European Journal of Operational Research 218 (2012) 377–381

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

A strongly polynomial FPTAS for the symmetric quadratic knapsack problem

Zhou Xu *

Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

ARTICLE INFO

Article history:
Received 21 December 2010
Accepted 31 October 2011
Available online 15 November 2011

Keywords:
Combinatorial optimization
Quadratic knapsack
FPTAS
Strongly polynomial time
Machine scheduling

ABSTRACT

The symmetric quadratic knapsack problem (SQKP), which has several applications in machine scheduling, is NP-hard. An approximation scheme for this problem is known to achieve an approximation ratio of $(1+\epsilon)$ for any $\epsilon > 0$. To ensure a polynomial time complexity, this approximation scheme needs an input of a lower bound and an upper bound on the optimal objective value, and requires the ratio of the bounds to be bounded by a polynomial in the size of the problem instance. However, such bounds are not mentioned in any previous literature. In this paper, we present the first such bounds and develop a polynomial time algorithm to compute them. The bounds are applied, so that we have obtained for problem (SQKP) a fully polynomial time approximation scheme (FPTAS) that is also strongly polynomial time, in the sense that the running time is bounded by a polynomial only in the number of integers in the problem instance.

1. Introduction

We consider a quadratic knapsack problem of the following structure:

$$\min \quad Z(x) = \sum_{1\leqslant i<j\leqslant n} \alpha_i\beta_j x_i x_j + \sum_{1\leqslant i<j\leqslant n} \alpha_i\beta_j(1-x_i)(1-x_j), + \sum_{j=1}^{n}\mu_j x_j + \sum_{j=1}^{n} v_j(1-x_j) + \Gamma, \quad (1)$$

$$\text{s.t.} \quad \sum_{j=1}^{n}\alpha_j x_j \leqslant A, \quad (2)$$

$$x_j \in \{0,1\}, \quad \text{for } 1\leqslant j\leqslant n, \quad (3)$$

where $n \geqslant 1$, $A \geqslant 0$, and all coefficients α_j, β_j, μ_j, v_j for $1 \leqslant j \leqslant n$ and Γ are non-negative integers. The problem is called the symmetric quadratic knapsack problem, or problem (SQKP) (Kellerer and Strusevich, 2010), because both the quadratic and the linear parts of the objective function (1) are separated into two terms, one depending on the variables x_j, and the other depending on the variables $(1-x_j)$, with the coefficients of the quadratic terms x_j^2 and $(1-x_j)^2$ being the same. In problem (SQKP), each α_j appears in both the quadratic terms of (1) and the linear constraint (2). We use x^* to denote an optimal solution to problem (SQKP). Accordingly, $Z(x^*)$ indicates the optimal objective value.

Problem (SQKP) was motivated by its applications in machine scheduling. It was first introduced by Kellerer and Strusevich (2010) in the study of a machine scheduling problem denoted by $1|h(1), N-res|\sum w_jC_j$, which aims to minimize the total weighted completion time for a single machine with a fixed non-availability interval $[s,t]$ for $0 \leqslant s < t$, and where the crossover job, i.e., the job affected by the non-availability interval, is not resumable. Kellerer and Strusevich (2010) showed that problem $1|h(1), N-res|\sum w_jC_j$ can be formulated as a special version of problem (SQKP) with $A = s$, $\alpha_i/\beta_i \leqslant \alpha_j/\beta_j$ for $1 \leqslant i < j \leqslant n$, $\mu_j = 0$ and $v_j = \beta_j t$ for $1 \leqslant j \leqslant n$, and $\Gamma = \sum_{j=1}^{n}\alpha_j\beta_j$. Moreover, Kellerer et al. (2010) studied a machine scheduling problem denoted by $1|d_j = d|\sum w_j(E_j + T_j)$, which aims to minimize the weighted earliness and tardiness for a single machine with respect to a common restrictive due date denoted by d. They showed that if it does not allow any straddling job, i.e., any job that starts before time d and is completed after time d, problem $1|d_j = d|\sum w_j(E_j + T_j)$ can be formulated as a special version of problem (SQKP), with $A = d$, $\alpha_i/\beta_i \leqslant \alpha_j/\beta_j$ for $1 \leqslant i < j \leqslant n$, $\mu_j = 0$ and $v_j = \alpha_j\beta_j$ for $1 \leqslant j \leqslant n$, and $\Gamma = 0$.

To see more applications of problem (SQKP), let us consider a problem which aims to minimize the total weighted completion time for two parallel machines, with machine 1 available only in a time interval $[s_1, t_1]$, and with machine 2 available only at time s_2 and afterwards, where $0 \leqslant s_1 < t_1$ and $0 \leqslant s_2$. We refer to this problem as the generalized capacitated sum of weighted job completion times problem (GCSWCP), because a capacitated sum of job completion times problem (CSCP), which was studied by Lee and Danusaputro Liman (1993), is a special version of this problem with $s_1 = s_2 = 0$ and with each job associated with a weight equal to one. Moreover, problem $1|h(1), N-res|\sum w_jC_j$ is also a special version of problem (GCSWCP), because a single machine with a non-availability interval $[s,t]$ can be split into two machines, with machine 1 available only in interval $[0,s]$ and machine 2 available only at time t and afterwards.

* Tel.: +852 34003624; fax: +852 25502704.
E-mail address: lgtzx@polyu.edu.hk

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.10.049

Using the interchanging argument, it is easy to see that there always exists an optimal schedule for problem (GCSWCP), such that the jobs scheduled on each machine are sequenced in the Weighted Shortest Processing Time (WSPT) order (Pinedo, 1995). Thus, we can assume that jobs are indexed according to the WSPT order so that $p_i/w_i \leqslant p_j/w_j$ for $1 \leqslant i < j \leqslant n$, where n is the number of jobs, and w_j and p_j denote the weight and the processing time of job j for $1 \leqslant j \leqslant n$. Let x_j for each $1 \leqslant j \leqslant n$ be a decision variable, with $x_j = 1$ and 0 indicating that job j is processed on machine 1 and machine 2 respectively. Problem (GCSWCP) can be formulated so as to minimize $\sum_{j=1}^{n} w_j x_j \left[s_1 + \sum_{i=1}^{j} p_i x_i\right] + \sum_{j=1}^{n} w_j(1-x_j)\left[s_2 + \sum_{i=1}^{j} p_i(1-x_i)\right]$, subject to $s_1 + \sum_{j=1}^{n} p_j x_j \leqslant t_1$, which is a special version of problem (SQKP), with $A = t_1 - s_1$, $\alpha_j = p_j$, $\beta_j = w_j$, $\mu_j = w_j s_1$, $\nu_j = w_j s_2$ for $1 \leqslant j \leqslant n$, and $\Gamma = \sum_{j=1}^{n} w_j p_j$. Thus, problem (CSCP) is also a special version of problem (SQKP), with $A = t_1$, $\alpha_j = p_j$, $\beta_j = 1$, $\mu_j = \nu_j = 0$ for $1 \leqslant j \leqslant n$, and $\Gamma = \sum_{j=1}^{n} p_j$.

Since problem (SQKP) is NP-hard (Kellerer and Strusevich, 2010), it calls for the design of approximation algorithms. Recall that a family of ρ-approximation algorithms is an approximation scheme if the approximation ratio ρ equals $1 + \epsilon$ for any given $\epsilon > 0$. An approximation scheme is a fully polynomial time approximation scheme (FPTAS) if its running time is polynomial in the size of the problem instance and $1/\epsilon$. An FPTAS is strongly polynomial time if its running time is bounded by a polynomial only in the number of integers in the problem instance (Korte and Vygen, 2008).

Based on a dynamic programming algorithm, Kellerer and Strusevich (2010) developed an approximation scheme for problem (SQKP) that can return a feasible solution x with $Z(x) \leqslant (1 + \epsilon)Z(x^*)$ for any $\epsilon > 0$. The approximation scheme needs to have an input of bounds L and U on $Z(x^*)$ with $L \leqslant Z(x^*) \leqslant U$. Accordingly, its running time is in $O((U/L)n^4/\epsilon^2)$, which depends on the value of U and L. Thus, the running time is polynomial only if U/L is bounded by a polynomial in the size of the problem instance.

Although, for some special versions of problem (SQKP), there exist polynomial time algorithms to compute such bounds L and U, so as to satisfy the conditions mentioned above, these algorithms cannot be extended to the general version. Kellerer et al. (2010) studied a special version of problem (SQKP) with two additional constraints, (i) $\alpha_i/\beta_i \leqslant \alpha_j/\beta_j$ for $1 \leqslant i < j \leqslant n$ and (ii) $\nu_j \geqslant \alpha_j\beta_j$ for $1 \leqslant j \leqslant n$. For this special version they developed a rounding algorithm that can compute L and U in $O(n^3)$ time, such that $L \leqslant Z(x^*) \leqslant U$ and $U/L \leqslant 9.9$. Thus, for problem $1|d_j = d|\sum w_j(E_j + T_j)$, since the corresponding special version of problem (SQKP) satisfies the two additional constraints, the rounding algorithm can be applied to compute L and U. In contrast, for problem $1|h(1), N-res|\sum w_j C_j$, since the corresponding special version of problem (SQKP) does not satisfy constraint (ii), the rounding algorithms cannot be applied. However, Kellerer and Strusevich (2010) showed that for this special version, a heuristic algorithm by Wang et al. (2005) can be used to compute L and U in $O(n^2)$ time, such that $L \leqslant Z(x^*) \leqslant U$ and $U/L \leqslant 4$. Moreover, for problem (CSCP), although the corresponding special version of problem (SQKP) does not satisfy constraint (ii), there exists a 2-approximation algorithm (Lee and Danusaputro Liman, 1993) that can compute a feasible solution x in polynomial time with $Z(x)/2 \leqslant Z(x^*) \leqslant Z(x)$. Thus, letting $L = Z(x)$ and $U = 2Z(x)$, one can obtain $L \leqslant Z(x^*) \leqslant U$ and $U/L = 2$. Hence, each of the above three scheduling problems has an FPTAS with a strongly polynomial running time of $O(n^4/\epsilon^2)$. However, for problem (GCSWCP), which has not been studied in previous literature, the corresponding special version of problem (SQKP) does not satisfy constraint (ii), and the 2-approximation algorithm by Lee and Danusaputro Liman (1993) for problem (CSCP) is not applicable.

Moreover, if the linear constraint (2) is relaxed, and $\sum_{1\leqslant i<j\leqslant n} \alpha_i\beta_j(1-x_i)(1-x_j)$ is removed from the objective function (1), problem (SQKP) becomes a positive half product minimization problem, which has several applications in scheduling with controllable processing times, and is known to admit an FPTAS (Janiak et al., 2005).

In this paper, we develop a polynomial time algorithm to compute a new lower bound L_1 and a new upper bound U_1 on $Z(x^*)$ for the general version of problem (SQKP), such that U_1/L_1 is in $O(n^2)$. Given such L_1 and U_1, the approximation scheme proposed by Kellerer and Strusevich (2010) is an FPTAS with a strongly polynomial running time of $O(n^6/\epsilon^2)$. We further improve the FPTAS to reduce the running time to $O(n^4 \log \log n + n^4/\epsilon^2)$. The FPTAS obtained can be directly applied to problem (GCSWCP).

The remainder of this paper is organized as follows. We first present the new bounds on $Z(x^*)$ in Section 2, followed by the algorithm to compute the bounds in Section 3. We then present the improved FPTAS in Section 4, and draw a conclusion in Section 5.

2. New bounds on $Z(x^*)$

To derive new lower and upper bounds on $Z(x^*)$, we first define $A(\lambda)$ as follows to be the minimum value of the total weight, $\sum_{j=1}^{n} \alpha_j x_j$, for $x \in \{0,1\}^n$, such that each of the quadratic terms and the linear terms of $Z(x)$ in (1) does not exceed λ for any given $\lambda \geqslant 0$.

$$A(\lambda) = \min \sum_{j=1}^{n} \alpha_j x_j \tag{4}$$

$$\text{s.t.} \quad \alpha_i\beta_j x_i x_j \leqslant \lambda, \quad \text{for } 1 \leqslant i < j \leqslant n, \tag{5}$$

$$\alpha_i\beta_j(1-x_i)(1-x_j) \leqslant \lambda, \quad \text{for } 1 \leqslant i < j \leqslant n, \tag{6}$$

$$\mu_j x_j \leqslant \lambda, \quad \text{for } 1 \leqslant j \leqslant n, \tag{7}$$

$$\nu_j(1-x_j) \leqslant \lambda, \quad \text{for } 1 \leqslant j \leqslant n, \tag{8}$$

$$x_j \in \{0,1\}, \quad \text{for } 1 \leqslant j \leqslant n. \tag{9}$$

If no $x \in \{0,1\}^n$ satisfies constraints (5)–(9), then we define $A(\lambda)$ to be ∞, greater than all integers. It can be seen that $A(\lambda)$ is non-increasing in λ.

Let Λ denote a set of zero and the values of all the coefficients of (1), i.e.,

$$\Lambda = \{0\} \cup \{\alpha_i\beta_j : 1 \leqslant i < j \leqslant n\} \cup \{\mu_j : 1 \leqslant j \leqslant n\} \cup \{\nu_j : 1 \leqslant j \leqslant n\}. \tag{10}$$

We define λ^* as follows to be the minimum value of all $\lambda \in \Lambda$ with $A(\lambda) \leqslant A$.

$$\lambda^* = \min\{\lambda : A(\lambda) \leqslant A, \lambda \in \Lambda\}. \tag{11}$$

Since for $\lambda = \max\{p : p \in \Lambda\}$, setting $x_j = 0$ for $1 \leqslant j \leqslant n$ satisfies constraints (5)–(9) and $\sum_{j=1}^{n} \alpha_j x_j = 0 \leqslant A$, we obtain $A(\max\{p : p \in \Lambda\}) = 0 \leqslant A$, which implies that

$$0 \leqslant \lambda^* \leqslant \max\{p : p \in \Lambda\}. \tag{12}$$

Accordingly, the following statement holds, suggesting new lower and upper bounds on $Z(x^*)$:

Lemma 1

$$\lambda^* + \Gamma \leqslant Z(x^*) \leqslant [n(n-1)/2 + n](\lambda^* + \Gamma)$$

Proof. Consider any $x \in \{0,1\}^n$ that satisfies (5)–(9) for $\lambda = \lambda^*$, and also satisfies $\sum_{j=1}^{n} \alpha_j x_j = A(\lambda^*)$. Thus, since $A(\lambda^*) \leqslant A$, we have $\sum_{j=1}^{n} \alpha_j x_j \leqslant A$, which implies that x is a feasible solution to problem (SQKP). Due to (5)–(8) and $x_j \in \{0,1\}$ for $1 \leqslant j \leqslant n$, we have

$$\sum_{1\leqslant i<j\leqslant n} \alpha_i\beta_j[x_i x_j + (1-x_i)(1-x_j)] \leqslant \lambda^* \sum_{1\leqslant i<j\leqslant n} [x_i x_j + (1-x_i)(1-x_j)] \leqslant \lambda^* \sum_{1\leqslant i<j\leqslant n} 1 \leqslant [n(n-1)/2]\lambda^*,$$