Computers & Operations Research 49 (2014) 97–106

Contents lists available at ScienceDirect

Computers & Operations Research

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

An efficient procedure for finding best compromise solutions to the multi-objective assignment problem

Lyes Belhoul, Lucie Galand, Daniel Vanderpooten *

PSL, Université Paris-Dauphine, LAMSADE, Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16, France

ARTICLE INFO

Available online 29 March 2014

Keywords:
Multi-objective assignment problem
Compromise solutions
Branch and bound
k-best algorithm

ABSTRACT

In this paper, we consider the problem of determining a best compromise solution for the multi-objective assignment problem. Such a solution minimizes a scalarizing function, such as the weighted Tchebychev norm or reference point achievement functions. To solve this problem, we resort to a ranking (or k-best) algorithm which enumerates feasible solutions according to an appropriate weighted sum until a condition, ensuring that an optimal solution has been found, is met. The ranking algorithm is based on a branch and bound scheme. We study how to implement efficiently this procedure by considering different algorithmic variants within the procedure: choice of the weighted sum, branching and bounding schemes. We present an experimental analysis that enables us to point out the best variants, and we provide experimental results showing the remarkable efficiency of the procedure, even for large size instances.

1. Introduction

Multi-objective combinatorial optimization deals with the optimization of multiple conflicting objective functions in a combinatorial optimization problem. When several objective functions are taken into account, there generally does not exist a unique optimal solution. The concept of optimality is indeed based on the Pareto dominance that induces a partial order on the solutions. The *Pareto optimal*, or *efficient*, solutions are the feasible solutions such that one cannot improve the performance of one objective without worsening at least another objective. As they represent all possible compromises between the objectives one can obtain, any rational decision maker (DM) would thus prefer an efficient solution. Nevertheless when the number of efficient solutions is large, selecting his/her most preferred solution among all the efficient ones can be difficult for a DM. Scalarizing functions are used to discriminate among the efficient solutions by aggregating the objective functions according to the preferences of the DM (see, e.g., [14,28]). An optimal solution according to such a scalarizing function represents a *best compromise* with respect to the DM's preferences. Unfortunately, the most relevant scalarizing functions are generally nonlinear, which makes their optimization over combinatorial domains hard. A best compromise solution could be determined by first generating all the efficient solutions, and then selecting an optimal solution among them with respect to the scalarizing function to be optimized. Nevertheless the potentially huge number of efficient solutions often makes the first step intractable over combinatorial domains [5]. Therefore, it seems more relevant to directly focus the search on a best compromise solution, without generating a priori all the efficient solutions. In particular, when resorting to an interactive exploration of the efficient solutions [10,26], each iteration of such a procedure precisely consists in generating an optimal solution with respect to a (nonlinear) scalarizing function.

The problem of determining a best compromise solution to a multi-objective combinatorial optimization problem has been studied in the literature for several multi-objective combinatorial problems (e.g., multi-objective path problem, multi-objective spanning tree problem, etc.) and several scalarizing functions (e.g., Euclidean distance [17], Max operator [7,12], Tchebychev norm [8], Choquet integral [9], etc.). A *ranking method* is often proposed in these works so as to determine a best compromise solution. It is based on the enumeration of the solutions in non-decreasing order of a linear aggregation of their objective functions until a best compromise solution is proved to be already enumerated. Depending on the aggregation function to be optimized, some conditions on the linear aggregation function used to enumerate the solutions are required to make the ranking method valid. Moreover an efficient algorithm that enumerates the solutions in non-decreasing order of their value (*k-best algorithm*) is also required to make the ranking method efficient.

In this paper, we consider the problem of directly determining a best compromise solution for the multi-objective assignment problem. The assignment problem is a standard combinatorial

* Corresponding author.
E-mail addresses: lyes.belhoul@dauphine.fr (L. Belhoul), lucie.galand@dauphine.fr (L. Galand), daniel.vanderpooten@dauphine.fr (D. Vanderpooten).

http://dx.doi.org/10.1016/j.cor.2014.03.016
0305-0548/© 2014 Elsevier Ltd. All rights reserved.

optimization problem [2], modeling a variety of situations where resources have to be assigned to tasks. In many contexts, several conflicting objectives, reflecting for example the cost and various measures of quality or performance of an assignment, are to be taken into account. Several procedures have been proposed to determine the set of all efficient solutions for multi-objective assignment problem (see [20,22] for the bi-objective case and [23] for the tri-objective case). However, to the best of our knowledge, no procedure has been proposed specifically for the problem of finding efficiently a best compromise assignment. This is the purpose of this work, where the notion of compromise among objectives is defined through a distance to a reference point in the objective space with respect to a weighted norm, such as the Tchebychev norm [1] or, more generally, using an achievement scalarizing function [27,28]. The problem is NP-hard since it contains, as a special case, the min max regret assignment problem which is known to be NP-hard [13]. Following previous works in the literature on other problems and aggregation functions, we apply the ranking method to the problem of determining a best compromise assignment. Several k-best algorithms have been proposed in the literature for the single objective assignment problem [3,15,16,18,19]. All these algorithms are based on a branch and bound procedure. In this paper, we propose a branch and bound procedure that implements the ranking method and we investigate different algorithmic issues within our branch and bound procedure: the choice of the linear aggregation function, the branching scheme and the bounding phase. In particular, we show that the choice of a relevant linear aggregation function, that limits the number of solutions to be enumerated within the ranking method, can be performed efficiently by solving a continuous linear program. Moreover, we point out the most convenient k-best algorithm, among the different algorithms of the literature. Furthermore, we propose some pruning techniques, that are specific to the assignment problem, and we study their impact on the efficiency of the procedure. The experimental analysis of all these algorithmic variants leads us to provide an efficient procedure, that can handle very quickly small and medium size instances, as well as large size instances.

In Section 2, the problem of determining a best compromise assignment and the basic definitions are presented. The generic ranking method, that does not depend on the problem, is presented in Section 3. Our branch and bound procedure that implements the ranking method for best compromise search in multi-objective assignment problem is proposed in Section 4. Experimental analysis of our procedure is performed in Section 5. Section 6 concludes this work.

2. Basic definitions and concepts

2.1. Multi-objective assignment problem

The multi-objective assignment problem can be formulated as follows:

$$\begin{aligned} \min z_k(x) &= \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}^k x_{ij}, \quad k=1,\dots,p \\ \text{s.t.} \sum_{j=1}^{n} x_{ij} &= 1, \quad i=1,\dots,n \\ \sum_{i=1}^{n} x_{ij} &= 1, \quad j=1,\dots,n \\ x_{ij} &\in \{0,1\}, \quad i=1,\dots,n, \; j=1,\dots,n \end{aligned} \tag{MOAP}$$

where n is the number of tasks (and agents), p is the number of objectives, and c_{ij}^k is the non-negative cost of assigning task i to agent j with respect to objective k. Decision variable x_{ij} equals 1 if task i is assigned to agent j and 0 otherwise. The multi-objective assignment problem can also be stated as a *multi-objective perfect matching problem* on a complete bipartite graph $G=(U\cup V,E)$ with $|U|=|V|=n$ such that any edge $(i,j)\in E$ links a vertex i in U to a vertex j in V with cost c_{ij}^k for any objective k. A perfect matching in G is equivalent to an assignment in (MOAP). Let $X\subset\{0,1\}^{n^2}$ denote the set of feasible solutions to Problem (MOAP). To any solution $x\in X$ is associated a set $E(x)$ of n edges such that an edge $(i,j)\in E$ of bipartite graph G is in $E(x)$ if and only if $x_{ij}=1$. A solution is therefore equivalently characterized by x or by $E(x)$. Finally, to any solution $x\in X$ is associated a criterion vector $z(x)=(z_1(x),\dots,z_p(x))$ which corresponds to its image in the objective space. The image set in the objective space of all the feasible solutions is denoted by $Z=\{z(x): x\in X\}$.

2.2. Non dominance and efficiency

Due to the conflicting nature of the different objectives, there is generally no feasible solution that minimizes simultaneously all the objectives. The concept of optimality is therefore based on the Pareto dominance relation on points in $\mathbb{R}^p$ which induces a partial order on solutions of X. A point $z\in\mathbb{R}^p$ *dominates* another point $z'\in\mathbb{R}^p$ if $z_k\le z'_k$ for all k in $\{1,\dots,p\}$ and $z\ne z'$. A point $z\in\mathbb{R}^p$ *strictly dominates* another point $z'\in\mathbb{R}^p$ if $z_k<z'_k$ for all k in $\{1,\dots,p\}$. A point z in Z is (*weakly*) *non-dominated* if there is no other point z' in Z such that z' (strictly) dominates z. Let Z_N (resp. Z_{WN}) denote the set of non-dominated (resp. weakly non-dominated) points. Among the points in Z_N, those which lie on $conv(Z)+\mathbb{R}^p_{\geq}$ (where $conv(Z)$ is the convex hull of Z) or, equivalently, which minimize, for at least one strictly positive weighting vector, a weighted sum of the objectives, are called *supported*. These notions can be transposed directly in the decision space and lead to the corresponding concepts of *weakly efficient*, *efficient* and *supported solutions*.

2.3. Determination of best compromise solutions

In order to determine a solution which achieves the best possible compromise among the objectives according to the preferences of a specific decision maker (DM), it is necessary to resort to a scalarizing function f_γ which aggregates these objectives taking into account the DM's preferences formalized through a parameter γ. The corresponding best compromise solution is then obtained by solving the following problem:

$$\min_{x\in X} f_\gamma(z_1(x),\dots,z_p(x)) \tag{Cγ}$$

The choice of a scalarizing function should be made considering the following requirements (see [6,28]):

- $R1$: Any optimal solution of (C_γ) corresponds to a non-dominated point,
- $R2$: Any non-dominated point can be reached by solving (C_γ),
- $R3$: Solving (C_γ) is reasonably easy.

More precisely requirement R1 imposes that solving Problem (C_γ), for any parameter γ, returns an efficient solution. Requirement R2 imposes that, for any non-dominated point, there exists at least one parameter γ such that Problem (C_γ) returns, preferably as a *unique* optimal solution, a solution corresponding to this point. Finally, for any combinatorial optimization problem like the assignment problem, requirement R3 can be interpreted as 'solving (C_γ) should ideally have the same complexity as solving the corresponding single objective problem', which means in our case that Problem (C_γ) should be a polynomial problem.