Computers & Operations Research 77 (2017) 1–10

Contents lists available at ScienceDirect

Computers & Operations Research

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

The boolean quadratic programming problem with generalized upper bound constraints ☆

Yang Wang [a,*], Abraham P. Punnen [b]

[a] School of Management, Northwestern Polytechnical University, 127 Youyi West Road, 710072 Xi'an, China
[b] Department of Mathematics, Simon Fraser University Surrey, 250-13450 102nd AV, Surrey, British Columbia, Canada V3T 0A3

ARTICLE INFO

Article history:
Received 14 February 2016
Received in revised form
7 June 2016
Accepted 7 July 2016
Available online 9 July 2016

Keywords:
Discrete optimization
Boolean quadratic programs
Quadratic assignment
Quadratic semi-assignment
Generalized upper bounds
Metaheuristics

ABSTRACT

We consider the boolean quadratic programming problem with generalized upper bound constraints (BQP-GUB) which subsumes the well-known quadratic semi-assignment problem. BQP-GUB has applications in engineering, production planning and biology. We present various complexity results on the problem along with different metaheuristic algorithms. Results of extensive experimental analysis are presented demonstrating the efficacy of our proposed algorithms.

1. Introduction

Let $E = \{1, 2, \ldots, n\}$ be a finite set and for each $i \in E$ a profit c_i is given. Also for each $(i, j) \in E \times E$, a profit q_{ij} is prescribed. The $n \times n$ matrix $Q = (q_{ij})$ is called the *quadratic profit matrix* and the vector $c = (c_1, c_2, \ldots, c_n)$ is called *the linear profit vector*. Let $S_1, S_2, \ldots S_m$ be a partition of E, i.e. $S_1, S_2, \ldots S_m$ are mutually disjoint subsets such that $E = \cup_{i=1}^{m} S_i$. Then the *boolean quadratic programming problem with generalized upper bound constraints* (BQP-GUB) can be stated as follows:

$$\text{Maximize} \quad f(x) = \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{n} \sum_{j=1}^{n} q_{ij} x_i x_j$$

$$\text{Subject to} \quad \sum_{j \in S_k} x_j = 1, \quad \text{for } k = 1, 2, \ldots, m. \tag{C1}$$

$$x_j \in \{0, 1\}, \quad \text{for } j = 1, 2, \ldots, n. \tag{C2}$$

BQP-GUB is a generalization of the *quadratic semi-assignment problem* (QSAP) studied by various authors. In fact, if $|S_k| = m$ for all $k = 1, 2, \ldots, m$ and $n = m^2$, BQP-GUB reduces to the QSAP. Also, it is possible to formulate BQP-GUB as a QSAP by introducing $O(n)$ additional variables (if necessary). In this sense, BQP-GUB and QSAP are equivalent. However, the general form of BQP-GUB presented here allows additional flexibility in terms of presentation, modeling, complexity and in identifying polynomially solvable special cases.

Applications of the QSAP model include register allocations in optimized compiler design [11,16,29], allocation of assets to tasks [12], co-clustering of image segments [33], Rotamer assignments in protein folding [15], Correlation clustering [7], scheduling [6,23,30,32], hub location [25], and metric labeling [19]. Thus each of these problems are applications of BQP-GUB as well and this rich applications base is one of the major motivations for our study.

A graph theoretic interpretation of BQP-GUB can be given as follows: let $G = (V, A)$ be an undirected graph. The vertex set $V = \{1, 2, \ldots, n\}$ is partitioned into m subsets $S_1, S_2, \ldots, S_m$. For each edge $(i, j) \in A$ a profit c_{ij} is prescribed. Then the cluster restricted maximum induced subgraph problem (CMISP) is to find a subset S of V such that S contains exactly one element from each S_k, $k = 1, 2, \ldots, m$ and the sum of the weights of the edges in the subgraph of G induced by S is maximized (see Fig. 1). By choosing $q_{ij} = c_{ij}$ if $(i, j) \in A$ and $q_{ij} = 0$ if $(i, j) \notin A$, CMISP can be solved as a BQP-GUB.

In the definition of CMISP if we include the additional restriction that the subgraph induced by S must be a clique, we get cluster restricted maximum clique problem (CMCP). By choosing $q_{ij} = c_{ij}$ if $(i, j) \in A$ and $q_{ij} = -M$ if $(i, j) \notin A$, where M is a large

☆A preliminary version of this paper was presented at the IFORS conference in Barcelona, July 2014.

* Corresponding author.
E-mail addresses: sparkle.wy@gmail.com (Y. Wang), apunnen@sfu.ca (A.P. Punnen).

http://dx.doi.org/10.1016/j.cor.2016.07.005
0305-0548/

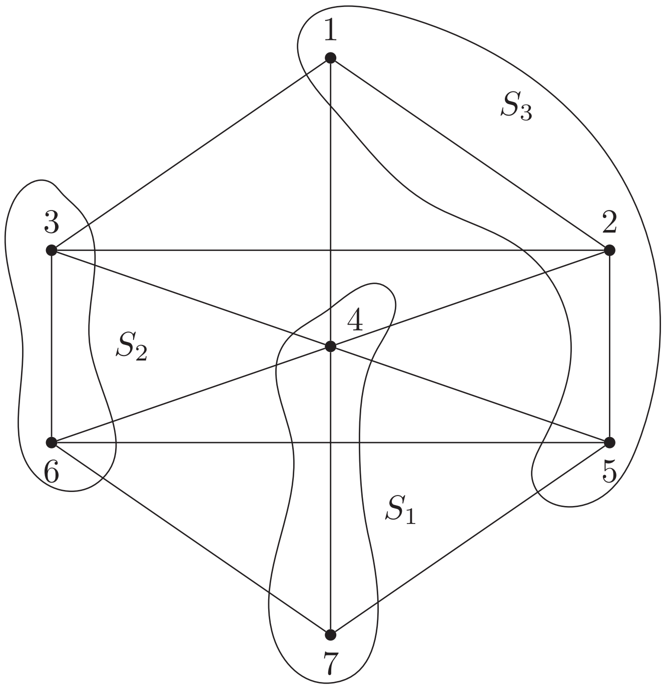

Fig. 1. An example of a graph with portioned node set and $c_{ij} = 1$ for all (i,j) in the edge set. The node set $\{2, 3, 4\}$ gives an optimal solution to the resulting CMISP.

positive number, we can solve CMCP as a BQP-GUB. The example given in Fig. 1 also serves as an example for CMCP.

Another problem closely related to BQP-GUB is the unconstrained boolean quadratic programming problem (UBQP) studied by various authors [22,34]. It is possible to formulate BQP-GUB as a UBQP by modifying q_{ij} values appropriately. Thus such a reformulation is a viable approach to solve BQP-GUB by making use of powerful UBQP solvers. However, it is generally accepted that specialized algorithms, exploiting the underlying problem structure, are likely to perform better than general purpose algorithms. Our experimental analysis confirms this assertion.

The problem BQP-GUB (in the form of QSAP) was introduced by Greenberg [14] in 1969. Since then several researchers have studied the problem in various contexts. Polynomially solvable special cases of QSAP were investigated in [2,10], lower bounds and reduction techniques were studied in [3,4], structure of the polytope associated with a 0-1 linear programming formulation of BQP-GUB was studied in [28] and a tabu search heuristic was proposed in [9] along with some experimental results involving small size problems. Surprisingly, a systematic and detailed analysis of heuristic algorithms for this problem is not available. Through this study, we expect to reduce this gap.

In this paper, we first conduct a preliminary analysis of the problem complexity to separate easy and hard special cases. BQP-GUB is trivial if $|S_k| = 1$ for all $k = 1, 2, \ldots, m$. Interestingly, if $|S_i| \le 2$ for all i, BQP-GUB is shown to be NP-hard. It remains NP-hard, even if the rank of Q is 1. Testing if the problem has a unique optimal solution is also NP-hard. If $m = O\left(\frac{\log n}{\log \mu_s}\right)$, where μ_s is the geometric mean of $|S_1|, |S_2|, \ldots, |S_m|$ then the problem can be solved in polynomial time. Also, BQP-GUB can be solved by a simple greedy algorithm if Q is a *sum matrix*. We then develop various heuristic algorithms for the problem based on local search, variable neighborhood search, tabu search, iterated tabu search, GRASP and iterated greedy. Results of extensive experimental analysis carried out using these algorithms on benchmark problems are also reported. In addition, we compared our algorithms against the best known heuristic for UBQP by reformulating BQP-GUB as a UBQP. Our specialized algorithm based on iterated tabu search (ITS) outperformed all other approaches we tested, including the approach of reformulating BQP-GUB as a UBQP and solving the resulting UBQP using state-of-the-art solvers. The advantage of ITS is even more pronounced for large scale problems.

The paper is organized as follows. In Section 2, we discuss computational complexity of BQP-GUB and present some polynomially solvable special cases. Various heuristic algorithms are presented in Section 3. Section 4 deals with analysis of experimental results using our algorithms. Finally, concluding remarks are presented in Section 5.

2. Complexity and solvable cases

Since QSAP is NP-hard [5], BQP-GUB is clearly NP-hard. If $|S_k| = 1$ for $k = 1, 2, \ldots, m$, then BQP-GUB is trivial since $x_j = 1$ for $j = 1, 2, \ldots, n$ is the unique optimal solution. We now observe that a slight modification of this trivial case leads to a hard instance of BQP-GUB.

Theorem 1. *BQP-GUB is strongly NP-hard even if* $|S_k| \le 2$ *for all* $k = 1, 2, \ldots, m$.

Proof. We reduce an instance of UBQP to an instance of BQP-GUB satisfying the conditions of the theorem. Note that UBQP is the optimization problem:

$$\text{Maximize} \quad g(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} q'_{ij} x_i x_j$$

$$\text{Subject to} \quad x_j \in \{0, 1\}, \quad \text{for } j = 1, 2, \ldots, n.$$

where q'_{ij} is a given real number for $i, j \in E$. Consider an instance of UBQP with cost matrix $Q' = \left(q'_{ij}\right)_{n\times n}$. Now construct an instance of BQP-GUB on 2n variables $x_1, x_2, \ldots x_{2n}$ with the quadratic cost matrix $Q = (q_{ij})_{2n\times 2n}$ such that

$$q_{ij} = \begin{cases} q'_{ij} & \text{for } i, j = 1, 2, \ldots, n \\ 0 & \text{otherwise.} \end{cases}$$

Define $S_i = \{i, n+i\}$ for $i = 1, 2, \ldots, n$, and set $c_j = 0$ for $j = 1, 2\ldots, 2n$. The resulting BQP-GUB instance has 2^n solutions and there is a one-to-one correspondence between solutions of this instance and the solutions of UBQP such that the objective function values of the corresponding solutions are the same. The result now follows from the strong NP-completeness of UBQP. □

It may be noted that the reduction discussed in the above proof preserves approximation ratios. This translates non-approximability results available for UBQP into non-approximability results for BQP-GUB even when $|S_k| \le 2$ for all $k = 1, 2, \ldots, m$.

In the proof of Theorem 1, we showed that a UBQP with n variables can be solved as a BQP-GUB with $2n$ variables. We now observe that BQP-GUB with n variables can be formulated as UBQP with n variables, yielding equivalence between these two problems.

Choose $\alpha > 1 + \sum_{i=1}^{n}\sum_{j=1}^{n} \max\{q_{ij}, 0\}$. Then BQP-GUB is equivalent to solving the problem

$$\text{Maximize} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij} x_i x_j + \sum_{j=1}^{n} c_j x_j - \alpha \sum_{k=1}^{m} \left(\sum_{j \in S_k} x_j - 1 \right)^2$$

$$\text{Subject to} \quad x_j \in \{0, 1\}, \quad \text{for } j = 1, 2, \ldots, n.$$

But this is equivalent to the UBQP where