European Journal of Operational Research 239 (2014) 731–743

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Decision Support

Distributed localized bi-objective search

Bilel Derbel [a,b], Jérémie Humeau [c], Arnaud Liefooghe [a,b,*], Sébastien Verel [d]

[a] Université Lille 1, LIFL – CNRS, 59655 Villeneuve d'Ascq Cedex, France
[b] Inria Lille-Nord Europe, 59650 Villeneuve d'Ascq, France
[c] École des Mines de Douai, départment IA, 59508 Douai, France
[d] Université du Littoral Côte d'Opale, LISIC, 62228 Calais, France

ARTICLE INFO

Article history:
Received 28 June 2012
Accepted 24 May 2014
Available online 4 June 2014

Keywords:
Multiple objective programming
Combinatorial optimization
Parallel and distributed computing
Evolutionary computation

ABSTRACT

We propose a new distributed heuristic for approximating the Pareto set of bi-objective optimization problems. Our approach is at the crossroads of parallel cooperative computation, objective space decomposition, and adaptive search. Given a number of computing nodes, we self-coordinate them locally, in order to cooperatively search different regions of the Pareto front. This offers a trade-off between a fully independent approach, where each node would operate independently of the others, and a fully centralized approach, where a global knowledge of the entire population is required at every step. More specifically, the population of solutions is structured and mapped into computing nodes. As local information, every node uses only the positions of its neighbors in the objective space and evolves its local solution based on what we term a '*localized fitness function*'. This has the effect of making the distributed search evolve, over all nodes, to a high quality approximation set, with minimum communications. We deploy our distributed algorithm using a computer cluster of hundreds of cores and study its properties and performance on ρMNK-landscapes. Through extensive large-scale experiments, our approach is shown to be very effective in terms of approximation quality, computational time and scalability.

1. Introduction

1.1. Context and motivation

Many real-life problems arising in a wide range of application fields can be modeled as multi-objective optimization problems. One of the most challenging issues in multi-objective optimization is to identify the set of Pareto optimal solutions, *i.e.*, solutions providing the best compromises between the objectives. It is well understood that computing such a set is a difficult task. Designing efficient heuristic algorithms for multi-objective optimization requires one to tackle the classical issues arising in the single-objective case (*e.g.*, intensification *vs.* diversification), but also and more importantly, to find a set of solutions having good properties in terms of trade-off distribution in the objective space.

When dealing with such sophisticated problems, it is with no surprise that most existing approaches are costly in terms of computational complexity. A natural idea is to subdivide the problem being solved into subtasks which can be processed in parallel. This is a very intuitive idea when dealing with computing intensive applications, not only in the optimization field but in computer science in general. Besides, with the increasing popularity of high performance (*e.g.*, clusters), massively parallel (*e.g.*, multi-cores, GPUs), and large-scale distributed platforms (*e.g.*, grids, clouds), it is more and more common to distribute the computations among available resources taking much benefit of the induced huge computational power. Many parallel/distributed models and algorithms have been designed for specific optimization contexts. This witnesses the hardness of the tackled problems and the complexity of related algorithmic issues. Multi-objective optimization does not stand for an exception, since the multi-objective nature of the problem being solved induces additional computing intensive tasks.

One can find an extensive literature on designing parallel/distributed multi-objective solving methods (Van Veldhuizen, Zydallis, & Lamont, 2003; Coello Coello, Lamont, & Van Veldhuizen, 2007; Talbi et al., 2008; Bui, Abbass, & Essam, 2009). Most existing approaches are designed in a top-down manner, starting with a centralized algorithm requiring a *global* information about the search state; and then trying to adapt its components to the distributed/parallel computing environment. This design process usually requires to tackle parallel-computing issues which

* Corresponding author at: LIFL, Cité scientifique, Bât. M3, Université Lille 1, 59655 Villeneuve d'Ascq cedex, France. Tel.: +33 3 59 35 86 30.

E-mail addresses: bilel.derbel@lifl.fr (B. Derbel), jeremie.humeau@mines-douai.fr (J. Humeau), arnaud.liefooghe@univ-lille1.fr (A. Liefooghe), verel@lisic.univ-littoral.fr (S. Verel).

http://dx.doi.org/10.1016/j.ejor.2014.05.040
0377-2217/

are challenging to solve efficiently and/or may impact the performance of the original sequential optimization algorithm. In contrast, *locality* in distributed computing is a well-known general paradigm that states that global information is not always necessary to solve a given problem and local information is often sufficient (see *e.g.*, Peleg (2000)). Therefore, adopting a *localized* approach when tackling a given problem can allow one to derive novel algorithms which are by essence parallel and designed in a bottom-up manner. Those algorithms are more likely to allow distributed resources to coordinate their actions/decisions locally, and to take full benefit of the available computational power.

1.2. Contribution overview

In this paper, we describe a new simple and effective generic scheme dedicated to bi-objective heuristic search in distributed/parallel environments. Our approach is inherently *local*, meaning that it is thought to be independent of any global knowledge. Consequently, its deployment on a large-scale distributed environment does not raise parallel-specific issues.

Generally speaking, each computing node contains a candidate solution and is able to search in a region of the search space in coordination with other neighboring nodes. The sub-region where a node operates is delimited implicitly in an adaptive way based on the relative position of its cooperating neighbors in the objective space. The way local cooperation is designed, as well as its induced optimization process, are the heart of our approach. In our study, we propose novel *localized cooperative strategies* inspired by the classical weighted-sum scalarizing function (Ehrgott, 2005) and hypervolume-based approaches (Zitzler & Thiele, 1999), without requiring any global knowledge about the search state. The designed rules allow distributed nodes to self-coordinate their decisions adaptively and in an autonomous way while communicating a minimal amount of information; thus being effective when deployed on a real and large-scale distributed environment. To evaluate the performance of our approach, we conduct extensive experiments involving more than two hundred computing cores, and using ρMNK-landscapes (Verel, Liefooghe, Jourdan, & Dhaenens, 2013) as a benchmark. As baseline algorithms, we consider both a pure parallel strategy and an inherently sequential approach. Our experimental results show that our localized approach is extremely competitive in terms of approximation quality; while being able to achieve near-linear speed-ups in terms of computational complexity. Besides, we provide a comprehensive analysis of our approach highlighting its properties and dynamics.

1.3. Outline

In Section 2, we review existing works related to multi-objective optimization, especially those dealing with parallel and distributed issues. In Section 3, we describe the distributed localized bi-objective search approach proposed in the paper, and give a generic fully distributed scheme which can be instantiated in several ways. In Section 4, we provide the experimental setup of our analysis. In Section 5, we present numerical results and we discuss the properties of our approach. Finally, we conclude the paper in Section 6 and we discuss some open research issues.

2. Background on multi-objective optimization

In the following, we first introduce the basics of multi-objective optimization and then we position our work with respect to the literature.

2.1. Definitions

A *multi-objective optimization problem* can be defined by an objective function vector $f = (f_1, f_2, \ldots, f_M)$ with $M \geqslant 2$, and a set $\mathcal{X}$ of feasible solutions in the *solution space*. In the combinatorial case, $\mathcal{X}$ is a discrete set. Let $\mathcal{Z} = f(\mathcal{X}) \subseteq \mathbb{R}^M$ be the set of feasible outcome vectors in the *objective space*. To each solution $x \in \mathcal{X}$ is then assigned exactly one objective vector $z \in \mathcal{Z}$, on the basis of the function vector $f : \mathcal{X} \to \mathcal{Z}$ with $z = f(x)$. In a maximization context, an objective vector $z \in \mathcal{Z}$ is dominated by an objective vector $z' \in \mathcal{Z}$, denoted by $z \prec z'$, iff $\forall m \in \{1, 2, \ldots, M\}$, $z_m \leqslant z'_m$ and $\exists m \in \{1, 2, \ldots, M\}$ such that $z_m < z'_m$. By extension, a solution $x \in \mathcal{X}$ is dominated by a solution $x' \in \mathcal{X}$, denoted by $x \prec x'$, iff $f(x) \prec f(x')$. A solution $x^\star \in \mathcal{X}$ is said to be *Pareto optimal* (or *efficient*, *non-dominated*), if there does not exist any other solution $x \in \mathcal{X}$ such that $x^\star \prec x$. The set of all Pareto optimal solutions is called the *Pareto set* (or the *efficient set*). Its mapping in the objective space is called the *Pareto front*. One of the most challenging task in multi-objective optimization is to identify a complete Pareto set of minimal size, *i.e.* one Pareto optimal solution for each point from the Pareto front.

However, in the combinatorial case, generating a complete Pareto set is often infeasible for two main reasons (Ehrgott, 2005): (i) the number of Pareto optimal solutions is typically exponential in the size of the problem instance and (ii) deciding if a feasible solution belongs to the Pareto set may be NP-complete. Therefore, the overall goal is often to identify a good *Pareto set approximation*. To this end, heuristics in general, and evolutionary algorithms in particular, have received a growing interest since the late eighties (Deb, 2001; Coello Coello et al., 2007).

2.2. Literature overview

A large body of literature exists concerning parallel multi-objective algorithms. Two interdependent issues are usually addressed: (i) how to decrease the computational complexity of a specific multi-objective algorithms and (ii) how to make parallel processes cooperate to improve the quality of the Pareto set approximation, see *e.g.*, Zhu and Leung (2002), Jozefowiez, Semet, and Talbi (2002), Deb, Zope, and Jain (2003), Coello Coello and Sierra (2004), Melab, Talbi, and Cahon (2006), Tan, Yang, and Goh (2006), Coello Coello et al. (2007), Mostaghim, Branke, and Schmeck (2007), Hiroyasu, Yoshii, and Miki (2007), Durillo, Nebro, Luna, and Alba (2008), Talbi et al. (2008), Figueira, Liefooghe, Talbi, and Wierzbicki (2010), Mostaghim (2010). Often, parallel and cooperative techniques implicitly come with the idea of *decomposing* the search into many sub-problems so that a diversified set of solutions, in terms of Pareto front quality, can be obtained. The main challenge is on defining efficient strategies to either divide the search space or the objective space.

For instance, the population induced by a particle swarm multi-objective algorithm is divided by Mostaghim et al. (2007) into sub-swarms which are then coordinated through a master–slave approach by injecting the so-called subswarm-guides in each sub-population. The diffusion model (Van Veldhuizen et al., 2003) and the island model (Tomassini, 2005) have also been extensively adopted to design distributed cooperative methods. In the so-called cone separation technique (Branke, Schmeck, Deb, & Reddy, 2004), the objective space is divided into regions distributed over some islands. Each island explores the same search space. When a solution is outside its corresponding objective space region, it is migrated to neighboring islands. This idea is refined by Streichert, Ulmer, and Zell (2005) using a clustering approach. Bui et al. (2009) propose a distributed framework where a number of adaptive spheres spanning the search space and controlled by an evolutionary algorithm is studied. In Zhu and Leung (2002), a