

# Chapter 1

## Introduction to QUBO



Abraham P. Punnen

**Abstract** This chapter provides a general introduction to the quadratic unconstrained binary optimization problem (QUBO). Starting with a brief historical review, we present some basic definitions and notations, equivalent representations, examples of important combinatorial optimization problems that are equivalent to QUBO and some additional motivating examples. We also discuss some of the basic mathematical programming formulations of QUBO along with relevant pointers to the contents of other chapters of the book.

### 1.1 Introduction

Mathematical programming models play a vital role in the socio-economic developments of the modern-day society. Optimization frameworks such as linear programming, quadratic programming, combinatorial optimization, and mixed integer programming are effectively used in engineering design, finance, healthcare, economics, medicine, transportation, supply chains, environment, telecommunications among others. Perhaps the most fundamental among the various optimization modeling tools is linear programming. This model building framework, as we know it today, was originated around 1938 and evolved into an applicable modelling technique in the early 1950s with the discovery of the simplex method [14]. Realization of the benefits associated with having integrality restrictions on some or all of the decision variables of a linear program led to the development of integer programming and significant advancements in this area continues to emerge [42].

The literature on unconstrained nonlinear optimization on the other hand, can be traced back to early days of calculus [66]. However, the development of successful methods for solving constrained nonlinear optimization problems took a very long time. Special nonlinear programs of minimizing convex quadratic functions

---

A. P. Punnen (✉)

Department of Mathematics, Simon Fraser University, Surrey, BC, Canada

e-mail: [apunnen@sfu.ca](mailto:apunnen@sfu.ca)

© Springer Nature Switzerland AG 2022

A. P. Punnen (ed.), *The Quadratic Unconstrained Binary Optimization Problem*,

[https://doi.org/10.1007/978-3-031-04520-2\\_1](https://doi.org/10.1007/978-3-031-04520-2_1)

over linear constraints can now be solved very efficiently. Study of constrained quadratic optimization problems started around the mid 1950s with applications in portfolio optimization [56, 57]. Systematic investigations on the quadratic programming problem with integrality restrictions on the decision variables took another decade to initiate [34, 53]. This book addresses a versatile unconstrained quadratic programming model called *quadratic unconstrained binary optimization problem* (QUBO) where the decision variables take values 0 or 1. The literature on QUBO is quite extensive and for a quick overview, we refer the reader to the survey papers [28, 45, 48].

Before getting into the technical details, let us start with some basic notations. All matrices are represented using bold capital letters and elements of a matrix are represented by the corresponding small letters along with accents, if any, and the location coordinates. For example the  $(i, j)$ th element of the matrix  $\mathbf{A}$  is  $a_{ij}$ , of the matrix  $\bar{\mathbf{B}}$  is  $\bar{b}_{ij}$ , and of the matrix  $\mathbf{D}^k$  is  $d_{ij}^k$ . Similarly, vectors are represented by boldface small letters along with appropriate accents, as applicable, and elements of the vector is represented using the same letter (without boldface) along with its location coordinate and accents, if any. For example, the  $i$ th element of the vector  $\mathbf{c}$  is  $c_i$ , of the vector  $\mathbf{x}^k$  is  $x_i^k$ , and of the vector  $\tilde{\mathbf{v}}$  is  $\tilde{v}_i$ . Exceptions to this rule will be stated explicitly and operators such as transpose etc. are not considered as accents in the case of vectors. The zero vector in any dimension is represented by  $\mathbf{0}$ . Additional notations will be introduced as need arises.

Let us now present a formal mathematical definition of QUBO. Let  $\mathbf{Q}$  be an  $n \times n$  matrix,  $\mathbf{c}^T = (c_1, c_2, \dots, c_n)$  be a row vector from  $\mathbb{R}^n$ , and  $\mathbf{x}^T = (x_1, x_2, \dots, x_n)$  be a row vector from  $\{0, 1\}^n$ . Then, QUBO is defined as the mathematical programming problem:

$$\begin{aligned} & \text{Maximize } \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ & \text{Subject to} \\ & \mathbf{x} \in \{0, 1\}^n. \end{aligned}$$

Since  $x_i^2 = x_i$  for any binary variable  $x_i$ , we can represent QUBO without the linear term by simply adding  $c_i$  to  $q_{ii}$  for  $i = 1, 2, \dots, n$ . Alternatively, we can assume  $q_{ii}$  to be zero by replacing  $c_i$  with  $c_i + q_{ii}$ . In fact, QUBO is represented in many other alternative forms which are discussed in more detail later in this chapter. QUBO sometimes is presented as a minimization problem, different from the standard maximization form. The data  $\mathbf{Q}, \mathbf{c}$  along with the orientation of optimization completely defines a problem instance. Thus, a maximization instance of QUBO can be represented by the triple  $(\mathbf{Q}, \mathbf{c}, \max)$  and a minimization instance of QUBO can be represented by  $(\mathbf{Q}, \mathbf{c}, \min)$ . Note that, the instances  $(\mathbf{Q}, \mathbf{c}, \max)$  and  $(-\mathbf{Q}, -\mathbf{c}, \min)$  are equivalent. Unless otherwise specified, we assume that QUBO is presented in the maximization form and we normally represent such an instance by the ordered pair  $(\mathbf{Q}, \mathbf{c})$ .

Let us now look at an interpretation of QUBO from the point of view of matrix theory. A principal submatrix of  $\mathbf{Q}$  is a square submatrix obtained by deleting rows

and columns corresponding to an index set  $S \subseteq \{1, 2, \dots, n\}$ . The value of a matrix is the sum of its elements. Without loss of generality assume  $\mathbf{c} = \mathbf{0}$ . Then QUBO is precisely the problem of computing a principal submatrix of  $\mathbf{Q}$  with maximum value.

A graph theoretic interpretation of QUBO can be given as follows. Let  $G = (V, E)$  be a graph such that  $V = \{1, 2, \dots, n\}$  and  $(i, j) \in E$  if and only if  $q_{ij} \neq 0$ . The graph  $G$  is called the *support graph* of  $\mathbf{Q}$ . Let  $q_{ij}$  be the cost of edge  $(i, j) \in E$  and  $q_{ii} + c_i$  be the cost of the vertex  $i \in V$ . Let  $S$  be a subset of  $V$  and  $G(S)$  be the subgraph of  $G$  induced by  $S$ . The cost of  $G(S)$  is defined as the sum of the cost of its edges and vertices. Then the QUBO seeks a subset  $S$  of  $V$  such that the cost of the induced subgraph  $G(S)$  of  $G$  is maximized.

Literature on QUBO, presented as a 0-1 quadratic program, can be traced back to 1960s, particularly with the work of Hammer and Rudeanu [34] on pseudo-Boolean functions and those of [13, 20, 21, 33, 80, 81]. Graph theoretic optimization problems such as the maximum weight stable set problem, the maximum weight clique problem, and the maximum cut problem have a hidden QUBO structure. As we will see later, these graph theoretic optimization problems are indeed equivalent to QUBO. In this sense, the history of QUBO is also linked to the origins of these graph theoretic structures.

A stable set of a graph  $G = (V, E)$  is a subset  $S$  of  $V$  such that no two vertices of  $S$  are adjacent in  $G$ . Let  $d_i$  be a given weight associated with vertex  $i \in V$ . Then, the *maximum weight stable set problem* is to find a stable set  $S$  in  $G$  such that  $\sum_{i \in S} d_i$  is maximized. It is easy to formulate the maximum weight stable set problem as a QUBO. Consider the binary decision variables  $x_i$  for  $i = 1, 2, \dots, n$  where  $x_i = 1$  represents the event that vertex  $i$  is in the stable set. Then, for any stable set  $S$  and  $(i, j) \in E$ , if the distinct vertices  $i, j \in S$  then  $x_i x_j = 0$ . This constraint can be forced using a quadratic term in the objective function. Define

$$q_{ij} = \begin{cases} d_i & \text{if } i = j \\ -M & \text{if } (i, j) \in E \\ 0 & \text{if } (i, j) \notin E, \end{cases}$$

where  $M$  is a large positive number. Also, choose  $\mathbf{c}$  as the zero vector. Then, the maximum weight stable set problem can be solved as the QUBO instance  $(\mathbf{Q}, \mathbf{c})$  where  $\mathbf{Q}$  and  $\mathbf{c}$  are as defined above. Interestingly, it is possible to formulate any instance  $(\mathbf{Q}, \mathbf{c})$  of QUBO as a maximum weight stable set problem and this will be discussed in Sect. 1.4 along with the other equivalent forms of QUBO.

A *clique* in the graph  $G$  is a subset  $S$  of  $V$  such that the subgraph  $G(S)$  of  $G$  induced by  $S$  is a complete graph. It is easy to see that  $S$  is a clique in  $G$  if and only if  $S$  is a stable set in the complement of  $G$ . Thus, in view of the discussion above, the maximum weight clique problem can also be formulated as a QUBO and it is also equivalent to QUBO.

Thus, from a historical perspective, cliques and stable sets play an indirect role in the evolution of QUBO. The literature on computing maximum weight clique and

maximum weight stable set is quite extensive and we will not attempt a detailed review of historical developments of these structures. However, we briefly discuss below some of the earlier works that introduced these notions. Although clique (stable set) is a well established concept in graph theory at present, the terminology has its roots in social sciences [19, 22, 55]. Luce and Perry [55] in 1949 defined a clique as follows:

A subset of the group forms a *clique* provided that it consists of three or more members each in the symmetric relation to each other member of the subset, and provided further that there can be found no element outside the subset that is in the symmetric relation to each of the elements of the subset.

The first algorithmic results on computing cliques was presented by Harrary and Ross [36]. Without using the name ‘clique’, the concept of complete subgraphs and independent sets (same as stable sets) were considered by Erdos and Szekeres [17] in 1935 while discussing Ramsey Theory in the context of graphs.

QUBO can also be written as a continuous optimization problem [70, 71]. Consider the box-constrained quadratic program

$$\begin{aligned} & \text{Maximize } \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} - M \mathbf{x}^T (\mathbf{e} - \mathbf{x}) \\ & \text{Subject to} \\ & \mathbf{x} \in [0, 1]^n \end{aligned}$$

where  $M$  is a large positive number and  $\mathbf{e}$  is the all-one vector in  $\mathbb{R}^n$ . Note that for large  $M$ , the objective function becomes convex and hence an optimal solution is attained at an extreme point of  $[0, 1]^n$  [70, 71]. The collection of extreme points of  $[0, 1]^n$  is precisely  $\{0, 1\}^n$  establishing the validity of the above representation of QUBO.

QUBO is strongly NP-hard. A thorough complexity analysis of the problem and various polynomially solvable special cases are discussed in Chap. 3 and complexity results in connection with approximability is discussed in Chap. 8.

## 1.2 The Ising Model

Long before the QUBO become popular within the operations research community, the model was used in statistical mechanics in an alternative form [75]. This is popularly known as the Ising model where the variables take values from  $\{-1, 1\}$ . The Ising model was originally proposed by Ernst Ising and Wilhelm Lenz in the 1920s to understand the behaviour of magnetic materials. This model considers a magnetic material as a set of molecules. The Molecules have spins which ‘align’ or ‘anti-align’ when a magnetic field is introduced and have pairwise interaction with each other [3, 5]. Let  $V = \{1, 2, \dots, n\}$  be a collection of molecules and  $x_i \in \{-1, 1\}$  represents the spin of molecule  $i$ . Let  $b_i$  be the strength of the magnetic field applied on molecule  $i$  and  $a_{ij}$  represents joint interaction field between neighbouring

spins of  $i$  and  $j$ . For a given spin vector  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , the corresponding energy value is defined as

$$E(\mathbf{x}) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j + \sum_{i=1}^n b_i x_i.$$

At low temperature levels, the system tends to low energy states and hence the sign of  $a_{ij}$  indicates possible spin direction. Thus, when  $a_{ij} > 0$  the system favors anti-aligned neighbouring spins, i.e.  $x_i x_j = -1$  leaving  $a_{ij} x_i x_j < 0$ . Likewise, when  $a_{ij} < 0$  the spins  $x_i$  and  $x_j$  are likely to align; i.e.  $x_i x_j = 1$  leaving  $a_{ij} x_i x_j < 0$ . This simple hypothetical model has been validated through experimental considerations by physicists for decades [10, 40, 61–63]. This underlying model is the basis of some of the current quantum inspired computing machines [5] which essentially brings a system to its lowest energy level which in turn indirectly computes the minimum of the energy function (maximum of  $-E(\mathbf{x})$ ) over variables that takes values  $-1$  or  $1$ .

The associated optimization problem, presented in the maximization problem, can be stated as

$$\begin{aligned} & \text{Maximize } \mathbf{y}^T \mathbf{A} \mathbf{y} + \mathbf{b}^T \mathbf{y} \\ & \text{Subject to} \\ & \mathbf{y} \in \{-1, 1\}^n \end{aligned}$$

where  $\mathbf{A}$  is an  $n \times n$  real valued matrix and  $\mathbf{b}^T = (b_1, b_2, \dots, b_n)$  is a row vector from  $\mathbb{R}^n$  and  $\mathbf{y}$  is a column vector in  $\{-1, 1\}^n$ . We refer to this version of QUBO as the *Ising QUBO*.

The models QUBO and the Ising QUBO are equivalent from an optimality point of view. Note that the linear transformation  $x_i = \frac{1}{2}(y_i + 1)$  for  $i = 1, 2, \dots, n$  reduces the QUBO  $(\mathbf{Q}, \mathbf{c})$  to the Ising QUBO  $(\mathbf{A}, \mathbf{b})$  where  $\mathbf{A} = \frac{1}{4}\mathbf{Q}$ ,  $b_i = \frac{1}{4} \left( 2c_i + \sum_{j=1}^n (q_{ij} + q_{ji}) \right)$  for  $i = 1, 2, \dots, n$  along with an additive constant  $q_0 = \frac{1}{4} \left( \sum_{i=1}^n \sum_{j=1}^n q_{ij} + \sum_{i=1}^n c_i \right)$ , which can be discarded. Similarly, an instance  $(\mathbf{A}, \mathbf{b})$  of the Ising QUBO can be reduced to an instance  $(\mathbf{Q}, \mathbf{c})$  of QUBO using the transformation  $y_i = 2x_i - 1$  for  $i = 1, 2, \dots, n$  with  $\mathbf{Q} = 4\mathbf{A}$ ,  $\mathbf{c} = 2\mathbf{b} + 2\mathbf{A}^T \mathbf{e} + 2\mathbf{A} \mathbf{e}$  along with the additive constant  $\mathbf{c}^T \mathbf{e} + \mathbf{e}^T \mathbf{A} \mathbf{e}$ .

The original Ising model was in fact a special case of the Ising QUBO. There are various extensions and generalizations of this popular model. For a historical account of various developments related to this model, we refer to the survey papers [10, 40, 61–63]. Although the name ‘Ising model’ is widely accepted within the physics community, it is not without controversy. For example, Barry Simon in his book on *The Statistical Mechanics of Lattice Gases* [75] writes:

Lattice models are caricatures invented to illuminate various aspects of elementary statistical mechanics, especially the phenomena of phase transitions and spontaneously broken symmetry. The simplest of all models is the Ising (or Lenz-Ising) model and this model

was suggested to Ising by his thesis adviser, Lenz. Ising solved the one-dimensional model, ..., and on the basis of the fact that the one-dimensional model had no phase transition, he asserted that there was no phase transition in any dimension. As we shall see, this is false. It is ironic that on the basis of an elementary calculation and erroneous conclusion, Ising's name has become among the most commonly mentioned in the theoretical physics literature. But history has had its revenge. Ising's name, which is correctly pronounced "E-zing," is almost universally mispronounced "I-zing."

Such discussions are probably important in physics to put contributions to scientific developments in context. We continue to use the terminology QUBO and Ising QUBO to distinguish between the nature of the underlying variables and to recognize the linkages between the Ising model and Ising QUBO.

For QUBO, we have seen that the linear term can be absorbed into the  $\mathbf{Q}$  matrix or the diagonal of  $\mathbf{Q}$  can be extracted and added to the linear term without altering the optimal solutions set. It is possible to discard the diagonal elements of  $\mathbf{A}$  from the Ising QUBO since these elements simply contribute a constant value to any feasible solution. Thus, it is customary to assume that diagonal entries of  $\mathbf{A}$  are zeros. Further, it is possible to reformulate Ising QUBO without the linear term [18, 38]. Define the  $(n + 1) \times (n + 1)$  matrix  $\bar{\mathbf{A}}$  where

$$\bar{a}_{ij} = \begin{cases} a_{ij}, & \text{if } 1 \leq i, j \leq n \\ \frac{1}{2}b_i & \text{for } j = n + 1, i = 1, 2, \dots, n \\ \frac{1}{2}b_j & \text{for } i = n + 1, j = 1, 2, \dots, n \\ 0 & \text{for } i = j = n + 1. \end{cases}$$

Now, consider the Ising QUBO,

$$\begin{aligned} \text{IQB: } & \text{Maximize } \mathbf{y}^T \bar{\mathbf{A}} \mathbf{y} \\ & \text{Subject to} \\ & \mathbf{y} \in \{-1, 1\}^{n+1}. \end{aligned}$$

Note that if  $\mathbf{y}$  is a solution to the Ising QUBO IQB then  $-\mathbf{y}$  is also a solution. Further, if both  $\mathbf{y}$  and  $-\mathbf{y}$  have the same objective function value in IQB. Thus, without loss of generality we can assume that  $y_{n+1} = 1$  in an optimal solution to IQB. Now, it can be verified that IQB is equivalent to the Ising QUBO.

The validity of the Ising model is not restricted to physics. Its relevance has been established in various other fields such as finance, biology, psychology, sociology etc. For example, the Ising model can be adapted to study human behavior by interpreting similarity between the behaviour of molecules and the behaviour of human, as argued by Galam [24]. Specific applications of the Ising QUBO from this point of view can be found in [49, 77]. Applications of QUBO and the Ising QUBO are covered in detail in Chap. 8.

### 1.3 Representations of the $\mathbf{Q}$ Matrix

Let  $f_{\mathbf{Q},\mathbf{c}}(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x}$  denote the objective function of the instance  $(\mathbf{Q}, \mathbf{c})$  of QUBO. We remove the suffix  $\mathbf{Q}, \mathbf{c}$  from  $f_{\mathbf{Q},\mathbf{c}}$  when the explicit consideration of  $\mathbf{Q}$  and  $\mathbf{c}$  is unimportant. An instance  $(\mathbf{Q}', \mathbf{c}')$  of QUBO is an *equivalent representation* of the instance  $(\mathbf{Q}, \mathbf{c})$  if

1.  $f_{\mathbf{Q},\mathbf{c}}(\mathbf{x}) = f_{\mathbf{Q}',\mathbf{c}'}(\mathbf{x})$  for all  $\mathbf{x} \in \{0, 1\}^n$
2.  $\mathbf{Q}$  and  $\mathbf{Q}'$  belongs to  $\mathbb{R}^{n \times n}$ ,  $\mathbf{c}$  and  $\mathbf{c}'$  belongs to  $\mathbb{R}^n$ .

Equivalent representations, although preserving optimality, could generate instances with different structural properties that may be exploited to obtain computational advantages. Let us now look at some simple and commonly used equivalent representations of QUBO [65]. More involved equivalent representations and corresponding applications in computing strong lower bounds are discussed in Chaps. 6 and 7.

*Remark 1.1*  $(\mathbf{Q}^T, \mathbf{c})$  and  $(\frac{1}{2}(\mathbf{Q} + \mathbf{Q}^T), \mathbf{c})$  are equivalent representations of  $(\mathbf{Q}, \mathbf{c})$ .

The proof of this remark is straightforward. Note that  $\frac{1}{2}(\mathbf{Q} + \mathbf{Q}^T)$  is a symmetric matrix. Thus, it is possible to assume without loss of generality that the matrix  $\mathbf{Q}$  in the definition of QUBO is symmetric and this assumption is required for some of the algorithms to solve the problem and in computing certain types of lower bounds. For any matrix  $\mathbf{Q} \in \mathbb{R}^{n \times n}$ ,  $\text{diagV}(\mathbf{Q})$  is the diagonal vector of  $\mathbf{Q}$ . i.e.  $\text{diagV}(\mathbf{Q})$  is the vector of size  $n$  and its  $i$ th element is  $q_{ii}$ .

*Remark 1.2 ([69])* If  $\mathbf{S}$  is a skew-symmetric matrix,  $\mathbf{D}$  is a diagonal matrix,  $\mathbf{Q}' = \mathbf{Q} + \mathbf{S} + \mathbf{D}$ , and  $\mathbf{c}' = \mathbf{c} - \text{diagV}(\mathbf{D})$ , then  $(\mathbf{Q}', \mathbf{c}')$  is an equivalent representation of  $(\mathbf{Q}, \mathbf{c})$ .

The proof of Remark 1.2 follows from the fact that  $\mathbf{x}^T\mathbf{S}\mathbf{x} = 0$  for any skew-symmetric matrix  $\mathbf{S}$  and  $x_i^2 = x_i$  for all binary variables  $x_i$ . Choosing  $\mathbf{S}$  and  $\mathbf{D}$  appropriately, we can get different standard representations of  $\mathbf{Q}$ . For example, choose  $\mathbf{D}$  such that  $d_{ii} = -q_{ii}$  for  $i = 1, \dots, n$  and choose  $\mathbf{S}$  such that

$$s_{ij} = \begin{cases} q_{ji} & \text{if } j > i \\ -q_{ij} & \text{if } j < i \\ 0 & \text{otherwise.} \end{cases}$$

Then, the resulting matrix  $\mathbf{Q}'$  is upper triangular with zeros on the diagonal. This is a common representation used in the QUBO literature and is also assumed in some of the chapters in this book. If we choose  $\mathbf{S}$  as the zero matrix and  $d_{ii} = M$  for  $i = 1, \dots, n$ , where  $M$  is a sufficiently large nonnegative number, the resulting matrix  $\mathbf{Q}'$  will be positive semidefinite. Consequently, without loss of generality one may assume  $\mathbf{Q}$  to be positive semidefinite. When  $\mathbf{Q}$  is symmetric and not positive semidefinite, choosing  $M$  to be the negative of the smallest eigenvalue of  $\mathbf{Q}$  is

sufficient to make  $\mathbf{Q}'$  to be positive semidefinite [32]. This makes the continuous relaxation of the objective function of QUBO a convex quadratic function. In a similar way, choosing  $M$  as a sufficiently small negative number, we can get a representation of QUBO where the  $\mathbf{Q}$  matrix is negative semidefinite and thereby making the continuous relaxation of the objective function of QUBO a concave quadratic function.

A QUBO  $(\mathbf{Q}, \mathbf{c})$  of size  $n$  can be represented as a QUBO  $(\mathbf{Q}', \mathbf{0})$  of size  $n + 1$  such that the row sum and column sums of  $\mathbf{Q}'$  are zeros. We call this *the zero sum representation*. Without loss of generality assume  $\mathbf{c} = \mathbf{0}$ . From the instance  $(\mathbf{Q}, \mathbf{0})$  construct the matrix  $\mathbf{Q}'$  as

$$q'_{ij} = \begin{cases} q_{ij} & \text{if } i, j \leq n \\ -\sum_{k=1}^n q_{ik} & \text{if } j = n + 1, i \leq n \\ -\sum_{k=1}^n q_{kj} & \text{if } i = n + 1, j \leq n \\ \sum_{k=1}^n \sum_{\ell=1}^n q_{k\ell} & \text{if } i = n + 1, j = n + 1 \end{cases}$$

It can be verified that the row and column sums of the matrix  $\mathbf{Q}'$  are zeros. Consequently, the sum of all elements of  $\mathbf{Q}'$  is also zero. If  $x_{n+1}$  is zero in an optimal solution for  $(\mathbf{Q}', \mathbf{0})$  then  $(\mathbf{Q}', \mathbf{0})$  is equivalent to  $(\mathbf{Q}, \mathbf{0})$ . Let  $\bar{\mathbf{x}} = \mathbf{e} - \mathbf{x}$  where  $\mathbf{e}$  is the all one vector in  $\mathbb{R}^{n+1}$ .

**Lemma 1.1** *If row and column sums of  $\mathbf{Q}'$  are zeros, then  $\mathbf{x}^T \mathbf{Q}' \mathbf{x} = \bar{\mathbf{x}}^T \mathbf{Q}' \bar{\mathbf{x}}$  for all  $\mathbf{x} \in \{0, 1\}^{n+1}$ .*

*Proof*  $\bar{\mathbf{x}}^T \mathbf{Q}' \bar{\mathbf{x}} = (\mathbf{e} - \mathbf{x})^T \mathbf{Q}' (\mathbf{e} - \mathbf{x}) = \mathbf{e}^T \mathbf{Q}' \mathbf{e} - \mathbf{x}^T \mathbf{Q}' \mathbf{e} - \mathbf{e}^T \mathbf{Q}' \mathbf{x} + \mathbf{x}^T \mathbf{Q}' \mathbf{x} = \mathbf{x}^T \mathbf{Q}' \mathbf{x}$ . The last equality follows from the fact that row and column sums of  $\mathbf{Q}'$  are zeros, which makes  $\mathbf{e}^T \mathbf{Q}' \mathbf{e} = 0$ ,  $\mathbf{x}^T \mathbf{Q}' \mathbf{e} = 0$  and  $\mathbf{e}^T \mathbf{Q}' \mathbf{x} = 0$ .  $\square$

From Lemma 1.1, if  $\mathbf{x}$  is an optimal solution to  $(\mathbf{Q}', \mathbf{0})$ , then  $\bar{\mathbf{x}}$  is also an optimal solution. Thus,  $x_{n+1} = 0$  in one of these optimal solutions of  $(\mathbf{Q}', \mathbf{0})$  and hence  $(\mathbf{Q}', \mathbf{0})$  and  $(\mathbf{Q}, \mathbf{0})$  are equivalent.

## 1.4 Some Equivalent Optimization Problems

We have seen that the problem of computing the maximum value principal minor of an  $n \times n$  matrix, the maximum weight induced subgraph problem, and the Ising QUBO are alternative ways of representing a QUBO. In this section, we discuss some additional problems that are equivalent to QUBO and these equivalent forms are used in some of the chapters to follow to derive structural properties and to develop solution algorithms.



### 1.4.1 QUBO as a Bilinear Program

In this subsection, we assume that  $\mathbf{Q}$  is symmetric and positive semidefinite. As discussed in the previous section, this assumption is without loss of generality. Then the continuous relaxation of QUBO is to

$$\begin{aligned} \text{QUBO(C):} \quad & \text{Maximize} \quad f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ & \text{Subject to:} \quad \mathbf{x} \in [0, 1]^n \end{aligned}$$

Since  $\mathbf{Q}$  is symmetric and positive semidefinite,  $f(\mathbf{x})$  is a convex function and hence there exists an optimal solution to QUBO(C) which at an extreme point of the hypercube  $[0, 1]^n$ . Now consider the *hypercube bilinear program*

$$\begin{aligned} \text{HBLP:} \quad & \text{Maximize} \quad \eta(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{Q} \mathbf{y} + \frac{1}{2} \mathbf{c}^T \mathbf{x} + \frac{1}{2} \mathbf{c}^T \mathbf{y} \\ & \text{Subject to:} \quad \mathbf{x} \in [0, 1]^n, \mathbf{y} \in [0, 1]^n \end{aligned}$$

**Lemma 1.2 ([51])** *There exists an optimal solution  $(\mathbf{x}^*, \mathbf{y}^*)$  to HBLP such that both  $\mathbf{x}^*$  and  $\mathbf{y}^*$  are extreme points of the hypercube  $[0, 1]^n$ .*

**Proof** Suppose that  $(\mathbf{x}^0, \mathbf{y}^0)$  is an optimal solution to HBLP. Now fix  $\mathbf{y} = \mathbf{y}^0$  in HBLP and let  $\bar{\mathbf{x}}$  be an optimal extreme point solution of the resulting linear program. Then  $(\bar{\mathbf{x}}, \mathbf{y}^0)$  is an optimal solution to HBLP. Next fix  $\mathbf{x} = \bar{\mathbf{x}}$  in HBLP and let  $\bar{\mathbf{y}}$  be an optimal extreme point solution of the resulting linear program. Then  $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$  is an optimal solution to HBLP and this completes the proof.  $\square$

Let us now prove a property of a symmetric positive semidefinite matrix which is used in the proof of the theorem that follows.

**Lemma 1.3** *If  $\mathbf{Q}$  is a symmetric positive semidefinite matrix such that  $\mathbf{x}^T \mathbf{Q} \mathbf{x} = 0$  then  $\mathbf{Q} \mathbf{x} = \mathbf{0}$ .*

**Proof** Since  $\mathbf{Q}$  is symmetric and positive semidefinite, there exists a matrix  $\mathbf{B}$  such that  $\mathbf{Q} = \mathbf{B}^T \mathbf{B}$ . Then,  $\mathbf{x}^T \mathbf{Q} \mathbf{x} = \mathbf{x}^T \mathbf{B}^T \mathbf{B} \mathbf{x} = (\mathbf{B} \mathbf{x})^T (\mathbf{B} \mathbf{x}) = \|\mathbf{B} \mathbf{x}\|^2$ . Thus,  $\mathbf{x}^T \mathbf{Q} \mathbf{x} = 0$  implies  $\mathbf{B} \mathbf{x} = \mathbf{0}$  and hence  $\mathbf{Q} \mathbf{x} = \mathbf{B}^T \mathbf{B} \mathbf{x} = \mathbf{0}$ .  $\square$

We now show that QUBO is equivalent to HBLP.

**Theorem 1.1 ([50, 51])** *If  $(\mathbf{x}^*, \mathbf{y}^*)$  is an optimal extreme point solution of HBLP then both  $\mathbf{x}^*$  and  $\mathbf{y}^*$  are optimal solutions of QUBO(C). Conversely, if  $\mathbf{x}^*$  is an optimal extreme point solution of QUBO(C) then  $(\mathbf{x}^*, \mathbf{x}^*)$  is an optimal solution to HBLP.*

**Proof** Let  $(\mathbf{x}^*, \mathbf{y}^*)$  be an optimal extreme point solution of HBLP and  $\mathbf{x}^0$  be an optimal extreme point solution of QUBO(C). Then,

$$f(\mathbf{x}^0) \geq f(\mathbf{x}^*) \text{ and } f(\mathbf{x}^0) \geq f(\mathbf{y}^*). \quad (1.1)$$

Further,

$$\begin{aligned}\eta(\mathbf{x}^*, \mathbf{y}^*) &= \max\{\eta(\mathbf{x}, \mathbf{y}) : \mathbf{x}, \mathbf{y} \in [0, 1]^n\} \\ &\geq \max\{\eta(\mathbf{x}, \mathbf{x}) : \mathbf{x} \in [0, 1]^n\} = f(\mathbf{x}^0)\end{aligned}\quad (1.2)$$

Since  $(\mathbf{x}^*, \mathbf{y}^*)$  is an optimal solution to HBLP,

$$\eta(\mathbf{x}^*, \mathbf{y}^*) - \eta(\mathbf{x}^*, \mathbf{x}^*) \geq 0 \text{ and } \eta(\mathbf{x}^*, \mathbf{y}^*) - \eta(\mathbf{y}^*, \mathbf{y}^*) \geq 0 \quad (1.3)$$

But

$$\eta(\mathbf{x}^*, \mathbf{y}^*) - \eta(\mathbf{x}^*, \mathbf{x}^*) = (\mathbf{x}^*)^T \mathbf{Q}(\mathbf{y}^* - \mathbf{x}^*) + \frac{1}{2} \mathbf{c}^T (\mathbf{y}^* - \mathbf{x}^*) \text{ and} \quad (1.4)$$

$$\eta(\mathbf{x}^*, \mathbf{y}^*) - \eta(\mathbf{y}^*, \mathbf{y}^*) = (\mathbf{y}^*)^T \mathbf{Q}(\mathbf{x}^* - \mathbf{y}^*) + \frac{1}{2} \mathbf{c}^T (\mathbf{x}^* - \mathbf{y}^*) \quad (1.5)$$

From (1.3), (1.4) and (1.5) we have,

$$(\mathbf{x}^*)^T \mathbf{Q}(\mathbf{y}^* - \mathbf{x}^*) + \frac{1}{2} \mathbf{c}^T (\mathbf{y}^* - \mathbf{x}^*) \geq 0 \text{ and} \quad (1.6)$$

$$(\mathbf{y}^*)^T \mathbf{Q}(\mathbf{x}^* - \mathbf{y}^*) + \frac{1}{2} \mathbf{c}^T (\mathbf{x}^* - \mathbf{y}^*) \geq 0. \quad (1.7)$$

Adding (1.6) and (1.7), we get  $(\mathbf{x}^* - \mathbf{y}^*)^T \mathbf{Q}(\mathbf{x}^* - \mathbf{y}^*) \leq 0$ . Since  $\mathbf{Q}$  is symmetric and positive semidefinite, from Lemma 1.3,  $\mathbf{Q}(\mathbf{x}^* - \mathbf{y}^*) = 0$ . Substituting this in (1.6) and (1.7) we get  $\mathbf{c}^T (\mathbf{x}^* - \mathbf{y}^*) = 0$ . Thus from (1.4) and (1.5) we have

$$\eta(\mathbf{x}^*, \mathbf{y}^*) = \eta(\mathbf{x}^*, \mathbf{x}^*) = \eta(\mathbf{y}^*, \mathbf{y}^*).$$

But  $\eta(\mathbf{x}^*, \mathbf{x}^*) = f(\mathbf{x}^*)$  and  $\eta(\mathbf{y}^*, \mathbf{y}^*) = f(\mathbf{y}^*)$ . Thus,

$$\eta(\mathbf{x}^*, \mathbf{y}^*) = f(\mathbf{x}^*) = f(\mathbf{y}^*) \quad (1.8)$$

From (1.1) and (1.8),  $\eta(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}^0)$  and the result follows in view of (1.2) and (1.8).  $\square$

### 1.4.2 The Maximum Cut Problem and QUBO

Let  $G = (V, E)$  be a graph with  $V = \{1, 2, \dots, n\}$  and for each edge  $(i, j) \in E$  a weight  $w_{ij}$  is prescribed. For any  $S \subseteq V$ , the pair  $(S, V \setminus S)$  is called a *cut* in  $G$ .

The weight of the cut  $(S, V \setminus S)$  is

$$w(S, V \setminus S) = \sum_{i \in S, j \in V \setminus S, (i, j) \in E} w_{ij}.$$

In general, our definition of a cut allows the possibility that  $S = \emptyset$  or  $S = V$  and in either case  $w(S, V \setminus S) = 0$ . The *minimum cut* problem seeks a cut  $(S, V \setminus S)$  with minimum  $w(S, V \setminus S)$  value. When  $w_{ij} \geq 0$  for all  $(i, j) \in E$ , we normally do not permit  $S = \emptyset$  or  $V$  for the minimum cut problem since otherwise the trivial solution  $S = \emptyset$  is optimal. Likewise, the *maximum cut problem* seeks a cut  $(S, V \setminus S)$  with maximum  $w(S, V \setminus S)$  value. When  $w_{ij} \geq 0$  for all  $(i, j) \in E$ , the minimum cut problem is solvable in polynomial time [44] but the maximum cut problem is NP-hard [23]. When  $w_{ij}$  is allowed to take positive and negative values, both minimum cut and maximum cut problems are NP-hard. We now observe that QUBO and the maximum cut problem are essentially the same, presented under different frameworks [7, 33, 38].

Let  $\mathbf{x} \in \{0, 1\}^n$  and  $\bar{\mathbf{x}} = \mathbf{e} - \mathbf{x}$  where  $\mathbf{e}$  is the all-one vector in  $\mathbb{R}^n$ . An  $\mathbf{x} \in \{0, 1\}^n$  is an *incidence vector* of  $S \subseteq V$  if and only if  $x_i = 1$  for all  $i \in S$ . Let  $\mathbf{Q}$  be the *weighted incidence matrix* of  $G$ . Define  $q_{ij} = q_{ji} = w_{ij}$  if  $(i, j) \in E$  (i.e.  $i$  and  $j$  are adjacent in  $G$ ) and zero, otherwise. Note that  $\mathbf{Q}$  is symmetric. Then, for any cut  $(S, \bar{S})$  (with  $\bar{S} = V \setminus S$ ) and the incidence vector  $\mathbf{x}$  of  $S$ , we have  $w(S, V \setminus S) = \mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}}$ . For example, consider the graph below with the associated edge weights and the corresponding matrix  $\mathbf{Q}$  (Fig. 1.1).

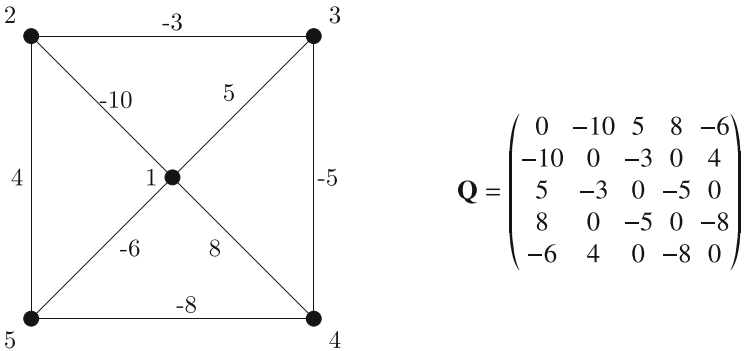


Fig. 1.1 An instance of WMCP

Now, choose the cut  $(S, \bar{S})$  where  $S = \{1, 4\}$  and the incidence vector  $\mathbf{x} = (1, 0, 0, 1, 0)^T$  associated with  $S$ . Then,  $w(S, \bar{S}) = -24 = \mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}}$ .

Thus, the maximum cut problem can be written as

$$\max_{S \subseteq V} w(S, V \setminus S) = \max_{\mathbf{x} \in \{0,1\}^n} \mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}} \quad (1.9)$$

For calculating  $\mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}}$ , the diagonal elements of  $\mathbf{Q}$  are irrelevant. For any  $n \times n$  matrix  $\mathbf{Q}$ ,

$$\mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}} = \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i \bar{x}_j = \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i (1 - x_j) = \sum_{i=1}^n r_i x_i - \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j \quad (1.10)$$

where  $r_i = \sum_{j=1}^n q_{ij}$ . Define the  $n \times n$  matrix  $\hat{\mathbf{Q}} = (\hat{q}_{ij})$  where

$$\hat{q}_{ij} = \begin{cases} -q_{ij} & \text{if } i \neq j \\ r_i - q_{ii} & \text{if } i = j. \end{cases}$$

Then  $\mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}} = \mathbf{x}^T \hat{\mathbf{Q}} \mathbf{x}$ , establishing that the maximum cut problem can be formulated as a QUBO. We now observe that any instance, say  $(\mathbf{Q}, \mathbf{c})$ , of QUBO can be formulated as a maximum cut problem. To see this, from Eq. (1.10),

$$\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} = \mathbf{r}^T \mathbf{x} - \mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}} + \mathbf{c}^T \mathbf{x} = (\mathbf{r} + \mathbf{c})^T \mathbf{x} - \mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}} \quad (1.11)$$

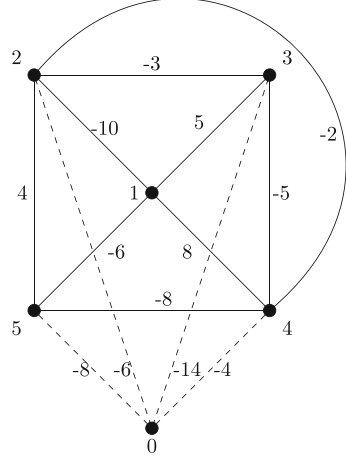
where  $\mathbf{r} = (r_1, r_2, \dots, r_n)$ .

Now, consider the graph  $G' = (V', E')$  where  $V' = \{0, 1, 2, \dots, n\}$ . The edge set  $E' = E \cup E^0$  where  $E = \{(i, j) : q_{ij} \neq 0; i, j = 1, 2, \dots, n, i \neq j\}$  and  $E^0 = \{(0, i) : r_i + c_i \neq 0\}$ . Now, define the weight  $w_{ij}$  of the edge  $(i, j)$  as  $-q_{ij}$  if  $(i, j) \in E$  and  $-(r_j + c_j)$  for  $(0, j) \in E^0$ . Let  $\alpha = \sum_{i=1}^n (r_i + c_i)$ . For any cut  $(S, \bar{S})$  in  $G'$ , without loss of generality assume that the node  $0 \in S$  and consider the binary variables  $x_1, x_2, \dots, x_n$  with  $x_i = 1$  if and only if  $i \in S$ . Let  $w(S, \bar{S})$  be the value of the cut  $(S, \bar{S})$  in  $G'$  with the edge weight function  $w$ . Then, it can be verified that  $w(S, \bar{S}) + \alpha = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x}$ . Thus, any instance  $(\mathbf{Q}, \mathbf{c})$  of the QUBO with  $n$  variables can be formulated as a maximum weight cut problem on a graph with  $n + 1$  nodes.

We now illustrate this construction using the following example. Let

$$\mathbf{Q} = \begin{pmatrix} 0 & 10 & -5 & -8 & 6 \\ 10 & 0 & 3 & 2 & -4 \\ -5 & 3 & 0 & 5 & 0 \\ -8 & 2 & 5 & 0 & 8 \\ 6 & -4 & 0 & 8 & 0 \end{pmatrix} \text{ and } \mathbf{c} = \begin{pmatrix} -3 \\ -5 \\ 11 \\ -3 \\ -2 \end{pmatrix}$$

**Fig. 1.2** The instance of WMCP equivalent to  $(\mathbf{Q}, \mathbf{c})$



Then,  $(\mathbf{r} + \mathbf{c})^T = (0, 6, 14, 4, 8)$  and  $\alpha = 32$ . The graph  $G'$  constructed following the procedure discussed above, is given in Fig. 1.2. The numbers on the edges are the corresponding weights. Edges with weight zero are removed from the graph.

The QUBO formulation of the maximum cut problem is also evident from the sum of squares formulation of the maximum cut problem given by

$$\max_{S \subseteq V} w(S, V \setminus S) = \max_{\mathbf{x} \in \{0,1\}^n} \sum_{(i,j) \in E} w_{ij} (x_i - x_j)^2.$$

Another popular formulation of the maximum cut problem is obtained by using variables that take values  $-1$  or  $1$ . Define  $y_i = 1$  if  $i \in S$  and  $y_i = -1$  if  $i \in V \setminus S$ . Assume that the edges of the graph  $G$  are labelled such that  $(i, j) \in E$  implies  $i < j$  and  $w_{ij} = 0$  if  $(i, j) \notin E$ . Consider the symmetric matrix  $\mathbf{W}$ . Then,

$$\begin{aligned} w(S, V \setminus S) &= \sum_{\substack{i \in S, j \in V \setminus S \\ (i,j) \in E}} w_{ij} = \frac{1}{2} \sum_{i=1}^n \sum_{j=i+1}^n w_{ij} (1 - y_i y_j) \\ &= \frac{1}{4} \sum_{i=1}^n \sum_{j=1}^n w_{ij} (1 - y_i y_j) = \frac{1}{4} \sum_{i=1}^n \sum_{j=1}^n w_{ij} \left( \frac{y_i^2 + y_j^2}{2} - y_i y_j \right) \\ &= \frac{1}{4} \left( - \sum_{i=1}^n \sum_{j=1}^n w_{ij} y_i y_j + \frac{1}{2} \sum_{i=1}^n \left( \sum_{j=1}^n w_{ij} \right) y_i^2 + \frac{1}{2} \sum_{j=1}^n \left( \sum_{i=1}^n w_{ij} \right) y_j^2 \right) \\ &= \frac{1}{4} \left( - \sum_{i=1}^n \sum_{j=1}^n w_{ij} y_i y_j + \sum_{i=1}^n \left( \sum_{j=1}^n w_{ij} \right) y_i^2 \right) = \frac{1}{4} \mathbf{y}^T \mathbf{L} \mathbf{y} \end{aligned}$$

where  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  and  $\mathbf{D}$  is a diagonal matrix such that  $d_{ii} = \sum_{j=1}^n w_{ij}$  assuming  $w_{ij} = 0$  for  $(i, j) \notin E$ . Thus, the maximum cut problem can be formulated as the Ising QUBO  $(\frac{1}{4}\mathbf{L}, \mathbf{0})$ .

Let  $G = (V, E)$  be a weighted graph with edge weight  $w_{ij}$  for each  $(i, j) \in E$  and  $V = \{1, 2, \dots, n\}$ . Choose  $w_{ij} = 0$  if  $(i, j) \notin E$  and define the weight matrix  $\mathbf{W}$  as the  $n \times n$  matrix with its  $(i, j)$ th element  $w_{ij}$ . Let  $\mathbf{D}$  be the weighted degree matrix which is the diagonal matrix with  $d_{ii} = \sum_{j=1}^n w_{ij}$ . Then the weighted Laplacian matrix of  $G$  is  $L = D - W$ . The matrix  $L$  is positive semidefinite and the row and columns sums of the matrix is zero. Note that the matrix  $L$  constructed in our reduction from the maximum weight cut problem to the Ising QUBO is precisely the Laplacian matrix of  $G$ .

In fact, any Ising QUBO can be written as a maximum cut problem as well. To see this, consider an instance  $(\mathbf{A}, \mathbf{b})$  of the Ising QUBO in  $n$  variables. Without loss of generality, assume that  $\mathbf{A}$  is symmetric and  $\mathbf{b}$  is the zero vector. Now, consider the graph  $G = (V, E)$  where  $V = \{1, 2, \dots, n\}$  and  $E = \{(i, j) : a_{ij} \neq 0\}$ . Choose the weight of the edge  $(i, j) \in E$  as  $-a_{ij}$  and let  $\mathbf{L}$  be the corresponding weighted Laplacian matrix. Then,  $\mathbf{A} = \mathbf{L} + \mathbf{D}$  where  $\mathbf{D}$  is a diagonal matrix with  $d_{ii} = -l_{ii} + a_{ii}$ . For any  $\mathbf{y} \in \{-1, 1\}^n$ ,

$$\mathbf{y}^T \mathbf{A} \mathbf{y} = \mathbf{y}^T \mathbf{L} \mathbf{y} + \mathbf{y}^T \mathbf{D} \mathbf{y} = \mathbf{y}^T \mathbf{L} \mathbf{y} + \text{tr}(\mathbf{D})$$

where  $\text{tr}(\mathbf{D})$  is the trace of  $\mathbf{D}$  which is a constant. Thus an optimal solution to the Ising QUBO  $(\mathbf{A}, \mathbf{0})$  is obtained by solving the Ising QUBO  $(\mathbf{L}, \mathbf{0})$  which is precisely the maximum cut problem on  $G$  with weights  $-a_{ij}$  for  $(i, j) \in E$ .

### 1.4.3 Equivalence with the Stable Set Problem

We have seen that the maximum weight stable set problem can be formulated as a QUBO. We now show that any instance of QUBO can be formulated as a maximum weight stable set problem. Our discussion here follows the paper by Hammer, Hansen and Simeone [35]. First, let us rewrite the objective function of a QUBO where all the quadratic terms have non-negative coefficients. After rewriting, the quadratic terms could involve both the original binary variables and their complements. Recall that

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} = \sum_{(i,j) \in P} q_{ij} x_i x_j + \sum_{(i,j) \in N} q_{ij} x_i x_j + \sum_{i=1}^n (q_{ii} + c_i) x_i,$$

where  $P = \{(i, j) : q_{ij} > 0, i \neq j\}$  and  $N = \{(i, j) : q_{ij} < 0, i \neq j\}$ . Also, let  $N_j = \{i : (i, j) \in N\}$  and  $\rho_j = \sum_{i \in N_j} q_{ij}$ . Let  $\bar{x}_i = 1 - x_i$  be the complement of

$x_i$ . i.e.  $x_i \in \{0, 1\}$  if and only if  $\bar{x}_i \in \{0, 1\}$ . Now,

$$\begin{aligned}
f(\mathbf{x}) &= \sum_{(i,j) \in P} q_{ij} x_i x_j + \sum_{(i,j) \in N} q_{ij} (1 - \bar{x}_i) x_j + \sum_{i=1}^n (q_{ii} + c_i) x_i \\
&= \sum_{(i,j) \in P} q_{ij} x_i x_j + \sum_{(i,j) \in N} (-q_{ij}) \bar{x}_i x_j + \sum_{(i,j) \in N} q_{ij} x_j + \sum_{i=1}^n (q_{ii} + c_i) x_i, \\
&= \sum_{(i,j) \in P} q_{ij} x_i x_j + \sum_{(i,j) \in N} (-q_{ij}) \bar{x}_i x_j + \sum_{i=1}^n (q_{ii} + c_i + \rho_i) x_i \quad (1.12) \\
&= h(\mathbf{x}), \text{ say.}
\end{aligned}$$

Note that the coefficients of  $x_i x_j$  and  $\bar{x}_i x_j$  in  $h(\mathbf{x})$  are positive. This representation  $h(\mathbf{x})$  of  $f(\mathbf{x})$  is sometimes referred to as Rhys form [35] indicating an early work on a special case of QUBO [72] and is one of the posiform representations of  $f(\mathbf{x})$  [7, 8, 35] when viewed as a pseudo-Boolean function. For interesting properties of posiforms of the QUBO objective function, we refer to Chap. 5. Note that maximizing  $f(\mathbf{x})$  over  $x_i \in \{0, 1\}$  is equivalent to maximizing  $h(\mathbf{x})$  over  $x_i, \bar{x}_i \in \{0, 1\}$ . We now show that the problem of maximizing  $h(\mathbf{x})$  over  $x_i, \bar{x}_i \in \{0, 1\}$  is equivalent to solving a maximum weight stable set problem. The proof is based on a construction given in [35].

For each  $(i, j) \in P \cup N$  introduce a node  $v_{ij}$  with weight  $|q_{ij}|$ . Also, for  $i = 1, 2, \dots, n$ , introduce two nodes  $i$  and  $i'$  with respective weights  $w_i = q_{ii} + c_i + \rho_i + M$  and  $w_{i'} = M$ , where  $M$  is a large positive number. Now introduce the edges  $(v_{ij}, i')$ ,  $(v_{ij}, j')$  for each  $(i, j) \in P$ , the edges  $(v_{ij}, i)$ ,  $(v_{ij}, j')$  for each  $(i, j) \in N$ . Finally connect each node  $i$  with node  $i'$  by an edge  $(i, i')$ , for  $i = 1, 2, \dots, n$ . Let  $G' = (V', E')$  be the resulting graph.

Define the product node variables  $y_{ij}$  where

$$y_{ij} = \begin{cases} 1 & \text{if vertex } v_{ij} \text{ is selected} \\ 0 & \text{if vertex } v_{ij} \text{ is not selected.} \end{cases}$$

Similarly, define the selection variables for nodes  $i$  and  $i'$  as

$$x_i = \begin{cases} 1 & \text{if vertex } i \text{ is selected} \\ 0 & \text{if vertex } i \text{ is not selected} \end{cases} \quad \text{and} \quad x'_i = \begin{cases} 1 & \text{if vertex } i' \text{ is selected} \\ 0 & \text{if vertex } i' \text{ is not selected} \end{cases}$$

Then, the standard integer programming formulation of the maximum weight stable set problem (MWSSP) on  $G'$  is

$$\text{SSIP: Maximize } \sum_{(i,j) \in P} q_{ij} y_{ij} + \sum_{(i,j) \in N} -q_{ij} y_{ij} + \sum_{i=1}^n (q_{ii} + c_i + r_i + M) x_i + \sum_{i=1}^n M x'_i$$

Subject to:

$$\left. \begin{array}{l} y_{ij} + x'_i \leq 1 \\ y_{ij} + x'_j \leq 1 \end{array} \right\} \text{for } (i, j) \in P \quad (1.13)$$

$$\left. \begin{array}{l} y_{ij} + x_i \leq 1 \\ y_{ij} + x'_j \leq 1 \end{array} \right\} \text{for } (i, j) \in N \quad (1.14)$$

$$x_i + x'_i \leq 1, i = 1, 2, \dots, n, \quad (1.15)$$

$$y_{ij}, x_i, x'_i \in \{0, 1\}. \quad (1.16)$$

**Theorem 1.2** *If  $(\mathbf{y}, \mathbf{x}, \mathbf{x}')$  is an optimal solution to the SSIP defined above, then  $\mathbf{x}$  is an optimal solution to the QUBO with objective function  $f(\mathbf{x})$ .*

**Proof** Recall that optimizing  $f(\mathbf{x})$  over  $\mathbf{x} \in \{0, 1\}^n$  is equivalent to optimizing  $h(\mathbf{x})$  over  $\mathbf{x}, \bar{\mathbf{x}} \in \{0, 1\}^n$ . Let  $\phi(\mathbf{y}, \mathbf{x}, \mathbf{x}')$  denote the objective function of SSIP. Since  $M$  is large enough,  $x'_i = 1 - x_i$  in every optimal solution to SSIP on  $G'$  for otherwise, a better solution can be obtained. The inequalities (1.13) imply that  $y_{ij} \leq \min\{x_i, x_j\}$  in every optimal solution and hence  $y_{ij} = 0$  if at least one of  $x_i$  or  $x_j$  is zero. If  $x_i = x_j = 1$  then,  $y_{ij} = 1$  in an optimal solution since  $q_{ij} > 0$  for  $(i, j) \in P$ . Thus  $y_{ij} = x_i x_j$  in an optimal solution of SSIP on  $G'$ . Similarly, we can show that  $y_{ij} = x'_i x'_j$  in an optimal solution. Noting that  $x'_i = 1 - x_i = \bar{x}_i$ , we have

$$\phi(\mathbf{y}, \mathbf{x}, \mathbf{x}') = h(\mathbf{x}) + nM \quad (1.17)$$

for every optimal solution  $(\mathbf{y}, \mathbf{x}, \mathbf{x}')$  of SSIP on  $G'$ .

Conversely, for any  $\mathbf{x} \in \{0, 1\}^n$  define  $y_{ij} = x_i x_j$  for all  $(i, j) \in P$ ,  $y_{ij} = (1 - x_i)x_j$  for all  $(i, j) \in N$ , and  $x'_i = \bar{x}_i = 1 - x_i$ . The solution  $(\mathbf{y}, \mathbf{x}, \mathbf{x}')$  constructed above is indeed a feasible solution to SSIP on  $G'$  satisfying

$$\phi(\mathbf{y}, \mathbf{x}, \mathbf{x}') = h(\mathbf{x}) + nM \quad (1.18)$$

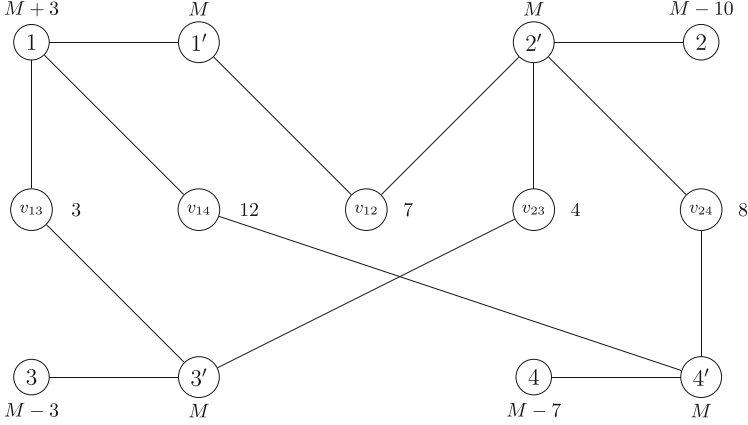
and the result follows.  $\square$

*Example 1.1* Consider the QUBO with objective function

$$f(\mathbf{x}) = 7x_1x_2 - 3x_1x_3 - 12x_1x_4 + 4x_2x_3 + 8x_2x_4 + 3x_1 - 10x_2 + 5x_4.$$

Then,  $P = \{(1, 2), (2, 3), (2, 4)\}$  and  $N = \{(1, 3), (1, 4)\}$ . The cost matrix  $\mathbf{Q}$ , the sets  $N_i$ ,  $i = 1, 2, 3, 4$ , and the values  $\rho_i$  and  $q_{ii} + c_i + \rho_i$ , for  $i = 1, 2, 3, 4$  are given





**Fig. 1.3** The graph for the maximum weight stable set problem constructed from the QUBO in Example 1.1.

below

$$\mathbf{Q} = \begin{pmatrix} 0 & 7 & -3 & -12 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$i$	$N_i$	$\rho_i$	$q_{ii} + \rho_i + c_i$
1	$\emptyset$	0	3
2	$\emptyset$	0	-10
3	{1}	-3	-3
4	{1}	-12	-7

Now construct graph  $G' = (V, E)$  for the stable set problem as discussed above. The resulting graph is given in Fig. 1.3. In the figure, the numbers shown outside the nodes represent the weight of the node.

It can be verified that  $x_1 = 0, x_2 = x_3 = x_4 = 1$  is an optimal solution to the QUBO with optimal objective function value 7. An optimal solution to the constructed maximum weight stable set problem is  $S = \{v_{13}, 1', v_{14}, 4, v_{23}, v_{24}, 2, 3\}$  with value equal to  $7 + 4M$ . Also, the QUBO solution recovered from  $S$  is  $x_1 = 0, x_2 = x_3 = x_4 = 1$  and this is optimal with optimal objective function value 7.

Considering the equivalence between the maximum weight stable set problem and the maximum weight clique problem, we can see that QUBO is equivalent to the maximum weight clique problem as well.

### 1.4.4 QUBO and the Generalized Stable Set Problem

Let  $G = (V, E)$  be a graph and  $c_i$  be the weight associated with vertex  $i \in V$ . Also, for each edge  $(i, j) \in E$  a cost  $q_{ij}$  is given. Recall that a stable set in  $G$  is

a subset  $S$  of  $V$  such that no two nodes in  $S$  are adjacent in  $G$ . In the *generalized stable set problem* [37, 39, 65], we relax the restriction that no two vertices in  $S$  are adjacent by imposing a penalty. That is, if two vertices in  $S$  are adjacent in  $G$ , a penalty  $q_{ij}$  is incurred. The generalized stable set problem on a complete graph is precisely a QUBO. If  $G$  is not complete, we can define  $q_{ij} = 0$  for  $(i, j) \notin E$  to yield a QUBO formulation. The generalized stable set problem and some of its variations are studied by many authors [4, 37, 39, 65].

### 1.4.5 Quadratic Pseudo-Boolean Optimization

A quadratic function in variables  $X = \{x_1, x_2, \dots, x_n, \bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$  where  $x_i, \bar{x}_i \in \{0, 1\}$  is called a *quadratic pseudo-Boolean function* (quadratic PBF) [8]. The objective function  $\mathbf{x}^T \mathbf{Q} \bar{\mathbf{x}}$  of the maximum-cut problem is a quadratic PBF and so is the objective function  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x}$  of QUBO. A quadratic PBF is in *posiform* if all associated coefficients (except possibly the constant term) are positive. A posiform is *homogeneous* if the associated constant is zero. A homogeneous posiform  $\zeta$  can be written as

$$\zeta(\mathbf{x}, \bar{\mathbf{x}}) = \sum_{i < j} (a_{ij} x_i x_j + b_{ij} \bar{x}_i x_j + c_{ij} x_i \bar{x}_j + d_{ij} \bar{x}_i \bar{x}_j) + \sum_{i=1}^n (\alpha_i x_i + \beta_i \bar{x}_i)$$

The function  $h(\mathbf{x})$  given in Eq. (1.12) has positive coefficients for quadratic terms but the associated linear terms still have positive and negative coefficients. Note that  $-x_i = \bar{x}_i - 1$  and  $-\bar{x}_i = x_i - 1$ . Using these transformations, we can construct a posiform  $h^1(\mathbf{x}, \bar{\mathbf{x}})$  such that  $f(\mathbf{x}) = \alpha_1 + h^1(\mathbf{x}, \bar{\mathbf{x}})$  for all  $\mathbf{x} \in \{0, 1\}$  and the coefficients of  $h^1$  are positive. Likewise, we can construct a posiform  $h^2(\mathbf{x}, \bar{\mathbf{x}})$  such that  $f(\mathbf{x}) = \alpha_2 - h^2(\mathbf{x}, \bar{\mathbf{x}})$  for all  $\mathbf{x} \in \{0, 1\}^n$ .

**Lemma 1.4 ([35])**  $f(\mathbf{x}) = \alpha_1 - \zeta^1(\mathbf{x}, \bar{\mathbf{x}}) = \alpha_2 + \zeta^2(\mathbf{x}, \bar{\mathbf{x}})$ , where  $\zeta^1$  and  $\zeta^2$  are in homogeneous posiform and  $\alpha_1$  and  $\alpha_2$  are constants.

**Proof** Without loss of generality, we assume that the matrix  $\mathbf{Q}$  is lower triangular and the diagonal entries are zero (see Sect. 1.3). Thus

$$f(\mathbf{x}) = \sum_{i < j} q_{ij} x_i x_j + \sum_{i=1}^n c_i x_i.$$

Note that when  $x_i$  and  $x_j$  are binary variables,

$$x_i x_j = \frac{1}{2} (x_i x_j + \bar{x}_i \bar{x}_j + x_i + x_j - 1)$$

and

$$-x_i x_j = \frac{1}{2} (x_i \bar{x}_j + \bar{x}_i x_j - x_i - x_j)$$

Substitute these values in  $f(\mathbf{x})$  and simplify. Then, in the linear terms, if coefficient of  $x_i$  or  $\bar{x}_i$  is negative, make a substitution using the equalities  $-x_i = \bar{x}_i - 1$  and  $-\bar{x}_i = x_i - 1$  and we get the first equality in the lemma. An analogous proof can be given for the second part also.  $\square$

The homogeneous posiforms  $\zeta^1$  and  $\zeta^2$  are not unique. The function  $h(x)$  constructed in Eq. (1.12) have positive coefficients for all quadratic terms and can be converted into posiform by applying the transformation for linear terms used in the proof above. There are many other ways to obtain the required posiforms in the lemma above.

Since  $\zeta^1(\mathbf{x}, \bar{\mathbf{x}})$  and  $\zeta^2(\mathbf{x}, \bar{\mathbf{x}})$  are positive for all  $\mathbf{x} \in \{0, 1\}^n$ , we have

$$\alpha_2 \leq f(\mathbf{x}) \leq \alpha_1 \text{ for all } \mathbf{x} \in \{0, 1\}^n.$$

Let  $\alpha$  be the smallest value real number for which there exist a homogeneous posiform  $\zeta(\mathbf{x}, \bar{\mathbf{x}})$  such that  $f(\mathbf{x}) = \alpha - \zeta(\mathbf{x}, \bar{\mathbf{x}})$ . Then,  $f(\mathbf{x}) \leq \alpha$  for all  $\mathbf{x} \in \{0, 1\}^n$  and  $\zeta(\mathbf{x}, \bar{\mathbf{x}})$  is called the complement of  $f(\mathbf{x})$ . Thus, by computing the complement of  $f(\mathbf{x})$  we get an immediate upper bound on  $f(\mathbf{x})$ . For interesting discussions on computing the best upper bound of this type and other related results, we refer to [35].

## 1.5 Roof Duality

Consider the objective function of QUBO in Rhs form  $h(\mathbf{x})$ . i.e.,

$$\sum_{(i,j) \in P} q_{ij} x_i x_j + \sum_{(i,j) \in N} (-q_{ij}) \bar{x}_i x_j + \sum_{i=1}^n l_i x_i.$$

Without loss of generality assume that  $q_{ij} = 0$  if  $i > j$ . Recall that  $P = \{(i, j) : q_{ij} x_i x_j \in h(\mathbf{x}), i \neq j, q_{ij} \neq 0\}$  and  $N = \{(i, j) : q_{ij} \bar{x}_i x_j \in h(\mathbf{x}), i \neq j, q_{ij} \neq 0\}$ . The set inclusion notation  $q_{ij} x_i x_j \in h(\mathbf{x})$  used here simply indicates that  $q_{ij} x_i x_j$  is a term in  $h(\mathbf{x})$ .

For  $q_{ij} \geq 0$  and  $x_i, x_j \in \{0, 1\}$ ,  $q_{ij} x_i x_j \leq \lambda_{ij} x_i + \mu_{ij} x_j$  and  $q_{ij} \bar{x}_i x_j \leq \lambda_{ij} (1 - x_i) + \mu_{ij} x_j$  for all  $\lambda_{ij}, \mu_{ij} = q_{ij}, \lambda_{ij} \geq 0, \mu_{ij} \geq 0$ . A *roof* of  $h(\mathbf{x})$  is a linear 0-1 function of the form [35]

$$r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \sum_{i=1}^n l_i x_i + \sum_{(i,j) \in P} (\lambda_{ij} x_i + \mu_{ij} x_j) + \sum_{(i,j) \in N} (\lambda_{ij} (1 - x_i) + \mu_{ij} x_j) \quad (1.19)$$

where  $\lambda_{ij} + \mu_{ij} = q_{ij}$  and  $\lambda_{ij} \geq 0, \mu_{ij} \geq 0$  for all  $(i, j) \in P \cup N$ . By construction

$$h(\mathbf{x}) \leq r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \text{ for all } \mathbf{x} \in \{0, 1\}^n.$$

Let  $G = (V, E)$  be a graph with  $V = \{1, 2, \dots, n\}$  and  $E = P \cup N$ . For each  $i \in V$ , define the set of outgoing and incoming arcs at node  $i$  by

$$O(i) = \{j \in V : (i, j) \in E\} \text{ and } I(i) = \{k \in V : (k, i) \in E\}.$$

Then the coefficient,  $p_i(\boldsymbol{\lambda}, \boldsymbol{\mu})$ , of  $x_i$  in  $r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$  is given by

$$p_i(\boldsymbol{\lambda}, \boldsymbol{\mu}) = l_i + \sum_{k \in I(i)} \mu_{ki} + \sum_{j \in O(i)} \delta_{ij} \lambda_{ij},$$

where

$$\delta_{ij} = \begin{cases} 1 & \text{if } (i, j) \in P \\ -1 & \text{if } (i, j) \in N \end{cases}$$

Thus,  $r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$  can be written as

$$r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \sum_{i=1}^n p_i(\boldsymbol{\lambda}, \boldsymbol{\mu}) x_i + \sum_{(i,j) \in N} \lambda_{ij}$$

where  $\lambda_{ij} + \mu_{ij} = q_{ij}$ ,  $\lambda_{ij} \geq 0, \mu_{ij} \geq 0$ . Then the *roof dual* of QUBO [7, 35], when given in terms of  $h(\mathbf{x})$ , is

$$\text{Minimize } \max\{r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) : \mathbf{x} \in \{0, 1\}^n\}$$

$$\text{Subject to: } \lambda_{ij} + \mu_{ij} = q_{ij} \text{ for all } (i, j) \in P \cup N$$

$$\lambda_{ij} \geq 0, \mu_{ij} \geq 0 \text{ for all } (i, j) \in P \cup N.$$

Since  $r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$  is linear in  $\mathbf{x}$ , its maximum is attained when  $x_i = 1$  for all  $p_i(\boldsymbol{\lambda}, \boldsymbol{\mu}) \geq 0$  and  $x_i = 0$  otherwise. i.e.,

$$\max\{r(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) : \mathbf{x} \in \{0, 1\}^n\} = \sum_{i=1}^n \max\{0, p_i(\boldsymbol{\lambda}, \boldsymbol{\mu})\} + \sum_{(i,j) \in N} \lambda_{ij}.$$

Thus the *roof dual* [6, 35] can be written as

$$\text{RD: Minimize } \sum_{i=1}^n u_i + \sum_{(i,j) \in N} \lambda_{ij}$$

$$\text{Subject to: } u_i \geq p_i(\boldsymbol{\lambda}, \boldsymbol{\mu})$$

$$\begin{aligned} \lambda_{ij} + \mu_{ij} &= q_{ij} \text{ for all } (i, j) \in P \cup N \\ \lambda_{ij} &\geq 0, \mu_{ij} \geq 0 \text{ for all } (i, j) \in P \cup N \\ u_i &\geq 0 \text{ for } i = 1, 2, \dots, n. \end{aligned}$$

Note that RD is a linear program and it is closely linked to network flows on the support graph  $G$ . The optimal objective function value of RD provides an upper bound for the optimal objective function value of QUBO. The dual of RD is precisely the continuous relaxation of the linearization of the Rhys form of QUBO. Thus, the upper bound obtained by the roof dual is precisely the upper bound obtained by solving the continuous relaxation of the linearization of Rhys form of QUBO. Let us take an example to illustrate the concept of roof dual.

*Example 1.2* Let  $f(\mathbf{x}) = 10x_1x_2 - 5x_1x_3 + 20x_1x_4 - 12x_2x_4 - 2x_3x_4 - 6x_1 - 5x_2 + 8x_3 + 5x_4$ . Then  $P = \{(1, 2), (1, 4)\}$  and  $N = \{(1, 3), (2, 4), (3, 4)\}$ . Using the transformation  $x_i = 1 - \bar{x}_i$  for  $(i, j) \in N$  we obtain the Rhys form  $h(x)$  given by

$$h(\mathbf{x}) = 10x_1x_2 + 5\bar{x}_1x_3 + 20x_1x_4 + 12\bar{x}_2x_4 + 2\bar{x}_3x_4 - 6x_1 - 5x_2 + 3x_3 - 9x_4.$$

The support graph associated with  $h(\mathbf{x})$  is given below (Fig. 1.4).

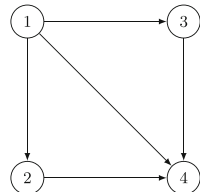
Note that,  $O(1) = \{2, 3, 4\}$ ,  $O(2) = \{4\}$ ,  $O(3) = \{4\}$ ,  $O(4) = \emptyset$  and  $I(1) = \emptyset$ ,  $I(2) = \{1\}$ ,  $I(3) = \{1\}$ ,  $I(4) = \{1, 2, 3\}$ . Therefore,

$$\begin{aligned} p_1(\lambda, \mu) &= -6 + \lambda_{12} - \lambda_{13} + \lambda_{14} \\ P_2(\lambda, \mu) &= -5 - \lambda_{24} + \mu_{12} \\ P_3(\lambda, \mu) &= 3 - \lambda_{34} + \mu_{13} \\ P_4(\lambda, \mu) &= -9 + \mu_{14} + \mu_{24} + \mu_{34}. \end{aligned}$$

Then, the roof dual RD is given by,

$$\begin{aligned} \text{RD: Minimize } & \sum_{i=1}^n u_i + \lambda_{13} + \lambda_{24} + \lambda_{34} \\ \text{Subject to: } & u_1 - \lambda_{12} + \lambda_{13} - \lambda_{14} \geq -6 \\ & u_2 + \lambda_{24} - \mu_{12} \geq -5 \end{aligned}$$

**Fig. 1.4** The support graph of  $h(\mathbf{x})$



$$\begin{aligned}
u_3 + \lambda_{34} - \mu_{13} &\geq 3 \\
u_4 - \mu_{14} - \mu_{24} - \mu_{34} &\geq -9 \\
\lambda_{ij} + \mu_{ij} &= q_{ij} \text{ for all } (i, j) \in P \cup N \\
\lambda_{ij} &\geq 0, \mu_{ij} \geq 0 \text{ for all } (i, j) \in P \cup N \\
u_i &\geq 0 \text{ for } i = 1, 2, \dots, 4.
\end{aligned}$$

The dual of this linear program is

$$\text{Maximize } -6x_1 - 5x_2 + 3x_3 - 9x_4 + 10y_{12} + 5y_{13} + 20y_{14} + 12y_{24} + 2y_{34}$$

$$\text{Subject to: } y_{12} \leq x_1, y_{12} \leq x_2$$

$$y_{13} \leq 1 - x_1, y_{13} \leq x_3$$

$$y_{14} \leq x_1, y_{14} \leq x_4$$

$$y_{24} \leq 1 - x_2, y_{24} \leq x_4$$

$$y_{34} \leq 1 - x_3, y_{34} \leq x_3$$

$$0 \leq x_i \leq 1, i = 1, 2, 3, 4$$

$$y_{ij} \text{ unrestricted for all } (i, j) \in P \cup N.$$

and this is the continuous relaxation of the linearization of  $h(\mathbf{x})$ . (The notion of linearization is discussed in Sect. 1.7 and Chap. 4 and 6.)

## 1.6 Model Building Using QUBO

Let us now look at some motivating applications of the QUBO model. For a detailed discussion on the applications of QUBO and its power of providing a unifying framework to represent a large class of combinatorial optimization problems we refer to Chap. 2.

### 1.6.1 *Person Detection and Tracking in a Crowded Environment*

Identifying and tracking people is an important problem within a variety of application scenarios. These include, examination and exploration of group behaviour, video surveillance, pedestrian detection systems, disaster management, among others. The problem however is very complex, particularly due to the high level of occlusions, and researchers in computer vision developed various techniques

to solve this problem making use of statistical machine learning, mathematical modelling, and optimization [16, 73, 74]. Let us now discuss one such application modelled as a QUBO by Rodriguez et al. [73].

Let  $N = \{1, 2, \dots, n\}$  be a set of points identified in an image as possible person detection locations and let  $c_i$  be an associated confidence score for location  $i$ , for  $i \in N$ . The point  $i$  and the score  $c_i$  can be estimated in different ways, as by appropriately trained preprocessing algorithms. We are also given person density information  $d_i$  (i.e. the number of people per pixel) estimated in a window of size  $\sigma$  at location  $i$ , for  $i \in N$ . We want to find locations of people in the image such that the sum of detector confidence score is maximized while making use of the density information to minimize selection of locations with significantly overlapping neighborhoods, which in turn minimizes potential counting errors accumulated due to multiple detection of the same person. Figure 1.5a gives a sample shot of a crowd and (b) represents the density contours. Part (c) of the figure gives potential locations and (d) gives the filtered solution produced by QUBO. Note

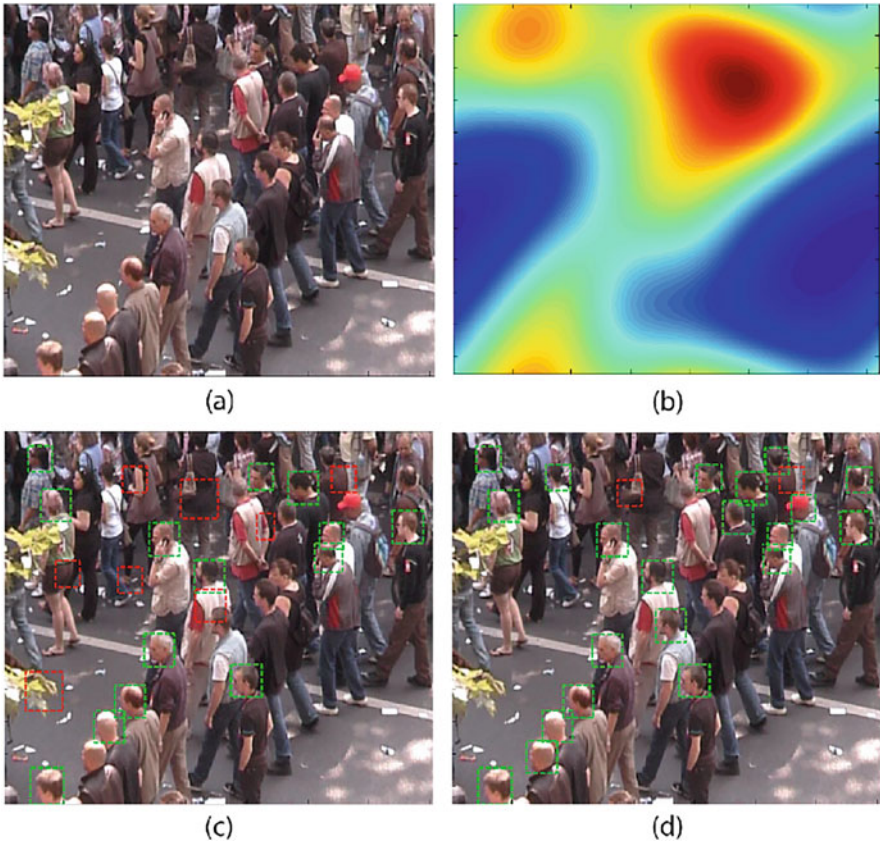


Fig. 1.5 Person detection sample and crowd density contours

that some of the red rectangles, that are potential locations, are either removed as irrelevant or confirmed in the QUBO solution.<sup>1</sup>

Let  $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \{0, 1\}^n$ , where  $x_i = 1$  implies a detection at location  $i$  is confirmed and 0, otherwise. Then  $\mathbf{c}\mathbf{x}$  measures the total confidence score of ‘confirmed’ locations. To make sure that only valid configurations of non-overlapping locations are selected, we construct a penalty matrix  $\mathbf{W}$ , where  $w_{ij} = -\infty$  if detections at locations  $i$  and  $j$  have significant area overlap ratio, and 0 otherwise. Then, maximizing  $\mathbf{c}^T\mathbf{x} + \mathbf{x}^T\mathbf{W}\mathbf{x}$  provides a meaningful model to represent the person detection problem. This, in fact, is a variation of a model proposed in [16].

To improve the accuracy of the model, Rodriguez et al. [73] introduced another quadratic term penalizing the difference between the density values estimated in two ways: (i) the vector  $\mathbf{d}$  obtained using a regression-based density estimator and (ii) the vector  $\mathbf{A}\mathbf{x}$  counting the density of active detections for an appropriately defined  $\mathbf{A}$  matrix. This leads to the penalty term  $\|\mathbf{d} - \mathbf{A}\mathbf{x}\|_1^2$ . Then, the objective function to be maximized is

$$\mathbf{c}^T\mathbf{x} + \mathbf{x}^T\mathbf{W}\mathbf{x} - \alpha\|\mathbf{d} - \mathbf{A}\mathbf{x}\|_1^2,$$

where  $\mathbf{x} \in \{0, 1\}^n$  and  $\alpha$  is a parameter. This problem can be written as QUBO by simplifying  $\|\mathbf{d} - \mathbf{A}\mathbf{x}\|_1^2$  and combining it with  $\mathbf{x}^T\mathbf{W}\mathbf{x}$ .

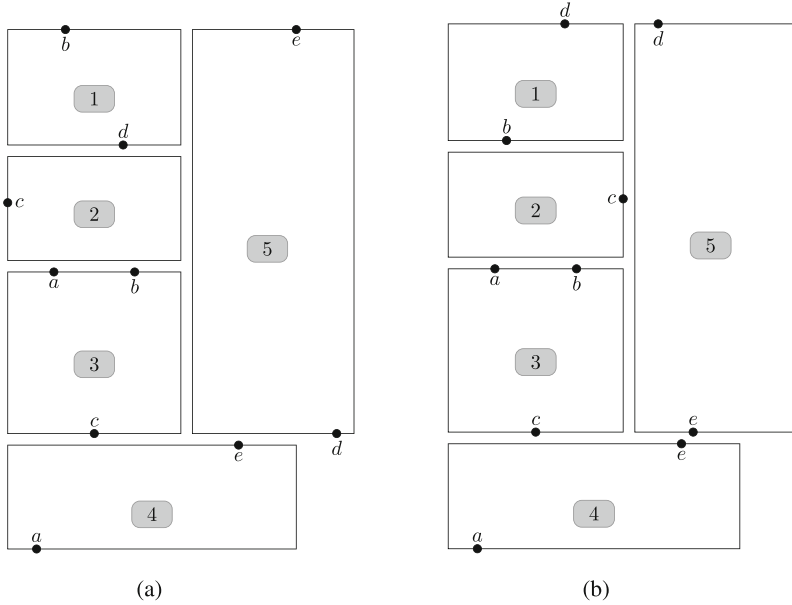
## 1.6.2 Module Flipping in Circuit Layout Design

Let us now look at another example of QUBO that arises in the layout design of VLSI (very large scale system integration). Our discussion here is based on the works of Boros and Hammer [7] and Cheng et al. [12]. In the layout design of circuits, rectangular modules containing one or more pins are embedded in a base board. Each pin on a module has a compatible pin on another module and these compatible pairs need to be connected using horizontal and/or vertical wiring. The reliability has an inverse relationship with wire length and hence we are interested in minimizing the total wire length. Each module can be placed in four different ways on its designated space on the base board by flipping horizontally, vertically, or both. In Fig. 1.6a, we give a layout of modules and in Fig. 1.6b we give a layout after vertically flipping module 1, horizontally flipping module 2, and a vertical flip followed by a horizontal flip on module 5. The compatible pairs of pins are labelled using the same alphabets. For example, a pin with label  $a$  needs to be connected to another with the same label  $a$  and so on.

---

<sup>1</sup> Figure 1.5 was provided by Mikel Rodriguez and reproduced here with his permission.





**Fig. 1.6** Module flipping example. Applied a vertical flip on module 1, horizontal flip on module 2, vertical flip followed by horizontal flip on module 5. (a) Original position. (b) After flip operations are applied on modules 1, 2, and 5

The module flipping problem is to find the orientations of the placement of each module on the base board, identified by one or more flipping operations, so that the total wire length is minimized. This problem was originally proposed by Cheng et al. [12] and they presented a maximum cut formulation, which as we know, is equivalent to QUBO. Boros and Hammer [7] presented a direct QUBO formulation of the problem and we discuss this model here.

Let  $N = \{1, 2, \dots, n\}$  be the collection of all modules placed on the base board with an initial layout. Consider the binary decision variable  $x_i$  which takes value 1 if module  $i$  is flipped horizontally, for  $i = 1, 2, \dots, n$ . Likewise, consider the binary decision variable  $y_i$  which takes value 1 if module  $i$  is flipped vertically, for  $i = 1, 2, \dots, n$ . For each pair  $(i, j)$  of modules and  $(r, s) \in \{0, 1\} \times \{0, 1\}$ , let  $H_{ij}^{(r,s)}$  denotes the total length of horizontal wire segments which connects the pins of modules  $i$  and  $j$  under the flipping sequence  $(r, s)$ . For example, if  $(r, s) = (0, 0)$  no horizontal flipping occurs for modules  $i$  and  $j$ , if  $(r, s) = (0, 1)$ , only module  $j$  is flipped horizontally, and so on. Similarly, let  $V_{ij}^{(r,s)}$  denotes the total length of vertical wire segments which connects the pins of modules  $i$  and  $j$  under the flipping sequence  $(r, s)$ . Now, the total wire-length, as a function of flipping operation of the

modules, can be expressed as

$$\begin{aligned} \phi(\mathbf{x}, \mathbf{y}) = & \sum_{i=1}^{n-1} \sum_{j=i+1}^n \left( H_{ij}^{(0,0)} \bar{x}_i \bar{x}_j + H_{ij}^{(0,1)} \bar{x}_i x_j + H_{ij}^{(1,0)} x_i \bar{x}_j + H_{ij}^{(1,1)} x_i x_j \right) \\ & + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \left( H_{ij}^{(0,0)} \bar{y}_i \bar{y}_j + H_{ij}^{(0,1)} \bar{y}_i y_j + H_{ij}^{(1,0)} y_i \bar{y}_j + H_{ij}^{(1,1)} y_i y_j \right) \end{aligned} \quad (1.20)$$

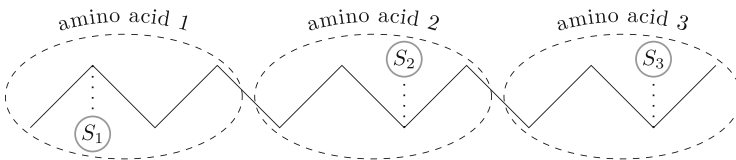
where  $\bar{x}_i = 1 - x_i$ . Then, the optimal flipping of the modules corresponds to assigning 0 – 1 values to the variables  $x_i$  and  $y_i$  for  $i \in N$  such that  $\phi(\mathbf{x}, \mathbf{y})$  is minimized. It can be verified that minimization of  $\phi(\mathbf{x}, \mathbf{y})$  decomposes into two problems, one with the  $\mathbf{x}$ -variables and the other with the  $\mathbf{y}$ -variables and each such problem is a QUBO.

Related applications of QUBO in equivalent forms such as Maximum Cut, weighted stable set etc., can be found in [3, 9, 15, 46, 54].

### 1.6.3 Side-Chain Positioning in Protein Design

The *side chain positioning problem* arises as a subproblem in protein structure prediction which has a natural QUBO formulation. Our discussion here is based on the works [11, 30, 31, 47]. While the formulation itself is straightforward, to make the discussion clearer, let us very briefly review some related concepts, terminology, and background information and for this purpose, we follow the articles [11, 31].

A *protein molecule* is composed of a chain of amino acids and each amino acid consists of a centralized single carbon atom along with a hydrogen atom, an amino group  $\text{NH}_2$ , a carboxyl group  $\text{COOH}$ , and a *side chain* which characterizes the amino acid. Carbon, hydrogen, the amino group, and the carboxyl group are called the *main atoms* and the *protein backbone* is formed by a repeating sequence of the main atoms (main chain) and a *side chain* is attached to the backbone for each element of this sequence (Fig. 1.7).



**Fig. 1.7** A chain of three amino acids. The labels  $S_1$ ,  $S_2$  and  $S_3$  indicate the corresponding side-chains

The chemical composition of a protein molecule is specified by the sequence of the associated amino acids. Every amino acid main chain has the freedom to *rotate* at specified points. A side chain of amino acids (with the exception of glycine) can also rotate at different points and the three dimensional structure of a protein is identified by the location of its backbone (main chain) atoms and the combined rotations of the main chain and the side chains. For an appropriate notion of *energy*, a protein structure folds into a minimal energy state to reach chemical stability. Quoting from [31], “Chemical stability of a protein comes from four sources: the internal stability of the three-dimensional structure of its backbone; the internal stability of each rotamer; the interaction of each rotamer with the main chain of the amino acid it is attached to; and the chemical interactions of the rotamers that are positioned close to each other, either on the protein sequence, or in three-dimensional space.” In protein synthesis, we want to maximize the chemical stability using appropriate quantitative stability measures.

The backbone structure is assumed to be given and our aim is to select one rotamer (side-chain molecule) to be associated with each amino acid sequence in the backbone. Let  $e_0$  be the stability measure for the backbone structure, which is constant. Suppose that the backbone consists of  $m$  amino acid sequences and for amino acid  $k$ , we select a rotamer from the candidate list  $V_k, k = 1, 2, \dots, m$ . For rotamer  $i \in V_k$  let  $e_i^k$  be the contribution of  $i$  to the stability measure which consists of the sum of the interaction stability measure of candidate rotamer  $i$  with its associated amino acid  $k$  and the internal stability measure of rotamer  $i$ . For each  $i \in V_k$  and  $j \in V_t$ , let  $e_{ir}^{kt}$  be the stability measure due to the interaction between rotamer  $i \in V_k$  and  $j \in V_t$ . This value depends on the nature of the rotamers involved, the nature of the amino acids they attach to, and the proximity level of  $i$  and  $j$ . Since the backbone is already known, the proximity level is also known. The values  $e_0$ ,  $e_i^k$ , and  $e_{ir}^{kt}$  are estimated either by experiments or by theoretical considerations. For our modeling purpose, it is not highly relevant the way these values are obtained or their magnitudes. The side-chain positioning problem is to select an  $i_k$  from  $V_k$  for each  $k = 1, 2, \dots, m$  such that

$$e_0 + \sum_{k=1}^m e_{i_k}^k + \frac{1}{2} \sum_{k=1}^m \sum_{t=1}^m e_{i_k i_t}^{kt}$$

is maximized. Let us now formulate this problem as QUBO.

Let  $V = \cup_{k=1}^m V_k$ . Without loss of generality, assume  $V = \{1, 2, \dots, n\}$ . Construct the complete  $m$ -partite undirected graph  $G = (V, E)$  where  $(i, j) \in E$  if  $i \in V_k, j \in V_t, t \neq k$ . Note that we need to select precisely one element for each  $V_k$  for  $k = 1, 2, \dots, m$ . For each  $i \in V$  consider the binary decision variable  $x_i$  where  $x_i = 1$  if rotamer  $i$  is selected and, 0 otherwise. Define the  $n \times n$  cost matrix  $Q$  with

its  $(i, j)$ th element  $q_{ij}$  as

$$q_{ij} = \begin{cases} 0 & \text{if } i = j \\ -M & \text{if } i \neq j \text{ and } i, j \in V_k \text{ for some } k, \\ \frac{1}{2}e_{rs}^{kt} & \text{if } (i, j) \in E, \end{cases}$$

where  $M$  is a large positive number and the  $n$  vector  $\mathbf{c}^T = (c_1, c_2, \dots, c_n)$  where  $c_i = e_j^k$  if  $i \in V^k$  and  $i = j$ . Now an optimal solution to the QUBO instance  $(\mathbf{Q}, \mathbf{c}, \max)$  gives an optimal solution to the side chain positioning problem. Note that, no two elements from the same set  $V_k$  will be selected in an optimal solution for any  $k$  because of the penalty value and since the objective function is of maximization type,  $c_i \geq 0$  for all  $i$  and  $e_{rs}^{kt} \geq 0$  for all  $k, t, r, s$  at least one elements from each  $V_k$  will be selected, establishing the validity of the model.

### 1.6.4 QUBO and Machine Scheduling

We now discuss a general framework of a special case of QUBO, which can be used to model several important machine scheduling problems. A matrix  $\mathbf{Q}$  is called a *half product matrix* if it is upper triangular with diagonal elements zero and there exist vectors  $\mathbf{a} = (a_1, a_2, \dots, a_n)$  and  $\mathbf{b} = (b_1, b_2, \dots, b_n)$  such that  $q_{ij} = a_i b_j$  for  $i = 1, 2, \dots, n$  and  $j > i$ . A *half-product QUBO* is a special case of QUBO where  $\mathbf{Q}$  is a half-product matrix. Thus, a QUBO with a half-product matrix can be written as

$$\begin{aligned} & \text{Maximize} && \sum_{1 \leq i < j \leq n} a_i b_j x_i x_j + \sum_{j=1}^n c_j x_j \\ & \text{Subject to} && \mathbf{x} \in \{0, 1\}^n \end{aligned}$$

The half-product QUBO was introduced by Badics and Boros [2] and independently by Kubiak [52]. Despite the apparently simple special structure, the half-product QUBO is still NP-hard since the subset sum problem is a special case of it [2].

Various single machine scheduling problems can be formulated as a half-product QUBO. This includes, the single-machine variance minimization problem [2, 52], Single machine scheduling to minimize total weighted earliness and tardiness [43] and scheduling with controllable processing times [41]. The Ising version of the half-product QUBO was considered by Kubiak [52], Amit [1], and Mattis [58]. Scheduling jobs on two machines to minimize makespan [43] as well as scheduling on two machines to minimize total weighted completion time [43] can also be formulated as a half-product QUBO. Let us discuss one such formulations using the example of *single machine scheduling with controllable processing times* (SCPT) [41].

There are  $n$  jobs to be processed on a single machine where the processing time of jobs are variables. For job  $j$ , let the processing time  $p_j \in [0, \alpha_j]$ ,  $j = 1, 2, \dots, n$ . To schedule the jobs, one needs to identify the optimal values of the processing times  $p_j$  and then order the jobs so that the sum of the total weighted completion time  $\sum_{j=1}^n w_j C_j$  and the total weighted processing time compression  $\sum_{j=1}^n v_j(u_j - p_j)$  is minimized. Here,  $C_j$  denote the completion time of job  $j$ . This scheduling problem, denoted by SCPT, was first studied by Vickson [78, 79] and the half-product QUBO formulation discussed here is from [41]. Vickson [78] showed that there exists an optimal processing time vector  $\mathbf{p} = (p_1, p_2, \dots, p_n)$  such that  $p_j = 0$  or  $\alpha_j$ . Combining this observation with the well-known weighted shortest processing time rule (SWPT) [76], we can see that there exists an optimal solution to SCPT such that jobs processing times  $p_j = \alpha_j$  are sequenced in the non-decreasing order of the ratios  $\frac{\alpha_j}{w_j}$  [41]. With this observation, define the decision variables

$$x_j = \begin{cases} 0 & \text{if } p_j = 0 \\ 1 & \text{if } p_j = \alpha_j \end{cases}$$

and the problem SCPT is well defined when the vectors  $\mathbf{w} = (w_1, w_2, \dots, w_n)$ ,  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ , and  $\mathbf{v} = (v_1, v_2, \dots, v_n)$  are given. Then, the objective function of SCPT is to minimize  $f(\mathbf{x})$ , where

$$\begin{aligned} f(\mathbf{x}) &= \sum_{j=1}^n w_j x_j \sum_{i=1}^j \alpha_i x_i + \sum_{j=1}^n v_j \alpha_j (1 - x_j) \\ &= \sum_{1 \leq i < j \leq n} \alpha_i w_j x_i x_j - \sum_{j=1}^n \alpha_j (v_j - w_j) x_j + \sum_{j=1}^n v_j \alpha_j \end{aligned}$$

Note that  $\sum_{j=1}^n v_j \alpha_j$  is a constant and can be discarded. Choosing,  $a_i = \alpha_i$ ,  $b_i = -w_i$  and  $c_i = \alpha_i (v_i - w_i)$ , we can see that minimization of  $f(\mathbf{x})$  can be accomplished by solving the half-product QUBO.

## 1.7 Mixed Integer Programming Formulations

Various mixed integer linear programming (MILP) formulations of QUBO are discussed in detail in other chapters of this book. Most of the basic formulations are based on writing the product of two variables (not necessarily both are of binary type) as a new variable and force the definition of the underlying product by imposing additional constraints, that may or may not exploit the sign of the  $q_{ij}$  values. Early research work on these types of reductions goes back to 1960s [13, 20, 29, 80, 81] where product of two binary variables are ‘linearized’. Later, Glover [25, 26] and Glover and Woolsey [27] considered linearization of the

product of a binary variable and a continuous variable. Most of these linearization techniques can be presented under a general approximation framework introduced by McCormick [59, 60].

Consider the bounded variables

$$l_1 \leq x_1 \leq u_1 \text{ and } l_2 \leq x_2 \leq u_2.$$

Then,

$$\begin{aligned} (x_1 - l_1)(u_2 - x_2) &\geq 0, & (u_1 - x_1)(x_2 - l_2) &\geq 0 \\ (x_1 - l_1)(x_2 - l_2) &\geq 0, & \text{and } (u_1 - x_1)(u_2 - x_2) &\geq 0. \end{aligned}$$

Expanding and rearranging the terms, we have

$$x_1x_2 \leq l_1x_2 + u_2x_1 - l_1u_2$$

$$x_1x_2 \leq l_2x_1 + u_1x_2 - l_2u_1$$

$$x_1x_2 \geq l_1x_2 + l_2x_1 - l_1l_2$$

$$x_1x_2 \geq u_2x_1 + u_2x_2 - u_1u_2$$

Now, let us replace the term  $x_1x_2$  by a new variable  $w$  in the above inequalities and we get

$$w \leq l_1x_2 + u_2x_1 - l_1u_2 \tag{1.21}$$

$$w \leq l_2x_1 + u_1x_2 - l_2u_1 \tag{1.22}$$

$$w \geq l_1x_2 + l_2x_1 - l_1l_2 \tag{1.23}$$

$$w \geq u_2x_1 + u_2x_2 - u_1u_2 \tag{1.24}$$

The inequalities (1.21) and (1.22) are called *lower McCormick envelopes* and (1.23), and (1.24) are called *upper McCormick envelopes* [59, 60]. The variable  $w$  along with the McCormick envelop inequalities provide an approximation to the product  $x_1x_2$ . This approximation becomes exact when one of the variables  $x_1$  or  $x_2$  is binary and it becomes the linearizations proposed by Glover [25, 26] and Glover and Woolsey [27]. To see the validity of the linearization, without loss of generality, assume  $x_2$  is binary. Then  $l_2 = 0$  and  $u_2 = 1$ . When  $x_2 = 0$ , the inequalities (1.22) and (1.23) guarantees that  $w = 0$ . Likewise, when  $x_2 = 1$ , inequalities (1.21) and (1.24) guarantees that  $w = x_1$ . The special case of McCormick envelop inequalities involving binary variables leads to a natural MILP formulation of QUBO which is the well-known Glover's linearization Glover [25, 26]. When both variables involved in a product are binary, McCormick envelop inequalities reduces to Glover-Woolsey linearization [27]. We also want to highlight that the linearization of product of two binary variables are reported in literature either explicitly or

implicitly, even prior to [27]. For example, Goldman [29] pointed out the works of Fortet [20, 21] and Dantzig [13] in this context.

Replace the product  $x_i x_j$  with a new variable  $w_{ij}$  in the objective function of QUBO and force the equality  $x_i x_j = w_{ij}$  by introducing the McCormick envelop inequalities. The resulting MILP formulation, known as Glover-Woolsey linearization or the *standard linearization*, is given below.

$$\text{GW: Maximize } \sum_{i=1}^n \sum_{j=1}^n q_{ij} w_{ij} + \sum_{i=1}^n c_i x_i$$

$$\text{Subject to: } w_{ij} - x_i \leq 0 \text{ for } i, j \in \{1, 2, \dots, n\} \quad (1.25)$$

$$w_{ij} - x_j \leq 0 \text{ for } i, j \in \{1, 2, \dots, n\} \quad (1.26)$$

$$x_i + x_j - w_{ij} \leq 1 \text{ for } i, j \in \{1, 2, \dots, n\} \quad (1.27)$$

$$w_{ij} \geq 0 \text{ for } i, j \in \{1, 2, \dots, n\} \quad (1.28)$$

$$x_i \in \{0, 1\} \text{ for } i = 1, 2, \dots, n. \quad (1.29)$$

Recall the definition of the sets  $P$  and  $N$  introduced in Sect. 1.4.3. For  $(i, j) \in P$  constraints (1.27) and (1.28) can be removed and for  $(i, j) \in N$  constraints (1.25) and (1.26) can be removed. Thus GW becomes the *reduced Glover-Woolsey linearization*

$$\text{RGW: Maximize } \sum_{i=1}^n \sum_{j=1}^n q_{ij} w_{ij} + \sum_{i=1}^n c_i x_i$$

$$\text{Subject to: } w_{ij} - x_i \leq 0 \text{ for } (i, j) \in P \quad (1.30)$$

$$w_{ij} - x_j \leq 0 \text{ for } (i, j) \in P \quad (1.31)$$

$$x_i + x_j - w_{ij} \leq 1 \text{ for } (i, j) \in N \quad (1.32)$$

$$w_{ij} \geq 0 \text{ for } (i, j) \in N \quad (1.33)$$

$$x_i \in \{0, 1\} \text{ for } i = 1, 2, \dots, n. \quad (1.34)$$

The convex hull of solutions of GW is called the *Boolean quadric polytope*[64] which is discussed in detail in Chap. 4. Various other MILP formulations of QUBO are known in literature and most of them uses Glover-Woolsey linearization or Glover's linearization in one form or another. For a through discussion of this, we refer to Chap. 6 and the recent papers [67, 68].

The continuous relaxation of GW (and RGW) is denoted by CGW (CRGW) and it is obtained by replacing constraint (1.29) (constraint (1.34)) by  $x_i \in [0, 1]$  for  $i = 1, 2, \dots, n$ . CGW and CRGW are linear programs and hence can be solved using general purpose linear programming solvers. However, their special structure also

brings us additional structural properties and efficient algorithms. Some of these connections and linkages with roof duality is discussed in Chap. 5.

The polytope defined by the constraints of CGW is called the *fractional Boolean quadric polytope*. Let us first present a useful necessary condition for a solution  $(\mathbf{x}, \mathbf{w})$  to be an extreme point of the fractional Boolean quadric polytope.

**Lemma 1.5** *If  $(\mathbf{x}, \mathbf{w})$  is an extreme point of the fractional Boolean quadric polytope then  $w_{ij} \in \{0, \min\{x_i, x_j\}, x_i + x_j - 1\}$  for all  $i, j$ .*

**Proof** Suppose  $(\mathbf{x}, \mathbf{w})$  is an extreme point of the fractional Boolean quadric polytope. Then, choose a cost matrix  $\mathbf{Q}$  and a cost vector  $\mathbf{c}$  such that  $(\mathbf{x}, \mathbf{w})$  is an optimal solution to CGW. Without loss of generality assume that  $q_{ij} \neq 0$  for  $i \neq j$  and  $c_i \neq 0$  for all  $i$ . Clearly, by feasibility  $0 \leq w_{ij} \leq \min\{x_i, x_j\}$  for every  $i, j, i \neq j$ . If possible let  $0 < w_{hk} < \min\{x_h, x_k\}$  for some  $h, k, h \neq k$  and  $x_h + x_k - 1 \neq w_{hk}$ . Define the solution  $(\mathbf{x}, \hat{\mathbf{w}})$  where

$$\hat{w}_{ij} = \begin{cases} w_{ij} + \epsilon & \text{if } (i, j) = (h, k) \\ w_{ij} & \text{Otherwise,} \end{cases}$$

where  $\epsilon \neq 0$ . If  $q_{hk} > 0$  then  $(\mathbf{x}, \hat{\mathbf{w}})$  is a feasible solution to CGW for sufficiently small  $\epsilon > 0$  which is better than  $(\mathbf{x}, \mathbf{w})$ , contradicting the optimality of  $(\mathbf{x}, \mathbf{w})$ . Similarly, if  $q_{hk} < 0$  then  $(\mathbf{x}, \hat{\mathbf{w}})$  is a feasible solution to CGW for sufficiently large  $\epsilon < 0$  which is better than  $(\mathbf{x}, \mathbf{w})$ , contradicting the optimality of  $(\mathbf{x}, \mathbf{w})$ . This completes the proof.  $\square$

We now prove the *half integrality property* of extreme points of the fractional Boolean quadric polytope [64].

**Theorem 1.3** *If  $(\mathbf{x}, \mathbf{w})$  is an extreme point of the fractional Boolean quadric polytope the  $x_i \in \{0, 1, \frac{1}{2}\}$  for all  $i$  and  $w_{ij} \in \{0, 1, \frac{1}{2}\}$  for all  $(i, j) \in P \cup N$ .*

**Proof** Let  $\mathfrak{F}$  denote the fractional Boolean quadric polytope and  $(\mathbf{x}, \mathbf{w})$  be an extreme point. Let

$$E^1 = \{i : 0 < x_i < \frac{1}{2}\} \text{ and } E^2 = \{i : \frac{1}{2} < x_i < 1\}.$$

If  $E^1 \cup E^2 = \emptyset$  then from Lemma 1.5,  $(\mathbf{x}, \mathbf{w})$  is of the required type. So assume that  $E^1 \cup E^2 \neq \emptyset$ . Now construct the solutions  $(\mathbf{x}^1, \mathbf{w}^1)$  and  $(\mathbf{x}^2, \mathbf{w}^2)$  as follows: For  $\epsilon > 0$  let

$$x_i^1 = \begin{cases} x_i + \epsilon & \text{if } i \in E^1 \\ x_i - \epsilon & \text{if } i \in E^2 \\ x_i & \text{Otherwise} \end{cases} \quad \text{and} \quad x_i^2 = \begin{cases} x_i - \epsilon & \text{if } i \in E^1 \\ x_i + \epsilon & \text{if } i \in E^2 \\ x_i & \text{Otherwise.} \end{cases}$$



From Lemma 1.5,  $w_{ij} \in \{0, \min\{x_i, x_j\}, x_i + x_j - 1\}$ . Define

$$w_{ij}^1 = \begin{cases} \min\{x_i^1, x_j^1\} & \text{if } w_{ij} = \min\{x_i, x_j\} \\ x_i^1 + x_j^1 - 1 & \text{if } w_{ij} = x_i + x_j - 1 \\ 0 & \text{Otherwise} \end{cases}$$

$$w_{ij}^2 = \begin{cases} \min\{x_i^2, x_j^2\} & \text{if } w_{ij} = \min\{x_i, x_j\} \\ x_i^2 + x_j^2 - 1 & \text{if } w_{ij} = x_i + x_j - 1 \\ 0 & \text{Otherwise} \end{cases}$$

For sufficiently small  $\epsilon > 0$ ,  $(\mathbf{x}^1, \mathbf{w}^1)$  and  $(\mathbf{x}^2, \mathbf{w}^2)$  are feasible solutions. It can be verified that  $(\mathbf{x}, \mathbf{w}) = \frac{1}{2}\{(\mathbf{x}^1, \mathbf{w}^1) + (\mathbf{x}^2, \mathbf{w}^2)\}$ , contradicting the fact that  $(\mathbf{x}, \mathbf{w})$  is an extreme point and the result follows.  $\square$

In Sect. 1.4 we observed that the maximum cut problem and QUBO are equivalent. In this sense, an integer programming formulation of the maximum cut problem provides an indirect integer programming representation of QUBO. Consider a complete graph  $K_n = (V, E)$  and let  $q_{ij}$  be the weight of edge  $(i, j) \in E$ . Then the maximum cut problem in  $K_n$  can be written as an MILP

$$\begin{aligned} \text{MCUT: Minimize} \quad & \sum_{i < j \leq n} q_{ij} x_{ij} \\ \text{Subject to:} \quad & x_{ij} + x_{ik} + x_{jk} \leq 2 \text{ for } 1 \leq i < j \leq n \\ & x_{ij} - x_{ik} - x_{jk} \leq 0 \text{ for } 1 \leq i < j \leq n, k \neq i, j \\ & x_{ij} \in \{0, 1\} \text{ for } 1 \leq i < j \leq n. \end{aligned}$$

The convex hull of feasible solutions of MCUT is called the *cut polytope*. The relationship between the Boolean quadric polytope and the cut polytope is discussed in Chap. 4.

## 1.8 Conclusion

The primary goal of this chapter was to give a brief introduction to QUBO along with basic definitions and motivating examples. Each of the chapters that follows is more focussed and is on a specific aspect of QUBO providing a comprehensive and in-depth analysis of the topic.

**Acknowledgments** I am thankful to Fred Glover and Navpreet Kaur for their feedback on an earlier version of this chapter which improved the presentation. This work was partially supported by an NSERC discovery grant.

## References

1. D.J. Amit, *Modeling Brain Function: The World of Attractor Neural Networks* (Cambridge University Press, Cambridge, 1989)
2. T. Badics, E. Boros, Minimization of half-products. *Math. Oper. Res.* **23**, 649–660 (1998)
3. F. Barahona, M. Grotschel, M. Junger, G. Reinelt, An application of combinatorial optimization to statistical physics and circuit layout design. *Oper. Res.* **36**, 493–513 (1988)
4. R. Bar-Yehuda, S. Even, A local-ratio theorem for approximating the weighted vertex cover problem. *Ann. Discrete Math.* **25**, 27–45 (1985)
5. Z. Bian, F. Chudak, W.G. Macready, G. Rose, The Ising model: teaching an old problem new tricks. D-Wave systems technical report, Aug 2010
6. E. Boros, P.L. Hammer, A max-flow approach to improved roof-duality in quadratic 0 – 1 minimization. RUTCOR Research Report RRR 15-1989, RUTCOR (1989)
7. E. Boros, P.L. Hammer, The Max-Cut problem and quadratic 0-1 optimization; polyhedral aspects, relaxations and bounds. *Ann. Oper. Res.* **33**, 151–180 (1991)
8. E. Boros, P.L. Hammer, Pseudo-Boolean optimization. *Discrete Appl. Math.* **123**, 155–225 (2002)
9. E. Boros, P.L. Hammer, M. Minoux, D.J. Rader Jr., Optimal cell flipping to minimize channel density in VLSI design and pseudo-Boolean optimization. *Discrete Appl. Math.* **90**, 69–88 (1999)
10. S.G. Brush, History of the Lenz-Ising model. *Rev. Mod. Phys.* **39**, 883–893 (1967)
11. B. Chazelle, C. Kingsford, M. Singh, A semidefinite programming approach to side chain positioning with new rounding strategies. *INFORMS J. Comput.* **16**, 380–392 (2004)
12. C.K. Cheng, S.Z. Yao, T.C. Hu, The orientation of modules based on graph decomposition. *IEEE Trans. Comput.* **C-40**, 774–780 (1991)
13. G.B. Dantzig, On the significance of solving linear programming problems with some integer variables. *Econometrica* **28**, 30–44 (1960)
14. G.B. Dantzig, Linear programming. *Oper. Res.* **50**, 42–47 (2002)
15. B. Das, A.K. Mahato, A.K. Khan, Via minimization for multi-layer channel routing in VLSI design, in *Fourth International Conference on Communication Systems and Network Technologies* (2014)
16. C. Desai, D. Ramanan, C.C. Fowlkes, Discriminative models for multi-class object layout. *Int. J. Comput. Vis.* **95**, 1–12 (2011)
17. P. Erdos, G. Szekeres, A combinatorial problem in geometry. *Compos. Math.* **2**, 463–470 (1935)
18. J.-A. Ferrez, K. Fukuda, Th.M. Lieblich, Solving the fixed rank convex quadratic maximization in binary variables by a parallel zonotope construction algorithm. *Eur. J. Oper. Res.* **166**, 35–50 (2005)
19. L. Festinger, The analysis of sociograms using matrix algebra. *Hum. Relat.* **2**, 153–158 (1949)
20. R. Fortet, Applications de l’algèbre de boole en recherche opérationnelle. *Rev. Fr. Rech. Opér.* **4**, 5–36 (1959)
21. R. Fortet, L’algèbre de boole et ses applications en recherche opérationnelle. *Cahiers Centre d’Etudes Rech. Opér.* **4**, 17–26 (1960)
22. E. Forsyth, L. Katz, A matrix approach to the analysis of sociometric data: preliminary report. *Sociometry* **9**, 340–347 (1946)
23. M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W. H. Freeman, San Francisco, 1979)
24. S. Galam, *Sociophysics: A Physicist’s Modeling of Psycho-Political Phenomena* (Springer, New York, 2012)
25. F. Glover, Improved linear integer programming formulations of nonlinear integer problems. *Manag. Sci.* **22**, 455–460 (1975)
26. F. Glover, An improved MIP formulation for products of discrete and continuous variables. *J. Inf. Optim. Sci.* **5**, 469–471 (1984)

27. F. Glover, E. Woolsey, Further reduction of zero-one polynomial programming problems to zero-one linear programming problems. *Oper. Res.* **21**, 141–161 (1973)
28. F. Glover, G. Kochenberger, Y. Du, A tutorial on formulating and using QUBO models. University of Colorado, Denver, 2019
29. A.J. Goldman, Linearization in 0-1 variables: a clarification. *Oper. Res.* **31**, 946–947 (1983)
30. R.F. Goldstein, Efficient rotamer elimination applied to protein side-chains and related spin glasses. *Biophys. J.* **66**, 1335–1340 (1994)
31. D. Gusfield, *Integer Linear Programming in Computational and Systems Biology: An Entry-Level Text and Course* (Cambridge University Press, Cambridge, 2019)
32. P.L. Hammer, A.A. Rubin, Some remarks on quadratic programming with 0-1 variables. *RAIRO-Oper. Res. Rech. Opér.* **3**, 67–79 (1970)
33. P.L. Hammer, S. Rudeanu, *Boolean Methods in Operations Research and Related Areas* (Springer, Berlin, 1968)
34. P.L. Hammer, I. Rosenberg, S. Rudeanu, On the determination of the minima of pseudo-Boolean functions. *Stud. Cerc. Mat.* **14**, 359–364 (1963)
35. P.L. Hammer, P. Hansen, B. Simone, Roof duality, complementations, and persistency in quadratic 0-1 optimization. *Math. Program.* **28**, 121–155 (1984)
36. F. Harary, I.C. Ross, A procedure for clique detection using the group matrix. *Sociometry* **20**, 205–215 (1957)
37. R. Hassin, A. Levin, The minimum generalized vertex cover problem, in *European Symposium on Algorithms* (2003), pp. 289–300
38. C. Helmberg, F. Rendl, Solving quadratic (0,1)-problems by semidefinite programs and cutting planes. *Math. Program.* **82**, 291–315 (1998)
39. D.S. Hochbaum, A. Pathria, Forest harvesting and minimum cuts: a new approach to handling spatial constraints. *For. Sci.* **43**, 544–554 (1997)
40. T. Ising, R. Folk, R. Kenna, B. Berche, Y. Holovatch, The fate of Ernst Ising and the fate of his model. *J. Phys. Stud.* **21**(3), 3002, 19 pp. (2017)
41. A. Janiak, M.Y. Kovalyov, W. Kubiak, F. Werner, Positive half-products and scheduling with controllable processing times. *Eur. J. Oper. Res.* **165**, 416–422 (2005)
42. M. Jünger, T. Lieblich, D. Naddef, G. Nemhauser, W. Pulleyblank, G. Reinelt, G. Rinaldi, L. Wolsey, *50 Years of Integer Programming 1958–2008: From the Early Years to the State-of-the-Art* (Springer, Berlin, 2010)
43. B. Jurisch, W. Kubiak, J. Józefowska, Algorithms for minclique scheduling problems. *Discret. Appl. Math.* **72**, 115–139 (1997)
44. D.R. Karger, C. Stein, A new approach to the minimum cut problem. *J. ACM* **43**, 601–640 (1996)
45. G. Kochenberger, F. Glover, A unified framework for modeling and solving combinatorial optimization problems: a tutorial, in *Multiscale Optimization Methods and Applications*, ed. by W. Hager, S.-J. Huang, P. Pardalos, O. Prokopyev (Springer, Berlin, 2006), pp. 101–124
46. H. Kim, An application algorithm for the via minimization problem in channel routing, in *Proceedings of the 1990 Symposium on Applied Computing* (1990)
47. C.L. Kingsford, B. Chazelle, M. Singh, Solving and analyzing side-chain positioning problems using linear and integer programming. *Bioinformatics* **21**, 1028–1036 (2005)
48. G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lu, H. Wang, Y. Wang, The unconstrained binary quadratic programming problem: a survey. *J. Comb. Optim.* **28**, 58–81 (2014)
49. G. Kohring, Ising models of social impact: the role of cumulative advantage. *J. Phys. I* **6**, 301–308 (1996)
50. H. Konno, Maximization of a convex quadratic function under linear constraints. *Math. Program.* **11**, 117–127 (1976)
51. H. Konno, Maximizing a convex quadratic function over a hypercube. *J. Oper. Res. Soc. Jpn.* **23**, 171–188 (1980)
52. W. Kubiak, New results on the completion time variance minimization. *Discrete Appl. Math.* **58**, 157–168 (1995)
53. R. Lazmy, Mixed integer quadratic programming. *Math. Program. Study* **22**, 332–349 (1982)

54. F. Liers, G. Pardella, Partitioning planar graphs: a fast combinatorial approach for max-cut. *Comput. Optim. Appl.* **51**, 323–344 (2012)
55. R.D. Luce, A.D. Perry, A method of matrix analysis of group structure. *Psychometrika* **14**, 95–116 (1949)
56. H.M. Markowitz, Portfolio selection. *J. Financ.* **7**, 77–91 (1952)
57. H.M. Markowitz, The optimization of a quadratic function subject to linear constraints. *Nav. Res. Logist. Q.* **3**, 111–133 (1956)
58. D.C. Mattis, Solvable spin systems with random interaction. *Phys. Lett.* **6A**, 412 (1976)
59. G.P. McCormick, Converting general nonlinear programming problems to separable nonlinear programming problems. Report T–267, The George Washington University, 1972
60. G.P. McCormick, Computability of global solutions to factorable nonconvex programs: part I - Convex underestimating problems. *Math. Program.* **10**, 147–175 (1976)
61. M. Niss, History of the Lenz-Ising model 1920–1950: from ferromagnetic to cooperative phenomena. *Arch. Hist. Exact Sci.* **59**, 267–318 (2005)
62. M. Niss, History of the Lenz-Ising model 1950–1965: from irrelevance to relevance. *Arch. Hist. Exact Sci.* **63**, 243–287 (2009)
63. M. Niss, History of the Lenz-Ising model 1965–1971: the role of a simple model in understanding critical phenomena. *Arch. Hist. Exact Sci.* **65**, 625–658 (2011)
64. M. Padberg, The Boolean quadric polytope: some characteristics, facets and relatives. *Math. Program.* **45**, 134–172 (1989)
65. P. Pandey, A.P. Punnen, The generalized vertex cover problem *Discret. Optim.* **30**, 121–143 (2018)
66. S.S. Petrova, A.D. Solo ve, The origin of the method of steepest descent. *Hist. Math.* **24**, 361–375 (1997)
67. A.P. Punnen, N. Kaur, Revisiting some classical explicit linearizations for the quadratic binary optimization problem. Research Report, Department of Mathematics, Simon Fraser University, 2021
68. A.P. Punnen, N. Kaur, On compact linearizations of the quadratic binary optimization problem. Research Report, Department of Mathematics, Simon Fraser University, 2021
69. A.P. Punnen, P. Pandey, M. Friesen, Representations of quadratic combinatorial optimization problems: a case study using the quadratic set covering problem. *Comput. Oper. Res.* **112**, 104769 (2019)
70. M. Raghavachari, On connections between zero-one integer programming and concave programming under linear constraints. *Oper. Res.* **17**, 680–684 (1969)
71. M. Raghavachari, Supplement. *Oper. Res.* **18**, 564–565 (1970)
72. J.M.W. Rhys, A selection problem of shared fixed costs and network flows. *Manag. Sci.* **17**, 200–207 (1970)
73. M. Rodriguez, I. Laptev, J. Sivic, J.-Y. Audibert, Density-aware person detection and tracking in crowds, in *2011 International Conference on Computer Vision, Barcelona* (2011), pp. 2423–2430
74. S. Rujikietgumjorn, R.T. Collins, Optimized pedestrian detection for multiple and occluded people, in *CVPR '13: Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 3690–3697
75. B. Simon, *The Statistical Mechanics of Lattice Gases*, vol. I (Princeton University Press, Princeton, 2014)
76. W.E. Smith, Various optimizers for single-stage production. *Nav. Res. Logist. Q.* **3**, 59–66 (1956)
77. D. Sornette, Physics and financial economics (1776–2014): puzzles, Ising and agent-based models. *Rep. Prog. Phys.* **77**(6), 062001, 28 pp. (2014)
78. R.G. Vickson, Two single machine sequencing problems involving controllable job processing times. *AIIE Trans.* **12**, 258–262 (1980)
79. R.G. Vickson, Choosing the job sequence and processing times to minimize total processing plus flow cost on single machines. *Oper. Res.* **28**, 1155–1167 (1980)

80. L.J. Watters, Reduction of integer polynomial programming problems to zero-one linear programming problems. *Oper. Res.* **15**, 1171–1174 (1967)
81. W.I. Zangwill, Media selection by decision programming. *J. Advert. Res.* **5**, 30–36 (1965)