

Swarm Intelligence for Multi-Objective Optimization in Engineering Design

A

Janga Reddy Manne

Indian Institute of Technology Bombay, India

INTRODUCTION

Nowadays, most of the engineering design problems are intrinsically complex and difficult to solve, because of diverse solution search space, complex functions, continuous and discrete nature of decision variables and hard constraints. Meta-heuristic algorithms are becoming popular in dealing with these kind of complexities. Evolutionary algorithms (EAs) and swarm intelligence (SI) algorithms are being population based random search techniques becoming attractive global optimization solvers. The algorithms use guided rules or heuristics inspired from nature to enable effective exploration of optimal solutions to complex engineering problems. In recent past, a number of swarm intelligence (SI) algorithms were proposed based on principles of co-operative group intelligence and collective behavior of self-organized systems. The approaches use agents to perform explorations while they interact with neighbors and the environment. However, the individual members have limited search capabilities, co-operative group intelligence and/or knowledge sharing among the swarm helps to obtain optimal solutions to complex engineering problems. The popular SI algorithms include particle swarm optimization (PSO) that was emerged from simulating the behavior of flocks of birds (Kennedy & Eberhart, 1995), ant colony optimization (ACO) that was emulating the behavior of ants foraging for food (Dorigo, 1992). Other SI algorithms include Honey-bee mating algorithm, glow swarm algorithm, bacterial Foraging and Cuckoo search algorithms etc.

Many times, practical engineering design problems are characterized by multiple conflicting goals. In contrast to single objective optimization, multi-objective optimization deals with simultaneous optimization of several non-commensurable and often competitive/conflicting objectives. Because of the multiple conflicting objectives, it may not be possible to find a single optimal solution that will satisfy all the stated goals, instead, the solution exists in the form of alternative trade-offs, also known as the non-inferior or non-dominated solutions. In the past, several studies have used classical optimization techniques such as linear programming (LP), dynamic programming (DP) and non-linear programming (NLP) to solve the multi-objective problems by adopting weighted-sum or constrained approach etc. These approaches may face difficulties while generating non-dominated solutions for practical problems. For example, in the weighted-sum approach, the multiple objectives of the problem are converted into a single objective optimization by adopting suitable weights to all the objectives. By using a single pair of fixed weights, only one point on Pareto-front can be obtained. Therefore, if one would like to obtain the complete set of Pareto optimal front, all possible Pareto solutions must first be derived. This requires the algorithms to be executed iteratively, so as to ensure that every weight combination has been evaluated. Obviously, it is unrealistic to reiterate the algorithms continually to exhaust all the weight combinations. Similarly, in the constraint method, it needs to reiterate the algorithm for a large number of times, which requires more computational effort. Also

conventional approaches may face difficulties, if optimal solution lies on non-convex or disconnected regions of the objective function space. Thus the classical approaches are not ideal to solve multi-objective optimization problems (MOOP). In developing an algorithm for solution of an MOOP, it should have an ability to learn from past performance, to direct proper selection of weights for further evolutions. To achieve these goals, multi-objective evolutionary algorithms (MOEAs) have been proposed and are suggested as effective means to deal with these issues (Reddy & Kumar, 2007a). Due to their efficiency and easiness to handle non-linear functions, ability to approximate the non-convex and disconnected Pareto optimal fronts of real-world problems, MOEAs are getting diverse applications in engineering design. Apart from that, the specific advantage of MOEAs over the classical approaches is that they generate a population of solutions in each iteration and offer a set of alternatives (Pareto optimal set) in a single run. Thus population based stochastic search techniques are becoming more popular to solve MOOPs.

In the following sections, first the principles and issues in developing multi-objective algorithms are discussed. Then swarm intelligence based algorithm for multi-objective optimization is presented. Subsequently, application of the methodology is illustrated though few multi-objective engineering design problems.

BACKGROUND

Multi-Objective Problem

A general multi-objective optimization problem can be defined as, minimize a set of functions $f(x)$, subject to p inequality and q equality constraints (Reddy & Kumar, 2007a).

$$\text{Min. } f(x) = \{f_1(x) f_2(x) \dots f_m(x)\}^T \quad (1)$$

$$x \in D$$

where $x \in R^n$, $f_i: R^n \rightarrow R$ and

$$D = \left\{ \begin{array}{ll} x \in R^n : & l_i \leq x \leq u_i, \quad \forall i = 1, \dots, n \\ & g_j(x) \geq 0, \quad \forall j = 1, \dots, p \\ & h_k(x) = 0, \quad \forall k = 1, \dots, q \end{array} \right. \quad (2)$$

where m is number of objectives; D is feasible search space; $x = \{x_1 x_2 \dots x_n\}^T$ is the set of n -dimensional decision variables (continuous, discrete or integer); R is the set of real numbers; R^n is n -dimensional hyper-plane or space; l_i and u_i are lower and upper limits of i -th decision variable.

Pareto Optimal Solution

In MOOP, the desired goals are often conflicting against each other and it is not possible to satisfy all the goals at a time, which leads to definition of *Pareto optimal* solutions. The Pareto optimal solution refers to a solution, around which there is no way of improving any objective without degrading at least one other objective.

Pareto front is a set of non-dominated solutions, being chosen as optimal, if no objective can be improved without sacrificing at least one other objective (Deb et al., 2002). On the other hand a solution x^* is referred to as dominated by another solution x , if and only if, x is equally good or better than x^* with respect to all objectives. The definition of Pareto optimality is very much useful in MOEAs to classify the population of solutions into dominated and non-dominated members, thereby helping in the selection of member solutions from one generation to next generation.

Multi-Objective Evolutionary Algorithms

In the last two decades, a number of evolutionary algorithms (EAs) were proposed to solve multi-objective optimization problems. The first generation MOEAs, Non-dominated Sorting Genetic

10 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/swarm-intelligence-for-multi-objective-optimization-in-engineering-design/183738

Related Content

Information Systems Design and the Deeply Embedded Exchange and Money-Information Systems of Modern Societies

G.A. Swanson (2008). *International Journal of Information Technologies and Systems Approach* (pp. 20-37).

www.irma-international.org/article/information-systems-design-deeply-embedded/2537

Research of Biogeography-Based Multi-Objective Evolutionary Algorithm

Hongwei Moand Zhidan Xu (2013). *Interdisciplinary Advances in Information Technology Research* (pp. 125-135).

www.irma-international.org/chapter/research-biogeography-based-multi-objective/74537

Implementing a Customer Relationship Management (CRM) System

Dimitra Skoumpopoulouand Benjamin Franklin (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 1605-1615).

www.irma-international.org/chapter/implementing-a-customer-relationship-management-crm-system/183875

Sustainability Factors of Accessible Information Systems and Technologies (IS&T)

Daryoush Daniel Vaziriand Dirk Schreiber (2015). *Encyclopedia of Information Science and Technology, Third Edition* (pp. 4185-4194).

www.irma-international.org/chapter/sustainability-factors-of-accessible-information-systems-and-technologies-ist/112860

Information Systems, Software Engineering, and Systems Thinking: Challenges and Opportunities

Doncho Petkov, Denis Edgar-Nevill, Raymond Madachyand Rory O'Connor (2008). *International Journal of Information Technologies and Systems Approach* (pp. 62-78).

www.irma-international.org/article/information-systems-software-engineering-systems/2534