European Journal of Operational Research 241 (2015) 697–707

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Integrating tabu search and VLSN search to develop enhanced algorithms: A case study using bipartite boolean quadratic programs ☆

Fred Glover [a], Tao Ye [b,*], Abraham P. Punnen [b], Gary Kochenberger [c]

[a] OptTek Systems, Boulder, CO, USA
[b] Department of Mathematics, Simon Fraser University, Surrey, British Columbia V3T 0A3, Canada
[c] School of Business, University of Colorado at Denver, Denver, CO, USA

ARTICLE INFO

Article history:
Received 1 June 2013
Accepted 21 September 2014
Available online 2 October 2014

Keywords:
Quadratic programming
Boolean variables
Metaheuristics
Tabu search
Worst-case analysis

ABSTRACT

The bipartite boolean quadratic programming problem (BBQP) is a generalization of the well studied boolean quadratic programming problem. The model has a variety of real life applications; however, empirical studies of the model are not available in the literature, except in a few isolated instances. In this paper, we develop efficient heuristic algorithms based on tabu search, very large scale neighborhood (VLSN) search, and a hybrid algorithm that integrates the two. The computational study establishes that effective integration of simple tabu search with VLSN search results in superior outcomes, and suggests the value of such an integration in other settings. Complexity analysis and implementation details are provided along with conclusions drawn from experimental analysis. In addition, we obtain solutions better than the best previously known for almost all medium and large size benchmark instances.

1. Introduction

Local search (also called neighborhood search) is typically incorporated as a fundamental component within metaheuristic methods such as evolutionary algorithms, simulated annealing, particle swarm optimization and (more directly) tabu search, producing methods of choice for solving many complex applied optimization problems. Traditional local search algorithms use exhaustive search over small neighborhoods or employ candidate list strategies as proposed in tabu search to handle large neighborhoods. In addition a number of recent local search algorithms use neighborhoods of exponential size that often can be searched for improving solutions in polynomial time. To distinguish between the use of exhaustive search over small neighborhoods and the use of polynomial time methods for searching large neighborhoods we call the former *simple neighborhood search* (SN search) and the latter *very large-scale neighborhood search* (VLSN search) (Ahuja, Ergun, Orlin, & Punnen, 2002). SN search algorithms are generally faster in exploring neighborhoods but take a large number of iterations to reach a locally optimal solution. Many VLSN search algorithms on the other hand take longer to search a neighborhood for an improving solution but often reach a locally optimal solution quickly within a relatively small number of iterations. In this paper we consider an integration of SN search and VLSN search within a tabu search framework to develop enhanced algorithms for an important combinatorial optimization problem called the *bipartite boolean quadratic programming problem* (BBQP). Our computational outcomes suggest the merit of such an integration in other settings.

1.1. Problem description

Let $Q = (q_{ij})$ be an $m \times n$ matrix, $c = (c_1, c_2, \ldots, c_m)$ be a row vector in R^m and $d = (d_1, d_2, \ldots, d_n)$ be a row vector in R^n. Then, the problem (BBQP) can be stated mathematically as

$$\text{BBQP}: \text{Maximize } f(x, y) = x^T Qy + cx + dy$$
$$\text{subject to } x \in \{0, 1\}^m, y \in \{0, 1\}^n.$$

An instance of BBQP is completely defined by the matrix Q and vectors c and d and hence may be represented by $\mathscr{P}(Q, c, d)$.

The problem can be viewed as a generalization of the well-studied *boolean quadratic programming problem* (BQP) (Glover, Kochenberger, & Alidaee, 1998; Glover & Hao, 2010; Lü, Glover, & Hao, 2010; Wang, Lü, Glover, & Hao, 2012)

$$\text{BQP}: \text{Maximize } f(x) = x^T Q'x + c'x$$
$$\text{subject to } x \in \{0, 1\}^n,$$

where Q' is an $n \times n$ matrix and c' is a row vector in R^n. As pointed out in Punnen, Sripratak, and Karapetyan (2012a,b), by choosing

☆ This research work was supported by an NSERC Discovery grant and an NSERC discovery accelerator grant awarded to Abraham P. Punnen.

* Corresponding author.
E-mail addresses: glover@opttek.com (F. Glover), yeetao@gmail.com (T. Ye), apunnen@sfu.ca (A. P. Punnen), Gary.Kochenberger@ucdenver.edu (G. Kochenberger).

http://dx.doi.org/10.1016/j.ejor.2014.09.036
0377-2217/

$$Q = Q' + 2MI, \quad c = \frac{1}{2}c' - Me \quad \text{and} \quad d = \frac{1}{2}c' - Me, \tag{1}$$

where I is an $n \times n$ identity matrix, $e \in R^n$ is an all one vector and M is a very large number, BQP can be formulated as a BBQP. In Eq. (1), the penalty parameter M forces $x_i = y_i$ in an optimal solution of this modified BBQP. This transformation is important since it provides additional flexibility in developing algorithms for BQP through BBQP formulations.

Since BBQP is a generalization of BQP, various applications studied in the context of BQP are also relevant to BBQP. Applications of the BQP model are very diverse, ranging from solving fundamental graph theoretic problems, to solving the Ising model of ferromagnetism in statistical mechanics, to designing special types of quantum computers. Rather than elaborating the details of such applications here, we discuss some applied problems that can be directly modeled as BBQP. Notably, many problems in pattern recognition and correlation clustering can be formulated as the problem of approximating a matrix by a rank-one binary matrix and this can be accomplished by solving a BBQP (Gillis & Glineur, 2011; Shen, Ji, & Ye, 2009; Lu, Vaidya, Atluri, Shin, & Jiang, 2011). This can be accomplished as follows. Suppose $A = (a_{ij})$ is a given $m \times n$ matrix and B is a rank-one binary matrix approximation of A. B is of size $m \times n$ and can be written as $B = xy^T$ for some $x \in \{0,1\}^m$ and $y \in \{0,1\}^n$. Thus B can be identified by minimizing $\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij} - x_iy_j)^2$. Since $x_i^2 = x_i$ and $y_j^2 = y_j$ for all i and j, we have

$$\begin{aligned}\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij} - x_iy_j)^2 &= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij}^2 - 2a_{ij}x_iy_j + x_i^2y_j^2)\\ &= \sum_{i=1}^{m}\sum_{j=1}^{n}a_{ij}^2 - \sum_{i=1}^{m}\sum_{j=1}^{n}x_i(2a_{ij}-1)y_j\\ &= K - x^TWy\end{aligned}$$

where $w_{ij} = 2a_{ij} - 1$ and $W = (w_{ij})_{m\times n}$. Thus, minimizing $\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij} - x_iy_j)^2$ is equivalent to solving a BBQP with $Q = W$, $c = 0$ and $d = 0$. If the components of B are required to belong to the set $\{-1, 1\}$ as in the case of some clustering applications, we can still model the problem as a BBQP using the above approach by converting the variables to binary using a linear transformation.

The BBQP model can be used to solve some graph theoretic optimization problems (Ambühl, Mastrolilli, & Svensson, 2011; Tan, 2008). For this, consider the bipartite graph $G = (U, V, E)$ and let $a_{ij} > 0$ be the weight of the edge $(i,j) \in E$. Then the *maximum weight biclique problem* (MWBP) is to find a biclique in G of maximum total edge-weight. Define

$$q_{ij} = \begin{cases} a_{ij} & \text{if } (i,j) \in E \\ -M & \text{otherwise,} \end{cases}$$

where M is a large positive number. Further, choose c and d as zero vectors. Then the solution to the resulting BBQP yields a solution to the MWBP. The problem MWBP has applications in data mining, clustering and bioinformatics (Chang, Vakati1, Krause, & Eulenstein, 2012; Tanay, Sharan, & Shamir, 2002).

Finally, we note that BBQP can also be used to solve the maximum weight cut problem in a bipartite graph and the maximum weight bipartite spanning subgraph problem in a general graph (Punnen, Sripratak, & Karapetyan, 2012a). The model can also be used to approximate the cutnorm of a matrix (Alon & Naor, 2006).

Despite its unifying role and various practical applications, BBQP has not been investigated thoroughly from an experimental analysis point of view. The only systematic study that we are aware of is by Karapetyan and Punnen (2012) who generated a class of test instances and provided experimental results with various heuristic algorithms. Some limited experimental study using a specific algorithm, called the alternating algorithm (Karapetyan and Punnen, 2012; Lu et al., 2011) is also available in the context of specific applications. After we released a preliminary version of our paper, Duarte, Laguna, Martíc, and Sánchez-Oroa (2014) reported experimental results with exact and heuristic algorithms for BBQP. They compared their algorithm with ours and reported that both algorithms perform well and neither dominates the other. We report additional analytical and experimental results in the current version of our work that provide new findings. We also suggest new directions for future research.

1.2. Research thrust

The focus of this paper is to develop efficient heuristic algorithms for solving BBQP. The major contributions can be summarized as follows:

- We present two neighborhood structures, a classic one-flip neighborhood and a new flip-float neighborhood, and based on them propose a one-flip move based tabu search algorithm, a flip-float move based coordinate method and a hybrid algorithm that combines the two. While the specific optimization problem addressed in this paper is BBQP, our approach for integrating tabu search (TS) and VLSN search is applicable to other settings to obtain hybrid algorithms that inherit individual properties of these algorithmic paradigms.
- We evaluate and analyze the proposed algorithms through computational experiments on a set of 85 benchmark instances (Karapetyan and Punnen, 2012). The experimental results show that the hybrid method is able to improve almost all the previous best-known solutions on the medium and large size instances, demonstrating its effectiveness and efficiency. They also disclose that the new flip-float neighborhood is the key reason for effectively solving the Bicluque and BMaxCut instances, and that the hybrid TS/VLSN method shows better performance in terms of both solution quality and robustness than either of its component methods in isolation.
- We compare our algorithms with the approach of using state-of-the-art mixed integer programming software by solving an integer programming formulation of BBQP using CPLEX (IBM, 2010). Because of the difficulty of obtaining exact (verifiably optimal) solutions, we focused on using appropriate parameter settings to guide the solver to seek good heuristic solutions within a prescribed time limit. For small size instances, this approach produced solutions comparable to those of our algorithms (but at the cost of additional time). For medium and large size instances, however, this approach proved to be impractical, reinforcing the need for special purpose algorithms to solve BBQP.
- We provide landscape analysis of the benchmark instances to identify the inherent difficulty of these instances for local search algorithms. Such a study has not previously been undertaken in the literature and provides additional insights into the structure of the benchmark instances.

The rest of the paper is organized as follows. Section 2 presents two neighborhood structures and describes in detail three heuristic algorithms. Section 3 reports and discusses computational statistics of the proposed algorithms on the standard benchmark instances. Landscape analysis of the benchmark instances is also discussed in this section. Finally, concluding remarks are provided in Section 4.

2. Neighborhoods and heuristics algorithms

We propose three heuristic algorithms for solving BBQP. The first algorithm adopts a classic one-flip neighborhood structure and a tabu search strategy. The second algorithm employs a new flip-float neighborhood structure and a coordinate ascent strategy. The third integrates the first and second algorithms to produce a hybrid method.