An Exact Penalty Method for Binary Optimization Based on MPEC Formulation

Ganzhao Yuan and Bernard Ghanem
King Abdullah University of Science and Technology (KAUST), Saudi Arabia
yuanganzhao@gmail.com, bernard.ghanem@kaust.edu.sa

Abstract

Binary optimization is a central problem in mathematical optimization and its applications are abundant. To solve this problem, we propose a new class of continuous optimization techniques, which is based on Mathematical Programming with Equilibrium Constraints (MPECs). We first reformulate the binary program as an equivalent augmented biconvex optimization problem with a bilinear equality constraint, then we propose an exact penalty method to solve it. The resulting algorithm seeks a desirable solution to the original problem via solving a sequence of linear programming convex relaxation subproblems. In addition, we prove that the penalty function, induced by adding the complementarity constraint to the objective, is exact, i.e., it has the same local and global minima with those of the original binary program when the penalty parameter is over some threshold. The convergence of the algorithm can be guaranteed, since it essentially reduces to block coordinate descent in the literature. Finally, we demonstrate the effectiveness of our method on the problem of dense subgraph discovery. Extensive experiments show that our method outperforms existing techniques, such as iterative hard thresholding and linear programming relaxation.

1 Introduction

In this paper, we mainly focus on the following binary optimization problem:

$$\min_{\mathbf{x}} \ f(\mathbf{x}), \ s.t. \ \mathbf{x} \in \{-1,+1\}^n, \ \mathbf{x} \in \Omega. \tag{1}$$

where the objective function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex but not necessarily smooth on some convex set Ω, and the non-convexity of (1) is only caused by the binary constraints. In addition, we assume $\{-1,1\}^n \cap \Omega \neq \emptyset$.

The optimization in (1) describes many applications of interest in both computer vision and machine learning, including graph bisection (Goemans and Williamson, 1995; Keuchel et al., 2003), Markov random fields (Boykov, Veksler, and Zabih, 2001), the permutation problem (Jiang, Liu, and Wen, 2016; Fogel et al., 2015), graph matching (Cour, Srinivasan, and Shi, 2007; Toshev, Shi, and Daniilidis, 2007; Zaslavskiy, Bach, and Vert, 2009), image (co-)segmentation (Shi and Malik, 2000; Joulin, Bach, and Ponce, 2010), image registration (Wang et al., 2016), and social network analysis (e.g. subgraph discovery (Yuan and Zhang, 2013; Ames, 2015), biclustering (Ames, 2014), planted clique and biclique discovery (Ames and Vavasis, 2011), and community discovery (He et al., 2016; Chan and Yeung, 2011)), etc.

The binary optimization problem is difficult to solve, since it is NP-hard. One type of method to solve this problem is continuous in nature. The simple way is to relax the binary constraint with Linear Programming (LP) relaxation constraints $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$ and round the entries of the resulting continuous solution to the nearest integer at the end. However, not only may this solution not be optimal, it may not even be feasible and violate some constraint. Another type of optimization focuses on the cutting-plane and branch-and-cut method. The cutting plane method solves the LP relaxation and then adds linear constraints that drive the solution towards integers. The branch-and-cut method partially develops a binary tree and iteratively cuts out the nodes having a lower bound that is worse than the current upper bound, while the lower bound can be found using convex relaxation, Lagrangian duality, or Lipschitz continuity. However, this class of method ends up solving all 2^n convex subproblems in the worst case. Our algorithm aligns with the first research direction. It relies on solving a convex LP relaxation subproblem iteratively, but it provably terminates in polynomial iterations.

In non-convex optimization, good initialization is very important to the quality of the solution. Motivated by this, several papers design smart initialization strategies and establish optimality qualification of the solutions for non-convex problems. For example, the work of (Zhang, 2010) considers a multi-stage convex optimization algorithm to refine the global solution by the initial convex method; the work of (Candès, Li, and Soltanolkotabi, 2015) starts with a careful initialization obtained by a spectral method and improves this estimate by gradient descent; the work of (Jain, Netrapalli, and Sanghavi, 2013) uses the top-k singular vectors of the matrix as initialization and provides theoretical guarantees for biconvex alternating minimization algorithm. The proposed method also uses a similar initialization strategy since it reduces to convex LP relaxation in the first iteration.

The contributions of this paper are three-fold. **(a)** We reformulate the binary program as an equivalent augmented

Copyright © 2017, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Table 1: Existing continuous methods for binary optimization.

	Method and Reference	Description
Relaxed Approximation	spectral relaxation (Cour and Shi, 2007)	$\{-1,+1\}^n \approx \{\mathbf{x} \mid \|\mathbf{x}\|_2^2 = n\}$
	linear programming relaxation (Komodakis and Tziritas, 2007)	$\{-1,+1\}^n \approx \{\mathbf{x} \mid -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}\}$
	SDP relaxation (Wang et al., 2016)	$\{0,+1\}^n \approx \{\mathbf{x} \mid \mathbf{X} \succeq \mathbf{x}\mathbf{x}^T, diag(\mathbf{X}) = \mathbf{x}\}$
		$\{-1,+1\}^n \approx \{\mathbf{x} \mid \mathbf{X} \succeq \mathbf{x}\mathbf{x}^T,\ diag(\mathbf{X}) = \mathbf{1}\}$
	doubly positive relaxation (Huang, Chen, and Guibas, 2014)	$\{0,+1\}^n \approx \{\mathbf{x} \mid \mathbf{X} \succeq \mathbf{x}\mathbf{x}^T, diag(\mathbf{X}) = \mathbf{x},\ \mathbf{x} \geq \mathbf{0},\ \mathbf{X} \geq \mathbf{0}\}$
	completely positive relaxation (Burer, 2009)	$\{0,+1\}^n \approx \{\mathbf{x} \mid \mathbf{X} \succeq \mathbf{x}\mathbf{x}^T, diag(\mathbf{X}) = \mathbf{x},\ \mathbf{x} \geq \mathbf{0},\ \mathbf{X}$ is CP$\}$
	SOCP relaxation (Kumar, Kolmogorov, and Torr, 2009)	$\{-1,+1\}^n \approx \{\mathbf{x} \mid \langle \mathbf{X} - \mathbf{x}\mathbf{x}^T, \mathbf{L}\mathbf{L}^T \rangle \geq 0,\ diag(\mathbf{X}) = \mathbf{1}\},\ \forall\, \mathbf{L}$
Equivalent Optimization	iterative hard thresholding (Yuan and Zhang, 2013)	$\min_{\mathbf{x}} \|\mathbf{x} - \mathbf{x}'\|_2^2,\ s.t.\ \mathbf{x} \in \{-1,+1\}^n$
	piecewise separable reformulation (Zhang et al., 2007)	$\{-1,+1\}^n \Leftrightarrow \{\mathbf{x} \mid (\mathbf{1}+\mathbf{x}) \odot (\mathbf{1}-\mathbf{x}) = \mathbf{0}\}$
	ℓ_0 norm non-separable reformulation (Yuan and Ghanem, 2016b)	$\{-1,+1\}^n \Leftrightarrow \{\mathbf{x} \mid \|\mathbf{x}+\mathbf{1}\|_0 + \|\mathbf{x}-\mathbf{1}\|_0 \leq n\}$
	ℓ_2 box non-separable reformulation (Murray and Ng, 2010)	$\{-1,+1\}^n \Leftrightarrow \{\mathbf{x} \mid -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ \|\mathbf{x}\|_2^2 = n\}$
	ℓ_p box non-separable reformulation (Wu and Ghanem, 2016)	$\{-1,+1\}^n \Leftrightarrow \{\mathbf{x} \mid -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ \|\mathbf{x}\|_p^p = n,\ 0 < p < \infty\}$
	ℓ_2 box non-separable MPEC [This paper]	$\{-1,+1\}^n \Leftrightarrow \{\mathbf{x} \mid -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ \|\mathbf{v}\|_2^2 \leq n,\ \langle \mathbf{x}, \mathbf{v} \rangle = n,\ \forall \mathbf{v}\}$

optimization problem with a bilinear equality constraint via a variational characterization of the binary constraint. Then, we propose an exact penalty method to solve it. The resulting algorithm seeks a desirable solution to the original binary program. **(b)** We prove that the penalty function, induced by adding the complementarity constraint to the objective is exact, i.e. the set of their globally optimal solutions coincide with that of (1) when the penalty parameter is over some threshold. Thus, the convergence of the algorithm can be guaranteed, since it reduces to block coordinate descent in the literature (Tseng, 2001; Bolte, Sabach, and Teboulle, 2014). To our knowledge, this is the first attempt to solve general non-smooth binary optimization with guaranteed convergence. **(c)** We provide numerical comparisons with state-of-the-art techniques, such as iterative hard thresholding (Yuan and Zhang, 2013) and linear programming relaxation (Komodakis and Tziritas, 2007; Kumar, Kolmogorov, and Torr, 2009) on dense subgraph discovery. Extensive experiments demonstrate the effectiveness of our proposed method.

Notations. We use lowercase and uppercase boldfaced letters to denote real vectors and matrices respectively. The Euclidean inner product between $\mathbf{x}$ and $\mathbf{y}$ is denoted by $\langle \mathbf{x}, \mathbf{y} \rangle$ or $\mathbf{x}^T\mathbf{y}$. $\mathbf{X} \succeq \mathbf{0}$ means that matrix $\mathbf{X}$ is positive semi-definite. Finally, sign is a signum function with $\text{sign}(0) = \pm 1$.

2 Related Work

This paper proposes a new continuous method for binary optimization. We briefly review existing related work in this research direction in the literature (see Table 1).

There are generally two types of methods in the literature. One is the relaxed approximation method. Spectral relaxation (Cour and Shi, 2007; Olsson, Eriksson, and Kahl, 2007; Shi and Malik, 2000) replaces the binary constraint with a spherical one and solves the problem using eigen decomposition. Despite its computational merits, it is difficult to generalize to handle linear or nonlinear constraints. Linear programming relaxation (Komodakis and Tziritas, 2007; Kumar, Kolmogorov, and Torr, 2009) transforms the NP-hard optimization problem into a convex box-constrained optimization problem, which can be solved by well-established optimization methods and software. Semi-Definite Programming (SDP) relaxation (Huang, Chen, and Guibas, 2014) uses a lifting technique $\mathbf{X} = \mathbf{x}\mathbf{x}^T$ and relaxes to a convex conic $\mathbf{X} \succeq \mathbf{x}\mathbf{x}^T$ [1] to handle the binary constraint. Combining this with a unit-ball randomized rounding algorithm, the work of (Goemans and Williamson, 1995) proves that at least a factor of 87.8% to the global optimal solution can be achieved for the graph bisection problem. Since the original paper of (Goemans and Williamson, 1995), SDP has been applied to develop numerous approximation algorithms for NP-hard problems. As more constraints lead to tighter bounds for the objective, doubly positive relaxation considers constraining both the eigenvalues and the elements of the SDP solution to be nonnegative, leading to better solutions than canonical SDP methods. In addition, Completely Positive (CP) relaxation (Burer, 2010, 2009) further constrains the entries of the factorization of the solution $\mathbf{X} = \mathbf{L}\mathbf{L}^T$ to be nonnegative $\mathbf{L} \geq \mathbf{0}$. It can be solved by tackling its associated dual co-positive program, which is related to the study of indefinite optimization and sum-of-squares optimization in the literature. Second-Order Cone Programming (SOCP) relaxes the SDP conic into the non-negative orthant (Kumar, Kolmogorov, and Torr, 2009) using the fact that $\langle \mathbf{X} - \mathbf{x}\mathbf{x}^T, \mathbf{L}\mathbf{L}^T \rangle \geq 0,\ \forall\, \mathbf{L}$, resulting in tighter bound than the LP method, but looser than that of the SDP method. Therefore it can be viewed as a balance between efficiency and efficacy.

Another type of methods for binary optimization relates to equivalent optimization. The iterative hard thresholding method directly handles the non-convex constraint via projection and it has been widely used due to its simplicity and

[1] Using Schur complement lemma, one can rewrite $\mathbf{X} \succeq \mathbf{x}\mathbf{x}^T$ as $\begin{pmatrix} \mathbf{X} & \mathbf{x} \\ \mathbf{x}^T & 1 \end{pmatrix} \succeq \mathbf{0}$.

efficiency (Yuan and Zhang, 2013). However, this method is often observed to obtain sub-optimal accuracy and it is not directly applicable, when the objective is non-smooth. A piecewise separable reformulation has been considered in (Zhang et al., 2007), which can exploit existing smooth optimization techniques. Binary optimization can be reformulated as an ℓ_0 norm semi-continuous optimization problem. Thus, existing ℓ_0 norm sparsity constrained optimization techniques such as quadratic penalty decomposition method (Lu and Zhang, 2013) and multi-stage convex optimization method (Zhang, 2010; Yuan and Ghanem, 2016b) can be applied. A continuous ℓ_2 box non-separable reformulation [2] has been used in the literature (Raghavachari, 1969; Kalantari and Rosen, 1982). A second-order interior point method (Murray and Ng, 2010; De Santis and Rinaldi, 2012) has been developed to solve the continuous reformulation optimization problem. A continuous ℓ_p box non-separable reformulation has recently been used in (Wu and Ghanem, 2016), where an interesting geometric illustration of ℓ_p-box intersection has been shown [3]. In addition, they infuse this equivalence into the optimization framework of Alternating Direction Method of Multipliers (ADMM). However, their guarantee of convergence is weak. In this paper, to tackle the problem of binary optimization, we propose a new framework that is based on Mathematical Programming with Equilibrium Constraints (MPECs). Our resulting algorithm is theoretically convergent and empirically effective.

Mathematical programs with equilibrium constraints are optimization problems, where the constraints include complementarities or variational inequalities. They are difficult to deal with because their feasible region may not necessarily be convex or even connected. Motivated by recent development of MPECs for non-convex optimization (Yuan and Ghanem, 2015, 2016a,b), we consider continuous ℓ_2 box non-separable MPEC for binary optimization [4].

3 An Exact Penalty Method

This section presents an exact penalty method for binary optimization, which is based on a new MPEC formulation. First, we present our reformulation of the binary constraint.

Lemma 1. ***ℓ_2 box non-separable MPEC.*** *We define* $\Theta \triangleq \{(\mathbf{x},\mathbf{v}) \mid \mathbf{x}^T\mathbf{v} = n,\ \|\mathbf{v}\|_2^2 \leq n, -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}\}$. *Assume that* $(\mathbf{x},\mathbf{v}) \in \Theta$, *then* $\mathbf{x} \in \{-1,+1\}^n,\ \mathbf{v} \in \{-1,+1\}^n$ *and* $\mathbf{x} = \mathbf{v}$.

Proof. (i) Firstly, we prove that $\mathbf{x} \in \{-1,+1\}^n$. Using the definition of Θ and the Cauchy-Schwarz Inequality, we have: $n = \mathbf{x}^T\mathbf{v} \leq \|\mathbf{x}\|_2\|\mathbf{v}\|_2 \leq \|\mathbf{x}\|_2\sqrt{n} = \sqrt{n\mathbf{x}^T\mathbf{x}} \leq \sqrt{n\|\mathbf{x}\|_1\|\mathbf{x}\|_\infty} \leq \sqrt{n\|\mathbf{x}\|_1}$. Thus, we obtain $\|\mathbf{x}\|_1 \geq n$. We define $\mathbf{z} = |\mathbf{x}|$. Combining $\|\mathbf{x}\|_\infty \leq 1$, we have the following constraint sets for $\mathbf{z}$: $\sum_i \mathbf{z}_i \geq n,\ \mathbf{0} \leq \mathbf{z} \leq \mathbf{1}$. Therefore, we have $\mathbf{z} = \mathbf{1}$ and it holds that $\mathbf{x} \in \{-1,+1\}^n$. (ii) Secondly, we prove that $\mathbf{v} \in \{-1,+1\}^n$. We have:

$$n = \mathbf{x}^T\mathbf{v} \leq \|\mathbf{x}\|_\infty\|\mathbf{v}\|_1 \leq \|\mathbf{v}\|_1 = |\mathbf{v}|^T\mathbf{1} \leq \|\mathbf{v}\|_2\|\mathbf{1}\|_2 \quad (2)$$

Thus, we obtain $\|\mathbf{v}\|_2 \geq \sqrt{n}$. Combining $\|\mathbf{v}\|_2^2 \leq n$, we have $\|\mathbf{v}\|_2 = \sqrt{n}$ and $\|\mathbf{v}\|_2\|\mathbf{1}\|_2 = n$. By the Squeeze Theorem, all the equalities in (2) hold automatically. Using the equality condition for Cauchy-Schwarz Inequality, we have $|\mathbf{v}| = \mathbf{1}$ and it holds that $\mathbf{v} \in \{-1,+1\}^n$. (iii) Finally, since $\mathbf{x} \in \{-1,+1\}^n$, $\mathbf{v} \in \{-1,+1\}^n$, and $\langle \mathbf{x},\mathbf{v}\rangle = n$, we obtain $\mathbf{x} = \mathbf{v}$.

□

Using Lemma 1, we can rewrite (1) in an equivalent form as follows.

$$\min_{-\mathbf{1}\leq\mathbf{x}\leq\mathbf{1},\ \|\mathbf{v}\|_2^2\leq n} f(\mathbf{x}),\ s.t.\ \mathbf{x}^T\mathbf{v} = n,\ \mathbf{x} \in \Omega \quad (3)$$

We remark that $\mathbf{x}^T\mathbf{v} = n$ is referred to as the complementarity (or equilibrium) constraint in the literature (Luo, Pang, and Ralph, 1996; Ralph and Wright, 2004) and it always holds that $\mathbf{x}^T\mathbf{v} \leq \|\mathbf{x}\|_\infty\|\mathbf{v}\|_1 \leq \sqrt{n}\|\mathbf{v}\|_2 \leq n$ for any feasible $\mathbf{x}$ and $\mathbf{v}$.

Algorithm 1 MPEC-EPM: An Exact Penalty Method for Solving MPEC Problem (3)

(S.0) Set $t = 0,\ \mathbf{x}^0 = \mathbf{v}^0 = \mathbf{0},\ \rho > 0,\ \sigma > 1$.
(S.1) Solve the following $\mathbf{x}$-subproblem [primal step]:

$$\mathbf{x}^{t+1} = \arg\min_{\mathbf{x}}\ \mathcal{J}(\mathbf{x},\mathbf{v}^t),\ s.t.\ -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ \mathbf{x} \in \Omega \quad (4)$$

(S.2) Solve the following $\mathbf{v}$-subproblem [dual step]:

$$\mathbf{v}^{t+1} = \arg\min_{\mathbf{v}}\ \mathcal{J}(\mathbf{x}^{t+1},\mathbf{v}),\ s.t.\ \|\mathbf{v}\|_2^2 \leq n \quad (5)$$

(S.3) Update the penalty in every T iterations:

$$\rho \Leftarrow \min(2L, \rho \times \sigma) \quad (6)$$

(S.4) Set $t := t + 1$ and then go to Step (S.1)

We now present our exact penalty method for solving the optimization problem in (3). It is worthwhile to point out that there are many studies on exact penalty for MPECs (refer to (Luo, Pang, and Ralph, 1996; Hu and Ralph, 2004; Ralph and Wright, 2004; Yuan and Ghanem, 2016b) for examples), but they do not afford the exactness of our penalty problem. In an exact penalty method, we penalize the complementary error directly by a penalty function. The resulting objective $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is defined in (7), where ρ is the penalty parameter that is iteratively increased to enforce the bilinear constraint.

$$\begin{aligned} &\mathcal{J}_\rho(\mathbf{x},\mathbf{v}) = f(\mathbf{x}) + \rho(n - \mathbf{x}^T\mathbf{v}) \\ &s.t.\ -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ \|\mathbf{v}\|_2^2 \leq n,\ \mathbf{x} \in \Omega \end{aligned} \quad (7)$$

In each iteration, we minimize over $\mathbf{x}$ and $\mathbf{v}$ alternatingly (Tseng, 2001; Bolte, Sabach, and Teboulle, 2014), while fixing the parameter ρ. We summarize our exact penalty method in Algorithm 1. The parameter T is the number of inner iterations for solving the biconvex problem and the parameter L

[2] They replace $\mathbf{x} \in \{0,1\}^n$ with $\mathbf{0} \leq \mathbf{x} \leq \mathbf{1},\ \mathbf{x}^T(\mathbf{1}-\mathbf{x}) = 0$. We extend this strategy to replace $\{-1,+1\}^n$ with $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\ (\mathbf{1}+\mathbf{x})^T(\mathbf{1}-\mathbf{x}) = 0$ which reduces to $\|\mathbf{x}\|_\infty \leq 1,\ \|\mathbf{x}\|_2^2 = n$.

[3] We adapt their formulation to our $\{-1,+1\}$ formulation.

[4] For $\{0,+1\}$ binary variable, we have: $\{0,+1\}^n \Leftrightarrow \{\mathbf{x} \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{1},\ \|2\mathbf{v}-1\|_2^2 \leq n,\ \langle 2\mathbf{x}-\mathbf{1}, 2\mathbf{v}-\mathbf{1}\rangle = n,\ \forall \mathbf{v}\}$

is the Lipschitz constant of the objective function $f(\cdot)$. We make the following observations about the algorithm.

(a) Initialization. We initialize $\mathbf{v}^0$ to $\mathbf{0}$. This is for the sake of finding a reasonable local minimum in the first iteration, as it reduces to convex LP relaxation (Komodakis and Tziritas, 2007) for the binary optimization problem.

(b) Exact property. One remarkable feature of our method is the boundedness of the penalty parameter ρ (see Theorem 1). Therefore, we terminate the optimization when the threshold is reached (see (6)). This distinguishes it from the quadratic penalty method (Lu and Zhang, 2013), where the penalty may become arbitrarily large for non-convex problems.

(c) $\mathbf{v}$-Subproblem. Variable $\mathbf{v}$ in (5) is updated by solving the following convex problem:

$$\mathbf{v}^{t+1} = \arg\min \ \langle \mathbf{v}, -\mathbf{x}^{t+1} \rangle \quad s.t. \ \|\mathbf{v}\|_2^2 \leq n \tag{8}$$

When $\mathbf{x}^{t+1} = \mathbf{0}$, any feasible solution is also an optimal solution. When $\mathbf{x}^{t+1} \neq \mathbf{0}$, the optimal solution will be achieved at the constraint boundary with $\|\mathbf{v}\|_2^2 = n$ and (8) is equivalent to solving: $\min_{\|\mathbf{v}\|_2^2=n} \frac{1}{2}\|\mathbf{v}\|_2^2 - \langle \mathbf{v}, \mathbf{x}^{t+1} \rangle$. Thus, we have the following optimal solution for $\mathbf{v}$:

$$\mathbf{v}^{t+1} = \begin{cases} \sqrt{n} \cdot \mathbf{x}^{t+1} / \|\mathbf{x}^{t+1}\|_2, & \mathbf{x}^{t+1} \neq 0; \\ \text{any } \mathbf{v} \text{ with } \|\mathbf{v}\|_2^2 \leq n, & \text{otherwise.} \end{cases} \tag{9}$$

(d) $\mathbf{x}$-Subproblem. Variable $\mathbf{x}$ in (4) is updated by solving a box constrained convex problem, which has no closed-form solution in general. However, it can be solved using Nesterov's proximal gradient method (Nesterov, 2003) or classical/linearized ADM (He and Yuan, 2012).

Theoretical Analysis. In the following, we present some theoretical analysis of our exact penalty method. The following lemma is very crucial and useful in our proofs.

Lemma 2. *Let $\mathbf{x} \in \mathbb{R}^n$ be an arbitrary vector with $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$. We define $sign(x) = \begin{cases} 1, & x > 0; \\ \pm 1, & x = 0; \\ -1, & x < 0. \end{cases}$ and assume $sign(\mathbf{x}) \neq \mathbf{x}$. The following inequalities hold:*

$$h(\mathbf{x}) \triangleq \frac{n - \sqrt{n}\|\mathbf{x}\|_2}{\|sign(\mathbf{x}) - \mathbf{x}\|_2} > n - \sqrt{n^2 - n} > 1/2 \tag{10}$$

Proof. (i) We prove the first inequality in (10). We define $\mathcal{N}(\mathbf{x})$ as the number of ± 1 binary variables in $\mathbf{x}$, i.e., $\mathcal{N}(\mathbf{x}) \triangleq \#(|\mathbf{x}| = 1)$. Clearly, the objective function $h(\mathbf{x})$ decreases as $\mathcal{N}(\mathbf{x})$ increases. Note that $\mathcal{N}(\mathbf{x}) \neq n$, since otherwise it violates the assumption that $\text{sign}(\mathbf{x}) \neq \mathbf{x}$. We consider the objective value $h(\mathbf{x})$ when $\mathcal{N}(\mathbf{x}) = n - 1$. In this situation, there exists only one coordinate such that $\text{sign}(\mathbf{x}_i) \neq \mathbf{x}_i$ with $\mathbf{x}_i = \pm\delta$, $0 < \delta < 1$ and the remaining coordinates take binary variable in $\{-1, +1\}$. Note that $\delta \neq 0$ and $\delta \neq 1$, since otherwise it also violates the assumption that $\text{sign}(\mathbf{x}) \neq \mathbf{x}$. Therefore, we derive the following inequalities:

$$\begin{aligned} \frac{n - \sqrt{n}\|\mathbf{x}\|_2}{\|\text{sign}(\mathbf{x}) - \mathbf{x}\|_2} &> \frac{n - \sqrt{n}\sqrt{(n-1) + \delta^2}}{\sqrt{(1-\delta)^2}} \\ &\geq \frac{n - \sqrt{n}(\sqrt{n-1} + \delta)}{(1-\delta)} \\ &= \frac{n - \sqrt{n}\sqrt{n-1}}{(1-\delta)} + \frac{\sqrt{n}\delta}{(1-\delta)} \\ &> \frac{n - \sqrt{n}\sqrt{n-1}}{1} + 0 \end{aligned}$$

where we use the inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, $\forall a, b > 0$ and the fact that $0 < \delta < 1$. Since the lower bound above can be applied to an arbitrary vector, we finish the proof of the first inequality. (ii) We prove the second inequality in (10). We have the following results: $1/4 > 0 \ \Rightarrow \ n^2 - n + 1/4 > n^2 - n \ \Rightarrow \ (n - 1/2) > \sqrt{n^2 - n} \ \Rightarrow \ n - \sqrt{n^2 - n} > 1/2$. □

The following lemma is useful in establishing the exactness property of the penalty function in Algorithm 1.

Lemma 3. *Consider the following optimization problem:*

$$(\mathbf{x}_\rho^*, \mathbf{v}_\rho^*) = \arg \min_{-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}, \|\mathbf{v}\|_2^2 \leq n, \ \mathbf{x} \in \Omega} \mathcal{J}_\rho(\mathbf{x}, \mathbf{v}). \tag{11}$$

Assume that $f(\cdot)$ is a L-Lipschitz continuous convex function on $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$. When $\rho > 2L$, $\langle \mathbf{x}_\rho^*, \mathbf{v}_\rho^* \rangle = n$ will be achieved for any local optimal solution of (11).

Proof. First of all, we focus on the $\mathbf{v}$-subproblem in (11): $\mathbf{v}_\rho^* = \arg\min_{\mathbf{v}} -\mathbf{x}^T\mathbf{v}, \ s.t. \ \|\mathbf{v}\|_2^2 \leq n$. Assume that $\mathbf{x}_\rho^* \neq \mathbf{0}$, we have $\mathbf{v}_\rho^* = \sqrt{n} \cdot \mathbf{x}_\rho^* / \|\mathbf{x}_\rho^*\|_2$ by (9). Then the biconvex optimization problem reduces to the following:

$$\mathbf{x}_\rho^* = \arg \min_{\mathbf{x} \in [-1,+1]^n \cap \Omega} p(\mathbf{x}) \triangleq f(\mathbf{x}) + \rho(n - \sqrt{n}\|\mathbf{x}\|_2) \tag{12}$$

For any $\mathbf{x}_\rho^* \in \Omega$, we derive the following inequalities:

$$\begin{aligned} & 0.5\rho\|\text{sign}(\mathbf{x}_\rho^*) - \mathbf{x}_\rho^*\|_2 \\ \leq \ & \rho(n - \sqrt{n}\|\mathbf{x}_\rho^*\|_2) \\ = \ & [\rho(n - \sqrt{n}\|\mathbf{x}_\rho^*\|_2) + f(\mathbf{x}_\rho^*)] - f(\mathbf{x}_\rho^*) \\ \leq \ & [\rho(n - \sqrt{n}\|\text{sign}(\mathbf{x}_{\boldsymbol{\rho}}^*)\|_2) + f(\text{sign}(\mathbf{x}_{\boldsymbol{\rho}}^*))] - f(\mathbf{x}_\rho^*) \\ = \ & f(\text{sign}(\mathbf{x}_{\boldsymbol{\rho}}^*)) - f(\mathbf{x}_\rho^*) \\ = \ & L\|\text{sign}(\mathbf{x}_\rho^*) - \mathbf{x}_\rho^*\|_2 \end{aligned} \tag{13}$$

where the first step uses Lemma 2 that $\|\text{sign}(\mathbf{x}) - \mathbf{x}\|_2 \leq 2(n - \sqrt{n}\|\mathbf{x}\|_2)$ for any $\mathbf{x}$ in $\|\mathbf{x}\|_\infty \leq 1$. The third step uses the optimality of $\mathbf{x}_\rho^*$ in (12), where $p(\mathbf{x}_\rho^*) \leq p(\mathbf{y})$ for any $\mathbf{y} \in [-1, +1]^n \cap \Omega$. The fourth step uses the fact that $\text{sign}(\mathbf{x}_\rho) \in \{-1, +1\}^n$ and $\sqrt{n}\|\text{sign}(\mathbf{x}_\rho)\|_2 = n$, while the last step exploits the Lipschitz continuity of $f(\cdot)$.

From (13), we have $\|\mathbf{x}_\rho^* - \text{sign}(\mathbf{x}_\rho^*)\|_2 \cdot (\rho - 2L) \leq 0$. Since $\rho - 2L > 0$, we conclude that it always holds that $\|\mathbf{x}_\rho^* - \text{sign}(\mathbf{x}_\rho^*)\|_2 = 0$. Thus, $\mathbf{x}_\rho^* \in \{-1, +1\}^n$. Finally, we have $\mathbf{x}_\rho^* = \sqrt{n} \cdot \mathbf{x}_\rho^* / \|\mathbf{x}_\rho^*\|_2 = \mathbf{v}_\rho^*$ and $\langle \mathbf{x}_\rho^*, \mathbf{v}_\rho^* \rangle = n$.

□

The following theorem shows that when the penalty parameter ρ is larger than some threshold, the biconvex objective function in (7) is equivalent to the original constrained MPEC problem in (3). This essentially implies the theoretical convergence of the algorithm, since it reduces to well-known block coordinate descent in the literature [5].

Theorem 1. ***Exactness of the Penalty Function.*** *Assume that $f(\cdot)$ is a L-Lipschitz continuous convex function on $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$. When $\rho > 2L$, the biconvex optimization $\min_{\mathbf{x},\,\mathbf{v}} \mathcal{J}_\rho(\mathbf{x},\mathbf{v})$, $s.t.\ -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$, $\|\mathbf{v}\|_2^2 \leq n$, $\mathbf{x} \in \Omega$ in (7) has the same local and global minima with the original problem in (3).*

Proof. We let $\mathbf{x}^*$ be any global minimizer of (3) and $(\mathbf{x}^*_\rho, \mathbf{v}^*_\rho)$ be any global minimizer of (7) for some $\rho > 2L$. (i) We now prove that $\mathbf{x}^*$ is also a global minimizer of (7). For any feasible $\mathbf{x}$ and $\mathbf{v}$, we derive the following inequalities:

$$\begin{aligned} & \mathcal{J}(\mathbf{x},\mathbf{v},\rho) \\ \geq & \min_{\|\mathbf{x}\|_\infty \leq 1,\ \|\mathbf{v}\|_2^2 \leq n,\ \mathbf{x}\in\Omega} f(\mathbf{x}) + \rho(n - \mathbf{x}^T\mathbf{v}) \\ = & \min_{\|\mathbf{x}\|_\infty \leq 1,\ \|\mathbf{v}\|_2^2 \leq n,\ \mathbf{x}\in\Omega} f(\mathbf{x}),\ s.t.\ \mathbf{x}^T\mathbf{v} = n \\ = & f(\mathbf{x}^*) + \rho(n - \mathbf{x}^{*T}\mathbf{v}^*) \\ = & \mathcal{J}(\mathbf{x}^*,\mathbf{v}^*,\rho) \end{aligned}$$

where the first equality holds due to the fact that the constraint $\mathbf{x}^T\mathbf{v} = n$ is satisfied at the local optimal solution when $\rho > 2L$ (see Lemma 3). Therefore, we conclude that any optimal solution of (3) is also an optimal solution of (7). (ii) We now prove that $\mathbf{x}^*_\rho$ is also a global minimizer of (3). For any feasible $\mathbf{x}$ and $\mathbf{v}$, we naturally have the following inequalities:

$$\begin{aligned} & f(\mathbf{x}^*_\rho) - f(\mathbf{x}) \\ = & f(\mathbf{x}^*_\rho) + \rho(n - \mathbf{x}^{*T}_\rho\mathbf{v}^*_\rho) - f(\mathbf{x}) - \rho(n - \mathbf{x}^T\mathbf{v}) \\ = & \mathcal{J}_\rho(\mathbf{x}^*_\rho,\mathbf{v}^*_\rho) - \mathcal{J}_\rho(\mathbf{x},\mathbf{v}) \\ \leq & 0 \end{aligned}$$

where the first equality uses Lemma 3. Therefore, we conclude that any optimal solution of (7) is also an optimal solution of (3). (iii) In summary, we conclude that when $\rho > 2L$, the biconvex optimization in (7) has the same local and global minima with the original problem in (3). □

The following theorem characterizes the convergence rate and asymptotic monotone property of Algorithm 1.

Theorem 2. ***Convergence Rate and Asymptotic Monotone Property of Algorithm 1.*** *Assume that $f(\cdot)$ is a L-Lipschitz continuous convex function on $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$. Algorithm 1 will converge to the first-order KKT point in at most $\lceil (\ln(L\sqrt{2n}) - \ln(\epsilon\rho^0))/\ln\sigma \rceil$ outer iterations [6] with the accuracy at least $n - \mathbf{x}^T\mathbf{v} \leq \epsilon$. Moreover, after $\langle \mathbf{x},\mathbf{v}\rangle = n$ is obtained, the sequence of $\{f(\mathbf{x}^t)\}$ generated by Algorithm 1 is monotonically non-increasing.*

Proof. We denote s and t as the outer iteration and inner iteration counters in Algorithm 1, respectively. (i) We now prove the convergence rate of Algorithm 1. Assume that Algorithm 1 takes s outer iterations to converge. We denote $f'(\mathbf{x})$ as the sub-gradient of $f(\cdot)$ in $\mathbf{x}$. According to the $\mathbf{x}$-subproblem in (12), if $\mathbf{x}^*$ solves (12), then we have the following mixed variational inequality condition (He and Yuan, 2012; Jiang et al., 2016):

$$\begin{aligned} & \forall \mathbf{x} \in [-1,+1]^n \cap \Omega,\ \langle \mathbf{x} - \mathbf{x}^*, f'(\mathbf{x}^*)\rangle + \\ & \rho(n - \sqrt{n}\|\mathbf{x}\|_2) - \rho(n - \sqrt{n}\|\mathbf{x}^*\|_2) \geq 0. \end{aligned}$$

Letting $\mathbf{x}$ be any feasible solution that $\mathbf{x} \in \{-1,+1\}^n \cap \Omega$, we have the following inequality:

$$\begin{aligned} n - \sqrt{n}\|\mathbf{x}^*\|_2 &\leq n - \sqrt{n}\|\mathbf{x}\|_2 + \tfrac{1}{\rho}\langle \mathbf{x} - \mathbf{x}^*, f'(\mathbf{x}^*)\rangle \\ &\leq \tfrac{1}{\rho}\|\mathbf{x} - \mathbf{x}^*\|_2\|f'(\mathbf{x}^*)\|_2 \leq L\sqrt{2n}/\rho \end{aligned} \tag{14}$$

where the second inequality is due to the Cauchy-Schwarz Inequality, the third inequality is due to the fact that $\|\mathbf{x} - \mathbf{y}\|_2 \leq \sqrt{2n}$, $\forall\ -\mathbf{1} \leq \mathbf{x}, \mathbf{y} \leq \mathbf{1}$ and the Lipschitz continuity of $f(\cdot)$ that $\|f'(\mathbf{x}^*)\|_2 \leq L$. (14) implies that when $\rho^s \geq L\sqrt{2n}/\epsilon$, Algorithm 1 achieves accuracy at least $n - \sqrt{n}\|\mathbf{x}\|_2 \leq \epsilon$. Noticing that $\rho^s = \sigma^s\rho^0$, we have that ϵ accuracy will be achieved when $\sigma^s\rho^0 \geq \frac{L\sqrt{2n}}{\epsilon}$. Thus, we obtain

$$\sigma^s \geq \frac{L\sqrt{2n}}{\epsilon\rho^0} \quad \Rightarrow \quad s \geq (\ln(L\sqrt{2n}) - \ln(\epsilon\rho^0))/\ln\sigma$$

(ii) We now prove the asymptotic monotone property of Algorithm 1. We naturally derive the following inequalities:

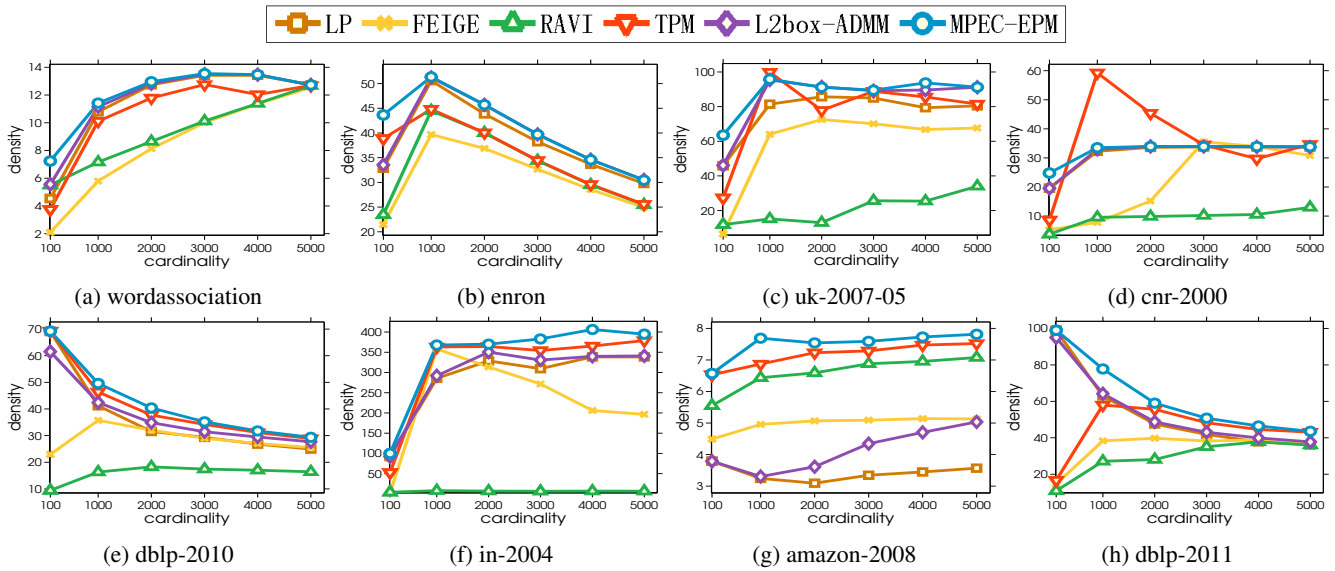
$$\begin{aligned} & f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \\ \leq & \rho(n - \langle \mathbf{x}^t,\mathbf{v}^t\rangle) - \rho(n - \langle \mathbf{x}^{t+1},\mathbf{v}^t\rangle) \\ = & \rho\left(\langle \mathbf{x}^{t+1},\mathbf{v}^t\rangle - \langle \mathbf{x}^t,\mathbf{v}^t\rangle\right) \\ \leq & \rho\left(\langle \mathbf{x}^{t+1},\mathbf{v}^{t+1}\rangle - \langle \mathbf{x}^t,\mathbf{v}^t\rangle\right) = 0 \end{aligned}$$

where the first inequality uses the fact that $f(\mathbf{x}^{t+1}) + \rho(n - \langle \mathbf{x}^{t+1},\mathbf{v}^t\rangle) \leq f(\mathbf{x}^t) + \rho(n - \langle \mathbf{x}^t,\mathbf{v}^t\rangle)$ holds because $\mathbf{x}^{t+1}$ is the optimal solution of (4). The second inequality uses the fact $-\langle \mathbf{x}^{t+1},\mathbf{v}^{t+1}\rangle \leq -\langle \mathbf{x}^{t+1},\mathbf{v}^t\rangle$ holds due to the optimality of $\mathbf{v}^{t+1}$ for (5). The last step uses $\langle \mathbf{x},\mathbf{v}\rangle = n$. Note that the equality $\langle \mathbf{x},\mathbf{v}\rangle = n$ together with the feasible set $-\mathbf{1} \leq \mathbf{x} \leq \mathbf{1}$, $\|\mathbf{v}\|_2^2 \leq n$ also implies that $\mathbf{x} \in \{-1,+1\}^n$. □

We have a few remarks on the theorems above. We assume that the objective function is L-Lipschits continuous. However, such hypothesis is not strict. Because the solution $\mathbf{x}$ is defined on the compact set, the Lipschits constant can always be computed for any continuous objective (e.g. norm function, min/max envelop function). In fact, it is equivalent

[5] Specifically, using Tseng's convergence results of block coordinate descent for non-differentiable minimization (Tseng, 2001), one can guarantee that every clustering point of Algorithm 1 is also a stationary point. In addition, stronger convergence results (Bolte, Sabach, and Teboulle, 2014; Yuan and Ghanem, 2016b) can be obtained by combining a proximal strategy and Kurdyka-Łojasiewicz inequality assumption on $\mathcal{J}(\cdot)$.

[6] Every time we increase ρ, we call it one outer iteration.



(a) wordassociation (b) enron (c) uk-2007-05 (d) cnr-2000

(e) dblp-2010 (f) in-2004 (g) amazon-2008 (h) dblp-2011

Figure 1: Experimental results for dense subgraph discovery.

to say that the (sub-) gradient of the objective is bounded by L [7]. Although exact penalty method has been study in the literature (Han and Mangasarian, 1979; Di Pillo and Grippo, 1989; Di Pillo, 1994), their results cannot directly apply here. The theoretical bound $2L$ (on the penalty parameter ρ) heavily depends on the specific structure of the optimization problem. Moreover, we also establish the convergence rate and asymptotic monotone property of our algorithm.

Based on the discussions above, we summarize the merits of our MPEC-based exact penalty method as follows. (a) It exhibits strong convergence guarantees, since it essentially reduces to block coordinate descent in the literature. (b) It seeks desirable solutions, since the LP convex relaxation method in the first iteration provides a good initialization. (c) It is efficient since it is amenable to the use of existing convex methods to solve the sub-problem. (d) It has a monotone/greedy property due to the complimentary constraints brought on by the MPEC. We penalize the complimentary error and ensure that it is decreasing in every iteration, leading to binary solutions.

4 Experimental Validation

This section demonstrates the advantages of our MPEC-based exact penalty method (MPEC-EPM) on the dense subgraph discovery problem. All codes are implemented in Matlab on an Intel 3.20GHz CPU with 8 GB RAM [8].

[7] For example, for the quadratic function $f(\mathbf{x}) = 0.5\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$, the Lipschits constant is bounded by $L \leq \|\mathbf{A}\mathbf{x}+\mathbf{b}\| \leq \|\mathbf{A}\|\|\mathbf{x}\| + \|\mathbf{b}\| \leq \|\mathbf{A}\|\sqrt{n} + \|\mathbf{b}\|$; for the ℓ_1 regression function $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_1$ with $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the Lipschits constant is bounded by $L \leq \|\mathbf{A}^T\partial|\mathbf{A}\mathbf{x}-\mathbf{b}|\| \leq \|\mathbf{A}^T\|\sqrt{m}$.

[8] For the purpose of reproducibility, we provide our MATLAB code at: yuanganzhao.weebly.com.

Table 2: The statistics of the web graph data sets used in our dense subgraph discovery experiments.

Graph	# Nodes	# Arcs	Avg. Degree
wordassociation	10617	72172	6.80
enron	69244	276143	3.99
uk-2007-05	100000	3050615	30.51
cnr-2000	325557	3216152	9.88
dblp-2010	326186	1615400	4.95
in-2004	1382908	16917053	12.23
amazon-2008	735323	5158388	7.02
dblp-2011	986324	6707236	6.80

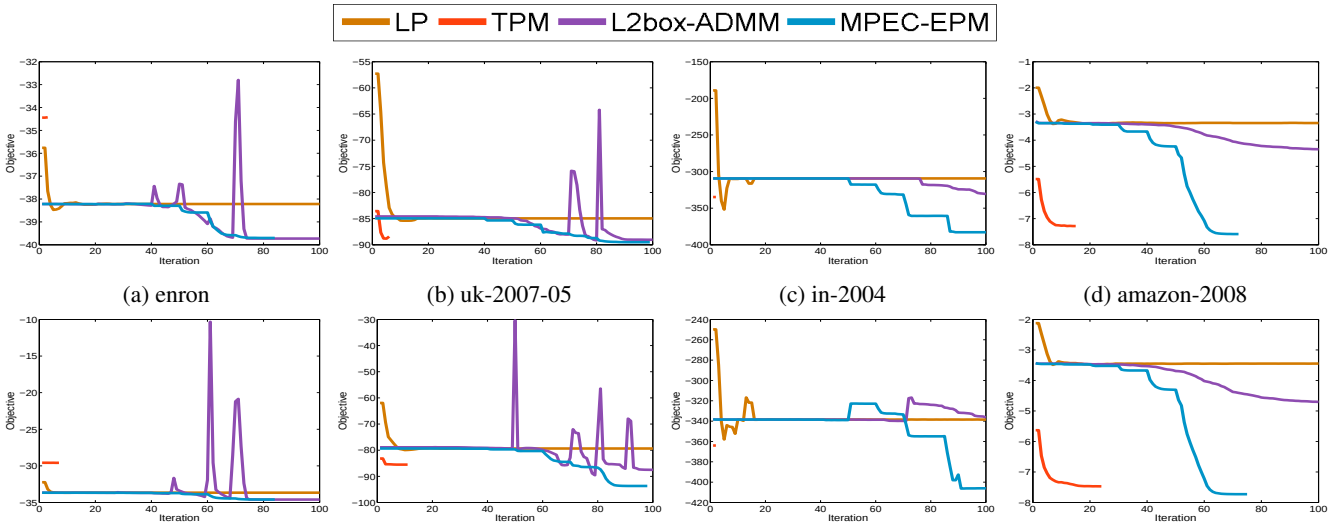
Dense subgraphs discovery (Ravi, Rosenkrantz, and Tayi, 1994; Feige, Peleg, and Kortsarz, 2001; Yuan and Zhang, 2013) is a fundamental graph-theoretic problem, as it captures numerous graph mining applications, such as community finding, regulatory motifs detection, and real-time story identification. It aims at finding the maximum density subgraph on k vertices, which can be formulated as the following binary program:

$$\max_{\mathbf{x}\in\{0,1\}^n} \ \mathbf{x}^T\mathbf{W}\mathbf{x}, \ s.t. \ \mathbf{x}^T\mathbf{1} = k \tag{15}$$

where $\mathbf{W} \in \mathbb{R}^{n\times n}$ is the adjacency matrix of the graph. Although the objective function in (15) may not be convex, one can append an additional term $\lambda\mathbf{x}^T\mathbf{x}$ to the objective with a sufficiently large λ such that $\lambda\mathbf{I} - \mathbf{W} \succeq 0$ (similar to (Ghanem, Cao, and Wonka, 2015)). This is equivalent to adding a constant to the objective since $\lambda\mathbf{x}^T\mathbf{x} = \lambda k$ in the effective domain. Therefore, we have the following equivalent problem:

$$\min_{\mathbf{x}\in\{0,1\}^n} f(\mathbf{x}) \triangleq \mathbf{x}^T(\lambda\mathbf{I} - \mathbf{W})\mathbf{x}, \ s.t. \ \mathbf{x}^T\mathbf{1} = k \tag{16}$$

In the experiments, λ is set to the largest eigenvalue of $\mathbf{W}$.



(a) enron (b) uk-2007-05 (c) in-2004 (d) amazon-2008

(e) enron (f) uk-2007-05 (g) in-2004 (h) amazon-2008

Figure 2: Convergence curve for dense subgraph discovery on different datasets with $k = 3000$ (first row) and $k = 4000$ (second row).

Compared Methods. In our experiments, we compare the following methods with different cardinality $k \in \{100,1000, 2000, 3000, 4000, 5000\}$ on 8 datasets [9] (see Table 2), which contain up to 1 million nodes and 7 million arcs. (i) Feige's greedy algorithm (GEIGE) (Feige, Peleg, and Kortsarz, 2001) is included in our comparisons. This method is known to achieve the best approximation ratio for general k. (ii) Ravi's greedy algorithm (RAVI) (Ravi, Rosenkrantz, and Tayi, 1994) starts from a heaviest edge and repeatedly adds a vertex to the current subgraph to maximize the weight of the resulting new subgraph. It has asymptotic performance guarantee of $\pi/2$, when the weights satisfy the triangle inequality. (iii) LP relaxation solves a capped simplex problem $\min_{\mathbf{x}} \ f(\mathbf{x}), \ s.t. \ \mathbf{0} \leq \mathbf{x} \leq \mathbf{1}, \ \mathbf{x}^T\mathbf{1} = k$ by proximal gradient descent method via $\mathbf{x}^{k+1} \Leftarrow \mathbf{proj}(\mathbf{x}^k - \nabla f(\mathbf{x}^k)/\eta)$ based on the current gradient $\nabla f(\mathbf{x}^k)$. Here, the projection operator $\mathbf{proj}(\mathbf{a}) \triangleq \arg\min_{\mathbf{0}\leq\mathbf{x}\leq\mathbf{1}, \ \mathbf{x}^T\mathbf{1}=k} \|\mathbf{x} - \mathbf{a}\|_2^2$ can be evaluated analytically and exactly in $n\log(n)$ time by a break point search method (Helgason, Kennington, and Lall, 1980). We use the Matlab implementation provided in (Yuan and Ghanem, 2016b). η is the gradient Lipschitz constant and it is set to the largest eigenvalue of $\lambda\mathbf{I}-\mathbf{W}$. (iv) Truncated Power Method (TPM) (Yuan and Zhang, 2013) considers an iterative procedure that combines power iteration and hard-thresholding truncation. It works by greedily decreasing the objective, while maintaining the desired binary property for the intermediate solutions. We use the code [10] provided by the authors. As suggested in (Yuan and Zhang, 2013), the initial solution is set to the indicator vector of the vertices with the top k weighted degrees of the graph in our experiments. (v) L2-box ADMM (Wu and Ghanem, 2016) applies ADMM directly to the ℓ_2 box non-separable reformulation: $\min_{\mathbf{x}} \ \mathbf{x}^T(\lambda\mathbf{I} - \mathbf{W})\mathbf{x}, \ s.t. \ \mathbf{0} \leq \mathbf{x} \leq \mathbf{1}, \ \mathbf{x}^T\mathbf{1} = k, \ \|2\mathbf{x} - 1\|_2^2 = n$. It introduces auxiliary variables to separate the two constraint sets and then performing block coordinate descend on each variable. (vi) MPEC-EPM (Algorithm 1) solves the NP-hard problem in (16) via successive convex LP relaxation. We stop Algorithm 1 when the complimentary constraint is satisfied up to a threshold, i.e., $n - \mathbf{x}^T\mathbf{v} \leq \epsilon$, where ϵ is set to 0.01. Moreover, we choose $\rho = 0.01, \ T = 10, \ \sigma = \sqrt{10}$.

Solution Quality. We compare the quality of the solution $\mathbf{x}^*$ by measuring the density of the extracted k-subgraphs, which can be computed as $\mathbf{x^*}^T\mathbf{W}\mathbf{x^*}/k$. Several observations can be drawn from Figure 1. (i) Both FEIGE and RAVI generally fail to solve the dense subgraph discovery problem and they lead to solutions with low density. (ii) LP relaxation gives better performance than the state-off-the-art technique TPM in some cases. (iii) L2-box ADMM outperforms LP relaxation for all cases, but it generates unsatisfying accuracy in 'dblp-2010', 'in-2004', 'amazon-2008' and 'dblp-2011'. (iv) Our proposed method MPEC-EPM generally outperforms all compared methods.

Convergence Curve. We demonstrate the convergence curve of the methods {LP, TPM, L2box-ADMM, MPEC-EPM} for dense subgraph discovery on different data sets. As can be seen in Figure 2, MPEC-EPM converges within 100 iterations. Moreover, its objective values generally decrease monotonically, and we attribute this to the greedy property of the penalty method.

Computational Efficiency. We provide some runtime comparisons for the four methods on different data sets. As can be seen in Table 3, even for the data set such as 'dblp-2011' that contains about one million nodes and 7 million edges, all the methods can terminate within 15 minutes. Moreover, the runtime efficiency of our method is several

[9] http://law.di.unimi.it/datasets.php

[10] https://sites.google.com/site/xtyuan1980/publications

times slower than LP and comparable with L2-box ADMM. This is expected, since (i) MPEC-EPM needs to call the LP procedure multiple times, and (ii) the methods {LP, L2-box ADMM, MPEC-EPM} are alternating methods and have the same computational complexity. Our method calls the convex LP procedure many times until convergence. Although we only present a simple projection method in our implementation, we argue that this convex LP procedure could be further significantly accelerated, by integrating exiting more advanced optimization techniques (such as coordinate gradient descent). However, this is outside the scope of this paper and left as future work.

Table 3: CPU time (in seconds) comparisons.

Graph	LP	TPM	L2box-ADM	MPEC-EPM
wordassoc.	1	1	7	2
enron	2	1	40	29
uk-2007-05	6	1	75	65
cnr-2000	16	1	210	209
dblp-2010	15	1	234	282
in-2004	79	2	834	1023
amazon-2008	49	5	501	586
dblp-2011	59	8	554	621

5 Conclusions and Future Work

This paper presents a new continuous MPEC-based optimization method to solve general binary programs. Although the problem is non-convex, we design an exact penalty method to solve its equivalent MPEC reformulation. It works by solving a sequence of convex relaxation subproblems, resulting in better and better approximations to the original non-convex formulation. We also shed some theoretical light on the equivalent formulation and optimization algorithm. Experimental results on binary problems demonstrate that our method generally outperforms existing solutions in terms of solution quality.

As for our future work, we plan to investigate the optimality qualification of our multi-stage convex relaxation method for some specific objective functions, e.g., as is done in (Goemans and Williamson, 1995; Zhang, 2010; Candès, Li, and Soltanolkotabi, 2015; Jain, Netrapalli, and Sanghavi, 2013).

Acknowledgments

This work was supported by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research through the Visual Computing Center (VCC) funding. Yuan is also supported by NSF-China (61402182). A special thanks is also extended to Prof. Shaohua Pan and Dr. Li Shen (South China University of Technology) for their helpful discussions on this paper.

References

Ames, B. P. W., and Vavasis, S. A. 2011. Nuclear norm minimization for the planted clique and biclique problems. *Mathematical Programming* 129(1):69–89. 1

Ames, B. P. 2014. Guaranteed clustering and biclustering via semidefinite programming. *Mathematical Programming* 147(1-2):429–465. 1

Ames, B. P. 2015. Guaranteed recovery of planted cliques and dense subgraphs by convex relaxation. *Journal of Optimization Theory and Applications* 167(2):653–675. 1

Bolte, J.; Sabach, S.; and Teboulle, M. 2014. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming* 146(1-2):459–494. 2, 3, 5

Boykov, Y.; Veksler, O.; and Zabih, R. 2001. Fast approximate energy minimization via graph cuts. *TPAMI* 23(11):1222–1239. 1

Burer, S. 2009. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming* 120(2):479–495. 2

Burer, S. 2010. Optimizing a polyhedral-semidefinite relaxation of completely positive programs. *Mathematical Programming Computation* 2(1):1–19. 2

Candès, E. J.; Li, X.; and Soltanolkotabi, M. 2015. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory* 61(4):1985–2007. 1, 8

Chan, E. Y. K., and Yeung, D. 2011. A convex formulation of modularity maximization for community detection. In *IJCAI*, 2218–2225. 1

Cour, T., and Shi, J. 2007. Solving markov random fields with spectral relaxation. In *AISTATS*, volume 2, 15. 2

Cour, T.; Srinivasan, P.; and Shi, J. 2007. Balanced graph matching. *NIPS* 19:313. 1

De Santis, M., and Rinaldi, F. 2012. Continuous reformulations for zero–one programming problems. *Journal of Optimization Theory and Applications* 153(1):75–84. 3

Di Pillo, G., and Grippo, L. 1989. Exact penalty functions in constrained optimization. *SIAM Journal on Control and Optimization* 27(6):1333–1360. 6

Di Pillo, G. 1994. Exact penalty methods. In *Algorithms for Continuous Optimization*. Springer. 209–253. 6

Feige, U.; Peleg, D.; and Kortsarz, G. 2001. The dense k-subgraph problem. *Algorithmica* 29(3):410–421. 6, 7

Fogel, F.; Jenatton, R.; Bach, F. R.; and d'Aspremont, A. 2015. Convex relaxations for permutation problems. *SIMAX* 36(4):1465–1488. 1

Ghanem, B.; Cao, Y.; and Wonka, P. 2015. Designing camera networks by convex quadratic programming. *Computer Graphics Forum (Proceedings of Eurographics)*. 6

Goemans, M. X., and Williamson, D. P. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM* 42(6):1115–1145. 1, 2, 8

Han, S.-P., and Mangasarian, O. L. 1979. Exact penalty functions in nonlinear programming. *Mathematical programming* 17(1):251–269. 6

He, B., and Yuan, X. 2012. On the $\mathcal{O}(1/n)$ convergence rate of the douglas-rachford alternating direction method. *SINUM* 50(2):700–709. 4, 5

He, L.; Lu, C.; Ma, J.; Cao, J.; Shen, L.; and Yu, P. S. 2016. Joint community and structural hole spanner detection via harmonic modularity. In *SIGKDD*, 875–884. 1

Helgason, R.; Kennington, J.; and Lall, H. 1980. A polynomially bounded algorithm for a singly constrained quadratic program. *Mathematical Programming* 18(1):338–343. 7

Hu, X., and Ralph, D. 2004. Convergence of a penalty method for mathematical programming with complementarity constraints. *Journal of Optimization Theory and Applications* 123(2):365–390. 3

Huang, Q.; Chen, Y.; and Guibas, L. J. 2014. Scalable semidefinite relaxation for maximum A posterior estimation. In *ICML*, 64–72. 2

Jain, P.; Netrapalli, P.; and Sanghavi, S. 2013. Low-rank matrix completion using alternating minimization. In *STOC*, 665–674. 1, 8

Jiang, B.; Lin, T.; Ma, S.; and Zhang, S. 2016. Structured nonconvex and nonsmooth optimization: Algorithms and iteration complexity analysis. *arXiv preprint*. 5

Jiang, B.; Liu, Y.-F.; and Wen, Z. 2016. ℓ_p-norm regularization algorithms for optimization over permutation matrices. *SIAM Journal on Optimization (SIOPT)* 26(4):2284–2313. 1

Joulin, A.; Bach, F. R.; and Ponce, J. 2010. Discriminative clustering for image co-segmentation. In *CVPR*, 1943–1950. 1

Kalantari, B., and Rosen, J. B. 1982. Penalty for zero–one integer equivalent problem. *Mathematical Programming* 24(1):229–232. 3

Keuchel, J.; Schnorr, C.; Schellewald, C.; and Cremers, D. 2003. Binary partitioning, perceptual grouping, and restoration with semidefinite programming. *TPAMI* 25(11):1364–1379. 1

Komodakis, N., and Tziritas, G. 2007. Approximate labeling via graph cuts based on linear programming. *TPAMI* 29(8):1436–1453. 2, 4

Kumar, M. P.; Kolmogorov, V.; and Torr, P. H. S. 2009. An analysis of convex relaxations for MAP estimation of discrete mrfs. *JMLR* 10:71–106. 2

Lu, Z., and Zhang, Y. 2013. Sparse approximation via penalty decomposition methods. *SIOPT* 23(4):2448–2478. 3, 4

Luo, Z.-Q.; Pang, J.-S.; and Ralph, D. 1996. *Mathematical programs with equilibrium constraints*. Cambridge University Press. 3

Murray, W., and Ng, K. 2010. An algorithm for nonlinear optimization problems with binary variables. *Computational Optimization and Applications* 47(2):257–288. 2, 3

Nesterov, Y. E. 2003. *Introductory lectures on convex optimization: a basic course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers. 4

Olsson, C.; Eriksson, A. P.; and Kahl, F. 2007. Solving large scale binary quadratic problems: Spectral methods vs. semidefinite programming. In *CVPR*, 1–8. 2

Raghavachari, M. 1969. On connections between zero-one integer programming and concave programming under linear constraints. *Operations Research* 17(4):680–684. 3

Ralph, D., and Wright, S. J. 2004. Some properties of regularization and penalization schemes for mpecs. *Optimization Methods and Software* 19(5):527–556. 3

Ravi, S. S.; Rosenkrantz, D. J.; and Tayi, G. K. 1994. Heuristic and special case algorithms for dispersion problems. *Operations Research* 42(2):299–310. 6, 7

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *TPAMI* 22(8):888–905. 1, 2

Toshev, A.; Shi, J.; and Daniilidis, K. 2007. Image matching via saliency region correspondences. In *CVPR*, 1–8. IEEE. 1

Tseng, P. 2001. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications* 109(3):475–494. 2, 3, 5

Wang, P.; Shen, C.; van den Hengel, A.; and Torr, P. 2016. Large-scale binary quadratic optimization using semidefinite relaxation and applications. *TPAMI*. 1, 2

Wu, B., and Ghanem, B. 2016. ℓ_p-box ADMM: A versatile framework for integer programming. In *arXiv preprint*. 2, 3, 7

Yuan, G., and Ghanem, B. 2015. $\ell_0 tv$: A new method for image restoration in the presence of impulse noise. In *CVPR*, 5369–5377. 3

Yuan, G., and Ghanem, B. 2016a. A proximal alternating direction method for semi-definite rank minimization. In *AAAI*, 2300–2308. 3

Yuan, G., and Ghanem, B. 2016b. Sparsity constrained minimization via mathematical programming with equilibrium constraints. In *arXiv preprint*. 2, 3, 5, 7

Yuan, X., and Zhang, T. 2013. Truncated power method for sparse eigenvalue problems. *JMLR* 14(1):899–925. 1, 2, 3, 6, 7

Zaslavskiy, M.; Bach, F. R.; and Vert, J. 2009. A path following algorithm for the graph matching problem. *TPAMI* 31(12):2227–2242. 1

Zhang, Z.; Li, T.; Ding, C.; and Zhang, X. 2007. Binary matrix factorization with applications. In *ICDM*, 391–400. 2, 3

Zhang, T. 2010. Analysis of multi-stage convex relaxation for sparse regularization. *JMLR* 11:1081–1107. 1, 3, 8