European Journal of Operational Research 228 (2013) 46–55

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

The generalized assignment problem with minimum quantities

Sven O. Krumke, Clemens Thielen *

University of Kaiserslautern, Department of Mathematics, Paul-Ehrlich-Str. 14, D-67663 Kaiserslautern, Germany

ARTICLE INFO

Article history:
Received 25 September 2012
Accepted 16 January 2013
Available online 30 January 2013

Keywords:
Assignment problems
Combinatorial optimization
Approximation algorithms
Computational complexity

ABSTRACT

We consider a variant of the generalized assignment problem (GAP) where the amount of space used in each bin is restricted to be either zero (if the bin is not opened) or above a given lower bound (a *minimum quantity*). We provide several complexity results for different versions of the problem and give polynomial time exact algorithms and approximation algorithms for restricted cases. For the most general version of the problem, we show that it does not admit a polynomial time approximation algorithm (unless P = NP), even for the case of a single bin. This motivates to study *dual* approximation algorithms that compute solutions violating the bin capacities and minimum quantities by a constant factor. When the number of bins is fixed and the minimum quantity of each bin is at least a factor $\delta > 1$ larger than the largest size of an item in the bin, we show how to obtain a polynomial time dual approximation algorithm that computes a solution violating the minimum quantities and bin capacities by at most a factor $1 - \frac{1}{\delta}$ and $1 + \frac{1}{\delta}$, respectively, and whose profit is at least as large as the profit of the best solution that satisfies the minimum quantities and bin capacities strictly. In particular, for $\delta = 2$, we obtain a polynomial time (1,2)-approximation algorithm.

1. Introduction

The generalized assignment problem (cf., for example, [1,2]) is a classical generalization of both the (multiple) knapsack problem and the bin packing problem. In the classical version of GAP, one is given m bins, a capacity B_j for each bin j, and n items such that each item i has size $s_{i,j}$ and yields profit $p_{i,j}$ when packed into bin j. The goal is to find a feasible packing of the items into the bins that maximizes the total profit. Applications of GAP include fixed charge location problems, grouping and loading for flexible manufacturing systems, vehicle routing, scheduling projects, allocating storage space, scheduling payments on accounts, designing communication networks, assigning software development tasks to programmers, assigning jobs to computers in networks, scheduling variable length TV commercials, and assigning ships to overhaul facilities. For details on these applications, we refer to [2] and the references therein.

In this paper, we consider a variation of the problem where the amount of space used in each bin is restricted to be either zero (if the bin is not opened) or above a given lower bound (a *minimum quantity*). This additional restriction is motivated from many practical packing problems where it does often not make sense to open an additional container (bin) if not at least a certain amount of space in it will be used. For example, in the classical application of GAP where the bins correspond to workers (or factories) and the items correspond to jobs that can be assigned to them, it may be unreasonable to hire an additional worker (open an additional factory) if not at least a certain amount of work will be assigned to the worker (the factory). This was pointed out, e.g., in [3].

Another motivation for adding minimum quantities to GAP is its application in unrelated machine scheduling [4]. Here, the bins correspond to machines and the items to jobs. In practice, it may be that running a machine is only feasible if a prescribed minimum load is attained on the machine. In a foundry or steel works, for example, a machine can only be used if it has a certain minimum amount of metal to process. Similar restrictions apply when scheduling electric power systems, where power plants must often be run at or above a non-zero minimum loading level [5].

Formally, the generalized assignment problem with minimum quantities (GAP-MQ) is defined as follows:

Definition 1 (*GAP with minimum quantities (GAP-MQ)*).

INSTANCE:	m bins with capacities $B_1, \dots, B_m \in \mathbb{N}$ and minimum quantities $q_1, \dots, q_m \in \mathbb{N}$ (where $q_j \leqslant B_j$ for all $j = 1, \dots, m$), and n items. For each item i and bin j, a size $s_{i,j} \in \mathbb{N}$ and a profit $p_{i,j} \in \mathbb{N}$
TASK:	Find a packing of a subset of the items into the bins such that the total space used in each bin j is either zero (if bin j is not opened) or at least q_j and at most B_j and the total profit is maximized

In the decision version of the problem, a bound $P \in \mathbb{N}$ on the total profit is given and the question is whether there exists a feasible packing with total profit at least P. When stating results about the

* Corresponding author. Tel.: +49 (631) 205 4590; fax: +49 (631) 205 4737.
E-mail addresses: krumke@mathematik.uni-kl.de (S.O. Krumke), thielen@mathematik.uni-kl.de (C. Thielen).

0377-2217/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ejor.2013.01.027

computational complexity of an optimization problem such as GAP-MQ, we will always mean the complexity of the corresponding decision problem.

Note that, in the above definition and throughout the paper, we always assume $\mathbb{N}$ to contain zero and denote the positive integers by $\mathbb{N}_+$. Moreover, we always assume that $q_j \leqslant \sum_{i=1}^{n} s_{i,j}$ for each bin j in an instance of GAP-MQ (otherwise, there does not exist any feasible solution that packs a nonempty set of items into bin j, so the bin can be removed from the instance).

Two important special cases of GAP-MQ motivated from the problem of assigning students to seminars at a university are the *seminar assignment problem* (SAP) and the *participant maximization problem* (PMP). SAP is the special case of GAP-MQ in which all item sizes are one. It can be motivated from the problem of assigning students (items) to seminars (bins) such that the number of participants in each seminar j is between q_j and B_j and the total satisfaction (profit) of the students is maximized. PMP is the special case of SAP in which all profits are one, so the objective is simply to maximize the number of students that receive a place in one of the seminars (the total number of participants). In these applications, the minimum quantities are a very natural restriction since holding a seminar only make sense if there is at least a certain minimum number of students giving a talk in the seminar.

By a *polynomial time α-approximation algorithm* for a maximization problem such as GAP-MQ, we mean an algorithm that, for any given instance I of encoding length $|I| \in \mathbb{N}_+$, finds a feasible solution with objective value at least $\frac{1}{\alpha}$ times optimal in time bounded by a polynomial in $|I|$ if the instance I admits a feasible solution, and outputs infeasibility of the instance after a number of steps bounded by a polynomial in $|I|$ if no feasible solution for instance I exists.

For problems which do not admit polynomial time approximation algorithms, a common approach is to study *dual* approximation algorithms that compute solutions violating some constraints of the problem by at most a constant factor. By a *polynomial time (α, β)-approximation algorithm* for a maximization problem, we mean an algorithm that, for any given instance I of encoding length $|I| \in \mathbb{N}_+$, achieves the following: If the instance I admits a feasible solution, the algorithm requires only time bounded by a polynomial in $|I|$ to find a solution that violates the constraints of the problem by at most a factor β and whose objective value is at least $\frac{1}{\alpha}$ times as large as the objective value of the optimal solution that satisfies the constraints strictly. If the instance I does not admit a feasible solution, the algorithm outputs infeasibility of the instance after a number of steps bounded by a polynomial in $|I|$.

1.1. Previous work

The classical GAP is well-studied in literature. A comprehensive introduction to the problem can be found in [1]. A survey of algorithms for GAP is given in [2]. For a survey on different variants of assignment problems studied in literature, we refer to [6].

GAP is known to be APX-hard [7], but there exists a 2-approximation algorithm [4,7]. Cohen et al. [8] showed how any polynomial time α-approximation algorithm for the knapsack problem can be translated into a polynomial time $(1 + \alpha)$-approximation algorithm for GAP. A $(1, 2)$-approximation algorithm for the equivalent minimization version of GAP, assigning item i to bin j causes a cost $c_{i,j}$, was provided by Shmoys and Tardos [4]: for every feasible instance of GAP, their algorithm computes a solution that violates the bin capacities by at most a factor of 2 and whose cost is at most as large as the cost of the best solution that satisfies the bin capacities strictly.

GAP is a generalization of both the (multiple) knapsack problem (cf. [1,7,9]) and the bin packing problem (cf. [10–12]). The multiple knapsack problem is the special case of GAP where the size and profit of an item are independent of the bin (knapsack) it is packed into. The bin packing problem can be seen as the special case of the decision version of GAP in which all bins have the same capacity and all profits are one. The question of deciding whether a packing of total profit equal to the number of items exists is then equivalent to asking whether all items can be packed into the given number of bins.

A dual version of bin packing (often called *bin covering*) in which minimum quantities are involved was introduced in [13,14]. Here, the problem is to pack a given set of items with sizes that do not depend on the bins so as to maximize the number of bins used, subject to the constraint that each bin contains items of total size at least a given threshold T (upper bin capacities are not considered due to the nature of the objective function). Hence, the bin covering problem can be seen as a variant of GAP-MQ in which the minimum quantity is the same for each bin and the objective is to maximize the number of bins used. Since any approximation algorithm with approximation ratio strictly smaller than 2 would have to solve the NP-complete partition problem when applied to instances in which the sizes of the items sum up to two, it follows that (unless P = NP) no polynomial time $(2 - \epsilon)$-approximation for bin covering exists for any $\epsilon > 0$. In contrast, the main result of Assmann et al. [14] is an $\mathcal{O}(n \log^2 n)$ time algorithm that yields an *asymptotic* approximation ratio of 4/3 for bin covering, while easier algorithms based on next fit and first fit decreasing are shown to yield asymptotic approximation ratios of 2 and 3/2, respectively. Later, an asymptotic PTAS [15] and an asymptotic FPTAS [16] for bin covering were developed.

Minimum quantities have recently been studied for minimum cost network flow problems [3,17,18]. In this setting, minimum quantities for the flow on each arc are considered, which results in the minimum cost flow problem becoming strongly NP-complete [17]. Moreover, it was shown in [17] that (unless P = NP) no polynomial time $g(|I|)$-approximation for the problem exists for any polynomially computable function $g : \mathbb{N}_+ \rightarrow \mathbb{N}_+$, where $|I|$ denotes the encoding length of the given instance.

1.2. Our contribution

We prove several complexity and approximation results on GAP-MQ and its special cases (see Table 1 for an overview).

We show that PMP is weakly NP-complete and admits a fully polynomial time approximation scheme (FPTAS). In contrast, we prove that SAP is strongly NP-complete and (unless P = NP) does not admit a polynomial time approximation scheme (PTAS) even if all profits are in $\{0, 1\}$. We show, however, that SAP can be solved in polynomial time by linear programming (or, more efficiently, by minimum cost flow computations) when the number of seminars (bins) is fixed. For the general case of GAP-MQ (which, by our results on SAP, is strongly NP-complete even if all item sizes are one), we show that the problem also remains strongly NP-complete if all profits are one or if the profit obtained from packing an item into any bin equals the size of the item. Both results hold even for the case that the size of an item is independent of the bin it is packed into. Moreover, we prove that (unless P = NP) no polynomial time approximation algorithm exists for GAP-MQ even for these restricted profit values and only one bin. We show, however, that GAP-MQ can be solved optimally in polynomial time when the profit of an item is independent of the bin it is packed into and the maximum bin capacity $B_{\max}$ as well as the number of different item types $(s_1, \ldots, s_m, p)$ are fixed.

For the case that the number of bins is fixed, we present a pseudo-polynomial time dynamic programming algorithm for GAP-MQ. More importantly, when the number of bins is fixed and $q_j \geqslant \delta s_{i,j}$ for all i, j and some $\delta > 1$, we show how to obtain a polynomial time dual approximation algorithm that computes a solution violating