Design Issues in a Multiobjective Cellular Genetic Algorithm

Antonio J. Nebro, Juan J. Durillo, Francisco Luna, Bernabé Dorronsoro, and Enrique Alba*

Departamento de Lenguajes y Ciencias de la Computación
E.T.S. Ingeniería Informática
Campus de Teatinos, 29071 Málaga (Spain)
{antonio,durillo,flv,bernable,eat}@lcc.uma.es

Abstract. In this paper we study a number of issues related to the design of a cellular genetic algorithm (cGA) for multiobjective optimization. We take as an starting point an algorithm following the canonical cGA model, i.e., each individual interacts with those ones belonging to its neighborhood, so that a new individual is obtained using the typical selection, crossover, and mutation operators within this neighborhood. An external archive is used to store the non-dominated solutions found during the evolution process. With this basic model in mind, there are many different design issues that can be faced. Among them, we focus here on the synchronous/asynchronous feature of the cGA, the feedback of the search experience contained in the archive into the algorithm, and two different replacement strategies. We evaluate the resulting algorithms using a benchmark of problems and compare the best of them against two state-of-the-art genetic algorithms for multiobjective optimization. The obtained results indicate that the cGA model is a promising approach to solve this kind of problem.

1 Introduction

Most optimization problems in the real world are multiobjective in nature. This feature, along with the facts that function evaluations can require a significant computation time and the search spaces tends to be very large, make metaheuristics popular techniques to solve multiobjective optimization problems (MOPs). Among them, evolutionary algorithms have been investigated by many authors, and some of the most well-known algorithms for solving MOPs belong to this class (e.g. NSGA-II [1], PAES [2], SPEA2 [3]). Nevertheless, in recent years there is a trend to adapt other kinds of metaheuristics (sometimes called "alternative methods", with reference to evolutionary algorithms) to the multiobjective field, such as tabu search [4] or scatter search [5].

Many evolutionary algorithms for solving MOPs are some kind of genetic algorithm (GA). These algorithms work over a set (*population*) of potential solutions

* This work has been partially funded by the Ministry of Science and Technology and FEDER under contract TIN2005-08818-C04-01 (the OPLINK project).

S. Obayashi et al. (Eds.): EMO 2007, LNCS 4403, pp. 126–140, 2007.

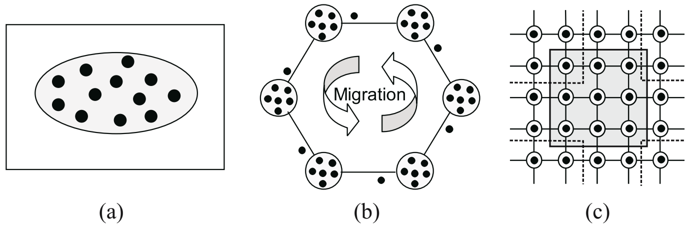

Fig. 1. Panmictic (a), distributed (b), and cellular (c) GAs

(*individuals*) which undergoes stochastic operators in order to search for better solutions. These operators are typically selection, crossover, and mutation. Most GAs use a single population (panmixia) of individuals and apply the operators to them as a whole (see Fig. 1a). Conversely, there exist the so-called structured GAs, in which the population is decentralized somehow. Among the many types of structured GAs, the distributed and cellular models are two popular variants [6,7] (see Fig. 1b and Fig. 1c, respectively). In many cases, these decentralized algorithms provide a better sampling of the search space, resulting in an improved numerical behavior with respect to an equivalent algorithm in panmixia.

In this work, we focus on the cellular model of GAs (cGAs). This kind of GAs uses the concept of (small) *neighborhood* in the sense that an individual may only interact with its nearby neighbors in the breeding loop [8,9,10]. The overlapped small neighborhoods of cGAs help in exploring the search space because the induced slow diffusion of solutions through the population provides a kind of exploration (diversification), while exploitation (intensification) takes place inside each neighborhood by genetic operations. It is worth mentioning that the neighborhood is defined among tentative solutions in the algorithm, with no relation to the geographical neighborhood definition in the problem space.

cGAs have proven to be very effective tools for solving a diverse set of single objective optimization problems from both classical and real world settings [11,12], but little attention has been paid to their use in the multiobjective optimization field [13,14,15,16,17]. In [18] we proposed the MultiObjective Cellular (MOCell) algorithm, which, together with cMOGA [16] is the unique existing adaptation of the canonical cGA model to the multiobjective field. MOCell uses an external archive to store the non-dominated solutions found during the execution of the algorithm, like many other multiobjective evolutionary algorithms do (e.g., PAES, SPEA2).

MOCell is characterized by selecting a fixed number of individuals from the archive to replace the same number of randomly chosen individuals from the population (archive feedback) at the end of each iteration of the algorithm. This is carried out with the hope of taking advantage of the search experience in order to find a Pareto front with good convergence and spread. This new replacement coexists with the typical replacement of a canonical cGA, in which the newly generated individual replaces the current one if the latter is worse than the former. However, there are many other different strategies that could