European Journal of Operational Research 200 (2010) 45–53

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

K-PPM: A new exact method to solve multi-objective combinatorial optimization problems

C. Dhaenens *, J. Lemesre, E.G. Talbi

Polytech'Lille/LIFL/USTL/INRIA, Parc Scientifique de la Haute Borne, 40, Avenue Halley, BâtA, Park Plaza, 59650 Villeneuve d'Ascq, France

ARTICLE INFO

Article history:
Received 26 October 2007
Accepted 24 December 2008
Available online 19 January 2009

Keywords:
Exact method
Multi-objective problem
Combinatorial optimization
Flow-shop problem

ABSTRACT

In this paper we propose an exact method able to solve multi-objective combinatorial optimization problems. This method is an extension, for any number of objectives, of the 2-Parallel Partitioning Method (2-PPM) we previously proposed. Like 2-PPM, this method is based on splitting of the search space into several areas, leading to elementary searches. The efficiency of the proposed method is evaluated using a multi-objective flow-shop problem.

1. Introduction

Multi-objective optimization problems (MOPs) are mostly NP-hard. Part of the difficulty comes from the existence of several optimal solutions (solutions that represent the best compromise between objectives). Hence, a lot of heuristic methods (in particular meta-heuristics) have been proposed to solve multi-objective combinatorial optimization problems.

In this paper, we are interested in solving MOPs exactly. Existing exact methods are mainly methods for tackling bi-objective problems (Two-phase method, ϵ-constraints approach, 2-PPM...). Only very few methods are able to find all efficient solutions for a problem with K objectives. The first one, a "Recursive algorithm for multi-objective combinatorial optimization problems with Q-criteria" [16,4] is an extension of the two-phases method [18]. The second one, an "Adaptive scheme to generate the Pareto front based on the ϵ-constraint method", is a generalization of the ϵ-constraint method [8].

In this paper, we present a new exact method able to solve multi-objective combinatorial optimization problems (with any number of objectives), named K-PPM [10]. This method is able to enumerate the whole set of Pareto solutions. In this paper, minimization problems are addressed, but the method may easily be adapted to maximization problems.

This article is organized as follows: Section 1 defines the main concepts of Multi-objective optimization. Section 2 exposes the existing bi-objective and multi-objective exact methods. Section 3 presents the extension of the 2-Parallel Partitioning Method (2-PPM) for any number of objectives. Section 4 presents a parallelization model of this method. Section 5 details the experiments. It presents the three-objective permutation flow-shop problem used as an illustration and discusses performance results of the different approaches. Finally, a conclusion and some perspectives of the work are given.

1.1. Multi-objective combinatorial optimization

In this part, we describe and define multi-objective optimization problems (MOPs) in the general case.

We assume that a solution x to such a problem can be described by a decision vector $(x_1, x_2, \ldots, x_n)$ in the decision space $\mathscr{X}$. This decision vector represents the values given to the variables of the problem. A cost function $f : \mathscr{X} \to \mathscr{Y}$ assigns to x an objective vector $(f_1(x), f_2(x), \ldots, f_K(x))$ in the objective space $\mathscr{Y}$. In this context, the multi-objective optimization problem consists in finding solutions in the decision space optimizing (minimizing or maximizing) K objectives (we note $\mathscr{K}$ for the set of the K considered objectives).

In multi-objective optimization problems, where an objective vector $(f_1, f_2, \ldots, f_K)$ has to be optimized, typically no single optimal solution exists but rather a set of solutions of the best compromise. These solutions form the Pareto front. They may be defined using the notion of dominance:

Definition 1. In a minimization problem, a solution x dominates a solution x' if and only if:

$$\begin{cases} \forall k \in \mathscr{K}, & f_k(x) \leqslant f_k(x'), \\ \exists k \in \mathscr{K}, & f_k(x) < f_k(x'). \end{cases}$$

* Corresponding author. Tel.: +33 3 59 57 78 80.
E-mail addresses: Clarisse.Dhaenens@lifl.fr (C. Dhaenens), El-Ghazali.Talbi@lifl.fr (E.G. Talbi).

0377-2217/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2008.12.034

In this context, the Pareto optimality definition is:

Definition 2. A solution is Pareto optimal if it is not dominated by any other solution of the feasible set. Such a solution is called 'efficient'.

Definition 3. A Pareto solution is called supported if it can be obtained by optimizing a linear combination of the objectives. Such a solution belongs to the convex hull of the Pareto front.

Let us introduce some notations useful for the rest of the article. Let $\mathscr{K}$ be the set of the K considered objectives.

Let Opt^K be the set of efficient solutions for a problem with K objectives, and Opt_λ^{K-l} be the set of efficient solutions for a problem with $K-l$ objectives, where λ represents the set of the $K-l$ objectives taken into consideration. Then let us define:

$$Opt^{K-l} = \bigcup_{\forall \lambda \subset \mathscr{K} | card(\lambda) = K-l} Opt_\lambda^{K-l},$$

the set of efficient solutions for all the sub-problems of size $K-l$ that can be created from the problem of size K, where the size of a problem is defined here by the number of objectives considered.

Particular points of the objective space will be used in the rest of the article and must be introduced here: the Ideal and the Nadir points. These points represent a lower and an upper bound of the Pareto front.

Definition 4. Ideal and Nadir points:

(1) Point $I = (f_1(I), \dots, f_K(I))$, with $f_k(I) = \min_{x \in \mathscr{X}} f_k(x), \forall k \in \mathscr{K}$, is called Ideal point.
(2) Point $N = (f_1(N), \dots, f_K(N))$, with $f_k(N) = \max_{x \in Opt^K} f_k(x)$, $\forall k \in \mathscr{K}$, is called Nadir point.

As we will see later (Section 3.1) the computation of the Nadir point is not straightforward when the number of objectives is greater than 2. This requires the calculation of Opt^K.

2. Existing exact methods

In this section, we present methods able to enumerate all the Pareto solutions of an MOP. First bi-objective methods are addressed and then two multi-objective methods (for any number of objectives) are described.

2.1. Bi-objective exact methods

There are several exact methods to solve bi-objective combinatorial optimization problems. A lot of these methods are linked to a particular problem, some others do not find the whole Pareto set (e.g., aggregation based methods). Nevertheless, some general methods able to find all the Pareto solutions for any combinatorial optimization problem exist. They are briefly presented here. To find more details on bi-objective exact methods, the reader may refer to [17].

2.1.1. The two-phases method

This method, proposed by Ulungu and Teghem [18], determines all the efficient solutions in two steps. The first phase consists in finding all supported solutions with the optimization of aggregations in the form $\lambda_1 f_1 + \lambda_2 f_2$. It starts by determining the extreme points of the front. Then supported solutions are found by searching recursively between two given supported solutions. Following this, the second phase consists in exploring all the triangles, underlying each pair of adjacent supported solutions, in order to find all the non-supported solutions. This method has been applied efficiently on the bi-objective assignment problem.

2.1.2. ϵ-constraint

The ϵ-constraint method is an application of the ϵ-constraint concept (introduced in [7]) to enumerate Pareto solutions. This method involves a constraint on one objective and optimizes the second objective. The constrained problem may be expressed by: $\min\{f_1(x) : x \in \mathscr{X},$ such that $f_2(x) \leqslant \epsilon\}$. The complete scheme is as follows: first, one extreme is computed, for example x_1 the extreme with the best value on the objective f_1. This solution determines a bound on objective f_2, and the best solution regarding f_1 has to be searched for below this bound on f_2. This operation is repeated until no new solution is found. With this scheme, the whole efficient set is found.

2.1.3. 2-PPM

We previously proposed the Parallel Partitioning Method for bi-objective problems in [10]. This method determines the whole Pareto front in three stages. In the first stage, the Ideal and the Nadir points are computed in order to limit the search space. These solutions may be found thanks to the extreme solutions in the bi-objective case. In the second stage, well distributed efficient solutions are searched for in order to divide the search space (during this step, supported as well as non-supported solutions are found). The third stage consists in finding the other efficient solutions by reducing the search space using solutions found during the second stage. This method has been successfully applied on a bi-objective flow-shop problem [10].

2.2. Multi-objective exact methods

All the methods described in the last section can only be applied to bi-objective optimization problems. In this section, two existing multi-objective exact methods are presented.

2.2.1. A recursive algorithm for multi-objective combinatorial optimization problems with K objectives

This method was initially proposed by Tenfelde-Podehl [16] and was tested on the quadratic assignment problem. It is a generalization of the two-phases method for more than two objectives.

This algorithm starts by finding all the solutions belonging to the set Opt^{K-1} in order to calculate the Nadir point of the K-objective problem. Then sub-spaces are determined with these solutions. For every sub-space, an axis of research is computed, and a single objective problem is solved. When a new solution is found, the search space is split and some new searches are launched. This method stops once all the search spaces have been examined and no new solution is found. With this method, the search space is split in order to visit only the pertinent sub-spaces and a single objective search is required for each Pareto solution. Therefore, if for a given problem, single objective searches are time-consuming, a lot of time is required to solve instances of this problem.

2.2.2. An adaptive scheme to generate the Pareto front based on the ϵ-constraint method

This method, proposed by Laumans et al. [8], is a generalization of the ϵ-constraint method.

To find Pareto optimal solutions, some lower and upper bounds are given to single objective searches. The first search is made on the whole search space, then each solution found is used to bound sub-spaces.

The core of the algorithm is a $K-1$ dimensional hypergrid, which splits the objective space into rectangles parallel to the axis $f_2, \dots, f_K$. When a new Pareto solution is found, the grid is split. If no new solution is found, the next rectangle of the grid is visited.