ParadisEO-MOEO: A Framework for Evolutionary Multi-objective Optimization

Arnaud Liefooghe, Matthieu Basseur, Laetitia Jourdan, and El-Ghazali Talbi

INRIA Futurs, Laboratoire d'Informatique Fondamentale de Lille (LIFL), CNRS
Bât. M3, Cité Scientifique, 59655 Villeneuve d'Ascq cedex, France
{liefooga,basseur,jourdan,talbi}@lifl.fr

Abstract. This paper presents ParadisEO-MOEO, a white-box object-oriented generic framework dedicated to the flexible design of evolutionary multi-objective algorithms. This paradigm-free software embeds some features and techniques for Pareto-based resolution and aims to provide a set of classes allowing to ease and speed up the development of computationally efficient programs. It is based on a clear conceptual distinction between the solution methods and the multi-objective problems they are intended to solve. This separation confers a maximum design and code reuse. ParadisEO-MOEO provides a broad range of archive-related features (such as elitism or performance metrics) and the most common Pareto-based fitness assignment strategies (MOGA, NSGA, SPEA, IBEA and more). Furthermore, parallel and distributed models as well as hybridization mechanisms can be applied to an algorithm designed within ParadisEO-MOEO using the whole version of ParadisEO. In addition, GUIMOO, a platform-independant free software dedicated to results analysis for multi-objective problems, is briefly introduced.

Keywords: object-oriented frameworks, design and code reuse, multi-objective optimization, evolutionary algorithms.

1 Introduction

Nowadays, the usefulness of Multi-Objective Optimization (MOO) is globally established in the whole operational research community. Furthermore, evolutionary algorithms (EAs) are commonly used to solve multi-criterion problems since they naturally found a well-diversified set of good-quality solutions. EAs [12] are stochastic optimization processes based on an iterative improvement of a population of solutions (called individuals). As discussed later in the paper, several frameworks such as MOEA [20], MOMHLib++, Open BEAGLE [9], PISA [2], TEA [7] (to quote only them) already attempt to simplify and accelerate the development process of evolutionary MOO applications. We here propose a new library, called ParadisEO-MOEO (MOEO for short), that aims to produce efficient programs while having a minimal programming effort and a maximum code reuse. MOEO (Multi-Objective Evolving Objects) is an extension of the Evolving Objects framework [15]. It includes a broad range of reusable features and

S. Obayashi et al. (Eds.): EMO 2007, LNCS 4403, pp. 386–400, 2007.

techniques related to Pareto-based MOO such as performance metrics, elitism, fitness sharing and the most common Pareto-based fitness assignment schemes: MOGA, NSGA, NSGA-II, SPEA, SPEA2, IBEA, ... The fine-grained components of MOEO confer a high genericity, flexibility, adaptability and extensibility. Thus, a genuine conceptual effort has been done in order to allow the user to write only the minimum problem-specific code and to incrementally adapt an algorithm rather than entirely re-implementing it. Moreover, MOEO is itself extented to compose the full ParadisEO framework which is devoted to hybridization and parallel/distributed computing. Besides, MOEO has already been used to solve various academic problems likewise real-world applications.

The remainder of the paper is organized as follows. Section 2 gives the necessary background about MOO. Sections 3 and 4 describe the aims, the implementation and the provided features of the MOEO framework. In section 5, we present ParadisEO as well as the most common parallel/distributed models and hybridization mechanisms for multi-objective problems. In section 6, we survey some existing MOEO-designed applications and we introduce a Graphical User Interface for Multi-Objective Optimization (GUIMOO). Finally, the last section concludes the paper and highlights several perspectives about this work.

2 Multi-objective Optimization

Widely investigated since the end of the 1980's, multi-objective optimization concerns many areas of the industry (telecommunication, transport, aeronautics, etc). In this section, we briefly present some required notions about Pareto-based multi-objective optimization such as the formulation of a multi-objective optimization problem (MOOP) and some concepts relating to Pareto optimality (the reader is referred to [4,5] for more details).

Multi-objective optimization problem. A MOOP is defined by a decision space D, an objective space Z, and $n \geq 2$ objective functions $f_1, f_2, \ldots, f_n$. Each objective function can be either minimized or maximized. A solution $x = (x_1, x_2, \ldots, x_k)$ is represented by a vector of k decision variables. To each solution $x \in D$ is assigned exactly one objective vector $z \in Z$ on the basis of a vector function $F : X \rightarrow Z$ with $z = F(x) = (f_1(x), f_2(x), \ldots, f_n(x))$.

Pareto optimality. A multi-objective algorithm aims to approximate the set of Pareto optimal solutions according to F. A solution $x_a \in D$ is Pareto optimal if there exists no solution $x_b \in D$ that dominates x_a. For a minimization problem, the Pareto dominance relation is defined as follows:

Definition 1. *A solution $x_a \in D$ dominates a solution $x_b \in D$ if and only if $\forall i \in [1..n]$, $f_i(x_a) \leq f_i(x_b)$ and $\exists i \in [1..n]$ such as $f_i(x_a) < f_i(x_b)$.*

The overall goal is then to find a well-converged and well-diversified set of Pareto optimal solutions.

These basic notions already emphasize the most important points to consider for the design of a library devoted to evolutionary multi-objective optimization.