European Journal of Operational Research 223 (2012) 595–604

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Path relinking for unconstrained binary quadratic programming

Yang Wang [a], Zhipeng Lü [b], Fred Glover [c], Jin-Kao Hao [a,*]

[a] LERIA, Université d'Angers, 2 Boulevard Lavoisier, 49045 Angers, France
[b] School of Computer Science and Technology, Huazhong University of Science and Technology, 430074 Wuhan, China
[c] OptTek Systems, Inc., 2241 17th Street Boulder, CO 80302, USA

ARTICLE INFO

Article history:
Received 8 June 2011
Accepted 6 July 2012
Available online 20 July 2012

Keywords:
UBQP
Path relinking
Tabu search
MaxCut

ABSTRACT

This paper presents two path relinking algorithms to solve the unconstrained binary quadratic programming (UBQP) problem. One is based on a greedy strategy to generate the relinking path from the initial solution to the guiding solution and the other operates in a random way. We show extensive computational results on five sets of benchmarks, including 31 large random UBQP instances and 103 structured instances derived from the MaxCut problem. Comparisons with several state-of-the-art algorithms demonstrate the efficacy of our proposed algorithms in terms of both solution quality and computational efficiency. It is noteworthy that both algorithms are able to improve the previous best known results for almost 40 percent of the 103 MaxCut instances.

1. Introduction

The objective of the unconstrained binary quadratic programming (UBQP) problem is to maximize the function:

$$f(x) = x'Qx = \sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}x_ix_j \quad (1)$$

where $Q = (q_{ij})$ is an n by n matrix of constants and x is an n-vector of binary (zero-one) variables, i.e., $x_i \in \{0,1\}$, $i = 1, \ldots, n$.

The formulation of UBQP can represent a wide range of important problems, including those from financial analysis [28], social psychology [20], computer aided design [25] and cellular radio channel allocation [9]. Moreover, a quite number of combinatorial optimization problems can be transformed into UBQP, such as graph coloring problem, maxcut problem, set packing problem, set partitioning problem, maximum clique problem, etc. Interested readers can refer to [23] for the general transformation procedures.

Given the interest of UBQP, many solution procedures have been reported in the literature during the past few decades. Exact methods based on branch and bound or branch and cut [6,21,35] are quite useful to obtain optimal solutions to instances of limited sizes. To handle larger instances, a number of heurisric and meta-heuristic methods have been developed, including local search [7], Simulated Annealing [4,22], Tabu Search [14,19,32,34,37,38], and Evolutionary and Memetic Algorithms [5,26,27,30,31].

Among the existing heuristics, tabu search (TS) based algorithms are the most successful ones. For example, the first adaptive memory tabu search algorithm for the UBQP [14] has been used to solve applications coming from a wide variety of settings. Also, several multi-start tabu search strategies have been explored in [32] and a sequel using an iterated tabu search algorithm has been investigated in [34], leading to very good results on large and challenging UBQP random instances. More recently, the diversification-driven tabu search method [19], a memetic algorithm [27] using embedded tabu search and a variable fixing tabu search method [37,38] have proved to be especially effective for solving the most challenging UBQP instances.

Although numerous algorithms and approaches have been proposed for this well-known problem, we are not aware of any study on applying path relinking to the UBQP in the literature. Path relinking is a general search strategy closely associated with tabu search and its underlying ideas share a significant intersection with the tabu search perspective [15–17], with applications in a variety of contexts where it has proved to be very effective in solving difficult problems. In this paper, we follow the general scheme described in [17] and propose two path relinking algorithms for the UBQP. These two algorithms differ from each other mainly on the way of generating the path, one employing a greedy strategy and the other employing a random construction. In order to assess the performance of our path relinking algorithms, we provide computational results on five sets of random and structured benchmarks with a total of 134 test instances. These results indicate that our proposed algorithms yield highly competitive outcomes on the tested instances.

The remaining part of the paper is organized as follows. Section 2 briefly reviews some representative approaches for the UBQP.

* Corresponding author. Tel.: +33 2 41 73 50 76.
E-mail addresses: yangw@info.univ-angers.fr (Y. Wang), zhipeng.lui@gmail.com, zhipeng.lv@hust.edu.cn (Z. Lü), glover@opttek.com (F. Glover), hao@info.univ-angers.fr (J.-K. Hao).

0377-2217/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ejor.2012.07.012

Section 3 describes the ingredients of our path relinking algorithms. Section 4 presents computational results and detailed comparisons with other state-of-the-art algorithms in the literature. Section 5 discusses the results obtained on two other well-known combinatorial problems. Concluding remarks are given in Section 6.

2. Previous work

This section reviews some representative heuristic approaches for the UBQP, including in particular those that are used as the reference methods for our experimental evaluation.

Glover et al. [14] introduced the first tabu search algorithm for the UBQP (AMTS). AMTS is based on the one-flip move and two types of memory structures to record recency and frequency information. Strategic oscillation is employed to alternate between constructive phases (progressively setting variables to 1) and destructive phases (progressively setting variables to 0), which are triggered by critical events, i.e., when the next move causes the objective function to decrease. The amplitude of the scillation is adaptively controlled by a span parameter. Computational results for instances with up to 500 variables show AMTS outperforms the best exact and heuristic methods previously reported in the literature.

Katayama and Narihisa [22] designed a simulated annealing algorithm (SA) that is also based on the one-flip move and an incremental neighborhood evaluation technique. To enhance its search ability, the SA algorithm adopts multiple annealing processes starting from different temperatures. Tested on instances with variables ranging from 500 to 2500, the proposed SA heuristic shows very competitive performances, particularly for the largest instances.

Merz and Katayama [31] conducted a landscape analysis and observed that local optima of the UBQP instances are contained in a small fraction of the search space. Based on this, they designed a memetic algorithm (MA) in which a dedicated crossover operator is utilized to generate good starting solutions for a k-opt local search. The proposed approach is remarkably effective in solving a set of problems with up to 2500 variables.

Palubeckis [32] presented several multistart tabu search strategies (MST) dedicated to the construction of the initial solution. An additional set of challenging random instances with up to 7000 variables were generated to evaluate the proposed MST algorithms. Subsequently, Palubeckis [34] developed an iterated tabu search algorithm (ITS) in which the perturbation mechanism operates on a specific set of variables. The experimental results indicated that the ITS consumes less computational effort to find the best solutions than several MSTS algorithms.

Glover et al. [19] presented a diversification-driven tabu search (D^2TS) algorithm that alternates between a basic tabu search procedure and a memory-based perturbation strategy guided by a long-term memory. Despite its simplicity, computational results showed that D^2TS is capable of matching or improving the previously reported results for the challenging instances introduced in [32].

Lü et al. [27] proposed a hybrid metaheuristic approach (HMA) which combines a basic tabu search procedure and the genetic search framework. HMA is characterized by its diversification-guided recombination operator and quality-and-distance-based population updating strategy. The dedicated recombination operator aims to generate diversified offspring solutions in order to explore new promising search regions while the tabu search procedure is responsible for intensified examination around the offspring solutions. Computational results showed HMA is among the current best performing procedures on the UBQP benchmark instances.

3. Path relinking algorithm

3.1. Main framework

Algorithm 1. Outline of the path relinking procedure

1: **Input**: matrix Q
2: **Output**: the best binary n-vector x^* found so far and its objective value f^*
3: **repeat**
4: Initialize $RefSet = \{x^1, \ldots, x^b\}$
5: Identify the best solution x^* and the worst solution x^w in $RefSet$ and record the objective value f^* of solution x^*
6: $Tag(i)$ = TRUE, $(i = \{1, \ldots, b\})$
7: $PairSet \leftarrow \{(i,j): x^i, x^j \in RefSet, x^i \neq x^j, Tag(i) \cup Tag(j) = \text{TRUE}\}$
8: $Tag(i)$ = FALSE, $(i = \{1, \ldots, b\})$
9: **while** ($PairSet \neq \emptyset$) **do**
10: Pick solution pair $(x^i, x^j) \in RefSet$ with index pair (i,j) in $PairSet$
11: Apply the Relinking Method to produce the sequence $x^i = x(1), \ldots, x(r) = x^j$
12: Select $x(m)$ from the sequence and apply the improvement method to $x(m)$
13: **if** $f(x(m)) > f^*$ **then**
14: $x^* = x(m)$, $f^* = f(x(m))$
15: **end if**
16: **if** (Update_RefSet($RefSet$, $x(m)$)) **then**
17: $RefSet \leftarrow RefSet \cup \{x(m)\} \backslash \{x^w\}$
18: $Tag(w)$ = TRUE
19: Record the new worst solution x^w in $RefSet$
20: **end if**
21: Apply the Relinking Method to produce the sequence $x^j = y(1), \ldots, y(r) = x^i$
22: Select $y(n)$ from the sequence and apply the improvement method to $y(n)$
23: **if** ($f(y(n)) > f^*$) **then**
24: $x^* = y(n)$, $f^* = f(y(n))$
25: **end if**
26: **if** (Update_RefSet($RefSet$, $y(n)$)) **then**
27: $RefSet \leftarrow RefSet \cup \{y(n)\} \backslash \{x^w\}$
28: $Tag(w)$ = TRUE
29: Record the new worst solution x^w in $RefSet$
30: **end if**
31: $PairSet \leftarrow PairSet \backslash (i,j)$
32: **end while**
33: **until** the stopping criterion is satisfied

Algorithm 1 shows the path relinking procedure for UBQP. It starts with the creation of an initial set of b elite solutions $RefSet$ (line 4, see Section 3.2) and identifies the best and worst solutions in $RefSet$ in terms of the objective function value for the purpose of updating $RefSet$ (line 5). For each elite solution $x_i \in RefSet$, a binary value $Tag(i)$ indicates whether x_i can take part in a relinking process. Initially, assigning each solution in $RefSet$ a TRUE Tag which becomes FALSE when it is selected as the initiating solution or the guiding solution. The set $PairSet$ contains the index pairs (i,j) designating the initiating and guiding solution from $RefSet$ used for the relinking process. $PairSet$ is initially composed of all the index pairs (i,j) such that at least one corresponding Tag has the value TRUE (line 7). As soon as $PairSet$ is constructed, all the Tag are marked FALSE (line 8).