The Bipartite Boolean Quadratic Programming Problem

by

Piyashat Sripratak

M.Math., University of Waterloo, 2009
B.Sc. (Hons.), Chulalongkorn University, 2006

Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Doctor of Philosophy

in the

Department of Mathematics
Faculty of Science

© Piyashat Sripratak  2014
SIMON FRASER UNIVERSITY
Fall 2014

All rights reserved.
However, in accordance with the Copyright Act of Canada, this work may be reproduced without authorization under the conditions for "Fair Dealing". Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.
APPROVAL

Name: Piyashat Sripratak

Degree: Doctor of Philosophy

Title: The Bipartite Boolean Quadratic Programming Problem

Examining Committee: Chair: Dr. Zhaosong Lu
Associate Professor

Dr. Abraham Punnen
Senior Supervisor
Professor

Dr. Tamon Stephen
Co-Supervisor
Associate Professor

Dr. Andrei Bulatov
Internal Examiner
Professor

Dr. Bala Krishnamoorthy
External Examiner
Associate Professor of Mathematics
Washington State University

Date Approved: November 25th, 2014
Partial Copyright Licence

The author, whose copyright is declared on the title page of this work, has granted to Simon Fraser University the non-exclusive, royalty-free right to include a digital copy of this thesis, project or extended essay[s] and associated supplemental files ("Work") (title[s] below) in Summit, the Institutional Research Repository at SFU. SFU may also make copies of the Work for purposes of a scholarly or research nature; for users of the SFU Library; or in response to a request from another library, or educational institution, on SFU’s own behalf or for one of its users. Distribution may be in any form.

The author has further agreed that SFU may keep more than one copy of the Work for purposes of back-up and security; and that SFU may, without changing the content, translate, if technically possible, the Work to any medium or format for the purpose of preserving the Work and facilitating the exercise of SFU’s rights under this licence.

It is understood that copying, publication, or public performance of the Work for commercial purposes shall not be allowed without the author’s written permission.

While granting the above uses to SFU, the author retains copyright ownership and moral rights in the Work, and may deal with the copyright in the Work in any way consistent with the terms of this licence, including the right to change the Work for subsequent purposes, including editing and publishing the Work in whole or in part, and licensing the content to other parties as the author may desire.

The author represents and warrants that he/she has the right to grant the rights contained in this licence and that the Work does not, to the best of the author’s knowledge, infringe upon anyone’s copyright. The author has obtained written copyright permission, where required, for the use of any third-party copyrighted material contained in the Work. The author represents and warrants that the Work is his/her own original work and that he/she has not previously assigned or relinquished the rights conferred in this licence.

Simon Fraser University Library
Burnaby, British Columbia, Canada

revised Fall 2013
Abstract

We consider the Bipartite Boolean Quadratic Programming Problem (BQP01), which generalizes the well-known Boolean Quadratic Programming Problem (QP01). The model has applications in graph theory, matrix factorization and bioinformatics, among others. BQP01 is NP-hard. The primary focus of this thesis is on studying algorithms and polyhedral structure from a linearization of its integer programming formulation.

We show that when the rank of the associated $m \times n$ cost matrix $Q$ is fixed, BQP01 can be solved in polynomial time. Further, for the rank one case, we provide an $O(n \log n)$ algorithm. The complexity reduces to $O(n)$ with additional assumptions. Further, we observe that BQP01 is polynomially solvable if $m = O(\log n)$. Similarly, when the minimum negative eliminator of $Q$ is $O(\log n)$, the problem is shown to be polynomially solvable.

We then develop several heuristic algorithms for BQP01 and analyze them using domination analysis. First, we give a closed-form formula for the average objective function value $A(Q, c, d)$ of all solutions. Computing the median objective function value however is shown to be NP-hard. We prove that any solution with objective function value no worse than $A(Q, c, d)$ dominates at least $2^{m+n-2}$ solutions and provide an upper bound for the dominance ratio of any polynomial time approximation algorithms for BQP01. Further, we show that some powerful local search algorithms could produce solutions with objective function value worse than $A(Q, c, d)$ and propose algorithms that guarantee a solution with objective function value no worse than $A(Q, c, d)$.

Finally, we study the structure of the polytope $BQP^{m,n}$ resulting from linearization of BQP01. We develop various approaches to obtain families of valid inequalities and facet-defining inequalities of $BQP^{m,n}$ from those of other related polytopes. These approaches include rounding coefficients, using the linear transformation between $BQP^{m,n}$ and the corresponding cut polytope, and applying triangular elimination.
Keywords: quadratic programming; 0-1 variables; polynomial algorithms; worst-case analysis; domination analysis; polytope
Acknowledgements

This thesis would not be completed without the help of many people. I would like to thank my supervisors Abraham Punnen and Tamon Stephen for guiding the writing of an academic work and working together on this research topic.

Moreover, it is a pleasure to thank all graduate students and postdoctoral fellows in the Operation Research group with whom I spent my graduate life. I am also appreciated their help on many technical issues while working on this thesis.

I am also grateful to the Royal Thai Government Scholarship who fully supports all expenses in my graduate study.

Last but not least, I would like to show my gratitude to my relatives and friends who encouraged me throughout my study in an overseas country.
# Contents

Approval ii  
Partial Copyright License iii  
Abstract iv  
Acknowledgements vi  
Contents vii  

List of Figures x

1 Introduction 1  
1.1 Equivalent Formulations 5  
1.2 BQP01 and Maximum Weight Cut in Bipartite Graphs 7  
1.3 BQP01 as a Bilinear Program 9  
1.4 Boolean Quadratic Programming Problem as BQP01 11  
1.5 QP01: Algorithms and Applications 13  
1.5.1 Polynomially Solvable Cases of QP01 13  
1.5.2 Exact and Heuristic Algorithms for QP01 14  
1.5.3 The Boolean Quadric Polytope 15  

2 Complexity and Polynomially Solvable Cases 18  
2.1 Problem with Fixed Parameter 19  
2.2 Problem with Special Structures on Objective Functions 21  
2.2.1 BQP01 with Fixed Rank $Q$ 22  
2.2.2 BQP01 with Rank One Matrix 28
2.2.3 BQP01 with the Objective Function in the Form \((a_0 + ax)(b_0 + by) + cx + dy\) where \(c = 0^T_m\) or \(d = 0^T_n\) .......................... 32

2.2.4 BQP01 with \(Q\) as an Additively Decomposable Matrix .................................. 33

2.2.5 BQP01 with \(Q\) as a \((2p + 1)\)-diagonal Matrix .................................... 36

3 Approximation Algorithms and Domination Analysis ............................................. 39

3.1 Average Value Based Analysis ................................................................. 40

3.1.1 Worse than Average Algorithms .......................................................... 42

3.2 No Worse than Average Algorithms ........................................................... 47

3.2.1 The Trivial Algorithm ........................................................................... 47

3.2.2 Rounding Algorithm ............................................................................. 48

3.2.3 No worse than Average One-pass Algorithm ......................................... 51

3.2.4 Median Objective Function Value .......................................................... 55

3.3 Domination Analysis .................................................................................... 58

4 Bipartite Boolean Quadric Polytope ................................................................. 66

4.1 Basic Properties of \(\text{BQP}^{m,n}\) .................................................................. 67

4.1.1 Trivial Facets of the Polytope ................................................................ 69

4.1.2 Properties as a Restriction of the Boolean Quadric Polytope ............... 74

4.1.3 Symmetry Theorem ............................................................................. 79

4.1.4 Partial Linear Relaxation of \(\text{BQP}^{m,n}\) ............................................... 82

4.2 Families of Valid Inequalities Obtained from Rounding Coefficients ........... 84

4.3 Odd-Cycle Inequality ............................................................................... 89

4.4 Valid Inequalities Obtained from Relationship between a Cut Polytope and \(\text{BQP}^{m,n}\) ................................................................. 93

4.4.1 Valid Inequalities Obtained from the Cut Polytope Corresponding to a Biclique with One Added Vertex ..................................... 95

4.4.2 Valid Inequalities Obtained from the Cut Polytope Corresponding to a Biclique with Two Added Vertices ................................. 100

4.5 Facets and Valid Inequalities Obtained from Triangular Elimination .......... 105

5 Conclusion ....................................................................................................... 110
Bibliography

Appendix A  Special Case on Rank One Matrix

Appendix B  BQP01 with $Q$ as a $(2p + 1)$-diagonal Matrix

Appendix C  Supplement on the Bipartite Boolean Quadratic Polytope
  C.1  Partial Linear Relaxation of BQP$^{m,n}$
  C.2  Families of Valid Inequalities Obtained from Rounding Coefficients
  C.3  Examples on Applying Triangular Elimination
    C.3.1  Hypermetric Inequalities
    C.3.2  Clique-Web Inequality
List of Figures

1.1 Graph $G = (I, J, E)$ when $I = \{1, 2, 3, 4\}$, $J = \{1, 2, 3, 4, 5, 6\}$ and a biclique $S'$ of $G$. ................................................................. 3

1.2 A complete bipartite graph $G$ and a maximum weight cut in $G$ with $U_1 = \{2, 3\}$ and $U_2 = \{2\}$. ................................................................. 8

2.1 An example of $h_1(\lambda)$ and $h_2(\mu)$ when $a = [2, 2, -3, 4, -2]$, $c = [4, 5, 6, 10, 5]$, $b = [1, 1, -4, 0, -1, -2, 1]$, $d = [5, -2, 3, 3, 4, 0, 2]$. ........................................ 31
Chapter 1

Introduction

An optimization problem can be stated as an ordered pair $P = (f, F)$ where $f$ represents the objective function and $F$ represents the family of feasible solutions. Solvability of the problem $P$ depends on the nature of $f$ as well as $F$.

When $f(u) = u^T Qu + cu$, where $Q$ is an $n \times n$ matrix, $u \in \mathbb{R}^n$ and $c$ is a row vector in $\mathbb{R}^n$, the problem $(f, F)$ is called a quadratic programming problem. In addition, if $F$ is a polyhedral set, we have a quadratic program with linear constraints. Such a problem can be solved effectively when $f$ is convex. In this case $(f, F)$ enjoys the property that a local optimum is also a global optimum.

If $f$ is concave, then $(f, F)$ has the property that there exists an optimal solution which corresponds to an extreme point of $F$. A local optimum however need not be a global optimum in this case. However, the extreme point optimality brings combinatorial structure into the problem and often such problems are intractable in nature.

In this thesis, we investigate a special quadratic programming problem where feasible solutions are extreme points of the unit cube in $\mathbb{R}^{m+n}$. Let $Q = (q_{ij})$ be an $m \times n$ matrix, $c = (c_1, \ldots, c_m)$ be a row vector in $\mathbb{R}^m$, $d = (d_1, \ldots, d_n)$ be a row vector in $\mathbb{R}^n$ and $c_0$ is a constant. Then the Bipartite Boolean Quadratic Programming Problem (BQP01) can be
defined as

$$\text{Maximize } f(x, y) = x^T Q y + c x + d y + c_0$$

Subject to $x \in \{0,1\}^m$, $y \in \{0,1\}^n$.

Note that the constant $c_0$ can be omitted for optimization, but we retain it to show the relationship between various equivalent formulations to be discussed later in this chapter.

A graph theoretic interpretation of BQP01 can be given as follows. Suppose $G(I, J, E)$ be a complete bipartite graph, where $I = \{1, \ldots, m\}$ and $J = \{1, \ldots, n\}$. Let $q_{ij}$ be the weight of the edge $\{i, j\}$ where $i \in I$ and $j \in J$, $c_i$ be the weight of vertex $i \in I$ and $d_j$ be the weight of vertex $j \in J$. Let $S(I', J', E')$ be a subgraph of $G$. The weight of $S$ is the total weight of its vertices and edges. Selecting a maximum weight complete bipartite subgraph of $G$ is precisely the BQP01. Throughout this thesis, we denote $ij$ the edge between vertices $i$ and $j$ for convenience. Moreover, if $ij$ is an edge in a bipartite graph $G(I, J, E)$, the first entry $i$ stands for the vertex in $I$ and the second entry $j$ stands for the vertex in $J$.

The model BQP01 has many applications. Consider a bipartite graph $G(I, J, E)$. A subgraph $S'(I', J', E')$ of $G$ is said to be a biclique if $S'$ is a complete bipartite graph.

Define

$$q_{ij} = \begin{cases} 1 & \text{if } ij \in E, \\ -M & \text{otherwise,} \end{cases}$$

where $M$ is a large positive number. Also choose $c$ and $d$ to be zero vectors of size $m$ and $n$, respectively. Then the Maximum Biclique Problem (MBCP) in $G$ can be solved as BQP01 using $Q, c, d$ as defined above.

If each edge $ij$ of $G$ has a corresponding weight $w_{ij}$, then the Maximum Weighted Biclique Problem (MWBCP) is to choose a biclique in $G$ of maximum weight. By modifying the definition of $q_{ij}$ as

$$q_{ij} = \begin{cases} w_{ij} & \text{if } ij \in E, \\ -M & \text{otherwise,} \end{cases}$$
Figure 1.1: Graph $G = (I, J, E)$ when $I = \{1, 2, 3, 4\}$, $J = \{1, 2, 3, 4, 5, 6\}$ and a biclique $S'$ of $G$.

we can solve the MWBCP as a BQP01.

The problem MBCP and MWBCP have been studied by various authors. Applications of these models include data mining, clustering and bioinformatics [24, 110].

The cut-norm of a matrix $A = (a_{ij})_{n \times n}$ is defined as

$$\|A\|_c = \max_{I' \subseteq N} \max_{J' \subseteq N} \left| \sum_{i \in I'} \sum_{j \in J'} a_{ij} \right|$$

where $N = \{1, 2, \ldots, n\}$. The cut-norm $\|A\|_c$ can be obtained by solving

$$\text{Maximize } \left| \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}x_iy_j \right|$$

Subject to $x \in \{0, 1\}^m$, $y \in \{0, 1\}^n$.

This problem can be solved using the BQP01 model [7].
Approximating matrices using low-rank \{0,1\}-matrices has applications in mining discrete patterns in binary data [103] and various clustering applications. Many authors considered the problem of approximating a matrix by a rank-one binary matrix [33, 65, 66, 103]. This problem can be formulated as a BQP01. Let \( A = (a_{ij}) \) be an \( m \times n \) matrix. An \( m \times n \) matrix \( B = (b_{ij}) \) is called a rank-one binary matrix if \( b_{ij} = u_i v_j \) and \( u_i, v_j \in \{0,1\} \) for all \( i = 1,\ldots,m \) and \( j = 1,\ldots,n \). Given the matrix \( A \), the goal is to find an \( m \times n \) rank-one binary matrix \( B = (b_{ij}) \) such that \( \sum_{i=1}^{m} \sum_{j=1}^{n} (a_{ij} - u_i v_j)^2 \) is minimized. We call \( B \) a rank one approximation of \( A \). Since \( u_i \) and \( v_j \) are binary variables,

\[
(a_{ij} - u_i v_j)^2 = a_{ij}^2 - 2 a_{ij} u_i v_j + u_i^2 v_j^2 = a_{ij}^2 - 2 a_{ij} u_i v_j + u_i v_j = a_{ij}^2 + (1 - 2 a_{ij}) u_i v_j.
\]

Thus, this rank one approximation matrix \( B \) can be obtained by solving the BQP01 with \( q_{ij} = 2 a_{ij} - 1 \) and \( c_i = d_j = c_0 = 0 \) for \( i = 1,\ldots,m \) and \( j = 1,\ldots,n \).

Here we give an example of using BQP01 to find a rank one approximation matrix. Consider a 6\times6 matrix of rank 6

\[
A = \begin{bmatrix}
0.55 & 1.1 & 2 & 0.5 & 1.5 & -1 \\
0.5 & 0.4 & 1.55 & 0.75 & 0.5 & 0.7 \\
0 & 0.75 & 0.5 & -1 & 0.5 & -0.5 \\
0.5 & 0 & -1.4 & 3.25 & 0.25 & 0.5 \\
1.75 & 0.5 & 1.55 & -0.6 & 0.25 & 1 \\
0.45 & 0 & 0.5 & 0.8 & 0.5 & 0.65
\end{bmatrix}.
\]

Then a rank one binary approximation matrix of \( A \) can be obtained by solving the BQP01 with

\[
Q = \begin{bmatrix}
0.1 & 1.2 & 3 & 0 & 2 & -3 \\
0 & -0.2 & 2.1 & 0.5 & 0 & 0.4 \\
-1 & 0.5 & 0 & -3 & 0 & -2 \\
0 & -1 & -3.8 & 5.5 & -0.5 & 0 \\
2.5 & 0 & 2.1 & -2.2 & -0.5 & 1 \\
-0.1 & -1 & 0 & 0.6 & 0 & 0.3
\end{bmatrix}.
\]
and \( c_i = d_j = c_0 = 0 \) for \( i = 1, \ldots, 6 \) and \( j = 1, \ldots, 6 \). An optimal solution of this problem is \( x^* = [1 \ 1 \ 0 \ 0 \ 1 \ 0]^T \) and \( y^* = [1 \ 1 \ 1 \ 0 \ 1 \ 0]^T \). Thus, \( B = x^*(y^*)^T \) is a rank one approximation matrix of \( A \).

Other application areas of BQP01 include correlation clustering [24, 110], bioinformatics [24, 110], approximating matrices using \( \{-1, 1\} \) entries [103], etc. Depending on the application areas, the problem BQP01 has been described in various equivalent forms. Let us now discuss these equivalent formulations of BQP01.

### 1.1 Equivalent Formulations

In many earlier descriptions of BQP01, the linear term \( cx \) and \( dy \) are not explicitly considered. We call a BQP01 with \( c_i = d_j = c_0 = 0 \) for \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \) the homogenous form and denoted by BQP01(H). Interestingly, any BQP01 can be formulated as an equivalent BQP01(H) by introducing two new variables. Let \( \bar{Q} = \begin{bmatrix} Q & c^T \\ d & M \end{bmatrix} \). BQP01 can be represented as

\[
\begin{align*}
\text{Maximize} & \quad \bar{x}^T \bar{Q} \bar{y} \\
\text{Subject to} & \quad \bar{x} \in \{0, 1\}^{m+1}, \bar{y} \in \{0, 1\}^{n+1}, \\
& \quad \bar{x}_{m+1} = 1, \bar{y}_{n+1} = 1,
\end{align*}
\]

where \( \bar{x}_{m+1} \) and \( \bar{y}_{n+1} \) are the \( (m + 1)^{th} \) and \( (n + 1)^{th} \) components of \( \bar{x} \) and \( \bar{y} \), respectively, where \( M = 0 \).

We can drop the conditions \( \bar{x}_{m+1} = 1 \) and \( \bar{y}_{n+1} = 1 \) by choosing \( M \) in \( \bar{Q} \) to be a large positive number. Hence, we obtain

\[
\text{BQP01(H): Maximize } \bar{x}^T \bar{Q} \bar{y} \\
\text{Subject to } \quad \bar{x} \in \{0, 1\}^{m+1}, \bar{y} \in \{0, 1\}^{n+1}. 
\]

BQP01(H) is equivalent to BQP01 in the sense that they have the same optimal set. If we consider the set of all optimal solutions of BQP01(H), dropping the last entry of \( \bar{x} \) and \( \bar{y} \).
in each solution in this set yields the set of all optimal solutions of BQP01.

In some applications of BQP01, the variables are restricted to \{-1, 1\} instead of \{0, 1\}. We denote such an instance by BQP(-1,1), which can be stated as

\[
\text{BQP(-1,1): Maximize } x^T Q y + c x + d y + c_0 \\
\text{Subject to } x \in \{-1, 1\}^m, y \in \{-1, 1\}^n.
\]

Any instance of BQP(-1,1) can be formulated as a BQP01. For any positive integer \(k\), let \(1_k\) be the vector in \(\mathbb{R}^k\) whose entries are all one. Substitute \(x = 2w - 1_m\) and \(y = 2z - 1_n\) in a BQP(-1,1) instance, we obtain

\[
\text{Maximize } w^T \tilde{Q} z + \tilde{c} w + \tilde{d} z + \tilde{c}_0 \\
\text{Subject to } w \in \{0, 1\}^m, z \in \{0, 1\}^n,
\]

where \(\tilde{Q} = 4Q\), \(\tilde{c} = 2(c - (Q1_n)^T)\), \(\tilde{d} = 2(d - 1_m^TQ)\) and \(\tilde{c}_0 = 1_m^TQ1_n - c1_m - d1_n + c_0\).

Conversely, any instance of BQP01 can be formulated as a BQP(-1,1). Let \(x = (w + 1_m)/2\) and \(y = (z + 1_n)/2\) in a BQP01 instance, we have

\[
\text{Maximize } w^T \hat{Q} z + \hat{c} w + \hat{d} z + \hat{c}_0 \\
\text{Subject to } w \in \{-1, 1\}^m, z \in \{-1, 1\}^n,
\]

where \(\hat{Q} = \frac{1}{4}Q\), \(\hat{c} = \frac{1}{4}(Q1_n)^T + \frac{1}{2}c\), \(\hat{d} = \frac{1}{4}1_m^TQ + \frac{1}{2}d\) and \(\hat{c}_0 = \frac{1}{4}1_m^TQ1_n + \frac{1}{2}c1_m + \frac{1}{2}d1_n + c_0\).

If we drop the linear and constant terms from the objective function of BQP(-1,1), we get the homogenous version of the problem, which is denoted by BQP(-1,1,H). Let \(\bar{Q}\) be defined as in the homogenous version of BQP01, \(\bar{x} \in \{-1, 1\}^{m+1}\) and \(\bar{y} \in \{-1, 1\}^{n+1}\). BQP01 is equivalent to

\[
\text{Maximize } f(\bar{x}, \bar{y}) = \bar{x}^T \bar{Q} \bar{y} \\
\text{Subject to } \bar{x} \in \{-1, 1\}^{m+1}, \bar{y} \in \{-1, 1\}^{n+1}, \\
\bar{x}_{m+1} = 1, \bar{y}_{n+1} = 1.
\]

We observe that the condition \(\bar{x}_{m+1} = 1\) and \(\bar{y}_{n+1} = 1\) can be replaced by \(\bar{x}_{m+1} = \bar{y}_{n+1}\). If \(\bar{x}_{m+1} = \bar{y}_{n+1}\), then either \(\bar{x}_{m+1} = \bar{y}_{n+1} = 1\) or \(\bar{x}_{m+1} = \bar{y}_{n+1} = -1\). If the latter holds, we
consider \((-\bar{x}, -\bar{y})\) which also satisfies \(-\bar{x}_{m+1} = -\bar{y}_{n+1} = 1\) and \(f(\bar{x}, \bar{y}) = f(-\bar{x}, -\bar{y})\). Thus, the problems using these two different conditions have the same set of optimal solutions.

Moreover, the condition \(\bar{x}_{m+1} = \bar{y}_{n+1}\) can be dropped by choosing \(M\) in \(\bar{Q}\) to be a large positive number. Then we obtain a homogeneous version of BQP(-1,1):

\[
\text{BQP(-1,1,H):} \quad \text{Maximize } \bar{x}^T \bar{Q} \bar{y} \\
\text{Subject to } \bar{x} \in \{-1, 1\}^{m+1}, \bar{y} \in \{-1, 1\}^{n+1}.
\]

Alon and Naor [7] showed that BQP(-1,1,H) is MAX SNP hard. Therefore, BQP01 is MAX SNP hard as well.

### 1.2 BQP01 and Maximum Weight Cut in Bipartite Graphs

Consider a graph \(G(V, E)\) with weight \(w_{ij}\) assigned on edge \(ij\). A cut \((U, W)\) of \(G\) is a partition of \(V\) and the total weight of the cut is \(\sum_{i \in U, j \in W} w_{ij}\). **Maximum Weight Cut Problem** (MaxCut) is to find a cut with maximum weight. If we restrict the underlying graph to be the bipartite graph \(G(I, J, E)\), we obtain the **Bipartite Maximum Weight Cut Problem** (B-MaxCut). If the given edge weights are nonnegative, B-MaxCut is a trivial problem since \((I, J)\) is clearly an optimal cut. Thus, the problems when the edge weights contains both positive and negative values are the interesting ones.

Consider a bipartite graph \(G(I, J, E)\) where \(I = \{1, 2, \ldots, m\}\) and \(J = \{1, 2, \ldots, n\}\). Two vectors \(x \in \{-1, 1\}^m\) and \(y \in \{-1, 1\}^n\) determine a cut \((U_1 \cup U_2, W_1 \cup W_2)\) in \(G\) if and only if \(U_1 = \{i \in I : x_i = 1\}\), \(W_1 = \{i \in I : x_i = -1\}\), \(U_2 = \{j \in J : y_j = 1\}\), and \(W_2 = \{j \in J : y_j = -1\}\). We call \((x, y)\) the **incidence vector** of the cut \((U_1 \cup U_2, W_1 \cup W_2)\). Let \(q_{ij}\) be the weight of the edge \(ij\) in \(E\). The weight of the cut \((U_1 \cup U_2, W_1 \cup W_2)\) is

\[
\sum_{i \in U_1, j \in W_2} q_{ij} + \sum_{i \in W_1, j \in U_2} q_{ij} = \sum_{x_i \neq y_j} q_{ij}.
\]

For example, the incidence vector of the max cut in Figure 1.2 is \(x = (-1, 1, 1)\) and \(y = (-1, 1, -1, -1)\). The cut has weight 13.
Theorem 1.1. BQP01 and B-MaxCut are equivalent in the sense that:

1. Given an instance of BQP01, one can construct a complete bipartite graph $G$ such that an optimal solution to the B-MaxCut problem on $G$ gives an optimal solution to BQP01.

2. Given an instance of B-MaxCut on a bipartite graph $G(I,J,E)$, one can construct an instance of BQP01 with an $m \times n$ cost matrix $Q$ such that an optimal solution to the BQP01 gives an optimal solution to the B-MaxCut problem on $G$.

Proof. Recall that in Section 1.1, we show that BQP01 is equivalent to BQP(-1,1,H). Without loss of generality, we assume that BQP01 is given in the equivalent BQP(-1,1,H) form. Note that

$$f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j = \sum_{x_i = y_j} q_{ij} (1 \cdot 1) + \sum_{x_i \neq y_j} q_{ij} (1 \cdot -1)$$

$$= \sum_{x_i = y_j} q_{ij} - \sum_{x_i \neq y_j} q_{ij} = \left( \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} \right) - \sum_{x_i \neq y_j} q_{ij} - \sum_{x_i = y_j} q_{ij}$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} - 2 \sum_{x_i \neq y_j} q_{ij}.$$ 

Since $\sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij}$ is a constant, maximizing $f(x, y)$ is equivalent to maximizing $-2 \sum_{x_i \neq y_j} q_{ij}$. Thus, we can solve BQP(-1,1,H) by solving the B-MaxCut problem on a complete bipartite graph $G$ and a maximum weight cut in $G$ with $U_1 = \{2, 3\}$ and $U_2 = \{2\}$. 

Figure 1.2: A complete bipartite graph $G$ and a maximum weight cut in $G$ with $U_1 = \{2, 3\}$ and $U_2 = \{2\}$. 

Theorem 1.1. BQP01 and B-MaxCut are equivalent in the sense that:

1. Given an instance of BQP01, one can construct a complete bipartite graph $G$ such that an optimal solution to the B-MaxCut problem on $G$ gives an optimal solution to BQP01.

2. Given an instance of B-MaxCut on a bipartite graph $G(I,J,E)$, one can construct an instance of BQP01 with an $m \times n$ cost matrix $Q$ such that an optimal solution to the BQP01 gives an optimal solution to the B-MaxCut problem on $G$. 

Proof. Recall that in Section 1.1, we show that BQP01 is equivalent to BQP(-1,1,H). Without loss of generality, we assume that BQP01 is given in the equivalent BQP(-1,1,H) form. Note that

$$f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j = \sum_{x_i = y_j} q_{ij} (1 \cdot 1) + \sum_{x_i \neq y_j} q_{ij} (1 \cdot -1)$$

$$= \sum_{x_i = y_j} q_{ij} - \sum_{x_i \neq y_j} q_{ij} = \left( \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} \right) - \sum_{x_i \neq y_j} q_{ij} - \sum_{x_i = y_j} q_{ij}$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} - 2 \sum_{x_i \neq y_j} q_{ij}.$$ 

Since $\sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij}$ is a constant, maximizing $f(x, y)$ is equivalent to maximizing $-2 \sum_{x_i \neq y_j} q_{ij}$. Thus, we can solve BQP(-1,1,H) by solving the B-MaxCut problem on a complete bipartite
graph $K_{m,n}$ with weight $w_{ij} = -2q_{ij}$.

As for the second part of the theorem, we consider the B-MaxCut problem on $G$ with edge weights $w_{ij}$ for $ij \in E$. Let $(U_1 \cup U_2, W_1 \cup W_2)$ be a cut in $G$ and $(x,y)$ be the corresponding incidence vector. Then $w(U,W) = \sum_{x_i \neq y_j} w_{ij}$ is the value of the cut $(U_1 \cup U_2, W_1 \cup W_2)$. Thus, one can verify that

$$w(U,W) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij}x_iy_j.$$ 

Similar to the first part, maximizing $w(U,W)$ is equivalent to solving a BQP(-1,1,H) with $q_{ij} = -w_{ij}/2$.

From the example in Figure 1.2, solving B-MaxCut on this instance is equivalent to solving BQP(-1,1,H) with $q_{ij} = -w_{ij}/2$.

The maximum weight cut provided corresponds to the order pair $x = (-1,1,1)$ and $y = (-1,1,-1,-1)$ which is an optimal solution for the BQP(-1,1,H) described above with objective function value $-1/2 \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij}x_iy_j = 9.5$. Note that $1/2 \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} - 1/2 \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij}x_iy_j = 3.5 + 9.5 = 13$ which is the weight of the given cut.

### 1.3 BQP01 as a Bilinear Program

A Bilinear Program (BLP) [62] is a quadratic programming problem which can be formulated as

$$\text{BLP: Maximize } x^TQy + cx + dy$$

$$\text{Subject to } Ax = b, x \geq 0,$$

$$A'y = b', y \geq 0.$$ 

Consider the constraint set defined by the $x$ variables represents the set $X = \{x \in \mathbb{R}^n | Ax = b, x \geq 0\}$ and the constraint set defined by the $y$ variables represents the set $Y = \{y \in \mathbb{R}^m | A'y = b', y \geq 0\}$. 

$$Q = \begin{bmatrix}
0.5 & -0.5 & 0 & 0 \\
0 & 0 & -2 & -2.5 \\
-1 & 2.5 & 0 & -0.5
\end{bmatrix}.$$
Konno [62] proved the existence of an optimal solution where $x$ and $y$ are extreme points of $X$ and $Y$, respectively. The proof of this observation is very simple and follows from the fundamental theorem of linear programming.

**Theorem 1.2.** [62] If $X$ and $Y$ are nonempty and bounded, then BLP has an optimal solution $(x^*, y^*)$ where $x^*$ is a basic feasible solution of the constraint equations defining $X$ and $y^*$ is a basic feasible solution of the constraint equations defining $Y$.

We consider a special case of the bilinear program called the *Unconstrained Bilinear Program* (UBLP), which is closely related to BQP01. The problem UBLP can be stated as

\[
\begin{align*}
\text{UBLP:} & \quad \text{Maximize } x^T Q y + cx + dy + c_0 \\
\text{Subject to} & \quad x \in [0, 1]^m, y \in [0, 1]^n.
\end{align*}
\]

Note that UBLP is a relaxation of BQP01 where the set $\{0, 1\}$ in the constraints of BQP01 is replaced by the interval $[0, 1]$ in UBLP. Comparing to BLP, $X = [0, 1]^m$ and $Y = [0, 1]^n$. It follows from Theorem 1.2 that there exists a binary optimal solution for UBLP. This theorem leads to the following relationship between BQP01 and UBLP.

**Corollary 1.3.** BQP01 is equivalent to UBLP in the sense that they have the same optimal objective function value.

Even though the optimal objective function value of these two problems are the same, the feasible sets are not the same. We can solve BQP01 by using algorithms for UBLP that focus on basic feasible solutions. Any algorithm that gives a non-basic optimal solution can also be used, but one needs to convert such a solution to a basic feasible solution to retain feasibility of BQP01. But this can be accomplished in polynomial time using standard methods. To the best of our knowledge, UBLP, exploiting its special structure, is not studied in the literature, except in some isolated cases.
1.4 Boolean Quadratic Programming Problem as BQP01

Boolean Quadratic Programming Problem (QP01) is a well studied combinatorial optimization problem with various applications [19]. The problem can be stated mathematically as

\[
\begin{align*}
\text{Maximize} & \quad u^T Q' u + c'u + c'_0 \\
\text{Subject to} & \quad u \in \{0,1\}^n,
\end{align*}
\]

where \(Q' = (q'_{ij})\) is an \(n \times n\) matrix, \(c'\) is a row vector of length \(n\), \(u\) is a column vector of size \(n\) and \(c'_0\) is a constant. QP01 is known to be NP-hard.

BQP01 can be viewed as a special case of QP01 in a higher dimension where

\[
Q' = \begin{bmatrix}
0_{m \times m} & Q \\
0_{n \times m} & 0_{n \times n}
\end{bmatrix},
\]

\[
c' = [c \ d]
\]

and

\[
u = \begin{bmatrix}
x \\
y
\end{bmatrix}.
\]

Here \(0_{h \times k}\) are zero matrices of size \(h \times k\).

In the above formulation, the size of the quadratic cost matrix is increased significantly. Thus, such a formulation, although useful since we can use existing results of QP01, could have practical limitations. Therefore, it is important to investigate BQP01 separately to derive results that exploit the problem structure.

Interestingly, QP01 can be formulated as a BQP01 as well. We denote by \(1_n\) the vector in \(\mathbb{R}^n\) whose entries are one. Choosing

\[
Q = Q' + 2MI_n, \quad c = \frac{1}{2}c' - M1_n \quad \text{and} \quad d = \frac{1}{2}c' - M1_n,
\]

where \(I_n\) is the identity matrix of size \(n\) and \(M\) is a very large number, QP01 can be reduced to BQP01. This transformation does not require enlarged problem size, making BQP01 a more general model than QP01.

Besides, for a special case when QP01 has a positive semidefinite cost matrix, QP01 and BQP01 are equivalence in the sense that they have the same objective function value.
Consider a quadratic programming problem
\[
\begin{align*}
\text{Maximize} & \quad u^T Q' u + c'u \\
\text{Subject to} & \quad Au = b, u \geq 0,
\end{align*}
\]
where \( Q' \in \mathbb{R}^{n \times n} \) is a symmetric positive semidefinite matrix, and a bilinear program
\[
\begin{align*}
\text{Maximize} & \quad x^T Q'y + \frac{1}{2}c'x + \frac{1}{2}c'y \\
\text{Subject to} & \quad Ax = b, x \geq 0, \\
& \quad Ay = b, y \geq 0.
\end{align*}
\]
Konno [62] proved that if \((x^*, y^*)\) is an optimal solution of the bilinear problem, then both \(x^*\) and \(y^*\) are optimal for the quadratic programming problem. We use this fact to show that we can obtain an optimal solution for QP01 via solving BQP01.

**Proposition 1.4.** If \((x^*, y^*)\) be an optimal solution of BQP01 problem with \(m = n\), \(Q = Q'\) is positive semidefinite and \(c = d = c'/2\), then \(x^*\) and \(y^*\) are optimal solutions for QP01.

**Proof.** By the equivalence between BQP01 and UBLP, \((x^*, y^*)\) is optimal for
\[
\begin{align*}
\text{Maximize} & \quad x^T Q'y + \frac{1}{2}c'x + \frac{1}{2}c'y \\
\text{Subject to} & \quad x, y \in [0,1]^n.
\end{align*}
\]
Thus, the relationship between a quadratic programming problem and a bilinear program mentioned previously shows that \(x^*\) and \(y^*\) are also optimal for
\[
\begin{align*}
\text{Maximize} & \quad u^T Q'u + c'u \\
\text{Subject to} & \quad u \in [0,1]^n.
\end{align*}
\]
Since \((x^*, y^*)\) is a solution of BQP01, each entry of \(x^*\) and \(y^*\) takes value in \(\{0,1\}\). Hence, \(x^*\) and \(y^*\) are also optimal for the QP01.

Applications of QP01 are abundant in literature. The model can be used in solving various problems in graph theory [52, 60, 61, 86, 87] and combinatorial optimization [5, 70]. Its applications also appear in various areas such as neural networks [90], financial analysis [77, 114], communication [54, 75], VLSI design [20], statistical mechanics [14, 52, 84, 105], among others. Because of the generality of the BQP01 model, each of the above applications are also relevant to BQP01 as well.
1.5 QP01: Algorithms and Applications

To further emphasize the importance of the QP01 model, it may be noted that many combinatorial optimizations can be represented as

\[
\begin{align*}
\text{Minimize} & \quad u^T Q'u \\
\text{Subject to} & \quad Au = b, \\
& \quad u \in \{0, 1\}^n.
\end{align*}
\]

Note that the constraints are equations. Kochenberger et al. [60] showed that this problem can be formulated as a QP01 (and hence BQP01). The basic idea is to introduce a penalty term \((Au - b)^T(Au - b)\) in the objective function with a positive weight \(\alpha\). Note that \(u^T Q'u + \alpha(Au - b)^T(Au - b) = u^T \hat{Q}'u + c\) for some \(\hat{Q}'\) that can be easily derived from \(Q', A\) and \(b\) by using the fact that \(u^2_i = u_i\). Computational results for solving various problems using this transformation by employing heuristic algorithms designed for QP01 were discussed in [60]. As an immediate application of this transformation, the well known Set Partitioning Problem [70], equality constrained 0-1 knapsack problem and various exact combinatorial optimizations [5, 37, 60, 61] can be formulated as a QP01.

Our work on BQP01 was considerably influenced by corresponding results for QP01. We primarily focus in this thesis on polynomially solvable (exact and approximate) special cases of BQP01 and study the polytope arising from linearization on the binary quadratic programming formulation of BQP01. Hence, it is interesting and relevant to review corresponding results for QP01.

1.5.1 Polynomially Solvable Cases of QP01

Most of the polynomially solvable cases of QP01 consider restrictions on the cost matrix \(Q'\). If all off-diagonal elements of \(Q'\) are nonnegative, the problem is polynomially solvable due to the total unimodularity of the coefficient matrix of a specific linearization [35]. It was shown in various papers that when \(Q'\) is a fixed rank positive semidefinite matrix with \(c' = 0\), QP01 can be solved in polynomial time [6, 28, 35, 58]. A polynomial time algorithm for QP01 with \(Q'\) being a tridiagonal matrix was proposed by D. Li et al. [35].
Some of the polynomially solvable special cases consider the restriction on the graph $G$ obtained from $Q'$ by introducing an edge $ij$ for all nonzero $q'_{ij}$. QP01 can be viewed as finding an induced subgraph with maximum weight from $G$ when edge $ij$ has weight $q'_{ij}$ and vertex $i$ has weight $c'_i$. Thus, QP01 is expressible in monadic second-order logic and Courcelle’s theorem [25] states that the problem can be solved in linear time when $G$ has bounded treewidth. Li et al. [72] showed that QP01 defined by a logic circuit can be solved in polynomial time and gave a strategy to identify a polynomially solvable subclass of QP01 using the Lagragian dual.

1.5.2 Exact and Heuristic Algorithms for QP01

Since QP01 is known to be NP-hard, algorithmic research on the problem was primarily concentrated on heuristics as well as enumerative type exact algorithms.

Heuristic algorithms can be classified in general as construction heuristics and improvement heuristics. While construction heuristics build a solution in sequence, improvement heuristics start with a solution and tries to improve the solution. One-pass heuristics [22, 35], including most of greedy algorithms [71, 78], take short computation times, but the solution quality is relatively poor. Heuristics with short computational time are usually used to obtain an initial solution for more complex heuristics. Many heuristic algorithms in literature for QP01 are concentrated on tabu search [5, 36, 38, 60, 61, 70, 85] and evolutionary methods [9, 22, 23, 73, 79]. Hasan et al. [55] reported comparison between simulated annealing, genetic algorithm, and tabu search heuristic for QP01. Their computational results show that the quality of the solutions from tabu search is relatively high, but it takes significantly longer running time to achieve this quality. Under the same computational time, genetic algorithm gave the best result. However, Glover et al. [35] noted that the quality of the output from the genetic algorithm significantly decreased when applied to the problems with dense $Q'$ matrix. Other alternative methods for finding a heuristic solution for the problem is using Mixed Integer Quadratic Programming (MIQP) solvers [17], heuristic method based on a global equilibrium search framework [88], and combination of neural network and estimation of distribution of the data [113].
Exact algorithms for QP01 are mostly of enumerative type involving branch and bound or branch and cut methods [15, 31, 64, 67, 77, 86, 89, 98]. Many of these algorithms use bounds generated by various relaxations including several linearization methods. The first linearization methods for QP01 was proposed by Fortet [29] by introducing additional integer variables. Glover and Woolsey [41] developed another linearization strategy where additional variables do not need to be integers. Many linearizations studied in literature have linear number of additional constraints and variables [34, 59, 82]. Other notable works in the area are due to Adams and Sherali [3, 4]. For additional information on linearization we refer to [1, 2, 44, 45, 53].

1.5.3 The Boolean Quadric Polytope

Let \( x^1, \ldots, x^k \) be points in \( \mathbb{R}^n \). Then \( x \) is a convex combination of \( x^1, \ldots, x^k \) if \( x = \sum_{i=1}^{k} \lambda_i x^i \), \( \sum_{i=1}^{k} \lambda_i = 1 \) and \( \lambda_i \geq 0 \) for \( i = 1, \ldots, k \). Given \( X \subseteq \mathbb{R}^n \), the convex hull of \( X \) is defined as

\[
\text{conv}(X) = \{x : \exists \{x^1, \ldots, x^k\} \subseteq X, x \text{ is a convex combination of } x^1, \ldots, x^k\}.
\]

A subset \( P \) of \( \mathbb{R}^n \) is a polytope if there exists a finite set \( X \subseteq \mathbb{R}^n \) such that \( P = \text{conv}(X) \).

Let \( a \) be a row vector in \( \mathbb{R}^n \) and \( a_0 \) be a constant. An inequality \( ax \leq a_0 \) is a valid inequality for a polytope \( P \) if \( ax \leq a_0 \) for all \( x \in P \). The points \( x^1, \ldots, x^k \) in \( \mathbb{R}^n \) are said to be affinely independent if the \( k \) vectors \( (x^1, 1), \ldots, (x^k, 1) \) are linearly independent. The dimension of \( P \), denoted \( \dim(P) \), is the maximum number of affinely independent points in \( P \) minus one. \( F \) is a face of \( P \) if there exists a valid inequality \( ax \leq a_0 \) such that \( F = \{ x \in P : ax = a_0 \} \). Then the valid inequality \( ax \leq a_0 \) is said to define the face \( F = \{ x \in P : ax = a_0 \} \). A face \( F \) of \( P \) is a facet if \( \dim(F) = \dim(P) - 1 \). If a face \( F = \{ x \in P : ax = a_0 \} \) is a facet, then \( ax \leq a_0 \) is called a facet-defining inequality of \( P \).
QP01 can be formulated as an integer linear programming problem by introducing additional constraints and variables $v \in \{0, 1\}^{(n^2-n)/2}$ where $v_{ij} = u_i u_j$ as

$$\text{Maximize } \sum_{i<j} q'_{ij} v_{ij} + c'u + c_0$$

Subject to

$$u_i + u_j - v_{ij} \leq 1,$$  \hspace{1cm} (1.1)

$$-u_i + v_{ij} \leq 0,$$  \hspace{1cm} (1.2)

$$-u_j + v_{ij} \leq 0,$$  \hspace{1cm} (1.3)

$$-v_{ij} \leq 0,$$  \hspace{1cm} (1.4)

$$u_i, v_{ij} \text{ integer, for all } 1 \leq i < j \leq n.$$  \hspace{1cm} (1.5)

Padberg [83] studied the convex hull $QP^n$ of all feasible solutions of (1.1) to (1.5), which is called the Boolean Quadric Polytope.

Four families of facets of the polytope $QP^n$ were studied by Padberg [83], which include: trivial facets, clique inequalities, cut inequalities, and generalized cut inequalities. In general, these four families of facets are not all facets of $QP^n$. Sherali et al. [104] applied a simultaneous lifting procedure to obtain the new family of facet-defining inequalities for $QP^n$. Macambira and Souza [76] studied the edge-weighted clique problem and found that some facet-defining inequalities for the polytope corresponding to this problem also define facets for $QP^n$.

It has been proved in various papers that there is a linear bijective transformation from the cut polytope to $QP^n$. (See, e.g., [106].) The equivalence between these two polytopes yields many more valid inequalities and facet-defining inequalities for $QP^n$ from those that are known for the cut polytope. Some examples of facet-defining inequalities of the cut polytope are the family of odd bicycle wheel facets [16] and the family of $(4k + 1, 2)$-circulant inequalities [91]. Deza and Laurent [26] presented a survey on families of valid and facet-defining inequalities of the cut polytope. De Simone [107] showed that the result of Sherali and Adams [104] can be obtained from a family of valid inequalities for the cut polytope by using a linear transformation. Boros and Lari [21] gave a new class of inequalities obtained from hypermetric inequalities which is valid for the cut polytope. They stated a necessary
and sufficient condition that allows them to define facets of QP\(^n\). More families of facets obtained from the cut polytope were studied by Boros and Hammer [18].

Given a point and a class of inequalities, a *Separation Problem* is to determine whether the point satisfies all of those inequalities or not. If it does not satisfy, then the algorithm outputs an inequality violated by that point. Note that the class of inequalities could be exponential in number and defined implicitly. Separation algorithms are available for some families of valid and facets-defining inequalities for QP01. Letchford [68] showed that a class of inequalities including odd bicycle wheel inequalities and \((p, 2)\)-circulant inequalities for the cut polytope can be separated in polynomial time, but the algorithm is mostly impractical. Recently, Letchford and Sørensen [69] gave a separation algorithm for the \(\{0, 1/2\}\) cut inequalities of QP\(^n\) with complexity \(O(n^4)\). Heuristic separation algorithms are also known for hypermetric inequalities [56, 108]. De Simone and Rinaldi [108] gave a heuristic separation algorithm for hypermetric inequalities of the cut polytope while Helmberg and Rendl [56] gave heuristic separation algorithms for both hypermetric inequalities and odd clique inequalities of QP\(^n\). Again, polyhedral results known for QP\(^n\) were considerably influenced our work in this thesis.
Chapter 2

Complexity and Polynomially Solvable Cases

Recall that the bipartite boolean quadratic programming problem (BQP01) is maximizing $x^T Q y + cx + dy + c_0$ subject to $x \in \{0,1\}^m$ and $y \in \{0,1\}^n$. Due to its relationship with MWBCP mentioned in Chapter 1, BQP01 is an NP-hard problem. In this chapter, we will focus on some special cases that the problem is polynomially solvable.

Hereafter, we assume without loss of generality that $m \leq n$ and $c_0 = 0$. Since a BQP01 instance is defined only on the value of $Q, c$ and $d$, we use $P(Q,c,d)$ to represent the instance of BQP01 with objective function $x^T Q y + cx + dy$. For any solution $(x^0,y^0)$, $f(x^0,y^0)$ is the objective function value $(x^0)^T Q y^0 + cx^0 + dy^0$ at $(x^0,y^0)$. We denote $0_k$ and $1_k$ the vectors in $\mathbb{R}^k$ whose entries are all zero and one, respectively, while $0_h \times k$ is the zero matrix of dimension $h \times k$.

Besides, note that if $c_i, d_j, q_{ij} \leq 0$ for all pair $ij$, it is clear that $(x,y) = (0_m,0_n)$ is an optimal solution. On the other hand, if $c_i, d_j, q_{ij} \geq 0$ for all pair $ij$, we can easily see that $(x,y) = (1_m,1_n)$ maximizes our objective function. Thus, we can assume that $Q, c$ and $d$ have both positive and negative entries.

We begin this chapter with introduction of polynomial time algorithms for special cases when some parameters, says $m$ and the size of minimum negative eliminator or minimum
positive eliminator, are fixed. Then we discuss an algorithm that solves BQP01 with cost matrix $Q$ of rank $p$ in $O\left(\binom{m}{p}2^p mn\right)$ time. We show that when $Q$ is a matrix of rank one, the problem can be solved more efficiently in $O(n \log n)$ time. The last section in this chapter concentrates on some special structures of the objective function that allow us to identify an optimal solution for BQP01 in polynomial time. We show that when $Q$ is a matrix of rank one, the problem can be solved more efficiently in $O(n \log n)$ time. The cases when $Q$ is additively decomposable and $Q$ is $(2p + 1)$-diagonal are polynomial as well.

## 2.1 Problem with Fixed Parameter

In this section, we consider algorithms that solve BQP01 by fixing the value of some associated parameters.

We first observe that once we fix the value of the $x$ variables, we can find $y$ that optimizes the objective function with respect to that fixed $x$. The algorithm is described as follows.

**Algorithm 2.1.** For each $x \in \{0, 1\}^m$, choose $y(x)$ such that

$$y(x)_j = \begin{cases} 1 & \text{if } \sum_{i=1}^{m} q_{ij} x_i + d_j > 0, \\ 0 & \text{otherwise}. \end{cases}$$

Output an ordered pair $(x, y(x))$ with maximum objective function value.

A similar algorithm can be obtained by fixing all possible value of $y \in \{0, 1\}^n$. Because of our assumption that $m \leq n$, such an algorithm will be inferior.

**Theorem 2.1.** BQP01 is polynomially solvable if $m$ is $O(\log n)$. However, if $m = \Omega(\sqrt[2k]{n})$ for any constant $k \in \mathbb{Z}^+$, BQP01 is NP hard.

**Proof.** When we fix the value for variable $x$, we can find the optimal $y$ with respect to this fixed $x$ in $O(mn)$ time. Algorithm 2.1 terminates in $2^m$ iterations. If $m$ is $O(\log n)$, it means that this algorithm solves BQP01 in $O(mn^2)$ time. Let us now establish the complexity.

We consider an instance $P(Q, c, d)$ of BQP01. Define

$$\hat{Q} = \begin{bmatrix} Q & 0_{m \times nk} \\ 0_{(n-m) \times (n)} & 0_{(n-m) \times nk} \end{bmatrix}, \hat{c} = [c 0_{n-m}], \hat{d} = [d 0_n],$$

19
and consider the instance $P(\hat{Q}, \hat{c}, \hat{d})$ of BQP01. Note that $\hat{Q}$ is an $\hat{m} \times \hat{n}$ matrix where $\hat{m} = n$ and $\hat{n} = n^k + n$. Thus, $\hat{m} = O(\sqrt[3]{n})$ and it satisfies the condition of the theorem.

For each solution of $P(Q, c, d)$, we can find a solution to $P(\hat{Q}, \hat{c}, \hat{d})$ by assigning value 0 to entries $(m+1)^{th}$ to $n$ of $x$ and entries $(n+1)^{th}$ to $(n^k+n)^{th}$ of $y$. These two solutions give the same objective function values. On the other hand, for each solution of $P(\hat{Q}, \hat{c}, \hat{d})$, we can obtain a solution of $P(Q, c, d)$ with the same objective function value by omitting entries $(m+1)^{th}$ to $n^{th}$ of $x$ and entries $(n+1)^{th}$ to $(n^k+n)^{th}$ of $y$. Moreover, since the additional entries in $\hat{Q}, \hat{c}$ and $\hat{d}$ are zeros, these corresponding solutions have same objective function value for $P(Q, c, d)$ and $P(\hat{Q}, \hat{c}, \hat{d})$. The result now follows from the fact that BQP01 is NP hard.

If $m$ is fixed, then Algorithm 2.1 solves BQP01 in $O(n)$ time. If $m$ is $O(\log^k n)$, the algorithm solves BQP01 in quasi-polynomial time.

Let us now consider another fixed parameter, polynomially solvable special cases. Let $I \subseteq \{1, 2, \ldots, m\}$ and $J \subseteq \{1, 2, \ldots, n\}$. We say that $I \cup J$ is a negative eliminator of $(Q, c, d)$ if $(Q^*, c^*, d^*)$ obtained from $(Q, c, d)$ by deleting rows corresponding to $I$ and columns corresponding $J$ has only non-negative entries. Similarly, if $(Q^*, c^*, d^*)$ contains only negative entries or zeros then $I \cup J$ is called a positive eliminator of $(Q, c, d)$. A negative eliminator (positive eliminator) of smallest cardinality is called a minimum negative eliminator (minimum positive eliminator). Denote by $L^- = I^- \cup J^-$ and $L^+ = I^+ \cup J^+$, respectively a minimum negative eliminator and a minimum positive eliminator of $(Q, c, d)$. Recall that BQP01 with $(Q^*, c^*, d^*)$ containing only non-negative entries or $(Q^*, c^*, d^*)$ containing only negative entries or zeros can be solved trivially.

**Theorem 2.2.** The minimum negative eliminator of $(Q, c, d)$ can be identified in polynomial time.

**Proof.** Let $I'$ and $J'$ be a copy of $I$ and $J$, respectively. Construct the bipartite graph $G^- = (\hat{I}, \hat{J}, \hat{E})$ where $\hat{I} = I \cup J'$, $\hat{J} = J \cup I'$, and $\hat{E} = \{ij : i \in I, j \in J, q_{ij} < 0\} \cup \{ii' : i \in I, i' \in I', c_i < 0\} \cup \{jj' : j \in J, j' \in J', d_j < 0\}$. Then a minimum negative eliminator of $(Q, c, d)$ can be obtained from a minimum vertex cover of $G^-$. König’s theorem shows
the equivalence between finding maximum matching and minimum vertex covering on a
bipartite graph and hence the vertex cover problem on a bipartite graph can be solved in
polynomial time. Therefore, the minimum negative eliminator of $Q$ can be identified in
polynomial time.

The algorithm is described as follows.

Algorithm 2.2. Fix variables in minimum negative eliminator

Step 1: Construct $G^-$ and find a minimum negative eliminator $L^-$

Step 2: Consider each $\{0,1\}$ value assignment to $x_i$ where $i \in I^-$ and $y_j$ for $j \in J^-$. Assign value 1 to all other entries.

Step 3: Output $(x,y)$ with maximum objective function value.

Similarly, we can construct $G^+ = (\hat{I}, \hat{J}, \hat{E})$ where $\hat{I} = I \cup J^\prime$, $\hat{J} = J \cup I^\prime$, and $\hat{E} = \{ij : i \in I, j \in J, q_{ij} > 0\} \cup \{ii' : i \in I, i' \in I^\prime, c_i > 0\} \cup \{jj' : j \in J, j' \in J^\prime, d_j > 0\}$ and apply similar process to identify a minimum positive vertex eliminator. In Algorithm 2.2, if we change $L^-$ to $L^+$, $I^-$ to $I^+$, $J^-$ to $J^+$ and assign value 0 to all other entries instead of 1, we obtain an algorithm using minimum positive eliminator instead.

Theorem 2.3. BQP01 is polynomially solvable if $|L^-|$ or $|L^+|$ is $O(\log n)$. However, if neither $|L^-|$ nor $|L^+|$ is $O(\log n)$ and $|L^-|$ or $|L^+|$ is $\Omega(\sqrt[n]{n})$, BQP01 is still NP hard.

Proof. Recall that minimum negative eliminator and minimum positive eliminator can be identified in polynomial time by finding a minimum vertex cover in the modified bipartite graphs. We need to explore $2^{\frac{|L^-|}{4}}$ or $2^{\frac{|L^+|}{4}}$ different fixations in Step 2. Thus, BQP01 can be solved in polynomial time if $|L^-|$ or $|L^+|$ is $O(\log n)$. The second part of the theorem can be completed by using the same arguments as in the proof of Theorem 2.1.

Consequently, if the number of negative elements in $(Q,c,d)$ or the number of positive elements in $(Q,c,d)$ are of $O(\log n)$, then BQP01 can be solved in polynomial time as well.

2.2 Problem with Special Structures on Objective Functions

Let us now explore additional polynomially solvable cases with more efficient algo-
rithms but with the cost of additional restrictions on the objective functions.
2.2.1 BQP01 with Fixed Rank $Q$

Recall that QP01 can be solved in polynomial time if rank of $Q'$ is fixed, $Q'$ is positive semidefinite matrix, and $c' = 0$ [6, 28, 35, 58]. If $c' \neq 0$, the problem is NP-hard even when $Q'$ has rank one [51]. We now show that BQP01 with fixed rank cost matrix is polynomially solvable if rank of $Q$ is fixed. Unlike QP01, no other restrictions are required. $Q$ does not need to be positive semidefinite and $c$ and $d$ can be arbitrary.

Our algorithm is inspired by the work of Konno et. al [63]. The basic idea is decomposing the problem into two multiparametric linear programs. An optimal solution of the original problem can be obtained from basic feasible solutions of these linear programs.

Let us first establish some general results. Consider the multiparametric linear programming problem (MLP)

$$h_1(\lambda) = \text{Maximum } cx$$
$$\text{Subject to } Ax = \lambda$$
$$x \in [0,1]^m,$$

where $A$ is a $p \times m$ matrix of full row rank and $\lambda^T = (\lambda_1, \ldots, \lambda_p) \in \mathbb{R}^p$. Then $h_1(\lambda)$ is a piecewise linear concave function [30].

A partition $(B, L, U)$ of $I = \{1, 2, \ldots, m\}$ where $|B| = p$ is a basis structure of MLP if the columns of $A$ corresponding to $B$ are linearly independent. Each basis structure $(B, L, U)$ corresponds to a basic feasible solution of MLP such that $L$ is the index set of nonbasic variables at the lower bound 0, $U$ is the index set of nonbasic variables at the upper bound 1, and $B = \{B_1, \ldots, B_p\}$ is the index set of basic variables. Denote $A^i$ the $i^{th}$ column of matrix $A$. Let $B$ be the $p \times p$ matrix whose $i^{th}$ column is $A^{B_i}$. We call the set $B$ a basis and $B$ is its corresponding basis matrix. Let $C^B$ be the row vector $(c_{B_1}, \ldots, c_{B_p})$. A basis structure $(B, L, U)$ is dual feasible if and only if [81]

$$C^B B^{-1} A^i - c_i \geq 0 \text{ for } i \in L \tag{2.1}$$
$$C^B B^{-1} A^i - c_i \leq 0 \text{ for } i \in U. \tag{2.2}$$
Recall that $0_k$ is the vector of size $k$ whose entries are equal to 0 and $1_k$ is the vector of size $k$ whose entries are equal to 1. Denote $A^L$ the submatrix whose columns are $A^i$'s where $i \in \mathbb{L}$ and $A^U$ the submatrix whose columns are $A^i$'s where $i \in \mathbb{U}$. Note that we have

$$\lambda = Ax = Bx_B + A^L0_{|\mathbb{L}|} + A^U1_{|\mathbb{U}|} = Bx_B + A^U1_{|\mathbb{U}|},$$

where $x_B$ is the restriction of $x$ on $\mathbb{B}$. Since $B$ is a basis matrix, it is invertible and hence $B^{-1}$ exists. Let $v = 1_{|\mathbb{U}|}$, the all one vector of size $|\mathbb{U}|$. From (2.3), we can write $x_B$ as $B^{-1}\lambda - B^{-1}A^Uv$. It follows that for each dual feasible basic structure $(\mathbb{B}, \mathbb{L}, \mathbb{U})$, the corresponding basic feasible solution is optimal for any $\lambda$ satisfying

$$0_p \leq B^{-1}\lambda - B^{-1}A^Uv \leq 1_p.$$  

(2.4)

We denote the polyhedral set formed by (2.4) the characteristic region of $(\mathbb{B}, \mathbb{L}, \mathbb{U})$. Moreover, a dual feasible basis structure $(\mathbb{B}, \mathbb{L}, \mathbb{U})$ is dual non-degenerate if (2.1) and (2.2) are satisfied as strict inequalities. If any of the inequalities in (2.1) or (2.2) holds as equality, $(\mathbb{B}, \mathbb{L}, \mathbb{U})$ is a dual degenerate basis structure. Without loss of generality, we assume that the basis structure $(\mathbb{B}, \mathbb{L}, \mathbb{U})$ of MLP are dual non-degenerate. If not, this can be achieved by suitable perturbation of the cost vector $c$.

Let $S$ be the set of all extreme points of the characteristic regions associated with all dual feasible basis structures of MLP. The next lemma gives an upper bound on the size of $S$.

**Lemma 2.4.** $|S| \leq \binom{m}{p}2^p$

**Proof.** We can see that for each choice of the basis $\mathbb{B}$, the value $C^B B^{-1}A^i - c_i$ is fixed. Since we assume that all dual feasible basis structures are non-degenerate, $C^B B^{-1}A^i - c_i \neq 0$ for every $i \in \mathbb{L} \cup \mathbb{U}$. Therefore, $i$ is in $\mathbb{L}$ if $C^B B^{-1}A^i - c_i > 0$ and in $\mathbb{U}$ otherwise. Hence, for any dual feasible basis structure $(\mathbb{B}, \mathbb{L}, \mathbb{U})$, given $\mathbb{B}$, $\mathbb{L}$ and $\mathbb{U}$ are uniquely determined. Since there are at most $\binom{m}{p}$ different ways to choose $\mathbb{B}$, there exists at most $\binom{m}{p}$ dual feasible basis structures that are dual non-degenerate.

There are $p$ pairs of inequalities in the form (2.4). Moreover, there is only one inequality in each pair that can be satisfied at equality. Hence, each characteristic region has at most $2^p$ extreme points. Consequently, the size of $S$ is bounded by $\binom{m}{p}2^p$. \qed
Let \( S(\mathcal{B}, \mathcal{L}, \mathcal{U}) \) be the set of all extreme points of the characteristic region associated with \((\mathcal{B}, \mathcal{L}, \mathcal{U})\) and \( \tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U}) \) be the collection of all basic feasible solutions of MLP associated with the extreme points in \( S(\mathcal{B}, \mathcal{L}, \mathcal{U}) \). We show that \( \tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U}) \) can be obtained easily without computing \( S(\mathcal{B}, \mathcal{L}, \mathcal{U}) \).

**Lemma 2.5.** \( \tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U}) \subseteq \{0, 1\}^m \) and \( |\tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U})| = 2^p \).

**Proof.** Let \( x(\lambda) \in \tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U}) \) be a basic feasible solution for MLP corresponding to the extreme point \( \lambda \in S(\mathcal{B}, \mathcal{L}, \mathcal{U}) \) and \( B \) be the basis matrix associated with \((\mathcal{B}, \mathcal{L}, \mathcal{U})\). Let \( x(\lambda)_B \) be the vector of basic variables of \( x(\lambda) \). Note that

\[
S(\mathcal{B}, \mathcal{L}, \mathcal{U}) = \{\lambda_w : w = B^{-1}\lambda_w - B^{-1}A^Uv, w \in \{0, 1\}^p\}.
\]

Hence, \( \lambda = Bw + A^Uv \) for some \( w \in \{0, 1\}^p \). Therefore,

\[
x(\lambda)_B = B^{-1}\lambda - B^{-1}A^Uv = B^{-1}(Bw + A^Uv) - B^{-1}A^Uv = w.
\]

The non-basic variables of \( x(\lambda) \) by definition take 0-1 values and hence \( x(\lambda) \in \{0, 1\}^m \). Thus, \( \tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U}) \subseteq \{0, 1\}^m \).

Since there are \( 2^p \) choices for \( w \) and non-basic variables in \( \mathcal{L} \) and \( \mathcal{U} \) are uniquely fixed for a given \((\mathcal{B}, \mathcal{L}, \mathcal{U}) \) (independent of the choice of \( w \)), we have \( |\tilde{S}(\mathcal{B}, \mathcal{L}, \mathcal{U})| = 2^p \). \qed 

Now let us consider BQP01 when \( Q \) has rank \( p \). We assume that \( Q \) is given in the rank-\( p \) factorized form \( Q = A'B' \) where \( A' \) is an \( m \times p \) matrix and \( B' \) is a \( p \times n \) matrix. Such a factorization can be constructed easily from the reduced row echelon form \( Q^R \) of \( Q \) by choosing \( A' \) as the matrix obtained from \( Q \) by deleting non-pivot columns and \( B' \) as the matrix obtained from \( Q^R \) by removing the zero rows. For \( k = 1, \ldots, p \), let \( a^k = (a^k_1, \ldots, a^k_m) \) be the transpose of the \( k^{th} \) column of \( A' \) and \( b^k = (b^k_1, \ldots, b^k_n) \) be the \( k^{th} \) row of \( B' \). Since \( Q = A'B' \), we can rewrite \( x^TQy \) as

\[
x^T A' B' y = \begin{bmatrix} x^T(a^1) & \ldots & x^T(a^p) \end{bmatrix} \begin{bmatrix} b^1 y \\ \vdots \\ b^p y \end{bmatrix} = \sum_{k=1}^{p} (a^k x)^T b^k y = \sum_{k=1}^{p} a^k x b^k y
\]
Because of the equivalence between BQP01 and UBLP (See Section 1.3), BQP01 with cost matrix of rank \( p \) can be stated as

\[
\text{UBLP}(p): \quad \text{Maximize } \sum_{k=1}^{p} a^k x b^k y + cx + dy
\]

Subject to \( x \in [0, 1]^m, y \in [0, 1]^n \).

Consider the MLP

\[
\text{MLP1: } h_1(\lambda) = \text{Maximum } cx
\]

Subject to \( a^k x = \lambda_k \) for \( k = 1, \ldots, p \)

\[
x \in [0, 1]^m,
\]

where \( \lambda = (\lambda_1, \ldots, \lambda_p)^T \). Note that \( h_1(\lambda) \) is a piecewise linear concave function. It is well known that \( h_1(\lambda) \) is linear when \( \lambda \) is restricted to a characteristic region with respect to some basis structure.

Let \( \lambda = (\lambda_1, \ldots, \lambda_p)^T \) be a vector in \( \mathbb{R}^p \). The problem UBLP(\( p \)) can be represented as

\[
\text{BLP1: } \quad \text{Maximize } \sum_{k=1}^{p} \lambda_k b^k y + cx + dy
\]

Subject to \( a^k x = \lambda_k \) for \( k = 1, \ldots, p \)

\[
x \in [0, 1]^m, y \in [0, 1]^n.
\]

Define \( h_2(\lambda) \) as

\[
h_2(\lambda) = \text{Maximum } \sum_{k=1}^{p} \lambda_k b^k y + dy
\]

Subject to \( y \in [0, 1]^n \).

Note that we can rewrite \( h_2(\lambda) = \sum_{k=1}^{p} \lambda_k b^k y + dy \) as \( (d + \lambda^T B') y \) which is known to be a piecewise linear convex function of \( \lambda \) [30]. Let \( h(\lambda) \) be the optimal objective function value of BLP1 where \( \lambda \) is fixed. Then we can write \( h(\lambda) \) as the sum of \( h_1(\lambda) \) and \( h_2(\lambda) \).

For each fixed \( \lambda \in S \), let \( x(\lambda) \) be an optimal basic feasible solution of the MLP with respect to \( \lambda \). We can consider the problem UBLP(\( p \)) when \( x \) is fixed to be \( x(\lambda) \). Let
$y(\lambda) = (y_1(\lambda), \ldots, y_n(\lambda))$ be an optimal solution to UBLP$(p)$ when $x$ is restricted to $x(\lambda)$. Given $x(\lambda)$, we can obtain $y(\lambda)$ efficiently as given in the following lemma.

**Lemma 2.6.** For each fixed $x(\lambda)$, an optimal solution $y(\lambda)$ to UBLP$(p)$ when $x$ is restricted to $x(\lambda)$ can be identified within $O(mnp)$ time.

**Proof.** It can be verified that the objective function $\sum_{k=1}^{p} \lambda_k b_k y + dy$ can be rewritten as $\sum_{j=1}^{n} (\sum_{k=1}^{p} \lambda_k b^k_j + d_j) y_j$. Once $x(\lambda)$ is given, $\lambda_k = \sum_{i=1}^{m} a_k^i x_i(\lambda)$. Thus, $y(\lambda)$ satisfies

$$y_j(\lambda) = \begin{cases} 
1 & \text{if } d_j + \sum_{k=1}^{p} b^k_j \sum_{i=1}^{m} a_k^i x_i(\lambda) > 0, \\
0 & \text{otherwise}. 
\end{cases}$$

For $j = 1, \ldots, n$, since $d_j + \sum_{k=1}^{p} b^k_j \sum_{i=1}^{m} a_k^i x_i(\lambda)$ can be obtained in $O(mp)$ time, $y(\lambda)$ can be found within $O(mnp)$ time. □

We can show that if we explore all $\lambda \in S$, we can find an optimal solution to UBLP$(p)$ in the set $\{(x(\lambda), y(\lambda)) : \lambda \in S\}$.

**Lemma 2.7.** There exists an optimal solution of UBLP$(p)$ in the set

$$S' = \{(x(\lambda), y(\lambda)) : \lambda \in S\}.$$ 

**Proof.** Since $h(\lambda)$ can be decomposed into $h_1(\lambda)$ and $h_2(\lambda)$, the optimal objective function value of UBLP$(p)$ can be obtained by finding the optimal value of $h(\lambda)$ over all $\lambda \in \mathbb{R}^p$ for which BLP1 is feasible. Recall that $h_1(\lambda)$ is linear when $\lambda$ is restricted to a characteristic region with respect to some basis structure and $h_2(\lambda)$ is piecewise linear convex. It follows that $h(\lambda) = h_1(\lambda) + h_2(\lambda)$ is also convex when $\lambda$ is restricted to a characteristic region associated with $h_1(\lambda)$. Therefore, the maximum value of $h(\lambda)$ is obtained at an extreme point $\lambda \in S$ of $h_1(\lambda)$. We call that extreme point $\lambda^*$. From Lemma 2.5 and Lemma 2.6, $(x(\lambda^*), y(\lambda^*))$ is binary and hence $(x(\lambda^*), y(\lambda^*))$ is an optimal solution for BQP01 instance. □

Note that in order to obtain $S'$, we do not need to compute set $S$. The solution set $\tilde{S} = \{x(\lambda) : \lambda \in S\}$ can be identified without computing any $\lambda \in S$. Let $(\mathbb{B}, \mathbb{L}, \mathbb{U})$ be a
dual feasible and dual non-degenerate basis structure. Also let $\mathcal{B} = \{B_1, \ldots, B_p\}$ and $B$ be the associated basis matrix. From the proof of Lemma 2.5, for each $w \in \{0, 1\}^p$, we get the corresponding extreme point $\lambda = Bw + A^Uv$ of its characteristic region and the corresponding basic variables $x(\lambda)_B$ is precisely $w$. The non-basic variables are determined by $L$ and $U$. Thus, for each basis structure $(\mathcal{B}, L, U)$ we can generate $2^p$ basic feasible solutions corresponding to its extreme points by varying $w \in \{0, 1\}^p$ (see proof of Lemma 2.5).

Our algorithm for solving BQP01 with rank-$p$ cost matrix $Q$ is summarized below.

**Algorithm 2.3. Rank-$p$ cost matrix**

**Step 1:** Let $A$ be the coefficient matrix of MLP1 whose $k^{th}$ row is $a^k$. Construct the collection $\Gamma$ of all dual feasible basis structures $(\mathcal{B}, L, U)$ associated with $A$ by considering every choice of basis $\mathcal{B}$. Due to dual non-degeneracy assumption, $L$ and $U$ can be uniquely determined by inequalities (2.1) and (2.2).

**Step 2:** For each basis structure $(\mathcal{B}, L, U) \in \Gamma$, construct the set of optimal basic feasible solutions $\bar{S}(\mathcal{B}, L, U)$ corresponding to the extreme points of its characteristic region by varying all $x(\lambda)_B \in \{0, 1\}^p$. Let $\bar{S}$ be the union of $\bar{S}(\mathcal{B}, L, U)$ over all basis structure $(\mathcal{B}, L, U) \in \Gamma$.

**Step 3:** For each $x(\lambda) \in \bar{S}$, compute $y(\lambda) \in \mathbb{R}^n$ from the formula in Lemma 2.6. Let $S' = \{(x(\lambda), y(\lambda)) : x(\lambda) \in \bar{S}\} = \{(x(\lambda), y(\lambda)) : \lambda \in S\}$.

**Step 4:** Output the best solution in $S'$.

**Theorem 2.8.** Algorithm 2.3 solves BQP01 in $O(\binom{m}{p}2^p mn)$ time when $\text{rank}(Q) = p$ and $Q$ is given in the factored form.

**Proof.** There are at most $\binom{m}{p}$ choices to choose basis $\mathcal{B}$. Recall that $C^B$ is the restriction of cost vector $c$ on the basis $\mathcal{B}$ and $A^i$ is the $i^{th}$ column of matrix $A$. For each basis $\mathcal{B}$, we need to compute $C^B B^{-1} A^i - c_i$ in order to use inequalities (2.1) and (2.2) to identify $L$ and $U$. The inverse of the associated basis matrix $B$ can be obtained in $O(p^3)$ time. Given $B^{-1}$, we can compute $C^B B^{-1}$ in $O(p^2)$ time. We can compute $C^B B^{-1} A^i - c_i$ for all $i = 1, \ldots, m$ in $O(mp)$ time when the $1 \times p$ matrix $C^B B^{-1}$ is given. Hence, for each basis $\mathcal{B}$, we can identify $L$ and $U$ in $O(p^3 + mp)$ time. Thus, $\Gamma$ in **Step 1** can be constructed in $O(\binom{m}{p} (p^3 + mp))$ time.
The characteristic region associated with each \((B, L, U) \in \Gamma\) has \(2^p\) extreme points and the optimal solution \(x(\lambda)\) to MLP1 when \(\lambda\) is fixed at these extreme points can be identified without explicitly computing \(\lambda\). Note that \(|\Gamma| \leq \binom{m}{p}\). Thus, given \(\Gamma\), \(\bar{S}\) in Step 2 can be identified in \(O\left(\binom{m}{p}2^p m^2\right)\) time.

Recall that from Lemma 2.5, \(|\bar{S}(B, L, U)| = 2^p\). Since \(|\Gamma| \leq \binom{m}{p}\), we have \(|\bar{S}| \leq \binom{m}{p}2^p\). From Lemma 2.6, for each \(x(\lambda) \in \bar{S}\), \(y(\lambda)\) can be obtained in \(O(mnp)\) time. Hence, \(S'\) in Step 3 can be identified in \(O\left(\binom{m}{p}2^p mn^2\right)\) time.

Lemma 2.7 ensures that it is enough to consider only \((x(\lambda), y(\lambda))\) where \(\lambda \in S\). We can see that the overall complexity is dominated by the complexity of the last step which is \(O\left(\binom{m}{p}2^p mn\right)\).

\[\text{2.2.2 BQP01 with Rank One Matrix}\]

Theorem 2.8 guarantees that if rank of \(Q\) is 1, BQP01 can be solved in \(O(m^2 n)\) time. We now show that the problem can be solved in \(O(n \log n)\) time by careful organization of our computations. We roughly state the idea of the algorithm and leave all details in Appendix A.

As in the general case, let us consider the bilinear equivalent version:

\[\text{UBLP}(1): \quad \text{Maximize } axby + cx + dy\]

Subject to: \(x \in [0, 1]^m, y \in [0, 1]^n\),

where \(a = (a_1, \ldots, a_m), c = (c_1, \ldots, c_m) \in \mathbb{R}^m\) and \(b = (b_1, \ldots, b_n), d = (d_1, d_2, \ldots, d_n) \in \mathbb{R}^n\). Let \(A^- = \{i : a_i < 0\}\) and \(A^+ = \{i : a_i > 0\}\). Define \(\lambda = \sum_{i \in A^-} a_i\) and \(\bar{\lambda} = \sum_{i \in A^+} a_i\), where summation over the empty set is taken as zero. Note that \(\lambda\) and \(\bar{\lambda}\) are respectively the smallest and the largest values of \(ax\) when \(x \in [0, 1]^m\). Consider the Parametric Continuous
Knapsack Problem (PKP(\(\lambda\))) given below.

\[
\text{PKP}(\lambda): \quad \text{Maximize } cx \\
\text{Subject to } ax = \lambda \\
x \in [0, 1]^m, \text{ and } \underline{\lambda} \leq \lambda \leq \overline{\lambda}.
\]

This is a special case of MLP1 for \(p = 1\). Let \(h_1(\lambda)\) be the optimal objective function value of PKP(\(\lambda\)) for a given \(\lambda\). Then for \(\underline{\lambda} \leq \lambda \leq \overline{\lambda}\), \(h_1(\lambda)\) is a piecewise linear concave function [30]. Let \(\underline{\lambda} = \lambda_0 < \lambda_1 < \ldots < \lambda_p = \overline{\lambda}\) be the breakpoints of \(h_1(\lambda)\) and \(x_k\) be an optimal basic feasible solution of PKP(\(\lambda\)) for \(\lambda \in [\lambda_{k-1}, \lambda_k], 1 \leq k \leq p\). Then \(x_k\) will be an optimal basic feasible solution to PKP(\(\lambda_k\)). Let \(y^k\) be an optimal solution to UBLP(1) when \(x\) is restricted to \(x^k\). From Lemma 2.6, the vector \(y^k\) can be identified by appropriate modification of the equation in Lemma 2.6. By Theorem 2.7, there exists an optimal solution to UBLP(1) amongst the solutions \((x^k, y^k)\) where \(k = 0, 1, \ldots, p\).

From Lemma 2.4, the number of breakpoints of \(h_1(\lambda)\) is at most \(2m\). We now observe that the number of breakpoints of \(h_1(\lambda)\) cannot be more than \(m + 1\) and obtain closed form values of these breakpoints.

Let \(T = \left\{ \frac{c_i}{a_i} : i = 1, \ldots, m, \ a_i \neq 0 \right\} \) and consider a descending arrangement

\[
\frac{c_{\pi(1)}}{a_{\pi(1)}} > \ldots > \frac{c_{\pi(p)}}{a_{\pi(p)}}, \quad (2.5)
\]

of all distinct elements of \(T\). Let \(T(k) = \left\{ i : \frac{c_{\pi(k)}}{a_{\pi(k)}} = \frac{c_i}{a_i} \right\}\). Then the breakpoints of \(h_1(\lambda)\) are given by

\[
\lambda_0 = \underline{\lambda} \text{ and } \lambda_k = \lambda_{k-1} + \sum_{i \in T(k)} |a_i| \text{ for } k = 1, \ldots, p.
\]

An optimal solution to PKP(\(\lambda\)) at \(\lambda = \lambda_k\) for \(k = 0, 1, \ldots, p\) can be identified recursively as

\[
x_i^0 = \begin{cases} 1 & \text{if } a_i = 0 \text{ and } c_i > 0 \text{ or } a_i < 0, \\ 0 & \text{otherwise,} \end{cases}
\]
and

\[ x_i^k = \begin{cases} 
  x_{i-1}^k & \text{if } i \notin T(k), \\
  1 & \text{if } i \in T(k) \text{ and } a_i > 0, \\
  0 & \text{otherwise}.
\end{cases} \]

Thus, it can be verified that given \( h(\lambda_{k-1}) \) and \( x^{k-1} \), \( h(\lambda_k) \) and \( x^k \) can be identified in \( O(|T(k)|) \) time. The complexity for generating these solutions and breakpoints are dominated by that of constructing the descending arrangement (2.5) which is \( O(m \log m) \). Note that \( h_1(\lambda) \) has at most \( m + 1 \) breakpoints and given \( x^k \), a corresponding solution \( y^k \) can be computed in \( O(n) \) time. This leads to a complexity of \( O(mn) \). The bottleneck operation here is the computation of \( y^k \) for \( k = 1, \ldots, p \). We now show that these points can be identified in \( O(n \log n) \) time.

Consider the Parametric Unconstrained Linear Optimization Problem

\[ \text{ULP}(\mu): \quad \text{Maximize } dy + \mu by \]

Subject to: \( y \in [0, 1]^n \) and \( \underline{\lambda} \leq \mu \leq \overline{\lambda} \).

Let \( h_2(\mu) \) be the optimal objective function value of \( \text{ULP}(\mu) \). Then \( h_2(\mu) \) is a piecewise linear convex function.

Let \( B^+ = \{ j : b_j > 0 \} \) and \( B^- = \{ j : b_j < 0 \} \). Also, let \( S = \{-\frac{d_j}{b_j} : j \in B^+ \cup B^- \} \) and \( \underline{\lambda} \). We consider the ascending order

\[ -\frac{d_{\sigma(1)}}{b_{\sigma(1)}} < \cdots < -\frac{d_{\sigma(s)}}{b_{\sigma(s)}} \]

of all different element in \( S \). Let \( \underline{\lambda} = \mu_0 < \mu_1 < \ldots < \mu_s = \overline{\lambda} \) be the breakpoints of \( h_2(\lambda) \). Then \( \mu_l = -\frac{d_{\sigma(l)}}{b_{\sigma(l)}} \) for \( l = 1, \ldots, s \). Let \( S_0^+ = \{ j : d_j + b_j \lambda = 0 \} \) and \( S(l) = \{ j : \frac{d_j}{b_j} = -\frac{d_{\sigma(l)}}{b_{\sigma(l)}} \} \). Then the optimal solution \( y^l \) corresponding to the breakpoint \( \mu_l \) for \( l = 1, \ldots, s \) is given recursively by

\[ y_j^l = \begin{cases} 
  y_j^{l-1} & \text{if } j \notin S(l), \\
  1 & \text{if } j \in S(l) \text{ and } y_j^{l-1} = 0, \\
  0 & \text{if } j \in S(l) \text{ and } y_j^{l-1} = 1,
\end{cases} \]

where \( y_j^0 = \begin{cases} 
  1 & \text{if } j \in S_0^+, \\
  0 & \text{otherwise}.
\end{cases} \]
Define
\[
D^0 = \sum_{j \in S_0^+} d_j, \quad D^l = D^{l-1} - \sum_{j \in S(l), y_j^{l-1} = 1} d_j + \sum_{j \in S(l), y_j^{l-1} = 0} d_j \quad \text{and} \\
B^0 = \sum_{j \in S_0^+} b_j, \quad B^l = B^{l-1} - \sum_{j \in S(l), y_j^{l-1} = 1} b_j + \sum_{j \in S(l), y_j^{l-1} = 0} b_j.
\]

Then the optimal objective function value at \( \mu_l \) is given by \( h_2(\mu_l) = D^l + \mu_l B^l \).

Given \( y^{l-1}, D^{l-1} \) and \( B^{l-1} \), we can compute \( y^l, D^l, \) and \( B^l \) in \( O(|S(l)|) \) time and, hence, \( h_2(\mu_l) \) and \( y^l \) can be identified in \( O(|S(l)|) \) time. Since \( S(l) \cap S(k) = \emptyset \) for \( l \neq k \), \( y^l \) and \( h_2(\mu^l) \) for \( l = 1, \ldots, s \) can be identified in \( O(n) \) time.

Now the algorithm for solving UBLP(1) can be described as follows. First, compute \( x^0, y^0, h_1(\lambda) \) and \( h_2(\lambda) \). Set \( f(x^0, y^0) = h_1(\lambda) + h_2(\lambda) \). Sort all breakpoints of \( h_1(\lambda) \) and \( h_2(\mu) \) for \( \lambda \leq \mu \leq \bar{\lambda} \) (see Fig: 2.1) and scan these breakpoints starting from \( \lambda \) in the increasing order. As we pass breakpoints of \( h_2(\mu) \) keep updating the solution \( y \) of ULP(\( \mu \)) corresponding to this breakpoint and the objective function value of this solution until we hit a breakpoint \( \lambda_k \) of \( h_1(\lambda) \). At this point compute the solution \( x^k \) and \( h_1(\lambda_k) \). The most recent solution \( y \) identified is selected as \( y^k \) and compute \( h_2(\lambda_k) \). Note that \( h_2(\lambda_k) \) can be obtained in \( O(1) \) time using slope of \( h_2(\mu) \) for the interval containing \( \lambda^k \). Update \( f(x^k, y^k) \) and the process is continued until all breakpoints of \( h_1(\lambda) \) including \( \bar{\lambda} \) are examined and...
the overall best solution is selected. It is not difficult to verify that the complexity of this procedure is $O(n \log n)$.

2.2.3 BQP01 with the Objective Function in the Form $(a_0 + ax)(b_0 + by) + cx + dy$ where $c = 0^T_m$ or $d = 0^T_n$

In this section, we consider a special case of BQP01 when $Q$ has rank one and the objective function can be written as $(a_0 + ax)(b_0 + by) + cx + dy$ where $c = 0^T_m$ or $d = 0^T_n$. The problem is formulated as

$$\text{BQP01}(1, c \lor d = 0): \text{Maximize } (a_0 + ax)(b_0 + by) + cx + dy$$

Subject to $x \in \{0, 1\}^m, y \in \{0, 1\}^n,$

where $c = 0^T_m$ or $d = 0^T_n$. We will show that this problem can be solved efficiently.

**Theorem 2.9.** An optimal solution to BQP01$(1, c \lor d = 0)$ can be obtained in $O(n)$ time.

**Proof.** Note that BQP01$(1, c \lor d = 0)$ is equivalent to

$$\text{UBLP}(1, c \lor d = 0): \text{Maximize } (a_0 + ax)(b_0 + by) + cx + dy$$

Subject to $x \in [0, 1]^m, y \in [0, 1]^n,$

where $c = 0^T_m$ or $d = 0^T_n$. Consider the case when $d = 0^T_n$. We consider solutions

$$y_j^0 = \begin{cases} 1 & \text{if } b_j > 0, \\ 0 & \text{otherwise} \end{cases}$$

and

$$y_j^* = \begin{cases} 1 & \text{if } b_j < 0, \\ 0 & \text{otherwise}. \end{cases}$$

We can see that $y^0 = (y_j^0, \ldots, y_n^0)$ maximizes $b_0 + by$ and $y^* = (y_j^*, \ldots, y_n^*)$ minimizes $b_0 + by$. Denote $\lambda^0 := b_0 + by^0$ and $\lambda^* := b_0 + by^*$. Then UBLP$(1, c \lor d = 0)$ where $d = 0^T_n$ is equivalent to

$$L(\lambda) = \text{Maximize } (a_0 + ax)\lambda + cx$$

Subject to $x \in [0, 1]^m, \lambda \in [\lambda^*, \lambda^0]$. 32
Note that \((a_0 + ax)\lambda + cx = a_0\lambda + (\lambda a + c)x\). Since \(a_0\lambda\) is a linear function of \(\lambda\) and from [30], \((\lambda a + c)x\) is a piecewise linear convex function of \(\lambda\), \(L(\lambda)\) is also a piecewise linear convex function of \(\lambda\). Convexity of \(L(\lambda)\) guarantees that its maximum is attained at \(\lambda = \lambda^0\) when \(y = y^0\) or \(\lambda^*\) when \(y = y^*\). Moreover, when \(y = y^0\), an optimal \(x^0\) is

\[
x^0_i = \begin{cases} 1 & \text{if } a_i\lambda^0 + c_i > 0, \\ 0 & \text{otherwise}, \end{cases}
\]

and when \(y = y^*\), an optimal \(x^*\) is

\[
x^*_i = \begin{cases} 1 & \text{if } a_i\lambda^* + c_i > 0, \\ 0 & \text{otherwise}. \end{cases}
\]

Therefore, the best solution among \((x^0, y^0)\) and \((x^*, y^*)\) is an optimal solution of \(\text{BQP01}(1, c\lor d = 0)\). It is easy to see that the complexity of the algorithm is \(O(n)\). The algorithm for the case that \(c = 0\) can be obtained in a similar way.

2.2.4 BQP01 with \(Q\) as an Additively Decomposable Matrix

Let us now examine the case when \(Q\) is an additively decomposable matrix, which means \(q_{ij} = a_i + b_j\) for \(i = 1, \ldots, m\) and \(j = 1, \ldots, n\). We observe that when \(q_{ij} = a_i + b_j\), rank of \(Q\) is at most 2 and hence can be solved in polynomial time. We now give an \(O(mn \log n)\) algorithm for solving this problem.

We first notice that for any feasible solution \((x, y) \in \{0, 1\}^{m+n}\),

\[
f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j + c_0
\]

\[
= \sum_{i=1}^{m} \sum_{j=1}^{n} (a_i x_i + b_j y_j) + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j + c_0
\]

\[
= \sum_{i=1}^{m} a_i x_i \sum_{j=1}^{n} y_j + \sum_{j=1}^{n} b_j y_j \sum_{i=1}^{m} x_i + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j + c_0.
\]
Let $\sum_{j=1}^{n} y_j = K$ and $\sum_{i=1}^{m} x_i = L$, where $K$ and $L$ are two parameters depending on the value of $y$ and $x$, respectively. Then

$$f(x, y) = K \sum_{i=1}^{m} a_i x_i + L \sum_{j=1}^{n} b_j y_j + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j + c_0$$

$$= \sum_{i=1}^{m} (Ka_i + c_i) x_i + \sum_{j=1}^{n} (Lb_j + d_j) y_j + c_0.$$ 

We now consider the optimization problem

$$\text{ILP}(K,L): \quad \text{Maximize} \quad \sum_{i=1}^{m} (Ka_i + c_i) x_i + \sum_{j=1}^{n} (Lb_j + d_j) y_j + c_0$$

$$\text{Subject to} \quad \sum_{j=1}^{n} y_j = K, \quad \sum_{i=1}^{m} x_i = L.$$

Observe that for fixed values of $K$ and $L$, we must choose exactly $L$ entries of $x$ and $K$ entries of $y$ to be one. Thus, to maximize the objective function, we assign value one to the first $L$ entries of $x$ with highest $Ka_i + c_i$ value and the first $K$ entries of $y$ with highest $Lb_j + d_j$ value. For $K = 0, 1, \ldots, n$, let $\alpha^K$ be a permutation of size $m$ such that $Ka_{\alpha^K(i)} + c_{\alpha^K(i)} \geq Ka_{\alpha^K(i+1)} + c_{\alpha^K(i+1)}$, $i = 1, \ldots, m - 1$. For $L = 0, 1, \ldots, m$, let $\beta^L$ be a permutation of size $n$ such that $Lb_{\beta^L(j)} + d_{\beta^L(j)} \geq Lb_{\beta^L(j+1)} + d_{\beta^L(j+1)}$, $j = 1, \ldots, n - 1$. Thus, the optimal $x = x^{K,L}$ can be obtained by setting $x_i = 1$ for $i = \alpha^K(1), \ldots, \alpha^K(L)$ and $x_i = 0$ otherwise. Similarly, the optimal $y = y^{K,L}$ can be obtained by setting $y_j = 1$ for $j = \beta^L(1), \ldots, \beta^L(K)$ and $y_j = 0$ otherwise.

Let $f_1^{K,L} = \sum_{i=1}^{m} (Ka_i + c_i) x_i^{K,L}$ and $f_2^{K,L} = \sum_{j=1}^{n} (Lb_j + d_j) y_j^{K,L}$. When the value of $f_1^{K,L}$ for some $K$ and $L$ is given, we can calculate $f_1^{K,L+1}$ from

$$f_1^{K,L+1} = f_1^{K,L} + Ka_{\alpha^K(L+1)} + c_{\alpha^K(L+1)}.$$ 

Note that $f_1^{K,0} = 0$. Hence, for a fixed $K$, we can calculate $f_1^{K,L}$ for each $L \in \{0, 1, \ldots, m\}$ recursively. Similarly, for a fixed $L$, $f_2^{K,L}$ for each $K = 0, 1, \ldots, n$ can be obtained recursively.

Let $(K^0, L^0)$ be the values of $K$ and $L$ that maximize $f_1^{K,L} + f_2^{K,L}$. Then the optimal solution of the BQP01 is $(x^{K^0,L^0}, y^{K^0,L^0})$. This process can be summarized in the following algorithm.

34
Algorithm 2.4. Algorithm for BQP01 with an additive decomposable matrix

Step 1: Compute all \((x^{K,L}, y^{K,L})\). Set \(K = 0\) and \(L = 0\).

Step 1.1: Compute \(K a_i + c_i\) for \(i = 1\) to \(m\). Then place all \(K a_i + c_i\) in the descending order to identify the permutation \(\alpha^K\). Set \(x^{K,0} = 0_m\). Then we obtain \(x^{K,L+1}\) for \(L = 0, \ldots, m - 1\) from \(x^{K,L}\) by changing the \(\alpha^K(L+1)\)th entry from 0 to 1. Set \(K := K + 1\). Repeat this step until \(K = n + 1\).

Step 1.2: Compute \(L b_j + d_j\) for \(j = 1\) to \(n\). Then place all \(L b_j + d_j\) in the descending order to identify the permutation \(\beta^L\). Set \(y^{0,L} = 0_n\). Then we obtain \(y^{K+1,L}\) for \(K = 0, \ldots, n - 1\) from \(y^{K,L}\) by changing the \(\beta^L(K+1)\)th entry from 0 to 1. Set \(L := L + 1\). Repeat this step. until \(L = m + 1\)

Step 2: Compute all \(f_1^{K,L}\) and \(f_2^{K,L}\). Set \(L = 0\) and \(f_1^{K,L} = 0\). For each \(K \in \{0,1,\ldots,n\}\), do the following substep.

Step 2.1: Compute \(f_1^{K,L+1} := f_1^{K,L} + K a_{\alpha^K(L+1)} + c_{\alpha^K(L+1)}\). Output \(f_1^{K,L+1}\) and set \(L := L + 1\). Repeat this step until \(L = m\).

Set \(K = 0\) and \(f_2^{K,L} = 0\). For each \(L \in \{0,1,\ldots,m\}\), do the following substep.

Step 2.2: Compute \(f_2^{K+1,L} := f_2^{K,L} + L b_{\beta^L(K+1)} + d_{\beta^L(K+1)}\). Output \(f_2^{K+1,L}\) and set \(K := K + 1\). Repeat this step until \(K = n\).

Step 3: Find an optimal solution. Compute all \(f_1^{K,L} + f_2^{K,L}\) and choose the maximum one. Let \((K^0, L^0)\) be the values that maximize \(f_1^{K,L} + f_2^{K,L}\). Output \((x^{K^0,L^0}, y^{K^0,L^0})\).

Here we show the complexity of Algorithm 2.4.

Theorem 2.10. If \(q_{ij} = a_i + b_j\) for \(i = 1, \ldots, m\) and \(j = 1, \ldots, n\), then BQP01 can be solved in \(O(mn \log n)\).

Proof. In Step 1.1, it takes \(O(m)\) time to compute \(K a_i + c_i\) for \(i = 1\) to \(m\). Order these \(m\) values can be done in \(O(m \log m)\) time. All \(x_i^{K,L}\) can be identified in \(O(m)\) time. So the computation time for each iteration of Step 1.1 is \(O(m + m \log m + m) = O(m \log m)\).

Since Step 1.1 runs \(n + 1\) times, the total complexity is \(O(mn \log m)\). Similarly, each iteration of Step 1.2 takes \(O(n \log n)\) time and it runs \(m + 1\) times. Thus, the complexity becomes \(O(mn \log n)\). Hence, the computation time of Step 1 is dominated by the process in Step 1.2 which takes \(O(mn \log n)\) time.
Step 2.1 runs for \(m(n+1)\) times while Step 2.2 runs for \((m+1)n\) times. Therefore, the complexity for Step 2 is \(O(mn)\). There are \(O(mn)\) pairs of \((K, L)\). Hence, Step 3 can be done in \(O(mn)\) time. Thus, the overall computation time is dominated by the computation time in Step 1 which is \(O(mn \log n)\).

2.2.5 BQP01 with \(Q\) as a \((2p + 1)\)-diagonal Matrix

In this section, we concentrate on the case that \(Q\) has a specific structure, say \(Q\) is a symmetric matrix with exactly \((2p + 1)\) diagonals where \(p = O(\log n)\) and all other entries are zero. Li et al. [72] gave a polynomial time algorithm for QP01 problem with tridiagonal matrix. There are various substitutions of variables throughout the process. Since BQP01 can be viewed as a QP01 problem with an appropriate objective function, we can apply this algorithm on our problem and we can show that when \(Q\) is a \((2p + 1)\)-diagonal matrix and \(p = O(\log n)\), the problem is polynomially solvable.

We consider \((x, y)\) as a vector \(z \in \mathbb{R}^{2n}\) where the odd-position entries are the entries of \(x\) and the even-position entries are the entries of \(y\). For \(k = 1, \ldots, 2n\), we denote

\[
\Theta_k(z) = f(z_1, \ldots, z_{k-1}, 0, z_{k+1}, \ldots, z_{2n}) \quad \text{and} \\
\Delta_k(z) = f(z_1, \ldots, z_{k-1}, 1, z_{k+1}, \ldots, z_{2n}) - \Theta_k(z).
\]

Note that we can write \(f(x, y) = f(z)\) as \(z_k \Delta_k(z) + \Theta_k(z)\). It is easy to see that any optimal solution \(z^*\) must satisfy

\[
z_k^* = \begin{cases} 
1 & \text{if } \Delta_k(z^*) > 0, \\
0 & \text{otherwise.}
\end{cases}
\]

The basic idea of this algorithm is that we can write \(f(z)\) as \(z_{2n} \Delta_{2n}(z) + \Theta_{2n}(z)\), where \(\Delta_{2n}(z)\) and \(\Theta_{2n}(z)\) are polynomial with \(2n - 1\) variables, says \(z_1, \ldots, z_{2n-1}\). We can find a polynomial \(\phi_{2n}(z) = z_{2n}\) with \(2n - 1\) variables and substitute in the equation. Now we obtain a polynomial representing \(f(z)\) with one variable fewer than the original. We call this polynomial \(f_{2n-1}(z) = \phi_{2n}(z) \Delta_{2n}(z) + \Theta_{2n}(z)\). We can repeat the same process to reduce the number of variables until we reach \(f_1(z)\). Then we have \(z_1^* = 1\) if \(f_1(1) > f_1(0)\) and \(z_1^* = 0\) otherwise. All other entries of \(z^*\) can be obtained from recursively computing...
$z_k^* = \phi_k(z_1^*, \ldots, z_{k-1}^*)$.

The most complicated part is determining $\phi_k(z)$ for $k = 1, \ldots, 2n$. Let $V_k$ be the set of all variable $z_1', \ldots, z_{|V_k|}'$ appearing in $\Delta_k(z)$. (Note that $V_k \subseteq \{z_1, \ldots, z_{k-1}\}$.) Consider all possibilities to assign value 0 or 1 to these $|V_k|$ variables. Thus, there are $2^{|V_k|}$ different assignments. Denote $g_1, \ldots, g_{2^{|V_k|}}$ be functions from $V_k$ to $\{0, 1\}$ corresponding to these assignments. For each assignment $g_h$, consider $\Delta_k(g_h(z)) = \Delta_k(g_h(z_1'), \ldots, g_h(z_{|V_k|}'))$. If $\Delta_k(g_h(z)) > 0$, we define a function with domain $V_k$ as follows:

$$g'_h(z'_l) = \begin{cases} 
z'_l & \text{if } g_h(z'_l) = 1, \\
1 - z'_l & \text{if } g_h(z'_l) = 0.
\end{cases}$$

Consider the product $\prod_{l \in V_k} g'_h(z'_l)$ for each assignment whose $\Delta_k(g_h(z)) > 0$. Adding all of these products forms the polynomial $\phi_k(z)$.

Note that in practice, we can simplify all polynomials in the algorithm by replacing $z_i^2$ with $z_i$ since $z_i^2 = z_i$ for all $z_i \in \{0, 1\}$. By doing this, each polynomial in our algorithm has at most $\binom{n}{2} + n + 1$ terms which is $O(n^2)$. The algorithm is described as follows.

**Algorithm 2.5.** [Algorithm 4.1 [72]]

**Step 1:** Define $f_k(z) = \sum_{i=1}^{n} \sum_{j=1}^{n} g_{ij} x_i y_j + \sum_{i=1}^{n} c_i x_i + \sum_{j=1}^{n} d_j y_j$ and set $k = 2n$.

**Step 2:** Compute $\Theta_k(z)$, $\Delta_k(z)$ and $\phi_k(z)$. Set $f_{k-1}(z) := \phi_k(z) \Delta_k(z) + \Theta_k(z)$. For all $l$ such that $\deg(z_l) = 2$, replace all $z_l^2$ in $f_{k-1}(z)$ by $z_l$. Set $k := k - 1$. If $k = 1$, go to **Step 3**. Otherwise, repeat this step.

**Step 3:** If $f_1(1) > f_1(0)$, set $x_1 := 1$. Otherwise, $x_1 := 0$. Set $h = 1$.

**Step 4:** Update $\phi_{h+1}(z)$. Set $h := h + 1$. If $h$ is even, set $h' := h/2$ and let $y_{h'} := \phi_h(z)$. Otherwise, set $h' := (h + 1)/2$ and let $x_{h'} := \phi_h(z)$.

If $h = 2n$, the algorithm terminates and output $z = (x, y)$. Otherwise, repeat this step.

Note that **Step 2** needs $2^{2p+1}$ computation time to compute $\Theta_k(z)$, $\Delta_k(z)$ and $\phi_k(z)$. Thus, when $Q$ is a $(2p + 1)$-diagonal matrix, the algorithm computes an optimal solution in polynomial time. Recall that $0_k$ is the vector in $\mathbb{R}^k$ whose all entries are zero. The proof is provided in Appendix B.
Theorem 2.11. Let $Q$ be a $(2p + 1)$-diagonal matrix and $p = O(\log n)$. Then Algorithm 2.5 solves BQP01 in polynomial time.
Chapter 3

Approximation Algorithms and Domination Analysis

In the previous chapter, we studied exact algorithms for BQP01 with polynomial time complexity. Here we continue our study on polynomial algorithms for BQP01, but now focus on heuristics rather than exact algorithms. Sometimes the terminology approximation algorithm is used for algorithms with a guaranteed performance ratio. We use this terminology in a broader context for any heuristic algorithm. The quality of a heuristic algorithm is measured using various criteria, including worst-case analysis, average-case analysis and empirical analysis. We will now consider some heuristic algorithms for BQP01 from the worst case analysis point of view.

The most popular measure used in the worst case analysis of a heuristic algorithm is the performance ratio [111]. Raghavendra and Steurer [97] and Alon and Naor [7] considered analysis of approximation algorithms for BQP01 based on performance ratio with some restrictions. The vast literature on approximation algorithms for the maximum weighted cut problems [8, 109] are also relevant to BQP01 in view of Theorem 1.1. Although these are all very interesting results, existing non-approximatably results [109] for max-clique, max-biclique and QP01 limits possible success in this line of research.

Our worst case analysis is based on average of the objective function values of all solutions and domination analysis. Average value based analysis of approximation algorithms
started in the Russian literature in the context of the Travelling Salesman Problem (TSP) [80, 100, 101, 102, 112]. Since then many researchers have considered this measure for TSP and other related problems [43, 46, 47, 48, 50, 92, 94]. To the best of our knowledge, such a study has not been carried out for BQP01.

Average value based analysis is closely related to another measure for worst-case performance of a heuristic algorithm, called domination ratio, introduced by Glover and Punnen [40]. Domination ratio is closely linked to the measure dominance number considered by many authors [43, 46, 47, 48, 49, 100, 101, 102] although the terminology of domination number was not used by many of these authors. Also, the concept of combinatorial leverage introduced by Glover [39] is linked to domination number of an algorithm.

We first discuss average based analysis. Here we analyze two well-known heuristics for BQP01 and give examples of BQP01 instances to show that these algorithms can be worse than average. Then three different algorithms with no worse than average guarantee are provided. Moreover, we show that even the average of objective function values can be obtained easily, computing median of all objective function values is an NP-hard problem. We close this chapter with the domination analyses on BQP01.

3.1 Average Value Based Analysis

Let \( P(Q, c, d) \) be an instance of BQP01 defined by matrix \( Q \in \mathbb{R}^{m \times n} \) and row vectors \( c \in \mathbb{R}^m \) and \( d \in \mathbb{R}^n \). In cases where we are not interested in a specific \((Q, c, d)\), we refer to the resulting instance of BQP01 as \( P \). Let \( \mathbb{B} \) be the collection of all instances of BQP01. For any instance of BQP01, its solution set is the entire set of \( \{0, 1\}^{m+n} \) and each solution in \( \{0, 1\}^{m+n} \) is represented as \((x, y)\) where \( x \in \{0, 1\}^m \) and \( y \in \{0, 1\}^n \). Hence, the total number of feasible solutions of BQP01 is \( 2^{m+n} \). Let \( A(Q, c, d) \) denote the average of the objective function value of all feasible solutions in \( \{0, 1\}^{m+n} \) and let \( f(x, y) = x^TQy + cx + dy \) for \((x, y) \in \{0, 1\}^{m+n}\). Then

\[
A(Q, c, d) = \frac{1}{2^{m+n}} \sum_{(x,y)\in\{0,1\}^{m+n}} f(x, y).
\]
Our next theorem gives a closed form expression to compute \( A(Q, c, d) \) in polynomial time.

**Theorem 3.1.** \( A(Q, c, d) = \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_{i} + \frac{1}{2} \sum_{j=1}^{n} d_{j} \).

**Proof.** Let \( M = 2^m \) and \( N = 2^n \). Also, let \( x^1, \ldots, x^M \) be all the elements in \( \{0, 1\}^m \) and \( y^1, \ldots, y^N \) be all the elements in \( \{0, 1\}^n \). Consider the sum

\[
\sum_{h=1}^{M} \sum_{k=1}^{N} \left( \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x^h_i y^k_j + \sum_{i=1}^{m} c_{i} x^h_i + \sum_{j=1}^{n} d_{j} y^k_j \right).
\]

For fixed pair \( i \) and \( j \), the number of appearances of \( q_{ij} \) in the sum is equal to the number of pairs \( x^h \) and \( y^k \) such that \( x^h_i = 1 \) and \( y^k_j = 1 \). There are \( 2^{m-1} \) \( x \)'s whose \( i^{th} \) entry is equal to 1 and \( 2^{n-1} \) \( y \)'s whose \( j^{th} \) entry is equal to 1. Thus, each \( q_{ij} \) appears in the sum for \( 2^{m-1} 2^{n-1} \) times. The number of appearances of \( c_{i} \) in the sum is equal to the number of pairs \( x^h \) and \( y^k \) such that \( x^h_i = 1 \) and \( y^k \) can be arbitrary. There are \( 2^{m-1} \) \( x \)'s whose \( i^{th} \) entry is equal to 1 and \( 2^n \) distinct \( y \)'s. Hence, the number of these pairs is \( 2^{m-1} 2^n \). Similarly, the number of appearances of \( d_{j} \) in the sum is equal to the number of pairs \( x^h \) and \( y^k \) such that \( y^k_j = 1 \) and \( x^h \) can be arbitrary. Therefore, the number of these pairs is \( 2^m 2^{n-1} \). Hence,

\[
\sum_{h=1}^{M} \sum_{k=1}^{N} f(x^h, y^k) = 2^{m-1} 2^{n-1} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + 2^{m-1} 2^n \sum_{i=1}^{m} c_{i} + 2^m 2^{n-1} \sum_{j=1}^{n} d_{j}.
\]

Since there are \( 2^m 2^n \) choices for \( x \) and \( y \),

\[
A(Q, c, d) = \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_{i} + \frac{1}{2} \sum_{j=1}^{n} d_{j}.
\]

The above result can be obtained by a simple probabilistic argument as well by considering \( x \) and \( y \) as random variables in \( \{0, 1\}^m \) and \( \{0, 1\}^n \) respectively and by computing the expected value. However, the algebraic proof discussed above is useful later in this section.

From Theorem 3.1, we can see that the average of the objective function values of all solutions can be computed in \( O(mn) \) time. Using a similar analysis, we can derive a formula to compute the average of all objective function values of solutions for an instance of QP01. This observation is summarized in the following corollary without proof.
Corollary 3.2. Let $Q \in \mathbb{R}^n \times \mathbb{R}^n$ and $c \in \mathbb{R}^n$. Then the average of the objective function value of all feasible solutions is 
\[
\frac{1}{4} \sum_{i \neq j} q_{ij} + \frac{1}{2} \sum_{i=1}^{n} c_i.
\]

We say that a solution $(x, y)$ is not worse than average or is a no worse than average solution if $f(x, y) \geq A(Q, c, d)$. If a heuristic algorithm always outputs a no worse than average solution, we call it a no worse than average algorithm.

In a probabilistic sense, computing a solution with objective function value no worse than $A(Q, c, d)$ appears to be simple. For example, one can choose random solutions repeatedly. As the number of trials increases, the best solution produced in the process is no worse than $A(Q, c, d)$ with high probability.

However, producing a solution that is guaranteed to be no worse than average may need some careful design strategy. To motivate this hypothesis, let us consider some local search algorithms that performed very well in experimental analysis [33, 57]. We will show that these algorithms could get trapped at a local optimum that is worse than average.

3.1.1 Worse than Average Algorithms

We first consider a local search algorithm called the alternating algorithm studied by various authors [33, 57, 74]. Alternating algorithm is a special case of block coordinate descent methods, used extensively in non-linear programming literature. There are two families of variables, $x$ and $y$ in BQP01. The algorithm alternatively fixes one variable and chooses the optimal value of the other and the process is continued until a local optimum is reached.

Let $x^0$ be a specific value of $x$. Then the best choice of $y$, say $y^0 = (y^0_1, \ldots, y^0_n)$, subject to the constraint that $x = x^0$ is given by

\[
y_j^0 = \begin{cases} 
1 & \text{if } \sum_{i=1}^{m} q_{ij} x_i^0 + d_j > 0, \\
0 & \text{otherwise.}
\end{cases}
\]
Likewise, if we fix $y$ at, say $y^0$, the best value of $x$, say $x^1 = (x^1_1, \ldots, x^1_m)$ is given by

$$
    x^1_i = \begin{cases} 
    1 & \text{if } \sum_{j=1}^n q_{ij} y_{j}^0 + c_i > 0, \\
    0 & \text{otherwise.} 
    \end{cases}
$$

Note that the alternating algorithm can start by fixing either $x$ or $y$ in the first iteration. If we start by fixing $x$, we call the resulting algorithm $x$-first alternating algorithm. On the other hand, if $y$ is fixed first, we call it $y$-first alternating algorithm. From a theoretical analysis point of view, the starting variable is irrelevant. However, in practice, the local solution to which the algorithm converges depends on the starting value and type of variable we fixed first.

Experimental analysis by Karapetyan and Punnen [57] shows that on average, the alternating algorithm gives reasonably good solutions. To the best of our knowledge, worst case analysis of this algorithm has not been studied before. Our next theorem shows that an output from the alternating algorithm can be worse than average.

**Theorem 3.3.** *The local optimum produced by the alternating algorithm could be arbitrarily bad and worse than average.*

**Proof.** Consider the BQP01 instance where $m = n$, $c = d = 0^T_n$, $q_{11} = 1$, $q_{nn} = M$ where $M$ is a large positive number and $q_{ij} = 0$ for all other entries. Let $x^0$ be the initial solution where $x^0_i = 1$ if $i = 1$ and $x^0_i = 0$ otherwise.

In the first iteration, the alternating algorithm chooses $y^0_1 = 1$ and $y^0_j = 0$ for $j \neq 1$. In the next step, the algorithm will give $x^1$ where $x^1_i = 1$ if $i = 1$ and $x^1_i = 0$ otherwise. We can see that $x^0 = x^1$ and there is no improvement in this step since $(x^0, y^0)$ and $(x^1, y^0)$ give the same objective function value equal to 1. So the algorithm stops and outputs $(x^1, y^0)$.

However, the optimal objective function value of this instance of BQP01 is $M + 1$. The average objective function values of this instance is $(M + 1)/4$. Hence, the solution from the alternating algorithm is worse than average if $M > 3$. Moreover, given $N > 0$, we can find $M > 0$ that is large enough to make the ratio of the optimal objective function value
and the objective function value of the solution, which is $M + 1$, larger than $N$. Therefore, the solution produced by the alternating algorithm can be arbitrarily bad.

Let us now consider a more general neighborhood, which is a variation of the $k$-exchange neighborhood studied for various combinatorial optimization problems. For any $(x^0, y^0) \in \{0, 1\}^{m+n}$, let $N^{hk}$ be the set of solutions in $\{0, 1\}^{m+n}$ obtained by switching at most $h$ components of $x^0$ and at most $k$ components of $y^0$. It is easy to see that $|N^{hk}| = \sum_{i=0}^{h} \binom{m}{i} \sum_{j=0}^{k} \binom{n}{j}$. The best solution in this neighborhood can be identified in polynomial time for fixed $h$ and $k$. A more powerful neighborhood is $N^\alpha = N^{m\alpha} \cup N^{an}$ and the size of this neighbourhood is

$$|N^\alpha| = 2^m \sum_{j=0}^{\alpha} \binom{n}{j} + \sum_{i=0}^{\alpha} \binom{m}{i} 2^n - \sum_{i=0}^{\alpha} \binom{m}{i} \sum_{j=0}^{\alpha} \binom{n}{j}.$$

When $\alpha$ is fixed, this neighborhood can be searched for an improving solution in polynomial time [57]. It may be noted that a solution produced by the alternating algorithm is locally optimal with respect to the neighborhood $N^0 = N^{m0} \cup N^{0n}$. Glover et al. [42] considered the neighborhoods $N^1$, $N^2$ and $N^{1.1}$. They provided fast and efficient algorithms for exploring these neighbourhoods supported by detailed computational analysis. They also developed a tabu search algorithm using these neighbourhoods in a hybrid form. Computational results with the algorithm provided very high quality solutions, improving several benchmark instances. Nonetheless, our next theorem shows that even such very powerful local search algorithms could provide solutions with objective function values that are inferior to $A(Q, c, d)$ even if we allow $\alpha$ to be a function of $n$.

**Theorem 3.4.** A locally optimal solution to BQP01 with respect to the neighbourhood $N^\alpha = N^{m\alpha} \cup N^{an}$ could be worse than average for any $\alpha \leq \lfloor n/5 \rfloor$.

**Proof.** Consider the matrix $Q$ defined as

$$q_{ij} = \begin{cases} 
\lambda & \text{if } i = m, j = n, \\
-1 & \text{if } i = m \text{ or } j = n \text{ but } (i, j) \neq (m, n), \\
a & \text{otherwise},
\end{cases}$$

where $\lambda = 6(n - 5)(n - 1)/(5n)$ and $a = 6/n$, and choose $c$ and $d$ as zero vectors. Without loss of generality, we assume $m = n$. Otherwise, we can extend the matrix $Q$ into an $n \times n$
matrix by adding \( n-m \) rows of zeros and extending the vector \( c \) into an \( n \)-vector by making the last \( n-m \) entries zeros. We also assume that \( n \) is a multiple of 5. Also, let \( \alpha = n/5 \) and assume \( n \geq 6 \). Consider the solution \( (x^0, y^0) \) where \( x^0_n = y^0_n = 1 \) and all other components are zero. We show that a local optimum with respect to \( N^\alpha \) and initial solution \( (x^0, y^0) \) can be worse than average.

For \( r = 0, 1, \ldots, \alpha \), let \( N^r_x(0) \) be the set of \( x \) obtained from switching exactly \( r \) entries of \( x^0 \) and last entry of \( x \) was switched from 1 to 0, and \( N^r_x(1) \) be the set of \( x' \) obtained from switching exactly \( r \) entries of \( x^0 \) and the last entry of \( x \) was not switched. We define \( N^s_y(0) \) and \( N^s_y(1) \) in the similar way. Note that for any \( (x, y) \in N^{\alpha \alpha^*} \), \( x \in N^r_x(0) \cup N^r_x(1) \) and \( y \in N^s_y(0) \cup N^s_y(1) \) for some \( 0 \leq r \leq \alpha \), \( 0 \leq s \leq n \) and for any \( (x, y) \in N^n \), \( x \in N^r_x(0) \cup N^r_x(1) \) and \( y \in N^s_y(0) \cup N^s_y(1) \) for some \( 0 \leq r \leq n \), \( 0 \leq s \leq \alpha \). Thus, for \( (x, y) \in N^\alpha \) we have

\[
f(x, y) = \begin{cases} 
(r-1)(s-1)a & \text{if } x \in N^r_x(0), y \in N^s_y(0) \text{ for } (r, s) \in I^a_n \cup I^a_\alpha, \\
(ra-1)(s-1) & \text{if } x \in N^r_x(1), y \in N^s_y(0) \text{ for } (r, s) \in I^a_n \cup I^a_\alpha, \\
(r-1)(sa-1) & \text{if } x \in N^r_x(0), y \in N^s_y(1) \text{ for } (r, s) \in I^a_n \cup I^a_\alpha, \\
ars-r-s+\lambda & \text{if } x \in N^r_x(1), y \in N^s_y(1) \text{ for } (r, s) \in I^a_n \cup I^a_\alpha,
\end{cases}
\]

where \( I^a_q = \{0, 1, \ldots, p\} \times \{0, 1, \ldots, q\} \). Thus, \( (x^0, y^0) \) is locally optimal with respect to \( N^\alpha \) if and only if

\[
(r-1)(s-1)a \leq \lambda \text{ for all } (r, s) \in I^a_n \cup I^a_\alpha, \\
(ra-1)(s-1) \leq \lambda \text{ for all } (r, s) \in I^a_n \cup I^a_\alpha, \\
(r-1)(sa-1) \leq \lambda \text{ for all } (r, s) \in I^a_n \cup I^a_\alpha, \\
ars-r-s \leq 0 \text{ for all } (r, s) \in I^a_n \cup I^a_\alpha.
\]

For \( (x, y) \in N^n \), we have \( 0 \leq r \leq n \) and \( 0 \leq s \leq \alpha \). Therefore, for any \( x \in N^r_x(0) \) and \( y \in N^s_y(0) \), we have \( f(x, y) = (r-1)(s-1)a \leq (n-1)(\alpha-1)a \). Similarly, for \( (x, y) \in N^{\alpha \alpha} \), we have \( 0 \leq r \leq \alpha \) and \( 0 \leq s \leq n \). Therefore, we obtain \( f(x, y) \leq (\alpha-1)(n-1)a \). Consider any \( x \in N^r_x(0) \) and \( y \in N^s_y(0) \) when \( (x, y) \in N^n \). We get \( f(x, y) = (r-1)(sa-1) \leq (n-1)(\alpha a-1) \). On the other hand, when \( (x, y) \in N^{\alpha \alpha} \), this bound become \( f(x, y) \leq (\alpha-1)(na-1) \). In the case that \( x \in N^r_x(1) \) and \( y \in N^s_y(0) \), we obtain \( f(x, y) = (s-1)(ra-1) \leq (\alpha-1)(na-1) \) when \( (x, y) \in N^{\alpha \alpha} \). If \( (x, y) \in N^n \), we
get \( f(x, y) \leq (n - 1)(aa - 1) \) instead. As for \( x \in N^x_n(1) \) and \( y \in N^y_n(1) \). We can see that \( ars - r - s \) is a bilinear function on variables \( r \) and \( s \). If \((x, y) \in N^{\alpha n} \), its maximum value is obtained when \((r, s) \in \{(0, 0), (0, \alpha)(n, 0)(\alpha, n)\} \). Since \( n \geq 6 \) and \( \alpha = n/5 \), it is easy to see that the maximum value of \( ars - r - s \) this case is 0 or \( a\alpha n - \alpha - n \). Similarly, for \((x, y) \in N^{\alpha n} \), \( ars - r - s \) reaches its maximum value when \((r, s) \in \{(0, 0), (0, \alpha)(n, 0)(\alpha, n)\} \). The maximum value for this case is also 0 or \( a\alpha n - \alpha - n \). When the maximum value is 0, \( ars - r - s + \lambda = \lambda \). Hence, these \((x, y)\) are dominated by \((x^0, y^0)\) with objective function value \( \lambda \). Thus, it is enough to consider only when \( ars - r - s = a\alpha n - \alpha - n \).

We can conclude that \((x^0, y^0)\) is locally optimal if

\[
(\alpha - 1)(n - 1)a \leq \lambda, \tag{3.1}
\]

\[
(n - 1)(aa - 1) \leq \lambda, \tag{3.2}
\]

\[
(\alpha - 1)(na - 1) \leq \lambda, \tag{3.3}
\]

\[
a\alpha n - \alpha - n \leq 0. \tag{3.4}
\]

Recall that \( a = 6/n = (\alpha + n)/\alpha n \). Then (3.4) holds with equality. Moreover, \( a\alpha n - \alpha - n = (n/5) \cdot n \cdot (6/n) - n/5 - n = 0 \geq 0 \). Thus, \( a\alpha n - \alpha - n \) is still the maximum value of \( ars - r - s \) with this choice of \( a \).

Note that we have the condition \( n \geq 6 \). Thus, \( a = 6/n \leq 1 \). It is easy to see that \( (\alpha - 1)(n - 1)a = (\alpha a - a)(n - 1) \geq (n - 1)(aa - 1) \) and \( (\alpha - 1)(n - 1)a = (\alpha - 1)(na - a) \geq (\alpha - 1)(na - 1) \). Hence, inequality (3.1) implies inequality (3.2) and (3.3). Therefore, we can consider only inequality (3.1) and (3.4).

Recall that \( \lambda = 6(n - 5)(n - 1)/(5n) = (\alpha - 1)(n - 1)a \). We have

\[
A(Q, 0^m_m, 0^n_n) - f(x, y) = \frac{1}{4n} \left( \frac{2}{5} n^2 + \frac{58}{5} n - 12 \right),
\]

which is positive since \( n \geq 6 \). It follows that this solution is worse than average. \( \square \)

As an immediate corollary, we have the following result.
Corollary 3.5. For any fixed $h$ and $k$, the objective function value of a locally optimal solution with respect to the neighborhood $N^{hk}$ could be worse than $A(Q,c,d)$ for sufficiently large $m$ and $n$.

3.2 No Worse than Average Algorithms

From the algorithms discussed above, it is clear that to get a performance guarantee of no worse than average, careful algorithmic strategies are required. In this section, we will discuss various algorithms that provide solutions that are guaranteed to be no worse than average.

3.2.1 The Trivial Algorithm

Let $\alpha = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij}$, $\beta = \sum_{i=1}^{m} c_{i}$ and $\gamma = \sum_{j=1}^{n} d_{j}$.

Theorem 3.6. $A(Q,c,d) \leq \max\{\alpha + \beta + \gamma, \beta, \gamma, 0\}$.

Proof. Let $u$ and $v$ be real numbers in $[0,1]$. Choose $x \in \{0,1\}^{m}$, $y \in \{0,1\}^{n}$ such that $x_{i} = u$ for all $i = 1,\ldots,m$ and $y_{j} = v$ for all $j = 1,\ldots,n$. Define function $g$ from $[0,1]^{2}$ to $\mathbb{R}$ by $g(u,v) = f(x,y) = \alpha uv + \beta u + \gamma v$. Note that $g(1/2,1/2) = A(Q,c,d)$. Thus, $\max\{g(u,v) : (u,v) \in [0,1]^{2}\} \geq A(Q,c,d)$. Since $g$ is a bilinear function, its maximum is attained at an extreme point of the square $[0,1]^{2}$. These extreme points are precisely $(0,0),(1,0),(0,1),(1,1)$. Thus,

$$\max\{g(u,v) : (u,v) \in [0,1]^{2}\} = \max\{g(0,0), g(1,0), g(0,1), g(1,1)\}$$

$$= \max\{0,\beta,\gamma,\alpha + \beta + \gamma\}$$

and the result follows. \qed

The proof of Theorem 3.6 immediately suggests an algorithm to compute a solution that is no worse than average. We call it the trivial algorithm.

Algorithm 3.1. Trivial algorithm

Step1: Compute $f(1_{m},1_{n}), f(0_{m},1_{n}), f(1_{m},0_{n})$ and $f(0_{m},0_{n})$.

Step2: Output the best solution.
From Theorem 3.6, the trivial algorithm produces a solutions with objective function value no worse than average.

We can compute $\alpha$, $\beta$ and $\gamma$ in $O(mn)$ time. Also, the objective function value of the four solutions can be identified in $O(mn)$ time. Then the complexity of the trivial algorithm is $O(mn)$. Interestingly, if $\alpha$, $\beta$ and $\gamma$ are given, then we can identify a solution to BQP01 with objective function value no worse than $A(Q,c,d)$ in $O(1)$ time. To compute a solution that is no worse than average, such a trivial approach was somewhat unexpected.

The solution produced by the trivial algorithm may not be of much practical value. Nevertheless, the simple upper bound on $A(Q,c,d)$ established by Theorem 3.6 is very interesting and have interesting consequences in domination analysis. It also makes local search algorithm no worse than average. For example, any multistart local search algorithm that starts with each of $(1_m, 1_n), (0_m, 1_n), (1_m, 0_n), (0_m, 0_n)$ obviously gives a performance guarantee no worse than average. We will address the consequence of this simple observation later by showing that only two starts are required in some cases to achieve this guarantee.

Since the output from trivial algorithm is so simple and may not be good in practice, we offer other more complicated algorithms with no worse than average guarantee.

### 3.2.2 Rounding Algorithm

Let us now consider another algorithm that guarantees a solution no worse than average. The algorithm takes fractional vectors $x \in [0,1]^m$ and $y \in [0,1]^n$ and applies a rounding scheme to produce a solution for BQP01. Let $x \in [0,1]^m$ and $y \in [0,1]^n$. We consider the solutions $y^* \in \{0,1\}^n$ and $x^* \in \{0,1\}^m$ given by

$$y_j^* = \begin{cases} 1 & \text{if } d_j + \sum_{i=1}^{m} q_{ij}x_i > 0, \\ 0 & \text{otherwise}, \end{cases}$$

(3.5)
and

\[ x_i^* = \begin{cases} 
1 & \text{if } c_i + \sum_{j=1}^{n} q_{ij} y_j^* > 0, \\
0 & \text{otherwise.} 
\end{cases} \tag{3.6} \]

Note that \( x^* \) is the optimal binary vector when \( y \) is fixed at \( y^* \), and equation (3.5) rounds the \( y \) to \( y^* \) using a prescribed rounding criterion. Thus, we call the process of constructing \((x^*, y^*)\) from \((x, y)\) a round-y optimize-x algorithm or RyOx-algorithm.

The next theorem establishes a lower bound on the objective function value of the solution obtained by the RyOx-algorithm. The notation

\[ f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j \]

was introduced by assuming \( x \) and \( y \) as binary vectors. We extend the same notation when \( x \) and \( y \) are real numbers.

**Theorem 3.7.** \( f(x^*, y^*) \geq f(x, y) \).

**Proof.** By construction of \( y^* \), we have

\[ \left( \sum_{i=1}^{m} q_{ij} x_i + d_j \right) y_j \leq \left( \sum_{i=1}^{m} q_{ij} x_i + d_j \right) y_j^* \]

for all \( j = 1, \ldots, n \). We also have

\[ \left( \sum_{j=1}^{n} q_{ij} y_j^* + c_i \right) x_i \leq \left( \sum_{j=1}^{n} q_{ij} y_j^* + c_i \right) x_i^* \]

for all \( i = 1, \ldots, m \) by construction of \( x^* \). Thus,

\[
f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j \\
= \left[ \sum_{j=1}^{n} \left( \sum_{i=1}^{m} q_{ij} x_i + d_j \right) \right] y_j + \sum_{i=1}^{m} c_i x_i \\
\leq \sum_{j=1}^{n} \left( \sum_{i=1}^{m} q_{ij} x_i + d_j \right) y_j^* + \sum_{i=1}^{m} c_i x_i \\
= \sum_{j=1}^{n} d_j y_j^* + \sum_{i=1}^{m} \left( \sum_{j=1}^{n} q_{ij} y_j^* + c_i \right) x_i
\]

49
\[
\leq \sum_{j=1}^{n} d_j y_j^* + \sum_{i=1}^{m} \left( \sum_{j=1}^{n} q_{ij} y_j^* + c_i \right) x_i^* \\
= f(x^*, y^*).
\]

Note that \((x^*, y^*)\) can be constructed in \(O(mn)\) time when \((x, y) \in [0,1]^{m+n}\) is given.

We can also start from rounding \(x\) first to obtain \(x^0 \in \{0,1\}^m\) and choose optimal \(y = y^0\) by fixing \(x\) at \(x^0\). This process can be done using the rounding scheme given by the following equations:

\[
x_i^0 = \begin{cases} 
1 & \text{if } c_i + \sum_{j=1}^{n} q_{ij}y_j > 0, \\
0 & \text{otherwise}, 
\end{cases}
\]

and

\[
y_j^0 = \begin{cases} 
1 & \text{if } d_j + \sum_{i=1}^{m} q_{ij}x_i^0 > 0, \\
0 & \text{otherwise}. 
\end{cases}
\]

The process of constructing \((x^0, y^0)\) is called round-x optimize-y algorithm or RxOy-algorithm. The complexity of RxOy-algorithm is also \(O(mn)\).

**Theorem 3.8.** \(f(x^0, y^0) \geq f(x, y)\).

We omit the proof of Theorem 3.8 since it follows along the same line as the proof of Theorem 3.7.

**Corollary 3.9.** A solution \((\bar{x}, \bar{y})\) for BQP01 satisfying \(f(\bar{x}, \bar{y}) \geq A(Q, c, d)\) can be obtained in \(O(mn)\) time using RxOy-algorithm or RyOx-algorithm.

**Proof.** Let \(x_i = 1/2\) for all \(i = 1, \ldots, m\) and \(y_j = 1/2\) for all \(j = 1, \ldots, n\). Then

\[
f(x,y) = \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} \frac{1}{2} \cdot \frac{1}{2} + \sum_{i=1}^{m} c_i \frac{1}{2} + \sum_{j=1}^{n} d_j \frac{1}{2} \\
= \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_i + \frac{1}{2} \sum_{j=1}^{n} d_j \\
= A(Q, c, d).
\]
However, \((x, y)\) is not feasible for BQP01. Now, we choose \((\bar{x}, \bar{y})\) as the output of either RyOx-algorithm or RxOy-algorithm. Then the result follows from Theorems 3.7 or 3.8.

### 3.2.3 No worse than Average One-pass Algorithm

In this section, we describe a one-pass heuristic, assigning the value for each entry only once and it will not be changed. Thus, the algorithm ends in \(n\) iterations. Our one-pass heuristic uses the idea of conditional expectation by computing the average of the value relating to the objective function value. We tried a standard implementation of the method of conditional expectation fixing variables \(x_1, \ldots, x_m, y_1, \ldots, y_n\) in turn. But we observed that it was worse than the rounding algorithm in practice. So we propose here a variant of the method that appears better in practice for this problem.

We describe how the algorithm works. Starting from \(J = \emptyset\) and \(y = 0_n\), we choose one index \(j\) of \(y\) to be added in \(J\) and switch \(y_j\) from 0 to 1 in each iteration. To choose this index \(j\), we define the potential \(p(j)\) of each \(j\). This represents a conditional expectation. We add index \(j\) with the highest positive potential in each iteration and update \(J\) and the potential of all indices. Keep adding entries to \(J\) until there are no entries of positive potential or \(J = \{1, \ldots, n\}\). At the end, the algorithm outputs \(y^J\), where \(y_j^J = 1\) if \(j \in J\) and \(y_j^J = 0\) otherwise, and its optimal pair \(x(y^J)\).

We first note that when there is a change on \(y\) value, the change on the objective function value occurs on \(\sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{j=1}^{n} d_j y_j\) while \(\sum_{i=1}^{m} c_i x_i\) is still the same.

**Definition 3.10.** Denote potential \(p(j)\) the average of the change in the value of

\[
\sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{j=1}^{n} d_j y_j
\]

when a new set of indices of \(y\) including index \(j\) are added in \(J\). The average is computed over all choices of \(2^m\) \(x\)'s and all sets of indices that have not been added to \(J\) containing index \(j\).

Let \(N = \{1, \ldots, n\}\) and \(\bar{J} = N \setminus J\) be the complement of \(J \subseteq N\). We can show that

\[
p(j) = d_j + \frac{1}{2} \sum_{k \in J \setminus \{j\}} d_k + \frac{1}{2} \sum_{i=1}^{m} q_{ij} + \frac{1}{4} \sum_{i=1}^{m} \sum_{k \in J \setminus \{j\}} q_{ik}.
\]
To compute \( p(j) \), we consider \( j \in \bar{J} \). There are \( 2^{\bar{J}-1} \) choices of the set of indices that have not been added to \( J \) that includes index \( j \). Let \( t \in \{1, \ldots, 2^{\bar{J}-1}\} \) and \( S_t \) be the set of indices other than \( j \) selected in each choice. We consider the sum of

\[
p(h, S_t) = d_j + \sum_{k \in S_t} d_k + \sum_{i=1}^{m} q_{ij} x^h + \sum_{i=1}^{m} \sum_{k \in S_t} q_{ik} x^h
\]

over all choices of \( S_t \) and \( x^h \) in \( \{0,1\}^m \), which is the change on objective function value when \( x^h \) is chosen and \( S_t \cup \{j\} \) is added to \( J \). Hence, this sum contains \( 2^m 2^{\bar{J}-1} = 2^{\bar{J}+m-1} \) of different \( p(h, S_t) \).

It is clear that \( d_j \) appears in every \( p(h, S_t) \) in the sum. Thus, \( d_j \) occurs \( 2^{\bar{J}+m-1} \) times in the sum. For other index \( k \in \bar{J} \), it appears in \( p(h, S_t) \) where \( S_t \) contains \( k \). There are exactly \( 2^{\bar{J}-2} \) \( S_t \)'s containing \( k \) and each \( S_t \) can be paired with \( 2^m x^h \)'s. Therefore, each \( d_k \) is found in the sum for \( 2^{\bar{J}+m-2} \) times.

As for \( q_{ij} \), it appears in the sum for any choice of \( 2^{\bar{J}-1} \) \( S_t \)'s. There are \( 2^{m-1} x^h \)'s where \( x_i^h = 1 \). Hence, \( q_{ij} \) is in \( 2^{\bar{J}+m-2} \) terms of the sum. Finally, we consider \( q_{ik} \) where \( k \in \bar{J} \). The number of set \( S_t \) containing \( k \) is \( 2^{\bar{J}-2} \) and each of them can be paired with different \( 2^{m-1} x^h \)'s where \( x_i^h = 1 \). Thus, there are \( 2^{\bar{J}+m-3} \) occurences of \( q_{ik} \).

Then we can express the formula for \( p(j) \) as follows:

\[
p(j) = \frac{1}{2^{\bar{J}+m-1}} (2^{\bar{J}+m-1} d_j + 2^{\bar{J}+m-2} \sum_{k \in S_t} d_k
+ 2^{\bar{J}+m-2} \sum_{i=1}^{m} q_{ij} + 2^{\bar{J}+m-3} \sum_{i=1}^{m} \sum_{k \in S_t} q_{ik})
= d_j + \frac{1}{2} \sum_{k \in J \setminus \{j\}} d_k + \frac{1}{2} \sum_{i=1}^{m} q_{ij} + \frac{1}{4} \sum_{i=1}^{m} \sum_{k \in J \setminus \{j\}} q_{ik}.
\]

Let \( p_0 = \frac{1}{2} \sum_{k=1}^{n} d_k + \frac{1}{4} \sum_{i=1}^{m} \sum_{k=1}^{n} q_{ik} \). When \( \bar{J} = N \), we can see that \( p(j) = p_0 + d_j/2 + \sum_{i=1}^{m} q_{ij}/4 \). Moreover, when entry \( j^m \neq j \) enters \( J \), the new potential \( p(j) \) with respect to this new \( J \) is \( p(j) - d_j m/2 - \sum_{i=1}^{m} q_{ij m}/4 \). Let \( x^*(y^*) \) be the vector in \( \{0,1\}^m \) that optimizes
the objective function value \( f(x, y) \) when \( y \) is fixed to be \( y^* \). One can verify that

\[
x^*(y^*)_i = \begin{cases} 
1 & \text{if } c_i + \sum_{j=1}^{n} q_{ij}y_j^* > 0, \\
0 & \text{otherwise}.
\end{cases}
\]

**Algorithm 3.2.** (Main Algorithm)

**Step 1:** Set \( J := \emptyset \) and compute \( p_0 \). If \( p_0 \leq 0 \), go to **Step 4**. Otherwise, for all \( j \in N \), set \( p(j) := p_0 + d_j/2 + \sum_{i=1}^{m} q_{ij}/4 \).

**Step 2:** If \( \bar{J} = \emptyset \), skip to **Step 4**. Otherwise, let \( j^m \) be the index such that \( p(j^m) = \max\{p(j') : j' \in \bar{J}\} \).

**Step 3:** If \( p(j^m) \leq 0 \), go to the next step. Otherwise, add \( j^m \) in \( J \) and update all potentials by set \( p(j') := p(j') - d_{j^m}/2 - \sum_{i=1}^{m} q_{ij^m}/4 \) for all \( j' \in \bar{J} \). Go back to **Step 2**.

**Step 4:** Set \( y_j := 1 \) if \( j \in J \) and \( y_j := 0 \) otherwise. Compute \( x(y) \) and return \((x(y), y)\).

In **Step 1**, we can compute \( p_0 \) in \( O(mn) \) time and obtain all \( p(j) \) in \( O(mn) \) time. It takes \( O(n) \) time to find \( j^m \) in **Step 2** and \( O(mn) \) time to update all potential \( p(j') \) in **Step 3**. These two steps are repeated at most \( n \) times. The outcome in **Step 4** can be computed in \( O(mn) \) time. Therefore, the complexity of Algorithm 3.2 is \( O(mn + n(n+mn) + mn) = O(mn^2) \). The next theorem shows that the algorithm is no worse than average.

**Theorem 3.11.** Algorithm 3.2 gives a no worse than average solution for BQP01.

**Proof.** Here we split the proof into two cases depending on the value of \( p_0 \).

**Case 1** \( p_0 \leq 0 \) That means \( \frac{1}{2} \sum_{j=1}^{n} d_j + \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} \leq 0 \). Let \( I^+ = \{i : c_i > 0\} \). Then

\[
A(Q, c, d) = \frac{1}{2} \sum_{j=1}^{n} d_j + \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_i \\
\leq \frac{1}{2} \sum_{i=1}^{m} c_i \leq \frac{1}{2} \sum_{i \in I^+} c_i \leq \sum_{i \in I^+} c_i = f(x(0_n), 0_n),
\]

which is the output from Algorithm 3.2 in this case.

**Case 2** \( p_0 > 0 \) We need to consider two subcases upto the termination of the algorithm. When we perform Algorithm 3.2 with \( p_0 > 0 \), it finishes when \( \bar{J} = \emptyset \) which implies that
\( J = N \), or stops at the beginning of **Step 3** when there is no \( j' \in \bar{J} \) in that iteration having positive potential.

**Case 2.1 The algorithm terminates with \( J = N \).** Then the solution from the algorithm is \( (x, y) = (x(1_n), 1_n) \) where \( 1_n \) is the vector in \( \mathbb{R}^n \) whose entries are all 1 and \( x(1_n) \) is optimally chosen. Let \( A^1(Q, c, d) \) be the average of the objective function value among all solutions with \( y = 1_n \). Thus, \( f(x, y) \geq A^1(Q, c, d) \). We consider

\[
A^1(Q, c, d) - A(Q, c, d) = \sum_{j=1}^{n} d_j + \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_i
- \frac{1}{2} \sum_{j=1}^{n} d_j - \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} - \frac{1}{2} \sum_{i=1}^{m} c_i
= \frac{1}{2} \left( \sum_{j=1}^{n} d_j + \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} \right) = \frac{p_0}{2}.
\]

Hence, \( f(x, y) \geq A^1(Q, c, d) \geq A(Q, c, d) \).

**Case 2.2 The algorithm terminates with \( J \neq N \).** The solution is \( (x, y) = (x(y'), y') \) here \( y' \) is the vector in \( \{0, 1\}^n \) where \( y'_j = 1 \) if \( j \in J \) and \( y'_j = 0 \) otherwise. We know that when the algorithm stops, every \( j' \in \bar{J} \) has potential at most zero since \( j^m \) is the index in \( \bar{J} \) with maximum potential. Hence, the sum \( \sum_{j' \in \bar{J}} p(j') \) must be at most zero as well.

Here we consider the value of the sum. We know that for each \( j \), \( d_j \) appears with coefficient 1 one time in \( p(j) \) and with coefficient \( 1/2 \) in \( |J| - 1 \) \( p(k) \)'s where \( k \in \bar{J} \setminus \{j\} \). Each \( q_{ij} \) also occurs with coefficient \( 1/2 \) only once in \( p(j) \) and with coefficient \( 1/4 \) in \( |J| - 1 \) \( p(k) \)'s where \( k \in \bar{J} \setminus \{j\} \). Therefore,

\[
\sum_{j \in J} p(j) = \sum_{j \in J} d_j + \frac{|J| - 1}{2} \sum_{j \in J} d_j + \frac{1}{2} \sum_{i=1}^{m} \sum_{j \in J} q_{ij} + \frac{|J| - 1}{4} \sum_{i=1}^{m} \sum_{j \in J} q_{ij}
= \frac{|J| + 1}{2} \sum_{j \in J} d_j + \frac{|J| + 1}{4} \sum_{i=1}^{m} \sum_{j \in J} q_{ij}.
\]
\[
\sum_{j=1}^{n} d_j + \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \sum_{i=1}^{m} c_i
\]

\[
- \frac{|J| + 1}{2} \sum_{j \in J} d_j - \frac{|J| + 1}{4} \sum_{i=1}^{m} \sum_{j \in J} q_{ij} - \frac{1}{2} \sum_{i=1}^{m} c_i
\]

\[
= (|J| + 1) \left( \frac{1}{2} \sum_{j=1}^{n} d_j + \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + \frac{1}{2} \sum_{i=1}^{m} c_i \right)
\]

\[
- (|J| + 1) \left( \frac{1}{2} \sum_{j \in J} d_j + \frac{1}{4} \sum_{i=1}^{m} \sum_{j \in J} q_{ij} + \frac{1}{4} \sum_{i=1}^{m} c_i \right) - \frac{|J| + 1}{4} \sum_{i=1}^{m} c_i.
\]

Denote by

\[
A^J(Q, c, d) = \sum_{j \in J} d_j + \frac{1}{2} \sum_{i=1}^{m} c_i + \frac{1}{2} \sum_{i=1}^{m} \sum_{j \in J} q_{ij}
\]

the average of objective function values of all solutions with \( y = y^J \). Since the sum is at most zero, we have

\[
0 \geq \sum_{j \in J} p(j) = (|J| + 1) A(Q, c, d) - \frac{|J| + 1}{2} A^J(Q, c, d) - \frac{|J| + 1}{4} \sum_{i=1}^{m} c_i.
\]

Since \(|J| + 1 > 0\), we obtain

\[
A^J(Q, c, d) \geq 2A(Q, c, d) - \frac{1}{2} \sum_{i=1}^{m} c_i.
\]

From the fact that \( x \) is optimally chosen, we get

\[
f(x, y) \geq A^J(Q, c, d) \geq 2A(Q, c, d) - \frac{1}{2} \sum_{i=1}^{m} c_i
\]

\[
= A(Q, c, d) + \frac{1}{2} \sum_{j=1}^{n} d_j + \frac{1}{4} \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij}
\]

\[
= A(Q, c, d) + p_0 > A(Q, c, d).
\]

\[\square\]

### 3.2.4 Median Objective Function Value

We have seen different algorithms with guarantee that a solution produced is no worse than average. When the objective function values follow the normal distribution,
such a solution is also no worse than median of the objective function values of all solutions. However, we do not have a nice formula or algorithm to compute a median of the objective function values. Unlike the average, computing this median value is a difficult problem as illustrated in the following theorem.

**Theorem 3.12.** Computing a median of all objective function value of BQP01 is NP-hard.

**Proof.** We show that the Partition Problem, which is known to be NP-hard [32], can be reduced to this problem. Assume that we have a polynomial time algorithm to determine the median of all objective function values $x^T Q y + c x + d y$. Let $S$ be a set of $n$ integers $\{a_1, \ldots, a_n\}$ and $N = \{1, \ldots, n\}$. The Partition Problem is to determine whether there is a partition $N_1$ and $N_2$ of $N$ such that $\sum_{j \in N_1} a_j = \sum_{j \in N_2} a_j$.

From an instance of Partition Problem, we construct an instance of BQP01 as follows: Define $c = \begin{bmatrix} 0 & 0 \end{bmatrix}$ and $d_j = a_j$ for $j = 1, \ldots, n$. Define the $2 \times n$ matrix $Q$ as $q_{1j} = a_j \epsilon$ and $q_{2j} = -a_j \epsilon$ for a positive number $\epsilon = 1/(\sum_{j=1}^n a_j + 1)$. Let $y^H$ be the incidence vector for index set $H \subseteq N$, that is $y^H_j = 1$ if and only if $j \in H$ and $y^H_j = 0$ otherwise.

Note that for each choice of $H$, there are four corresponding solutions $(x, y^H)$, and their objective function values are

$$f(0, 0, y^H) = \sum_{j \in H} a_j,$$

$$f(1, 0, y^H) = \sum_{j \in H} a_j (1 + \epsilon),$$

$$f(0, 1, y^H) = \sum_{j \in H} a_j (1 - \epsilon),$$

$$f(1, 1, y^H) = \sum_{j \in H} a_j.$$

Thus, for each $H$, we have $f(0, 1, y^H) < f(0, 0, y^H) = f(1, 1, y^H) < f(1, 0, y^H)$. Denote $S(H) = \sum_{j \in H} a_j$. Let $S_1, \ldots, S_\nu$ be an increasing order of all different sums $S(H)$ for all choices of index set $H \subseteq N$. Let $n_k$ be the number of index sets $H$ such that $S(H) = S_k$ and $H^{k,1}, \ldots, H^{k,n_k}$ be those $n_k$ index sets.

Before we continue our proof, we give some useful observations here.
Observation 1 For any subset $H^{k,l}$, there exists $l'$ such that $H^{k,l} = N\setminus H^{\nu+1-k,l'}$. Since $H^{k,l}, H^{\nu+1-k,l'} \subseteq N$, we also have $H^{\nu+1-k,l'} = N\setminus H^{k,l}$.

Observation 2 $n_k = n_{\nu+1-k}$. (This result follows from the first observation.)

Observation 3 $S_k + S_{\nu+1-k} = \sum_{j \in N} a_j$. Since we have $S_k = S(H^{k,l})$ for some $l$ and from Observation 1, we get $H^{k,l} = N\setminus H^{\nu+1-k,l'}$ for some $l'$, we obtain

$$S_k = S(N\setminus H^{\nu+1-k,l'}) = \sum_{j \in N\setminus H^{\nu+1-k,l'}} a_j$$

$$= \sum_{j \in N} a_j - \sum_{j \in H^{\nu+1-k,l'}} a_j = \sum_{j \in N} a_j - S_{\nu+1-k}.$$ 

Therefore, $S_k + S_{\nu+1-k} = \sum_{j \in N} a_j$.

Consider an ascending arrangement of $f(x,y)$ for all solutions $(x,y)$ of the BQP01 constructed. This can be grouped as blocks of values $B_1 < B_2 < \ldots < B_\nu$ where the block $B_k$ has the structure

\[
\underbrace{S_k(1-\epsilon)}_\text{repeated } n_k \text{ times} = \cdots = \underbrace{S_k(1-\epsilon)}_\text{repeated } n_k \text{ times} < S_k = \cdots = S_k < \underbrace{S_k(1+\epsilon)}_\text{repeated } n_k \text{ times} = \cdots = S_k(1+\epsilon)
\]

for $k = 1, 2, \ldots, \nu$. There are $2^n$ different index sets $H$ and $2^{n+2}$ different solutions. Place all objective function values in a nondecreasing order. Since there are even number of solutions, the median of all objective function value are the $(2^n+1)^{th}$ value and the $(2^n+1+1)^{th}$ value.

We claim that the two medians are the same if and only if $\nu$ is odd. If $\nu$ is odd, it is easy to see from the structure of blocks $B_k$ and Observation 2 that they are the same since both medians are equal to $S_{\nu+1/2}$. On the other hand, when $\nu$ is even, the $(2^n+1)^{th}$ value is $S_{\nu+1/2}(1+\epsilon)$ while the $(2^n+1+1)^{th}$ value is $S_{\nu+1/2}(1-\epsilon)$. Suppose that these two values are the same, we obtain

$$S_{\nu+1/2}(1+\epsilon) = S_{\nu+1/2}(1-\epsilon)$$

$$S_{\nu+1} + \epsilon S_{\nu+1} = S_{\nu+1} - \epsilon S_{\nu+1}$$

$$\epsilon S_{\nu+1} + \epsilon S_{\nu+1} = S_{\nu+1} - S_{\nu+1}$$

$$\epsilon(S_{\nu+1} + S_{\nu+1}) = S_{\nu+1} - S_{\nu+1} \geq 1.$$
From Observation 3, $S_\nu S_\nu + S_\nu S_{\nu-1} = S_\nu S_{\nu+1} = \sum a_j$. Hence, the left hand size is equal to $\epsilon \sum_{j \in N} a_j = \sum_{j \in N} a_j / (\sum_{j \in N} a_j + 1) < 1$, a contradiction. Hence, the two medians are not the same.

Now we show that a required partition exists if and only if both the median values are the same. Assume that there exists a required partition $(P_1, P_2)$ of $S$. Define $H_1 = \{j : a_j \in P_1\}$ and $H_2 = \{j : a_j \in P_2\}$. It follows that $H_1 = H^{k,l}$ for some $k \in \{1, \ldots, \nu\}$ and $l \in \{1, \ldots, n_k\}$. From the construction, we have $H_2 = N \setminus H^{k,l}$. From Observation 1, there exists $l'$ such that $H^{\nu+1-k,l} = N \setminus H^{k,l}$. Hence, $S(H_1) = S_k$ and $S(H_2) = S_{\nu+1-k}$. Since $S_1, S_2, \ldots, S_\nu$ are all different, $P_1$ and $P_2$ give the same sum if and only if $k = \nu + 1 - k$. It implies that $(\nu + 1)/2 = k \in \mathbb{Z}$. Thus, $\nu$ must be odd. From the previous claim, both medians are the same. As for the only if part, assume that the median values are the same. From the claim, $\nu$ is odd. Let $K = (\nu + 1)/2$. Then we have the median value equal to $S_K$. Consider index set $H^{K,1}$ of $N$. From Observation 1, there exists $l$ such that $H^{K,1} = N \setminus H^{\nu+1-K,l} = N \setminus H^{\nu+1-(\nu+1)/2,l} = N \setminus H^{(\nu+1)/2,l} = N \setminus H^{K,l}$. Hence, we have $S(H^{K,1}) = S_K = S(H^{K,l})$. Since $H^{K,1} = N \setminus H^{K,l}$, $\{a_j : j \in H^{K,1}\}$ and $\{a_j : j \in H^{K,l}\}$ is a partition where $\sum_{j \in H^{K,1}} a_j = S_K = \sum_{j \in H^{K,l}} a_j$ as required.

Thus, if we can determine the median of the objective function values in polynomial time, we can answer the Partition Problem in polynomial time by forming the BQP01 as described above, finding the median of all objective function values, and checking the equality of the medians. \qed

It may be noted that the above theorem does not rule out (unless P=NP) the possibility of a polynomial algorithm that guarantees a solution no worse than the median value. Our next section provide additional insight into this intriguing question.

### 3.3 Domination Analysis

Let us now analyze heuristic algorithms for BQP01 from the point of view of domination analysis.
Let \((x, y), (x', y') \in \{0, 1\}^{m+n}\) be two solutions of BQP01. We say that \((x', y')\) dominates \((x, y)\) if \(f(x, y) \leq f(x', y')\). Let \(\Gamma\) be a heuristic algorithm for BQP01 and recall that \(\mathcal{B}\) is the collection of all instances of BQP01. For an instance \(P \in \mathcal{B}\), denote
\[
\mathcal{G}^\Gamma(P) = \{(x, y) \in \{0, 1\}^{m+n} : f(x, y) \leq f^\Gamma_{\min}\},
\]
where \(f^\Gamma_{\min}\) is the worst possible objective function value of the solutions from algorithm \(\Gamma\) for the instance \(P\). Then the domination number of \(\Gamma\), denoted by \(\text{dom}\, N(\Gamma)\) is given by
\[
\text{dom}\, N(\Gamma) = \inf_{P \in \mathcal{B}} |\mathcal{G}^\Gamma(P)|.
\]
Another closely related measure is the domination ratio of \(\Gamma\), denoted by \(\text{dom}\, R(\Gamma)\) which is defined as
\[
\text{dom}\, R(\Gamma) = \inf_{P \in \mathcal{B}} \frac{|\mathcal{G}^\Gamma(P)|}{|F|},
\]
where \(F\) is the family of all feasible solutions. Note that \(\Gamma\) is an exact algorithm if and only if its domination ratio is equal to 1. Thus, it is interesting to develop polynomial time approximation algorithms with domination ratio close to 1. Since the number of feasible solutions are the same for all instances of BQP01 with the same dimension of \(Q\), the concepts of \(\text{dom}\, N(\Gamma)\) and \(\text{dom}\, R(\Gamma)\) are equivalent in our case. More precisely, \(\text{dom}\, N(\Gamma) = 2^{m+n} \text{dom}\, R(\Gamma)\)

Our next theorem links the average of the objective function values of BQP01 to domination number (domination ratio) of an algorithm.

**Theorem 3.13.** If \(\Gamma\) is a heuristic algorithm for BQP01 that produces a solution \((x^0, y^0)\) such that \(f(x^0, y^0) \geq A(Q, c, d)\), then \(\text{dom}\, N(\Gamma) \geq 2^{m+n-2}\) and \(\text{dom}\, R(\Gamma) \geq 1/4\).

**Proof.** Let \(\mathbb{W} = \{(x, y) \in \{0, 1\}^{m+n} : f(x, y) \leq A(Q, c, d)\}\). By hypothesis, \((x^0, y^0)\) dominates all solutions in \(\mathbb{W}\). We claim that \(|\mathbb{W}| \geq 2^{m+n-2}\). It then follows immediately that \(\text{dom}\, N(\Gamma) \geq 2^{m+n-2}\) and \(\text{dom}\, R(\Gamma) \geq 1/4\).

Partition the set of all solutions of BQP01 into \(2^{m+n-2}\) partite sets of size four, and show that the worst solution of each partite set is worse than average. Denote \(\overline{x} = 1_m - x\) and \(\overline{y} = 1_n - y\). For any \((x, y) \in \{0, 1\}^{m+n}\), we define
\[
P(x, y) = \{(x, y), (x, \overline{y}), (\overline{x}, y), (\overline{x}, \overline{y})\}.
\]
Note that \( \bar{x} = 1_m - x = 1_m - 1_m + x = x \) and \( \bar{y} = 1_n - y = 1_n - 1_n + y = y \). Hence, we have

\[
P(x, y) = \{(x, y), (x, y), (\bar{x}, y), (\bar{x}, y)\},
\]
\[
P(x, y) = \{(x, y), (\bar{x}, y), (x, y), (x, y)\} \text{ and }
\]
\[
P(x, y) = \{(\bar{x}, y), (\bar{x}, y), (x, y), (x, y)\}.
\]

Therefore, \( P(x) = P(x, y) = P(x, y) = P(x, y) \). Let \( (x, y) \) and \( (x', y') \) be two different solutions. If \( (x', y') \in P(x, y) \), then \( P(x, y) = P(x', y') \). Otherwise, \( (x', y') \notin P(x, y) \). Suppose that there exists \( (x^*, y^*) \in P(x, y) \cap P(x', y') \). Then \( P(x, y) = P(x^*, y^*) = P(x', y') \). Hence, \( (x', y') \in P(x', y') = P(x, y) \), a contradiction. Therefore, we can partition the set of all solutions into \( 2^{m+n}/4 = 2^{m+n-2} \) disjoint sets \( P(x^k, y^k) \) for \( k = 1, 2, \ldots, 2^{m+n-2} \).

Consider

\[
f(x, y) + f(x, y) + f(x, y) + f(x, y)
\]
\[= f(x, y) + f(1_m - x, y) + f(x, 1_n - y) + f(1_m - x, 1_n - y)
\]
\[= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i y_j + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j y_j
\]
\[+ \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} (1 - x_i) y_j + \sum_{i=1}^{m} c_i (1 - x_i) + \sum_{j=1}^{n} d_j y_j
\]
\[+ \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i (1 - y_j) + \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} d_j (1 - y_j)
\]
\[+ \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} (1 - x_i) (1 - y_j) + \sum_{i=1}^{m} c_i (1 - x_i) + \sum_{j=1}^{n} d_j (1 - y_j)
\]
\[= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} x_i (1 - x_i) y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} (x_i + 1 - x_i)(1 - y_j)
\]
\[+ \sum_{i=1}^{m} c_i (x_i + 1 - x_i + x_i + 1 - x_i) + \sum_{j=1}^{n} d_j (y_j + y_j + 1 - y_j + 1 - y_j)
\]
\[= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} (1 - y_j) + \sum_{i=1}^{m} c_i (2) + \sum_{j=1}^{n} d_j (2)
\]
\[= \sum_{i=1}^{m} \sum_{j=1}^{n} q_{ij} + 2 \sum_{i=1}^{m} c_i + 2 \sum_{j=1}^{n} d_j
\]
\[= 4A(Q, c, d).
\]
For \( k = 1, 2, \ldots, 2^m + n - 2 \), let \((\hat{x}^k, \hat{y}^k)\) be a solution with the minimum objective function value in \( P(x^k, y^k) \). Then we have

\[
4f(\hat{x}^k, \hat{y}^k) \leq f(x^k, y^k) + f(x^k, \hat{y}^k) + f(\hat{x}^k, y^k) + f(\hat{x}^k, \hat{y}^k)
\]

\[
\leq 4A(Q, c, d).
\]

It follows that \((\hat{x}^k, \hat{y}^k)\) gives the objective function value that is not better than average. Thus, \{\((\hat{x}^k, \hat{y}^k) : k = 1, \ldots, 2^m + n - 2\) \} \subseteq \mathbb{W} and hence \(|\mathbb{W}| \geq 2^{m+n-2} \). \(\square\)

We can give an example to show that the bound obtained in the proof of Theorem 3.13 is tight.

**Proposition 3.14.** The lower bound on \(|\mathbb{W}|\) established in the proof of Theorem 3.13 is the best possible.

**Proof.** Consider an instance \( P(Q, c, d) \) where matrix \( Q \) are defined by

\[
q_{ij} = \begin{cases} 
-1 & \text{if } i = m \text{ and } j = n, \\
0 & \text{otherwise},
\end{cases}
\]

where \( c = 0^T_m \) and \( d = 0^T_n \). Then we have \( A(Q, c, d) = -1/4 \) and \( \mathbb{W} = \{(x, y) : x_m = y_n = 1\} \).

For each entry \( x_k \) where \( k = 1, \ldots, m - 1 \), there are two possibilities for its value, says zero or one. Thus, there are \( 2^{m-1} \) choices for \( x \) whose \( m^{th} \) entry is one. Similarly, we have \( 2^{n-1} \) choices for \( y \) whose \( n^{th} \) entry is one. Hence, there are exactly \( 2^{m-1} \cdot 2^{n-1} = 2^{m+n-2} \) elements in \( \mathbb{W} \). Therefore, the bound obtained in Theorem 3.13 is tight. \(\square\)

Recall that in the previous section, we considered various algorithms that guarantee a solution with objective function value no worse than average. From Theorem 3.13, the domination numbers of these algorithms are at least \( 2^{m+n-2} \) and their domination ratios are at least \( 1/4 \). This also brings an interesting connection between average and first quartile of the objective function values. Any solutions that is no worse than average is also no worse than the first quartile of the objective function values. However, computing the first quartile itself is NP-hard. The proof follows with appropriate modification of Theorem 3.12.
In Section 3.1.1, we established that the objective function value produced by the alternating algorithm could be arbitrarily bad and could be less than $A(Q,c,d)$. However, the domination number of this algorithm is at least $2^m + 2^n - 1$. This is easy to verify since $x$ and $y$ are optimally chosen in each step, by fixing one.

We can strengthen the alternating algorithm to achieve a better domination number by carefully choosing the initial solution and running the algorithm twice.

**Theorem 3.15.** The best solution amongst the solutions produced by the $x$-first alternating algorithm with starting solutions $(1_m,1_n)$ and $(0_m,0_n)$ dominates $2^n - 1 (2^{m-1} + 3)$ solutions.

**Proof.** Let $(x^*, y^*)$ be the best solution obtained in this two phase alternating algorithm. Consider the case when the starting solution is $(x^0, y^0) = (1_m,1_n)$ and $y^1$ is optimally chosen with respect to $x^0$. Thus,

$$f(x^*, y^*) \geq f(x^0, y^1) \geq \max\{f(1_m,1_n), f(1_m,0_n)\}.$$

Similarly, when $(x^0, y^0) = (0_m,0_n)$ is the starting solution, we have

$$f(x^*, y^*) \geq \max\{f(0_m,0_n), f(0_m,1_n)\}.$$

Thus, by Theorem 3.6, $f(x^*, y^*) \geq A(Q,c,d)$ and hence by Theorem 3.13, $(x^*, y^*)$ dominates at least $2^m+n-2$ solutions.

As for the remaining solutions that are dominated by $(x^*, y^*)$, we partition the set of all solutions into $2^{m+n-2}$ disjoint sets $P(x^k, y^k)$ as in the proof of Theorem 3.13. For each $P(x^k, y^k)$, let $(x^{km}, y^{km})$ be the unique solution from $P(x^k, y^k)$ with the minimum objective function value and let $P_m = \{(x^{km}, y^{km}) : k = 1, \ldots, 2^{m+n-2}\}$. Let $B = \{(1_m, y), (0_m, y) : y \in \{0,1\}^n\}$. We can see that $|B| = 2^{n+1}$. Recall that we denote $\bar{y} = 1_n - y$. For any $y \in \{0,1\}^n$, we have $P(0_m, y) = P(1_m, y)$ and $P(0_m, y) = \{(0_m, y), (1_m, y), (0_m, \bar{y}), (1_m, \bar{y})\}$. Thus, $B$ can be partitioned into sets of the form $P(0_m, y^k)$ and there are $2^{n+1}/4 = 2^{n-1}$ such sets. By the construction of $P_m$, exactly one element from each $P(0_m, y^k)$ is in $P_m$. Therefore, $|P_m \cap B| = 2^{n-1}$ and hence $|P_m \cup B| = 2^{m+n-2} + 2^{n-1} - 2^{n-1} = 2^{m-1} (2^{m-1} + 3)$. 

\[\square\]
From Theorem 3.15 the domination ratio of the $x$-first alternating algorithm is at least $1/4 + 3/2^{m+1}$ when starting solutions are specifically chosen and the algorithm is applied twice. Note that to achieve this domination ratio, we simply need to perform only one iteration each, when the algorithm starts with $(1_m, 1_n)$ and $(0_m, 0_n)$. Thus, we can achieve this domination ratio in polynomial time. A similar result can be derived for $y$-first alternating algorithm.

**Theorem 3.16.** Let $(x', y')$ be the best solution among the solutions produced by RyOx-algorithm starting with solutions $(1_m, 1_n)$ and $(0_m, 0_n)$ dominates $2^{n-1}(2^{m-1} + 3)$ solutions.

**Proof.** Let $(v^*, w^*)$ be the output from RyOx-algorithm starting with $(1_m, 1_n)$ and $(x^*, y^*)$ be the output from RyOx-algorithm starting with $(0_m, 0_n)$. By Theorem 3.7 and the fact that $v^*$ and $x^*$ are optimally chosen, $f(v^*, w^*) \geq \max\{f(0_m, 1_n), f(1_m, 1_n)\}$ and $f(x^*, y^*) \geq \max\{f(0_m, 0_n), f(1_m, 0_n)\}$. Since $f(x^*, y^*) = \max\{f(v^*, w^*), f(x^*, y^*)\}$, we have

$$f(x^*, y^*) \geq \max\{f(0_m, 1_n), f(1_m, 1_n), f(0_m, 0_n), f(1_m, 0_n)\}.$$

By Theorem 3.6, $f(x^*, y^*) \geq A(Q, c, d)$ and hence by Theorem 3.13, $(x^*, y^*)$ dominates at least $2^{m+n-2}$ solutions. The proof for the existence of the remaining dominated $3 \cdot 2^{n-1}$ solutions is the same as in the proof of Theorem 3.15. \qed

We have established lower bounds on domination number for several algorithms for BQP01. We now provide an upper bound on the domination number for any polynomial time algorithm for BQP01.

**Theorem 3.17.** Unless P=NP, there are no polynomial time algorithms for BQP01 with domination number larger than $2^{m+n} - 2^{\left\lceil \frac{m+n}{\alpha} \right\rceil}$ for any fixed positive integers $a$ and $b$ such that $\alpha = a/b > 1$.

**Proof.** Let $a$ and $b$ be positive integers such that $\alpha = a/b > 1$. Suppose for a contradiction that we have a polynomial time algorithm $\Omega$ for BQP01 with dominance number at least $2^{m+n} - 2^{\left\lceil \frac{m+n}{\alpha} \right\rceil} + 1$. We show that we can use $\Omega$ to compute an optimal solution to BQP01 which is an NP hard problem.
Consider a BQP01 with instance $P(Q, c, d)$. We construct an $abm \times abn$ matrix $Q^*$ by defining

$$q^*_{ij} = \begin{cases} q_{ij} & \text{if } 1 \leq i \leq m \text{ and } 1 \leq j \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, we define

$$c^*_i = \begin{cases} c_i & \text{if } 1 \leq i \leq m, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$d^*_j = \begin{cases} d_j & \text{if } 1 \leq j \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

From the construction, we can see that $Q^*$ is composed of the submatrix $Q$ in the upper left section, and all other entries are zero. As for $c^*$, it contains one copy of $c$ in the first $m$ entries and the remaining $(ab - 1)m$ entries are zero. Similarly, $d^*$ contains one copy of $d$ in the first $n$ entries and the remaining $(ab - 1)n$ entries are zero.

It is easy to see that for any optimal solution $(x^*, y^*)$ of $P(Q^*, c^*, d^*)$, we can obtain an optimal solution $(x^o, y^o)$ for $P(Q, c, d)$ by defining $x^o_i = x^*_i$ for $i = 1, \ldots, m$ and $y^o_j = y^*_j$ for $j = 1, \ldots, n$. Moreover we can see that for each optimal solution $(x^o, y^o)$ of $P(Q^*, c^*, d^*)$ by extending $(ab - 1)m$ entries of $x^o$ and $(ab - 1)n$ entries of $y^o$. Thus, $P(Q^*, c^*, d^*)$ has at least $2(\frac{(ab - 1)(m+n)}{2})$ optimal solutions.

Applying algorithm $\Omega$ gives a solution with objective function value not worse than at least $2^{abm + abn} - 2^{\left\lfloor \frac{abm + abn}{a+b} \right\rfloor} + 1 = 2^{ab(m+n)} - 2^{\left\lfloor \frac{ab(m+n)}{a+b} \right\rfloor} + 1 = 2^{ab(m+n)} - 2^{b^2(m+n)} + 1$ solutions.

Since $a/b = \alpha > 1$, we have $a > b$. Note that the integrality of $a$ and $b$ implies that $a - 1 \geq b$. Hence, we have $ab - 1 \geq ab - b = b(a - 1) \geq b^2$. It follows that $2(\frac{(ab - 1)(m+n)}{2}) \geq 2^{b^2(m+n)}$. Therefore, this solution is an optimal solution for $P(Q^*, c^*, d^*)$. Since, we can recover an optimal solution for $P(Q, c, d)$ from this solution and $a$ and $b$ are fixed, this is a polynomial reduction from BQP01 of size $m \times n$ to BQP01 of size $abm \times abn$. By the NP hardness of BQP01, we get a contradiction.

Note that Theorem 3.17 implies that unless P=NP, there are no polynomial time algorithms with dominance ratio greater than

$$\frac{2^{m+n} - 2^{\left\lfloor \frac{m+n}{a} \right\rfloor} - (m+n)}{2^{m+n}} = 1 - 2^{\left\lfloor \frac{m+n}{a} \right\rfloor} - (m+n) \leq 1 - 2^{\frac{m+n}{a} - (m+n)} = 1 - 2^{(1-a)(m+n)}.$$
for any fixed positive integers $a$ and $b$ such that $\alpha = a/b$. Moreover, we can relax the assumption that $a$ and $b$ are fixed but another condition need to be added to guarantee polynomial time reduction.

**Corollary 3.18.** Unless $P=NP$, there are no polynomial time algorithms for BQP01 with dominance number larger than $2^{m+n} - 2\left\lceil \frac{m+n}{\alpha} \right\rceil$ for any fixed $\alpha = a/b > 1$ where $a$ and $b$ are relatively prime and bounded above by a polynomial function of the input length of BQP01.

We can also establish a similar result for QP01 using the arguments modified appropriately.

**Corollary 3.19.** Unless $P=NP$, there are no polynomial time algorithms for QP01 with dominance number larger than $2^n - 2\left\lceil \frac{n}{\alpha} \right\rceil$ for any fixed $\alpha = a/b > 1$ where $a$ and $b$ are relatively prime and bounded above by a polynomial function of $n$. 
Chapter 4

Bipartite Boolean Quadric Polytope

As well as QP01, we are interested in the linearization of BQP01 and its polytope. In this chapter, we present some basic properties of the polytope as well as some families of facets and valid inequalities. We provide trivial inequalities and $I_{mm22}$ Bell inequalities which define facets of the polytope. Odd-cycle inequalities are a family of valid inequalities which are facet-defining under a condition on the underlying cycle. Besides, we discuss strategies to obtain a facet or a valid inequality from that of another closely related polytope. We use rounding coefficient technique to obtain a valid inequality from a facet-defining inequality of the polytope corresponding to QP01. Cut polytope is another polytope relating to our polytope. We can obtain various families of facet-defining inequalities and valid inequalities from those of the cut polytope via covariance mapping and triangular elimination.

Similar to QP01, linearization of BQP01 can be achieved by introducing a new variable $z_{ij} = x_i y_j$ which has zero-one value and satisfies

\begin{align*}
x_i + y_j - z_{ij} &\leq 1, \\
-x_i + z_{ij} &\leq 0, \\
-y_j + z_{ij} &\leq 0, \\
-z_{ij} &\leq 0, \\
x_i, y_j, z_{ij} & \text{ integer},
\end{align*}

for $i = 1, \ldots, m$ and $j = 1, \ldots, n$. 

66
We denote the convex hull of the solutions of (4.1)-(4.5)

\[ \text{BQP}^{m,n} = \text{conv} \{ (x, y, z) \in \mathbb{R}^{m+n+mn} : (x, y, z) \text{ satisfies } (4.1), \ldots, (4.5) \} \]

the Bipartite Boolean Quadric Polytope and

\[ \text{BQP}^{m,n}_{LP} = \{ (x, y, z) \in \mathbb{R}^{m+n+mn} : (x, y, z) \text{ satisfies } (4.1), \ldots, (4.4) \} \]

the linear relaxation of \( \text{BQP}^{m,n} \). It is relevant in developing algorithms for various applications of the BQP01 model.

### 4.1 Basic Properties of \( \text{BQP}^{m,n} \)

Padberg [83] gave many results on the boolean quadric polytope \( \text{QP}^n \). We try to establish similar properties for \( \text{BQP}^{m,n} \) and many proofs in this section follow his outline.

It is easy to see that the following two facts, known for \( \text{QP}^n \) (see e.g. [83]), are also true for \( \text{BQP}^{m,n} \) and \( \text{BQP}^{m,n}_{LP} \).

**Proposition 4.1.** If \( (x, y, z) \in \text{BQP}^{m,n} \) or \( (x, y, z) \in \text{BQP}^{m,n}_{LP} \), then \( 0 \leq x_i \leq 1 \), \( 0 \leq y_j \leq 1 \) and \( 0 \leq z_{ij} \leq 1 \) for all \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \). It follows that \( \text{BQP}^{m,n} \subseteq \{0, 1\}^{m+n+mn} \).

**Proof.** Let \( i \in \{1, \ldots, m\} \) and \( j \in \{1, \ldots, n\} \). From (4.2) and (4.4), we have \( x_i \geq z_{ij} \geq 0 \). Similarly, from (4.3) and (4.4), we have \( y_j \geq z_{ij} \geq 0 \). Adding (4.1) and (4.2), we get \( y_j = x_i + y_j - z_{ij} - x_i + z_{ij} \leq 1 + 0 = 1 \). Similarly, we can add (4.1) and (4.3) to obtain \( x_i = x_i + y_j - z_{ij} - y_j + z_{ij} \leq 1 + 0 = 1 \). Moreover, \( z_{ij} \leq x_i \leq 1 \). Then we are done. \( \square \)

Since all \( 2^{m+n} \{0, 1\}-points \) are solutions of BQP01, this proposition implies that these \( 2^{m+n} \) points are vertices of \( \text{BQP}^{m,n} \). However, it does not imply that they are also vertices of \( \text{BQP}^{m,n}_{LP} \) in which non-integral vertices are allowed. The next proposition shows that these \( 2^{m+n} \) points are vertices of \( \text{BQP}^{m,n}_{LP} \) as well.

**Proposition 4.2.** Let \( (x, y, z) \in \{0, 1\}^{m+n+mn} \) be a point satisfying (4.1)-(4.4). Then \( (x, y, z) \) is a vertex of \( \text{BQP}^{m,n}_{LP} \).
Proof. Let \((x, y, z) \in \{0, 1\}^{m+n+mn}\) be a point satisfying (4.1)-(4.4). Suppose that \((x^1, y^1, z^1)\) and \((x^2, y^2, z^2)\) are points in \(\text{BQP}_{LP}^{m,n}\) such that \((x, y, z) = \frac{1}{2}((x^1, y^1, z^1) + (x^2, y^2, z^2))\).

For \(i = 1, \ldots, m\), \(x_i = \frac{1}{2}(x_i^1 + x_i^2)\). From Proposition 4.1, \(x_i^1, x_i^2 \in [0, 1]\). We obtain \(x_i^1 = x_i^2 = 0 = x_i\) if \(x_i = 0\) and \(x_i^1 = x_i^2 = 1 = x_i\) if \(x_i = 1\). We get the same result for \(y\) and \(z\). Thus, \((x, y, z) = (x^1, y^1, z^1) = (x^2, y^2, z^2)\). Hence, \((x, y, z)\) is a vertex of \(\text{BQP}_{LP}^{m,n}\). \(\square\)

We denote by \(u_i\) the vector in \(\mathbb{R}^m\) with all zero entries except the \(i^{th}\) which is equal to 1, by \(v_j\) the vector in \(\mathbb{R}^n\) with all zero entries except the \(j^{th}\) which is equal to 1 and by \(w_{ij}\) the vector in \(\mathbb{R}^{mn}\) with all zero entries except the \(ij\) entry which is equal to 1. Recall that \(0_k\) is the zero vector of length \(k\).

**Proposition 4.3.** \(\text{BQP}^{m,n}\) and \(\text{BQP}_{LP}^{m,n}\) are full-dimensional. That is \(\dim \text{BQP}^{m,n} = \dim \text{BQP}_{LP}^{m,n} = m + n + mn\).

**Proof.** Consider \(m + n + mn + 1\) vertices in \(\text{BQP}^{m,n}\), namely \(0_{m+n+mn}, (u_i, 0_{n+mn})\) for \(i = 1, \ldots, m\), \((0_m, v_j, 0_{mn})\) for \(j = 1, \ldots, m\) and \((u_i, v_j, w_{ij})\) for \(i = 1, \ldots, m\) and \(j = 1, \ldots, n\).

Let \(\alpha_0, \alpha_i, \beta_j\), and \(\gamma_{ij}\) be scalars for \(i = 1, \ldots, m\) and \(j = 1, \ldots, n\). Assume that

\[
\alpha_0(0_{m+n+mn+1}) + \sum_{i=1}^{m} \alpha_i(u_i, 0_{n+mn+1}) + \sum_{j=1}^{n} \beta_j(0_m, v_j, 0_{mn+1}) + \sum_{i=1}^{m} \sum_{j=1}^{n} \gamma_{ij}(u_i, v_j, w_{ij+1}) = 0.
\]

From the \((m + n + 1)^{th}\) entry to \((m + n + mn)^{th}\) entry, we have \(\gamma_{ij} = 0\) for every pair \(ij\). It follows that for each \(i^{th}\) entry, we have \(0 = \alpha_i + \sum_{j=1}^{n} \gamma_{ij} = \alpha_i\) for \(i = 1, \ldots, m\). Similarly, we have \(\beta_j = 0\) for \(j = 1, \ldots, n\).

Moreover, from the \((m + n + mn + 1)^{th}\) entry, we have \(0 = \alpha_0 + \sum_{i=1}^{m} \alpha_i + \sum_{j=1}^{n} \beta_j + \sum_{i=1}^{m} \sum_{j=1}^{n} \gamma_{ij} = \alpha_0\). Hence, \(\text{BQP}^{m,n}\) contains \(m + n + mn + 1\) affinely independent extreme points. Therefore, the polytope has dimension \(m + n + mn\) and it is full-dimensional.

For \(\text{BQP}_{LP}^{m,n}\), recall that a polytope is full-dimensional if and only if it has an interior point. Consider the point \(\omega = (x, y, z) \in \mathbb{R}^{m+n+mn}\) where all \(x_i = y_j = 1/2\) and \(z_{ij} = 1/4\).
for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. It is easy to see that $\omega$ satisfies (4.1)-(4.4) and none of them holds with equality. Therefore, it is an interior point.

\section{4.1.1 Trivial Facets of the Polytope}

Next, we show that all inequalities in the form (4.1)-(4.4) define facets of $\text{BQP}^{m,n}$ and $\text{BQP}^{m,n}_{LP}$. We call these facets *trivial facets* of $\text{BQP}^{m,n}$. Recall that for any full-dimensional polytope, each facet-defining inequality is unique up to multiplication by a positive scalar.

To prove that an equality defines a facet, we show that the plane corresponding to the inequality contains $m + n + mn - 1$ affinely independent points in the polytope. Here, we denote $u_i, v_j, w_{ij}$ and $0_n$ in the same way as in the previous part.

**Lemma 4.4.** The inequalities $-z_{ij} \leq 0$ defines a facet of $\text{BQP}^{m,n}$ and $\text{BQP}^{m,n}_{LP}$ for $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

**Proof.** Let $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$. Define

$$F = \{(x, y, z) \in \text{BQP}^{m,n} : z_{ij} = 0\}.$$ 

We consider $(u_h, 0_n, 0_{mn}) \in \text{BQP}^{m,n}$ for $h = 1, \ldots, m$ and $(0_m, v_k, 0_{mn}) \in \text{BQP}^{m,n}$ for $k = 1, \ldots, n$. It is clear that these $m + n$ points are in $F$. Moreover, we pick $(u_h, v_k, w_{hk}) \in \text{BQP}^{m,n}$ for $h = 1, \ldots, m$, $k = 1, \ldots, n$ and $hk \neq ij$. These $mn - 1$ points also satisfy $z_{ij} = 0$.

Assume that

$$\sum_{h=1}^{m} \alpha_h(u_h, 0_n, 0_{mn}, 1) + \sum_{k=1}^{n} \beta_k(0_m, v_k, 0_{mn}, 1) + \sum_{hk \neq ij} \gamma_{hk}(u_h, v_k, w_{hk}, 1) = 0_{m+n+mn+1},$$

where $\alpha_h$ for $h = 1, \ldots, m$, $\beta_k$ for $k = 1, \ldots, n$ and $\gamma_{hk}$ for $h = 1, \ldots, m$, $k = 1, \ldots, n$ and $hk \neq ij$ are scalar. Consider entries $z_{hk}$ where $hk \neq ij$. We have $\gamma_{hk} = 0$ for all $h = 1, \ldots, m$, $k = 1, \ldots, n$ and $hk \neq ij$.

69
It follows that for entries $x_h$ for $h = 1, \ldots, m$, we obtain $\alpha_h = \alpha_h + \sum_{k=1}^{n} \gamma_{hk} = 0$ for $h = 1, \ldots, m$ and $h \neq i$, and $\alpha_i = \alpha_i + \sum_{k \neq j} \gamma_{ik} = 0$. Similarly, for entries $y_k$ for $k = 1, \ldots, n$, we have $\beta_k = 0$ for $k = 1, \ldots, n$.

Since the coefficient of each point is zero, these $m + n + mn - 1$ points in $F$ are affinely independent. Therefore, $-z_{ij} \leq 0$ defines a facet of $\text{BQP}^{m,n}$. Note that these $m + n + mn - 1$ points are also in $\text{BQP}_{LP}^{m,n}$. Hence, $-z_{ij} \leq 0$ also defines a facet of $\text{BQP}_{LP}^{m,n}$.

**Lemma 4.5.** The inequalities $-x_i + z_{ij} \leq 0$ and $-y_j + z_{ij} \leq 0$ define facets of $\text{BQP}^{m,n}$ and $\text{BQP}_{LP}^{m,n}$ for $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

**Proof.** We will show the proof for $-x_i + z_{ij} \leq 0$. Similar arguments can be applied for $-y_j + z_{ij} \leq 0$. Let $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$. Define

$$F = \{(x, y, z) \in \text{BQP}^{m,n} : -x_i + z_{ij} = 0\}.$$ 

We consider $(u_h, 0_n, 0_{mn}) \in \text{BQP}^{m,n}$ for $h = 1, \ldots, m$ where $h \neq i$ and $(0_m, v_k, 0_{mn}) \in \text{BQP}^{m,n}$ for $k = 1, \ldots, n$. It is clear that these $m + n - 1$ points are in $F$. Besides, we pick $(u_h, v_k, w_{hk}) \in \text{BQP}^{m,n}$ such that $h = 1, \ldots, m$, $h \neq i$ and $k = 1, \ldots, n$. These $(m-1)n$ points also satisfy $-x_i + z_{ij} = 0$. Next, we choose $(u_i, v_j, w_{ij}) \in \text{BQP}^{m,n}$. This point is contained in $F$ as well. Then we consider $(u_i, v_j + v_k, w_{ij} + w_{ik}) \in \text{BQP}^{m,n}$ for $k = 1, \ldots, n$ and $k \neq j$. It is easy to see that these $n - 1$ points are in $F$.

Assume that

$$\sum_{h \neq i} \alpha_h (u_h, 0_n, 0_{mn}, 1) + \sum_{k=1}^{n} \beta_k (0_m, v_k, 0_{mn}, 1) + \sum_{h \neq i} \gamma_{hk} (u_h, v_k, w_{hk}, 1)$$

$$+ \gamma_{ij} (u_i, v_j, w_{ij}, 1) + \sum_{k \neq j} \delta_k (u_i, v_j + v_k, w_{ij} + w_{ik}, 1) = 0_{m+n+mn+1},$$

where $\alpha_h$ for $h = 1, \ldots, m$, $\beta_k$ for $k = 1, \ldots, n$, $\gamma_{hk}$ for $h = 1, \ldots, m$, $k = 1, \ldots, n$ and $h \neq i$, $\gamma_{ij}$ and $\delta_k$ for $k = 1, \ldots, n$ and $k \neq j$ are scalar.

i) For entries $z_{hk}$ where $hk \neq ij$, we have $\gamma_{hk} = 0$ for all $h = 1, \ldots, m$, $k = 1, \ldots, n$ and $h \neq i$, and we obtain $\delta_k = 0$ for $k = 1, \ldots, n$ and $k \neq j$. 

70
ii) For the \( z_{ij} \) entry, \( \gamma_{ij} = \gamma_{ij} + \sum_{k \neq j} \delta_k = 0 \).

iii) For entries \( x_h \) where \( h \neq i \), we obtain \( \alpha_h = \alpha_h + \sum_{k=1}^{n} \gamma_{hk} = 0 \) for \( h = 1, \ldots, m \) and \( h \neq i \).

iv) For entries \( y_k \) for \( k = 1, \ldots, n \), we have \( \beta_k = \beta_k + \sum_{h \neq i} \gamma_{hk} + \gamma_{ij} + \sum_{k \neq j} \delta_k = 0 \).

Since the coefficient of each point is zero, these \( m + n - 1 + (m-1)n + 1 + n - 1 = m + n - 1 + mn - n + 1 + n - 1 = m + n + mn - 1 \) points in \( F \) are affinely independent. Therefore, \(-x_i + z_{ij} \leq 0\) defines a facet of \( \text{BQP}^{m,n} \). Note that these \( m + n + mn - 1 \) points are also in \( \text{BQP}^{m,n}_{LP} \). Hence, \(-x_i + z_{ij} \leq 0\) also defines a facet of \( \text{BQP}^{m,n}_{LP} \). \( \square \)

**Lemma 4.6.** The inequalities \( x_i + y_j - z_{ij} \leq 1 \) define facets of \( \text{BQP}^{m,n} \) and \( \text{BQP}^{m,n}_{LP} \) for \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \).

**Proof.** Let \( i \in \{1, \ldots, m\} \) and \( j \in \{1, \ldots, n\} \). Define

\[ F = \{(x, y, z) \in \text{BQP}^{m,n} : x_i + y_j - z_{ij} = 1\}. \]

We consider \((u_i, 0_n, 0_{mn}), (0_m, v_j, 0_{mn}) \in \text{BQP}^{m,n}\). It is clear that these two points are in \( F \). Then we pick \((u_i + u_h, v_j + v_k, w_{ij} + w_{ik} + w_{hk} + w_{hk}) \in \text{BQP}^{m,n}\) such that \( h = 1, \ldots, m, h \neq i \) and \( k = 1, \ldots, n, k \neq j \). These \((m-1)(n-1)\) points satisfy \( x_i + y_j - z_{ij} = 1 \). Moreover, we pick \((u_h, v_j, w_{ij}), (u_i + u_h, v_{j'}, w_{ij'} + w_{hj'}) \in \text{BQP}^{m,n}\) for \( h = 1, \ldots, m, h \neq i \) and \( j' \neq j \). We can see that these \(2m - 2\) points are in \( F \). Next, we consider \((u_i, v_k, w_{ik}), (u_{i'}, v_j + v_k, w_{ij} + w_{ik}) \in \text{BQP}^{m,n}\) for \( k = 1, \ldots, n, k \neq j \) and \( i' \neq i \). It is easy to see that these \(2n - 2\) points are in \( F \).

Assume that
\[
\begin{align*}
\alpha_i(u_i, 0_n, 0_{mn}, 1) &+ \alpha_j(0_m, v_j, 0_{mn}, 1) \\
+ \sum_{h \neq i, k \neq j} \alpha_{hk}(u_i + u_h, v_j + v_k, w_{ij} + w_{ik} + w_{hk}, 1) \\
+ \sum_{h \neq i} \beta_h(u_h, v_j, w_{hj}, 1) + \sum_{k \neq j} \gamma_k(u_i, v_k, w_{ik}, 1) \\
+ \sum_{h \neq i} \delta_h(u_i + u_h, v_{j'}, w_{ij'} + w_{hj'}, 1) + \sum_{k \neq j} \epsilon_k(u_{i'}, v_j + v_k, w_{i'j} + w_{i'k}, 1) \\
&= 0_{n+n+mn+1},
\end{align*}
\]

where \(\alpha_i, \alpha_j; \alpha_{hk}\) for \(h = 1, \ldots, m, h \neq i\) and \(k = 1, \ldots, n, k \neq j\); \(\beta_h, \delta_h\) for \(h = 1, \ldots, m, h \neq i\); \(\gamma_k, \epsilon_k\) for \(k = 1, \ldots, n, k \neq j\) are scalar.

i) For entries \(z_{hk}\) where \(h \neq i, i'\) and \(k \neq j, j'\), we have \(\alpha_{hk} = 0\) for all \(h = 1, \ldots, m, h \neq i, i'\) and \(k = 1, \ldots, n, k \neq j, j'\).

ii) For entries \(z_{hj'}\) where \(h = 1, \ldots, m, h \neq i, i'\), we have \(\alpha_{hj'} + \delta_h = 0\). Thus, \(\alpha_{hj'} = -\delta_h\).

iii) For entries \(z_{i'k}\) where \(k = 1, \ldots, n, k \neq j, j'\), we have \(\alpha_{i'k} + \epsilon_k = 0\). Thus, \(\alpha_{i'k} = -\epsilon_k\).

iv) For \(z_{i'j'},\) we have \(\alpha_{i'j'} + \delta_{i'} + \epsilon_{j'} = 0\). Thus, \(\alpha_{i'j'} = -\delta_{i'} - \epsilon_{j'}\).

v) For entries \(z_{hj}\) where \(h = 1, \ldots, m, h \neq i, i'\), we have \(\alpha_{hj'} + \beta_h = \sum_{k \neq j} \alpha_{hk} + \beta_h = 0\). Thus, \(\beta_h = -\alpha_{hj'} = \delta_h\).

vi) For entries \(z_{ik}\) where \(k = 1, \ldots, n, k \neq j, j'\), we have \(\alpha_{i'k} + \gamma_k = \sum_{h \neq i} \alpha_{hk} + \gamma_k = 0\). Thus, \(\gamma_k = -\alpha_{i'k} = \epsilon_k\).

vii) For \(z_{i'j}\), we have \(\sum_{k \neq j} \alpha_{i'k} + \sum_{k \neq j} \epsilon_k + \beta_{i'} = 0\). From iii) and iv), we have \(\sum_{k \neq j} \alpha_{i'k} + \sum_{k \neq j} \epsilon_k = \alpha_{i'j'} + \sum_{k \neq j} (-\epsilon_k) + \epsilon_{j'} + \sum_{k \neq j} \epsilon_k = 0\). From iii) and iv), we have \(\sum_{k \neq j} \alpha_{i'k} + \sum_{k \neq j} \epsilon_k = -\delta_{i'} - \epsilon_{j'} + \epsilon_{j'} = -\delta_{i'}\). Hence, we get \(\beta_{i'} = \delta_{i'}\).
viii) For $z_{ij'}$, we have \( \sum_{h \neq i} \alpha_{hj'} + \sum_{h \neq i} \delta_h + \gamma_{j'} = 0 \). From ii) and iv), we have \( \sum_{h \neq i} \alpha_{hj'} + \sum_{h \neq i} \delta_h = \alpha_{i'j'} + \sum_{h \neq i, i'} (-\delta_h) + \delta_{i'} + \sum_{h \neq i, i'} \delta_h = -\delta_{i'} - \epsilon_{j'} + \delta_{i'} = -\epsilon_{j'} \). Hence, we get \( \gamma_{j'} = \epsilon_{j'} \).

ix) For entries $x_h$ where $h = 1, \ldots, m$, $h \neq i, i'$, we have \( \beta_h = -\delta_h + \beta_h + \delta_h = \alpha_{hj'} + \beta_h + \delta_h = \sum_{k \neq j} \alpha_{hk} + \beta_h + \delta_h = 0 \). From v), we also get \( \delta_h = \beta_h = 0 \) and from ii), it follows that \( \alpha_{hj'} = -\delta_h = 0 \).

x) For entries $y_k$ where $k = 1, \ldots, n$, $k \neq j, j'$, we have \( \gamma_k = -\epsilon_k + \gamma_k + \epsilon_k = \alpha_{i'k} + \gamma_k + \epsilon_k = \sum_{h \neq i} \alpha_{hk} + \gamma_k + \epsilon_k = 0 \). From vi), we also get \( \epsilon_k = \gamma_k = 0 \) and from iii), it follows that \( \alpha_{i'k} = -\epsilon_k = 0 \).

xi) For $x_{i'}$, we have \( 0 = \sum_{k \notin j} \alpha_{i'k} + \beta_{i'} + \delta_{i'} + \sum_{k \notin j} \epsilon_k = \alpha_{i'j'} + \beta_{i'} + \delta_{i'} + \epsilon_{j'} = -\delta_{i'} - \epsilon_{j'} + \delta_{i'} + \delta_{i'} + \epsilon_{j'} = \delta_{i'} \). Hence, \( \delta_{i'} = 0 \). From vii), we also get \( \beta_{i'} = \delta_{i'} = 0 \).

xii) For $y_{j'}$, we have \( 0 = \sum_{h \neq i} \alpha_{hj'} + \gamma_{j'} + \epsilon_{j'} + \sum_{h \neq i} \delta_h = \alpha_{i'j'} + \gamma_{i'} + \epsilon_{j'} + \delta_{i'} = -\delta_{i'} - \epsilon_{j'} + \epsilon_{i'} + \delta_{i'} + \epsilon_{j'} = \gamma_{j'} \). Hence, \( \epsilon_{i'} = 0 \). From viii), we also get \( \gamma_{j'} = \epsilon_{j'} \). From iv) and xi), we obtain \( \alpha_{i'j'} = -\delta_{i'} - \epsilon_{j'} = 0 \).

xiii) For $x_i$, we have \( \alpha_i = \alpha_i + \sum_{h \neq i, k \neq j} \alpha_{hk} + \sum_{k \neq j} \gamma_k + \sum_{h \neq i} \delta_h = 0 \), and for $y_j$, we have

\[
\alpha_j = \alpha_j + \sum_{h \neq i, k \neq j} \alpha_{hk} + \sum_{k \neq j} \epsilon_k + \sum_{h \neq i} \beta_h = 0.
\]

Since the coefficient of each point is zero, these \( 2 + (m - 1)(n - 1) + 2m - 2 + 2n - 2 = mn - m - n + 1 + 2m + 2n - 2 = m + n + mn - 1 \) points in \( F \) are affinely independent. Therefore, \( x_i + y_j - z_{ij} \leq 1 \) defines a facet of \( \text{BQP}^{m,n} \). Note that these \( m + n + mn - 1 \) points are also in \( \text{BQP}^{m,n}_{LP} \). Hence, \( x_i + y_j - z_{ij} \leq 1 \) also defines a facet of \( \text{BQP}^{m,n}_{LP} \).

These results can be summarized in the next Theorem.

**Theorem 4.7.** (i) \( \dim \text{BQP}^{m,n} = \dim \text{BQP}^{m,n}_{LP} = m + n + mn \)

(ii) The inequalities (4.1)-(4.4) define all facets of \( \text{BQP}^{m,n}_{LP} \)

(iii) The inequalities (4.1)-(4.4) define trivial facets of \( \text{BQP}^{m,n} \)
4.1.2 Properties as a Restriction of the Boolean Quadric Polytope

In Section 1.4, we have shown that BQP01 can be formulated as QP01 in higher dimension. This leads to the idea that the bipartite boolean quadric polytope can be viewed as a restriction of the boolean quadric polytope into the biclique of corresponding size.

**Definition 4.8.** [83] Let $Q$ be an upper triangular matrix of size $n \times n$. Then we can define $G = (V, E)$ as a graph on $n$ vertices spanned by the edges $e = ij$ given by nonzero coefficient $q_{ij}$ of $Q$. Note that $G$ has no loops because all diagonal entries of $Q$ are zeros. Without loss of generality, we can assume that $G$ has no isolated vertices. We denote

$$QP^G = \text{conv}\left\{(u, v) \in \mathbb{R}^{\vert V \vert + \vert E \vert} : (u, v) \text{ satisfies } (1.1) , \ldots , (1.5) \text{ for all } ij \in E\right\}$$

the Boolean Quadric Polytope associated with $G$.

From this definition, we can see $QP^n$ as $QP^G$ where $G = K_n$. Since we do not know all facets of $QP^n$, it is reasonable to consider $QP^G$ such that $G$ is a sparse graph. The properties of $QP^G$ where $G$ is an acyclic graph or a series-parallel graph were studied by Padberg in Section 5 and 6 [83].

We can see from the definition that $BQP^{m,n}$ can be viewed as a boolean quadric polytope associated with a biclique whose partite sets have order $m$ and $n$, that is $BQP^{m,n} = QP^{K_{m,n}}$. We can apply some properties of the boolean quadric polytope provided by Padberg [83] to $QP^G$ where $G$ is bipartite.

Proposition 8 [83] states that $QP^G$ has only trivial facets if and only if $G$ is acyclic. Since $K_{m,n}$ is acyclic if and only if $m = 1$ or $n = 1$, we obtain the following result for $BQP^{m,n}$.

**Proposition 4.9.** $BQP^{m,n} = BQP^{m,n}_{LP}$ if and only if $m$ or $n$ is 1.

Consequently, $BQP^{m,n}$ has only trivial facets if $m$ or $n$ is 1. Next, we introduce some definitions and notations before showing other results obtained from Padberg’s work.
Let $N$ be the index set of size $n$ ($|N| = n$). For $S \subseteq N$, we define $\tau^S = (u^S, v^S) \in QP^n$ by

$$u^S_i = \begin{cases} 1 & \text{if } i \in S, \\ 0 & \text{otherwise}, \end{cases} \quad v^S_{ij} = \begin{cases} 1 & \text{if } ij \in E(S), \\ 0 & \text{otherwise}, \end{cases}$$

where $E(S) = \{ij : i, j \in S\}$.

Similarly for $BQP^{m,n}$, we let $V = I \cup J$ be the index set of size $m + n$ where $|I| = m$ and $|J| = n$. For $S = S_1 \cup S_2 \subseteq V$ where $S_1 \subseteq I$ and $S_2 \subseteq J$, we define $\omega^S = (x^S, y^S, z^S) \in BQP^{m,n}$ by

$$x^S_i = \begin{cases} 1 & \text{if } i \in S, \\ 0 & \text{otherwise}, \end{cases} \quad y^S_j = \begin{cases} 1 & \text{if } j \in S, \\ 0 & \text{otherwise}, \end{cases} \quad z^S_{ij} = \begin{cases} 1 & \text{if } ij \in (S_1 : S_2), \\ 0 & \text{otherwise}, \end{cases}$$

where $(S_1 : S_2) = \{ij : i \in S_1, j \in S_2\}$. It is clear that there exists a one-to-one correspondence between the power set of $I \cup J$ and the set of vertices in $BQP^{m,n}$.

**Definition 4.10.** [83] For any polytopes $P$ and $Q$ such that $Q$ is contained in $P$, we say that $P$ has the Trubin property with respect to $Q$ if the set of edges of $Q$ is a subset of the set of edges of $P$.

For $S, T \subseteq V$, the face $F^S_T$ is a minimal dimension face of $QP^G_{LP}$ containing the vertices $\tau^S = (u^S, v^S)$ and $\tau^T = (u^T, v^T)$ of $QP^n$. Theorem 8 [83] is related to $F^S_T$ and Trubin property. Let $\text{co}(S)$ denote the number of connected components of the subgraph of $G$ induced by $S$. We state Theorem 8 in [83] here.

**Proposition 4.11.** [Theorem 8 from [83]]

(i) For $S, T \subseteq V$, the face $F^S_T$ of minimal dimension of $QP^G_{LP}$ containing the vertices $\tau^S$ and $\tau^T$ of $QP^n$ has only zero-one vertices and $\dim(F^S_T) = \text{co}(S \Delta T)$, where $S \Delta T$ is symmetric difference of $S$ and $T$.

(ii) $QP^n_{LP}$ has the Trubin property with respect to $QP^n$.

Substituting $G = K_{m,n}$ in the theorem gives the results for the bipartite boolean quadric polytope as in the following proposition.

**Proposition 4.12.** (i) For $S, T \subseteq V$, the face $F^S_T$ of minimal dimension of $BQP^{m,n}_{LP}$ containing the vertices $\omega^S$ and $\omega^T$ of $BQP^{m,n}$ has only zero-one vertices and $\dim(F^S_T) = \text{co}(S \Delta T)$.
co(S\Delta T).

(ii) BQP_{LP}^{m,n} has the Trubin property with respect to BQP^{m,n}.

As for QP^n, we can say that it is associated to the complete graph K_n. For any S, T \subseteq V, the subgraph of K_n induced by S\Delta T is connected, that is co(S\Delta T) = 1. Proposition 4.11 (ii) shows that dim(F_{ST}) = co(S\Delta T) = 1. It implies that for any two vertices \tau^S and \tau^T of QP^n, they are contained in a one dimensional face, which is an edge. In other words, QP^n is 2-neighbourly, that is every pair of vertices of QP^n are connected by an edge of the polytope. It was shown by Deza et al. [27] that QP^n is 3-neighbourly which means any three or fewer vertices of QP^n forms a face, but it is not 4-neighbourly.

However, this property does not hold in BQP^{m,n} since the subgraph of K_{m,n} induced by S\Delta T is not necessarily complete. We can give an example of a pair of vertices that are not connected by an edge of the polytope. Let S = \{s\} \subseteq I and T = \{t\} \subseteq I. Thus, the edge st is not in E. Moreover, since S and T are disjoint, \omega^{S}, \omega^{T}, \omega^{S\cap T} and \omega^{S\cup T} are all different vertices of BQP^{m,n}. Recall that we denote by u_i the vector in \mathbb{R}^m with all zero entries except the i^{th} which is equal to 1. Then we get

\[
\frac{1}{2}(\omega^{S} + \omega^{T}) = \frac{1}{2}((u_s, 0_n, 0_{mn}) + (u_t, 0_n, 0_{mn}))
= \frac{1}{2}((0_m, 0_n, 0_{mn}) + (u_s + u_t, 0_n, 0_{mn}))
= \frac{1}{2}(\omega^{S\cap T} + \omega^{S\cup T}).
\]

We can see that the line connecting \omega^{S} and \omega^{T} and the line connecting \omega^{S\cap T} and \omega^{S\cup T} intersect in the middle. So both lines are not edges of the polytope.

Padberg defined the support graph for each valid inequality as follows:

Definition 4.13. [83] Let \alpha = (\alpha^1, \alpha^2) be a row vector of size (n^2 + n)/2 where \alpha^1 has size n and \alpha^2 has size (n^2 - n)/2. For any valid inequalities \alpha \tau := \alpha^1 u + \alpha^2 v \leq \alpha_0 of QP^n, we denote G(\alpha) = (V_\alpha, E_\alpha) its support graph where E_\alpha = \{e \in E : \alpha^2_e \neq 0\} and V_\alpha is the subset of V spanned by E_\alpha.

Similarly, let \alpha = (a^1, a^2, a^3) be a row vector of size m + n + mn where a^1 is a row vector of size m, a^2 is a row vector of size n and a^3 is a row vector of size mn. For any valid
inequality \( a_\omega := a^1x + a^2y + a^3z \leq a_0 \) of \( \text{BQP}^{m,n} \), we can define the support graph for any valid inequality of \( \text{BQP}^{m,n} \).

**Definition 4.14.** For any valid inequality \( a_\omega \leq a_0 \) of \( \text{BQP}^{m,n} \), we denote \( G(a) = (V_a, E_a) \) its support graph where \( E_a = \{ij \in E : a_{ij}^3 \neq 0\} \) and \( V_a \) is the subset of \( V \) spanned by \( E_a \).

It is clear that \( G(a) \) is bipartite.

For a facet-defining inequality \( \alpha \tau \leq \alpha_0 \) of \( \text{QP}^n \), Lemma 1 [83] gives some properties on vector \( \alpha \) and the support graph \( G(\alpha) \). We give a version of Lemma 1 (i) [83] for \( \text{BQP}^{m,n} \).

**Proposition 4.15.** If \( a_\omega := a^1x + a^2y + a^3z \leq a_0 \) defines a facet of \( \text{BQP}^{m,n} \), there is a pair \( ij \) such that \( a_{ij}^3 \neq 0 \)

**Proof.** We prove the assertion by contrapositive. Suppose that \( a_{ij} = 0 \) for all \( ij \) where \( i = 1,\ldots,m \) and \( j = 1,\ldots,n \). Then we have \( a_\omega = a^1x + a^2y \leq a_0 \). Since \( a_\omega \leq a_0 \) is valid for \( \text{BQP}^{m,n} \) and \( 1_{m+n+mn} \in \text{BQP}^{m,n} \), we have \( a^11_m + a^21_n \leq a_0 \). We can derive \( x_i \leq 1 \) and \( y_j \leq 1 \) from inequalities (4.1)-(4.4). Hence, we obtain \( a^1_i x_i \leq a^1_i \) for \( i = 1,\ldots,m \) and \( a^2_j y_j \leq a^2_j \) for \( j = 1,\ldots,n \) from (4.1)-(4.4). Sum all \( m+n \) inequalities together. Then we get \( a_\omega = a^1x + a^2y = \sum_{i=1}^m a^1_i x_i + \sum_{j=1}^n a^2_j y_j \leq \sum_{i=1}^m a^1_i + \sum_{j=1}^n a^2_j = a^11_m + a^21_n \leq a_0 \). Thus, \( a_\omega \leq a_0 \) does not define a facet. \( \Box \)

In general, when \( H(V_H, E_H) \) is a subgraph of \( G \), we can talk about the restriction of a valid inequality \( \alpha^G \tau \leq \alpha_0^G \) for \( \text{QP}^G \) to \( \text{QP}^H \) which we can naturally obtain from \( \alpha^G \tau \leq \alpha_0^G \) by removing all components \( \alpha_i^G \) where \( i \in V \setminus V_H \) and \( \alpha_{ij}^G = 0 \) where \( ij \in E \setminus E_H \).

Similarly, the extension of a valid inequality \( \alpha^H \tau \leq \alpha_0^H \) for \( \text{QP}^H \) to \( \text{QP}^G \) is obtained from \( \alpha^H \tau \leq \alpha_0^H \) by adding components \( \alpha_i^G = 0 \) where \( i \in V \setminus V_H \) and \( \alpha_{ij}^G = 0 \) where \( ij \in E \setminus E_H \).

Proposition 7 and Corollary 2 [83] show some relationships of valid inequalities and facet-defining inequalities of two boolean quadric polytopes associated to different graph \( G \) and \( H \) where \( H \subseteq G \). If we add the condition that both polytopes must be associated to bipartite graphs, we can obtain the same relationships for the bipartite boolean quadric polytope.
Proposition 4.16. For any bipartite graph $G$ and $H$ where $H \subseteq G$,
(i) The extension of any valid inequality for $QP^H$ is valid for $QP^G$.
(ii) The restriction of any valid inequality for $QP^G$ is valid for $QP^H$ if $H$ is an induced subgraph of $G$.
(iii) The restriction of a facet-defining inequality $aw \leq a_0$ for $QP^G$ defines a facet for $QP^G(a)$.

Proposition 4.17. If $a \omega \leq a_0$ defines a facet for $QP^H$, its extension defines a facet for $QP^G$ for any bipartite $G$ containing $H$ as an induced subgraph.

Moreover, from this fact, we can easily see that if $H$ is a biclique, this proposition applies to any bipartite $G$ containing $H$. Note that $BQP^{m,n}$ is $QP^H$ where $H = K_{m,n}$. Let $M \geq m$ and $N \geq n$, then $BQP^{M,N} = QP^G$ where $G = K_{M,N}$. It follows that $K_{M,N}$ contains $H = K_{m,n}$ as an induced subgraph. Hence, a facet-defining inequality for $BQP^{m,n}$ also defines a facet for $BQP^{M,N}$.

Let $a \omega := a_1x + a_2y + a_3z \leq a_0$ be a valid inequality for $BQP^{m,n}$. The canonical extension $\hat{a} \omega \leq a_0$ of $a \omega \leq a_0$ to $BQP^{M,N}$ for $M \geq m$ and $N \geq n$ is

$$
\hat{a}_i^1 = \begin{cases} a_i & \forall i \in I, \\
0 & \forall i \in I' \setminus I \end{cases},
\hat{a}_j^2 = \begin{cases} a_j & \forall j \in J, \\
0 & \forall j \in J' \setminus J \end{cases},
\text{ and } \hat{a}_{ij}^3 = \begin{cases} a_{ij} & \forall i \in I, j \in J, \\
0 & \text{otherwise}, \end{cases}
$$

where $I = \{1, \ldots, m\}, J = \{1, \ldots, n\}, I' = \{1, \ldots, M\}$, and $J' = \{1, \ldots, N\}$. Then we obtain the following corollary.

Corollary 4.18. If $a \omega := a_1x + a_2y + a_3z \leq a_0$ defines a facet of $BQP^{m,n}$, then the canonical extension $\hat{a} \omega \leq a_0$ defines a facet of $BQP^{M,N}$ for $M \geq m$ and $N \geq n$.

Now, we can give a new family of facet-defining inequalities for $BQP^{m,n}$. The $I_{nm22}$ Bell inequalities are the inequalities in the form

$$
-x_1 - \sum_{i=1}^{m} (m-j)y_j - \sum_{2 \leq i,j \leq m, i+j=m+2} z_{ij} + \sum_{1 \leq i,j \leq m, i+j \leq m+1} z_{ij} \leq 0,
$$

for $m \geq 1$. Avis and Ito [12] gave a proof that this inequality is facet-defining for $BQP^{m,m}$.

To the best of our knowledge, the separation problem on $I_{nm22}$ Bell inequalities has not been studied. Applying Corollary 4.18, we get the following result.
Corollary 4.19. The canonical extension of an inequality (4.6) for $\text{BQP}^{m,m}$ to $\text{BQP}^{m,n}$ defines a facet for $\text{BQP}^{m,n}$.

4.1.3 Symmetry Theorem

Next, we introduce the symmetry theorem. Given two elements in $\text{BQP}^{m,n}$ and a facet containing one of them, the theorem can give another facet containing the other point via a linear transformation. Padberg [83] gave a symmetry theorem for $\text{QP}^n$ and he applied this theorem to families of facet-defining inequalities. A new family of facet-defining inequalities for $\text{QP}^01$ was obtained by applying the symmetry theorem on the family of cut inequalities.

We establish the symmetry theorem for $\text{BQP}^01$ and apply it on families of facet-defining inequalities for $\text{BQP}^01$ using the same choice of mapping as in Padberg’s work [83]. However, in this case, new families are not obtained. But it is still worthwhile to extend Padberg’s result.

For any subsets of vertices $A$ and $B$, we denote by $(A : B)$ the set of all edges $ij$ where $i \in I \cap A$ and $j \in J \cap B$. In case that $A$ or $B$ is a singleton, we allow using the single element in the set instead of the set in the notation. For example, $(A : \{v\})$ can be represented as $(A : v)$ for short. Denote $a^3(A : B)$ the sum of $a^3_{ij}$ where $ij \in (A : B)$.

For subset $M = M_1 \cup M_2$ of $V$ where $M_1 = M \cap I$ and $M_2 = M \cap J$, we define the mapping $\Psi_M(\omega)$ by

$$
\Psi_M(x_i) = \begin{cases} 
1 - x_i & \forall i \in M_1, \\
 x_i & \forall i \in I \setminus M_1,
\end{cases}
$$

$$
\Psi_M(y_j) = \begin{cases} 
1 - y_j & \forall j \in M_2, \\
y_j & \forall j \in J \setminus M_2,
\end{cases}
$$
\[ \Psi_M(z_{ij}) = \begin{cases} 
-z_{ij} + y_j & \forall ij \in (M_1 : J \setminus M_2), \\
-z_{ij} + x_i & \forall ij \in (I \setminus M_1 : M_2), \\
1 + z_{ij} - x_i - y_j & \forall ij \in (M_1 : M_2), \\
z_{ij} & \forall ij \in (I \setminus M_1 : J \setminus M_2). 
\end{cases} \]

We define the inequality \( \tilde{a} \omega = (\tilde{a}^1, \tilde{a}^2, \tilde{a}^3) \omega \leq \tilde{a}_0 \) as follows:

\[ \tilde{a}_i^1 = \begin{cases} 
a_i^1 - a^3(i : M_2) & \forall i \in M_1, \\
a_i^1 + a^3(i : M_2) & \forall i \in I \setminus M_1, 
\end{cases} \]

\[ \tilde{a}_j^2 = \begin{cases} 
a_j^2 - a^3(M_1 : j) & \forall j \in M_2, \\
a_j^2 + a^3(M_1 : j) & \forall j \in J \setminus M_2, 
\end{cases} \]

\[ \tilde{a}_{ij}^3 = \begin{cases} 
a_{ij}^3 & \forall ij \in (M_1 : M_2) \cup (I \setminus M_1 : J \setminus M_2), \\
-a_{ij}^3 & \forall ij \in (M_1 : J \setminus M_2) \cup (I \setminus M_1 : M_2), 
\end{cases} \]

\[ \tilde{a}_0 = a_0 - a^3(M_1 : M_2) - a(M). \]

**Theorem 4.20.** [Symmetry theorem] Let \( G(V, E) \) be a bipartite graph with partite sets \( I \) and \( J \). For \( R, S \subseteq V \) where \( R_1 = R \cap I \), \( R_2 = R \cap J \), \( S_1 = S \cap I \), \( S_2 = S \cap J \) and any facet \( a \omega \leq a_0 \) of \( \text{BQP}^{m,n} \) such that \( a \omega^R = a_0 \), one can give a facet defined by the inequality \( \tilde{a} \omega \leq \tilde{a}_0 \) of \( \text{BQP}^{m,n} \) such that \( \tilde{a} \omega^S = \tilde{a}_0 \) where \( \tilde{a} \) and \( \tilde{a}_0 \) are defined by the formula above for \( M_1 = R_1 \Delta S_1 \) and \( M_2 = R_2 \Delta S_2 \).

**Proof.** We will show that \( \tilde{a} \omega \leq \tilde{a}_0 \) defines a facet for \( \text{BQP}^{m,n} \). To prove the claim, we split it into three steps:

**Step 1** Observe that for any \( \omega \in \text{BQP}^{m,n} \), the inequality \( \tilde{a} \Psi_M(\omega) \leq \tilde{a}_0 \) holds. Details of this calculation are omitted.

**Step 2** \( \Psi_M(\text{BQP}^{m,n}) = \text{BQP}^{m,n} \). To show that \( \Psi_M(\text{BQP}^{m,n}) \subseteq \text{BQP}^{m,n} \), we check (4.1)-(4.4) and it is enough to work only on integer points. To show that \( \text{BQP}^{m,n} \subseteq \Psi_M(\text{BQP}^{m,n}) \), we show that for any \( \omega^R \in \text{BQP}^{m,n} \), there exists \( \omega^S \in \text{BQP}^{m,n} \) such that...
\[ \Psi_M(\omega^S) = \omega^R \] by choosing \( S = M \Delta R \). The details of these calculations are omitted.

**Step 3** Let \( N := m + n + mn \) which is the dimension of \( \text{BQP}^{m,n} \) and \( \{\omega^1, \ldots, \omega^{N-1}\} \) be an affine basis of size \( N - 1 \) of the facet defined by inequality \( a\omega \leq a_0 \). We claim that \( \{\Psi_M(\omega^1), \ldots, \Psi_M(\omega^{N-1})\} \) is a basis of size \( N - 1 \) of the face corresponding to the inequality \( \tilde{a}\omega \leq \tilde{a}_0 \). It follows that \( \tilde{a}\omega \leq \tilde{a}_0 \) defines a facet of \( \text{BQP}^{m,n} \).

To complete the last step, we assume that \( \{\omega^1, \ldots, \omega^{N-1}\} \) is an affine basis of size \( N - 1 \) of the facet defined by inequality \( a\omega \leq a_0 \). Since \( N - 1 \) is finite, it is enough to show that \( \{\Psi_M(\omega^1), \ldots, \Psi_M(\omega^{N-1})\} \) is affinely independent. Since \( \omega^1, \ldots, \omega^{N-1} \) are affinely independent, \( \det \begin{pmatrix} \omega^1 & \ldots & \omega^{N-1} \\ 1 & \ldots & 1 \end{pmatrix} = 0 \). It is easy to see that we can use row operations on \( \begin{pmatrix} \omega^1 & \ldots & \omega^{N-1} \\ 1 & \ldots & 1 \end{pmatrix} \) to obtain \( \begin{pmatrix} \Psi_M(\omega^1) & \ldots & \Psi_M(\omega^{N-1}) \\ 1 & \ldots & 1 \end{pmatrix} \). Thus, we have

\[
\det \begin{pmatrix} \Psi_M(\omega^1) & \ldots & \Psi_M(\omega^{N-1}) \\ 1 & \ldots & 1 \end{pmatrix} = \det \begin{pmatrix} \omega^1 & \ldots & \omega^{N-1} \\ 1 & \ldots & 1 \end{pmatrix} = 0.
\]

Therefore, \( \{\Psi_M(\omega^1), \ldots, \Psi_M(\omega^{N-1})\} \) is affinely independent. Hence, we have shown that \( \tilde{a}\omega \leq \tilde{a}_0 \) defines a facet for \( \text{BQP}^{m,n} \).

To prove the theorem, we set \( M = R \Delta S \). Then \( \Psi_M \) maps \( \omega^R \) to \( \omega^{R \Delta(R \Delta S)} = \omega^S \). It follows that the equality \( \tilde{a}\omega \leq \tilde{a}_0 \) corresponding to \( M = R \Delta S \) defines a facet of \( \text{BQP}^{m,n} \). From the assumption, we have \( a\omega^R = a_0 \). From the proof in Step 1, it implies that \( \omega^S = \Psi_M(\omega^R) \) also satisfies \( \tilde{a}\omega \leq \tilde{a}_0 \) at equality. Therefore, \( \tilde{a}\omega^S = \tilde{a}_0 \) as required.

For any facet-defining inequality \( a\omega \leq a_0 \), we call the inequality \( \tilde{a}\omega \leq \tilde{a}_0 \) defined as in the proof of Theorem 4.20 the symmetric pair of \( a\omega \leq a_0 \) with respect to \( M \).

Since we already get trivial facets in Section 4.1.1, we apply the theorem to trivial facets with \( M = V \) and we found that the facets in the form (4.1) give the facets in the form (4.4). On the contrary, the facets in the form (4.4) give the facets in the form (4.1). Similarly,
the facets in the form (4.2) give the facets in the form (4.3) and the facets in the form (4.3) give the facets in the form (4.2).

### 4.1.4 Partial Linear Relaxation of BQP

When we define the linear relaxation of BQP, we allow all entries to be non-integral. In this section, we consider the case when we keep the integral constraints on only one of \(x, y\) or \(z\).

Denote \(\text{BQP}^{m,n}_x\) the convex hull of points \((x,y,z) \in \mathbb{R}^{m+n+mn}\) satisfying (4.1)-(4.4) and \(x_i \in \{0,1\}\) for \(i = 1, \ldots, m\); \(\text{BQP}^{m,n}_y\) the convex hull of points \((x,y,z) \in \mathbb{R}^{m+n+mn}\) satisfying (4.1)-(4.4) and \(y_j \in \{0,1\}\) for \(j = 1, \ldots, n\) and \(\text{BQP}^{m,n}_z\) the convex hull of points \((x,y,z) \in \mathbb{R}^{m+n+mn}\) satisfying (4.1)-(4.4) and \(z_{ij} \in \{0,1\}\) for \(i = 1, \ldots, m\) and \(j = 1, \ldots, n\).

Many properties of \(\text{BQP}^{m,n}\) shown earlier in this section are also true for \(\text{BQP}^{m,n}_x\), \(\text{BQP}^{m,n}_y\) and \(\text{BQP}^{m,n}_z\) with the same proofs. These three polytopes are bounded in \([0,1]^{m+n+mn}\) and full-dimensional. Four trivial facets of \(\text{BQP}^{m,n}\) are also facets for \(\text{BQP}^{m,n}_x\), \(\text{BQP}^{m,n}_y\) and \(\text{BQP}^{m,n}_z\). In fact, we can show that \(\text{BQP}^{m,n}_x = \text{BQP}^{m,n}_y = \text{BQP}^{m,n}_z = \text{BQP}^{m,n}\).

**Theorem 4.21.** \(\text{BQP}^{m,n}_x = \text{BQP}^{m,n}_y = \text{BQP}^{m,n}_z = \text{BQP}^{m,n}\).

**Proof.** By symmetry, it suffices to prove that \(\text{BQP}^{m,n}_x = \text{BQP}^{m,n}\). We give the proof using a theorem of Hoffman and Kruskal [115]. Consider the polyhedron with the constraints

\[
\begin{align*}
x_i + y_j - z_{ij} & \leq 1, \\
-x_i & \quad + z_{ij} \leq 0, \\
-y_j + z_{ij} & \leq 0, \\
x_i, y_j, z_{ij} & \geq 0,
\end{align*}
\]
for \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \). We can fix \( x \) and treat the \( x_i \) as constants in \( \{0, 1\} \). Then we can rewrite them as

\[
\begin{align*}
y_j - z_{ij} & \leq 1 - x_i, \\
z_{ij} & \leq x_i, \\
-y_j + z_{ij} & \leq 0, \\
y_j, z_{ij} & \geq 0,
\end{align*}
\]

for \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \). The Hoffman and Kruskal Theorem \cite{HoffmanKruskal1959} states that the polyhedron \( P_x := \{(y, z) : A(y, z) \leq b, y, z \geq 0\} \) is integral for any \( b \in \mathbb{Z}^{3mn} \), that is each vertex of the polyhedron has all integral entries, if and only if the constraint matrix \( A \) is totally unimodular. The constraint matrix \( A \) is

\[
\begin{bmatrix}
I_{n \times n} & -I_{mn} \\
\vdots & \\
I_{n \times n} \\
-I_{n \times n} & I_{mn} \\
\vdots & \\
-I_{n \times n} & I_{mn} \\
0_{mn \times n} & I_{mn}
\end{bmatrix},
\]

which has dimension \( 3mn \times (mn + n) \). We claim that the constraint matrix is totally unimodular.

We can see that each entry of \( A \) is in \( \{-1, 0, 1\} \). Moreover, each row of \( A \) has at most two entries. Let \( C \) be the set of all columns of \( A \). Partition \( C \) into \((C, \emptyset)\). For each row \( i \) with exactly two entries, there is one 1 and one \(-1\). Hence,

\[
\sum_{j \in C} a_{ij} = 1 - 1 = 0 = \sum_{j \in \emptyset} a_{ij}.
\]

Thus, \( A \) is totally unimodular. It follows that \( P_x \) is integral. For any vertex \((x, y, z)\) in \( \text{BQP}^{m,n}_x \), \((y, z)\) must be a vertex in \( P_x \). Since \( P_x \) is integral, \( y \) and \( z \) have integral value. Therefore, \( \text{BQP}^{m,n}_x \) is also integral. Since \( \text{BQP}^{m,n}_x \) is a relaxation of \( \text{BQP}^{m,n}_x \), we can conclude that \( \text{BQP}^{m,n}_x = \text{BQP}^{m,n} \).

\[ \square \]
We can use this technique to show that $\text{BQP}_z^{m,n} = \text{BQP}^{m,n}$. The detail of the proof is shown in Appendix C.1.

**Theorem 4.22.** $\text{BQP}_z^{m,n} = \text{BQP}^{m,n}$.

### 4.2 Families of Valid Inequalities Obtained from Rounding Coefficients

Since the boolean quadric polytope is very well-studied, we try to obtain a family of facets and valid inequalities for $\text{BQP}^{m,n}$ from a family for $\text{QP}^{m+n}$. In this section, we provide the results from rounding the coefficients of some families of facets and valid inequalities for $\text{QP}^{m+n}$.

We have pointed out that while we can reformulate a BQP01 with $m + n + mn$ variables as a QP01 with $(m + n)(m + n + 1)/2$ variables, $\text{BQP}^{m,n}$ and $\text{QP}^{m+n}$ are not the same. We can see $\text{BQP}^{m,n}$ as a slice of $\text{QP}^{m+n}$ where the value of variables corresponding to the edges in $K_{m+n}$ but not in $K_{m,n}$ are fixed.

Consider a valid inequality $\alpha \tau = \alpha^1 u + \alpha^2 v \leq \alpha_0$ for $\text{QP}^{m+n}$. If we form a related inequality for $\text{BQP}^{m,n}$ by simply naming $u_{m+1}, \ldots, u_{m+n}$ by $y_1, \ldots, y_n$ and naming $v_{ij}$ by $z_{ij}$, many edges $(i, j)$ in $K_{m+n}$ do not appear in $K_{m,n}$. Thus, we cannot include terms corresponding to these disappearing edges in the corresponding inequality formulated for $\text{BQP}^{m,n}$. It follows that this inequality may be invalid or may be valid but not facet-defining as in the following examples.

Taking $\text{BQP}^{2,2}$ with $I = \{1, 2\}$ and $J = \{3, 4\}$. Consider $\text{QP}^4$ and a trivial facet-defining inequality $u_1 + u_2 - v_{12} \leq 1$ which is in the form (4.1). Using the process mentioned in the previous paragraph, we get $x_1 + x_2 \leq 1$ which is not valid for $\text{BQP}^{2,2}$. Now consider another trivial facet for $\text{QP}^4$, $-u_1 + v_{12} \leq 0$ which is in the form (4.2). Using the same process, we obtain $-x_1 \leq 0$ which is valid for $\text{BQP}^{2,2}$, but is not facet-defining.
Instead of throwing all of these terms away, we replace the variables with positive coefficients by zero and replace the variables with negative coefficient by one. We show that this strategy gives us valid inequalities, but they are not necessarily be facet-defining.

**Definition 4.23.** Denote \( V = \{1, \ldots, m, m+1, \ldots, m+n\} \), \( G(V, E) = K_{m,n} \) and \( G'(V, E') = K_{m+n} \).

Let \((u, v)\) be a vertex of \( \text{QP}^{m+n} \) which is a polytope corresponding to \( G' \). Then the bipartite restriction of \((u, v)\) is \((\tilde{x}, \tilde{y}, \tilde{z})\) where \( \tilde{x} \) is obtained from the first \( m \) entries of \( u \), \( \tilde{y} \) is obtained from the last \( n \) entries of \( u \) and \( \tilde{z} \) is obtained from \( v \) by discarding entries \( ij \in E' \setminus E \).

Let \((x, y, z)\) be a vertex of \( \text{BQP}^{m,n} \) which is a polytope corresponding to \( G \). Then the bipartite extension of \((x, y, z)\) is \((\hat{u}, \hat{v})\) where

\[
\hat{u}_i = \begin{cases} 
  x_i & i \in \{1, \ldots, m\}, \\
  y_i & i \in \{m + 1, \ldots, m + n\},
\end{cases}
\]

and

\[
\hat{v}_{ij} = \begin{cases} 
  z_{ij} & ij \in E, \\
  x_i y_j & ij \in E' \setminus E.
\end{cases}
\]

Let \( \alpha \tau = \alpha^1 u + \alpha^2 v \leq \alpha_0 \) be a valid inequality for \( \text{QP}^{m+n} \). Denote \( E^- = \{ij \in E' \setminus E : \alpha_{ij}^2 < 0\} \). The rounded inequality of \( \alpha \tau \leq \alpha_0 \) is defined by

\[
\tilde{\alpha} \tilde{\omega} = \tilde{\alpha}^1 \tilde{x} + \tilde{\alpha}^2 \tilde{y} + \tilde{\alpha}^3 \tilde{z} \leq \alpha_0 - \sum_{ij \in E^-} \alpha_{ij}^2, \quad (4.7)
\]

where \( \tilde{\alpha}^1 \) obtained from the first \( m \) entries of \( \alpha^1 \), \( \tilde{\alpha}^2 \) obtained from the last \( n \) entries of \( \alpha^1 \) and \( \tilde{\alpha}^3 \) obtained from \( \alpha^2 \) by discarding entries \( ij \in E' \setminus E \).

It is easy to verify that inequality (4.7) is valid for \( \text{BQP}^{m,n} \).

Clique inequalities and cut inequalities are the main families of facet-defining inequalities besides the trivial ones found by Padberg [83]. We remark that while an arbitrary clique
and cut in $K_n$ defines a facet-defining inequality for $\text{QP}^{m+n}$, inequalities obtained from simpler rounding by discarding the terms corresponding to the missing edges in complete bipartite graph are not even valid. Moreover, we do not get any facet-defining inequalities for $\text{BQP}^{m,n}$ from rounding applied to these inequalities. Besides, we give alternative modifications on these two families of inequalities, but they are not facet-defining. Some of these inequalities are tight, but some of them are not tight anywhere. All proofs for these families of valid inequalities are provided in Appendix C.2.

We give a set of notations for defining inequalities for $\text{QP}^{m+n}$ and $\text{BQP}^{m,n}$. Let $G(V,E)$ be the underlying graph of the polytope that we consider. For any subset $S \subseteq V$, denote $E(S)$ the set of edges with both endpoints in $S$. For any variable $\xi_i$ whose index corresponds to a vertex in $V$, denote $\xi(S) = \sum_{i \in S} \xi_i$. For any variable $\zeta_{ij}$ whose index corresponds to an edge in $E$, denote $\zeta(E(S)) = \sum_{ij \in E(S)} \zeta_{ij}$. Let $T$ be a subset of $V$ where $S \cap T = \emptyset$. We can extend the notation of Section 4.1.3 so that $\zeta(S : T) = \sum_{i \in S, j \in T} \zeta_{ij}$ In case that $S = \{s\}$, we can use $\zeta(s : T)$ instead of $\zeta(\{s\} : T)$ for convenience. The same routine applies for the case when $T$ is singleton.

**Rounded Clique Inequalities**

Let $G'(V,E') = K_{m+n}$. Let $S \subseteq V$ with $|S| \geq 3$ and $\alpha$ be an integer in $\{1, \ldots, |S| - 2\}$. The original clique inequality for $\text{QP}^{m+n}$ given by Padberg [83] is in the form

$$Cq(\tau) := \alpha u(S) - v(E(S)) \leq \frac{\alpha(\alpha + 1)}{2}.$$  

This inequality defines a facet in $\text{QP}^{m+n}$. We consider $G(V,E) = K_{m,n}$ and let $S_1 = S \cap I$ and $S_2 = S \cap J$.

Here we obtain a family of valid inequalities for $\text{BQP}^{m,n}$ from rounding a family of clique inequalities for $\text{QP}^{m+n}$. Let $S \subseteq V$ where $|S| \geq 3$ and $\alpha$ be an integer such that $1 \leq \alpha \leq |S| - 2$. The rounded clique inequality is

$$RCq(\omega) := \alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq \beta,$$  

(4.8)
where $\beta = \alpha(\alpha + 1)/2 + |S_1|(|S_1| - 1)/2 + |S_2|(|S_2| - 1)/2$. The inequality is valid for $\text{BQP}^{m,n}$, but it is not facet-defining.

**Theorem 4.24.** Inequality (4.8) does not define a facet for $\text{BQP}^{m,n}$. Moreover, (4.8) is tight only when $S_1$ or $S_2$ is a singleton and $\alpha = |S| - 2$.

**Biclique Inequalities**

Here we provide an alternative modification on clique inequality defined on a biclique in $K_{m,n}$. We add some constant terms and conditions to the inequality to ensure that it is valid. However, few of these new inequalities are facet-defining.

**Theorem 4.25.** For any subsets $S_1 \subseteq I$ and $S_2 \subseteq J$, let $\alpha = \max \{|S_1|, |S_2|\}$. The biclique inequality

$$BCq(\omega) := \alpha(x(S_1) + y(S_2)) - z(S_1 : S_2) \leq \alpha^2,$$

is valid for $\text{BQP}^{m,n}$. However, (4.9) does not define a facet for $\text{BQP}^{m,n}$ unless $|S_1| = |S_2| = 1$.

We can also generate inequalities by allowing two constants, $\alpha_1$ for $x(S_1)$ and $\alpha_2$ for $y(S_2)$. Using a similar strategy, the inequality in the form $\alpha_2 x(S_1) + \alpha_1 y(S_2) - z(S_1 : S_2) \leq \alpha_1 \alpha_2$ is valid only if $\alpha_i = |S_i|$ for $i = 1, 2$. The inequality becomes

$$|S_2|x(S_1) + |S_1|y(S_2) - z(S_1 : S_2) \leq |S_1||S_2|$$

which is valid for $\text{BQP}^{m,n}$. However, this is just the summation over all $i \in S_1$ and $j \in S_2$ of $x_i + y_j - z_{ij} \leq 1$ (4.1).
Rounded Cut Inequalities

Different from clique inequality, cut inequality involving two disjoint sets of vertices. Let $G'(V, E') = K_{m+n}$, $S$ and $T$ be disjoint subsets of $V$ with $|S| \geq 1$ and $|T| \geq 2$. The cut inequality for $QP^{m+n}$ given by Padberg [83] is

$$
Cut(\tau) := -u(S) - v(E(S)) + v(S : T) - v(E(T)) \leq 0,
$$
defining a facet for $QP^{m+n}$. If we consider instead bipartite graph $G(V, E) = K_{m,n}$ and we try to obtain a new valid inequality for $BQP^{m,n}$ by dropping the terms corresponding to edges in $E' \setminus E$, the inequality will no longer be valid.

Let $S$ and $T$ be disjoint subsets of $V$ with $|S| \geq 1$ and $|T| \geq 2$. Denote $S_1 = S \cap I$, $S_2 = S \cap J$, $T_1 = T \cap I$ and $T_2 = T \cap J$. The rounded version of the cut inequality is

$$
RCut(\omega) := -x(S_1) - y(S_2) - z(S_1 : S_2) + z(S_1 : T_2) + z(T_1 : S_2) - z(T_1 : T_2) \leq \beta, \tag{4.10}
$$
where $\beta = \left(\frac{|S_1|}{2}\right) + \left(\frac{|S_2|}{2}\right) + \left(\frac{|T_1|}{2}\right) + \left(\frac{|T_2|}{2}\right)$. This rounded inequality is valid for $BQP^{m,n}$. We can strengthen this inequality by changing the constant on the right hand side. The strengthened version is

$$
SRCut(\omega) := -x(S_1) - y(S_2) - z(S_1 : S_2) + z(S_1 : T_2) + z(T_1 : S_2) - z(T_1 : T_2) \leq \gamma, \tag{4.11}
$$
where $\gamma = |S_1|(|T_2| - 1) + |S_2|(|T_1| - 1)$. This inequality is valid for $BQP^{m,n}$ but it does not define a facet. Note that for $a, b \in \mathbb{Z}^+$. $a(b - 1) \leq \left(\frac{a}{2}\right) + \left(\frac{b}{2}\right)$ since $\left(\frac{a}{2}\right) + \left(\frac{b}{2}\right) = a(b - 1) = \frac{1}{2}((a - b)^2 + (a - b)) \geq 0$. Hence, inequality (4.11) is stronger than (4.10).

**Theorem 4.26.** (4.11) is valid for $BQP^{m,n}$ but is not facet-defining. Moreover, there is no vertices $\omega \in BQP^{m,n}$ where $SRCut(\omega) = \gamma$.

Alternative Cut Inequalities for $BQP^{m,n}$

To establish another family of valid inequalities for $BQP^{m,n}$, we omit the terms related to edges not in $K_{m,n}$ and modify the left hand side of the cut inequality by changing the
term \(-y(S_2)\) to \(x(T_1)\) so that the value of the left hand side at a given vertex \(\omega^R\) of \(\text{BQP}^{m,n}\) can be factorized in the similar way to Padberg’s original cut inequality. We also give an upper bound for the left hand side of the modified inequality. Adding this upper bound term to the right hand side of the modified inequality makes it valid, but it is not facet-defining.

**Theorem 4.27.** For any disjoint subsets \(S, T \subseteq V\), where \(S_1 := S \cap I\), \(S_2 := S \cap J\), \(T_1 := T \cap I\) and, \(T_2 := T \cap J\), the alternative cut inequality

\[
\text{ACut}(\omega) = \text{ACut}(x, y, z) := -x(S_1) + x(T_1) - z(S_1 : S_2) + z(S_1 : T_2) + z(T_1 : S_2) - z(T_1 : T_2) \leq \delta,
\]

(4.12)

where \(\delta = \max \{|S_1||T_2| - 1, |T_1||S_2| + 1\}\) is valid for \(\text{BQP}^{m,n}\). However, (4.12) does not define a facet for \(\text{BQP}^{m,n}\).

Besides, we can see that this \(\delta\) is the best possible since the inequality becomes tight at \(\omega^R\) where \(R = S_1 \cup T_2\) if \(\delta = s_1(t_2 - 1)\) and at \(\omega^R\) where \(R = T_1 \cup S_2\) if \(\delta = t_1(s_2 + 1)\).

### 4.3 Odd-Cycle Inequality

Odd-cycle inequality is an important family of valid inequalities for \(\text{QP}^G\). Padberg introduced the family of odd-cycle inequalities [83] for \(\text{QP}^n\). The odd-cycle inequality for \(\text{QP}^n\) is shown to be a consequence of triangle inequalities, a family of valid inequalities for \(\text{QP}^m\). Since there are no triangle inequalities for \(\text{BQP}^{m,n}\), we should check the validity and facetness of odd-cycle inequalities for \(\text{BQP}^{m,n}\). The separation problem for odd-cycle inequalities can be solved in \(O(n^3)\) time using the algorithm provided for the family of valid inequalities of the cut polytope that is closely related to odd-cycle inequalities [16].

We show that odd-cycle inequalities are valid for \(\text{QP}^G\). This fact is implicit in Padberg’s work [83], but not elaborated. Here we give the detail of the proof for completeness. Since \(\text{BQP}^{m,n}\) can be viewed as \(\text{QP}^G\) where \(G = K_{m,n}\), odd-cycle inequalities are also valid for \(\text{BQP}^{m,n}\). Moreover, we apply the assertions for odd-cycle inequalities [83] to obtain some results for \(\text{BQP}^{m,n}\).
Let $G(V,E)$ be a graph and $C(V_C,E_C)$ be any simple cycle of length at least 3 in $G$.

For any subset $M$ of $E_C$ where $|M|$ is odd, define

$$S_M = \{ w \in V_C : \exists e \neq f \in M \text{ with } e \cap f = w \},$$

and

$$S'_M = \{ w \in V_C : \exists e \neq f \in E_C \setminus M \text{ with } e \cap f = w \}.$$

An odd-cycle inequality for $QP^G$ is an inequality in the form

$$OC(u,v) := u(S_M) - u(S'_M) + v(E_C \setminus M) - v(M) \leq \left\lfloor \frac{|M|}{2} \right\rfloor,$$  \hspace{1cm} \text{(4.13)}

where $u$ is the characteristic vector in $\{0,1\}^{|V|}$ and $v$ is the characteristic vector in $\{0,1\}^{|E|}$.

**Proposition 4.28.** An odd-cycle inequality is a valid inequality for $QP^G$.

**Proof.** Let $C(V_C,E_C)$ be a cycle in $G$ and $M$ be a subset of edges in $E_C$ of odd order. A maximal path in $C$ containing only edges in $M$ is called an $M$-path, and a maximal path in $C$ containing only edges in $C \setminus M$ is called an $M'$-path. Note that $C$ is composed of the same number of $M$-paths and $M'$-paths appearing alternatingly. Let $P_{M_1}, P_{M'_1}, P_{M_2}, P_{M'_2}, \ldots, P_{M_k}, P_{M'_k}$ be the sequence of $M$-paths $P_{M_i}$ for $i = 1, \ldots, k$ and $M'$-paths $P_{M'_i}$ for $i = 1, \ldots, k$ appearing consecutively on cycle $C$. For $i = 1, \ldots, k$, denote $M_i$ and $M'_i$ the sets of edges of $P_{M_i}$ and $P_{M'_i}$, respectively. We define $m_i := |M_i|$ and $m'_i := |M'_i|$ for $i = 1, \ldots, k$. Assume that $P_{M_i} := w_{M_i,1}, w_{M_i,2}, \ldots, w_{M_i,m_i+1}$ and $P_{M'_i} := w'_{M'_i,1}, w'_{M'_i,2}, \ldots, w'_{M'_i,m'_i+1}$ for $i = 1, \ldots, k$ where $w_{M_i,m_i+1} = w'_{M'_i,1}$ for $i = 1, \ldots, k$. For any path $P$, denote by $V(P)$ the set of vertices in path $P$.

Let $(u^R,v^R) \in QP^G$. Then $OC(\omega^R) = u^R(S_M) - u^R(S'_M) + v^R(E_C \setminus M) - v^R(M)$.

We note that $u^R(S_M) - u^R(M)$ achieves its maximum value at $\sum_{i=1}^k |m_i/2| \leq |M|/2$ when $R \cap V(P_{M_i}) = \{ w_{M_i,2}, w_{M_i,4}, \ldots, w_{M_i,2|m_i/2} \}$ for all $i$. For each $i$, $v^R(M'_i) - u^R(S'_M \cap V(P_{M'_i}))$ value is at most 1. However, if $v^R(M'_i) - u^R(S'_M \cap V(P_{M'_i})) = 1$, we have $w_{M_{i+1},1} w_{M_{i+1},2}$ in $M \cap E_R$ which decreases the value of $u^R(S_M \cap V(P_{M_{i+1}})) - v^R(M_{i+1})$ by one. Hence, whenever $v^R(E_C \setminus M) - u^R(S'_M)$ increases by one, $u^R(S_M) - v^R(M)$ also decreases by at least one. Therefore, $OC(\omega^R) \leq |M|/2$. \hspace{1cm} \Box

We can write an odd-cycle inequality for BQP$^{m,n}$ as

$$x(S_M) - x(S'_M) + y(S_M) - y(S'_M) + z(E_C \setminus M) - z(M) \leq \left\lfloor \frac{|M|}{2} \right\rfloor,$$  \hspace{1cm} \text{(4.14)}

90
Note that when $G$ is bipartite, $C$ is an even cycle. Thus, we have $|C| \geq 4$. Since $\text{BQP}^{m,n} = \text{QP}^{K_{m,n}}$, we get the following corollary.

**Corollary 4.29.** An odd-cycle inequality is a valid inequality for $\text{BQP}^{m,n}$.

The following propositions are the results of applying the assertions about odd-cycle inequalities [83] with $\text{BQP}^{m,n}$.

**Proposition 4.30.** [Obtained from Theorem 9 [83]] An odd-cycle inequality (4.14) defines a facet of $\text{BQP}^{m,n}$ if and only if $C(V_C, E_C)$ is a chordless cycle of $K_{m,n}$. When $C$ is a chordless cycle of $K_{m,n}$, we also have

$$\text{BQP}^C = \text{BQP}_{L_P}^{C} \cap \left\{ \omega \in \mathbb{R}^{2|C|} : \omega \text{satisfies all inequalities (4.14)} \right\}.$$

Note that in a complete bipartite graph, $|C| = 4$ if and only if $C$ is chordless. Thus, we obtain the following corollary.

**Corollary 4.31.** The inequality (4.14) defines a facet for $\text{BQP}^{m,n}$ if and only if $|C| = 4$.

Moreover, when $m$ or $n$ is equal to 2, we can apply Proposition 4.30 to this polytope and obtain some interesting results.

**Corollary 4.32.** $\text{BQP}^{2,n}$ has $4n^2 - 4n$ nontrivial facets obtained from (4.14) where $C = C_4$ and $M$ has 1 or 3 elements.

**Proof.** $\text{BQP}^{2,n}$ is a boolean quadric polytope associated with the bipartite graph $K_{2,n}$. Since every cycle in $K_{2,n}$ is $C_4$, from Corollary 4.31, every cycle in $K_{2,n}$ gives a facet-defining inequality in the form (4.14). For each cycle, there are four choices of $M$ of order 1 and four choices of $M$ of order 3 giving different 8 facets. Since there are $n(n-1)/2$ $C_4$’s in $K_{2,n}$, $\text{BQP}^{2,n}$ has $4n^2 - 4n$ nontrivial facets defined by odd-cycle inequalities.

**Proposition 4.33.** $\text{BQP}^{2,2}$ has exactly 8 nontrivial facets. Every nontrivial facet is obtained from (4.14) where $C = C_4$ and $M$ has 1 or 3 elements. More precisely, we have

$$\text{BQP}^{2,2} = \text{BQP}_{L_P}^{2,2} \cap \left\{ \omega \in \mathbb{R}^8 : \omega \text{satisfies all inequalities (4.14)} \right\}.$$

Therefore, $\text{BQP}^{2,2}$ has exactly 24 facets.
Proof. From Proposition 4.32, there are 8 nontrivial facets defined by odd-cycle inequalities. Choose $C = K_{2,2}$. From Proposition 4.30,

$$\text{BQP}^{2,2} = \text{BQP}^C = \text{BQP}^{2,2}_{LP} \cap \{ \omega \in \mathbb{R}^8 : \omega \text{ satisfies all inequalities (4.14)} \}.$$ 

It implies that all nontrivial facets of $\text{BQP}^{2,2}$ are these 8 odd-cycle inequalities. There are $4mn = 16$ trivial inequalities. Hence, there are exactly 24 facets in total. 

The following two propositions give sufficient conditions for a nontrivial facet-defining inequality to be an odd-cycle inequality.

**Proposition 4.34.** [Obtained from Lemma 6 [83]] Let $G(I,J)$ be a bipartite graph with $|I| \leq m$ and $|J| \leq n$ and $a\omega \leq a_0$ define a nontrivial facet of $\text{BQP}^G$. If $G(a)$ contains a cycle $C$ without chord such that at most two nodes of $C$ have degree greater than two, then $a\omega \leq a_0$ is an odd-cycle inequality (4.14).

**Proposition 4.35.** [Obtained from Theorem 10 [83]] Let $G$ be a series-parallel bipartite graph and $a\omega \leq a_0$ define a nontrivial facet of $\text{BQP}^G$. Then $a\omega \leq a_0$ is an odd-cycle inequality (4.14).

When applying the symmetry theorem with $M = V$ to the family of odd-cycle inequalities which define a facet under some conditions, $OC(\omega)$ is the symmetric pair of itself. We need the following lemma to show our assertion.

**Lemma 4.36.** For any cycle $C$ in $K_{m,n}$ and an odd-order subset $M$ of the edge set of $C$, $|C| - 2|M| + |S_M| - |S'_M| = 0$.

**Proof.** We count the number of vertices in $C$. Each edge in $M$ has two endpoints. Since each vertex in $S_M$ is an endpoint of exactly two edges in $M$, it is counted twice in $2|M|$. Thus, $2|M| - |S_M|$ is the exact number of vertices incident to any edges in $M$. Combining with the other vertices in $S'_M$ that are not incident to any edges in $M$, we have $|C| = 2|M| - |S_M| + |S'_M|$ as required. 

**Corollary 4.37.** The symmetric pair of an odd-cycle inequality $OC(\omega) \leq a_0$ in the form (4.14) is $OC(\omega) \leq a_0$ itself.
Proof. Let \( C(V_C, E_C) \) be a cycle in \( K_{m,n} \) and \( M \) be an odd-order subset of the edge set \( E_C \). Let \( a_\omega \leq a_0 \) be the inequality in the form (4.14) with respect to these \( C \) and \( M \).

Note that a vertex \( i \in S_M \) is incident to 2 edges in \( M \), a vertex \( i \in S'_M \) is incident to 2 edges in \( E_C \setminus M \), and a vertex \( i \notin S_M \cup S'_M \) is incident to one edge in \( M \) and one edge in \( E_C \setminus M \). The symmetric pair of (4.14) is

\[
\tilde{a}_1^1 = \begin{cases} 
-1 - 2(-1) = 1 & i \in S_M, \\
-(-1) - 2(1) = -1 & i \in S'_M, \\
-0 - (-1 + 1) = 0 & i \notin S_M \cup S'_M,
\end{cases}
\]

\[
\tilde{a}_2^2 = \begin{cases} 
-1 - 2(-1) = 1 & j \in S_M, \\
-(-1) - 2(1) = -1 & j \in S'_M, \\
-0 - (-1 + 1) = 0 & j \notin S_M \cup S'_M,
\end{cases}
\]

\[
\tilde{a}_3^3 = \begin{cases} 
1 & ij \in E_C \setminus M, \\
-1 & ij \in M, \\
0 & \text{otherwise},
\end{cases}
\]

\[
\tilde{a}_0 = a_0 - a^3(I : J) - a(V)
= \left\lfloor \frac{|M|}{2} \right\rfloor - (|C| - |M| - |M|) - (|S_M| - |S'_M|)
= \left\lfloor \frac{|M|}{2} \right\rfloor - (|C| - 2|M| + |S_M| - |S'_M|).
\]

From Lemma 4.36, \( \tilde{a}_0 = \left\lfloor \frac{|M|}{2} \right\rfloor \). Thus, \( \tilde{a}_\omega \leq \tilde{a}_0 \) is \( x(S_M) - x(S'_M) + y(S_M) - y(S'_M) + z(E_C \setminus M) - z(M) \leq \left\lfloor \frac{|M|}{2} \right\rfloor \) (4.14).

4.4 Valid Inequalities Obtained from Relationship between a Cut Polytope and BQP\(^{m,n}\)

We have pointed out in the introduction that the boolean quadric polytope and the cut polytope are affinely equivalent [106]. Thus, we can obtain facets of boolean quadric polytope from those of the cut polytopes. Generally, the cut polytope corresponds to the
complete graph, but there are some works studying the cases of non-complete graphs.

In this section, we state the relationship between the cut polytope corresponding to a graph $G$ and the bipartite boolean quadric polytope of a graph closely related to $G$. Furthermore, we use the results for the cut polytopes to obtain new families of valid inequalities and facets for $BQP^{m,n}$. However, the facets that we see are from previously identified classes.

For any graph $G(V,E)$, let $C(G)$ denote the set of all vectors $w^S$ corresponding to $S \subseteq V$ such that

$$w_{ij} = \begin{cases} 
1 & \text{if precisely one of } i, j \text{ is in } S, \\
0 & \text{otherwise.}
\end{cases}$$

Barahona [13] denoted $PC(G)$ the convex hull of $C(G)$. Let $BQ(G)$ be the set of vectors $(u,v)$ such that $u \in \{0,1\}^{|V|}$, $v \in \{0,1\}^{|E|}$ and $v_{ij} = u_i u_j$ for all $ij \in E$. $QP^G$ defined by Padberg [83] is just the convex hull of $BQ(G)$.

Let $G + k$ denote the graph obtained from $G$ by adding a new vertex $k$ and edges from $k$ to each vertex in $G$. De Simone [106] stated that $QP^G$ is the image of $PC(G + k)$ under a linear transformation. We use his results in case that $G$ is complete bipartite.

Let $G$ be the biclique on $m + n$ vertices whose partite set $I$ contains $m$ vertices and partite set $J$ contains $n$ vertices. For any $w \in C(G + k)$, denote $g(w)$ the vector $(x, y, z)$ where $x \in \{0,1\}^m$, $y \in \{0,1\}^n$ and $z \in \{0,1\}^{mn}$ such that $x_i = w_{ik}$ for all $i \in I$, $y_j = w_{jk}$ for all $j \in J$ and $z_{ij} = x_i y_j$ for all $ij \in E$. De Simone [106] gave an observation that $g$ is a linear bijection between $C(G + k)$ and $BQ(G)$, and

$$x_i = w_{ik}, y_j = w_{jk} \text{ and } z_{ij} = \frac{1}{2}(w_{ik} + w_{jk} - w_{ij}), \quad (4.15)$$

for all $i \in I$, $j \in J$ and $ij \in E$. De Simone concluded that $g(PC(G + k)) = QP^G$ which is the same as $BQP^{m,n}$, where $g$ is the linear transformation defined by $g(w) = (x, y, z)$ such that $x, y, z$ are defined as in (4.15). This linear transformation $g$ is called the covariance mapping.
Therefore, for row vectors $a \in \mathbb{R}^m$, $b \in \mathbb{R}^n$ and $c \in \mathbb{R}^{mn}$,

$$
\begin{bmatrix}
    a & b & c
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix} = dw,
$$

for some row vector $d \in \mathbb{R}^{m+n+mn}$. Consequently, we can derive that

$$
\begin{bmatrix}
    a & b & c
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix} = \sum_{i \in I} a_i w_{ik} + \sum_{j \in J} b_j w_{jk} + \sum_{i \in I} \sum_{j \in J} c_{ij} \frac{1}{2}(w_{ik} + w_{jk} - w_{ij}).
$$

Hence, we have

$$
d_{ik} = a_i + \frac{1}{2} \sum_{j \in J} c_{ij}, \quad d_{jk} = b_j + \frac{1}{2} \sum_{i \in I} c_{ij} \quad \text{and} \quad d_{ij} = -\frac{1}{2} c_{ij},
$$

for all $i \in I$, $j \in J$ and $ij \in E$. On the other hand, we can write the coefficients of variables $x$, $y$ and $z$ in terms of the coefficients of variable $w$ as

$$
c_{ij} = -2d_{ij},
$$

$$
a_i = d_{ik} - \frac{1}{2} \sum_{j \in J} (-2d_{ij}) = \sum_{j \in J \cup \{k\}} d_{ij} \quad \text{and}
$$

$$
b_j = d_{jk} - \frac{1}{2} \sum_{i \in I} (-2d_{ij}) = \sum_{i \in I \cup \{k\}} d_{ij}.
$$

Our arguments are also true if $G$ is just a bipartite graph. We can summarize these results in the next proposition.

**Proposition 4.38.** Let $G$ be a bipartite graph and $a$, $b$, $c$ and $d$ be vectors satisfying

$$
a_i = \sum_{j \in J \cup \{k\}} d_{ij}, \quad b_j = \sum_{i \in I \cup \{k\}} d_{ij} \quad \text{and} \quad c_{ij} = -2d_{ij}.
$$

Then $dw \leq e$ is valid (facet-defining) for $PC(G+k)$ if and only if $ax + by + cz \leq e$ is valid (facet-defining) for $BQP^G$. We say that inequality $ax + by + cz \leq e$ is the covariance image of inequality $dw \leq e$.

### 4.4.1 Valid Inequalities Obtained from the Cut Polytope Corresponding to a Biclique with One Added Vertex

Most of the literature on the cut polytope considers only the polytope associated with the complete graph, but there are some results for the cut polytope corresponding to
a general graph. Barahona [16] gives five types of facets for the cut polytope corresponding to a general graph. Many kinds of operations on a facet of the cut polytope on general graphs can be found in the work of Deza and Laurent [26] and Barahona and Mahjoub [16].

The first type of facets given by Barahona and Mahjoub [16] requires that the underlying graph must contain a clique of size at least three and the inequality defines a facet only when it is an odd clique.

**Theorem 4.39.** [Theorem 2.1 [16]] Let $G = (V,E)$ be a graph and let $(W,F)$ be a complete subgraph of order $p \geq 3$ of $G$. Then

$$\sum_{ij \in F} w_{ij} \leq \left\lceil \frac{p}{2} \right\rceil \left\lfloor \frac{p}{2} \right\rfloor$$

(4.17)

is a valid inequality with respect to $P_C(G)$. Furthermore, (4.17) defines a facet of $P_C(G)$ if and only if $p$ is odd.

Note that if $G$ is bipartite, the only possible complete subgraphs of size at least three in $G + k$ are any triangles containing $k \notin V$, $u \in I$ and $v \in J$. However, the facets obtained from these are the trivial facets of $BQP^G$. In particular, one can check that for $u \in I$ and $v \in J$, the covariance image of (4.17) is $2x_u + 2y_v - 2z_{uv} \leq 2$ which is just a positive scalar multiple of (4.1).

The second one relates to a bicycle $(2c + 1)$-wheel, the joining of $K_2$ and $C_{2c+1}$. However, it is easy to see that any graph $G + k$ where $G$ is bipartite does not contain a bicycle $(2c + 1)$-wheel.

The following theorem states another type of facet for the cut polytope.

**Theorem 4.40.** [Theorem 2.4 [16]] Let $H = (W,F)$ be a complete subgraph of order $q$ where $W = \{1,2,\ldots,q\}$. Let positive integers $t_i (1 \leq i \leq q)$ satisfy $\sum_{i=1}^q t_i = 2h + 1$, $h \geq 3$ and $\sum_{t_i > 1} t_i \leq h - 1$. Set

$$a_{ij} = \begin{cases} t_i t_j, & 1 \leq i < j \leq q, \\ 0, & ij \notin F. \end{cases}$$
Then $aw \leq h(h + 1)$ defines a facet of $P_C(G)$.

However, when $G$ is a bipartite graph, there are no such subgraphs in $G + k$.

**Lemma 4.41.** Let $H$ be a subgraph of $G + k$, where $G$ is bipartite. Then $H$ does not satisfy the conditions in Theorem 4.39.

**Proof.** Suppose that there is $K_4$ with vertices $u_1, u_2, u_3, u_4$ in $G + k$. Since $u_1, u_2, u_3$ forms a triangle, we can assume without loss of generality that $k = u_1$. However, $u_2, u_3, u_4$ forms another triangle not containing $k$, a contradiction. Since any $K_n$ where $n \geq 5$ contains $K_4$ as a subgraph, $H$ can be only a clique of size 1, 2, or 3. Suppose that $H$ satisfies conditions in Theorem 4.40

**Case 1** $q = 1$. Then we have only $t_1$. It follows that $t_1 = 2h + 1, h \geq 3$ and $\sum_{t_i > 1} t_i = t_1 = 2h + 1 > h - 1$, a contradiction.

**Case 2** $q = 2$. Hence, $t_1 + t_2 = 2h + 1, h \geq 3$.

**Case 2.1** $t_1 = 1$ or $t_2 = 1$. Without loss of generality, $t_1 = 1$. Thus, $t_2 = 2h \geq 2 \cdot 3 = 6 > 1$. Therefore, $\sum_{t_i > 1} t_i = t_2 = 2h > h - 1$, a contradiction.

**Case 2.2** $t_1 > 1$ and $t_2 > 1$. It follows that $\sum_{t_i > 1} t_i = t_1 + t_2 = 2h + 1 > h - 1$, a contradiction.

**Case 3** $q = 3$. We have $t_1 + t_2 + t_3 = 2h + 1, h \geq 3$.

**Case 3.1** $t_1 = 1$, $t_2 = 1$ or $t_3 = 1$. Without loss of generality, $t_1 = 1$.

**Case 3.1.1** $t_2 = 1$ or $t_3 = 1$. Without loss of generality, $t_2 = 1$. Thus, $2h + 1 = t_1 + t_2 + t_3 = t_3 + 2$. Then $t_3 = 2h + 1 - 2 = 2h - 1 \geq 5 > 1$. Hence, $\sum_{t_i > 1} t_i = t_3 = 2h - 1 > h - 1$, a contradiction.

**Case 3.1.2** $t_2 > 1$ and $t_3 > 1$. Since $t_1 = 1$, we have $2h + 1 = t_1 + t_2 + t_3 = t_2 + t_3 + 1$. Then $\sum_{t_i > 1} t_i = t_2 + t_3 = 2h > h - 1$, a contradiction.

**Case 3.2** $t_1 > 1$, $t_2 > 1$ and $t_3 > 1$. It follows that $\sum_{t_i > 1} t_i = t_1 + t_2 + t_3 = 2h + 1 > h - 1$, a contradiction.

\qed
Similar to the boolean quadric polytope, the cut polytope also has a family of valid inequalities called odd-cycle inequalities. If the cycle relating to the inequality does not contain the added vertex $k$, we can show that the covariance image of an odd-cycle inequality for the cut polytope is an odd-cycle inequality of the bipartite boolean quadric polytope.

If the related cycle contains $k$, the family of odd-cycle inequalities for cut polytope gives another family of valid inequalities associated to a path in a biclique. We call this family of valid inequalities *odd-path inequalities*. Let $C(V_C, E_C)$ be a cycle in any graph $G$ and $M$ be a subset of odd order of $E_C$. Recall that for the odd-cycle inequality, we define

$$S_M = \{ v \in V_C : \exists e \neq f \in M \text{ with } e \cap f = v \},$$

and

$$S'_M = \{ v \in V_C : \exists e \neq f \in E_C \setminus M \text{ with } e \cap f = v \}.$$

The same definitions apply in this section.

**Theorem 4.42.** [Theorem 3.3 [16]] *The inequality*

$$\sum_{ij \in M} w_{ij} - \sum_{ij \in E_C \setminus M} w_{ij} \leq |M| - 1 \quad (4.18)$$

*is valid for $P_C(G)$ and it defines a facet if and only if $C$ is a chordless cycle.*

Note that the facet-defining condition in this theorem is the same as in Theorem 9 [83] for $Q^n$. We observe the followings by substituting

$$d_{ij} = \begin{cases} 1 & \forall i, j \in M, \\ -1 & \forall i, j \in E_C \setminus M, \\ 0 & \forall i, j \notin E_C, \end{cases}$$

in (4.16). Then we obtain

$$a_i = \begin{cases} 2 & \forall i \in S_M, \\ -2 & \forall i \in S'_M, \\ 0 & \forall i \notin V_C, \end{cases}$$
\[
\begin{align*}
b_j &= \begin{cases} 
2 & \forall j \in S_M, \\
-2 & \forall j \in S'_M, \\
0 & \forall j \notin V_C,
\end{cases} \\
c_{ij} &= \begin{cases} 
-2 & \forall ij \in M, \\
2 & \forall ij \in E_C \setminus M, \\
0 & \forall ij \notin E_C.
\end{cases}
\end{align*}
\]

and

\[
\begin{align*}
\text{It follows that } ax + by + cz & \leq |M| - 1 \text{ is the covariance image of (4.18). Since } |M| \text{ is odd, } (|M| - 1)/2 = \lfloor |M|/2 \rfloor. \text{ The resulting inequality is in the form of (4.14).}
\end{align*}
\]

**Lemma 4.43.** The covariance image of inequality (4.18) with the cycle not containing \( k \) is an odd-cycle inequality for \( \text{BQP}^G \).

For any path \( P \) in a bipartite graph \( G \), let \( p, q \) be the endpoints of this path. Add vertex \( k \) and edges \( pk, qk \) to form a cycle \( C(V_C, E_C) \). Let \( M \subseteq E_C \) be a subset of odd order. Then we define an odd-path inequality as

\[
x(S_M) - x(S'_M) + y(S_M) - y(S'_M) \\
+ z(E_P \setminus M) - z(E_P \cap M) \leq \left\lfloor \frac{|M|}{2} \right\rfloor. \tag{4.19}
\]

As with the previous lemma, we can substitute

\[
d_{ij} = \begin{cases} 
1 & \forall ij \in M, \\
-1 & \forall ij \in E_P \setminus M, \\
0 & \forall ij \notin E_P,
\end{cases}
\]

in (4.16) to get inequalities of this form.

**Theorem 4.44.** The covariance image of inequality (4.18) with the cycle containing \( k \) is an odd-path inequality for \( \text{BQP}^G \).

For \( G = K_{m,n} \), we have \( \text{BQP}^G = \text{BQP}^{m,n} \). Since \( k \) is adjacent to all vertices in \( K_{m,n} \), any cycle containing \( k \) has a chord except all triangles containing \( k \). From Theorem 4.42,
inequality (4.18) defines a facet if and only if the corresponding cycle is chordless. Thus, only triangles containing $k$ are facet-defining.

For any edge $pq$ where $p \in I$ and $q \in J$, let $C$ be the cycle formed by $pq$, added vertex $k$ and edges $pk, qk$. Since $M$ must have an odd order, $|M| = 1$ or $|M| = 3$. There are three possible ways to choose $M$ of size one: $M = \{pq\}$, $M = \{pk\}$, or $M = \{qk\}$. Then inequality in the form (4.19) corresponding to path $P = pq$ becomes $-z_{pq} \leq 0$ (4.4), $-y_q + z_{pq} \leq 0$ (4.3), or $-x_p + z_{pq} \leq 0$ (4.2), respectively. If $M$ has size 3, (4.19) becomes $x_p + y_q - z_{pq} \leq 1$ (4.1). Therefore, they define trivial facets of $BQP_{m,n}$.

For any edge $e \in E$, it is clear that the inequality

$$0 \leq w_e \leq 1 \quad (4.20)$$

is valid for the cut polytope. The last theorem in this topic states the facet-defining condition of (4.20).

**Lemma 4.45.** Inequality (4.20) defines a facet if and only if $e$ does not belong to a triangle.

However, for a biclique $G$ and an added vertex $k$, any edge $e = xy$ in $E(G + k)$ belongs to the triangle $x, y, k$. Thus, the family of valid inequalities for $BQP_{m,n}$ obtained from inequality (4.20) does not define a family of facets. If edge $e$ contains $k$, inequality (4.20) becomes $0 \leq x_i \leq 1$ if another endpoint $i$ of $e$ is in $I$ and becomes $0 \leq y_j \leq 1$ if another endpoint $j$ of $e$ is in $J$. In case that $e = ij$ does not contain $k$, inequality $w_e \leq 0$ becomes $-x_i - y_j + 2z_{ij} \leq 0$ which is the sum of facet-defining inequality in the form (4.2) and (4.3), while $w_e \leq 1$ becomes $x_i + y_j - 2z_{ij} \leq 1$ which is the sum of facet-defining inequality in the form (4.1) and (4.4).

### 4.4.2 Valid Inequalities Obtained from the Cut Polytope Corresponding to a Biclique with Two Added Vertices

Consider the formulation of BQP01 as maximum weight cut problem in Theorem 1.1. This corresponds to adding vertex $h$ in $I$, vertex $k$ in $J$, edges from $h$ to each vertex in $J$ and edges from $k$ to each vertex in $I$. 100
The bipartite maximum weight cut problem, whose polytope is the cut polytope corresponding to a biclique, is equivalent to BQP01 via another formulation of BQP01 whose solution is in \(\{-1,1\}^{m+n+mn}\). The formulation is

\[
\begin{align*}
\text{Maximize} & \quad x'^T Q' y' \\
\text{Subject to} & \quad x' \in \{-1,1\}^m, y' \in \{-1,1\}^n,
\end{align*}
\]

where \(x'_i = 2x_i - 1\) for \(i = 1, \ldots, m\) and \(y'_j = 2y_j - 1\) for \(j = 1, \ldots, n\). We will use this formulation to construct our linear transformation.

Let \(G(I, J, E)\) be a bipartite graph where \(|I| = m\) and \(|J| = n\). Let \(G'\) be a bipartite graph with partite sets \(I' = I \cup \{h\}\) and \(J' = J \cup \{k\}\) and the edge set \(E' = E \cup \{ik : i \in I\} \cup \{hj : j \in J\}\). Consider any \(w \in C(G')\). Denote \(g'(w)\) the vector \[
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix}
\]
where \(x' \in \{-1,1\}^m, y' \in \{-1,1\}^n, z' \in \{-1,1\}^{mn}\) such that \(x'_i = 2w_{ik} - 1\) for all \(i \in I\), \(y'_j = 2w_{hj} - 1\) for all \(j \in J\) and \(z'_{ij} = -2w_{ij} + 1\) for all \(ij \in E\). One can verify easily that for any \((x', y', z') \in \text{BQP}^G\), we have \(x'_iy'_j = z'_{ij}\).

Consider row vectors \(a \in \mathbb{R}^m, b \in \mathbb{R}^n, c \in \mathbb{R}^{mn}\). Then

\[
\begin{bmatrix}
a & b & c
\end{bmatrix}
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix} = \sum_{i \in I} 2a_iw_{ik} - \sum_{i \in I} a_i + \sum_{j \in J} 2b_jw_{hj} - \sum_{j \in J} b_j + \sum_{i \in I} \sum_{j \in J} (-2)c_{ij}w_{ij} + \sum_{i \in I} \sum_{j \in J} c_{ij}.
\]

Let \(e_0 = \sum_{i \in I} \sum_{j \in J} c_{ij} - 2 \sum_{i \in I} a_i - \sum_{j \in J} b_j\) and define a row vector \(d \in \mathbb{R}^{m+n+mn}\) where \(d_{ik} = 2a_i\) for all \(i \in I\), \(d_{hj} = 2b_j\) for all \(j \in J\) and \(d_{ij} = -2c_{ij}\) for all \(ij \in E\). We have

\[
\begin{bmatrix}
a & b & c
\end{bmatrix}
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix} = dw + e_0.
\] (4.21)

Thus, for any valid inequalities for \(P_C(G')\), we have the following proposition.
Proposition 4.46. Let $G$ be a bipartite graph. Then $\sum_{i \in I} \sum_{j \in J} d_{ij}w_{ij} \leq e$ is valid (facet-defining) for $P_C(G')$ if and only if $\sum_{i \in I} a_i x'_i + \sum_{j \in J} b_j y'_j + \sum_{i \in I} \sum_{j \in J} c_{ij} z'_{ij} \leq e + e_0$ is valid (facet-defining) for $BQP^G$.

We study the same families of valid inequalities for the cut polytope as in the previous linear transformation. Since $G'$ is bipartite, it does not contain any triangle. Thus, a largest clique in $G'$ is $K_2$. Hence, the family of valid inequalities in Theorem 4.39 which relate to a clique of size at least three does not appear in $P_C(G')$. Since $G'$ is a subgraph of $G' + u$ for some vertex $u$, we observed that that $G' + u$ does not contain a bicycle $(2c + 1)$-wheel, and so does $G'$. The proof of Proposition 4.41 implies that a clique that satisfies the conditions in Theorem 4.40 must have size at least four. Thus, the family of facets from this theorem also does not appear in $P_C(G')$.

Now we consider the odd-cycle inequality for the cut polytope. We substitute $d_{ij} = \begin{cases} 1 & \forall ij \in M, \\ -1 & \forall ij \in E_C \setminus M, \\ 0 & \forall ij \notin E_C, \end{cases}$ in (4.21) to obtain $(a, b, c)$. Then we substitute $x'_i = 2w_{ik} - 1$ for all $i \in I$, $y'_j = 2w_{kj} - 1$ for all $j \in J$ and $z'_{ij} = -2w_{ij} + 1$ for all $ij \in E$. Similar to the previous linear transformation, an odd-cycle inequality of $P_C(G')$ associating with a cycle not containing $h$ or $k$ corresponds to an odd-cycle inequality of $BQP^G$. While an odd-cycle inequality of $P_C(G')$ associating with a cycle containing only one of $h$ or $k$ corresponds to an odd-path inequality of $BQP_G^G$. The new family of valid inequalities obtained from this new transformation is associated with two paths in $G$, which comes from a cycle containing both $h$ and $k$ in $G'$.

Lemma 4.47. An inequality for $BQP^G$ corresponding to inequality (4.18) (under the transformation in Proposition 4.46) with the cycle not containing $h$ or $k$ is an odd-cycle inequality for $BQP^G$.

Lemma 4.48. An inequality for $BQP^G$ corresponding to inequality (4.18) (under the transformation in Proposition 4.46) with the cycle containing only one of $h$ or $k$ is an odd-path inequality for $BQP^G$. 

102
However, none of odd-path inequalities defines a facet for $\text{BQP}^{m,n}$. It can be verified by checking several small cases. In each case, the inequality is expressed as the sum of two trivial facets from (4.1)-(4.4).

**Corollary 4.49.** An odd-path inequality does not define a facet for $\text{BQP}^{m,n}$.

Note that since we do not have edge $hk$ in $G'$, when $h$ and $k$ appear together in a cycle $C$ in $G'$, $h$ must have two neighbours $q, q'$ in $J$ while $k$ must have two neighbours $p, p'$ in $I$. Thus, removing vertices $h$ and $k$ and their incident edges results two disjoint even paths. We use these two even paths in $G$ to define a new family of valid inequalities for $\text{BQP}^{m,n}$.

Let $P$ be an even-length path in a bipartite graph $G$ whose endpoints $p, p' \in I$, $Q$ be an even-length path in $G$ whose endpoints $q, q' \in J$ and $P$ and $Q$ are disjoint. Add vertices $h, k$ and edges $pk, p'k, hq, hq'$ to form a cycle $C(V_C, E_C)$. Let $M \subseteq E_C$ be a subset of odd order. Define $S_M$ and $S'_M$ as in the previous section. Let $E'$ be the set of edges in path $P$ and $Q$. Then we define an odd-2-path inequality as

\[
x(S_M) - x(S'_M) + y(S_M) - y(S'_M) + z(E' \setminus M) - z(E' \cap M) \leq \left\lfloor \frac{|M|}{2} \right\rfloor.
\]  

(4.22)

We can obtain this family of valid inequalities from (4.18) using the same routine as in Lemma 4.47 and Lemma 4.48.

**Lemma 4.50.** An inequality for $\text{BQP}^G$ corresponding to inequality (4.18) (under the transformation in Proposition 4.46) with the cycle containing both $h$ and $k$ is an odd-2-path inequality for $\text{BQP}^G$.

For the bipartite boolean quadric polytope, the corresponding graph is a biclique $G$. Let $P$ and $Q$ be disjoint even paths in $G$. Let $p, p'$ be the endpoints of $P$ and $q, q'$ be the endpoints of $Q$. Form the cycle $C(V_C, E_C)$ by adding vertices $h, k$ and edges $pk, p'k, hq, hq'$. Since $pk \in E_C$, only one of $pq$ and $pq'$ can be in $C$. The other one that does not belong to $C$ becomes a chord in $C$. Therefore, from Theorem 4.42, it does not define a facet for $\text{BQP}^{m,n}$.

The last family of valid inequalities that we discuss in this study is the inequalities in the form (4.20). However, the corresponding inequalities are not facet defining for $\text{BQP}^{m,n}$. 

103
Proposition 4.51. Let \( G(I, J, E) \) be a biclique. We define a bipartite graph \( G'(I', J', E') \) where \( I' = I \cup \{ h \} \), \( J' = J \cup \{ k \} \) and \( E' = E \cup \{ hj : j \in J \} \cup \{ ik : i \in I \} \). Let \( e \) be an edge in \( G' \). Then a valid inequality for BQP\(^{m,n}\) corresponding to an inequality in the form (4.20) does not define a facet for BQP\(^{m,n}\).

Proof. From Theorem 4.45, an inequality in the form (4.20) define a facet if and only if \( e \) does not belong to a triangle. Since \( G' \) is still bipartite, every edge \( e \in E(G') \) satisfies the condition in the theorem.

For the edges in \( E' \setminus E \), by symmetry, we can study only an edge \( uk \) where \( u \in I \). Firstly, we consider the inequality \(-w_{uk} \leq 0\). We have only \( d_{uk} = -1 \) appears in the inequality while all other coefficients are zero. Then we have \( a_u = d_{uk}/2 = -1/2 \) and \( a_i = 0 \) for all \( i \neq u \), \( b_j = 0 \) for all \( j \in J \) and \( c_{ij} = 0 \) for all \( i \in I \) and \( j \in J \). Hence, \( e_0 = \sum_{i \in I} \sum_{j \in J} c_{ij} - \sum_{i \in I} a_i - \sum_{j \in J} b_j = 0 - (-1/2) - 0 = 1/2 \). Thus, its corresponding inequality for BQP\(^{m,n}\) is \(-x_u'/2 \leq 0 + 1/2 = 1/2 \). Since \( x_u' = 2x_u - 1 \), we have \(-x_u + 1/2 \leq 1/2 \). It implies that \(-x_u \leq 0 \) which we can obtain from \(-x_u + z_{uv} \leq 0 \) (4.2) and \(-z_{uv} \leq 0 \) (4.4) for some \( v \in J \). Hence, it does not define a facet for BQP\(^{m,n}\). Next, we transform the inequality \( w_{uk} \leq 1 \) into its corresponding form for BQP\(^{m,n}\). Since we have only \( d_{uk} = 1 \) appearing in the sum, we have \( a_u = d_{uk}/2 = 1/2 \) and \( a_i = 0 \) for all \( i \neq u \), \( b_j = 0 \) for all \( j \in J \) and \( c_{ij} = 0 \) for all \( i \in I \) and \( j \in J \). Hence, \( e_0 = \sum_{i \in I} \sum_{j \in J} c_{ij} - \sum_{i \in I} a_i - \sum_{j \in J} b_j = 0 - 1/2 - 0 = -1/2 \). Thus, its corresponding inequality for BQP\(^{m,n}\) is \( x_u'/2 \leq 1 - 1/2 = 1/2 \). Since \( x_u' = 2x_u - 1 \), we have \( x_u - 1/2 \leq 1/2 \). It implies that \( x_u \leq 1 \) which we can obtain from \( x_u + y_v - z_{uv} \leq 1 \) (4.1) and \(-y_v + z_{uv} \leq 0 \) (4.3) for some \( v \in J \). Therefore, it does not define a facet for BQP\(^{m,n}\).

As for any edge \( uv \in E \), we start from the inequality \(-w_{uv} \leq 0 \). In this case, we have only \( d_{uv} = -1 \). Hence, \( c_{uv} = -d_{uv}/2 = 1/2 \). Since all other coefficients are zero, we have \( a_i = b_j = c_{ij} = 0 \) for all \( i \in I, j \in J \) and \( i j \neq uv \). Thus, \( e_0 = \sum_{i \in I} \sum_{j \in J} c_{ij} - \sum_{i \in I} a_i - \sum_{j \in J} b_j = 1/2 - 0 - 0 = 1/2 \). Then the inequality becomes \( z_{uv}'/2 \leq 0 + 1/2 = 1/2 \). Substituting \( z_{uv}' = x_u'y_v' = (2x_u - 1)(2y_v - 1) \), the left hand side is \( 2x_u y_v - y_v - x_u + 1/2 = 2z_{uv} - y_v - x_u + 1/2 \). Hence, we have \( 2z_{uv} - y_v - x_u \leq 0 \) which can be obtained from
\(-x_u + z_{uv} \leq 0\) (4.2) and \(-y_v + z_{uv} \leq 0\) (4.3). Therefore, it does not define a facet. Lastly, we consider \(w_{uv} \leq 1\). Since \(d_{uv}\) is the only one nonzero coefficient, we obtain 
\[c_{uv} = -d_{uv}/2 = -1/2\] and \(a_i = b_j = c_{ij} = 0\) for all \(i \in I, j \in J\) and \(ij \neq uv\). Thus, 
\[e_0 = \sum_{i \in I} \sum_{j \in J} c_{ij} - \sum_{i \in I} a_i - \sum_{j \in J} b_j = -1/2 - 0 - 0 = -1/2.\] Hence, the inequality becomes 
\[z'_{uv}/2 \leq 1 - 1/2 = 1/2.\] Substituting \(z'_{uv} = (2x_u - 1)(2y_v - 1)\), the left hand side is 
\[-2x_u y_v + y_v + x_u - 1/2 = -2z_{uv} + y_v + x_u - 1/2.\] Thus, we have \(-2z_{uv} + y_v + x_u \leq 1\) which can be obtained from \(x_u + y_v - z_{uv} \leq 1\) (4.1) and \(-z_{uv} \leq 0\) (4.4). Therefore, it does not define a facet.

4.5 Facets and Valid Inequalities Obtained from Triangular Elimination

We continue in the same direction as in Section 4.4 with a different construction. Avis et al. [11] proposed an operation called triangular elimination mapping a facet-defining inequality for \(P_C(K_T)\) to a facet-defining inequality for \(P_C(K_{m,n} + u)\) for some proper \(m, n\). Consequently, we can find a proper \(K_T\) to map a facet-defining inequality for \(P_C(K_T)\) to a facet-defining inequality for \(P_C(K_{m,n} + u)\) which leads to a facet-defining inequality for \(BQPM_{m,n}\).

Here we establish a strategy to transform known families of facets and valid inequalities for \(P_C(K_T)\) to families of facets and valid inequalities for \(P_C(K_{m,n} + u)\) for some proper \(m, n\). Then we obtain new families of facets and valid inequalities for \(BQPM_{m,n}\) via linear transformation discussed in Section 4.4. Our results can be applied to any \(BQPM_{M,N}\) such that \(m \leq M\) and \(n \leq N\) by using Corollary 4.18. We give examples of applying this strategy on families of valid inequalities without detailed proof in Appendix C.3.

Beside the cut polytope, many studies also concentrated on the cut cone. Since the origin is a vertex of any cut polytope, a useful fact about these two structures is pointed out by Avis et al. [10].

**Proposition 4.52.** [Proposition 1 [10]] Inequality \(dw \leq 0\) is valid (facet-defining) for the cut cone \(CUT(G)\) if and only if it is valid (facet-defining) for \(P_C(G)\).
In this section, we will exploit the result of this proposition to obtain families of valid inequalities and facets from the cut cone $\text{CUT}(K_n)$ as well as from the cut polytope $P_C(K_n)$.

Triangular elimination was introduced by Avis et al. [11]. This operation is a lifting operation for a valid inequality of the cut polytope, an operation to lift a valid inequality of a cut polytope to a valid inequality of another cut polytope in higher dimension. It is a combination of zero-lifting operation and Fourier-Motzkin elimination using the triangle inequality for the cut polytope. A generalization for general graphs is presented by Avis et al. [10].

**Definition 4.53.** [10] Let $G = (V, E)$ be a graph and $t$ be an integer. Let $F$ be a subset of edges $\{p_lq_l : l = 1, \ldots, t\}$. We say that $G'(V', E')$ is a triangular elimination of $G$ with respect to $F$ if $V' = V \cup \{r_1, \ldots, r_t\}$ where $r_1, \ldots, r_t$ are the new added vertices, $E' \cap E = E \setminus F$ and $\{p_lr_lq_l : l = 1, \ldots, t\} \subseteq E'$. Moreover, for $l = 1, \ldots, t$, the vertex $r_l$ is said to be associated with the edge $p_lq_l$.

Note that a triangular elimination for a graph $G$ with respect to $F$ is not unique since each $r_l$ can be adjacent to a vertex other than $p_l$ and $q_l$. Denote

$$\begin{align*}
\Delta^1_l &= w_{p_lq_l} - w_{p_lr_l} - w_{q_lr_l}, \\
\Delta^2_l &= w_{p_lr_l} - w_{p_lq_l} - w_{q_lr_l}, \\
\Delta^3_l &= w_{q_lr_l} - w_{p_lq_l} - w_{p_lr_l} \text{ and} \\
\Delta^4_l &= w_{p_lr_l} + w_{p_lq_l} + w_{q_lr_l} - 2.
\end{align*}$$

The triangular elimination for an inequality is defined as follows.

**Definition 4.54.** [10] Let $G' = (V', E')$ be a triangular elimination of $G = (V, E)$. Let $d$ be a row vector in $\mathbb{R}^{|E|}$, $w$ be a vector in $\mathbb{R}^{|E|}$, $d'$ be a row vector in $\mathbb{R}^{|E'|}$, $w'$ be a vector in $\mathbb{R}^{|E'|}$ and $e, e' \in \mathbb{R}$. Then inequality $d'w' \leq e'$ is a triangular elimination of $dw \leq e$ if

$$d'w' - e' = dw - e + \sum_{l=1}^t |d_{p_lq_l}| \Delta^k_l,$$

where each $k_l \in \{1, 2, 3, 4\}$.

The relationship between the original graph and its triangular elimination was shown by Avis et al. [10].
Theorem 4.55. [Theorem 3 [10]] Let $G' = (V', E')$ be a triangular elimination of $G = (V, E)$ and $d'w' \leq e'$ be a triangular elimination of $dw \leq e$. Then $d'w' \leq e'$ is valid for $P_C(G')$ if and only if $dw \leq e$ is valid for $P_C(G)$.

When the initial graph $G$ is a complete graph, Avis et al. gave sufficient conditions ensuring that if a non-triangle inequality $dw \leq e$ is facet-defining for $P_C(G)$, then its triangular elimination $d'w \leq e'$ defines a facet for $P_C(G')$, where $G'$ is a triangular elimination of $G$.

Theorem 4.56. [Theorem 5 [10]] Let $G = (V, E)$ be the complete graph of size $n$ where $n \geq 5$. Let $V = V^1 \cup \ldots \cup V^m$ be a partition of the vertex set $V$. Denote $E^k = \{p^k_{11}q^k_{11}, \ldots, p^k_{n_kq^k_{n_k}}\}$ the set of edges in the subgraph induced by $V^k$, where $n_k = |E^k| = \binom{|V^k|}{2}$, and let $F = E^1 \cup \ldots \cup E^m$.

Let $G' = (V', E')$ be a graph on $n + \sum_{k=1}^m n_k$ vertices whose vertex set $V' = V \cup R^1 \cup \ldots \cup R^m$ where $|R^k| = n_k$ and $R^k = \{r^k_{1}, \ldots, r^k_{n_k}\}$ for $k = 1, \ldots, m$. Suppose that the following conditions hold.

(i) The subgraph of $G'$ induced by $V$ is the complete $m$-partite graph $K_{|V^1|, \ldots, |V^m|}$ whose partite sets are $V^1, \ldots, V^m$.

(ii) $R^k$ is an independent set in $G'$ for $k = 1, \ldots, m$.

(iii) For $k = 1, \ldots, m$ and $l = 1, \ldots, n_k$, $p^k_{l1}q^k_{l1}, \ldots, r^k_{l1}r^k_{l1} \in E'$.

Then $G'$ is a triangular elimination of $G$ with respect to $F$ associating vertex $r^k_{l1}$ with edge $p^k_{l1}q^k_{l1}$, and a triangular elimination of any non-triangle facet-defining inequality for $P_C(G)$ defines a facet for $P_C(G')$.

To build a triangular elimination in the form $K_{m,n} + u$ for some proper $m, n$ from a complete graph $K_T$, we need to partition the vertex set into three partite sets, where one of them has exactly one element $u$. We construct the required triangular elimination as follow.

Definition 4.57. Let $G(V, E)$ be a complete graph on $T$ vertices $K_T$. Let $V^0 = \{u\}, V^1, V^2$ be a partition of $V$. Choose $F = E^1 \cup E^2$, where $E^k$ is the edge set of the subgraph induced
by $V^k$ for $k = 1, 2$. We denote $R^1 = \{r_1, \ldots, r_{n_1}\}$ where $r_l$ is the vertex associated with the edge $p_lq_l \in E^1$ and denote $R^2 = \{r_{n_1+1}, \ldots, r_{n_1+n_2}\}$ where $r_l$ is the vertex associated with the edge $p_lq_l \in E^2$.

Let $G'(V', E')$ be a graph obtained from $G$ where $V' = V \cup R^1 \cup R^2$ and

$$E' = E \setminus F \cup \{rv : r \in R^1, v \in V^0 \cup V^1 \cup R^2\} \cup \{rv : r \in R^2, v \in V^0 \cup V^2 \cup R^1\}.$$

We say that $G'(V', E')$ is a standard tripartite triangular elimination (STTE) of $G(V, E)$, and its bipartite subgraph $G^*(V^*, E^*) = G'(V', E') \setminus u$ is called a standard bipartite triangular elimination (SBTE) of $G(V, E)$.

Let $S = |V^1|$. Then we have $|V^2| = T - S - 1$. Hence, $n_1 = |R^1| = \binom{S}{2}$ and $n_2 = |R^2| = \binom{T-S-1}{2}$. We can see that $G'(V', E')$ is a spanning subgraph of $K_{m,n} + u$, whose partite sets are $V^0, V^1 \cup R^2$ of size $m = |V^1| + |R^2| = S + \binom{T-S-1}{2}$ and $V^2 \cup R^1$ of size $n = |V^2| + |R^1| = T - S - 1 + \binom{S}{2}$, while $G^*(V^*, E^*)$ is a spanning subgraph of $K_{m,n}$.

As for the labellings, note that for each $i \in V^1$, it is labelled as $p_l$ or $q_l$ for $|V^1| - 1$ times and $l \in \{1, \ldots, n_1\}$. Similarly, for any $j \in V^2$, it is labelled as $p_l$ or $q_l$ for $|V^2| - 1$ times and $l \in \{n_1 + 1, \ldots, n_1 + n_2\}$. Moreover, for a given $v \in V^1 \cup V^2$, only one of these three statements holds: $v = p_l, v = q_l, \text{ or } v \notin \{p_l, q_l\}$. Besides, for any edge $uv \in F$, we denote by $[u, v]$ the edge $uv$ where $u$ is labelled as $p_l$ and $v$ is labelled as $q_l$.

It is easy to see that our constructed $G'$ satisfies all conditions in Theorem 4.56. We will use this constructed $G'$ and $G^*$ as the triangular elimination of $G = K_T$ throughout this section. The following corollary conclude the result from our construction.

**Corollary 4.58.** If $dw \leq e$ is facet-defining for $P_C(K_T)$, then a triangular elimination $d'w' \leq e'$ with respect to the STTE $G'$ of $K_T$ is facet-defining for $P_C(G')$. Moreover, its covariance image also defines a facet for $BQP^{m,n}$ and hence by lifting theorem, its canonical extension defines a facet for $BQP^{M,N}$ for $M \geq m$ and $N \geq n$. 

108
Note that by the definition of triangular elimination, an inequality can have various forms of triangular elimination. We give the following definition to associate a triangular elimination inequality with a structure on the set of indices of all eliminated edges.

**Definition 4.59.** Consider inequality $dw \leq e$ and its triangular elimination $d'w' \leq e'$ which is in the form

$$dw + \sum_{l=1}^{n_1+n_2} |d_{pq_l}| \Delta_{kl}^l \leq e.$$ 

For $k \in \{1, 2, 3, 4\}$, define $L^k = \{l : k_l = k\}$. Denote by $L^* = \{L^1, L^2, L^3, L^4\}$ the eliminated structure of the triangular elimination inequality $d'w' \leq e'$.

On the other hand, if we consider an arbitrary $L^* = \{L^1, L^2, L^3, L^4\}$ whose elements are disjoint subsets of $L = \{1, \ldots, n_1 + n_2\}$ and the union of all elements is $L$, we can see that a triangular elimination whose eliminated structure is $L^*$ may not exist. The following definition describes a structure ensuring that any triangular elimination with respect to STTE of a given inequality defined from this structure is well-defined, that is the terms associating with all eliminated edges are disappeared.

**Definition 4.60.** Let $G^*$ be SBTE of $G$ and $L = \{1, \ldots, n_1 + n_2\}$. Denote by $L^*$ the family $\{L^1, L^2, L^3, L^4\}$ where $L^1, L^2, L^3, L^4$ are disjoint subsets of $L$ such that $L^1 \cup L^2 \cup L^3 \cup L^4 = L$, and each of them can be empty. We write $L^{h,k} := L^h \cup L^k$ for short. For each $l \in L^k$, set $k_l = k$ for $k = 1, \ldots, 4$. The structure $(G^*, L^*)$ is acceptable for inequality $dw \leq e$ if there exists a triangular elimination $d'w' \leq e'$ of $dw \leq e$ whose eliminated structure is $L^*$, that is, for all edge $pq_l \in F$, the coefficient of $w_{pq_l}$ in $dw + \sum_{l=1}^{n_1+n_2} |d_{pq_l}| \Delta_{kl}^l$ is zero.

We demonstrate the application of triangular elimination on some families of valid inequalities and facets for the cut polytope in Appendix C.3 without giving any proof detail.
Chapter 5

Conclusion

In this thesis, we investigated the problem BQP01 from an algorithmic and polyhedral point of view. Various applications and equivalent formulas of the problem were discussed. In particular, we showed that BQP01 is equivalent to MaxCut problem on a bipartite graph. The BQP01 model generalizes the QP01 model studied extensively in literature.

We showed that when the rank of the cost matrix $Q$ is fixed, BQP01 is polynomially solvable. A corresponding result for QP01 is available but with additional assumption that $Q$ is positive semidefinite [6, 28, 35, 58]. Without the assumption of positive semidefiniteness, QP01 is NP-hard even if rank of $Q$ is one. This may raise concerns that our general problem BQP01 does not need positive semidefiniteness to establish polynomial solvability for fixed rank cost matrix. It may be noted that when QP01 reduced to BQP01, the reduction may alter the rank of the corresponding cost matrix and hence the fixed rank assumption may not hold. However, if the cost matrix of QP01 is also positive semidefinite, we have a better reduction to BQP01 that does not alter the rank of the cost matrix and our results for BQP01 can also be used for QP01. This gives an alternative algorithm for solving QP01 with fixed rank cost matrix. Thus, our result is a proper generalization of corresponding results for QP01. When rank of $Q$ is one, we presented an $O(n \log n)$ algorithm. We also provided some additional polynomially solvable special cases of BQP01.
We then obtained a closed-form formula to compute the average of all objective function values of BQP01. As a direct consequence, this average value can be identified in $O(mn)$ time. Different algorithms that we presented guarantee solutions which are no worse than average. The domination number of such algorithms are shown to be at least $2^{m+n-2}$ and showed that this bound is tight. We also presented algorithms with improved domination numbers. In addition to lower bounds, we have provided an upper bound on the domination number for any polynomial time algorithm, unless $P=NP$. Many algorithms that works well in practice are shown to be get trapped at solutions worse than average, establishing that average based analysis is non trivial and is a useful tool in measuring the quality of a heuristic solution. Unlike average, we do not have an efficient way to compute the median of the objective function value of BQP01. In fact, we showed that computing the median value is NP-hard.

Finally, we investigated the polytope $BQP_{m+n}$ arising from linearization of an integer programming formulation of BQP01. Various strategies to obtain new classes of valid inequalities and facet-defining inequalities are presented. In particular, we obtained the family of trivial facets and family of odd-cycle inequalities directly from the corresponding families for $QP_{m+n}$. Rounding coefficients technique has been applied on families of clique inequalities and cut inequalities for $QP_{m+n}$. The $I_{mm22}$ Bell inequalities for $BQP_{m,n}$ is a result from applying a lifting theorem on correlation polytope. Then we established the relationship between $BQP_{m,n}$ and cut polytope to obtain valid inequalities and facets from those known for cut polytope, using covariance mapping and triangular elimination.

There are many topics related to BQP01 that needs further study. Note that QP01 has been studied extensively using heuristic and exact algorithms. We believe that results discussed in this thesis could motivate studies along this direction. In particular, efficient separation algorithm from our inequalities, powerful branch and cut algorithm, new valid inequalities and facet defining inequalities and effective metaheuristic algorithm are all topics for further research.
Another possible line of research is to unify the bottleneck version of BQP01 with BQP01 itself, yielding a generalization. Note that the bottleneck version of BQP01 is polynomially solvable [96]. Thus, it is meaningful to study various generalizations along the lines of [93, 95, 99] which studied linear and bottleneck problems, to adapt to the case of BQP01, especially if meaningful applications warrant the study of such generalizations.
Bibliography


117


120
Appendix A

Special Case on Rank One Matrix

Here we give the full detail on our algorithm for BQP01 with cost matrix of rank one. Recall that we start from considering the bilinear equivalent version:

UBLP(1): Maximize $axby + cx + dy$
Subject to: $x \in [0,1]^m$, $y \in [0,1]^n$,

where $a = (a_1, \ldots, a_m), c = (c_1, \ldots, c_m) \in R^m$ and $b = (b_1, \ldots, b_n), d = (d_1, d_2, \ldots, d_n) \in R^n$. Besides, we have defined $A^- = \{i: a_i < 0\}$, $A^+ = \{i: a_i > 0\}$, $\lambda = \sum_{i \in A^-} a_i$ and $\bar{\lambda} = \sum_{i \in A^+} a_i$, where summation over the empty set is taken as zero. We also consider the Parametric Continuous Knapsack Problem (PKP($\lambda$)) given below.

PKP($\lambda$): Maximize $cx$
Subject to $ax = \lambda$
$x \in [0,1]^m$, and $\underline{\lambda} \leq \lambda \leq \bar{\lambda}$.

Recall that $h_1(\lambda)$ is the optimal objective function value of PKP($\lambda$) for a given $\lambda$ and for $\underline{\lambda} \leq \lambda \leq \bar{\lambda}$, $h_1(\lambda)$ is a piecewise linear concave function [30]. Let $\underline{\lambda} = \lambda_0 < \lambda_1 < \ldots < \lambda_t = \bar{\lambda}$ be the breakpoints of $h_1(\lambda)$ and $x^k$ be an optimal basic feasible solution of PKP($\lambda$) for $\lambda \in [\lambda_{k-1}, \lambda_k], 1 \leq k \leq t$. Then $x^k$ will be an optimal basic feasible solution to PKP($\lambda^k$).

From Lemma 2.4, the number of breakpoints of $h_1(\lambda)$ is at most $2m$. We now observe that the number of breakpoints of $h_1(\lambda)$ cannot be more than $m + 1$. Let $B = \{b\}$ be a given basis. Consider the left hand side of the dual feasibility conditions (2.1) and (2.2) which is

$$C^B B^{-1} A^i - c_i = \frac{c_b}{a_b} a_i - c_i = a_i \left(\frac{c_b}{a_b} - \frac{c_i}{a_i}\right).$$
We can see that the ratio $c_i/a_i$ of each entry $i$ plays an important role in the dual feasible conditions. Besides, entry $b$ where $a_b = 0$ is not qualified to be a basis since $B = a_b = 0$ is invertible. Thus, we let $T = \left\{ \frac{c_i}{a_i} : i = 1, \ldots, m, \ a_i \neq 0 \right\}$ and consider the nonincreasing arrangement $\frac{c_{\rho(1)}}{a_{\rho(1)}} \geq \cdots \geq \frac{c_{\rho(r)}}{a_{\rho(r)}}$ of all elements in $T$. Denote by $B_k$ the basis containing entry $\rho(k)$. Now apply the parametric simplex algorithm by moving from the basis $B_1 = \{\rho(1)\}$ to $B_r = \{\rho(r)\}$. For a given basis $B_k$, from (2.4), the two values of $\kappa$ which are the candidates to be breakpoints corresponding to the characteristic interval for this basis can be obtained from solving $B_{-1}^1 \kappa - B_{-1}^2 A^U \upsilon = 0$ or $B_{-1}^1 \kappa - B_{-1}^2 A^U \upsilon = 1$ which lead to $\kappa = A^U \upsilon$ or $\kappa = B + A^U \upsilon$. For $k = 1, \ldots, r$, denote $\kappa^L_k$ and $\kappa^U_k$ the minimum and maximum between these two values. We show that there are $r + 1$ different values obtained.

**Proposition A.1.** $|\{\kappa^L_1, \kappa^U_1, \ldots, \kappa^L_r, \kappa^U_r\}| = r + 1.$

**Proof.** Let $U(k) = A^U \upsilon$ for basis $B_k$. We first claim that for $B_k, k = 1, \ldots, r,$

\[
U(k) = \begin{cases} 
\lambda & \text{if } a_{\rho(k)} > 0 \text{ and } k = 1, \\
\lambda + \sum_{l=1}^{k-1} |a_{\rho(l)}| & \text{if } a_{\rho(k)} > 0 \text{ and } k > 1, \\
\lambda + \sum_{l=1}^{k} |a_{\rho(l)}| & \text{if } a_{\rho(k)} < 0.
\end{cases}
\]

Since $B = a_{\rho(k)}$, we get

\[
\kappa^L_k = U(k) = \begin{cases} 
\lambda & \text{if } k = 1, \\
\lambda + \sum_{l=1}^{k-1} |a_{\rho(l)}| & \text{if } k > 1,
\end{cases}
\]

and

\[
\kappa^U_k = U(k) + B = \lambda + \sum_{l=1}^{k} |a_{\rho(l)}|,
\]

when $a_{\rho(k)} > 0$, while

\[
\kappa^U_k = U(k) = \lambda + \sum_{l=1}^{k} |a_{\rho(l)}| \text{ and}
\]

\[
\kappa^L_k = U(k) + B = \begin{cases} 
\lambda & \text{if } k = 1, \\
\lambda + \sum_{l=1}^{k-1} |a_{\rho(l)}| & \text{if } k > 1.
\end{cases}
\]

when $a_{\rho(k)} < 0$. Thus, we can see that for $k = 1, \ldots, r - 1, \kappa^U_k = \kappa^L_{k+1}$ and the result follows.

Now it remains to prove the claim which can be done by induction.
**Basis Step** Let we consider \( U(1) \). Then \( \frac{c_b}{a_b} - \frac{c_i}{a_i} \geq 0 \) for all \( i \). Thus, \( a_i \left( \frac{c_b}{a_b} - \frac{c_i}{a_i} \right) \geq 0 \) if \( a_i > 0 \) and \( a_i \left( \frac{c_b}{a_b} - \frac{c_i}{a_i} \right) \leq 0 \) if \( a_i < 0 \). From (2.1) and (2.2), we can construct a dual feasible solution by putting entries in \( A^+ \) in \( U \) and entries in \( A^- \) in \( U \). Hence, if \( a_{\rho(1)} > 0 \), then \( U(1) = \lambda \) while if \( a_{\rho(1)} < 0 \), then \( U(1) = \sum_{i \in A^-} a_i - a_{\rho(1)} = \lambda + |a_{\rho(1)}| \).

**Inductive Step** Assume that the claim is true for \( U(k) \). Consider \( U(k+1) \). Note that now, \( \frac{c_b}{a_b} \) move from \( c_{\rho(k)}/a_{\rho(k)} \) to \( c_{\rho(k+1)}/a_{\rho(k+1)} \) which is equal to \( c_{\rho(k)}/a_{\rho(k)} \) or is the next smaller. Thus, \( \frac{c_{\rho(k+1)}}{a_{\rho(k+1)}} - \frac{c_{\rho(k)}}{a_{\rho(k)}} \leq 0 \) and the change in the corresponding dual feasible solution only occurs on entries \( \rho(k) \) and \( \rho(k+1) \).

**Case 1** \( a_{\rho(k)} > 0 \) and \( a_{\rho(k+1)} > 0 \). In this case, \( \rho(k) \) leaves basis and go to \( U \). Hence, \( U(k+1) = U(k) + a_{\rho(k)} = \lambda + \sum_{l=1}^{k} |a_{\rho(l)}| \).

**Case 2** \( a_{\rho(k)} > 0 \) and \( a_{\rho(k+1)} < 0 \). In this case, \( \rho(k) \) also leaves basis and go to \( U \) while \( \rho(k+1) \) leaves \( U \) and enters the new basis. Then \( U(k+1) = U(k) + a_{\rho(k)} - a_{\rho(k+1)} = \lambda + \sum_{l=1}^{k+1} |a_{\rho(l)}| \).

**Case 3** \( a_{\rho(k)} < 0 \) and \( a_{\rho(k+1)} > 0 \). Then \( \rho(k) \) go to \( L \) after leaving basis. Thus, \( U(k+1) = U(k) = \lambda + \sum_{l=1}^{k} |a_{\rho(l)}| \).

**Case 4** \( a_{\rho(k)} < 0 \) and \( a_{\rho(k+1)} < 0 \). The only change in \( U \) is the leave of \( \rho(k+1) \). Thus, \( U(k+1) = U(k) - a_{\rho(k+1)} = \lambda + \sum_{l=1}^{k+1} |a_{\rho(l)}| \).

However, it is not necessary that all of these \( r + 1 \) values are breakpoints due to the degeneracy of bases. To ensure that (2.1) and (2.2) hold as strict inequalities, we consider a descending arrangement \( \frac{c_{\pi(1)}}{a_{\pi(1)}} > \cdots > \frac{c_{\pi(p)}}{a_{\pi(p)}} \) of all distinct elements of \( T \). Denote by \( T(k) = \{ i : \frac{c_{\pi(k)}}{a_{\pi(k)}} = \frac{c_i}{a_i} \} \). Then the breakpoints of \( h_1(\lambda) \) are given by

\[
\lambda_0 = \lambda \quad \text{and} \quad \lambda_k = \lambda_{k-1} + \sum_{i \in T(k)} |a_i| \quad \text{for} \quad k = 1, \ldots, p.
\]

Now we consider an optimal solution to PKP(\( \lambda \)) at \( \lambda = \lambda_k \) for \( k = 0, 1, \ldots, p \). It is clear that \( x^0 \) corresponding to \( \lambda_0 = \lambda \) must satisfy \( x^0_i = 1 \) for all \( i \) where \( a_i < 0 \). Moreover, setting entry \( x^0_i \) with \( a_i = 0 \) to be 1 does not affect the feasibility of \( x^0 \). Thus, we assign \( x^0 \)
to be
\[ x_i^0 = \begin{cases} 
1 & \text{if } a_i = 0 \text{ and } c_i > 0 \text{ or } a_i < 0, \\
0 & \text{otherwise.}
\end{cases} \]

Here we get \( \mathbb{B}_0 = \{\pi(1)\} \), \( L_0 = \{i : x_i^0 = 0, i \neq \pi(1)\} \) and \( U_0 = \{i : x_i^0 = 1, i \neq \pi(1)\} \). For \( k = 1, \ldots, p \), we can identify \( x^k \) recursively as
\[ x_i^k = \begin{cases} 
  x_i^{k-1} & \text{if } i \notin T(k), \\
  1 & \text{if } i \in T(k) \text{ and } a_i > 0, \\
  0 & \text{otherwise.}
\end{cases} \]

Then \( \mathbb{B}_k = \{\pi(k)\} \), \( \mathbb{L}_k = \{i : x_i^k = 0, i \neq \pi(k)\} \) and \( \mathbb{U}_k = \{i : x_i^k = 1, i \neq \pi(k)\} \).

**Theorem A.2.** The basis structure \((\mathbb{B}_k, \mathbb{L}_k, \mathbb{U}_k)\) satisfies the reduced cost optimality conditions (2.1) and (2.2) for \( k = 0, 1, \ldots p \).

**Proof.** We establish the result using mathematical induction.

**Basis step** For \( k = 0 \), \( \mathbb{B}_0 = \{\pi(1)\} \). So we have \( \frac{c_{\pi(1)}}{a_{\pi(1)}} - \frac{c_i}{a_i} > 0 \) for all \( i \notin T(1) \). Thus, entries with \( a_i > 0 \) in \( L_0 \setminus T(1) \) satisfy (2.1) and those with \( a_i < 0 \) in \( U_0 \setminus T(1) \) satisfy (2.2). As for \( i \) where \( a_i = 0 \) or \( i \in T(1) \), we have \( a_i \left( \frac{c_{\pi(1)}}{a_{\pi(1)}} - \frac{c_i}{a_i} \right) = 0 \). Hence, it does not matter whether it is in \( L_0 \) or \( U_0 \) since it satisfies both (2.1) and (2.2).

**Inductive step** Assume that \((\mathbb{B}_k, \mathbb{L}_k, \mathbb{U}_k)\) satisfies conditions (2.1) and (2.2). We consider the dual feasibility of \( x^{k+1} \) corresponding to \((\mathbb{B}_{k+1}, \mathbb{L}_{k+1}, \mathbb{U}_{k+1})\). Note that if \( a_i = 0 \) or \( i \in T(k + 1) \), it satisfies both (2.1) and (2.2). Hence, we can explore only entries \( i \in T \setminus T(k + 1) \) of \( x^{k+1} \).

**Case 1** \( i \in \bigcup_{j=1}^{k-1} T(j) \). Then
\[ \frac{c_{\pi(k+1)}}{a_{\pi(k+1)}} < \frac{c_{\pi(k)}}{a_{\pi(k)}} < \frac{c_i}{a_i}. \]
Since \( i \notin T(k + 1) \), we have \( x_i^{k+1} = x_i^k \). If \( x_i^{k+1} = x_i^k = 1 \), then \( i \in \mathbb{U}_k \) which implies that \( a_i \left( \frac{c_{\pi(k)}}{a_{\pi(k)}} - \frac{c_i}{a_i} \right) \leq 0 \). Since \( \frac{c_{\pi(k)}}{a_{\pi(k)}} - \frac{c_i}{a_i} < 0 \), we know that \( a_i > 0 \). Since \( \frac{c_{\pi(k+1)}}{a_{\pi(k+1)}} - \frac{c_i}{a_i} < 0 \), it follows that \( x_i^{k+1} \) satisfies (2.2). If \( x_i^{k+1} = x_i^k = 0 \), then \( i \in \mathbb{L}_k \) which implies that \( a_i \left( \frac{c_{\pi(k)}}{a_{\pi(k)}} - \frac{c_i}{a_i} \right) \geq 0 \). Since \( \frac{c_{\pi(k)}}{a_{\pi(k)}} - \frac{c_i}{a_i} < 0 \), we know that \( a_i < 0 \). Since \( \frac{c_{\pi(k+1)}}{a_{\pi(k+1)}} - \frac{c_i}{a_i} < 0 \), it follows that \( x_i^{k+1} \) satisfies (2.1).
Note that we can obtain $\lambda_i$ if $a_i > 0$ and $x_i^k = 0$ otherwise. Since $i \notin T(k + 1)$, we have $x_i^{k+1} = x_i^k$. If $x_i^{k+1} = x_i^k = 1$, then $a_i > 0$. Since $c_i/a_i > c_{\pi(k)}/a_{\pi(k)}$ and $x_i^k = 1$ if $a_i > 0$ and $x_i^k = 0$ otherwise, $\lambda_i < 0$, it follows that $x_i^{k+1}$ satisfies (2.2). If $x_i^{k+1} = x_i^k = 0$, then $a_i < 0$. Since $c_i/a_i < c_{\pi(k)}/a_{\pi(k)}$, it follows that $x_i^{k+1}$ satisfies (2.1).

Case 3 $i \in \bigcup_{j=k+2}^r T(j)$. Then $c_i/a_i < c_{\pi(k+1)}/a_{\pi(k+1)} < c_{\pi(k)}/a_{\pi(k)}$. Since $i \notin T(k + 1)$, we have $x_i^{k+1} = x_i^k$. If $x_i^{k+1} = x_i^k = 1$, then $a_i \left( c_{\pi(k)}/a_{\pi(k)} - c_i/a_i \right) \leq 0$. Since $c_{\pi(k)}/a_{\pi(k)} - c_i/a_i > 0$, we know that $a_i < 0$. Since $c_{\pi(k+1)}/a_{\pi(k+1)} - c_i/a_i > 0$, it follows that $x_i^{k+1}$ satisfies (2.2). If $x_i^{k+1} = x_i^k = 0$, then $a_i \left( c_{\pi(k)}/a_{\pi(k)} - c_i/a_i \right) \geq 0$. Since $c_{\pi(k)}/a_{\pi(k)} - c_i/a_i > 0$, we know that $a_i > 0$. Since $c_{\pi(k+1)}/a_{\pi(k+1)} - c_i/a_i > 0$, it follows that $x_i^{k+1}$ satisfies (2.1).

From the formula for $x^k$, we know that

$$h_1(\lambda_k) - h_1(\lambda_{k-1}) = \sum_{x_i^{k-1} = 1} \sum_{x_i^k = 0} \sum_{c_i} - \sum_{x_i^{k-1} = 1} \sum_{x_i^k = 0} c_i = \sum_{i \in T(k)} c_i - \sum_{i \notin T(k)} c_i.$$

Therefore, the corresponding objective function values are

$$h_1(\lambda_0) = \sum_{i=1}^m c_i x_i^0 \text{ and } h_1(\lambda_k) = h_1(\lambda_{k-1}) + \sum_{i \in T(k)} c_i - \sum_{i \notin T(k)} c_i.$$

Thus, given $\lambda_{k-1}, x^{k-1}$ and $h(\lambda_{k-1}), \lambda_k, x^k$ and $h(\lambda_k)$ can be identified in $O(|T(k)|)$ time. Note that we can obtain $\lambda_0, x^0$ and $h_1(\lambda_0)$ in $O(m)$ time. Since all $T(k)$ are disjoint, the complexity for generating breakpoints $\lambda_1, \ldots, \lambda_p$, corresponding solutions and corresponding objective function values is $O(|T(1)| + \ldots + O(T(k)) = O(m)$. Thus, the overall complexity for computing these values is $O(m)$.

Let $y^k$ be an optimal solution to UBLP(1) when $x$ is restricted to $x^k$. From Lemma 2.6, the vector $y^k$ as well as its corresponding objective function value can be obtained from $x^k$ in $O(n)$ time by the formula

$$y^k_j = \begin{cases} 1 & \text{if } d_j + ax^k b_j > 0, \\ 0 & \text{otherwise} \end{cases}.$$

By Theorem 2.7, there exists an optimal solution to UBLP(1) amongst the solutions $(x^k, y^k)$ where $k = 0, 1, \ldots, p$. Thus, we can obtain an optimal solution from these $p + 1 \leq m + 1$
ordered pairs. Therefore, this step takes $O(mn)$ time once all $x^k$ are given.

It takes $O(m \log m)$ time to identify the descending order of $c_i/a_i$. All basic feasible solution $x^k$ and its corresponding objective function values can be obtained in $O(m)$ times. Obtaining all $y^k$ and identify an optimal solution can be done in $O(mn)$ time which dominates the complexity of all previous processes. We now show that all $(x^k, y^k)$ and its corresponding objective function values can be identified in $O(n \log n)$ time.

Consider the *Parametric Unconstrained Linear Optimization Problem*

$$\text{ULP}(\mu): \quad \text{Maximize } dy + \mu by$$

$$\text{Subject to: } y \in [0, 1]^n \text{ and } A \leq \mu \leq \bar{\lambda}.$$  

Let $h_2(\mu)$ be the optimal objective function value of ULP($\mu$). Then $h_2(\mu)$ is a piecewise linear convex function.

Let $\underline{\lambda} = \mu_0 < \mu_1 < \ldots < \mu_t = \bar{\lambda}$ be the breakpoints of $h_2(\lambda)$. Let $S^+_t = \{ j : d_j + b_j \mu_t \geq 0 \}$, and $S^-_t = \{ j : d_j + b_j \mu_t < 0 \}$. Also let $B^+ = \{ j : b_j > 0 \}$, $B^* = \{ j : b_j = 0 \}$ and $B^- = \{ j : b_j < 0 \}$. Then an optimal solution $y^l$ of ULP($\mu_l$) is given by

$$y^l_j = \begin{cases} 
1 & \text{if } j \in S^+_l, \\
0 & \text{otherwise}. 
\end{cases}$$

Suppose $\mu$ is increased from $\mu = \mu_l$ to $\mu_l + \epsilon$ for some small $\epsilon > 0$. Then $y^l$ continues to be optimal for PLP at $\mu = \mu_l + \epsilon$ if

$$d_j + (\mu_l + \epsilon)b_j \geq 0 \text{ for all } j \in S^+_l \text{ and }$$

$$d_j + (\mu_l + \epsilon)b_j \leq 0 \text{ for all } j \in S^-_l.$$  

Hence,

$$\epsilon \leq -\frac{d_j}{b_j} - \mu_l \text{ if } j \in S^+_l \cap B^-,$$

$$\epsilon \geq -\frac{d_j}{b_j} - \mu_l \text{ if } j \in S^+_l \cap B^+,$$

$$\epsilon \leq -\frac{d_j}{b_j} - \mu_l \text{ if } j \in S^-_l \cap B^+ \text{ and }$$

$$\epsilon \geq -\frac{d_j}{b_j} - \mu_l \text{ if } j \in S^-_l \cap B^-.$$  

If $(S^+_l \cap B^-) \cup (S^-_l \cap B^+) \neq \emptyset$ then choose $r$ such that

$$-\frac{d_r}{b_r} = \min \left\{ -\frac{d_j}{b_j} : j \in (S^+_l \cap B^-) \cup (S^-_l \cap B^+) \text{ and } -\frac{d_j}{b_j} > \mu_l \right\}.$$
The largest value of $\epsilon$ for which $y'$ remains optimal in $[\mu_t, \mu_t + \epsilon]$ is given by
\[
\epsilon = \begin{cases} 
-\mu_t - \frac{d_r}{b_r} & \text{if } (S^+_l \cap B^-) \cup (S^-_l \cap B^+) \neq \emptyset, \\
\infty & \text{otherwise.}
\end{cases}
\]

If $\epsilon \neq \infty$ then $\mu_{t+1} = \mu_t + \epsilon = -(d_r/b_r)$. Let $S = \{-\frac{d_j}{b_j} : j \in B^+ \cup B^-, -\frac{d_j}{b_j} > \lambda\}$. We consider the ascending order $-\frac{d_{\sigma(1)}}{b_{\sigma(1)}} < \cdots < -\frac{d_{\sigma(s)}}{b_{\sigma(s)}}$ of all different element in $S$ and claim that $S \cup \{\lambda, \overline{\lambda}\}$ is the set of all breakpoints of $h_{22}(\mu)$ and for $l = 1, \ldots, s$, $\mu_l = -\frac{d_{\sigma(l)}}{b_{\sigma(l)}}$.

We first show some facts that are useful for our proofs.

**Lemma A.3.** For any $j \in B^+ \cup B^-$, if $-\frac{d_j}{b_j} > \lambda$, then $j \in (S^+_0 \cap B^-) \cup (S^-_0 \cap B^+)$. 

**Proof.** We consider $j \notin (S^+_0 \cap B^-) \cup (S^-_0 \cap B^+)$. Hence, $j \in (S^+_0 \cap B^+) \cup (S^-_0 \cap B^-)$. If $j \in S^+_0 \cap B^+$, then $d_j + b_j \lambda \geq 0$ and $b_j > 0$. Therefore, $-\frac{d_j}{b_j} \leq \lambda$. If $j \in S^-_0 \cap B^-$, then $d_j + b_j \lambda < 0$ and $b_j < 0$. Thus, $-\frac{d_j}{b_j} < \lambda$. \hfill \Box

**Theorem A.4.** For any $l = 1, \ldots, s$, we have $\mu_l = -\frac{d_{\sigma(l)}}{b_{\sigma(l)}}$.

**Proof.** Recall that
\[
\mu_{t+1} = \min \{-\frac{d_j}{b_j} : j \in (S^+_l \cap B^-) \cup (S^-_l \cap B^+) \text{ and } -\frac{d_j}{b_j} > \mu_l\}.
\]

We prove the result by induction.

**Basis Step** Consider
\[
\mu_1 = \min \{-\frac{d_j}{b_j} : j \in (S^+_0 \cap B^-) \cup (S^-_0 \cap B^+) \text{ and } -\frac{d_j}{b_j} > \mu_0 = \lambda\}.
\]

From Lemma A.3, the condition $j \in (S^+_0 \cap B^-) \cup (S^-_0 \cap B^+)$ is redundant and we get $\mu_1 = \min \{-\frac{d_j}{b_j} : -\frac{d_j}{b_j} > \lambda\} = -\frac{d_{\sigma(1)}}{b_{\sigma(1)}}$.

**Inductive Step** Assume that $\mu_k = -\frac{d_{\sigma(k)}}{b_{\sigma(k)}}$. Then
\[
\mu_{k+1} = \min \{-\frac{d_j}{b_j} : j \in (S^+_k \cap B^-) \cup (S^-_k \cap B^+) \text{ and } -\frac{d_j}{b_j} > \mu_k\}
\]
\[
= \min \{-\frac{d_j}{b_j} : (j \in B^-, d_j + b_j \mu_k \geq 0) \text{ or } j \in B^+, d_j + b_j \mu_k < 0\}
\]
\[
\text{and } -\frac{d_j}{b_j} > \mu_k
\]

127
The basis step is clear from the definition of means $d_j = 0$, we have 

$$
\min\left\{ \frac{d_j}{b_j} : j \in B^-, \frac{d_j}{b_j} \geq \mu_k \text{ or } j \in B^+, \frac{d_j}{b_j} > \mu_k \right\}
$$

and 

$$
\min\left\{ \frac{d_j}{b_j} : j \in B^+ \cup B^-, \frac{d_j}{b_j} > \mu_k = \frac{d_{\sigma(k)}}{b_{\sigma(k)}} = \frac{d_{\sigma(k+1)}}{b_{\sigma(k+1)}} \right\}
$$

Let $S(i) = \{ j : -\frac{d_j}{b_j} = -\frac{d_{\sigma(i)}}{b_{\sigma(i)}} \}$. Then the optimal solution $y^l$ corresponding to the breakpoint $\mu_i$ is given recursively by 

$$
y^l_j = \begin{cases}
y^{l-1}_j & \text{if } j \notin S(l), \\
1 & \text{if } j \in S(l) \text{ and } y^{l-1}_j = 0, \\
0 & \text{if } j \in S(l) \text{ and } y^{l-1}_j = 1,
\end{cases}
$$

where $y^0_j = \begin{cases}
1 & \text{if } j \in S_0^+, \\
0 & \text{otherwise}.
\end{cases}$

**Theorem A.5.** For $l = 0, 1, \ldots, s$, $y^l$ is optimal for ULP($\mu_l$).

**Proof.** The basis step is clear from the definition of $y^0$. We assume that $y^l$ is optimal for ULP($\mu_l$) for $l = 1, \ldots, k - 1$.

We first observe that since all $S(l)$ are disjoint, by strong induction hypothesis, if we consider the value of a given $j^{th}$ entry $y^l_0, y^l_1, \ldots, y^l_s$, this entry changes value only once at $y^l$ where $j \in S(l)$. It follows that $y^0_j = y^1_j = \ldots = y^{l-1}_j \neq y^l_j = y^{l+1}_j = \ldots = y^s_j$.

Now we consider an optimal solution $y^*$ for ULP($\mu_k$). We show that $y^* = y^k$.

**Case 1** $j \in S(k)$. From Lemma A.3, we have $j \in (S_0^+ \cap B^-) \cup (S_0^- \cap B^+)$. If $y^{k-1}_j = 0$, we have $y^0_j = y^{k-1}_j = 0$. Thus, $j \in S_0^-$. Hence, $j \in B^+$, which means $d_j + b_j \mu_k > d_j + b_j \mu_{k-1} = d_j + b_j(-d_{\sigma(k-1)}/b_{\sigma(k-1)}) = d_j + b_j(-d_j/b_j) = 0$. It follows that $y^*_j = 1$. If $y^{k-1}_j = 1$, we have $y^0_j = y^{k-1}_j = 1$. Thus, $j \in S_0^+$. Hence, $j \in B^-$, which means $d_j + b_j \mu_k < 0$. It follows that $y^*_j = 0$.

**Case 2** $j \notin S(k)$. We consider three subcases.

**Case 2.1** $j \notin \bigcup_{l=1}^{s} S(l)$. From the previous observation, we can see that $y^0_j = y^1_j = \ldots = y^s_j$. If $j \in B^+$, then $d_j + b_j \mu_k = d_j + b_j \mu_{k-1}$. Thus, $y^*_j = y^{k-1}_j$. If $j \in S_0^+ \cap B^+$, we
have \( d_j + b_j \mu_k > d_j + b_j \Delta \geq 0 \). Hence, \( y_j^* = 1 = y_j^0 = y_j^{k-1} \). If \( j \in S_0^- \cap B^- \), we have \( d_j + b_j \mu_k < d_j + b_j \Delta < 0 \). Hence, \( y_j^* = 0 = y_j^0 = y_j^{k-1} \).

**Case 2.2** \( j \in S(l), l > k \). So \( y_j^0 = y_j^{k-1} \) and \(-d_j/b_j > -d_{\sigma(k)}/b_{\sigma(k)} = \mu_k\). If \( j \in B^+ \), then \( d_j + b_j \mu_k < 0 \) and hence \( y_j^* = 0 \). From Lemma A.3, since \( j \in B^+ \), \( j \in S_0^- \). Hence, \( y_j^* = 0 = y_j^0 = y_j^{k-1} \). If \( j \in B^- \), then \( d_j + b_j \mu_k > 0 \) and hence \( y_j^* = 1 \). From Lemma A.3, \( j \in S_0^+ \). Hence, \( y_j^* = 1 = y_j^0 = y_j^{k-1} \).

**Case 2.3** \( j \in S(l), l < k \). Then \( y_j^0 = 0 \) if and only if \( y_j^{k-1} = 1 \). It follows that \( y_j^{k-1} = 1 - y_j^0 \). Moreover, \(-d_j/b_j = -d_{\sigma(l)}/b_{\sigma(l)} < -d_{\sigma(k)}/b_{\sigma(k)} = \mu_k\). If \( j \in B^+ \), then \( d_j + b_j \mu_k > 0 \) and hence \( y_j^* = 1 \). From Lemma A.3, \( j \in S_0^- \). Hence, \( y_j^* = 1 - y_j^0 = y_j^{k-1} \). If \( j \in B^- \), then \( d_j + b_j \mu_k < 0 \) and hence \( y_j^* = 0 \). From Lemma A.3, \( j \in S_0^+ \). Hence, \( y_j^* = 0 = 1 - y_j^0 = y_j^{k-1} \).

Define

\[
D^0 = \sum_{j \in S_0^+} d_j, \quad D^l = D^{l-1} - \sum_{j \in S(l), \ y_j^{l-1} = 1} d_j + \sum_{j \in S(l), \ y_j^{l-1} = 0} d_j \quad \text{and}
\]

\[
B^0 = \sum_{j \in S_0^-} b_j, \quad B^l = B^{l-1} - \sum_{j \in S(l), \ y_j^{l-1} = 1} b_j + \sum_{j \in S(l), \ y_j^{l-1} = 0} b_j.
\]

Then the optimal objective function value at \( \mu_l \) is given by \( h_2(\mu_l) = D^l + \mu_l B^l \).

Given \( y^{l-1}, D^{l-1} \) and \( B^{l-1} \), we can compute \( y^l, D^l \), and \( B^l \) in \( O(|S(l)|) \) time and, hence, \( h_2(\mu_l) \) and \( y^l \) can be identified in \( O(|S(l)|) \) time. Since \( S(l) \cap S(k) = \emptyset \) for \( l \neq k \), \( y^l \) and \( h_2(\mu_l) \) for \( l = 1, \ldots, t \) can be identified in \( O(n) \) time.

Now the algorithm for solving UBLP(1) can be described as follows.

**Algorithm A.1.** \( O(n \log n) \) Algorithm for UBLP(1).

**Step 1:** Compute and sort all breakpoints of \( h_1(\lambda) \) and \( h_2(\mu) \) for \( \lambda < \lambda, \mu \leq \bar{\lambda} \).

**Step 2:** Compute \( x^0, y^0, h_1(\lambda) \) and \( h_2(\lambda) \). Set \( f(x^0, y^0) = h_1(\lambda) + h_2(\lambda) \). Set \( k := 0 \), \( (x^*, y^*) := (x^0, y^0) \) and \( f(x^*, y^*) := f(x^0, y^0) \).

**Step 3:** We scan all breakpoints starting from \( \lambda \) in the increasing order.

**Step 3.1:** Start from \( \lambda_k \), when we pass breakpoints \( \mu_l < \lambda_{k+1} \) of \( h_2(\mu) \), keep updating \( y^l \) and \( h_2(\mu_l) \) until we hit \( \lambda_{k+1} \). Set \( k := k + 1 \).

**Step 3.2:** Compute \( x^k \) and \( h_1(\lambda_k) \). The most recent solution \( y^l \) identified is selected.
as $y^k$ and compute $h_2(\lambda_k)$ using slope of $h_2(\mu)$ for the interval containing $\lambda^k$.

**Step 3.3:** Update $f(x^k, y^k) = h_1(\lambda_k) + h_2(\lambda_k)$. If $f(x^k, y^k) > f(x^*, y^*)$, set $(x^*, y^*) := (x^k, y^k)$ and $f(x^*, y^*) := f(x^k, y^k)$. If $k < p$, go back to **Step 3.1**. Otherwise, output $(x^*, y^*)$.

Recall that $m \leq n$. It takes $O(m + n) = O(n)$ time to obtain all $p + 1 + l \leq m + n + 1$ breakpoints in **Step 1**. Thus, the complexity of **Step 1** is dominated by the sorting time which is $O(n \log n)$. It is easy to see that **Step 2** can be completed in $O(n)$ time. We have shown before that all $x^k, h_1(\lambda_k), y^l, h_2(\lambda_l)$ can be determined in $O(m + n) = O(n)$. Moreover, **Step 3.2** computes $h_2(\lambda_k)$ in $O(1)$ time using slope. Thus, **Step 3** is done in $O(n)$ time. Therefore, the overall complexity of Algorithm A.1 is $O(n \log n)$. 

130
Appendix B

BQP01 with $Q$ as a $(2p + 1)$-diagonal Matrix

Theorem 2.11. Let $Q$ be a $(2p + 1)$-diagonal matrix and $p = O(\log n)$. Then Algorithm 2.5 solves BQP01 in polynomial time.

Proof. Assume that $Q$ is an $n \times n$ symmetric matrix with exactly $2p + 1$ diagonals where $p$ is $O(\log n)$. Then Step 2 and Step 4 in the algorithm end in $2n - 1$ iterations. The bottleneck process in each iteration is computing $\phi_k(z)$ which can be done in $O(|V_k|2^{|V_k|})$ time where $V_k$ is the set of all variables appearing in $\Delta_k(z)$. We claim that in all iteration, $|V_k| \leq 2p$. Thus, we can compute $\phi_k(z)$ in polynomial time since $p$ is $O(\log n)$.

To prove the claim, we will prove the following statement by induction on $h$:

$P(h) : f_{2n-2h}(z) = r(x_{n-h-p+1}, y_{n-h-p+1}, \ldots, x_{n-h}, y_{n-h})$
$+ f(x_1, y_1, \ldots, x_{n-h}, y_{n-h}, 0_{2h}),$

for some polynomial $r$ with variables $x_{n-h-p+2}, y_{n-h-p+2}, \ldots, x_{n-h}, y_{n-h}$. From now on, for any function $g(z)$, we denote $r_g(z_{\pi(1)}, \ldots, z_{\pi(l)})$ the polynomial that represents $g(z)$ and $z_{\pi(1)}, \ldots, z_{\pi(l)}$ are the variables involving $g(z)$. Note that any variable in $g(z)$ with nonzero coefficients must be in $\{z_{\pi(1)}, \ldots, z_{\pi(l)}\}$ and $\{z_{\pi(1)}, \ldots, z_{\pi(l)}\} \subseteq \{z_1, \ldots, z_{2n}\}$. However, the set of variables with nonzero coefficients in $g(z)$, $\{z_{\pi(1)}, \ldots, z_{\pi(l)}\}$ and $\{z_1, \ldots, z_{2n}\}$ are not necessarily the same.
**Basis Step** To find $f_{2n-2}(z)$, we start from computing $f_{2n-1}(z)$. In the first iteration, we have

\[
\begin{align*}
\Theta_{2n}(z) &= f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n, 0) \\
\Delta_{2n}(z) &= f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n, 1) - f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n, 0) \\
&= \sum_{i=n-p}^{n} q_{in}x_i + d_n.
\end{align*}
\]

There are at most $p + 1$ variables with nonzero coefficients in $\Delta_{2n}(z)$. Thus, $\phi_{2n}(z)$ is a polynomial in the form $r\phi_{2n}(x_{n-p}, \ldots, x_n)$ which can be determined in at most $2^{p+1}$ iterations.

Therefore, we have

\[
\begin{align*}
f_{2n-1}(z) &= r\phi_{2n}(x_{n-p}, \ldots, x_n)(\sum_{i=n-p}^{n} q_{in}x_i + d_n) \\
&\quad + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n, 0) \\
&= r_{2n-1}(x_{n-p}, \ldots, x_n) + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n, 0),
\end{align*}
\]

where $r_{2n-1}(x_{n-p}, \ldots, x_n) = r\phi_{2n}(x_{n-p}, \ldots, x_n)(\sum_{i=n-p}^{n} q_{in}x_i + d_n)$. Hence, in the second iteration, we have

\[
\begin{align*}
\Theta_{2n-1}(z) &= r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 0) + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, 0, 0) \\
\Delta_{2n-1}(z) &= r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 1) + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, 1, 0) \\
&\quad - r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 0) - f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, 0, 0) \\
&= r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 1) - r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 0) \\
&\quad + \sum_{j=n-p}^{n-1} q_{nj}y_j + c_n \\
&= r_{2n-1}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}).
\end{align*}
\]

Since there are at most $2p$ variables with nonzero coefficients in $\Delta_{2n-1}(z)$, $\phi_{2n-1}(z)$ is in the form $r\phi_{2n-1}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1})$ which can be determined in at most $2^{2p}$
iterations. It follows that

\[ f_{2n-2}(z) = r_{\phi_{2n-1}}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) \]
\[ \cdot r_{\Delta_{2n-1}}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) \]
\[ + r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 0) + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, 0, 0) \]
\[ = r_{2n-2}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) \]
\[ + f(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, 0, 0), \]

where

\[ r_{2n-2}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) = r_{\phi_{2n-1}}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) \]
\[ \cdot r_{\Delta_{2n-1}}(x_{n-p}, y_{n-p}, \ldots, x_{n-1}, y_{n-1}) + r_{2n-1}(x_{n-p}, \ldots, x_{n-1}, 0). \]

**Induction Step** Assume \( P(h) \). Consider \( f_{2n-2h-1} \). We have

\[ \Theta_{2n-2h}(z) = r_{2n-2h}(x_{n-h-p+1}, y_{n-h-p+1}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0) \]
\[ + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0_{2h+1}) \text{ and} \]
\[ \Delta_{2n-2h}(z) = r_{2n-2h}(x_{n-h-p+1}, y_{n-h-p+1}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 1) \]
\[ - r_{2n-2h}(x_{n-h-p+1}, y_{n-h-p+1}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0) \]
\[ + \sum_{i=n-h-p}^{n-h} q_{i,n-h}x_i + d_{n-h} \]
\[ = r_{\Delta_{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}). \]

Since there are at most \( 2p + 1 \) variables involving in \( \Delta_{2n-2h}(z) \), \( \phi_{2n-2h}(z) \) is in the form \( r_{\phi_{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \) which can be determined in at most \( 2^{2p+1} \) iterations. It follows that

\[ f_{2n-2h-1}(z) = r_{\phi_{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \]
\[ \cdot r_{\Delta_{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \]
\[ + r_{2n-2h}(x_{n-h-p+1}, y_{n-h-p+1}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0) \]
\[ + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0_{2h+1}) \]
\[ = r_{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \]
\[ + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0_{2h+1}). \]
where
\[
\begin{align*}
    r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \\
    &= r_{\phi^{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \\
    &\quad \cdot r_{\Delta^{2n-2h}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}) \\
    &\quad + r^{2n-2h}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, x_{n-h}, 0).
\end{align*}
\]

In the next iteration, we have
\[
\begin{align*}
    \Theta^{2n-2h-1}(z) &= r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, 0) \\
    &\quad + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, 0, 2h+2) \quad \text{and} \\
    \Delta^{2n-2h-1}(z) &= r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, 1) \\
    &\quad - r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, 0) \\
    &\quad + \sum_{j=n-h}^{n-h-1} q_{n-h,j}y_j + c_{n-h} \\
    &= r_{\Delta^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}).
\end{align*}
\]

Since there are at most \(2p\) variables involving in \(\Delta^{2n-2h-1}(z)\), \(\phi^{2n-2h-1}(z)\) is in the form \(r_{\phi^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1})\) which can be determined in at most \(2^{2p}\) iterations. It follows that
\[
\begin{align*}
    f^{2n-2(h+1)}(z) &= f^{2n-2h-2}(z) \\
    &= r_{\phi^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &\quad \cdot r_{\Delta^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &\quad + r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, 0) \\
    &\quad + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, 0, 2h+2) \\
    &= r^{2n-2h-2}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &\quad + f(x_1, y_1, \ldots, x_{n-h-1}, y_{n-h-1}, 0, 2h+2) \\
    &= r^{2n-2(1+1)}(x_{n-(h+1)-p+1}, y_{n-(h+1)-p+1}, \ldots, x_{n-(h+1)}, y_{n-(h+1)}) \\
    &\quad + f(x_1, y_1, \ldots, x_{n-(h+1)}, y_{n-(h+1)}, 0, 2(h+1)),
\end{align*}
\]

where
\[
\begin{align*}
    r^{2n-2h-2}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &= r_{\phi^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &\quad \cdot r_{\Delta^{2n-2h-1}}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}) \\
    &\quad + r^{2n-2h-1}(x_{n-h-p}, y_{n-h-p}, \ldots, x_{n-h-1}, y_{n-h-1}, 0).
\end{align*}
\]
It follows from the arguments in the induction that $\phi_k(z)$ can be verified in $O(2^{2p})$ time which is polynomial since $p = O(\log n)$. \qed
Appendix C

Supplement on the Bipartite Boolean Quadratic Polytope

C.1 Partial Linear Relaxation of $\text{BQP}^{m,n}$

Here we provide the omitted proof of Theorem 4.22.

**Theorem 4.22.** $\text{BQP}^{m,n}_z = \text{BQP}^{m,n}$.

**Proof.** We consider the same polyhedron as in the previous proof. Fix $z \in \{0,1\}^{mn}$ and see $z_{ij}$ as a constant in $\{0,1\}$. Hence, we can restate the constraints as

\[
\begin{align*}
    x_i + y_j &\leq 1 + z_{ij}, \\
    -x_i &\leq -z_{ij}, \\
    -y_j &\leq -z_{ij}, \\
    x_i, y_j &\geq 0,
\end{align*}
\]

for $i = 1, \ldots, m$ and $j = 1, \ldots, n$. By the Hoffman and Kruskal Theorem, the polyhedron $P_z := \{ (x, y) : A(x, y) \leq b, x, y \geq 0 \}$ is integral for any $b \in \mathbb{Z}^{3mn}$ if and only if $A$ is totally unimodular.

Using notation $u_i$ as defined earlier in this section, we can write the constraint matrix $A$ as
\[ A = \begin{bmatrix}
  u^T_1 & I_n & \\
  \vdots & \ddots & I_n \\
  u^T_m & \vdots & 0_{mn \times n} \\
  -I_m & \vdots & -I_n \\
  -I_m & \vdots & 0_{mn \times m} \\
  0_{mn \times m} & \vdots & -I_m 
\end{bmatrix}, \]

which has dimension \(3mn \times (m + n)\).

Note that each entry of \(A\) is in \([-1, 0, 1]\). Besides, each row of \(A\) has at most two entries. Let \(C\) be the set of all columns of \(A\). Partition \(C\) into \(\{\{1, \ldots, m\}, \{m + 1, \ldots, m + n\}\}\). For each row \(i\) with exactly two entries, we have,

\[
\sum_{i=1}^{m} a_{ij} = 1 = \sum_{i=m+1}^{m+n} a_{ij}.
\]

Hence, \(A\) is totally unimodular. It follows that \(P_z\) is integral. Similar to the previous proof, we can summarize that \(BQP_{m,n}^z = BQP_{m,n}^m\).

### C.2 Families of Valid Inequalities Obtained from Rounding Coefficients

The proofs for all theorem in Section 4.2 are presented here.

**Theorem 4.24.** Inequality (4.8) does not define a facet for \(BQP_{m,n}^z\). Moreover, (4.8) is tight only when \(S_1\) or \(S_2\) is a singleton and \(\alpha = |S| - 2\).

**Proof.** Let \(s := |S|,\ \ h := |S_1|\) and \(k := |S_2|\). Without loss of generality, we assume that \(h \leq k\).
We first note that if $S_1$ or $S_2$ is empty, the term $z(S_1 : S_2)$ would disappear and by Proposition 4.15, (4.8) is not facet defining. Without loss of generality, let $S_1 = \emptyset$. So we have $k = s$. Then the inequality becomes

$$
\alpha y(S_2) \leq \frac{\alpha(\alpha + 1)}{2} + \frac{k(k-1)}{2}.
$$

(c0)

The right hand side reaches its maximum value at $\alpha k$ when $R \cap S = S_2$. Since

$$
2(\alpha k - \frac{\alpha(\alpha + 1)}{2} - \frac{k(k-1)}{2}) = (\alpha - k + 1)(\alpha - k),
$$

(c0) holds at equality only when $\alpha = k - 1$ or $\alpha = k$. Recall that $\alpha \leq s - 2 = k - 2$. Then there are no vertices $\omega^R$ of BQP$^{m,n}$ where $RCq(\omega^R) = \beta$ when $S_1$ or $S_2$ is empty.

Besides, consider the case when $S_1$ or $S_2$ is a singleton. We assume without loss of generality that $S_1 = \{u\}$. Hence, we have $s = k + 1$. Consider the sum of inequalities $x_u + y_j - z_{uj} \leq 1$ over all $j \in S_2$, $(\alpha - k)x_u \leq 0$ and $(\alpha - 1)y_j \leq \alpha - 1$ for all $j \in S_2$. Then we get

$$
\alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq k + k(\alpha - 1) = \alpha k.
$$

(c1)

When $h = 1$, $\beta = \frac{\alpha(\alpha + 1)}{2} + \frac{k(k-1)}{2}$. Then

$$
2(\beta - \alpha k) = (\alpha - k)^2 + (\alpha - k) \geq 0.
$$

Thus, (4.8) is not facet-defining. Furthermore, when $R \cap S = S_2$ and $\alpha = k - 1 = s - 2$, (c1) holds with equality and $\beta = \alpha k$. Therefore, (4.8) is tight in this case.

From now on, we assume that $h, k \geq 2$. Here we consider three different cases.

**Case 1** $h \leq k < \alpha$. We consider the sum of all $hk$ pairs of $x_i + y_j - z_{ij} \leq 1$ where $ij \in (S_1 : S_2)$. Since each $x_i$ appears $k$ times and each $y_j$ appears $h$ times in this sum, we add the inequalities $(\alpha - k)x_i \leq \alpha - k$ for all $i \in S_1$ and $(\alpha - h)y_j \leq \alpha - h$ for all $j \in S_2$. Then we get

$$
\alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq hk + h(\alpha - k) + k(\alpha - h)
$$

$$
= \alpha h + \alpha k - hk.
$$

Since we have

$$
2(\beta - \alpha h - \alpha k + hk) = (\alpha - h - k)^2 + (\alpha - h - k) \geq 0,
$$

(g1)

(4.8) is not facet-defining. However, we can see from (g1) that $\beta = \alpha h + \alpha k - hk$ if and only if $\alpha = h + k = s$ or $\alpha = h + k - 1 = s - 1$. Since $\alpha \leq s - 2$, there is no $\omega^R$ where
\( RCq(\omega^R) = \beta \) in Case 1.

**Case 2** \( h < \alpha \leq k \). We again consider the sum of \( x_i + y_j - z_{ij} \leq 1 \) where \( ij \in (S_1 : S_2) \) and add the inequalities \((\alpha - k)x_i \leq 0 \) for all \( i \in S_1 \) and \((\alpha - h)y_j \leq \alpha - h \) for all \( j \in S_2 \) to obtain

\[
\alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq hk + k(\alpha - h) = k\alpha.
\]

Since

\[
2(\beta - \alpha k) = (\alpha - k)^2 - (\alpha - k) + h^2 - h \geq 0, \quad (g2)
\]

inequality (4.8) is redundant. From (g2), we can see that if \( \beta = \alpha k \), then \( h^2 - h \) need to be zero. Since \( h \geq 2 \), (4.8) is also not tight in this case.

**Case 3** \( \alpha \leq h \leq k \). Let

\[
U := \frac{-1 + \sqrt{1 + 4(s - (h - k)^2)}}{2} \quad \text{and}
\]

\[
L := \frac{2s - 1 - \sqrt{8h(k - 1) + 1}}{2}.
\]

To proof this case, we need to show the three following claims.

**Claim 1** For a given \( s = h + k > 0 \), there exists no \( \alpha \in \mathbb{Z} \) such that \( L < \alpha < U \).

**Claim 2** (4.8) is not facet-defining if \( \alpha \geq U \).

**Claim 3** (4.8) is not facet-defining if \( \alpha \leq L \).

**Claim 1** implies that for a given \( \alpha \), \( \alpha \leq L \) or \( \alpha \geq U \). From **Claim 2** and **Claim 3**, (4.8) with this \( \alpha \) does not define a facet. Then we are done.

As for **Claim 1**, since we assume that \( h, k \geq 2 \), \( S \) has size at least four. It is easy to verify that **Claim 1** holds for \( 4 \leq s \leq 6 \). Thus, we can assume that \( s \geq 7 \).

We first consider the polynomial \( p(s) := s^4 - 8s^3 + 10s^2 - 8s + 1 \) which is increasing for \( s \geq 6 \), and non-negative for \( s \geq 6.787 \). Then for \( s \geq 7 \), we obtain

\[
0 \leq s^4 - 8s^3 + 10s^2 - 8s + 1 = (s^2 + 4s - 1)^2 - 4s^2(4s + 1)
\]

\[
2s\sqrt{4s + 1} \leq s^2 + 4s - 1
\]

\[
2s^2 - 4s + 3 \leq 4s^2 - 4s\sqrt{4s + 1} + 4s + 1 = (2s - \sqrt{4s + 1})^2
\]

\[
\sqrt{4s + 1} \leq 2s - \sqrt{2s^2 - 4s + 3}.
\]
It follows that
\[
U = \frac{-1 + \sqrt{1 + 4(s - (h - k)^2)}}{2} \leq \frac{\sqrt{4s + 1} - 1}{2} \leq \frac{2s - 1 - \sqrt{2s^2 - 4s + 3}}{2} \leq \frac{2s - 1 - \sqrt{8h(k - 1) + 1}}{2} = L.
\]

Thus, we get the claim.

To show Claim 2, consider the sum of \(x_i + y_j - z_{ij} \leq 1\) where \(ij \in (S_1 : S_2)\), \((\alpha - k)x_i \leq 0\) for all \(i \in S_1\) and \((\alpha - h)y_j \leq 0\) for all \(j \in S_2\). This yields
\[
\alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq hk. \tag{s3.1}
\]
Since \(\alpha \geq U\),
\[
2(\beta - hk) = \alpha^2 + \alpha + (h - k)^2 - s \geq 0. \tag{g3.1}
\]
Thus, (4.8) does not define a facet.

We consider the different sum of valid inequalities to prove Claim 3. Let \(S_1 = \{1, \ldots, h\}\) and \(S_2 = \{1, \ldots, k\}\). Take the sum of \(x_i + y_j - z_{ij} \leq 1\) for \(i = 1, \ldots, h\), \((\alpha - 1)x_i \leq \alpha - 1\) for all \(i = 1, \ldots, h\), \((\alpha - 1)y_j \leq \alpha - 1\) for all \(j = 1, \ldots, h\), \(\alpha y_j \leq \alpha\) for \(j = h + 1, \ldots, k\), and \(-z_{ij} \leq 0\) for all \(i = 1, \ldots, h\) and \(j = 1, \ldots, k\) where \(i \neq j\). The sum of these inequalities becomes
\[
\alpha x(S_1) + \alpha y(S_2) - z(S_1 : S_2) \leq h + h(\alpha - 1) + h(\alpha - 1) + \alpha(k - h) = \alpha s - h. \tag{s3.2}
\]
Since \(\alpha \leq L\), we have
\[
2(\beta - \alpha s + h) = \alpha^2 + (1 - 2s)\alpha + k^2 - k + h^2 + h \geq 0. \tag{g3.2}
\]
Thus, we get the claim.

Suppose that (4.8) is tight, which means (s3.1), (g3.1), (s3.2) and (g3.2) are tight. Claim 1 states that \(\alpha \geq U\) or \(\alpha \leq L\), so we can consider it in two cases.

Case \(\alpha \geq U\). We consider (s3.1) and (g3.1), that is there exist \(\omega^R\) satisfying (s3.1) with equality and \(S_1\) and \(S_2\) such that \(\beta = hk\). It follows that \(x_i^R + y_j^R - z_{ij}^R = 1\) for all \(ij \in (S_1 : S_2)\), \((\alpha - k)x_i^R = 0\) for all \(i \in S_1\) and \((\alpha - h)y_j^R = 0\) for all \(j \in S_2\).

When \(\alpha \neq h\) and \(\alpha \neq k\), we have \(x_i^R = 0\) for all \(i \in S_1\) and \(y_j^R = 0\) for all \(j \in S_2\). It follows that \(z_{ij}^R = x_i^R y_j^R = 0\) for all \(ij \in (S_1 : S_2)\). Hence, \(x_i^R + y_j^R - z_{ij}^R = 0 < 1\) for all
If \( \alpha = h \) or \( \alpha = k \), since \( \alpha \leq h \leq k \), in both cases, we obtain \( \alpha = h \). Since \( \alpha^2 + \alpha + (h-k)^2 - s = 2(\beta - hk) = 0 \) and \( \alpha \geq U \), we get \( \alpha = U \). From \( h = \alpha = U \), one can derive that \( h = (2k + \sqrt{8k - 4k^2})/4 \). Since \( h \) is a real number, we have \( 0 \leq k \leq 2 \). Since we assume that \( k \geq 2 \), it implies that \( k = 2 \) and it yields \( h = 1 \) which contradicts to the assumption that \( h \geq 2 \).

**Case** \( \alpha \leq L \). We consider (s3.2) and (g3.2), that is there exist \( \omega^R \) satisfying (s3.2) with equality and \( S_1 \) and \( S_2 \) such that \( \beta = \alpha s - h \). Thus, \( x_i^R + y_i^R - z_{ij}^R = 1 \) for \( i = 1, \ldots, h \), \( (\alpha - 1)x_i^R = \alpha - 1 \) for all \( i = 1, \ldots, h \), \( (\alpha - 1)y_j^R = \alpha - 1 \) for all \( j = 1, \ldots, h \), \( \alpha y_j^R = \alpha \) for \( j = h + 1, \ldots, k \), and \( -z_{ij}^R = 0 \) for all \( i = 1, \ldots, h \) and \( j = 1, \ldots, k \) where \( i \neq j \).

If \( \alpha \neq 1 \), we have \( x_i^R = 1 \) for all \( i = 1, \ldots, h \) and \( y_j^R = 1 \) for all \( j = 1, \ldots, k \). Since \( h, k \geq 2 \), there exists \( z_{ij} \) where \( ij \in (S_1 : S_2) \) and \( i \neq j \) such that \( -z_{ij} = -x_i y_j = -1 < 0 \), a contradiction.

When \( \alpha = 1 \), note that \( \alpha^2 + (1-2s)\alpha + k^2 - k + h^2 + h = 2(\beta - \alpha s + h) = 0 \) and \( \alpha \leq L \) implies that \( \alpha = L \). From \( L = \alpha = 1 \), one can derive that \( h(h-1) + (k-1)(k-2) = 0 \). Since \( h, k \geq 2 \), we have \( h(h-1) > 0 \) and \( (k-1)(k-2) \geq 0 \), a contradiction.

Therefore, we can conclude that there are no \( \omega^R \) such that \( RCq(\omega^R) \) holds with equality for Case 3. \( \square \)

**Theorem 4.25.** For any subsets \( S_1 \subseteq I \) and \( S_2 \subseteq J \), let \( \alpha = \max \{|S_1|, |S_2|\} \). Then (4.9) is valid for BQP\(^{m,n}\). However, (4.9) does not define a facet for BQP\(^{m,n}\) unless \( |S_1| = |S_2| = 1 \).

**Proof.** Recall that for a subset \( R = R_1 \cup R_2 \subseteq V \) where \( R_1 \subseteq I \) and \( R_2 \subseteq J \), \( \omega^R = (x^R, y^R, z^R) \) is

\[
\begin{align*}
x_i^R &= \begin{cases} 
1 & \text{if } i \in R, \\
0 & \text{otherwise,}
\end{cases} \\
y_j^R &= \begin{cases} 
1 & \text{if } j \in R, \\
0 & \text{otherwise,}
\end{cases} \\
z_{ij}^R &= \begin{cases} 
1 & \text{if } ij \in (R_1 : R_2), \\
0 & \text{otherwise.}
\end{cases}
\end{align*}
\]

We denote \( k_i = |R \cap S_i| \) for \( i = 1, 2 \). Define \( f(\omega^R) = BCq(\omega^R) - \alpha^2 \). Then

\[
f(\omega^R) = \alpha(x^R(S_1) + y^R(S_2)) - z^R(S_1 : S_2) - \alpha^2 = -(\alpha - k_1)(\alpha - k_2). \quad (\ast)
\]

Since \( k_i = |R \cap S_i| \leq |S_i| \leq \max \{|S_1|, |S_2|\} = \alpha \) for \( i = 1, 2 \), \( f(\omega^R) \leq 0 \).
Let $F$ be a subset of $\text{BQP}^{m,n}$ where $F = \{ \omega^R \in \text{BQP}^{m,n} : BCq(\omega^R) = 0 \}$. For any $\omega^R \in \text{BQP}^{m,n}$, from (*), $BCq(\omega^R) = 0$ if and only if $\alpha = k_1 = |R \cap S_1|$ or $\alpha = k_2 = |R \cap S_2|$.

Firstly, we assume that $\alpha = |S_1| \geq |S_2|$. It follows that $|S_1| = \alpha = k_1 = |R \cap S_1|$. Hence, for any $\omega^R \in F$, $x_i^R = 1$ for all $i \in S_1$. Consequently, $\text{dim}(F) \leq m + n + mn - |S_1|$. $F$ can have dimension equal to $m + n + mn - 1$ only when $1 \geq |S_1| \geq |S_2|$. Similarly, if $\alpha = |S_2| \geq |S_1|$, we can conclude that $F$ can have dimension equal to $m + n + mn - 1$ only when $1 \geq |S_2| \geq |S_1|$. Therefore, we have only four possibilities as follow:

Case 1 $|S_1| = 0$ and $|S_2| = 0$. We obtain $0 \leq 0$ which does not define a facet.

Case 2 $|S_1| = 1$ and $|S_2| = 0$. The inequality is in the form $x_i \leq 1$ which is the summation of $x_i + y_j - z_{ij} \leq 1$ (4.1) and $-y_j + z_{ij} \leq 0$ (4.3). Thus, it is redundant and not facet-defining.

Case 3 $|S_1| = 0$ and $|S_2| = 1$. In this case, (4.9) becomes $y_j \leq 1$ which is the summation of $x_i + y_j - z_{ij} \leq 1$ (4.1) and $-x_i + z_{ij} \leq 0$ (4.2). As well as the previous case, it does not define a facet for $\text{BQP}^{m,n}$.

Case 4 $|S_1| = 1$ and $|S_2| = 1$. We have $x_i + y_j - z_{ij} \leq 1$ which is exactly the same as (4.1), defining a trivial facet of $\text{BQP}^{m,n}$.

**Theorem 4.26.** (4.11) is valid for $\text{BQP}^{m,n}$ but is not facet-defining. Moreover, there is no vertices $\omega \in \text{BQP}^{m,n}$ where $\text{SRCut} \omega = \gamma$.

**Proof.** Let $u \in T_1$ and $v \in T_2$. Consider the left hand side of (4.10) which is

$$-x(S_1) - y(S_2) - z(S_1 : S_2) + z(S_1 : T_2) + z(T_1 : S_2) - z(T_1 : T_2)$$

$$= -x(S_1) - z(S_1 : S_2) + z(S_1 : v) + z(S_1 : T_2 \{v\}) +$$

$$-y(S_2) - z(T_1 : T_2) + z(u : S_2) + z(T_1 \{u\} : S_2).$$

Adding

$$-x_i + z_{iv} \leq 0, \quad i \in S_1,$$

$$-y_j + z_{aj} \leq 0, \quad j \in S_2,$$

$$z_{ij} \leq 1, \quad i \in S_1, j \in T_2 \{v\} \text{ or } i \in T_1 \{u\}, j \in S_2,$$

$$-z_{ij} \leq 0 \quad i \in S_1, j \in S_2 \text{ or } i \in T_1, j \in T_2$$
-x(S_1) - z(S_1 : S_2) + z(S_1 : v) + z(S_1 : T_2 \{v\}) +
- y(S_2) - z(T_1 : T_2) + z(u : S_2) + z(T_1 \{u\} : S_2)
\leq |S_1|(|T_2| - 1) + |S_2|(|T_1| - 1).

Therefore, (4.11) is valid for BQP \(^{m,n}\) but does not define a facet.

Let \(\omega = (x, y, z)\) be a vertex in BQP \(^{m,n}\). Consider an edge \(pq\) where \(p \in S_1\) and \(q \in S_2\). If \(z_{pq} = 1\), then \(-z_{pq} = -1 < 0\). Thus, \(SRCut(\omega) < \gamma\). If \(z_{pq} = 0\), then \(x_p\) or \(y_q\) must be zero. Without loss of generality, we assume that \(x_p = 0\). Then \(z_{pj} = 0 < 1\) for all \(j \in T_2 \{v\}\). So we obtain \(SRCut(\omega) < \gamma\) as well.

**Theorem 4.27.** For any disjoint subsets \(S, T \subseteq V\), where \(S_1 := S \cap I\), \(S_2 := S \cap J\), \(T_1 := T \cap I\) and \(T_2 := T \cap J\), then (4.12) is valid for BQP \(^{m,n}\). However, (4.12) does not define a facet for BQP \(^{m,n}\).

**Proof.** Let \(R \subseteq V\) and \(\omega^R = (x^R, y^R, z^R)\) is a vertex of BQP \(^{m,n}\). Denote \(s_i = |S_i|\), \(t_i = |T_i|\), \(p_i = |S_i \cap R|\) and \(q_i = |T_i \cap R|\) for \(i = 1, 2\). Then

\[
\text{ACut}(\omega^R) = -x^R(S_1) + x^R(T_1) - z^R(S_1 : S_2) + z^R(S_1 : T_2)
+ z^R(T_1 : S_2) - z^R(T_1 : T_2)
= (p_1 - q_1)(q_2 - p_2 - 1). \quad (*)
\]

We can see that when \(\text{ACut}(\omega^R) \leq 0\), we have \(\text{ACut}(\omega^R) \leq 0 \leq t_1(s_2 + 1) \leq \delta\). Thus, (4.12) is valid in this case. When \(\text{ACut}(\omega^R) > 0\), we can consider in two cases. Let \(P = p_1 - q_1\) and \(Q = q_2 - p_2 - 1\).

**Case 1** \(P > 0\) and \(Q > 0\). Then \(P = p_1 - q_1 \leq s_1\) and \(Q = q_2 - p_2 - 1 \leq t_2 - 1\). It follows that \(\text{ACut}(\omega^R) = PQ \leq s_1(t_2 - 1) \leq \delta\).

**Case 2** \(P < 0\) and \(Q < 0\). Then \(|P| = q_1 - p_1 \leq t_1\) and \(|Q| = 1 - q_2 + p_2 \leq s_2 + 1\). It follows that \(\text{ACut}(\omega^R) = PQ = |P||Q| \leq t_1(s_2 + 1) \leq \delta\).

Now we consider three cases depending on the sizes of \(S_1, S_2\) and \(T_2\).
**Case 1** \( S_1 \) is empty. Then (4.12) becomes
\[
x(T_1) + z(T_1 : S_2) - z(T_1 : T_2) \leq \max \{0, t_1(s_2 + 1)\}
\]
\[
= t_1s_2 + t_1.
\]

From Proposition 4.1, we obtain \( x_i \leq 1 \) and \( z_{ij} \leq 1 \) from (4.1)-(4.4). Summing over all \( i \in T_1 \), we have
\[x(T_1) = \sum_{i \in T_1} x_i \leq t_1\]
and \( z(T_1 : j) = \sum_{i \in T_1} z_{ij} \leq t_1 \). Summing the second inequality over all \( j \in S_2 \), we get
\[z(T_1 : S_2) = \sum_{j \in S_2} \sum_{i \in T_1} z_{ij} \leq t_1s_2.\]

Moreover, from \(-z_{ij} \leq 0 \) (4.4), Summing over all \( i \in T_1 \) and \( j \in T_2 \), then we have
\[-z(T_1 : T_2) \leq 0.\]

Combining these three inequalities together, we get (c1).

**Case 2** \( s_2 + t_2 \leq 1 \). When \( S_2 = T_2 = \emptyset \), (4.12) becomes \(-x(S_1) + x(T_1) \leq \delta \). By Proposition 4.15, it can not define a facet. Thus, one of them must be a singleton and the other one must be empty.

**Case 2.1** \( S_2 \) is empty and \( T_2 = \{j\} \). Then (4.12) is in the form
\[-x(S_1) + x(T_1) + z(S_1 : j) - z(T_1 : j) \leq \max \{0, t_1\} = t_1. \quad \text{(c2.1)}\]
Consider the sum of \(-x_i + z_{ij} \leq 0 \) (4.1) over all \( i \in S_1 \). We get
\[-x(S_1) + z(S_1 : j) \leq 0.\]
Furthermore, we have \( x_i \leq 1 \) obtained from trivial facet-defining inequalities and \(-z_{ij} \leq 0 \) (4.4). Summing them over all \( i \in T_1 \), we obtain
\[x(T_1) \leq t_1 \text{ and } -z(T_1 : j) \leq 0,\]
respectively. We can get (c2.1) by adding these three inequalities together.
Case 2.2 $T_2$ is empty and $S_2 = \{j\}$. In this case, we have

$$-x(S_1) + x(T_1) - z(S_1 : j) + z(T_1 : j) \leq max\{ -s_1, 2t_1 \} = 2t_1. \ (c2.2)$$

Again, we consider the inequality

$$x(T_1) \leq t_1.$$

From Proposition 4.1, we obtain $z_{ij} \leq 1$ and $-x_i \leq 0$ from the trivial facet inequalities. Sum these two inequalities over all $i \in T_1$ and $i \in S_1$, respectively. Then we get

$$z(T_1 : j) \leq t_1 \quad \text{and} \quad -x(S_1) \leq 0,$$

respectively. Besides, when summing (4.4) over all $i \in S_1$, we have

$$-z(S_1 : j) \leq 0.$$

Combining these four inequalities together, we can get (c2.2).

Case 3 $S_1 \neq \emptyset$ and $s_2 + t_2 > 1$. We denote

$$F = \{ \omega^R \in \text{BQP}^{m,n} : ACut(\omega^R) = 0 \}.$$

Let $\omega^R \in F$, that is $ACut(\omega^R) = 0$. From (+), we have $ACut(\omega^R) = 0$ if and only if $(p_1 - q_1)(q_2 - p_2 - 1) = \delta$.

Thus, for $P \geq 0$ and $Q \geq 0$, we have

$$(p_1 - q_1)(q_2 - p_2 - 1) = \delta \geq s_1(t_2 - 1).$$

Since $p_1 \leq s_1$ and $q_1 \geq 0$, we have $0 \leq p_1 - q_1 \leq s_1$. Note that $s_1 \neq 0$ since $S_1$ is not empty. Consequently, we have

$$q_2 - p_2 - 1 \geq \frac{p_1 - q_1}{s_1}(q_2 - p_2 - 1) \geq t_2 - 1.$$

It follows that $q_2 - p_2 \geq t_2$. Since $q_2 \leq t_2$, the only possibility is when $q_2 = t_2$ and $p_2 = 0$. Hence, for any $\omega^R \in F$, $y^R_j = 1$ for all $j \in T_2$ and $y^R_j = 0$ for all $j \in S_2$.

For $P < 0$ and $Q < 0$, we consider

$$(q_1 - p_1)(1 + p_2 - q_2) = \delta \geq t_1(s_2 + 1).$$
Since $0 < q_1 - p_1 \leq t_1$, we obtain

$$1 + p_2 - q_2 \geq \frac{q_1 - p_1}{t_1}(1 + p_2 - q_2) \geq s_2 + 1.$$  

It follows that $p_2 - q_2 \geq s_2$. The only possibility is when $p_2 = s_2$ and $q_2 = 0$. Hence, for any $\omega^R \in F$, $y_j^R = 1$ for all $j \in S_2$ and $y_j^R = 0$ for all $j \in T_2$.

Consequently, $\dim(F) \leq m + n + mn - s_2 - t_2 < m + n + mn - 1$. Thus, (4.12) does not define a facet. 

C.3 Examples on Applying Triangular Elimination

C.3.1 Hypermetric Inequalities

The first nontrivial family of valid inequalities for the cut polytope is hypermetric inequalities. Some of these inequalities are also facet-defining for $P_C(K_n)$.

**Definition C.1.** [26] Let $h = (h_1, \ldots, h_T)$ be a vector in $(Z^*)^T$ where $\sum_{i=1}^T h_i = 1$ and $Z^*$ is the set of nonzero integers. Then the hypermetric inequality defined by $h$ is an inequality in the form

$$Q(h)^T w := \sum_{1 \leq i < j \leq T} h_i h_j w_{ij} \leq 0.$$  

A hypermetric inequality is shown to be valid for $CUT(K_T)$. From Proposition 4.52, it is also valid for $P_C(K_T)$. Consequently, its triangular elimination is valid for $P_C(K_{m+n} + u)$. It follows that the covariance image of its triangular elimination is also valid for $BQP^{m,n}$.

Next, we give conditions that make $(G^*, L^*)$ be acceptable for a hypermetric inequality and the closed form of hypermetric inequalities for $BQP^{m,n}$.

**Proposition C.2.** Let $G^*$ be SBTE of $G = K_T$. The structure $(G^*, L^*)$ is acceptable for a hypermetric inequality $Q(h)^T w \leq 0$ if $l \in L^{1,1}$ if and only if $h_pl h_q < 0$ and $l \in L^{2,3}$ if and only if $h_pl h_q > 0$. We say that a structure $(G^*, L^*, h)$ is hypermetric acceptable for short.

Now we can define a hypermetric inequality for $BQP^{m,n}$.

**Definition C.3.** Let $(G^*, L^*, h)$ be a hypermetric acceptable structure. Denote $H_l = |h_pl h_q|$ and

$$C_k(P) = - \sum_{l \in L^{1,3}: k=p_l} P_l + \sum_{l \in L^{2,4}: k=p_l} P_l - \sum_{l \in L^{1,2}: k=q_l} P_l + \sum_{l \in L^{3,4}: k=q_l} P_l,$$  

146
for any index \( k \in V^1 \cup V^2 \) and vector \( P \in \mathbb{R}^{|L^1|} \). Let \( H \) be the vector in \( \mathbb{R}^{|L^1|} \) whose entries are \( H_l \). Then the hypermetric inequality defined by \( h \) is an inequality in the form

\[
ax + by + cz \leq 2 \sum_{l \in L^1} H_l, \tag{C.1}
\]

where

\[
a_i = \begin{cases}
\sum_{j \in V^1 \cup T} h_i h_j + C_i(H) & i \in V^1, \\
-2H_i & i = r_1 \in R^2, l \in L^1, \\
2H_i & i = r_1 \in R^2, l \in L^4, \\
0 & \text{otherwise};
\end{cases}
\]

\[
b_j = \begin{cases}
\sum_{i \in V^1 \cup T} h_i h_j + C_j(H) & j \in V^2, \\
-2H_i & j = r_1 \in R^1, l \in L^1, \\
2H_i & j = r_1 \in R^1, l \in L^4, \\
0 & \text{otherwise};
\end{cases}
\]

\[
c_{ij} = \begin{cases}
-2h_i h_j & i \in V^1, j \in V^2, \\
2H_i & ij = pqr_1, l \in L^{1,3} \text{ or } ij = qrl, l \in L^{3,2}, \\
-2H_i & ij = pqr_1, l \in L^{2,4} \text{ or } ij = qrl, l \in L^{3,4}, \\
0 & \text{otherwise}.
\end{cases}
\]

**Theorem C.4.** Let \((G^*, L^*, h)\) be a hypermetric acceptable structure. Inequality (C.1) is valid for \( \text{BQP}^{m,n} \) where \( m = |V^1| + |R^2| \) and \( n = |V^2| + |R^1| \).

Inequality (C.1) is the covariance image of a triangular elimination of inequality \( Q(h)^T w \leq 0 \) with respect to a STTE of \( G \). Thus, if \( Q(h)^T w \leq 0 \) is facet-defining, by Corollary 4.58, (C.1) also defines a facet for \( \text{BQP}^{m,n} \) and its canonical extension defines a facet for any \( \text{BQP}^{M,N} \) with \( M \geq m \) and \( N \geq n \). An important theorem giving many hypermetric facets for the cut cone corresponding to the complete graph was given by Deza and Laurent [26].

**Theorem C.5.** [Theorem 28.2.4 [26]] Let \( h \in (\mathbb{Z}^*)^{|T|} \) such that \( \sum_{i=1}^T h_i = 1 \) and \( h_1 \geq \ldots \geq h_p > 0 > h_{p+1} \geq \ldots \geq h_T \), where \( 2 \leq p \leq T - 1 \).

(i) If \( p = 2 \), then \( Q(h)^T w \leq 0 \) defines a facet if and only if \( T = 3 \), \( h_1 = h_2 = 1 \) and \( h_3 = -1 \).
(ii) If \(2 \neq p = T - 1\), \(Q(h)^T w \leq 0\) does not define a facet.

(iii) If \(p = T - 2\), then
   
   a) If \(Q(h)^T w \leq 0\) defines a facet, then \(h_1 = 1\).
   
   b) If \(h_{T-1} = -1\), then \(Q(h)^T w \leq 0\) defines a facet if and only if \(h_T = -T + 4\) and \(h_1 = \ldots = h_{T-2} = 1\).

(iv) If \(3 \leq p \leq T - 3\) and \(h_{T-1} = -1\), \(Q(h)^T w \leq 0\) defines a facet if and only if \(h_1 + h_2 \leq T - p - 1 + \text{sign}(|h_1 - h_p|)\) where \(\text{sign}(|h_1 - h_p|) = 1\) if \(h_1 > h_p\) and \(\text{sign}(|h_1 - h_p|) = 0\) if \(h_1 = h_p\).

Since the right hand sides of these inequalities are zero, by Proposition 4.52, they are also facet-defining for \(P_C(K_T)\). Note that from Theorem 4.56, if \(Q(h)^T w \leq 0\) defines a facet for \(P_C(G)\), then its triangular elimination is facet-defining for \(P_C(G')\), where \(G'\) is a STTE of \(G\). It follows that the covariance image of its triangular elimination is also facet-defining for \(BQP^{m,n}\).

Consider part (i) of the theorem. Since Theorem 4.56 requires that the size of the complete graph \(K_T\) must be at least 5, we cannot obtain a facet-defining inequality by using the triangular elimination with the result of (i) with \(T = 3\). However, if we start from a facet-defining inequality \(ax + by + cz \leq 0\) for \(BQP^{m,n}\), we can see that its corresponding inequality \(dw\) for \(P_C(G')\) is facet-defining. By the way, Theorem 4.56 cannot tell whether there is a facet-defining inequality for \(P_C(G)\) whose triangular elimination is \(dw \leq 0\) or not. Thus, we cannot derive anything from part (ii) and (iiia) of the Theorem. The following corollary comes directly from Theorem 4.56 and (iv) of Theorem C.5.

**Corollary C.6.** Let \((G^*, L^*, h)\) be a hypermetric acceptable structure and \(h_1 \geq \ldots \geq h_p > 0 > h_{p+1} \geq \ldots \geq h_T\), where \(3 \leq p \leq T - 3\), \(h_{T-1} = -1\) and \(h_1 + h_2 \leq T - p - 1 + \text{sign}(|h_1 - h_p|)\) where \(\text{sign}(|h_1 - h_p|) = 1\) if \(h_1 > h_p\) and \(\text{sign}(|h_1 - h_p|) = 0\) if \(h_1 = h_p\). Then inequalities in the form (C.1) define facets for \(BQP^{m,n}\) where \(m = |V^1| + |R^2|\) and \(n = |V^2| + |R^1|\).

The next theorem is the result from (iiib) of Theorem C.5. Different choices of the partite sets to which \(T - 1\) and \(T\) belong in the STTE \(G'\) of \(G = K_T\) yield different facet-defining inequalities. Note that by the assumption in Theorem C.5, \(-1 = h_{T-1} \geq h_T = -T + 4\). It follows that \(T \geq 5\).

**Theorem C.7.** Let \((G^*, L^*, h)\) be a hypermetric acceptable structure where \(h\) is a vector of length \(T \geq 5\) such that \(h_1 = \ldots = h_{T-2} = 1\), \(h_{T-1} = -1\), and \(h_T = -T + 4\). Recall the
definition of \(C_i(P)\) and \(C_j(P)\) in Definition C.3. Moreover, we define

\[
C_k(P)' = - \sum_{l \in L^1: k = p_l, T \notin \{p_l, q_l\}} P_l + \sum_{l \in L^2: k = p_l, T \notin \{p_l, q_l\}} P_l - \sum_{l \in L^1: k = q_l, T \notin \{p_l, q_l\}} P_l + \sum_{l \in L^2: k = q_l, T \notin \{p_l, q_l\}} P_l.
\]

We simply write \(C_k\) or \(C_k'\) when \(P\) is a vector whose all entries are equal to 1. For any \(p_l q_l \in F\), denote \(l(p_l) = q_l\) and \(l(q_l) = p_l\). Then a hypermetric inequality defined by \(h\) is in the form

\[
ax + by + cz \leq \sum_{l \in L^1, T \notin \{p_l, q_l\}} 2 + \sum_{l \in L^1, T \in \{p_l, q_l\}} (2T - 8),
\]

where \(T - 1 \in V^0, T \in V^1\) and

\[
a_i = \begin{cases} 
3 - S + C'_i & i \in V^1 \setminus \{T\} \text{ and } \exists l \in L^1, l(T) = i, \\
2T - S - 5 + C'_i & i \in V^1 \setminus \{T\} \text{ and } \exists l \in L^4, l(T) = i, \\
(T - S - 1 - C_T)(4 - T) & i = T, \\
0 & \text{otherwise};
\end{cases}
\]

\[
b_j = \begin{cases} 
S - T + 2 + C_j & j \in V^2, \\
-2T + 8 & j = r_l, l \in L^1, T \in \{p_l, q_l\}, \\
2T - 8 & j = r_l, l \in L^4, T \in \{p_l, q_l\}, \\
0 & \text{otherwise};
\end{cases}
\]

\[
c_{ij} = \begin{cases} 
-2 & ij \in (V^1 \setminus \{T\} : V^2) \text{ or } \\
ij = p_l r_l, l \in L^2, T \notin \{p_l, q_l\} \text{ or } \\
ij = q_l r_l, l \in L^3, T \notin \{p_l, q_l\} , \\
2 & ij = p_l r_l, l \in L^3, T \notin \{p_l, q_l\} \text{ or } \\
ij = q_l r_l, l \in L^2, T \notin \{p_l, q_l\} , \\
2T - 8 & ij \in (T : V^2) \text{ or } \\
ij \in \{p_l r_l, q_l r_l\}, l \in L^1, T \in \{p_l, q_l\}, \\
-2T + 8 & ij \in \{p_l r_l, q_l r_l\}, l \in L^4, T \in \{p_l, q_l\}, \\
0 & \text{otherwise};
\end{cases}
\]

\[
ax + by + cz \leq 2|L^4|,
\]
where $T \in V^0$, $T - 1 \in V^1$ and

\[
a_i = \begin{cases} 
4 - S + C_i & i \in V^1 \setminus \{T - 1\}, \\
S - 4 + C_{T-1} & i = T - 1, \\
0 & \text{otherwise};
\end{cases}
\]

\[
b_j = \begin{cases} 
S - T + 2 + C_j & j \in V^2, \\
-2 & j = r_l, l \in L^1, T - 1 \in \{p_l, q_l\}, \\
2 & j = r_l, l \in L^4, T - 1 \in \{p_l, q_l\}, \\
0 & \text{otherwise};
\end{cases}
\]

\[
c_{ij} = \begin{cases} 
-2 & ij \in (T - 1 : V^2) \text{ or } ij = p_l r_l, l \in L^{2.4} \text{ or } ij = q r_l, l \in L^{3.4}, \\
2 & ij \in (V^1 \setminus \{T - 1\} : V^2) \text{ or } ij = p_l r_l, l \in L^{1.3} \text{ or } ij = q r_l, l \in L^{1.2}, \\
0 & \text{otherwise};
\end{cases}
\]

\[ax + by + cz \leq \sum_{l \in L^1 \setminus \{p_l, q_l\}} 2 + \sum_{l \in L^4 \setminus \{p_l, q_l\}} (2T - 8), \quad (C.4)\]

where $T - 1 \in V^1$, $T \in V^2$ and

\[
a_i = \begin{cases} 
4 - S + C_i & i \in V^1 \setminus \{T - 1\}, \\
S - 4 + C_{T-1} & i = T - 1, \\
-2T + 8 & i = r_l, l \in L^1, T \in \{p_l, q_l\}, \\
2T - 8 & i = r_l, l \in L^4, T \in \{p_l, q_l\}, \\
0 & \text{otherwise};
\end{cases}
\]

\[
b_j = \begin{cases} 
S - T + 3 + C'_i & j \in V^2 \setminus \{T\} \text{ and } \exists l \in L^1, l(T) = j, \\
S + T - 5 + C'_i & j \in V^2 \setminus \{T\} \text{ and } \exists l \in L^4, l(T) = j, \\
(S - 1 - C_T)(4 - T) & j = T, \\
-2 & j = r_l, l \in L^1, T - 1 \in \{p_l, q_l\}, \\
2 & j = r_l, l \in L^4, T - 1 \in \{p_l, q_l\}, \\
0 & \text{otherwise};
\end{cases}
\]
\[ c_{ij} = \begin{cases} 
-2 & i j \in (V^1 \setminus \{T - 1\} : V^2 \setminus \{T\}) \text{ or } \\
& i j = p_i r_l, l \in L^{2,4}, T \notin \{p_i, q_l\} \text{ or } \\
& i j = q_r l, l \in L^{3,4}, T \notin \{p_i, q_l\}, \\
2 & i j \in (T - 1 : V^2 \setminus \{T\}) \text{ or } \\
& i j = p_i r_l, l \in L^{1,1}, T \notin \{p_i, q_l\} \text{ or } \\
& i j = q_r l, l \in L^{1,2}, T \notin \{p_i, q_l\}, \\
2 T - 8 & i j \in (V^1 \setminus \{T - 1\} : T) \text{ or } \\
& i j \in \{p_i r_l, q_r l\}, l \in L^1, T \in \{p_i, q_l\}, \\
-2 T + 8 & i j \in \{T - 1, T\} \text{ or } \\
& i j \in \{p_i r_l, q_r l\}, l \in L^4, T \in \{p_i, q_l\}, \\
0 & \text{otherwise; or } \\
\end{cases} \]

\[ a x + b y + c z \leq \sum_{l \in L^1, T \notin \{p_i, q_l\}} 2 + \sum_{l \in L^4, T \in \{p_i, q_l\}} (2 T - 8), \quad (C.5) \]

where \( T - 1, T \in V^1 \) and

\[ a_i = \begin{cases} 
5 - S + C'_i & i \in V^1 \setminus \{T - 1, T\} \text{ and } \exists l \in L^1, l(T) = i, \\
2 T - S - 3 + C'_i & i \in V^1 \setminus \{T - 1, T\} \text{ and } \exists l \in L^4, l(T) = i, \\
S - 5 + C'_{T-1} & i = T - 1 \text{ and } \\
& (\exists l \in L^2, p_l q_l = [i, T] \text{ or } \exists l \in L^3, p_l q_l = [T, i]), \\
S - 2 T + 3 + C'_{T-1} & i = T - 1 \text{ and } \\
& (\exists l \in L^3, p_l q_l = [i, T] \text{ or } \exists l \in L^2, p_l q_l = [T, i]), \\
(T - S + 1 - C_T)(4 - T) & i = T, \\
0 & \text{otherwise; } \\
\end{cases} \]
\[
\begin{align*}
  b_j = & \begin{cases}
    S - T + 2 + C_j & j \in V^2, \\
    -2 & j = r_l, l \in L^1, T - 1 \notin \{p_l, q_l\}, T \notin \{p_l, q_l\}, \\
    2 & j = r_l, l \in L^4, T - 1 \notin \{p_l, q_l\}, T \notin \{p_l, q_l\}, \\
    -2T + 8 & j = r_l, l \in L^1, T - 1 \notin \{p_l, q_l\}, T \in \{p_l, q_l\}, \\
    2T - 8 & j = r_l, l \in L^4, T - 1 \notin \{p_l, q_l\}, T \in \{p_l, q_l\}, \\
    0 & \text{otherwise};
  \end{cases} \\

c_{ij} = & \begin{cases}
    -2 & ij \in (V^1 \setminus \{T - 1, T\} : V^2) \text{ or} \\
    ij = p_lr_l, l \in L^{2,4}, T \notin \{p_l, q_l\} \text{ or} \\
    ij = q_lr_l, l \in L^{3,4}, T \notin \{p_l, q_l\}, \\
    2 & ij \in (T : V^2) \text{ or} \\
    ij = p_lr_l, l \in L^{1,3}, T \notin \{p_l, q_l\} \text{ or} \\
    ij = q_lr_l, l \in L^{1,2}, T \notin \{p_l, q_l\}, \\
    2T - 8 & ij \in (T : V^2) \text{ or} \\
    ij = p_lr_l, l \in L^{1,3}, T \in \{p_l, q_l\} \text{ or} \\
    ij = q_lr_l, l \in L^{1,2}, T \in \{p_l, q_l\}, \\
    -2T + 8 & ij = p_lr_l, l \in L^{2,4}, T \in \{p_l, q_l\} \text{ or} \\
    ij = q_lr_l, l \in L^{3,4}, T \in \{p_l, q_l\}, \\
    0 & \text{otherwise};
  \end{cases}
\end{align*}
\]

and it defines a facet for \( \text{BQP}^{m,n} \) where \( m = |V^1| + |R^2| \) and \( n = |V^2| + |R^1| \).
C.3.2 Clique-Web Inequality

Another main family of valid inequalities for the cut polytope is clique-web inequality. This inequality can be constructed from a hypermetric inequality by adding the terms associating to a graph structure called an antiweb. The basic subfamily of clique-web inequalities called pure clique-web inequalities was shown to be facet-defining for the cut cone by Deza and Laurent [26]. We can derive a family of facet-defining for BQP \( m,n \) from them. The general form of clique-web inequalities was defined by Deza and Laurent [26] and some of them are facet-defining.

**Definition C.8.** [26] Let \( p,r \in \mathbb{N} \) with \( p \geq 2r+3 \). The antiweb \( AW^r_p \) with parameters \( p \) and \( r \) is the graph \( G(V,E) \) where \( V = \{1,\ldots,p\} \) and \( E = \{ij : j \equiv i + k(\text{mod } p), 1 \leq k \leq r\} \). The web \( W^r_p \) is the complement in the complete graph \( K_p \) of \( AW^r_p \).

**Definition C.9.** [26] Let \( n,p,q,r \in \mathbb{N} \) with \( n = p + q \), \( p - q = 2r + 1 \) and \( q \geq 2 \). Let \( AW^r_p \) be the antiweb with parameters \( r \) and \( p \) whose vertex set is \( \{1,\ldots,p\} \). The pure clique-web Inequality \( CW^r_n(w) \leq 0 \) with parameters \( n,p,q,r \) is the inequality

\[
\sum_{1 \leq i < j \leq n} h_i h_j w_{ij} - \sum_{ij \in E(W^r_p)} w_{ij} \leq 0,
\]

where \( h \) is a vector of length \( n \) where \( h_i = 1 \) for \( i = 1,\ldots,p \) and \( h_i = -1 \) for \( i = p+1,\ldots,n \).

Note that the first \( p \) entries of \( h \) are equal to 1 while the last \( q \) entries of \( h \) are equal to \(-1\). Let \( P \) be the set of the first \( p \) indices and \( Q \) be the set of the first \( q \) indices. Let \( A,B \subseteq V \). Denote by \( AB \) the set of edges \( ij \) where \( i \in A \) and \( j \in B \). Then \( AB = BA \). We use the notation \( E(A) \) for the set of edges whose both endpoints are in \( A \). Using these notations, it is shown by Deza and Laurent [26] that we can write the pure clique-web inequality as

\[
\sum_{ij \in E(W^r_p)} w_{ij} + \sum_{ij \in E(Q)} w_{ij} - \sum_{ij \in PQ} w_{ij} \leq 0.
\]

The pure clique-web inequality is shown to be facet-defining for the cut cone by Deza and Laurent [26]. Thus, by Proposition 4.52, it is also facet-defining for the cut polytope.
To define clique-web inequality for general value of $h$, we use the following operation given by Deza and Laurent [26].

**Definition C.10.** [26] Consider graphs $G^p(V^p, E^p)$ and $G^p(V^p, E^p)$ on $P$ and $p$ vertices, respectively. Let $\pi = (I^1, \ldots, I^p)$ be a partition of $V^p$. For any $v \in \mathbb{R}^{|E^p|}$, the $\pi$-collapse $v^\pi \in \mathbb{R}^{|E^p|}$ is defined by

$$v^\pi_{hk} = \sum_{i \in I^h, j \in I^k} v_{ij}$$

for $hk \in E^p$.

Then we define general clique-web inequality as follow.

**Definition C.11.** [26] Consider the complete graph $G(V, E) = K_T$. We partition $V$ into two partite sets $P$ and $Q$. Label the vertices in $P$ with $1, \ldots, p$ and the vertices in $Q$ with $p + 1, \ldots, T$. Let $h \in (Z^*)^T$ such that $\sum_{i=1}^T h_i = 2r + 1$ for some $r \geq 0$ and

$$h_1, \ldots, h_p > 0 > h_{p+1}, \ldots, h_T.$$  

Denote $P = \sum_{i=1}^p h_i$ and $v \in \{0, 1\}^{P(P-1)/2}$ the characteristic vector of $AW^r_p$, that is for any $ij \in E(K_P)$, $v_{ij} = 1$ if $ij \in E(AW^r_p)$ and $v_{ij} = 0$ otherwise. We define $\pi(h)$ to be the partition $(I^1, \ldots, I^p)$ where $I^1 = \{1, \ldots, h_1\}$ and $I^k = \{h_1 + \ldots + h_{k-1} + 1, \ldots, h_1 + \ldots + h_{k-1} + h_k\}$ for $k = 2, \ldots, p$. Let $v^\pi$ be the $\pi(h)$-collapse of $v$. Then the (general) clique-web inequality $CW^r_T(h)^T w \leq 0$ is

$$\sum_{1 \leq i < j \leq T} h_i h_j w_{ij} - \sum_{ij \in E(AW^r_p)} v^\pi_{ij} w_{ij} \leq 0. \quad (C.6)$$

**Proposition C.12.** Let $G^*$ be SBTE of $G = K_T$. The structure $(G^*, L^*)$ is acceptable for a clique-web inequality $CW^r_T(h)^T w \leq 0$ if and only if for any $l \in L$, $l \in L^{1, 4}$ if and only if $p_{lq_l} \in PQ$ or $(p_{lq_l} \in E(AW^r_p)$ and $h_p h_{q_l} - v_{p_{lq_l}} < 0$) and $l \in L^{2, 3}$ if and only if $p_{lq_l} \in E(Q) \cup E(W^r_p)$ or $(p_{lq_l} \in E(AW^r_p)$ and $h_p h_{q_l} - v_{p_{lq_l}} > 0$). We say that a structure $(G^*, L^*, h)$ is clique-web acceptable for short.
Recall that for any subsets of vertices $A$ and $B$, $(A : B)$ denotes the set of every edge $ij$ in a bipartite graph $G(I,J,E)$ where $i \in I \cap A$ and $j \in J \cap B$. Let $[A : B]$ be the subset of edges in $F$ whose endpoints are $p_l \in A$ and $q_l \in B$. We use these notations in the following theorem.

**Theorem C.13.** Let $(G^*, L^*, h)$ be a clique-web acceptable structure where $h_k = -1$ for all $k \in Q$. Denote $P^k = V^k \cap P$ and $Q^k = V^k \cap Q$ for $k = 1, 2$. We define

$$C_k(E, P_l) = -\sum_{l \in L^{1,3}; k=p_l, p_q \in E} P_l + \sum_{l \in L^{2,4}; k=p_l, p_q \in E} P_l - \sum_{l \in L^{1,2}; k=q_l, p_q \in E} P_l + \sum_{l \in L^{3,4}; k=q_l, p_q \in E} P_l,$$

for any index $k \in V^1 \cup V^2$ and $E \subseteq E^*$. For each $k = r_l \in R$, let

$$r(CW)_k = \begin{cases} 
-2|h_{p_l}h_{q_l} - v_{p_q}| & l \in L^1, p_lq_l \in E(\text{AW}^*_p), \\
2|h_{p_l}h_{q_l} - v_{p_q}| & l \in L^4, p_lq_l \in E(\text{AW}^*_p), \\
-2h_{p_l} & l \in L^1, p_lq_l \in E(W^*_p), \\
2h_{p_l} & l \in L^4, p_lq_l \in E(W^*_p), \\
-2h_{q_l} & l \in L^1, p_lq_l \in [P : Q], \\
2h_{q_l} & l \in L^4, p_lq_l \in [P : Q], \\
-2 & l \in L^1, p_lq_l \in [Q : P], \\
2 & l \in L^4, p_lq_l \in [Q : P], \\
0 & \text{otherwise.} 
\end{cases}$$

Then the following clique-web inequality

$$ax + by + cz \leq 2 \sum_{l \in L^4} e_l,$$  \hspace{1cm} (C.7)

defines a facet for $\text{BQP}^{m,n}$ where $m = |V^1| + |R^2|$ and $n = |V^2| + |R^1|$ if
\[ c_{ij} = \begin{cases} 
2v_{ij} - 2h_i h_j & ij \in E(AW^r_p) \cap P^1 P^2, \\
-2h_i h_j & ij \in E(W^r_p) \cap P^1 P^2 \text{ or} \\
2h_i h_j & ij = pr_i, l \in L^{2,4}, p_i q_i \in E(W^r_p) \text{ or} \\
2h_i & ij = q_i r_i, l \in L^{1,3}, p_i q_i \in E(W^r_p) \\
2h_i & ij = (P : Q), \\
2h_j & ij = (Q : P), \\
-2|hp_i h_{qi} - v_{p_i q_i}| & ij = pr_i, l \in L^{2,4}, p_i q_i \in E(AW^r_p) \text{ or} \\
-2h_{pi} & ij = pr_i, l \in L^{2,4}, p_i q_i \in [P : Q] \text{ or} \\
2h_{pi} & ij = pr_i, l \in L^{1,3}, p_i q_i \in [P : Q] \text{ or} \\
-2h_{qi} & ij = q_i r_i, l \in L^{3,4}, p_i q_i \in [Q : P] \text{ or} \\
2h_{qi} & ij = q_i r_i, l \in L^{1,2}, p_i q_i \in [Q : P] \text{ or} \\
-2 & ij \in E(Q), ij = pr_i, l \in L^{2,4}, p_i q_i \in E(Q) \text{ or} \\
2 & ij = q_i r_i, l \in L^{3,4}, p_i q_i \in E(Q) \text{ or} \\
0 & \text{otherwise}; 
\end{cases} \]

\[ e_l = \begin{cases} 
v_{p_i q_i} - h_{pi} h_{qi} & p_i q_i \in E(AW^r_p), \\
h_{pi} & p_i q_i \in [P : Q], \\
h_{qi} & p_i q_i \in [Q : P]; 
\end{cases} \]
\[
\begin{align*}
\alpha_i &= \begin{cases} 
  h_i h_u + \sum_{ij \in P^1 P^2} h_i h_j - \sum_{ij \in P^1 P^2} v_{ij} & 
  -h_i|Q^2| + C_i(E(AW^r_p), |h_{pq} h_{qi} - v_{pqqi}|) 
  +C_i(E(W^r_p), h_{pq}, h_{qi}) 
  +C_i([Q : P], h_{pq}) + C_i([Q : P], h_{qi}) & 
  \quad i \in P^1 \text{ and } iu \in E(W^r_p), 
  h_i h_u - v_{iu} + \sum_{ij \in P^1 P^2} h_i h_j - \sum_{ij \in P^1 P^2} v_{ij} & 
  -h_i|Q^2| + C_i(E(AW^r_p), |h_{pq} h_{qi} - v_{pqqi}|) 
  +C_i(E(W^r_p), h_{pq}, h_{qi}) 
  +C_i([Q : P], h_{pq}) + C_i([Q : P], h_{qi}) & 
  \quad i \in P^1 \text{ and } iu \in E(AW^r_p), 
  -h_u - \sum_{j \in P^2} h_j + |Q^2| + C_i(E(Q), 1) & 
  +C_i([Q : P], h_{pq}) + C_i([Q : P], h_{qi}) & 
  \quad i \in Q^1, 
  r(CW)_i & 
  i \in R^2; 
\end{cases}
\end{align*}
\]

\[
\begin{align*}
\beta_j &= \begin{cases} 
  h_j h_u + \sum_{ij \in P^1 P^2} h_i h_j - \sum_{ij \in P^1 P^2} v_{ij} & 
  -h_j|Q^1| + C_j(E(AW^r_p), |h_{pq} h_{qi} - v_{pqqi}|) 
  +C_j(E(W^r_p), h_{pq}, h_{qi}) 
  +C_j([Q : P], h_{pq}) + C_j([Q : P], h_{qi}) & 
  \quad j \in P^1 \text{ and } ju \in E(W^r_p), 
  h_j h_u - v_{ju} + \sum_{ij \in P^1 P^2} h_i h_j - \sum_{ij \in P^1 P^2} v_{ij} & 
  -h_j|Q^1| + C_j(E(AW^r_p), |h_{pq} h_{qi} - v_{pqqi}|) 
  +C_j(E(W^r_p), h_{pq}, h_{qi}) 
  +C_j([Q : P], h_{pq}) + C_j([Q : P], h_{qi}) & 
  \quad j \in P^1 \text{ and } ju \in E(AW^r_p), 
  -h_u - \sum_{i \in P^1} h_i + |Q^1| + C_j(E(Q), 1) & 
  +C_j([Q : P], h_{pq}) + C_j([Q : P], h_{qi}) & 
  \quad j \in Q^1, 
  r(CW)_j & 
  j \in R^1; 
\end{cases}
\end{align*}
\]
if $u \in P$ and

$$a_i = \begin{cases} 
-h_i + \sum_{ij \in P^1P^2} h_i h_j - \sum_{ij \in P^1P^2 \cap E(AW'_p)} v_{ij} \\
-h_i |Q^2| + C_i(E(AW'_p), |h_{pq} h_{ql} - v_{pqql}|) \\
+ C_i(E(W'_p), h_{pq} h_{ql}) \\
+ C_i([P : Q], h_{pq}) + C_i([Q : P], h_{ql}) & i \in P^1, \\
1 - \sum_{j \in P^2} h_j + |Q^2| + C_i(E(Q), 1) \\
+ C_i([P : Q], h_{pq}) + C_i([Q : P], h_{ql}) & i \in Q^1, \\
r(CW)_i & i \in R^2; 
\end{cases}$$

if $u \in Q$.

$$b_j = \begin{cases} 
-h_j + \sum_{ij \in P^1P^2} h_i h_j - \sum_{ij \in P^1P^2 \cap E(AW'_p)} v_{ij} \\
-h_j |Q^1| + C_j(E(AW'_p), |h_{pq} h_{ql} - v_{pqql}|) \\
+ C_j(E(W'_p), h_{pq} h_{ql}) \\
+ C_j([P : Q], h_{pq}) + C_j([Q : P], h_{ql}) & j \in P^1, \\
1 - \sum_{i \in P^1} h_i + |Q^1| + C_j(E(Q), 1) \\
+ C_j([P : Q], h_{pq}) + C_j([Q : P], h_{ql}) & i \in Q^1, \\
r(CW)_j & j \in R^1; 
\end{cases}$$