European Journal of Operational Research 229 (2013) 59–66

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Competitive strategies for an online generalized assignment problem with a service consecution constraint

Feifeng Zheng [a], Yongxi Cheng [b,*], Yinfeng Xu [b,c], Ming Liu [d]

[a] Glorious Sun School of Business and Management, Donghua University, Shanghai 200051, China
[b] School of Management, Xi'an Jiaotong University, Xi'an 710049, China
[c] State Key Lab for Manufacturing Systems Engineering, Xi'an 710049, China
[d] School of Economics & Management, Tongji University, Shanghai 200092, China

ARTICLE INFO

Article history:
Received 24 June 2012
Accepted 2 February 2013
Available online 13 February 2013

Keywords:
Assignment
Online strategy
Service consecution constraint
Competitive ratio
Lower bound

ABSTRACT

This work studies a variant of the *online generalized assignment problem*, where there are $m \geqslant 2$ heterogeneous servers to process n requests which arrive one by one over time. Each request must either be assigned to one of the servers or be rejected upon its arrival, before knowing any information of future requests. There is a corresponding weight (or revenue) for assigning each request to a server, and the objective is to maximize the total weights obtained from all the requests. We study the above problem with a *service consecution constraint*, such that at any time each server is only allowed to process up to d consecutive requests.

We investigate both deterministic and randomized online strategies for this problem. When the ratio ρ between the largest and smallest possible weights obtained from assigning a request to a server is known in advance, we present an optimal deterministic online strategy with competitive ratio $\rho^{\frac{1}{d}}$. For randomized strategies, we first prove a lower bound on the competitive ratio, then we present a randomized strategy with competitive ratio less than 2, which does not need to know the value of ρ or d. Computational tests show that our proposed strategies have very good practical performance.

1. Introduction

The *assignment problem* (AP) is a well-known optimization problem due to its extensive applications (Pentico, 2007). Roughly speaking, there are n requests (or tasks) to be assigned to m servers (or agents). For each request, there is a subset of servers that are available to process it, and the request can only be assigned to one of these available servers (in certain scenario a request is allowed to be rejected, that is not assigned to any server). Each assignment pair formed by a request and a server processing the request has a specific weight. Depending on different applications, the weights of assignments represent either revenue or cost, and the objective is to maximize or minimize the assignment weight, that is the total weight obtained from all the requests.

In the classical assignment problem, which aims to optimize the total weight, the total number of servers m is equal to the total number of requests n, and each request shall be assigned to some server and each server processes only one request. Since Kuhn (1955) proposed the famous Hungarian method for the classical assignment problem, there have been many variations of the problem proposed in the literature, such as the *bottleneck assignment problem* to minimize the maximum weight obtained from each request (Ravindran and Ramaswami, 1977; Aneja and Punnen, 1999) and the *balanced assignment problem* to minimize the difference between the maximum and minimum weight obtained from each request (Martello et al., 1984). The reader is referred to Pentico (2007) for a survey on more variations of the classical assignment problem.

The most general version of the assignment problem that allows each server to process multiple requests, is the *generalized assignment problem* (GAP). The generalized assignment problem has wide applications including routing (Fisher and Jaikumar, 1981), facility location (Ross and Soland, 1977), loading for flexible manufacturing systems (Mazzola et al., 1989), allocating cross-trained workers to multiple departments (Campbell and Diaby, 2002), etc. Numerous variations of the generalized assignment problem have been studied by, among others, Martello and Toth (1995), Arora and Puri (1998), Chang and Ho (1998), Moccia et al. (2009), etc. The reader is referred to Cattrysse and Van Wassenhove (1992), and Oncan (2007) for surveys on the applications of and algorithms for the generalized assignment problem.

1.1. The online assignment problem and competitive analysis

In the last decades, the *online* version of the classical assignment problem has caught interest due to real applications. In the

* Corresponding author. Tel.: +86 29 82668382.
E-mail addresses: ffzheng@dhu.edu.cn (F. Zheng), chengyx@mail.xjtu.edu.cn (Y. Cheng), yfxu@mail.xjtu.edu.cn (Y. Xu), minyivg@gmail.com (M. Liu).

0377-2217/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ejor.2013.02.004

online assignment problem where the total number of requests n is equal to the total number of servers m, requests arrive one by one over time and each request must be assigned to one of the available servers upon its arrival, that is the decision of assigning the currently arrived request to which server must be made before knowing any information of future requests. At the end each server processes exactly one request. A strategy for solving the above online assignment problem is called an *online assignment strategy*. This online version of the classical assignment problem is also known as the *online bipartite matching problem*, where the request set and the server set are viewed as the two disjoint subsets of vertices of a bipartite graph G, and an assignment pair formed by a request and a server processing the request is viewed as an edge in G.

The performance of an online assignment strategy is measured by the *competitive ratio* (Borodin and El-Yaniv, 1998), which is a widely used measure for the performance of online algorithms. Consider an online assignment problem to maximize the total weight as in this paper (the problem to minimize the total weight can be similarly discussed). For any input request sequence σ, let $|\mathcal{A}(\sigma)|$ and $|O(\sigma)|$ be the total assignment weights obtained by a deterministic online strategy $\mathcal{A}$ and by an optimal offline strategy OPT, respectively. Then, define

$$\gamma = \sup_{\sigma} \frac{|O(\sigma)|}{|\mathcal{A}(\sigma)|}.$$

Clearly, for the maximization problem we have $\gamma \geqslant 1$. If γ is finite, then strategy $\mathcal{A}$ is said to be γ-competitive, and γ is called the competitive ratio of $\mathcal{A}$. By this measurement, an online strategy for a maximization problem with smaller competitive ratio has better performance.

In this paper, for a randomized online strategy $\mathcal{B}$, the competitive ratio of $\mathcal{B}$ is measured with respect to an oblivious adversary (Raghavan and Snir, 1994; Ben-David et al., 1994), which is standard in the analysis of randomized online algorithms. Different from adaptive adversaries, an oblivious adversary must generate a complete request sequence in advance, without knowing the outcome of the random coin tosses made by $\mathcal{B}$ (or the specific actions taken by $\mathcal{B}$ as a result of the coin tosses) on the requests. However, the adversary does know the complete description of the online strategy $\mathcal{B}$, and knows the probability distribution of actions taken by $\mathcal{B}$ for a given input request sequence. For an input request sequence σ, the total assignment weight $|\mathcal{B}(\sigma)|$ obtained by a randomized online strategy $\mathcal{B}$ from σ is a random variable. The competitive ratio of $\mathcal{B}$ is then defined as the ratio between the total weight $|O(\sigma)|$ obtained by an optimal (deterministic) offline strategy OPT, and the expected total weight $E(|\mathcal{B}(\sigma)|)$ obtained by $\mathcal{B}$, on an input request sequence σ in the worst case. That is, define

$$\gamma_r = \sup_{\sigma} \frac{|O(\sigma)|}{E(|\mathcal{B}(\sigma)|)}.$$

If γ_r is finite then the randomized strategy $\mathcal{B}$ is said to be γ_r-competitive, and γ_r is called the competitive ratio of $\mathcal{B}$.

Karp et al. (1990) considered an online unweighted bipartite matching problem, where for each request there is a subset of available servers and the objective is to maximize the total number of satisfied requests. They showed that a simple greedy strategy that, for each request arbitrarily selects an available server for it, if any, is optimally 2-competitive. Moreover, they presented an $\left(\frac{e}{e-1} + o(1)\right)$-competitive randomized strategy against an oblivious adversary, where e is the base of the natural logarithm. They also proved that the competitive ratio $\frac{e}{e-1} + o(1)$ is best possible for any randomized online strategy against an oblivious adversary, up to lower order terms.

For the online weighted bipartite matching problem in non-metric space, neither the maximization nor the minimization problem has deterministic strategies with bounded competitive ratio (Kalyanasundaram and Pruhs, 1993). For the online maximum weighted bipartite matching problem in metric space, Kalyanasundaram and Pruhs (1993) proved that a simple greedy strategy that always selects the available server with the largest weight to process the currently arrived request, reaches the optimal competitive ratio of 3. For the online minimum weighted bipartite matching problem in metric space, both Kalyanasundaram and Pruhs (1993), and Khuller et al. (1994) gave $(2n-1)$-competitive deterministic algorithms, where n is the number of requests (and is also the number of servers), and showed that no better deterministic algorithm is possible even for the star graph. Meyerson et al. (2006) gave an $O(\log^3 n)$-competitive randomized algorithm for the online minimum weighted bipartite matching problem in metric space, which is the first poly-logarithmic competitive online algorithm for this problem. Improved randomized algorithms are proposed later by Csaba and Pluhar (2008) with competitive ratio $O(\log^3 n/\log\log n)$, and by Bansal et al. (2007) with competitive ratio $O(\log^2 n)$. The competitive ratios of all the above mentioned randomized online algorithms are measured under the oblivious adversary model.

1.2. Our contribution

In this paper we investigate a variant of the online generalized assignment problem with a *service consecution constraint*, which is specified by an integer parameter $d \geqslant 1$, such that at any time each server is only allowed to process at most $d \geqslant 1$ consecutive requests. We investigate both deterministic and randomized online strategies for this problem.

The *online generalized assignment problem with service consecution constraint* studied in this paper is formally described as follows. We have $m \geqslant 2$ heterogeneous servers $s_1, s_2, \ldots, s_m$, and n requests that arrive over time one by one in the ordering $r_1, r_2, \ldots, r_n$, where m is known while n is unknown in advance. For $1 \leqslant i \leqslant n$, each request r_i is associated with an m-dimensional positive weight vector $W_i = (w_{i,1}, \ldots, w_{i,m})$, where $w_{i,j} > 0$ is the weight obtained if request r_i is assigned to server s_j, for $1 \leqslant j \leqslant m$. If r_i is rejected, that is not assigned to any server, then no weight is obtained from r_i. The decision of rejecting r_i or assigning r_i to which server must be made upon the arrival of r_i, without knowing any information of future requests. The assignment is required to satisfy the service consecution constraint, that is at any time it is only allowed to assign up to d consecutive requests to any server, where $d \geqslant 1$ is given. The objective is to maximize the total weight obtained from all the requests.

We assume that $w_{i,j} \in [M_1, M_2]$ $(0 < M_1 \leqslant M_2)$ for $1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m$. For convenience, we normalize the weight interval $[M_1, M_2]$ to $[1, \rho]$ where $\rho = M_2/M_1 \geqslant 1$. We use $OGAP|d \geqslant 1$ to denote the above online generalized assignment problem with service consecution constraint specified by parameter $d \geqslant 1$. For the case where d is fixed to be some constant d_0, we denote the problem by $OGAP|d = d_0$.

Problems in operations research with various consecution constraints have been studied in the literature. In parallel machine scheduling where the activity of machine maintenance is a necessary requirement, to ensure that each machine is available during job processing, an upper limit on the maximum consecutive working time between two adjacent maintenance activities for any machine is required (Xu et al., 2008; Sun and Li, 2010). Another example is the traveling tournament problem (TTP) motivated by scheduling Major League Baseball (MLB) in North America (Easton et al., 2001), where each team plays games in a home/away pattern, and the problem is to minimize the total travel distance of the teams, under the constraint that the maximum number of consecutive home games as well as consecutive away games for each