The chaotic ϵ-constraint approach

Mohamed F. El-Santawy and A. N. Ahmed

Departement of Mathematical Statistics, Institute of Statistical Studies and Research (ISSR),
Cairo University, Egypt. lost_zola@yahoo.com

Abstract: in this paper, we try to tackle the drawbacks of the well-known Multi-Objective technique ϵ-constraint method, namely the computational difficulties and obtaining proper efficient solutions. We incorporate a well-known chaotic function, so-called the logistic map to the classical ϵ-constraint method for improving its results. A well known bench-mark test function is adopted for validation of the new approach, showing its ability to explore various areas of the pareto-optimal front in an efficient way. The chaotic ϵ-constraint approach obtains diversified as well as well representative solutions.
[Mohamed F. El-Santawy and A. N. Ahmed. **The chaotic ϵ-constraint approach.** Journal of American Science 2011;7(7):470-472].(ISSN: 1545-1003). http://www.americanscience.org.

Keywords: Chaos**,** ϵ-constraint method, Logistic Map, Multi-Objective Optimization

1. Introduction

Multi-Objective programming is a part of mathematical programming dealing with decision problems characterized by multiple and conflicting objective functions that are to be optimized over a feasible set of decisions. Such problems, referred to as Multi-Objective Programs (MOPs), are commonly encountered in many areas of human activity including engineering, management, and others [7]. In general, a k-objective minimization problem can be formulated as:

$$\min \{(f_1(x),...,f_k(x)): x \in X\} \quad (1)$$

We usually assume that the set X is given implicitly in the form of constraints resulted in the feasible region in the decision space [10], i.e.,

$$X := \{x \in R^n : g_j(x) \le 0, j = 1,...,s;\ h_j(x) = 0, j = 1,...,m\} \quad (2)$$

One of the main differences between single objective and multi-objective optimization is that in multi-objective optimization the objective functions constitute a multi-dimensional space, in addition to the usual variable space. For each solution x in the variable space, there exists a point in the objective space, denoted by $f(x) = z = (z_1, z_2, \ldots, z_k)^T$. A mapping exists between an n-dimensional solution vector and a k-dimensional objective vector through the objective function, constraints, and variable bounds [5]. Fig.1 illustrates these two spaces and a mapping between them.

Definition 1 (Pareto Dominance): Without loss of generality in a minimization problem, a decision vector $x_1 \in X$ is said to *dominate* a decision vector $x_2 \in X$ *iff* the following two conditions are satisfied:

1. The decision vector x_1 is not worse than x_2 in all objectives, or $\forall\ i \in \{1,2,\ldots,k\} : f_i(x_1) \le f_i(x_2)$.
2. The decision vector x_1 is strictly better than x_2 in at least one objective, or $\exists\ i \in \{1,2,\ldots,k\} : f_i(x_1) < f_i(x_2)$.

If any of the above conditions is violated, then x_1 does not dominate x_2. A decision vector $x_1 \in X$ is called *Pareto-optimal* if there is no another $x_2 \in X$ that dominates it and in this case x_1 is called nondominated with respect to X; also an objective vector is called *Pareto-optimal* if the corresponding decision vector is *Pareto-optimal.*

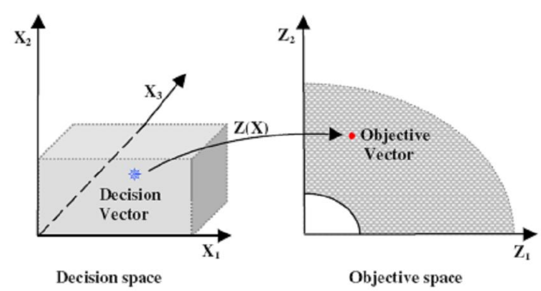

Fig. 1. Decision and Objective spaces in MOP [3]

Definition 2 (Pareto Optimal Set): The Pareto Optimal Set P^* is defined by [11]:

$$P^* = \{x \in X \mid x \text{ is } \textit{pareto-optimal}\} \quad (3)$$

Many researchers had developed a lot of mathematical programming techniques to solve Multi-Objective Optimization problems, some representatives of this class of techniques are the weighting method, the ϵ-constraint method, and the goal programming; also some authors had adopted the Evolutionary Algorithms (EAs), Simulated Annealing (SA), Genetic Algorithms (GA), as well as Swarm Intelligence (SI) techniques to deal with Multi-Objective Problems [6].

The main purpose of this paper to obtain the pareto optimal solutions in an efficient way by incorporating chaos to the classical ϵ-constraint approach to enhance the performance and tackle the main drawback of the original method. The rest of the paper is structured as following; in section 2 the ϵ-constraint method is illustrated, section 3 is devoted to chaos, section 4 is made for the proposed method, finally section 5 is for the numerical example.

2. The ϵ-constraint approach

The ϵ-constraint method first appeared in [9] and is discussed in detail in Changkong and Haimes [4]. It is based on a scalarization where one of the objective functions is minimized while all the other objective functions are bounded from above by means of additional constraints in the following manner:

$$\min \{(f_p(x): f_i(x) \le \varepsilon_i,\ i \ne p,\ x \in X\} \qquad (4)$$

Where $\epsilon_P = (\epsilon_1,...,\epsilon_{P-1},\epsilon_{P+1},...,\epsilon_k)^T \in R^{k-1}$ and p $\in \{1,\dots,k\}$.

The main idea of the ϵ-constraint method is to solve a sequence of ϵ-constraint problems P(ϵ), that are defined by transforming one of the objectives into a constraint [8]. Many issues aroused considering the limitations and drawbacks after the original version appeared, especially, when dealing with high-dimensional problems. The process of obtaining a proper efficient solution of the reduced single program, stability of the solution, beside the feasibility of the single programs depending on values chosen of (ϵ) which definitely will be reflected on the number of computations done. Although, Recently Few proposals tried to tackle the limitations mentioned [7,8], it seems to be open challenging issues for researchers.

3. Chaos

The scientific meaning of the term Chaotic System or Chaos for short is a phenomenon that has deterministic rules behind irregular appearances [2]. Chaos is a kind of common nonlinear phenomenon, which has diverse, complex and sophisticated native under apparent disorder. Chaotic motion is characterized by ergodicity, randomness, and 'regularity' which can traverse all status according to its own 'rule' without repetition [12]. In recent years, the theories and applications of nonlinear dynamics, especially of chaos, have drawn more and more attention in many fields. One is chaos controlling, and synchronization. Another field is the potential applications of chaos in various disciplines including optimization [13]. Mathematically, chaos is randomness of a simple deterministic dynamical system and chaotic system may be considered as sources of randomness. A chaotic map is a discrete-time dynamical system running in chaotic state of the following form:

$$z_{k+1} = f(z_k)\ ,\ 0 < z_k < 1, \quad k = 0,1,2,... \qquad (5)$$

The chaotic sequence $\{ z_k : k = 0,1,2,...\}$ can be used as spread-spectrum sequence as random number sequence. Merely a few functions (chaotic maps) and few parameters (initial conditions) are needed even for very long sequences. In addition, an enormous number of different sequences can be generated simply by changing its initial condition. Recently, chaotic sequences have been adopted instead of random sequences and very interesting and somewhat good results have been shown in many applications [1]. The approach proposed through this paper based on a well known chaotic map in the field, so-called the logistic map [13]:

$$z_{k+1} = 4\, z_k\,(1 - z_k) \qquad (6)$$

4. The chaotic ϵ-constraint approach

The aim of the ϵ-constraint is to approximate the Pareto set by solving a sequence of constrained single-objective problems depending on the ϵ values for all objectives considered, choosing these values improperly might leads that the single programs are not feasible or provide inefficient solutions. Chaotic sequences generated from maps (the logistic map is considered in this paper) are employed to guarantee the feasibility of the single programs, while the traverse without repetition of chaos make it explore the pareto front more efficiently, resulting in better diversification of solutions. Without loss of generality, the chaotic ϵ-constraint method for *k*- minimization problem will be as following:

Step 1. For $i \ne j,\ j = 1,2,...,i\text{-}1,i+1,...,k$
Set $i = 1$

Step 2. Compute the Ideal and Nadir vectors:
$f^I = (f_1^I, f_2^I, f_3^I,..., f_k^I)$
$f^N = (f_1^N, f_2^N, f_3^N,..., f_k^N)$

Step 3. Initiate the vector of chaotic variables (z) $\in R^{k-1}$ Such that $0 < z_j < 1$

Step 4. While $m \le$ No. of iterations, perform the following steps:
- (a) $\epsilon_j = f_j^N - z_j\,(f_j^N - f_j^I)$
- (b) Solve $P_m(\epsilon)$ and add solutions obtained to the set of nondominated solutions
- (c) $z_j = 4\, z_j\,(1 - z_j)$, $m = m+1$

Step 5. While $i < k$, Set $i = i+1$, j as shown in step 1, m =1, and then go to step 4

5. Numerical example

In this section, we adopted one of the well known test functions in the specialized literature called ZDT1 [14], this bench-mark test function being used widely in Multi-Objective Optimization area, have different points of difficulties, which make the covering process of the pareto front is so complicated. ZDT1 have two objectives to be optimized over a feasible set originated from 30 design variables included, Pareto-optimal front is convex, continuous, and have uniform distribution of solutions.

ZDT1 (2-objective, 30 parameters):

$$\min\ f_1(x_1) = x_1$$

$$\min\ f_2(\vec{x}) = g(x_2,....,x_n).h(f_1,g)$$

$$g(x_2,....,x_n) = 1 + 9(\sum_{i=2}^{n} x_i)/(n-1) \quad (7)$$

$$h(f_1,g) = 1 - \sqrt{f_1/g}$$

$$x_i \in [0,1], n = 30, i = 1,....,n$$

A code was developed under MATLAB software environment to solve the ZDT1 problem, applying m = 100, k = 2 to solve this bi-objective problem. Fig. 2 shows the pareto-optimal front produced by the proposed approach in the Objective space after solving 200 single-objective constrained problems (100 for each objective).

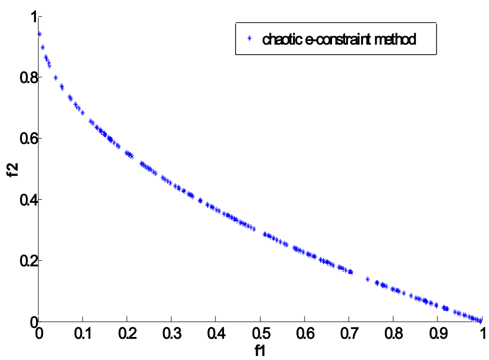

Fig. 2. Pareto-optimal front produced in ZDT1

As shown in Fig. 2, the pareto-optimal front shows the diversified set of nondominated solutions along all the pareto front, which reflects the ability of the new proposed approach to explore the whole solution set efficiently to produce well representative solution set. The new approach did not stuck into infeasible single-objective programs to be solved like the previous related work due to the ergodicity, and randomness of the chaotic variables leads to avoiding the limitation of improper efficient solutions, also the chaotic nature implies more diversification in solutions due to non repetition of the chaotic variables which limits extra computations.

References

1. Alatas, B. (2010). *Chaotic bee colony algorithms for global numerical optimization*", Expert System with Applications, 37: 5682–5687.
2. Alligood, K. T., Sauer, T. D. and Yorke, J. A. (2000). Chaos: An Introduction to Dynamical Systems, Springer, Berlin.
3. Adra, S. F. (2007).*Improving Convergence, Diversity and Pertinency in Multiobjective Optimisation.* PhD Thesis, Department of Automatic Control and Systems Engineering, The University of Sheffield, United Kingdom.
4. Chankong, V., Haimes, Y. (1983). Multiobjective Decision Making Theory and Methodology, Elsevier, New York.
5. Deb, k. (2001). Multi-objective optimization using evolutionary algorithms, John Wiley and Sons.
6. El-Santawy, M. F. (2009). *An Intelligent Search Technique for Solving Multi-Objective Optimization Problems*. MSc thesis, Department of Operations Research, Institute of Statistical Studies and Research (ISSR), Cairo University, Egypt.
7. Ehrgott, M. , Ruzika, S. (2008). *Improved ε-Constraint Method for Multiobjective Programming*. Journal of Optimization Theory and Applications, (138): 375–396.
8. Grandinetti, L., Guerriero, F., Lagana, D. and Pisacane, O. (2010). *An Approximate ε-Constraint Method for the Multi-objective Undirected Capacitated Arc Routing Problem*. SEA 2010, Lecture Notes in Computer Science (6049) 214–225.
9. Haimes, Y. Y., Lasdon, L. S.,Wismer, D. A. (1971). *On a bicriterion formulation of the problems of integrated system identification and system optimization.* IEEE Transactions on Systems, Man, and Cybernetics, 1: 296–297.
10. Ishibuchi, H. *et al.* (2008). *An empirical study on similarity-based mating for evolutionary multiobjective combinatorial optimization*. European Journal of Operational Research, 188 (1): 57-75.
11. Mostaghim, S. and Teich, J. (2003). *Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)*. In Proceedings of the 2003IEEE Swarm Intelligence Symposium, pages 26–33, Indianapolis, Indiana, USA, IEEE Service Center.
12. Shuang-xin, W. *et al.* (2009).*Nonlinear Predictive Load Control of Boiler-Turbine-Generating Unit Based on Chaos Optimization*". In the proceedings of the 2nd Chaotic Modeling and Simulation International Conference, 1-5 June, Chania Crete Greece.
13. Yang, D. *et al.* (2007). *On the efficiency of chaos optimization algorithms for global optimization*. Chaos Solitons and Fractals, 34: 1366–1375.
14. Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD Thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.

6/6/2011