Constraint-Handling Method for Multi-objective Function Optimization: Pareto Descent Repair Operator

Ken Harada, Jun Sakuma, Isao Ono, and Shigenobu Kobayashi

Department of Computational Intelligence and Systems Science,
Tokyo Institute of Technology,
4259 Nagatsuta-cho Midori-ku Yokohama-shi Kanagawa-ken 226-8502, Japan
{ken,jun}@fe.dis.titech.ac.jp,{isao,kobayasi}@dis.titech.ac.jp
http://www.fe.dis.titech.ac.jp/

Abstract. Among the multi-objective optimization methods proposed so far, Genetic Algorithms (GA) have been shown to be more effective in recent decades. Most of such methods were developed to solve primarily unconstrained problems. However, many real-world problems are constrained, which necessitates appropriate handling of constraints. Despite much effort devoted to the studies of constraint-handling methods, it has been reported that each of them has certain limitations. Hence, further studies for designing more effective constraint-handling methods are needed.

For this reason, we investigated the guidelines for a method to effectively handle constraints. Based on these guidelines, we designed a new constraint-handling method, Pareto Descent Repair operator (PDR), in which ideas derived from multi-objective local search and gradient projection method are incorporated. An experiment comparing GA that use PDR and some of the existing constraint-handling methods confirmed the effectiveness of PDR.

1 Introduction

Multi-objective optimization (MOO) has many real-world applications, e.g. portfolio optimization, for which multiple conflicting objective functions are to be simultaneously optimized. MOO whose variables are real-valued is called multi-objective function optimization, which is the subject of this paper. Genetic Algorithms (GA) are known to be relatively efficient and effective MOO methods [1]. GA applies crossover and selection to a set of solutions and converge them to entire Pareto-optimal solutions. Selection for MOO consists of *ranking*, which brings solutions closer to Pareto-optimal solutions, and *sharing*, which enhances the diversity of solutions.

Most MOO methods, including GA, were designed for solving primarily unconstrained problems. However, real-world problems often have constraints, and the handling of them can substantially influence the performance of the optimization methods. When GA is applied to constrained problems, two major difficulties arise.

S. Obayashi et al. (Eds.): EMO 2007, LNCS 4403, pp. 156–170, 2007.

One of them is that some GA require feasible solutions to start with. The most naive way of obtaining feasible solutions is to randomly generate solutions until a prespecified number of them are found. However, this approach fails when the probability of obtaining a feasible solution in such a way is very low. Therefore, feasible solutions must be explicitly searched for, which is one role that constraint-handling methods play.

The other difficulty is that, on problems whose Pareto-optimal solutions lie on feasible region boundaries (boundaries hereafter), GA may not be able to obtain solutions close to the Pareto-optimal solutions. The most commonly used constraint-handling method in GA is death penalty (DP), which simply discards infeasible solutions. The solutions that GA generates can be mostly infeasible on problems whose Pareto-optimal solutions lie on boundaries. Extreme examples of such problems are ZDT1 and ZDT2 [1] whose Pareto-optimal solutions form line segments at which 29 constraint boundaries intersect perpendicularly. When the solutions that GA maintains come near the Pareto-optimal solutions, most of the solutions that GA generates are infeasible and discarded by DP, which implies that GA cannot obtain solutions close to the Pareto-optimal solutions. Therefore, effective constraint-handling methods which facilitate searching for Pareto-optimal solutions on boundaries are necessary.

One class of constraint-handling methods modify solution representation and/or crossover so that infeasible solutions can never be generated [2]. However, these methods are not applicable to general problems. Another class of methods attempt to search for feasible solutions from infeasible solutions by reducing constraint violations. The existing methods of this kind are known to have certain limitations as described in Sect. 2.2.

In order to design an effective constraint-handling method, we first investigate the guidelines for a method to effectively handle constraints. We then explain the concepts and calculations necessary to meet these guidelines and propose them as Pareto Descent Repair operator (PDR).

Section 2 formulates constrained multi-objective function optimization, explains Pareto-optimality, and reviews existing constraint-handling methods. Section 3 presents the guidelines for effective constraint handling and explains the details of PDR. To demonstrate the effectiveness of PDR, Sect. 4 shows the results of experiments comparing PDR and other constraint-handling methods when they are used in GA. Lastly, Sect. 5 summarizes this paper.

2 Constraint Handling in Multi-objective Function Optimization

2.1 Constrained Multi-objective Function Optimization

Formulation. Constrained multi-objective function optimization problem can generally be formulated as

$$\text{Minimize } \boldsymbol{f}(\boldsymbol{x}) \text{ subject to } \boldsymbol{x} \in S \ , \tag{1}$$