Multi-objective MaxiMin Sorting Scheme

E. J. Solteiro Pires[1], P. B. de Moura Oliveira[2], and J. A. Tenreiro Machado[3]

[1] Universidade de Trás-os-Montes e Alto Douro, Dep. de Engenharia Electrotécnica,
Quinta de Prados, 5000–911 Vila Real, Portugal
http://www.utad.pt/~epires

[2] Universidade de Trás-os-Montes e Alto Douro, CETAV,
Quinta de Prados, 5000–911 Vila Real, Portugal
{epires,oliveira}@utad.pt
http://www.utad.pt/~oliveira

[3] Instituto Superior de Engenharia do Porto, Dep. de Engenharia Electrotécnica,
Rua Dr. António Bernadino de Almeida, 4200-072 Porto, Portugal
jtm@dee.isep.ipp.pt
http://www.dee.isep.ipp.pt/~jtm

Abstract. Obtaining a well distributed non-dominated Pareto front is one of the key issues in multi-objective optimization algorithms. This paper proposes a new variant for the elitist selection operator to the NSGA-II algorithm, which promotes well distributed non-dominated fronts. The basic idea is to replace the crowding distance method by a maximin technique. The proposed technique is deployed in well known test functions and compared with the crowding distance method used in the NSGA-II algorithm. This comparison is performed in terms of achieved front solutions distribution by using distance performance indices.

1 Introduction

Multi-objective techniques using genetic algorithms (GAs) have been increasing in relevance as a research area. In 1989, Goldberg [1] suggested the use of a GA to solve multi-objective problems and since then other investigators have been developing new methods, such as multi-objective genetic algorithm (MOGA) [2], non-dominated sorted genetic algorithm (NSGA) [3] and niched Pareto genetic algorithm (NPGA) [4], among many other variants [5].

Achieving a well-spread and well-diverse Pareto solution front can be a time consuming computational problem, associated with multi-objective evolutionary algorithms (MOEAs). A good background review about the use of bounded archive population in MOEAs can be found in [6]. The computational complexity is directly related with the level of diversity and distribution the MOEAs aims to obtain. The higher this level, the larger computational power will be required. Indeed, as it was stated in [7] "For example, NSGA-II uses a crowding approach which has a computational complexity of $O(N \log N)$, where N is the population size. On the other hand, SPEA uses a clustering approach which has computational complexity of $O(N^3)$". Also it was found that while for two objectives problems the difference in terms of the achieved solution diversity with NSGA-II and SPEA is not significant, for three objectives problem the SPEA proved to be clearly better, but at the expensive of a higher computational load.

C. A. Coello Coello et al. (Eds.): EMO 2005, LNCS 3410, pp. 165–175, 2005.

Thus, new computational schemes which can find good distributed Pareto fronts with reasonable computational effort are a highly desired feature.

Maximin is a well known method used in classic multi-attribute problems [8] and in the game theory [9, 10]. Recently Balling [11] proposed a multi-objective optimization technique based on a fitness function derived from using the maximin strategy [9] and Li [12] used the maximin fitness in a particle swarm multi-objective optimizer.

Bearing these ideas in mind, this paper, proposes a sorting scheme to select the best solutions in order to promote its diversity within MOEAs. In each generation, the achieved set is initially formed by the best solutions for each objective. Then, the achieved population is completed, one solution at a time, by the maximum of the minimal norm between a solution and the set of solutions already selected.

The article is organized as follows: section 2 describes the proposed method. Section 3 presents the MOEA settings, test functions and performance indices used for performance comparison. Section 4 shows the results and analysis of experiments carried out with maximin sorting scheme. Finally, section 5 outlines the main conclusions.

2 MaxiMin Sorting Scheme

This section presents the maximin sorting algorithm to render the following generation, having a good solution distribution.

The problem addressed by the proposed sorting scheme (maximin) is the selection of the best distributed N solutions from an original population with size M ($M > N$). As it is well known, this is a very useful feature in elitism MOEAs when it is necessary to choose the solutions which passes to the following generation. Similarly to the NSGA-II, the proposed algorithm is based on non-dominated fronts [13]. However, when the last allowed front is being considered, and there are more solutions in the last front than the remaining slots in the population, a maximin function is called to select the last solutions, in spite of using the crowding distance.

The main idea behind the maximin sorting scheme is to select the solutions in order to decrease the large gap areas existing in the already selected population. For example, let us consider the non-dominated solutions in figure 1. Initially the extreme solutions are selected $S \equiv \{a, b\}$. Then, solution c is selected because it has the greater distance to the set S. After that, solutions d and e are selected into the set $S \equiv \{a, b, c\}$, for the same reason. The process is repeated until the S set is completed.

The maximin sorting scheme is depicted in algorithm 1, assuming a minimization problem. Table 1 presents the notation used in algorithm 1. In each generation the new population is merged with the archive population, resulting in set T, from with the new archive is obtained by applying the proposed algorithm (lines 0–1). After that, the algorithm selects the best solutions for each objective (lines 2-4) into the S set. Then the non-dominated front is removed from the non-selected population T into S set until the last allowed front being considered does not fit into the S set (lines 5-9). Therefore, the square of the distance, c_i (1a), between each non-dominated solution and the set of solutions already selected, S, is evaluated, selecting the solution whose distance to the set is greater (1b). Every time a solution enter to the set S the cost c_i, of non-dominated solutions, is reevaluated (lines 10-19). This process ends when the S set is completed.