi-objective methods, involving Tchebyshev methods or surrogate trade-off information [10] can be used to help choose a particular solution from the Pareto-optimal set.

In the following, we suggest an interactive-EMO procedure. The parameters to be supplied by the decision-maker (DM) are mentioned in parenthesis.

Step 1: Apply an EMO to obtain a non-dominated front close to the true Pareto-optimal front with following options:

- 1.1 Compute the complete front (DM: no parameter)
- 1.2 Compute a partial front (DM: limiting trade-off values)
- 1.3 Compute *knee* solutions only (DM: limiting angle)
- 1.4 Compute the *robust* Pareto-frontier using one of the following two strategies:
 - 1.4.1 Using mean objective values (DM: neighborhood size)
 - 1.4.2 Constrained approach (DM: neighborhood size and limiting robustness parameter)

Step 2: Improve the obtained non-dominated front using other optimization methods:

- 2.1 Extreme solution improvement: Single-objective optimization of each objective

- 2.2 Intermediate solution improvement: Local search from solutions chosen as follows:
 - 2.2.1 Automated selection of solutions from non-dominated front: Clustering (DM: number of solutions)
 - 2.2.2 User-defined solutions:
 - 2.2.2.1 Weighted-sum approach (DM: weight vectors)
 - 2.2.2.2 Utility function based approach (DM: utility functions)
 - 2.2.2.3 Tchebyshev function approach (DM: ideal points and L_p norm)
 - 2.2.2.4 Geoffrion proper Pareto-optimal solutions (DM: trade-off limit)
 - 2.3 Construct the modified front
- Step 3:** Final improvement (if any) with a classical approach
 - 3.1 ϵ -constraint method (one or multi-objective) (DM: ϵ -vector)
 - 3.2 Construct the final front
- Step 4:** User chooses a particular region of interest (single or multiply-connected) for focussed search using the above procedure
- Step 5:** Until satisfied, go to Step 1, else declare the chosen solution(s)

We now give a somewhat detail description of the I-EMO procedure.

3.1 Description of the I-EMO Procedure

The main difference between our proposed approach and the existing classical interactive methods [10] is that in our approach, we first attempt to find and depict the DM the extent and shape of the Pareto-optimal front using a few solutions. This procedure, in addition to providing the ideal and nadir points of the problem, will also paint a good picture in the mind of the DM about the shape of the Pareto-optimal frontier which will help the DM later to concentrate on a particular region on the front. Thus, in the very first step of the I-EMO, we propose to apply an EMO (for example, NSGA-II, an efficient multi-objective optimizer [5]) on the problem to obtain a non-dominated front (Step 1). To do so, we provide four options to the DM. In the first option, the DM can find the complete front without applying any restriction on it. But if DM is aware and is interested in focussing on a specific portion of the front, the DM can obtain a partial front using the guided-domination approach [2]. In another option, the partial front can also be obtained by using pre-specified bounds on the objectives at the time of problem definition.

The knee solutions are those from which there is not much motivation for a DM to move out, as by definition one unit of improvement in one objective from a knee point requires a large sacrifice in other objectives [10, 1]. Thus, a DM may be interested in finding only the knee solutions right in the beginning of a simulation and would like to choose one such knee points. This is also kept as an option. It is a commonly agreed that if an optimal solution is sensitive to local perturbation to the decision variables, it is of not much use. This is because in practice, a solution is often difficult to implement exactly with an infinite precision. Thus, the DM may be interested in robust solutions (relatively insensitive to variable perturbation) [6], thereby interested in knowing the robust

frontier right in the beginning. The multi-objective robust study [6] introduced two different ways of defining a robust solution. These ideas would be available for the DM to find the robust frontier.

But it is recommended that the DM start with finding the complete non-dominated front and then in steps prefer knee or robust solutions. Such an initial task will give the DM a good idea of the shape and extent of the Pareto-optimal front before the DM focusses on a particular region.

Once the preliminary front is established through Step 1, the next step is to improve the obtained front by using other optimization concepts. In some problems, the NSGA-II may not entirely converge to the true Pareto-optimal front in a finite number of generations. To speed up the convergence, the I-EMO provides a number of ways to improve the front. First, the extreme solutions of the non-dominated front can be improved by using a single-objective genetic algorithm (SGA) on each objective independently. Second, to improve the intermediate solutions of the non-dominated front, a local-search scheme can be applied to some selected solutions from the NSGA-II front based on the DM's preference. The DM can select these specific solutions by using a number of methods. In the absence of any preference, a well-distributed set of solutions can be chosen by using the k -mean clustering procedure. In this process, almost an evenly-distributed set of solutions will be picked up for local-search [4]. The DM can also choose solutions corresponding to optimal solutions of a *weighted-sum approach*, in which the DM provides preferred weight vectors. The DM can also choose solutions corresponding to optimal solutions of some *utility functions* which may be preferred for the problem. The DM can also choose solutions corresponding to optimal solutions of *Tchebyshev* metrics. The DM can also choose solutions which are similar in concept to Geoffrion proper Pareto-optimal solutions [10]. After performing the local searches the previous NSGA-II front can be updated with the new solutions. If the local-search generates better solutions than the NSGA-II non-dominated solutions, then old solutions can be deleted and the front will be updated using the new solutions.

In the next step, final improvement of the non-dominated front is achieved using the classical ϵ -constraint method [4]. This step is included more for checking if the obtained NSGA-II-cum-local-search procedure has actually obtained the true Pareto-optimal front or not. In this step, the DM can use the basic ϵ -constraint method of converting $(M - 1)$ objectives into constraints or can also use a modified method in which any $(M - j)$ (where $1 \leq j \leq M - 1$) objectives can be converted into constraints and j objectives can be used as objective functions. This process re-confirm or improve the non-dominated set obtained in the earlier step. Taking all these solutions (Initial NSGA-II run, local-search and ϵ -constraint method) together the final non-dominated front is constructed.

The use of different methods in the above steps will give adequate confidence to the DM about the optimality of the final front. Since the above procedure will use only a few solutions to provide a description of the front, the DM can be in a position to analyze them and make a decision for focussing on a particular portion(s) of the front. The focussed region can be a single continuous portion of

the front or some disjoint portions of the front (equally important to the DM). With this description, the the complete procedure above will be run again on the restricted portion of the search space and will be continued till the DM is happy with a particular solution of his/her choice.

3.2 In the Making of the I-EMO Software

At present to carry out such a regorous interactive study, there exists no software. For an adequate user interaction we need an algorithm with powerful GUI with a good file-support facility. To realize such a procedure leading to a software, we are currently developing a software where all the above mentioned activities will be carried out. To this date, a major part of the above activities are worked out and in the following screen-shots, we give a flavour of the upcoming software.

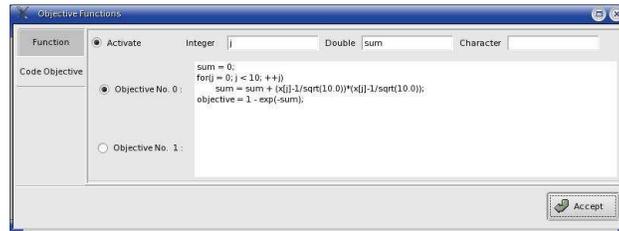


Fig. 1. Window for coding objective function.

In the first screen, the DM needs to define the main parameters of I-EMO, i.e. number of variables, objective functions and constraints for a particular problem. Each parameter is associated with an *Edit* button. Using this further details of a parameter can be provided in the software. An example is shown for the objective function button (Figure 1). In that window the DM can code his objective function(s) using the C-language syntax and the variables must be written as $x[0], x[1] \dots x[i]$. Figure 2 shows the screen for providing the GA parameter values. The input are population size, maximum generation number, crossover and mutation probabilities etc. Figure 3 shows the obtained non-dominated front for an example problem. On this window, several menu buttons are available. At present, the available menus are *Point Menu*, *Utility Menu*, ϵ -*constraint Menu* and *Individual function Optimization*. Application of these menus are described in the previous section. Once the non-dominated front is obtained using the NSGA-II procedure, the ‘Individual Optimization’ button is pressed for individual optimization of each objective and the resultant *ideal objective vector* is plotted on the screen. The pop-up window for ‘Utility’ menu is shown in figure 4. In the first option the DM can give his priority of objective functions in the corresponding text-boxes and press the ‘Find’ button to get the best match on the non-dominated front. Similarly, there exists other buttons for providing information via other methods. For example, for *Tchebycheff’s* weight metric method,

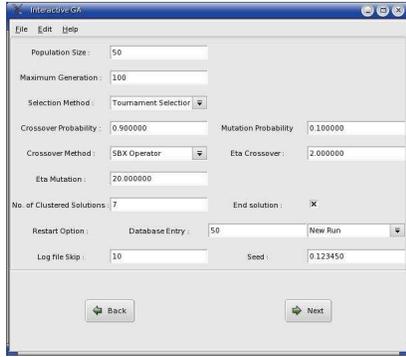


Fig. 2. Window for GA parameters.

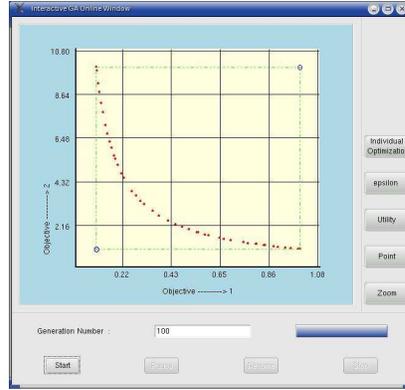


Fig. 3. Plot window for 2d Problem



Fig. 4. Utility Function Window

the DM can pick the *reference point* on the plot window and define the norm and weight-vector in the menu and ask for the best match. The ϵ -constraint menu is shown in the Figure 6. Here any objective function can be chosen as constraints using the toggle button and the corresponding ϵ value is given in the associated text-box. Then SGA is run with this modification and the data is shown in the same window.

Pop No.	Obj No. 0	Obj No. 1
4	Obj Value 0.311289	3.230895
	Weight 0.504320	0.495680

Fig. 5. Point Selection Window



Fig. 6. ϵ -Constraint Window

The I-EMO software can take at-most three objectives for analysis. I-EMO is having an user friendly visualization interface for three objective problem also.

However, a convenient visualization tool is needed for describing Pareto-optimal fronts for more than three objectives. Although there exists some procedures [10, 9], more research in this direction are first needed before one can be implemented for an interactive EMO procedure.

4 Conclusion

In this paper, we, for the first time, have proposed a truly interactive procedure for solving multi-objective optimization and decision-making problems. In order to arrive at the procedure, we have used most salient research results from classical and evolutionary multi-objective optimization literatures in a synergistic manner. The procedure not only finds the true Pareto-optimal front and then helps the DM to choose a particular solution, the procedure includes *checks and balances* at various steps so that the DM is more confident than ever in arriving at a particular solution. Till now, no such combined (classical and EMO) software is available for this task. With the ground-breaking research and application studies using EMO, it is now time for researchers to think and develop such interactive methods which will make give the multi-objective optimization studies a real practical meaning and useful.

Acknowledgements

The support from ST Microelectronics under the ME/STMICR/20040094 Project is acknowledged.

References

1. J. Branke, K. Deb, H. Dierolf, and M. Osswald. Finding Knees in multi-objective Optimization. Technical Report 2004010, Indian Institute of Technology, Kanpur: Kanpur Genetic Algorithms Laboratory (KanGAL), 2004.
2. J. Branke, T. Kaußler, and H. Schmeck. Guidance in Evolutionary Multi-objective Optimization. *Advances in Engineering Software*, 32:499–507, 2001.
3. V. Chankong and Y. Y. Haimes. *Multiobjective Decision Making Theory and Methodology*. New york: North-Holland, first edition edition, 1983.
4. K. Deb. *Multi-objective Optimization using Evolutionary Algorithms*. Chichester, UK: Wiley, second edition edition, 2001.
5. K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
6. K. Deb and H. Gupta. Introducing Robustness in Multi-objective Optimization. Technical Report 2004016, Indian Institute of Technology, Kanpur: Kanpur Genetic Algorithms Laboratory (KanGAL), 2004.
7. K. Deb and S. Jain. Running performance metrics for evolutionary multi-objective optimization. In *Proceedings of the Fourth Asia-Pacific Conference on Simulated Evolution and Learning (SEAL-02)*, pages 13–20, 2002.

8. C. M. Fonseca and P. J. Fleming. Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part II: Application example. *IEEE Transactions on Systems, Man, and Cybernetics: Part A: Systems and Humans*, 28(1):38–47, 1998.
9. A. V. Lotov, V. A. Bushenkov, and H. K. Kamenev. *Interactive Decision Maps*. Kluwer, Boston, 2004.
10. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer’s Academic Publisher, Boston, 1999.
11. K. C. Tan, T. H. Lee, D. Khoo, and E. F. Khor. A multiobjective evolutionary algorithm toolbox for computer-aided multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 31(4):537–556, 2001.
12. A.P. Wierzbicki. Reference point approaches. In T. Gal, T. Stewart, and T. Hanne, editors, *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications*, pages 9.1–9.39. Kluwer Academic Publishers, Boston, 1999.