The Pennsylvania State University
The Graduate School
College of Engineering

INTERIOR POINT SCHEMES FOR MIXED-BINARY QUADRATIC PROGRAMS: COMPUTATIONAL INVESTIGATIONS AND APPLICATIONS TO UNIT COMMITMENT PROBLEMS

A Thesis in
Industrial Engineering and Operations Research
by
Wendian Wan

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August 2015

The thesis of Wendian Wan was reviewed and approved* by the following:

Uday V. Shanbhag
Associate Professor of Industrial Engineering
Thesis Advisor

George Kesidis
Professor of Computer Science and Engineering
Thesis Reader

Harriet B. Nembhard
Professor of Industrial Engineering
Department Head of Industrial Engineering

*Signatures are on file in the Graduate School.

Abstract

We consider a class of mixed-binary quadratic programs, a subclass of mixed-integer quadratic programs. When the continuous relaxation of such problems is convex, then a host of algorithms exist for the resolution of such problems, including a range of branching schemes as well as outer-approximation techniques. We consider an alternate approach that relies on a smoothing-based interior-point approach and does not utilize any convexity properties of the relaxation.

Our approach relies on the equivalence between the original discrete optimization and a continuous variant in which the binary restrictions are replaced by a suitably defined penalty function. Inspired by work by Murray and Ng [1], we develop an interior-point scheme that uses an augmented Lagrangian merit function for purposes of globalization. Furthermore, we present a distinct scheme for updating the penalty parameter associated with the integrality residual that relies on either a fixed penalty or an increasing penalty parameter.

Preliminary numerical results on a class of mixed-binary quadratic programs from a class of unit commitment problems with quadratic costs appear to be promising. In particular, it is observed that the scheme obtains global or near-global solutions when the Hessians of the quadratic function are positive semidefinite.

Table of Contents

List of Tables **vi**

Chapter 1
Introduction **1**
1.1 A Review of Algorithms 2
1.1.1 Branch and bound schemes 2
1.1.2 Lagrangian Relaxation 3
1.1.3 Other Algorithms for MINLP 4
1.2 Unit Commitment Problem 6
1.2.1 The unit commitment problem 6
1.3 Motivation, Contributions, and Organization 9

Chapter 2
Description of algorithm **11**
2.1 Interior point methods 11
2.2 Continous Approaches for discrete optimization 15
2.3 Description of Algorithm 17
2.3.1 Introduction of barrier and penalty functions 17
2.3.2 KKT conditions of smoothed problems 18
2.3.3 Newton Direction 19
2.3.4 Linesearch 20

Chapter 3
Numerical investigations **23**
3.1 Fixed penalty parameter 23
3.2 Increasing penalty parameter 24

Chapter 4
Conclusion **26**

Bibliography **27**

List of Tables

3.1 Fixed Penalty $\rho = 999999$ 24
3.2 Increasing Penalty with $\rho_0 = 1$ 25
3.3 Increasing Penalty with $\rho_0 = 99999$ 25

Chapter 1 Introduction

In this thesis, we consider the class of mixed-binary quadratic programs of the following form:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Q x + c^T x \\ \text{subject to} \quad & Ax \;=\; b, \\ & x \;\geq\; 0, x_i \in \{0,1\} \text{ for } i \;\in\; \mathcal{I}, \end{aligned} \tag{MBQP}$$

where $Q \in \mathbb{R}^{n \times n}, c \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, x \in \mathbb{R}^n$, and $|\mathcal{I}| \subseteq \{1, 2, \ldots, n\}$. Naturally, when $|\mathcal{I}| = 0$, this problem reduces to a continuous quadratic program for which a broad class of techniques can be employed, based on whether Q is a positive semidefintite matrix or not. We refer to (MBQP) as a mixed-binary quadratic program and represents a subclass of mixed-integer quadratic programs, in which a subset of decision variables may take on integral, rather than binary, values. An important special case of such problems is mixed-binary linear programs when $Q \equiv 0$. Much of the literature in the area of discrete optimization applies to the more general class of mixed-integer programs and we provide a brief review of these techniques.

1.1 A Review of Algorithms

We summarize the main schemes that have found applicability in the resolution of mixed-integer programs.

1.1.1 Branch and bound schemes

One of the most successful approaches for solving mixed-integer programming problems is the branch-and-bound scheme [6]. In [2], a more general survey and discussion of the application of branch-and-bound was provided. An application to nonlinear integer programming with convex relaxations was introduced in [3–5]. The branch-and-cut variant [6–9] increased flexibility by adding polyhedral cutting planes by introducing a new cut after branching so that it may use more information of fathomed nodes. We discuss the scheme in the context of (MBQP) where Q is assumed to be positive semidefinite for the present. In order to start the branch-and-bound algorithm, first we relax all integrality constraints in (MBQP) and get a relaxed problem (MBQP_0).

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Q x + c^T x \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0, \text{ for } i \in \mathcal{I}. \end{aligned} \tag{MBQP_0}$$

Since (MBQP_0) is a relaxation with x^0 as an optimal solution, the optimal value of (MBQP_0) is a lower bound of the optimal value of (MBQP). Further, if x^0 satisfies all integrality requirements in (MBQP), it is a feasible solution to (MBQP) and also the optimal solution. Otherwise, there must be a variable $x_j \in x^0$ which is not integer. Then we may form two subproblems from (MBQP_0) by adding bound $x_j \leq [x_j]$ to one and $x_j \geq [x_j + 1]$ to the other, where $[x_j]$ represent the largest integer not greater than x_j. The subproblem (MBQP_1) is as shown below:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Q x + c^T x \\ \text{subject to} \quad & Ax = b, \\ & x_j \leq [x_j], \\ & x \geq 0, \text{ for } i \in \mathcal{I}. \end{aligned} \tag{MBQP_1}$$

The subproblem (MBQP_2) may be defined as follows:

$$\begin{aligned} \min \quad & \frac{1}{2}x^TQx + c^Tx \\ \text{subject to} \quad & Ax \ = \ b, \\ & x_j \ \geq \ [x_j + 1], \\ & x \ \geq \ 0, \ \text{for } i \ \in \ \mathcal{I}. \end{aligned} \tag{MBQP_2}$$

The process of forming subproblems is referred to as branching. The convex programming subproblems are continuous problems and may be solved via standard schemes and the process may be repeated. The entire algorithm continues as a tree search is performed with each node representing a continuous subproblem. In most cases, it is not necessary to search the entire tree. Once we obtain a feasible integer solution to one of the continuous problems, the corresponding value of the objective function is an upper bound to the original problem. The subproblems with continuous solutions with objective value higher than the upper bound may be excluded from further consideration. This node is then fathomed. In addition, if it is recognized that a node leads to infeasibility, then it will also be fathomed since further branching from that node cannot lead to feasibility. The branching and fathoming continues as the upper bound keeps reducing while the lower bounds keep increasing through the addition of constraints. Thus, the gap between upper bound and lower bound keeps decreasing and scheme can be terminated when this gap is sufficiently small.

1.1.2 Lagrangian Relaxation

Another popular method for solving mixed-integer programming problems is the Lagrangian relaxation scheme [10]. In 1970 [11, 12], a Lagrangian technique based on minimum spanning tree was used to devise a algorithm for the traveling salesman problem. In [13, 14], such techniques were applied to mixed-integer programming. More details about Lagrangian relaxation in integer programming is provided in [15]. Consider a general optimization problem given below:

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & g(x) \ = \ b. \end{aligned} \tag{P}$$

We may introduce the Lagrange multiplier λ corresponding to the constraint $g(x) = b$ and define the Lagrangian function as below:

$$L(x, \lambda) \triangleq f(x) + \lambda^T (g(x) - b)$$

Thus, the basic Lagrangian relaxation for (MBQP) may be stated as below:

$$\begin{aligned} L(u) = \min \quad & \frac{1}{2} x^T Q x + c^T x + u^T (Ax - b) \\ \text{subject to} \quad & x_j \;\leq\; \lceil x_j \rceil, \\ & x \;\geq\; 0, \;\text{for } i \;\in\; \mathcal{I}, \end{aligned} \tag{LR(u)}$$

where u denotes the Lagrange multiplier. In many cases, the relaxation (LR_u) may be easier to solve compared to (P). Suppose we denote z as the origin optimal objective value and x^* as the optimal solution to the original problem. Then we may show that $L(u) \leq \frac{1}{2}(x^*)^T Q x^* + c^T x^* + u^T (Ax^* - b) = z$. The solution to the Lagrangian subproblem allows for updating the multiplier estimate u which in turn will provide a new primal solution. In the limit, it may be shown that the sequence of primal solutions is the solution to (P) while the sequence of dual solutions converges to the true Lagrange multiplier. When the primal problem has integrality constraints, then this relaxation allows for decomposition of this problem into smaller (and possibly) structured problems that are amenable to faster solutions.

1.1.3 Other Algorithms for MINLP

Consider a general MINLP problem of the following form:

$$\begin{aligned} Z \;= \min \quad & c^T y + f(x) \\ \text{subject to} \quad & By + g(x) \;\leq\; 0, \\ & x \;\in\; \mathcal{X}, \; y \;\in\; \{0, 1\}, \end{aligned} \tag{MINLP}$$

where $\mathcal{X}$ is a polyhedral set and f and g are convex functions in their arguments. The generalized Benders decomposition in [16] algorithm divides variables into sets of complicating and noncomplicating variables. Using a sequence of nonlinear programming (NLP) subproblems and mixed-integer linear programming (MILP) master problems to solve the original MINLP. The master problem, a MILP, can be stated as below:

$$
\begin{aligned}
z = \min \quad & \alpha \\
\text{subject to} \quad & \alpha \;\geq\; c^T y + f(x^k) + (\lambda^k)^T[By + g(x^k)], \qquad k \;=\; 1, \ldots, K_{\text{feasible}}, \\
& (\lambda^k)^T[By + g(x^k)] \;\leq\; 0, \qquad k \;=\; 1, \ldots, K_{\text{infeasible}}, \\
& x \in \mathcal{X}, \;\; y \;\in\; \{0,1\},
\end{aligned}
$$

where z denotes the lower bound, (x^k, λ^k) are the optimal primal and dual variables of the NLP subproblems, and $K_{\text{feasible}}, K_{\text{infeasible}}$ refer to feasible and infeasible subproblems. The solution of the master problem provides a new set of binary variables that are parameters in the NLP subproblems. The NLP subproblems provide a decreasing sequence of upper bounds while the master problems provides an increasing sequence of lower bounds. When these bounds lie within a suitable tolerance, the scheme terminates. In [17–19] an outer approximation scheme for solving MINLP is presented. Similarly, the outer approximation algorithm also includes a sequence of NLPs while the master problems is defined below:

$$
\begin{aligned}
z = \min \quad & \alpha \\
\text{subject to} \quad & \alpha \;\geq\; c^T y + f(x^k) + \nabla f(x^k)^T[x - x^k], \;\; k \;=\; 1, ..., K, \\
& By + g(x^k) + \nabla g(x^k)^T(x - x^k) \;\leq\; 0, \;\; k \;=\; 1, ..., K, \\
& y \;\in\; \{0,1\}, \;\; x \;\in\; \mathcal{X}.
\end{aligned}
$$

where x^k is the solution to a fixed y^k of the NLP subproblem. In [20], an extended cutting plane method for solving convex MINLP problems was introduced. Consider a MINLP given as follows:

$$
\begin{aligned}
\min \; & f(x, y) \\
\text{subject to} \quad & g(x, y) \;\leq\; 0, \\
& x \;\in\; \mathcal{R}, \;\; y \;\in\; \{0,1\},
\end{aligned}
$$

where $g(x, y)$ is a vector of continuous convex functions. Then

$$
g_i(x^k, y^k) + \nabla_x g_{i,k}^T(x - x^k) + \nabla_y g_{i,k}^T(y - y^k) \;\leq\; g_i(x, y) \;\leq\; 0,
$$

For any feasible point (x^k, y^k), we define a linearization as follows:

$$l_k(x,y) \ = \ f_k(x^k,y^k) + \nabla_x g_{i,k}^T(x-x^k) + \nabla_y g_{i,k}^T(y-y^k).$$

The scheme requires adding a sequence of such linearizations and the resulting sequence of lower bounds keeps increasing. By utilizing the decreasing nature of the upper bounds, the gap is guaranteed to fall within a threshold.

1.2 Unit Commitment Problem

The unit commitment problem considers the determination of the optimal production schedule of power generating units, so that in a certain amount of time the operational cost may be minimized while meeting demand requirements and physical constraints. Basically, binary variables represent the status of unit. An overview of unit commitment problem in literature was provided in [21]. In [22,23], branch and bound schemes were used to solve the unit commitment problem while the Lagrangian relaxation is also widely used in solving the unit commitment problem [24–28].

1.2.1 The unit commitment problem

In an unit commitment problem, we consider the scheduling of power production over a certain period of time. Define the planning horizon as T, and each time step $t \in \mathcal{T} = 1, \ldots, T$. The generators we hold is set $\mathcal{I} = 1, ..., I$. For any generator, once it is switched on, it cannot be turned off immediately. Similarly, it cannot be turn on right after it is switched off. Each generator must follow a minimum up and down times rule. As stated in [29, 30], we can define decision variable as following. For every time step $t \in \mathcal{T}$ and generator $i \in \mathcal{I}$, we can denote the generator state (on or off), startup, and production decisions by y_{it}, z_{it}, and $g_{i,t}$ respectively. In a two-stage decision process, we note that y_{it} and z_{it} are first stage decisions which are made before actual production. While g_{it} stand for production decisions we made.

Objective function The objective of unit commitment problem is to minimize the operation cost. Every time a generator is turned on, a startup cost need to be considered due to the fuel and electrical power consumed. We denote the startup cost by f_{it}^y as determined by y_{it}. During the running time of a generator, a running cost needs to be considered. We denote the running cost by f_{it}^z which is determined by z_{it}. The cost of generation may be presented as $\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f_{it}^g$ which is determined by g_{it}. The objective function is defined as follows:

$$f(y,z,g) = \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f_{it}^y(y_{it}) + \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f_{it}^z(z_{it}) + \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f_{it}^g(g_{it}). \tag{1.1}$$

Startup and shutdown constraints We adopt the linear formulation of [29, 30]. We observe that the startup decision y and the operational decision z are related. The variable $y_{i,t+1}$ will be 1 only when a unit is off at time t but on at time $t+1$. Also we have $y_{it} \geq 0$, we can represent it as the following constraint

$$y_{it} \geq z_{it} - z_{i,t-1}, y_{it} \geq 0. \tag{1.2}$$

Each unit has a minimum up and down time constraint and denote these times by L_i and l_i corresponding to generator $i \in \mathcal{I}$. Then the constraint can be written as:

$$\begin{aligned} z_{it} - z_{i,t-1} \leq z_{i,\gamma}, 2 \leq t \leq T, \forall\gamma \in \{t+1, ..., \min(t+L_i-1, T)\}, \forall i \in \mathcal{I}, \\ z_{i,t-1} - z_{it} \leq 1 - z_{i,\gamma}, 2 \leq t \leq T, \forall\gamma \in \{t+1, ..., \min(t+l_i-1, T)\}, \forall i \in \mathcal{I}. \end{aligned} \tag{1.3}$$

Once generator i has been switched on at time t, the unit continues to be on for at least $L_i - 1$ time units. Using $z_{it} - z_{i,t-1}$ to represent y_{it} we get the above constraint. The down time constraint goes the same.

Generation constraints For each generator, there must be a minimum and a maximum generations bound. Let Q_i and q_i be the maximum and minimum generation level for generator i. We have the following constraints:

$$q_i z_{it} \leq g_{it} \leq Q_i z_{it},\ \forall i \in \mathcal{I},\ \forall t \in \mathcal{T}. \tag{1.4}$$

Meanwhile, the predicted demand need to be satisfied.

$$\sum_{i\in\mathcal{I}} g_{it} \geq d_t,\ \forall t \in \mathcal{T}. \tag{1.5}$$

General Form By combining objective function (1.1) and constraints (1.2) - (1.5), we can get the following model of unit commitment problem:

$$\begin{aligned}
\min \quad & f(y,z,g) = \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f^y_{it}(y_{it}) + \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f^z_{it}(z_{it}) + \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} f^g_{it}(g_{it}) \\
\text{subject to} \quad & y_{it} \geq z_{it} - z_{i,t-1}, \\
& z_{it} - z_{i,t-1} \leq z_{i,\gamma}, 2 \leq t \leq T, \gamma \in \{t+1, ..., \min(t+L_i-1, T)\}, i \in \mathcal{I}, \\
& z_{i,t-1} - z_{it} \leq 1 - z_{i,\gamma}, 2 \leq t \leq T, \gamma \in \{t+1, ..., \min(t+L_i-1, T)\}, i \in \mathcal{I}, \\
& q_i z_{it} \leq g_{it} \leq Q_i z_{it}, i \in \mathcal{I}, t \in \mathcal{T}, \\
& \sum_{i\in\mathcal{I}} g_{it} \geq d_t, t \in \mathcal{T}, \\
& g_{it} \geq 0, t \in \mathcal{T}, i \in \mathcal{I}, \\
& z_{it}, y_{it} \in \{0,1\}, t \in \mathcal{T}, i \in \mathcal{I}.
\end{aligned}$$

Matrix Form Under settings with quadratic objectives, the general form can be represented as a stochastic mixed integer linear programming problem as the follows:

$$\begin{aligned}
\min \quad & \tfrac{1}{2} x_b^T Q x_b + c_b^T x_b + c_g^T g \\
\text{subject to} \quad & A_b x_b + Bg \leq 0, \\
& Dg \leq u, \\
& E x_b \leq 0, \\
& x_b \in \{0,1\},
\end{aligned} \tag{1.6}$$

where

$$x_b = \begin{pmatrix} y \\ z \end{pmatrix}, z = \begin{pmatrix} z_{11} \\ \vdots \\ z_{1T} \\ \vdots \\ z_{IT} \end{pmatrix}, A_b = \begin{pmatrix} 0 & q_1 I \\ \vdots & \vdots \\ 0 & q_I I \\ 0 & -Q_1 I \\ \vdots & \vdots \\ 0 & -Q_1 I \end{pmatrix},$$

$$B^\omega = \begin{pmatrix} -I & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & -I \\ I & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & I \end{pmatrix}, E = \begin{pmatrix} E_1 \\ E_2 \\ -E_2 \end{pmatrix},$$

$$E_1 = \begin{pmatrix} -1 & 1 & \cdots & 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -1 & 1 & 0 & 0 & \cdots & -1 \end{pmatrix}.$$

Adding slack variables to (1.6), we obtain the following:

$$\begin{aligned} \min \quad & \tfrac{1}{2}x^T Q x + c^T x \\ \text{subject to} \quad & Mx + s = q, \\ & x_i \in \{0,1\}, i \in \{0,1\}. \end{aligned} \tag{1.7}$$

where

$$x = \begin{pmatrix} x_b \\ g \end{pmatrix}, c = \begin{pmatrix} c_b \\ c_g \end{pmatrix}, M = \begin{pmatrix} A & B^1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A & 0 & \cdots & B^n \\ 0 & D^1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D^n \\ E & 0 & \cdots & 0 \end{pmatrix}, q = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ u^1 \\ \vdots \\ u^n \\ 0 \end{pmatrix}.$$

1.3 Motivation, Contributions, and Organization

One of the challenges associated with standard techniques for resolving mixed-integer is the lack of scalability as problem size grows. Furthermore, such techniques do not allow for decomposability in the context of structured discrete optimization problems (such as in the context of two-stage stochastic mixed-integer programs). Our focus lies on examining the computational behavior of recently investigated techniques for mixed-binary programs. In particular, this thesis does the following:

(i) A C++ implementation of a linesearch-based interior point scheme is presented for computing feasible solutions of (MBQP). In addition, we consider the addition of objective-value cuts to improve the peformance of the scheme;

(ii) We examine the behavior of the scheme on a class of mixed-binary quadratic programs as well as a class of unit commitment problems.

The remainder of this thesis is organized into three chapters. In Chapter 2, we describe the structure of the interior-point scheme while in Chapter 3, we discuss the performance of these schemes on a class of mixed-integer quadratic programs. The thesis concludes with Chapter 4.

Chapter 2 Description of algorithm

We propose a scheme based on that proposed by Murray and Ng [1] (also see [31]). The scheme has its roots in the two distinct threads of research, the first being interior-point schemes and the second being continuous approaches to solve discrete optimization problems. In Section 2.1, we introduce interior point methods and outline continuous approaches for discrete optimization problems in Section 2.2. We present our algorithmic scheme in Section 2.3.

2.1 Interior point methods

In 1955, the notion of interior-point methods was proposed in [32]. The global convergence of such methods was proved in [33]. In 1984, a polynomial-time algorithm was presented in [34]. In [36], primal-dual schemes were presented. In the context of nonconvex programming, there is also significant work on interior point methods. In [37–39], the monotone nonlinear complementary problem was discussed. The linearly constrained nonlinear programming problem was mentioned in [37]. Various interior point method formulations were pointed out for the general nonlinear programming problem in [41]. In [42], an globally convergent algorithm and theory was provided. In [43–46], interior point method was also presented in the context of nonlinear programming problems. In [47] a more detailed interior point scheme for quadratic programming problems was discussed. For general nonlinear programming using interior point method, an algorithm and theory is mentioned in [48]. Consider a general nonlinear programming problem as below:

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & h(x) = 0, \\ & g(x) \geq 0, \end{aligned}$$

The Lagrangian function would be

$$L(x, y, z) = f(x) + y^T h(x) - z^T g(x).$$

The KKT conditions for the problem could be

$$\begin{aligned} \nabla_x L(x, y, z) &= 0, \\ h(x) &= 0, \\ g(x) &\geq 0, \\ Zg(x) &= 0, \\ z &\geq 0. \end{aligned}$$

By adding slack variables we can have this form:

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & h(x) = 0, \\ & g(x) - s = 0, \\ & s \geq 0. \end{aligned}$$

We have the KKT condition as below:

$$\begin{aligned} \nabla f(x) + \nabla h(x) y - \nabla g(x) w &= 0, \\ w - z &= 0, \\ h(x) &= 0, \\ g(x) - s &= 0, \\ ZSe &= 0, \\ (s, z) &\geq 0. \end{aligned}$$

Based on the KKT conditions above, the algorithm is using Newton's method to solve the linear system below:

$$F_\mu(x,y,s,w,z) \;=\; \begin{pmatrix} \nabla f(x) + \nabla h(x)y - \nabla g(x)w \\ w - z \\ h(x) \\ g(x) - s \\ ZSe - \mu e \end{pmatrix} \;=\; 0,\;\; (s,w,z) \;\geq\; 0.$$

At k^{th} iteration, let $v_k = (x_k, y_k, s_k, w_k, z_k)$. We can get the Newton step $\Delta v_k = (\Delta x_k, \Delta y_k, \Delta s_k, \Delta w_k, \Delta z_k)$corresponding to the parameter μ_k which is the solution to the Newton system

$$F'_\mu(v_k) \;=\; -F_\mu(v_k).$$

Then we can choose certain step length $\alpha_x, \alpha_y, \alpha_s, \alpha_w, \alpha_z$ to update v_k as

$$v_{k+1} \;=\; v_k + \Lambda_k \Delta v.$$

First compute the quantities

$$\begin{aligned} \hat{\alpha}_s &= \frac{-1}{\min((S_k)^-1\Delta s_k, -1)}, \\ \hat{\alpha}_w &= \frac{-1}{\min((W_k)^-1\Delta w_k, -1)}, \\ \hat{\alpha}_z &= \frac{-1}{\min((Z_k)^-1\Delta z_k, -1)}. \end{aligned}$$

Then choose a $\gamma_k \;\in\; (0,1]$ and $\alpha_p \;\in\; (0,1]$, such that

$$\phi(v_k + \Lambda_k \Delta v) \;\leq\; \phi(v_k) + \beta \alpha_p \nabla \phi(v_k)^T \Delta v_k$$

For some fixed $\beta \in (0,1)$ and for Λ_k there are

$$\begin{aligned}
\alpha_x &= \alpha_p, \\
\alpha_y &= \alpha_p, \\
\alpha_s &= \min(1, \gamma_k \hat{\alpha}_s), \\
\alpha_w &= \min(1, \gamma_k \hat{\alpha}_w), \\
\alpha_z &= \min(1, \gamma_k \hat{\alpha}_z),
\end{aligned}$$

Define $\mu_k = \sigma_k \min(S_k Z_k e)$. As it is proved in [48], a sequence of $\{v_k\}$ generated by the Newton step above could converge to a solution v^*, if

1. $\gamma_k \to 1$ and $\sigma_k \to 0$, then the sequence $\{v_k\}$ converges to v^* Q-superlinearly.

2. $\gamma_k = 1 + O(\|F(v_k)\|)$ and $\sigma_k = O(\|F(v_k)\|)$, then the sequence $\{v_k\}$ converges to v^* Q-quadratically.

Meanwhile, a globally convergent algorithm was also mentioned in [48]. Based on the algorithm above, a particular choice of the merit function ϕ was considered. For a slack-variable form of KKT conditions are as follows:

$$F(x,y,s,z) = \begin{pmatrix} G(x,y,s,z) \\ ZSe \end{pmatrix} = 0, \ (s,z) \geq 0.$$

where

$$G(x,y,s,z) = \begin{pmatrix} \nabla_x L(x,y,s,z) \\ h(x) \\ g(x) - s \end{pmatrix} = 0.$$

For a given starting point $v_0 = (x_0, y_0, z_0, s_0)$ with $(s_0, z_0) > 0$, let

$$r_1 = \min(Z_0 S_0 e)/[(z_0)^T s_0/p], \ r_2 = (z_0)^T s_0/\|G(v_0)\|_2.$$

Define

$$\begin{aligned}
f^I(\alpha) &= \min(Z(\alpha)s(\alpha)) - \gamma r_1 z(\alpha)^T s(\alpha)/p, \\
f^{II}(\alpha) &= z(\alpha)^T s(\alpha) - \gamma r_2 \|G(v(\alpha))\|_2.
\end{aligned}$$

where $\gamma \in (0, 1)$ is a constant. For $i = I, II$, define

$$\alpha^i = \max_{\alpha \in [0,1]} \{\alpha : f^i(\alpha) \geq 0, \forall \alpha' \geq \alpha\}.$$

The merit function used for linesearch is the squared l_2 norm of the residual, i.e.

$$\phi(v) = \|F(v)\|_2^2.$$

Define ϕ_k as the value of $\phi(v_k)$ and $\phi_k(\alpha)$ as the value of $\phi(v_k + \alpha\Delta_k)$ and $\phi_k = \phi_k(0) = \phi(v_k)$. For the step length selection, there are two step:

1. Compute $\alpha^i, i = I, II$ and let

$$\alpha_k = \min(\alpha^I, \alpha^{II}).$$

2. Update $\alpha_k = \rho^t \alpha_k$ until α_k satisfies

$$\phi_k(\alpha_k) \leq \phi_k(0) + \alpha_k \beta \phi_k'(0).$$

where $\rho \in (0, 1)$ and $\beta \in (0, \frac{1}{2}]$. By doing so we can always keep the Newton direction is a descent direction for the merit function ϕ, $\{\|F(v_k)\|\}$ is monotone decreasing and converge to 0. Under global convergent algorithm, $\{F(v_k)\}$ were proved to converge to zero for any limit point $v^* = (x^*, y^*, z^*, s^*)$ of v_k, and x^* is a KKT point. More details including the convergence of this Newton interior point method for general nonlinear programming problem are provided in [48].

2.2 Continous Approaches for discrete optimization

Apart from standard discrete optimization approaches, continuous approaches have also been used. In [49], the authors introduced the exact penalized formulation where a MBQP can be reduced to a concave minimization problem with same set of global minimizers:

$$\begin{aligned} \min \quad & f(x) \;=\; c^T x + \mu x^T(e-x) \\ \text{subject to} \quad & Ax \;\leq\; b, 0 \;\leq\; x \;\leq\; e \\ & x \;\in\; \mathcal{R}, \end{aligned}$$

where μ is a sufficient large positive number. The concave function reaches its minimum at a vertex. For a large enough μ, at a global minimum $x^T(e-x) = 0$. In fact [49], the original discrete problem has the same set of global minimizers as its continuous counterpart. In [50], an exact penalty-based global optimization approach for mixed-integer programming problems was introduced. Based on [49], a penalty term $\varphi(x,\xi)$ is added. In [50], we first compute a x^k such that

$$f(x^k) + \varphi(x^k,\xi^k) \;\leq\; f(x) + \varphi(x,\xi^k) + \delta^k.$$

Then if x^k is not a feasible solution to original problem and for $z^k = [x^k]$, we have that

$$\varphi(x^k,\xi^k) - \varphi(z^k,\xi^k) \;\leq\; f(z^k) - f(x^k) + \xi^k \|x^k - z^k\|^\alpha.$$

Then we update the parameter ξ such that $\xi^{k+1} = \sigma\xi^k$ and δ remains the same. The exact penalty global optimization approach may be used for obtaining global solutions to mixed-integer programming problems. In [1], instead of adding non-linear constraints, they also add a penalty term $x^T(e-x)$ to the objective. In addition, they also consider a smoothing method, which is based on modifying the objective function as follows:

$$F(x,\mu) \;=\; f(x) + \mu\Phi(x).$$

For problems with $0 \leq x \leq e$ constraints, barrier functions are employed of the form:

$$\Phi(x) \;=\; -\sum_{j=1}^{n} \ln x_j - \sum_{j=1}^{n} \ln(1-x_j).$$

For a mixed binary programming problem, the objective function with penalty and smoothing term is given by the following:

$$f(x) - \mu \sum_{j=1}^{n} [\ln x_j + \ln(1 - x_j)] + \rho \sum_{j \in \mathcal{J}} x_j(1 - x_j).$$

The new model may be defined as follows:

$$\begin{aligned} \min \quad & f(x) - \mu \sum_{j=1}^{n} [\ln x_j + \ln(1 - x_j)] + \rho \sum_{j \in \mathcal{J}} x_j(1 - x_j) \\ \text{subject to} \quad & Ax \ = \ b, \\ & 0 \ \leq \ x \ \leq \ e. \end{aligned}$$

As $\rho \to \infty$ and $\mu \to 0$, for $j \in \mathcal{J}$, x_j converges to 0 or 1.

2.3 Description of Algorithm

Following the idea of [1, 49, 50], we add a penalty term to replace the binary constraints and use a line-search based interior point method [48] to solve the smoothed quadratic programming problem with penalty term. Once a solution to this problem is obtained, we round the integer variables to meet the integrality requirements and fix these variables prior to solving the continuous QP to get the final feasible integer solution. Here are the details of our algorithm.

2.3.1 Introduction of barrier and penalty functions

Consider the solution of the convex quadratic program given by

$$\begin{aligned} \min \quad & \tfrac{1}{2} x^T Q x + c^T x \\ \text{subject to} \quad & Ax \ = \ b, \\ & x \ \geq \ 0. \end{aligned} \tag{QP}$$

Given the problem (QP), we may use barrier functions to articulate a related equality-constrained QP [1]:

$$\begin{aligned} \min \quad & \tfrac{1}{2}x^TQx + c^Tx - \mu\sum_{i=1}^{N} x_i \\ \text{subject to} \quad & Ax \;=\; b. \end{aligned} \tag{BQP}$$

If some of the decision variables are binary variables, then we define those variables separately. We consider the following mixed-binary quadratic program:

$$\begin{aligned} \min \quad & \tfrac{1}{2}x^TQx + c^Tx + \tfrac{1}{2}y^TRy + d^Ty \\ \text{subject to} \quad & Ax + By \;=\; b \\ & x \;\geq\; 0,\; y \;\in\; \{0,1\}^p. \end{aligned} \tag{MQP}$$

We now consider a related equality constrained problem given by the following:

$$\begin{aligned} \min \quad & \tfrac{1}{2}x^TQx + c^Tx + \tfrac{1}{2}y^TRy + d^Ty - \mu\sum_{i=1}^{n} \ln x_i \\ & - \eta\sum_{j=1}^{p} \ln[y_j + \ln(1-y_j)] + \rho y^T(e-y) \\ \text{subject to} \quad & Ax + By \;=\; b. \end{aligned} \tag{MBQP}$$

2.3.2 KKT conditions of smoothed problems

We may now derive the first-order conditions of this problem:

$$\begin{aligned} Qx + c - \mu X^{-1}e - A^T\lambda \;&=\; 0 \\ Ax - b \;&=\; 0. \end{aligned} \tag{KKT-BQP}$$

We may now define a variable z such that $Xz = \mu e$ giving us the following equivalent system:

$$\begin{aligned} Qx + c - z - A^T\lambda \;&=\; 0 \\ Ax - b \;&=\; 0 \\ -\mu e + Xz \;&=\; 0. \end{aligned} \tag{EKKT-BQP}$$

Similarly, we can have the first-order KKT conditions of (MBQP)

$$\begin{aligned} Qx + c - \mu X^{-1}e - A^T\lambda &= 0 \\ Ry + d - \eta Y^{-1}e + \eta(I-Y)^{-1}e + \rho(e-2y) - B^T\lambda &= 0 \\ Ax + By - b &= 0 \end{aligned} \qquad \text{(KKT-MBQP)}$$

We will now add variables z, v and w, such that:

$$\begin{aligned} Xz &= \mu e, \\ Yv &= \eta e, \\ (I-Y)w &= \eta e. \end{aligned}$$

We may now state the system with the extra variables as follows:

$$\begin{aligned} Qx + c - z - A^T\lambda &= 0 \\ Ry + d - v + w + \rho(e-2y) - B^T\lambda &= 0 \\ Ax + By - b &= 0 \\ -\mu e + Xz &= 0 \\ -\eta e + Yv &= 0 \\ -\eta e + (I-Y)w &= 0 \end{aligned} \qquad \text{(EKKT-MBQP)}$$

2.3.3 Newton Direction

We can use Newton's method to find the solution of KKT conditions. For QP, the linearized KKT conditions allow for constructing a step:

$$\begin{pmatrix} Q & -A^T & -I \\ A & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \delta x \\ \delta\lambda \\ \delta z \end{pmatrix} = - \begin{pmatrix} Qx + c - z - A^T\lambda \\ Ax - b \\ -\mu e + Xz \end{pmatrix}.$$

Thus, based on Newton's method we can have the following interior point algorithm: Similarly, we may use the same procedure to compute the Newton direction.

Algorithm 1 Interior point scheme

initialization: $k = 0,\ x_0, \lambda_0, z_0, \mu_0, \sigma$;
while $\|$`kkt-resid`$\| > \epsilon$; Continue **do**
  Compute $\delta x_k, \delta\lambda_k, \delta z_k$ based on (QPStep);
  Select α so that $x_k + \alpha\delta x_k \geq 0$ and $z_k + \alpha\delta z_k \geq 0$;
  Set $\alpha = 0.99\alpha$ and update x_k, y_k, z_k as $x_k + \alpha\delta x_k, y_k + \alpha\delta y_k, z_k + \alpha\delta z_k$;
  $k = k+1, \mu_k = \mu_k\sigma$;
end while
return x_k;

We define the following matrix H:

$$H = \begin{pmatrix} Q & & -A^T & -I & & \\ & R - 2\rho I & -B^T & & -I & I \\ A & B & & & & \\ Z & & & X & & \\ & V & & & Y & \\ & W & & & & I - Y \end{pmatrix}.$$

We may now arrive at the step:

$$H \begin{pmatrix} \delta x \\ \delta y \\ \delta\lambda \\ \delta z \\ \delta v \\ \delta w \end{pmatrix} = - \begin{pmatrix} Qx + c - z - A^T\lambda \\ d - v + w + \rho(e - 2y) - B^T\lambda \\ Ax + By - b \\ -\mu e + Xz \\ -\eta e + Yv \\ -\eta e + (I - Y)w \end{pmatrix},$$

2.3.4 Linesearch

Due to the nonconvexity of MBQP, we utilize a line search to ensure global convergence. This requires defining a merit function for determining the steplength [48,

Algorithm 2 Interior point scheme for MBQP

initialization: $k = 0$, $x_0, y_0, \lambda_0, z_0, v_0, w_0, \mu_0, \eta_0, \sigma, \rho$;
while $\|$`int-resid`$\| > \epsilon$; Continue **do**
  use LU decomposition to compute $\delta x_k, \delta y_k, \delta \lambda_k, \delta z_k, \delta v_k, \delta w_k$ based on (MBQPStep);
  select α so that
  $x_k + \alpha \delta x_k \geq 0$,
  $y_k + \alpha \delta y_k \geq 0$,
  $z_k + \alpha \delta z_k \geq 0$,
  $v_k + \alpha \delta v_k \geq 0$,
  $w_k + \alpha \delta w_k \geq 0$;
  set $\alpha = 0.99\alpha$ and update $x_k, y_k, \lambda_k, z_k, v_k, w_k$ as $x_k + \alpha \delta x_k, y_k + \alpha \delta y_k, \lambda_k + \alpha \delta \lambda_k, z_k + \alpha \delta z_k, v_k + \alpha \delta v_k, w_k + \alpha \delta w_k$;
  compute $\|$`int-resid`$\|$
  $k = k + 1, \mu_k = \sigma \mu_k, \eta_k = \sigma \eta_k$;
end while
round y_k to y_{int}, set y fix to y_{int}, then solve the QP using Algorithm 1 and get x_{QP}.
return x_{QP}, y_{int};

51–53]. We utilize the augmented Lagrangian merit function which is defined as

$$\begin{aligned} M(x, y, \lambda) \triangleq & \tfrac{1}{2}x^T Q x + c^T x + + \tfrac{1}{2} y^T R y + d^T y - \mu \sum_{i=1}^{n} \ln x_i \\ & - \eta \sum_{j=1}^{p} [\ln y_j + \ln(1 - y_j)] + \rho y^T (e - y) - \lambda^T (Ax + By - b) \\ & + \tfrac{1}{2} \rho_m (Ax + By - b)^T (Ax + By - b) \end{aligned}$$

Then we may define a function $\phi(\alpha)$:

$$\begin{aligned} \phi(\alpha) &= M(x + \alpha \delta x, y + \alpha \delta y, \lambda + \alpha \delta \lambda) \\ \phi'(\alpha) &= M'(x + \alpha \delta x, y + \alpha \delta y, \lambda + \alpha \delta \lambda). \end{aligned}$$

For a given interval $[\alpha_l, \alpha_u]$, compute $\phi(\alpha_l), \phi(\alpha_u), \phi(\alpha_{mid})$ where $\alpha_{mid} = \frac{1}{2}\alpha_l + \frac{1}{2}\alpha_u$. Using quadratic interpolation, we obtain the following α_{trial}

$$\alpha_{trial} = \frac{1}{2} \frac{\phi(\alpha_l)(\alpha_u^2 - \alpha_{mid}^2) + \phi(\alpha_u)(\alpha_{mid}^2 - \alpha_l^2) + \phi(\alpha_{mid})(\alpha_l^2 - \alpha_u^2)}{\phi(\alpha_l)(\alpha_u - \alpha_{mid}) + \phi(\alpha_u)(\alpha_{mid} - \alpha_l) + \phi(\alpha_{mid})(\alpha_l - \alpha_u)}.$$

By using the Strong Wolfe conditions as termination criteria and for $0 < \sigma \leq \eta < 1$, we have the following:

$$\begin{aligned} \phi(\alpha) &\leq \phi(0) + \sigma\alpha\phi'(0) \\ |\phi'(\alpha)| &\leq \eta|\phi'(0)|. \end{aligned}$$

Our linesearch algorithm is defined as follows.

Algorithm 3 Linesearch algorithm for MBQP

```
initialization: ρ_m, σ, η, α_l = 0, α_u = 1, α_trial = 1;
while α_trial does not give sufficient decent  do
    Update [α_l, α_u] to [α_l, α_trial] or [α_trial, α_u]
    Compute new α_trial
end while
α* = α_trial
return α*
```

Chapter 3 | Numerical investigations

3.1 Fixed penalty parameter

The experiments are conducted on a OS X Yosemite machine with 1.3 GHz Intel Core i5 processor and 4 GB 1600 MHz DDR3 memory. We use C++ in version 6.4 Xcode to run the experiments. For practical purposes, we use the IEEE bus system data in [54]. The network consists of 24 buses, 26 firms, and 32 lines. The cost linear cost coefficients and the network data were also defined in [54]. The quadratic costs are randomly generated while Q is chosen to be positive semidefinite. In the results, the term n is number of integer variables and p stands for number of continuous variables. We have two quadratic programming procedures. The first quadratic programming procedure uses an interior point method to solve the smoothed problem with penalty term. Once we obtain the first result, we round the integer variables so that we may guarantee integrality. After that we fix these integer variables and solve the problem with continuous variables again to get the final result. We compare out result with that from Gurobi and record the relative difference in final objective function value. The term i_res is $\|x_b - x_b^{round}\|_\infty$ is the integrality residual of integer variables after first solving process. The term $qp1\_res$ and $qp2\_res$ stand for the residual of KKT conditions which can indicate the convergence. The term $it1$ and $it2$ represents the number of iterations in the two procedures. First, we use 999999 as a penalty parameter and compare the result with Gurobi. As seen in Table 3.1, in most cases, our algorithm reaches the same solution as Gurobi. Also, generally the KKT residuals terminate at a relatively low level as the algorithm generates a sequence that converges to a KKT

Table 3.1. Fixed Penalty $\rho = 999999$

n	p	i_res	qp1_res	qp2_res	it1	it2	obj	time	obj_difference
8	16	1.64E-21	1.48E-10	5.21E-09	501	34	1.14E+06	0.37	0.000%
8	16	1.31E-21	1.48E-10	5.21E-09	501	34	1.14E+06	0.32	0.000%
8	16	3.24E-21	1.48E-10	8.99E-09	501	31	8.66E+05	0.35	0.000%
8	16	1.63E-17	8.54E-11	8.32E-09	365	30	7.31E+05	0.29	0.000%
8	16	1.70E-17	9.27E-11	5.84E-09	407	29	1.22E+06	0.31	0.000%
8	16	1.63E-22	1.08E-10	7.49E-09	501	29	1.83E+06	0.38	0.000%
8	16	2.57E-17	9.41E-11	8.69E-09	389	31	8.83E+05	0.30	0.000%
16	32	3.07E-18	2.10E-10	6.02E-05	501	501	4.44E+06	3.90	0.000%
16	32	9.10E-19	8.52E-11	5.60E-09	431	40	3.42E+06	2.00	0.988%
16	32	4.32E-18	5.62E-11	5.60E-09	479	37	1.58E+06	1.79	0.000%
24	48	1.30E-14	1.64E-08	7.36E-09	501	48	6.57E+06	6.48	0.000%
24	48	1.66E-14	2.07E-08	1.51E-04	501	501	5.17E+06	10.20	0.465%
24	48	1.50E-15	1.48E-09	1.21E-04	501	501	8.00E+06	11.13	0.000%
24	48	1.81E-12	2.26E-06	8.66E-05	501	501	8.56E+06	10.04	0.000%
24	48	9.96E-15	1.24E-08	1.64E-04	501	501	6.44E+06	10.58	0.000%
24	48	8.89E-14	1.11E-07	1.67E-04	501	501	6.20E+06	10.69	0.000%
24	48	4.09E-15	5.01E-09	1.09E-04	501	501	4.61E+06	10.24	1.432%
24	48	4.02E-15	5.08E-09	1.39E-04	501	501	1.02E+07	9.95	0.047%
24	48	4.00E-15	5.28E-09	5.27E-09	501	41	3.94E+06	6.39	0.053%

point. It is also observed that the integrality residuals are relatively low in this scheme, implying that in most cases, we obtain a feasible integer solution.

3.2 Increasing penalty parameter

Although for a large enough penalty term, the penalized problem is equivalent to the original problem. However, as the penalty term grows, the convergence rate is adversely affected. Following the idea of [1], we set the penalty term at a relatively small value at the outset. If the integrality residual increases, we increase the penalty term in an appropriate rate. When we set this parameter to 1 at the outset, the results are presented in Table 3.2. As we see from the table, in most cases we reach almost same result as Gurobi. Compared to Table 3.1, in some cases we use less iterations to solve the problem. However, the KKT residuals are not as good as those in the fixed penalty case. In some cases, the convergence to a KKT point is not guaranteed after considerable iterations. We also tested increasing penalty from a large value. The results are presented in Table 3.3. It appears that the larger penalty term is, the better the performance from the standpoint of integrality residual. However, we may need more iterations to converge.

Table 3.2. Increasing Penalty with $\rho_0 = 1$

n	p	i_res	qp1_res	qp2_res	it1	it2	obj	time	obj_difference
8	16	4.87E-12	8.86E-11	5.21E-09	249	34	1.14E+06	0.21	0.00%
8	16	3.89E-11	6.08E-11	8.99E-09	247	31	8.66E+05	0.21	0.00%
8	16	7.89E-11	9.86E-11	8.32E-09	255	30	7.31E+05	0.21	0.00%
8	16	5.24E-11	8.19E-11	5.84E-09	245	29	1.22E+06	0.21	0.00%
8	16	1.32E-01	3.64E+00	7.49E-09	501	29	1.83E+06	0.31	0.00%
8	16	9.04E-12	8.42E-11	8.69E-09	235	31	8.83E+05	0.19	0.00%
16	32	2.14E-11	8.01E-11	5.65E-05	261	501	1.93E+06	2.62	0.00%
16	32	1.35E-11	7.53E-11	6.02E-05	281	501	4.44E+06	2.60	0.00%
16	32	1.26E-01	1.51E-10	8.73E-05	501	501	3.38E+06	3.25	0.00%
16	32	6.58E-02	2.91E+00	5.60E-09	501	37	1.58E+06	1.92	0.00%
24	48	3.44E-02	2.71E+01	7.36E-09	501	48	6.57E+06	6.12	0.00%
24	48	4.58E-01	2.37E+00	1.51E-04	501	501	5.17E+06	9.04	0.47%
24	48	2.66E-01	2.13E-10	8.47E-05	501	501	8.11E+06	10.76	1.38%
24	48	1.42E-01	1.41E-10	8.66E-05	501	501	8.56E+06	10.97	0.00%
24	48	1.13E-01	1.75E+04	1.64E-04	501	501	6.44E+06	11.74	0.00%
24	48	1.74E-01	2.64E-10	1.67E-04	501	501	6.20E+06	12.10	0.00%
24	48	3.48E-01	1.68E-10	6.72E-05	501	501	4.61E+06	12.66	1.43%
24	48	1.66E-01	9.94E-11	7.95E-05	501	501	1.02E+07	11.55	0.00%
24	48	1.75E-01	8.27E-11	4.86E-05	501	501	3.94E+06	12.45	0.00%

Table 3.3. Increasing Penalty with $\rho_0 = 99999$

n	p	i_res	qp1_res	qp2_res	it1	it2	obj	time	obj_difference
8	16	2.22E-16	8.54E-11	5.21E-09	329	34	1.14E+06	0.23	0.00%
8	16	5.55E-16	9.80E-11	8.99E-09	335	31	8.66E+05	0.26	0.00%
8	16	3.64E-16	8.22E-11	8.32E-09	329	30	7.31E+05	0.25	0.00%
8	16	6.03E-16	8.89E-11	5.84E-09	369	29	1.22E+06	0.26	0.00%
8	16	7.77E-16	8.45E-11	7.49E-09	359	29	1.83E+06	0.24	0.00%
8	16	4.44E-16	7.93E-11	8.69E-09	353	31	8.83E+05	0.27	0.00%
16	32	5.32E-17	8.88E-11	5.65E-05	447	501	1.93E+06	3.12	0.00%
16	32	5.02E-16	7.75E-11	6.02E-05	405	501	4.44E+06	3.33	0.00%
16	32	1.82E-18	7.85E-11	5.60E-09	393	40	3.42E+06	1.29	0.99%
16	32	3.33E-16	8.31E-11	5.60E-09	427	37	1.58E+06	1.69	0.00%
24	48	2.16E-01	3.91E+04	1.44E-04	501	501	6.67E+06	10.99	1.53%
24	48	5.00E-01	4.52E+04	1.51E-04	501	501	5.17E+06	8.92	0.47%
24	48	2.46E-01	2.13E-10	8.47E-05	501	501	8.11E+06	10.39	1.38%
24	48	8.03E-13	5.98E-07	6.01E-05	501	501	8.56E+06	10.25	0.00%
24	48	6.32E-14	2.25E-07	1.62E-04	501	501	6.44E+06	11.07	0.00%
24	48	1.50E-01	3.51E-09	1.67E-04	501	501	6.20E+06	10.76	0.00%
24	48	3.93E-11	1.04E-03	7.08E-05	501	501	4.60E+06	10.08	1.08%
24	48	8.97E-02	2.21E-10	1.39E-04	501	501	1.02E+07	10.63	0.05%
24	48	3.24E-14	7.90E-09	5.27E-09	501	41	3.94E+06	6.037	0.05%

Chapter 4
Conclusion

In this dissertation, we have considered the solution of mixed-binary quadratic programs via a smoothing-based interior-point scheme. Inspired by the efforts by Murray and Ng [1], we consider a smoothing-based approach in which the discrete problem is replaced by a smoothed continuous problem. An interior-point scheme is developed for such a problem and enjoys some crucial distinctions with prior approaches. Amongst these, the scheme utilizes a penalization on the integrality residual in which the penalty parameter is either maintained constant or updated in accordance with a change in the residual.

We applied the interior-point scheme on a class of mixed-binary quadratic programs arising from a class of unit commitment problems. We observe that our schemes obtain global or near-global solutions that satisfy integrality residuals. Early results suggest that the choice of penalty parameters is crucial to the progress and performance of the scheme.

Bibliography

[1] Murray, W. and K.-M. Ng (2010) "An algorithm for nonlinear optimization problems with binary variables," **Computational Optimization and Applications**, **47**(2), pp. 257–288.

[2] Lawler, E. L. and D. E. Wood (1966) "Branch-and-bound methods: A survey," **Operations research**, **14**(4), pp. 699–719.

[3] Gupta, O. K. and A. Ravindran (1985) "Branch and bound experiments in convex nonlinear integer programming," **Management science**, **31**(12), pp. 1533–1546.

[4] Quesada, I. and I. E. Grossmann (1992) "An LP/NLP based branch and bound algorithm for convex MINLP optimization problems," **Computers & chemical engineering**, **16**(10), pp. 937–947.

[5] Fletcher, R. and S. Leyffer (1998) "Numerical experience with lower bounds for MIQP branch-and-bound," **SIAM Journal on Optimization**, **8**(2), pp. 604–616.

[6] Wolsey, L. A. and G. L. Nemhauser (2014) **Integer and combinatorial optimization**, John Wiley & Sons.

[7] Padberg, M. and G. Rinaldi (1987) "Optimization of a 532-city symmetric traveling salesman problem by branch and cut," **Operations Research Letters**, **6**(1), pp. 1–7.

[8] ——— (1991) "A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems," **SIAM review**, **33**(1), pp. 60–100.

[9] Stubbs, R. A. and S. Mehrotra (1999) "A branch-and-cut method for 0-1 mixed convex programming," **Mathematical programming**, **86**(3), pp. 515–532.

[10] Geoffrion, A. M. (1974) **Lagrangean relaxation for integer programming**, Springer.

[11] Held, M. and R. M. Karp (1970) "The traveling-salesman problem and minimum spanning trees," **Operations Research**, **18**(6), pp. 1138–1162.

[12] ——— (1971) "The traveling-salesman problem and minimum spanning trees: Part II," **Mathematical programming**, **1**(1), pp. 6–25.

[13] Shapiro, J. F. (1971) "Generalized Lagrange multipliers in integer programming," **Operations Research**, **19**(1), pp. 68–76.

[14] Fisher, M. L. and J. F. Shapiro (1974) "Constructive duality in integer programming," **SIAM Journal on Applied Mathematics**, **27**(1), pp. 31–52.

[15] Fisher, M. L. (2004) "The Lagrangian relaxation method for solving integer programming problems," **Management science**, **50**(12_supplement), pp. 1861–1871.

[16] Geoffrion, A. M. (1972) "Generalized benders decomposition," **Journal of optimization theory and applications**, **10**(4), pp. 237–260.

[17] Duran, M. A. and I. E. Grossmann (1986) "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," **Mathematical programming**, **36**(3), pp. 307–339.

[18] Viswanathan, J. and I. E. Grossmann (1990) "A combined penalty function and outer-approximation method for MINLP optimization," **Computers & Chemical Engineering**, **14**(7), pp. 769–782.

[19] Fletcher, R. and S. Leyffer (1994) "Solving mixed integer nonlinear programs by outer approximation," **Mathematical programming**, **66**(1-3), pp. 327–349.

[20] Westerlund, T. and F. Pettersson (1995) "An extended cutting plane method for solving convex MINLP problems," **Computers & Chemical Engineering**, **19**, pp. 131–136.

[21] Sheble, G. B. and G. N. Fahd (1994) "Unit commitment literature synopsis," **Power Systems, IEEE Transactions on**, **9**(1), pp. 128–135.

[22] Lauer, G., N. Sandell Jr, D. Bertsekas, and T. Posbergh (1982) "Solution of large-scale optimal unit commitment problems," **Power Apparatus and Systems, IEEE Transactions on**, (1), pp. 79–86.

[23] Cohen, A. I. and M. Yoshimura (1983) "A branch-and-bound algorithm for unit commitment," **IEEE Trans. Power Appar. Syst.;(United States)**, **102**(2).

[24] Merlin, A. and P. Sandrin (1983) "A new method for unit commitment at Electricité de France," **IEEE Trans. Power Appar. Syst.;(United States)**, **102**(5).

[25] Tong, S. and S. Shahidehpour (1989) "Combination of Lagrangian-relaxation and linear-programming approaches for fuel-constrained unit-commitment problems," in **IEE Proceedings C (Generation, Transmission and Distribution)**, vol. 136, IET, pp. 162–174.

[26] Aoki, K., M. Itoh, T. Satoh, K. Nara, and M. Kanezashi (1989) "Optimal long-term unit commitment in large scale systems including fuel constrained thermal and pumped-storage hydro," **Power Systems, IEEE Transactions on**, **4**(3), pp. 1065–1073.

[27] Feltenmark, S., K. C. Kiwiel, and P.-O. Lindberg (1997) "Solving unit commitment problems in power production planning," in **Operations Research Proceedings 1996**, Springer, pp. 236–241.

[28] Zheng, Q. P., J. Wang, P. M. Pardalos, and Y. Guan (2013) "A decomposition approach to the two-stage stochastic unit commitment problem," **Annals of Operations Research**, **210**(1), pp. 387–410.

[29] Goez, J., J. Luedtke, D. Rajan, and J. Kalagnanam (2008) **Stochasic Unit Committment Problem**, **Tech. rep.**, IBM.

[30] Abhyankar, K., T. Fogarty, J. L. Kimber, A. Lin, S. Seo, and S. Takriti (1998) "Deterministic and stochastic models for the unit commitment problem," .

[31] Doyle, M. (2003) **A barrier algorithm for large nonlinear optimization problems**, Ph.D. thesis, stanford university.

[32] Frisch, K. (1955) "The logarithmic potential method of convex programming," **Memorandum, University Institute of Economics, Oslo**, **5**(6).

[33] McCormick, G. P. **Nonlinear programming**, SIAM.

[34] Karmarkar, N. (1984) "A new polynomial-time algorithm for linear programming," in **Proceedings of the sixteenth annual ACM symposium on Theory of computing**, ACM, pp. 302–311.

[35] Gill, P. E., W. Murray, M. A. Saunders, J. A. Tomlin, and M. H. Wright (1986) "On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method," **Mathematical programming**, **36**(2), pp. 183–209.

[36] Megiddo, N. (1989) **Pathways to the optimal set in linear programming**, Springer.

[37] Wright, M. H. (1992) "Interior methods for constrained optimization," **Acta numerica**, **1**, pp. 341–407.

[38] Nash, S. G. and A. Sofer (1993) "A barrier method for large-scale constrained optimization," **ORSA Journal on Computing**, **5**(1), pp. 40–53.

[39] Wright, S. and D. Ralph (1996) "A superlinear infeasible-interior-point algorithm for monotone complementarity problems," **Mathematics of operations Research**, **21**(4), pp. 815–838.

[40] Monteiro, R. D., J.-S. Pang, and T. Wang (1995) "A positive algorithm for the nonlinear complementarity problem," **SIAM Journal on Optimization**, **5**(1), pp. 129–148.

[41] Lasdon, L., G. Yu, and J. Plummer (1992) "An interior-point algorithm for solving general nonlinear programming problems," in **SIAM Conference on Optimization, Chicago, Illinois**.

[42] Yamashita, H. (1998) "A globally convergent primal-dual interior point method for constrained optimization," **Optimization Methods and Software**, **10**(2), pp. 443–469.

[43] McCormick, G. P. and J. Falk (1991) "The superlinear convergence of a nonlinear primal-dual algorithm," **Re-port T-550/91, Department of Operations Research, The George Washington University, Washington, DC**.

[44] Anstreicher, K. M. and J.-P. Vial (1994) "On the convergence of an infeasible primal-dual interior-point method for convex programming," **Optimization methods and software**, **3**(4), pp. 273–283.

[45] Kojima, M., N. Megiddo, and T. Noma (1991) "Homotopy continuation methods for nonlinear complementarity problems," **Mathematics of Operations Research**, **16**(4), pp. 754–774.

[46] Monteiro, R. and S. Wright (1992) "A globally and superlinearly convergent potential reduction interior point method for convex programming," **Unpublished Manuscript**.

[47] Gill, P. E., W. Murray, D. B. Ponceleon, and M. A. Saunders (1991) **Solving reduced KKT systems in barrier methods for linear and quadratic programming**, **Tech. rep.**, DTIC Document.

[48] El-Bakry, A., R. A. Tapia, T. Tsuchiya, and Y. Zhang (1996) "On the formulation and theory of the Newton interior-point method for nonlinear programming," **Journal of Optimization Theory and Applications**, **89**(3), pp. 507–541.

[49] Pardalos, P. M., O. A. Prokopyev, and S. Busygin (2006) "Continuous approaches for solving discrete optimization problems," in **Handbook on modelling for discrete optimization**, Springer, pp. 39–60.

[50] Lucidi, S. and F. Rinaldi (2010) "An exact penalty global optimization approach for mixed-integer programming problems," **Department of Computer and System Sciences Antonio Ruberti Technical Reports**, **2**(17), p. 9.

[51] Gill, P. E., W. Murray, M. A. Saunders, and M. H. Wright (1986) **Some Theoretical Properties of an Augmented Lagrangian Merit Function.**, **Tech. rep.**, DTIC Document.

[52] Murray, W. and M. H. Wright (1994) "Line search procedures for the logarithmic barrier function," **SIAM Journal on Optimization**, **4**(2), pp. 229–246.

[53] Facchinei, F. and J. Soares (1997) "A new merit function for nonlinear complementarity problems and a related algorithm," **SIAM Journal on Optimization**, **7**(1), pp. 225–247.

[54] Wang, S., S. Shahidehpour, D. Kirschen, S. Mokhtari, and G. Irisarri (1995) "Short-term generation scheduling with transmission and environmental constraints using an augmented Lagrangian relaxation," **Power Systems, IEEE Transactions on**, **10**(3), pp. 1294–1301.