Applied Mathematical Modelling 38 (2014) 2000–2014

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

An interactive dynamic approach based on hybrid swarm optimization for solving multiobjective programming problem with fuzzy parameters

M.A. Abo-Sinna [a], Y. Yousria Abo-Elnaga [b,c], A.A. Mousa [d,e,*]

[a] Department of Mathematics, Faculty of Science, Princess Nora Bint Abdul Rahman University, Riyadh, Saudi Arabia
[b] Department of Mathematics, Faculty of Sciences, Tebah University, Saudi Arabia
[c] Department of Basic Science, Higher Technological Institute, Tenth of Ramadan City, Egypt
[d] Department of Basic Engineering Science, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt
[e] Department of Mathematics and Statistics, Faculty of Sciences, Taif University, Saudi Arabia

ARTICLE INFO

Article history:
Received 2 December 2011
Received in revised form 7 September 2013
Accepted 8 October 2013
Available online 25 October 2013

Keywords:
Multiobjective programming
Dynamic programming
Swarm optimization
Genetic algorithm

ABSTRACT

Real engineering design problems are generally characterized by the presence of many often conflicting and incommensurable objectives. Naturally, these objectives involve many parameters whose possible values may be assigned by the experts. The aim of this paper is to introduce a hybrid approach combining three optimization techniques, dynamic programming (DP), genetic algorithms and particle swarm optimization (PSO). Our approach integrates the merits of both DP and artificial optimization techniques and it has two characteristic features. Firstly, the proposed algorithm converts fuzzy multiobjective optimization problem to a sequence of a crisp nonlinear programming problems. Secondly, the proposed algorithm uses H-SOA for solving nonlinear programming problem. In which, any complex problem under certain structure can be solved and there is no need for the existence of some properties rather than traditional methods that need some features of the problem such as differentiability and continuity. Finally, with different degree of α we get different α-Pareto optimal solution of the problem. A numerical example is given to illustrate the results developed in this paper.

1. Introduction

Multiobjective optimization (MOO) is an important research topic for both scientists and engineers. In MO, a set of non-dominated solutions is usually produced instead of single recommended solution. According to the concept of dominance, also referred as Pareto optimality, a solution to a MO problem is non-dominated, or Pareto optimal, if no objective can be improved without worsening at least one other objective. Naturally, these objective functions and constraints involve many parameters whose possible values may be assigned by the experts. In the traditional approaches, such parameters are fixed at some values in an experimental and/or subjective manner through the experts' understanding of the nature of the parameters. In practice, however, it is natural to consider that the possible values of these parameters are often only ambiguously known to experts' understanding of the parameters as fuzzy numerical data which can be represented by means of fuzzy subsets [1] of the real line known as fuzzy numbers. Recently, Sakawa [2] introduced the concept of α-multiobjective programming based on the α-level sets of the fuzzy numbers and several interactive decision-making methods were introduced

* Corresponding author at: Department of Basic Engineering Science, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt.
E-mail address: a_mousa15@yahoo.com (A.A. Mousa).

0307-904X/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.apm.2013.10.013

[3]. In [4] a method for solving linear programming problems where all the coefficients are, in general, fuzzy numbers was proposed. They offer for the decision-maker (DM) the optimal solution for several different degrees of feasibility. With this information the DM is able to establish a fuzzy goal, and build a fuzzy subset in the decision space whose membership function represents the balance between feasibility degree of constraints and satisfaction degree of the goal. In [5] an effective particle swarm optimization algorithm was presented to find a good approximation of Pareto frontier to fuzzy multi-objective unrelated parallel machines scheduling problem, the proposed algorithm exploits new selection regimes for preserving global as well as personal best solutions. Moreover, a generalized dominance concept in a fuzzy environment is employed to find locally Pareto-optimal frontier. In [6] a hybrid fuzzy multi-objective programming model including both quantities and qualitative constraints and objectives is proposed to determine the optimal price markdown policy and aggregate production planning in a two echelon supply chain. The model aims to maximize the total profit of manufacturer, the total profit of retailer and improving service aspects of retailing simultaneously.

The use and development of heuristic-based optimization techniques have significantly grown. Since they use a population of solutions in their search, it is more likely to find the global solution of a given problem. In recent years there have been a lot of reported works focused on the hybridization of PSO with other heuristic based optimization techniques [7–9]. In [10], PSACO (particle swarm ant colony optimization) algorithm was proposed for highly non convex optimization problems. In [11], GA has been incorporated into PSO as a hybrid method combining two heuristic optimization techniques for the global optimization of nonlinear optimization problem. In [12] an improved multi-objective particle swarm optimization algorithm was applied to cope with a stochastic multi-objective DEED problem. To enhance the overall performance and effectiveness of the particle swarm optimization, a fuzzy adaptive technique, Theta-search and self-adaptive learning strategy for velocity updating are used to tune the inertia weight factor and to escape from local optima, respectively. In [13], a hybrid intelligent algorithm by combining the particle swarm optimization (PSO) with chaos searching technique (CST) is presented for solving nonlinear bilevel programming problems where particles in the front of list are updated by PSO, while particles in the end of list are updated by CST. The CST used here is not only to enhance the particles but also to improve the diversity of the particle swarm so as to avoid PSO trapping the local optima. In [14] predator–prey optimization (PPO) is used as a base level search in the global search space and Powell's method as a local search technique. Predator–prey model is based on particle swarm optimization (PSO). In [15] a hybrid multiobjective evolutionary algorithm combining two heuristic optimization techniques was proposed. It integrates the merits of both genetic algorithm (GA) and particle swarm optimization (PSO), and has two characteristic features. Firstly, the algorithm is initialized by a set of random particles which is flown through the search space. In order to get approximate nondominated solutions PND, an evolution of this particle is performed. Secondly, the local search (LS) scheme is implemented as a neighborhood search engine to improve the solution quality.

Dynamic programming is a useful mathematical technique for making a sequence of interrelated decisions. It provides a systematic procedure for determining the optimal combination of decisions. In contrast to linear programming, there does not exist a standard mathematical formulation of the" dynamic programming problem. Rather, dynamic programming is a general type of approach to problem solving, and the particular equations used must be developed to fit each situation. Therefore, a certain degree of ingenuity and insight into the general structure of dynamic programming problems is required to recognize when and how a problem can be solved by dynamic programming procedures. These abilities can best be developed by an exposure to a wide variety of dynamic programming applications and a study of the characteristics that are common to all these situations. A large number of illustrative examples are presented for this purpose. Dynamic programming technique that divides the problem to be solved into a number of sub problems and then solves each sub-problem in such a way that the overall solution is optimal to the original problem. Dynamic programming is a applicable when subproblem are not independent, that is, when subproblem share subproblems. A dynamic programming algorithms solve every problem just once and then saves its answer in the table, thereby avoiding the work of recomputing the answer every time the Subproblem is encountered. It is typically applied to optimization problems. In such problems there are many possible solutions each solution has a value, and we wish to find a solution with optimal (minimum or maximum) value. We call such a solution an optimal solution to the problem. Dynamic programming has a variety of applications as follows,

1. Bioinformatics [16].
2. Molecular biology [17].
3. Control theory [18–20].
4. Information theory [21,22].
5. Operations research [23].
6. Computer science: theory, graphics, AI [24,25].

Unfortunately, dynamic programming has often been dismissed because it suffers from "the curse of dimensionality." In fact, there are up to three curses of dimensionality: the state space, the outcome space and the action space. The term curse of dimensionality was coined by Richard E. Bellman when considering problems in dynamic optimization [26,27].

In this paper, we present an interactive approach for generating an α-Pareto optimal solution of multiple objective mathematical programming problems with fuzzy parameters which posses certain structure [1,28]. This Method is formally, a natural extension of the work already given by the author Osman et al. 2003 [28] based on evolutionary Genetic algorithm, but the present approach is based on hybrid swarm optimization (H-SOA), which integrates the merits of both GA and PSO.