# Improving the Performance of MIQP Solvers for Quadratic Programs with Cardinality and Minimum Threshold Constraints: A Semidefinite Program Approach

Xiaojin Zheng
School of Economics and Management, Tongji University, Shanghai 200092, P. R. China, xjzheng@tongji.edu.cn

Xiaoling Sun
Department of Management Science, School of Management, Fudan University, Shanghai 200433, P. R. China, xls@fudan.edu.cn

Duan Li
Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N. T., Hong Kong, dli@se.cuhk.edu.hk

We consider in this paper quadratic programming problems with cardinality and minimum threshold constraints which arise naturally in various real-world applications such as portfolio selection and subset selection in regression. This class of problems can be formulated as mixed-integer 0-1 quadratic programs. We propose a new semidefinite program (SDP) approach for computing the "best" diagonal decomposition that gives the tightest continuous relaxation of the perspective reformulation of the problem. We also give an alternative way of deriving the perspective reformulation by applying a special Lagrangian decomposition scheme to the diagonal decomposition of the problem. This derivation can be viewed as a "dual" method to the convexification method employing the perspective function on semi-continuous variables. Computational results show that the proposed SDP approach can be advantageous for improving the performance of MIQP solvers when applied to the perspective reformulations of the problem.

*Key words:* Quadratic programming with semi-continuous variables and cardinality constraint; perspective reformulation; diagonal decomposition, semidefinite program; Lagrangian decomposition

---

# 1. Introduction

## 1.1. Problem and background

We consider in this paper the following cardinality constrained quadratic programs:

$$
\begin{aligned}
\text{(P)} \qquad & \min \; q(x) := x^T Q x + c^T x \\
& \text{s.t. } Ax \le b, && (1)\\
& \quad |\text{supp}(x)| \le K, && (2)\\
& \quad x_i \ge \alpha_i, \; \forall i \in \text{supp}(x), && (3)\\
& \quad 0 \le x_i \le u_i, \; i = 1, \ldots, n, && (4)
\end{aligned}
$$

where $Q$ is an $n \times n$ positive semidefinite matrix, $c \in \Re^n$, $A \in \Re^{m\times n}$, $b \in \Re^m$, $K$ is an integer satisfying $1 \le K \le n$, $0 < \alpha_i < u_i$, and $\text{supp}(x) = \{i \in \{1, \ldots, n\} \mid x_i \ne 0\}$. Constraint (2), referred to as the cardinality constraint, limits the number of nonzero variables in the feasible solution, and constraint (3), referred to as the minimum threshold constraint, sets lower bounds for nonzero variables. Note that (3) and (4) can be expressed together as

$$
x_i \in \{0\} \cup [\alpha_i, u_i], \qquad i = 1, \ldots, n,
$$

which are also called *semi-continuous* variables. Problem (P) is, in general, NP-hard as testing the feasibility of (P) is already NP-complete when $A$ has three rows (see Bienstock (1996)).

Cardinality constraint is often encountered in optimization models when the decision vector is required to be *sparse*. An important application of problem formulation (P) is *portfolio selection* in financial optimization. Consider a market consisting of $n$ risky assets with random return vector $R = (R_1, \ldots, R_n)^T$. Suppose that the expected return vector and the covariance matrix of $R$ are known as $\mu$ and $Q$, respectively. Then, according to Markowitz's doctrine, a mean-variance optimizer should solve the following quadratic problem:

$$
\begin{aligned}
& \min \; x^T Q x \\
& \text{s.t. } \mu^T x \ge \rho, \\
& \quad \sum_{i=1}^{n} x_i = 1,
\end{aligned}
$$

where $x_i$ represents the proportion of the total capital invested in the $i$th asset and $\rho$ is a prescribed return level set by the investor. In real-world, however, almost no investor faithfully follows the classical mean-variance model due to market frictions, such as management and transaction fees. What most investors actually do is to confine themselves in choosing a small number of stocks to invest. In other words, most investors would favor an investment decision derived from a cardinality constrained problem formulation such as problem (P). In (P), constraints (1) and (4) represent the budget constraints, restriction on short selling, sector constraints and the maximum position the investor can hold for each asset. The cardinality constraint (2) in (P) limits the total number of different assets in the optimal portfolio. The constraint (3) in (P), often referred to as *buy-in threshold* constraint or *minimum transaction level* constraint, prevents the investors from holding some assets with very small amount.

Another application of (P) is the subset selection problem in multivariate linear regression. For given $m$ observed data points $(a^i, b^i)$ with $a^i \in \Re^n$ and $b^i \in \Re$, while one always wants to minimize the least square measure of $\sum_{i=1}^m (a_i^T x - b_i)^2$, he/she often wants to achieve the goal with only a subset of the prediction variables in $x$ (see Arthanari and Dodge (1993); Bertsimas and Shioda (2009); Miller (2002)). This subset selection problem can be formulated as follows:

$$\begin{aligned} &\min \|Ax - b\|^2 \\ &\text{s.t. } |\text{supp}(x)| \le K, \end{aligned}$$

where $A^T = (a^1, \ldots, a^m)$, $b = (b_1, \ldots, b_m)^T$ and $K$ is a positive integer satisfying $1 \le K < n$. The subset selection problem is a special case of (P) where the constraints of semi-continuous variables (3) and (4) are absent. In practice, we can always impose lower bound and upper bound on $x$, i.e., $-l_i \le x_i \le u_i$, $i = 1, \ldots, n$, for some sufficiently large positive numbers $l_i$ and $u_i$.

Portfolio optimization problems with cardinality and minimum threshold constraints have been investigated in the literature by many researchers. Bonami and Lejeune (2009) proposed an exact solution for the mean-variance portfolio selection model under stochastic and integer constraints including cardinality and minimum threshold constraints. The solution method in Bonami and Lejeune (2009) is a branch-and-bound method based on continuous relaxation and special branching rules. Bertsimas and Shioda (2009) presented a specialized branch-and-bound method for (P) where a convex quadratic programming relaxation at each node is solved via Lemke's method. Bienstock (1996) developed a branch-and-cut method for solving cardinality constrained quadratic programming problems using a surrogate constraint approach. Li et al. (2006) proposed an exact solution method for cardinality constrained mean-variance models under round lot constraints and concave transaction costs, using some geometric methods and Lagrangian relaxation scheme in a branch-and-bound framework. Xie et al. (2008) proposed a randomized approach to find a good approximated solution to the mean-variance portfolio selection model with cardinality constraint and other side constraints. Shaw et al. (2008) presented a branch-and-bound method for cardinality constrained mean-variance portfolio problems, where the asset returns are driven by a factor model. Unlike other existing branch-and-bound methods in the literature where standard quadratic programming relaxation is adopted as the bounding technique, Shaw et al. (2008) used Lagrangian relaxation with cost splitting to generate a lower bound at each node of the binary search tree and employed subgradient method to compute the Lagrangian bound. Cui et al. (2013) investigated a class of cardinality constrained portfolio selection problems with different risk measures and tracking error control. Utilizing the natural decomposition of factor models, a second-order cone program relaxation and an MIQCQP reformulation were derived in Cui et al. (2013) for this class of problems. Recently, a novel geometric approach is proposed in Gao and Li (2013) for minimizing a quadratic function subject to a cardinality constraint. Based on this geometric approach, a branch-and-bound method is then developed in Gao and Li (2013) for solving cardinality-constrained portfolio selection problems.

Heuristic and local search methods for portfolio selection models with cardinality constraints and minimum threshold have been also studied by many other authors in the context of limited-diversification, small portfolios and empirical study for comparing different portfolio selection models with real features (see, e.g., Chang et al. (2000); Blog et al. (1983); Jacob (1974); Jobst et al. (2001); Maringer and Kellerer (2003); Mitra et al. (2007)).

## 1.2. Research motivation and main contributions

In this paper, we focus on the mixed-integer quadratic program (MIQP) reformulations of problem (P). By introducing a 0-1 variable $y_i$ to enforce $x_i = 0$ or $x_i \neq 0$ in (P), problem (P) can be reformulated as the following standard mixed-integer 0-1 quadratic program:

$$(\mathrm{MIQP}_0) \qquad \min\ x^T Q x + c^T x$$
$$\text{s.t. } Ax \leq b, \tag{5}$$

$$e^T y \le K, \tag{6}$$

$$\alpha_i y_i \le x_i \le u_i y_i,\ i = 1, \ldots, n, \tag{7}$$

$$y \in \{0,1\}^n, \tag{8}$$

where $e$ is the all one column vector. Standard MIQP solvers, which are based on a branch-and-bound framework, can be then applied to (MIQP$_0$) to find a global solution or a sub-optimal solution of (P). It is well known that the efficiency of branch-and-bound methods largely depends on the tightness of the lower bounds generated by the continuous relaxation. Numerical test, however, suggests that the continuous relaxation of (MIQP$_0$) often provides poor lower bounds of (MIQP$_0$) and the continuous relaxation based branch-and-bound methods thus cannot solve the problem even with small to medium size (e.g., $n = 100$) within reasonable computation time.

Frangioni and Gentile (2006, 2007) proposed a novel perspective reformulation for quadratic programs with semi-continuous variables (see also Günlük and Linderoth (2010)). Let $d \in \Re^n_+$ be such that $d \ge 0$ and $Q - D \succeq 0$, where $D = \mathrm{diag}(d)$ denotes the diagonal matrix with $d$ being the diagonal vector. The quadratic objective function of (P) can be then decomposed as

$$q(x) = x^T(Q - D)x + c^T x + x^T D x. \tag{9}$$

Recall that the perspective function of a univariate function $f(p)$ is $tf(p/t)$ for $t \ge 0$, where it is assumed that $0/0 = 0$ (see Hiriart-Urruty and Lemaréchal (1993)). Replacing the separable term $x^T D x$ with its convex envelope over the semi-continuous variables, which is the sum of the perspective functions of $d_i x_i^2$ over $x_i \in \{0\} \cup [\alpha_i, u_i]$ for $i = 1, \ldots, n$, the perspective reformulation of (P) then has the following form:

$$(\mathrm{PR}(d)) \qquad \min\ x^T(Q - \mathrm{diag}(d))x + c^T x + \sum_{i=1}^n d_i(x_i^2/y_i)$$
$$\text{s.t. } (5),\ (6),\ (7),\ (8).$$

However, the fractional terms in the objective function of (PR($d$)) prevent a direct application of efficient solution methods to (PR($d$)). To overcome this difficulty, two tractable reformulations of (PR($d$)) were proposed in the literature. The first one is the second-order cone programming (SOCP) reformulation (see Aktürk et al. (2009); Günlük and Linderoth (2010); Tawarmalani and Sahinidis (2001)), where an additional variable $\phi_i = x_i^2/y_i$ is first introduced for each $i$ and the constraint $\phi_i \geq x_i^2/y_i$ is then rewritten as an SOCP constraint. The resulting SOCP reformulation of (PR($d$)) has the following form:

$$\begin{aligned}
\text{(SOCP}(d)) \qquad & \min\ x^T(Q - \text{diag}(d))x + c^T x + \phi^T d \\
& \text{s.t.} \left\| \begin{matrix} x_i \\ \frac{\phi_i - y_i}{2} \end{matrix} \right\| \leq \frac{\phi_i + y_i}{2},\ i = 1, \dots, n, \qquad (10) \\
& (5),\ (6),\ (7),\ (8).
\end{aligned}$$

The second reformulation technique is the perspective cut (P/C) reformulation proposed by Frangioni and Gentile (2006, 2007), where the epigraph of $x_i^2/y_i$ on $\text{conv}(\{0\} \cup [\alpha_i, u_i])$ is represented by infinite many perspective cut inequalities. The resulting P/C reformulation of (PR($d$)) takes the following form:

$$\begin{aligned}
\text{(PC}(d)) \qquad & \min\ x^T(Q - \text{diag}(d))x + c^T x + d^T v \\
& \text{s.t.}\ v_i \geq 2\bar{x}_i x_i - \bar{x}_i^2 y_i, \quad \forall \bar{x}_i \in [\alpha_i, u_i],\ i = 1, \dots, n, \\
& (5),\ (6),\ (7),\ (8).
\end{aligned}$$

Although problem (PC($d$)) cannot be solved directly, "localized" subproblems of (PC($d$)) with a small finite subset of perspective cuts can be embedded in a branch-and-cut framework, where the violated perspective cuts with $\bar{x}_i = x_i^*/y_i^*$ are added at each node when the optimal solution of the continuous subproblem is $(x^*, y^*, v^*)$. This solution scheme can be either implemented by tailor-made branch-and-cut method (see Frangioni and Gentile (2006, 2007)) or by means of `cutcallback` procedures in `CPLEX` (see Frangioni and Gentile (2009)). Computational results in Frangioni and Gentile (2009) show that, if properly implemented, the P/C reformulation can be much more efficient than the SOCP reformulation.

A key issue in implementing the perspective reformulation (PR($d$)), or its two tractable reformulations (SOCP($d$)) and (PC($d$)), is how to choose the parameter vector $d$. A natural choice is $d = (\lambda_{\min} - \epsilon)e$ when $Q$ is positive definite, where $\lambda_{\min}$ is the minimum eigenvalue of $Q$ and $\epsilon > 0$ is a sufficiently small scalar. Frangioni and Gentile (2007) suggested to use a heuristic method to find a diagonal matrix $D = \text{diag}(d)$ by solving a simple semidefinite program (SDP):

$$\text{(SDP}_\text{s}) \qquad \max\{e^T d \mid Q - \text{diag}(d) \succeq 0,\ d \geq 0\}, \qquad (11)$$

which we will call the "small" SDP problem. Numerical results in Frangioni and Gentile (2007) show that this approach compares favorably with the minimum eigenvalue method. A further question arises: How to find a "better" $d$ in the perspective reformulation?

To answer the above question, we present in this paper a new approach to compute the parameter vector $d$ in the perspective reformulation. Our approach is based on finding a vector $d = d_l$ in (PR($d$)) such that the continuous relaxation of (PR($d$)) is the tightest among all admissible $d$. We show that the problem of finding such a $d_l$ can be reduced to an SDP problem with size larger than that of the "small" SDP problem ($\text{SDP}_\text{s}$). This "large" SDP problem can still be solved efficiently by interior-point based methods due to its simple structure. Numerical results suggest that using the parameter vector $d_l$ computed by the "large" SDP formulation can considerably improve the performance of the perspective reformulations, largely due to the improvement of the continuous bounds.

Stimulated by the new SDP problem formulation, we also propose a new way of deriving the perspective reformulation of problem (P) via a special Lagrangian decomposition scheme of (P). Our derivation reveals that the continuous bound of the perspective reformulation is the same as the dual bound of (P) via the Lagrangian decomposition scheme. In some sense, our SDP approach is "dual" to the method in Frangioni and Gentile (2006) where the perspective function is used to construct the convex envelope of the objective function on the semi-continuous variables.

## 1.3. Outline of the paper

The rest of the paper is organized as follows. In Section 2, we show how to reduce the problem of finding the tightest continuous relaxation of (PR($d$)) into an SDP problem. We devote Section 3 to a new derivation of the perspective reformulation by applying a special Lagrangian decomposition scheme to (P). In Section 4, we conduct computational experiments comparing the performance of perspective reformulations using different choices of $d$ for test problems arising from portfolio selection and subset selection. Finally, we conclude the paper in Section 5 with some concluding remarks.

*Notation*: Throughout the paper, we denote by $v(\cdot)$ the optimal value of problem $(\cdot)$, $\Re^n_+$ the nonnegative orthant of $\Re^n$. For any $a \in \Re^n$, we denote by $\text{diag}(a) = \text{diag}(a_1, \ldots, a_n)$ the diagonal matrix with $a_i$ being the $i$th diagonal element.

# 2. A New SDP Approach for Computing Diagonal Decomposition in Perspective Reformulation

In this section, we discuss how to select the parameter vector $d$ in the perspective reformulation ($\mathrm{PR}(d)$) such that the continuous relaxation of ($\mathrm{PR}(d)$) is the tightest.

Let ($\overline{\mathrm{PR}}(d)$) denote the continuous relaxation of ($\mathrm{PR}(d)$). The tightest continuous relaxation of ($\mathrm{PR}(d)$) can be found by solving the following problem:

$$\max\{v(\overline{\mathrm{PR}}(d)) \mid d \geq 0,\ Q - \mathrm{diag}(d) \succeq 0\}. \tag{12}$$

Introducing $\phi_i = x_i^2/y_i$ in the objective function of ($\overline{\mathrm{PR}}(d)$) and relaxing it to $\phi_i \geq x_i^2/y_i$, we can reformulate ($\overline{\mathrm{PR}}(d)$) as an SOCP problem:

$$\begin{aligned}
(\overline{\mathrm{SOCP}}(d)) \qquad & \min\ x^T(Q - \mathrm{diag}(d))x + c^T x + \phi^T d \\
& \text{s.t. } 0 \leq y_i \leq 1,\ i = 1, \ldots, n, \\
& \qquad (5),\ (6),\ (7),\ (10).
\end{aligned}$$

Problem (12) is then equivalent to the following problem:

$$\max\{v(\overline{\mathrm{SOCP}}(d)) \mid d \geq 0,\ Q - \mathrm{diag}(d) \succeq 0\}. \tag{13}$$

In the sequel, we always assume the following constraint qualification for ($\overline{\mathrm{SOCP}}(d)$).

**Assumption 1** *The feasible set of* ($\overline{\mathrm{SOCP}}(d)$) *has a (relative) interior point.*

A necessary and sufficient condition for ensuring Assumption 1 is that the continuous relaxation of ($\mathrm{MIQP}_0$), the standard mixed-integer quadratic program reformulation of (P), has a (relative) interior point.

In the following, we show that problem (13) can be reduced to an SDP problem. We first observe that the constraint (7) in ($\mathrm{SOCP}(d)$) can be replaced by

$$\phi_i - (\alpha_i + u_i)x_i + \alpha_i u_i y_i \leq 0,\ i = 1, \ldots, n, \tag{14}$$

since $x_i^2 = \phi_i$ always holds at the optimal solution of ($\mathrm{SOCP}(d)$). Indeed, if $y_i = 0$, then (10) implies $x_i = 0$; otherwise if $y_i = 1$, then constraint (14) and $x_i^2 = \phi_i$ imply $\alpha_i \leq x_i \leq u_i$. Also, the second-order cone constraint in (10) can be rewritten as

$$\begin{pmatrix} \phi_i & x_i \\ x_i & y_i \end{pmatrix} \succeq 0,\ i = 1, \ldots, n. \tag{15}$$

Therefore, $(\overline{\text{SOCP}}(d))$ can be written as:

$$
\begin{aligned}
(\overline{\text{SOCP}}_1(d)) \qquad & \min \ x^T(Q-\text{diag}(d))x + c^Tx + \phi^Td \\
& \text{s.t. } y_i - 1 \leq 0, \ i = 1,\dots,n, \\
& \qquad (5), \ (6), \ (14), \ (15),
\end{aligned}
$$

where the nonnegative constraints of $y_i \geq 0$ $(i = 1,\dots,n)$ have been implied by (15).

**Theorem 1** *Problem (13) is equivalent to the following SDP problem:*

$$
\begin{aligned}
(\text{SDP}_1) \qquad & \max \ -Ks - e^T\pi - \tau \\
& \text{s.t.} \begin{pmatrix} d_i + \mu_i & \frac{1}{2}(c_i - \lambda_i - \beta_i\mu_i) \\ \frac{1}{2}(c_i - \lambda_i - \beta_i\mu_i) & \pi_i + s + \alpha_i u_i \mu_i \end{pmatrix} \succeq 0, \ i = 1,\dots,n, && (16) \\
& \begin{pmatrix} Q - \text{diag}(d) & \frac{1}{2}(\lambda + A^T\eta) \\ \frac{1}{2}(\lambda + A^T\eta)^T & -\eta^Tb + \tau \end{pmatrix} \succeq 0, && (17) \\
& (s,\eta,\mu,\pi,d) \in \Re_+ \times \Re_+^m \times \Re_+^n \times \Re_+^n \times \Re_+^n, \ (\tau,\lambda) \in \Re \times \Re^n,
\end{aligned}
$$

*where* $\beta_i := \alpha_i + u_i$ *for* $i = 1,\dots,n$.

Proof. Let $d \in \Re_+^n$ satisfy $Q - \text{diag}(d) \succeq 0$. We first express $(\overline{\text{SOCP}}_1(d))$ by its dual form. Associate the following multipliers to the constraints in $(\overline{\text{SOCP}}_1(d))$:

- $\eta \in \Re_+^m$ for (5): $Ax \leq b$ and $s \in \Re_+$ for (6): $e^Ty \leq K$;
- $\mu_i \in \Re_+$ for (14): $\phi_i - (\alpha_i + u_i)x_i + \alpha_i u_i y_i \leq 0$, $i = 1,\dots,n$;
- $\begin{pmatrix} \rho_i & \gamma_i \\ \gamma_i & \xi_i \end{pmatrix} \succeq 0$ for (15): $\begin{pmatrix} \phi_i & x_i \\ x_i & y_i \end{pmatrix} \succeq 0$, $i = 1,\dots,n$;
- $\pi_i \in \Re_+$ for $y_i - 1 \leq 0$, $i = 1,\dots,n$.

Let $\mu,\rho,\gamma,\xi,\pi$ denote the column vectors formed by $\mu_i,\rho_i,\gamma_i,\xi_i,\pi_i$ $(i = 1,\dots,n)$, respectively. Let $d(\omega)$ denote the Lagrangian dual function of $(\overline{\text{SOCP}}_1(d))$, where $\omega$ denote the dual variables introduced above. Then, the Lagrangian dual of $(\overline{\text{SOCP}}_1(d))$ is

$$
\max\{d(\omega) \mid \begin{pmatrix} \rho_i & \gamma_i \\ \gamma_i & \xi_i \end{pmatrix} \succeq 0, \ i = 1,\dots,n, \ (s,\eta,\mu,\pi) \in \Re_+ \times \Re_+^m \times \Re_+^n \times \Re_+^n\}. \tag{18}
$$

We can calculate that

$$
d(\omega) \ = \ \min_{x,y,\phi}\{x^T(Q-\text{diag}(d))x + c^Tx + \phi^Td + \eta^T(Ax-b) + s(e^Ty - K)
$$

$$
\begin{aligned}
&+\sum_{i=1}^{n}\mu_i[\phi_i-(\alpha_i+u_i)x_i+\alpha_i u_i y_i]-\sum_{i=1}^{n}(\phi_i\rho_i+2x_i\gamma_i+y_i\xi_i)+y^T\pi-e^T\pi\} \\
= \;& \min_{x,y,\phi}\{[c+A^T\eta-\mathrm{diag}(\mu)(\alpha+u)-2\gamma]^Tx+x^T(Q-\mathrm{diag}(d))x \\
&+\sum_{i=1}^{n}(\pi_i+s+\mu_i\alpha_i u_i-\xi_i)y_i+(d+\mu-\rho)^T\phi+(-\eta^Tb-Ks-e^T\pi)\} \\
= \;& \begin{cases} -Ks-e^T\pi+\min_x \tilde{q}(x), & \text{if } \rho=d+\mu \text{ and } \xi_i=\pi_i+s+\mu_i\alpha_i u_i, \\ -\infty, \text{ otherwise,} & \end{cases}
\end{aligned}
$$

where $\tilde{q}(x)=x^T(Q-\mathrm{diag}(d))x+(A^T\eta+\lambda)^Tx-\eta^Tb$ and $\lambda=c-\mathrm{diag}(\mu)(\alpha+u)-2\gamma$. Thus, the dual problem (18) can be written as

$$
\begin{aligned}
&\max -Ks-e^T\pi-\tau \\
&\text{s.t.} \begin{pmatrix} d_i+\mu_i & \gamma_i \\ \gamma_i & \pi_i+s+\alpha_i u_i\mu_i \end{pmatrix}\succeq 0, \quad i=1,\ldots,n, && (19)\\
&\quad x^T(Q-\mathrm{diag}(d))x+(A^T\eta+\lambda)^Tx-\eta^Tb\geq -\tau, \quad \forall x\in\Re^n, && (20)\\
&\quad \lambda=c-\mathrm{diag}(\mu)(\alpha+u)-2\gamma, && (21)\\
&\quad (s,\eta,\mu,\pi)\in\Re_+\times\Re^m_+\times\Re^n_+\times\Re^n_+,\ (\tau,\lambda)\in\Re\times\Re^n.
\end{aligned}
$$

Note that (21) implies that $\gamma_i=\frac{1}{2}(c_i-\lambda_i-\beta_i\mu_i)$, where $\beta_i:=\alpha_i+u_i$. Thus, (19) is equivalent to (16). Also, (20) is equivalent to the SDP constraint in (17). Therefore, the dual problem (18) can be reduced to the SDP problem:

$$
\begin{aligned}
(\mathrm{D_s}(d)) \qquad & \max \; -Ks-e^T\pi-\tau && (22)\\
& \text{s.t. } (16),\ (17), \\
& \quad (s,\eta,\mu,\pi)\in\Re_+\times\Re^m_+\times\Re^n_+\times\Re^n_+,\ (\tau,\lambda)\in\Re\times\Re^n.
\end{aligned}
$$

By Assumption 1 and the conic duality theorem (see, e.g., Vandenberghe and Boyd (1996)), the strong duality between ($\overline{\mathrm{SOCP}}_1(d)$) and its dual holds. Therefore, problem (13) is equivalent to

$$
\max\{v(\mathrm{D_s}(\mathrm{d}))\mid d\geq 0, Q-\mathrm{diag}(d)\succeq 0\}.
$$

which is ($\mathrm{SDP_l}$) by (22) and noting that $Q-\mathrm{diag}(d)\succeq 0$ is implied by constraint (17). □

From the above derivation, we can explain the "large" SDP formulation ($\mathrm{SDP_l}$) as the SDP representation of the problem of finding the parameter $d$ that gives the tightest continuous bound of the perspective reformulation (PR($d$)) or its two tractable reformulations

(SOCP($d$)) and (PC($d$)). Compared with the "small" SDP formulation ($\text{SDP}_\text{s}$) proposed by Frangioni and Gentile (2007), the formulation ($\text{SDP}_\text{l}$) has a drawback of having a larger dimension: There are $4n+m+2$ variables in ($\text{SDP}_\text{l}$) compared to only $n$ variables in ($\text{SDP}_\text{s}$). Also, ($\text{SDP}_\text{l}$) has $n$ additional $2\times 2$ linear matrix inequalities. In spite of its larger size, ($\text{SDP}_\text{l}$) can still be computed efficiently by the interior-point based solvers such as `SeDuMi` due to its simple structure. The longer time spent on solving the "large" SDP problem ($\text{SDP}_\text{l}$) could be well compensated by the savings in the computation time of branch-and-cut method for the perspective reformulations, as witnessed in our computational experiments.

# 3. Derivation of Perspective Reformulation via Lagrangian Decomposition

In this section, we present a new approach to derive the SOCP reformulation (SOCP($d$)). Since the three reformulations (PR($d$)), (SOCP($d$)) and (PC($d$)) are equivalent to each other in terms of the optimal solutions and the continuous bounds, the derivation in this section can be viewed as an alternative way of constructing perspective reformulation for (P).

The approach of our derivation is motivated by the construction of a tight SDP relaxation of problem (P) via a special Lagrangian decomposition scheme. It turns out that the conic dual of this SDP relaxation is exactly the continuous relaxation of (SOCP($d$)). This reveals that the continuous bound of (SOCP($d$)) is nothing but the Lagrangian bound of (P). As a result, we obtain a new derivation of (SOCP($d$)) via Lagrangian decomposition of the original problem (P).

The construction of the SDP relaxation consists of the following three steps:

- Decomposing $Q$ as $Q = (Q - \text{diag}(d)) + \text{diag}(d)$, where $d \in \Re^n_+$ and $Q - \text{diag}(d) \succeq 0$;
- Constructing a convex relaxation of (P) by a special Lagrangian decomposition scheme via copying constraints;
- Reducing the Lagrangian dual to an SDP formulation.

Using the technique of copying variables (see Guignard and Kim (1987); Michelon and Maculan (1991); Shaw et al. (2008)), problem (P) can be reformulated as

$$\begin{aligned} &\min\ x^T\text{diag}(d)x + c^Tx + z^T(Q - \text{diag}(d))z \qquad (23)\\ &\text{s.t. } Az \le b, \end{aligned}$$

$$x = z,$$
$$|\text{supp}(x)| \le K,$$
$$x_i \in \{0\} \cup [\alpha_i, u_i],\ i = 1, \dots, n.$$

Dualizing the constraint $x = z$ with multiplier vector $\lambda \in \Re^n$ yields the following Lagrangian relaxation:

$$d(\lambda) = d_1(\lambda) + d_2(\lambda),$$

where

$$\begin{aligned} d_1(\lambda) &= \min\ x^T \text{diag}(d) x + (c - \lambda)^T x && (24) \\ &\quad \text{s.t. } |\text{supp}(x)| \le K,\ x_i \in \{0\} \cup [\alpha_i, u_i],\ i = 1, \dots, n, \\ d_2(\lambda) &= \min\ z^T (Q - \text{diag}(d)) z + \lambda^T z, && (25) \\ &\quad \text{s.t. } Az \le b. \end{aligned}$$

Thus, the Lagrangian dual of (P) is

$$(\text{D}(d)) \qquad \max\{d_1(\lambda) + d_2(\lambda) \mid \lambda \in \Re^n\}. \tag{26}$$

By weak duality, $v(\text{D}(d)) \le v(\text{P})$ for $d \in \Re^n_+$ satisfying $Q - \text{diag}(d) \succeq 0$, and the tightest dual bound can be found via solving the following problem:

$$(\text{D}) \qquad \max\{v(\text{D}(d)) \mid d \in \Re^n_+,\ Q - \text{diag}(d) \succeq 0\}. \tag{27}$$

**Proposition 1** *Let* (QP) *denote the continuous relaxation of* ($\text{MIQP}_0$)*. Then, for any fixed* $d \in \Re^n_+$ *satisfying* $Q - \text{diag}(d) \succeq 0$*, it holds*

$$v(\text{D}) \ge v(\text{D}(d)) \ge v(\text{QP}). \tag{28}$$

Proof. The first inequality is obvious. Let $d \in \Re^n_+$ satisfy $Q - \text{diag}(d) \succeq 0$. (QP) can be reformulated as

$$\begin{aligned} &\min\ x^T \text{diag}(d) x + c^T x + z^T (Q - \text{diag}(d)) z \\ &\text{s.t. } Az \le b, \\ &\qquad x = z, \\ &\qquad e^T y \le K,\ y \in [0,1]^n, \\ &\qquad \alpha_i y_i \le x_i \le u_i y_i,\ i = 1, \dots, n. \end{aligned}$$

Applying the Lagrangian decomposition scheme to the above problem in a similar way as we did for problem (23) and using the strong duality of convex quadratic program, we obtain

$$v(\mathrm{QP}) = \max\{\hat{d}_1(\lambda) + d_2(\lambda) \mid \lambda \in \Re^n\},$$

where $d_2(\lambda)$ is defined in (25) and

$$\begin{aligned} \hat{d}_1(\lambda) \quad = \quad & \min \ x^T \mathrm{diag}(d)x + (c-\lambda)^T x \\ & \text{s.t. } e^T y \le K, \ y \in [0,1]^n, \\ & \quad \alpha_i y_i \le x_i \le u_i y_i, \ i = 1, \ldots, n. \end{aligned}$$

Note that the above problem is a continuous relaxation of the subproblem in (24). Thus, $\hat{d}_1(\lambda) \le d_1(\lambda)$. Consequently,

$$v(\mathrm{QP}) = \max\{\hat{d}_1(\lambda) + d_2(\lambda) \mid \lambda \in \Re^n\} \le \max\{d_1(\lambda) + d_2(\lambda) \mid \lambda \in \Re^n\} = v(\mathrm{D}(d)),$$

which is the second inequality of (28). □

**Remark 1** In Shaw et al. (2008), a Lagrangian decomposition scheme similar to (24) and (25) is applied to a cardinality constrained portfolio selection problem with objective function $x^T Qx + c^{\mathrm{T}} x$, where $Q = H^T H + \mathrm{diag}(d)$, and without the minimum threshold constraints. The dual problem in Shaw et al. (2008) is solved by subgradient method to obtain a lower bound for fixed $H$ and $d$. The dual problem (D) can be viewed as a generalized and strengthened version of the Lagrangian dual in Shaw et al. (2008).

In the following, we show that problem (D($d$)) and thus (D) can be reduced to an SDP problem. Let

$$q_i = \min_{x_i \in \{0\} \cup [\alpha_i, u_i]} \{d_i x_i^2 + (c_i - \lambda_i)x_i\}, \ i = 1, \ldots, n. \tag{29}$$

Let $q = (q_1, \ldots, q_n)^T$. We see that $d_1(\lambda)$ defined in (24) is equal to the sum of the $K$ smallest elements of $q$. Denote by $S_K(x)$ the sum of the $K$ largest elements of $x \in \Re^n$. Since $q_i \le 0$ ($i = 1, \ldots, n$), we have

$$d_1(\lambda) = \max\{-t \mid -S_K(-q) \ge -t\} = \max\{-t \mid S_K(-q) \le t\}. \tag{30}$$

The following lemma is a special case of the linear matrix inequality representation of the sum of $K$ largest eigenvalues of a symmetric matrix (see, e.g., Page 147, Ben-Tal and Nemirovski (2001)). For the sake of self-containedness, we give here a simple proof of the lemma.

**Lemma 1** *For any vector $p \in \Re^n_+$, the following two sets, $\Gamma_1$ and $\Gamma_2$, are identical:*

$$\begin{aligned}\Gamma_1 &= \{(p,t) \mid S_K(p) \le t\},\\ \Gamma_2 &= \{(p,t) \mid \exists\ (\pi,s) \in \Re^n \times \Re \ satisfying\ (a),\ (b)\ and\ (c)\},\end{aligned}$$

*where*

(*a*) $t - Ks - e^T\pi \ge 0$;

(*b*) $\pi \ge 0,\ s \ge 0$;

(*c*) $\pi - p + se \ge 0$.

Proof. For any $(p,t) \in \Gamma_2$, there exists $(\pi, s) \in \Re^n \times \Re$ satisfying $(a)$, $(b)$ and $(c)$. By $(c)$, we have $S_K(p) \le S_K(\pi + se) = S_K(\pi) + sK$. Together with $(b)$, we get $S_K(p) \le e^T\pi + sK$, and then by $(a)$, we have $S_K(p) \le t$. Therefore, $(p,t) \in \Gamma_1$, i.e., $\Gamma_2 \subseteq \Gamma_1$.

Conversely, for any $(p,t) \in \Gamma_1$, we rank $p_i$ $(i = 1,\dots,n)$ in a descending order: $p_{i_1} \ge p_{i_2} \ge \cdots \ge p_{i_n} \ge 0$. Let $s = p_{i_K}$. For $j = 1,\dots,n$, let

$$h_{i_j} = \begin{cases} p_{i_j} - p_{i_K}, & \text{if } j \le K,\\ 0, & \text{otherwise.}\end{cases}$$

It is easy to verify that $(p,t,\pi,s)$ satisfies $(b)$ and $(c)$. Notice that

$$Ks + e^T\pi = Kp_{i_K} + \sum_{j=1}^{K}(p_{i_j} - p_{i_K}) = \sum_{j=1}^{K} p_{i_j} = S_K(p).$$

Thus, $t - Ks - e^T\pi = t - S_k(p) \ge 0$, i.e., $(a)$ holds, thus yielding $(p,t) \in \Gamma_2$. Therefore, $\Gamma_1 \subseteq \Gamma_2$. □

**Lemma 2** *The value of $d_1(\lambda)$ in (30) is equal to the optimal value of the following SDP problem:*

$$\begin{array}{lll}(\mathrm{D}_1) & \max & -Ks - e^T\pi \\ & \text{s.t.} & \begin{pmatrix} d_i + \mu_i & \frac{1}{2}(c_i - \lambda_i - \beta_i\mu_i) \\ \frac{1}{2}(c_i - \lambda_i - \beta_i\mu_i) & \pi_i + s + \alpha_i u_i \mu_i \end{pmatrix} \succeq 0, \quad i = 1,\dots,n, \\ & & (s,\mu,\pi) \in \Re_+ \times \Re^n_+ \times \Re^n_+,\end{array}$$

*where $\beta_i := \alpha_i + u_i$ for $i = 1,\dots,n$.*

Proof. By (30) and Lemma 1, $d_1(\lambda)$ is equal to the optimal value of the following SDP problem:

$$(\tilde{\mathrm{D}}_1) \qquad \begin{aligned} \max\ & -Ks - e^T\pi \\ \text{s.t. } & \pi \geq 0,\ s \geq 0 \\ & \pi + q + se \geq 0, \end{aligned}$$

where $q = (q_1, \ldots, q_n)^T$ with $q_i$ being defined in (29). Also, we can rewrite $q_i$ as

$$\begin{aligned} q_i &= \min\{d_i x_i^2 + (c_i - \lambda_i)x_i \mid x_i \in \{0\} \cup [\alpha_i, u_i]\} \\ &= \min\{0, \min_{x_i \in [\alpha_i, u_i]} d_i x_i^2 + (c_i - \lambda_i)x_i\} \\ &= \max\ - w_i \\ &\quad\ \text{s.t. } - w_i \leq 0, \\ &\qquad\quad - w_i \leq \min\{d_i x_i^2 + (c_i - \lambda_i)x_i \mid x_i^2 - (\alpha_i + u_i)x_i + \alpha_i u_i \leq 0\} \\ &= \max\ - w_i \\ &\quad\ \text{s.t. } w_i \geq 0,\ \mu_i \geq 0, \\ &\qquad \begin{pmatrix} d_i + \mu_i & \frac{1}{2}(c_i - \lambda_i - \mu_i(\alpha_i + u_i)) \\ \frac{1}{2}(c_i - \lambda_i - \mu_i(\alpha_i + u_i)) & w_i + \alpha_i u_i \mu_i \end{pmatrix} \succeq 0, \end{aligned}$$

where $\mu_i \geq 0$ is the multiplier for the constraint: $x_i^2 - (\alpha_i + u_i)x_i + \alpha_i u_i \leq 0$ and the last equality holds due to $\mathcal{S}$-Lemma (see, e.g., Pólik and Terlaky (2007)). Thus, the constraint $\pi + q + se \geq 0$ can be rewritten as

$$\begin{aligned} & -\pi_i - s \leq -w_i, \\ & w_i \geq 0,\ \mu_i \geq 0, \\ & \begin{pmatrix} d_i + \mu_i & \frac{1}{2}(c_i - \lambda_i - \mu_i(\alpha_i + u_i)) \\ \frac{1}{2}(c_i - \lambda_i - \mu_i(\alpha_i + u_i)) & w_i + \alpha_i u_i \mu_i \end{pmatrix} \succeq 0, \end{aligned}$$

for $i = 1, \ldots, n$. Since $s \geq 0$ and $\pi_i \geq 0$ $(i = 1, \ldots, n)$, it must hold $-w_i = -\pi_i - s$ at the optimal solution of $(\tilde{\mathrm{D}}_1)$. Therefore, $(\tilde{\mathrm{D}}_1)$ is equivalent to $(\mathrm{D}_1)$ and hence $d_1(\lambda)$ is equal to the optimal value of $(\mathrm{D}_1)$. □

**Theorem 2** *For any fixed $d \in \Re^n_+$ satisfying $Q - \mathrm{diag}(d) \succeq 0$, the dual problem $(\mathrm{D}(d))$ in (26) is equivalent to the SDP problem $(\mathrm{D_s}(d))$ in (22) with*

$$v(\mathrm{D}(d)) = v(\mathrm{D_s}(d)) = v(\overline{\mathrm{SOCP}}(d)). \tag{31}$$

Proof. Since $Q - \mathrm{diag}(d) \succeq 0$, the subproblem in (25) is a convex quadratic program and hence the strong duality holds. Thus, we can express $d_2(\lambda)$ as the optimal value of the dual problem of (25) which has the following SDP representation:

$$
\begin{aligned}
(\mathrm{D}_2) \quad d_2(\lambda) \;=\; & \max \; -\tau \\
& \text{s.t.} \begin{pmatrix} Q - \mathrm{diag}(d) & \frac{1}{2}(\lambda + A^T\eta) \\ \frac{1}{2}(\lambda + A^T\eta)^T & -\eta^T b + \tau \end{pmatrix} \succeq 0, \\
& \eta \geq 0,
\end{aligned}
$$

where $\eta \in \Re^m_+$ is the multiplier for the constraints $Ax \leq b$ in (25). It then follows from Lemma 2 that the dual problem $(\mathrm{D}(d))$ in (26) is equivalent to problem $(\mathrm{D_s}(d))$ with $v(\mathrm{D}(d)) = v(\mathrm{D_s}(d))$. The second equality in (31) follows from the strong duality between $(\mathrm{D_s}(d))$ and $(\overline{\mathrm{SOCP}}_1(d))$, which is equivalent to $(\overline{\mathrm{SOCP}}(d))$ (see the proof of Theorem 1). □

Theorem 2 and its proof suggest a new way of deriving the SOCP reformulation $(\mathrm{SOCP}(d))$. We first construct the SDP relaxation $(\mathrm{D}(d))$ of $(\mathrm{MIQP}_0)$ by the Lagrangian decomposition scheme (24)-(25) and then write the conic dual of $(\mathrm{D}(d))$, which is exactly $(\overline{\mathrm{SOCP}}(d))$. Changing $y \in [0,1]^n$ in $(\overline{\mathrm{SOCP}}(d))$ to $y \in \{0,1\}^n$, we obtain $(\mathrm{SOCP}(d))$. Theorem 2 also reveals that the continuous bound of $(\mathrm{SOCP}(d))$ is nothing but the dual bound of (P) via the Lagrangian decomposition scheme (24)-(25). In some sense, this SDP approach for deriving the SOCP reformation $(\mathrm{SOCP}(d))$ is "dual" to the perspective reformulation method in Frangioni and Gentile (2006, 2007) where the perspective function is used to construct the convex envelope of the objective function on the primal (semi-continuous) variables.

# 4. Computational Results

In this section, we conduct computational experiments to demonstrate the effectiveness of the new SDP approach for computing the diagonal decomposition in perspective formulations. We will consider in our computational experiments test problems arising from portfolio selection and subset selection.

Recall that the perspective reformulation $(\mathrm{PR}(d))$ has two tractable reformulations: the SOCP reformulation $(\mathrm{SOCP}(d))$ and the perspective cut reformulation $(\mathrm{PC}(d))$. Although these two reformulations are equivalent to each other in terms of the optimal solutions and the continuous bounds, they may have quite different computational performance when implemented and solved by branch-and-cut methods. Extensive computational results in

Frangioni and Gentile (2009) suggest that if properly implemented, the perspective cut reformulation (PC($d$)) is much more efficient than the SOCP reformulation (SOCP($d$)). We shall therefore focus in this section on the implementation of perspective cut reformulations using different choices of $d$.

## 4.1. Implementation issues

We consider the comparison of the perspective cut (P/C) reformulations (PC($d$)) using the following three choices of $d$:

- ($\mathrm{PC_s}$): the reformulation (PC($d$)) with $d = d_s$, where $d_s$ is computed by ($\mathrm{SDP_s}$);
- ($\mathrm{PC_l}$): the reformulation (PC($d$)) with $d = d_l$, where $d_l$ is computed by ($\mathrm{SDP_l}$);
- ($\mathrm{PC_c}$): the reformulation (PC($d$)) with $d$ being a convex combination: $d = \frac{1}{2}(d_s + d_l)$.

As suggested in Frangioni and Gentile (2009), we implemented a branch-and-cut method for the three P/C reformulations using `CPLEX 12.4` through C, where the dynamic generation of the perspective cuts was implemented by means of `cutcallback` procedures. We use `CPLEX` default settings for the branch-and-cut method, which leads to using `dual simplex QP optimizer` for solving the continuous relaxation subproblem at each node of the branch-and-cut tree.

The C programs were developed and compiled using Microsoft Visual Studio 2008. The SDP problems ($\mathrm{SDP_s}$) and ($\mathrm{SDP_l}$) were modeled by `CVX 1.2` (Grant and Boyd (2009)), a Matlab-based modeling system for convex optimization, and solved by `SeDuMi` 1.2 within `CVX`. The Matlab version for running `CVX` is 7.12.0 (R2011a, 64-bit). The numerical tests have been performed on a personal computer equipped with Intel Pentium G630 CPU (2.70 GHz) and 8 GB of RAM, running Windows 7 (64-bit). All the data files of the test problems, the `CVX` Matlab codes and the C program codes in our numerical tests are available at `http://my.gl.fudan.edu.cn/teacherhome/xlsun/ccqp/`.

## 4.2. Portfolio selection problems

Let $\mu$ and $Q$ be the mean and covariance matrix of $n$ risky assets, respectively. The mean-variance portfolio selection problem with cardinality and minimum threshold constraints can be formulated as

$$
\begin{aligned}
\text{(MV)} \qquad & \min \ x^T Q x \\
& \text{s.t. } e^T x = 1,\ \mu^T x \geq \rho, \\
& \quad |\text{supp}(x)| \leq K, \\
& \quad x_i \geq \alpha_i,\ \forall i \in \text{supp}(x), \\
& \quad 0 \leq x_i \leq u_i,\ i = 1, \ldots, n.
\end{aligned}
$$

To build the test problems of (MV), we use the 90 instances of mean-variance portfolio selection problem with semi-continuous variables created by Frangioni and Gentile (2007), 30 instances each for $n = 200$, 300 and 400. The 30 instances for each $n$ are divided into three subsets denoted by $n^+$, $n^0$ and $n^-$, 10 instances in each subset, with different diagonal dominance in matrix $Q$. The parameters $\rho$, $\alpha_i$ and $u_i$ are uniformly drawn at random from intervals $[0.002, 0.01]$, $[0.075, 0.125]$ and $[0.375, 0.425]$, respectively. The data files of these instances are available at `http://www.di.unipi.it/optimize/Data/MV.html`. Since $\alpha_i \geq 0.075$, the maximum number of nonzero variables in a feasible solution to (MV) is at most 13. Adding a cardinality constraint with $K = 6, 8, 10, 12$, respectively, to each of the 90 instances and considering the 90 instances without cardinality constraint, we then have 450 instances of (MV).

We first compare the continuous bounds of (PC($d$)) with $d = d_s$ and $d = d_l$ for the 450 instances of (MV). Since the continuous relaxations of (PC($d$)) and (SOCP($d$)) are equivalent, the continuous bound of (PC($d$)) can be computed by solving the SOCP problem ($\overline{\text{SOCP}}(d)$). The comparison results are reported in Table 1, where $b_s$ denotes the average continuous bound of (PC($d$)) with $d = d_s$ for the 10 instances, $b_l$ is the average continuous bound of (PC($d$)) with $d = d_l$ for the 10 instances, and "imp.ratio" denotes the average (relative) improvement ratio of $b_l$ over $b_s$ defined by imp.ratio=$(b_l - b_s)/b_s$ (%). We see from Table 1 that the improvement ratios vary from 0.10% to 5.11% for different types of instances and different $K$. Among the three types of instances with different diagonal dominance of $Q$, the instances of $n^+$ type have the smallest improvement of continuous bounds while the instances of $n^-$ type have the largest improvement. We also observe that the improvement ratio of $b_l$ over $b_s$ tends to decrease as the cardinality $K$ increases. Since only a finite number of perspective cuts in (PC($d$)) are used in the subproblems of the branch-and-cut method for solving (PC($d$)), the continuous bounds at the root node and subnodes of the branch-and-cut method are usually weaker than those of (PC($d$)). Nevertheless, the improvement of the

Table 1: Comparison of continuous bounds for (MV)

| Problem | $K$ | $b_s$ | $b_l$ | imp.ratio | Problem | $K$ | $b_s$ | $b_l$ | imp.ratio | Problem | $K$ | $b_s$ | $b_l$ | imp.ratio |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 6 | 344.08 | 346.17 | 0.61 | | 6 | 117.32 | 121.54 | 3.60 | | 6 | 84.20 | 88.49 | 5.11 |
| | 8 | 261.59 | 262.70 | 0.42 | | 8 | 90.29 | 92.95 | 2.95 | | 8 | 65.45 | 68.21 | 4.24 |
| $200^+$ | 10 | 214.58 | 215.48 | 0.42 | $200^0$ | 10 | 74.62 | 76.44 | 2.44 | $200^-$ | 10 | 54.50 | 56.42 | 3.56 |
| | 12 | 192.71 | 193.18 | 0.24 | | 12 | 67.40 | 68.40 | 1.48 | | 12 | 49.41 | 50.59 | 2.42 |
| | nc | 191.88 | 192.29 | 0.21 | | nc | 67.33 | 68.23 | 1.34 | | nc | 49.30 | 50.36 | 2.17 |
| | 6 | 505.57 | 509.12 | 0.70 | | 6 | 176.11 | 181.98 | 3.35 | | 6 | 135.16 | 140.80 | 4.24 |
| | 8 | 382.39 | 384.47 | 0.54 | | 8 | 134.68 | 138.47 | 2.83 | | 8 | 104.24 | 107.91 | 3.60 |
| $300^+$ | 10 | 310.57 | 312.00 | 0.46 | $300^0$ | 10 | 110.70 | 113.14 | 2.22 | $300^-$ | 10 | 86.54 | 89.12 | 3.06 |
| | 12 | 269.24 | 269.93 | 0.26 | | 12 | 98.27 | 99.50 | 1.25 | | 12 | 79.57 | 80.99 | 1.83 |
| | nc | 267.07 | 267.44 | 0.14 | | nc | 97.97 | 99.10 | 1.16 | | nc | 79.43 | 80.72 | 1.65 |
| | 6 | 680.15 | 684.94 | 0.71 | | 6 | 233.33 | 240.29 | 2.99 | | 6 | 164.17 | 171.64 | 4.57 |
| | 8 | 514.68 | 517.26 | 0.51 | | 8 | 177.83 | 182.30 | 2.52 | | 8 | 125.70 | 130.67 | 3.97 |
| $400^+$ | 10 | 419.00 | 420.43 | 0.35 | $400^0$ | 10 | 145.43 | 148.52 | 2.13 | $400^-$ | 10 | 103.35 | 106.91 | 3.46 |
| | 12 | 367.30 | 367.95 | 0.18 | | 12 | 128.37 | 130.03 | 1.30 | | 12 | 91.37 | 93.35 | 2.18 |
| | nc | 364.50 | 364.88 | 0.10 | | nc | 127.55 | 128.83 | 1.01 | | nc | 90.88 | 92.40 | 1.68 |

"nc" denotes the instances without cardinality constraint

continuous bounds of (PC($d$)) by using $d = d_l$ does have an impact on the performance of the branch-and-cut method for (PC($d$)), as will be seen below.

Table 2 summarizes the numerical results of the three P/C reformulations for the 450 instances of (MV), where the time limit of `CPLEX` is set as 10000 seconds. The results in Table 2 are average for the 10 instances in each subset of the 450 instances. The notations in Table 2 are explained as follows. The columns "$\mathrm{time_s}$" and "$\mathrm{time_l}$" are the computation time (in seconds) for finding parameter vector $d$ via solving SDP problems ($\mathrm{SDP_s}$) and ($\mathrm{SDP_l}$) using `CVX`, respectively. The column "gap" is an output parameter of `CPLEX 12.4` which measures the relative gap (in percentage) of the incumbent solution when `CPLEX 12.4` is terminated. The number in parenthesis next to the gap is the number of unsolved instances within 10000 seconds. Note that the default tolerance of relative gap in `CPLEX` 12.4 is 0.01%. Finally, the columns "time" and "nodes" are the computing time (in seconds) and the number of nodes explored by `CPLEX 12.4`, respectively.

From Table 2, we can see that the average computation time and the number of nodes of reformulation ($\mathrm{PC_l}$) are significantly less than those of ($\mathrm{PC_s}$) for all instances of types $n^0$ and $n^-$, while ($\mathrm{PC_s}$) performs slightly better than ($\mathrm{PC_l}$) for instances of type $n^+$. Moreover, ($\mathrm{PC_l}$) appears to be particularly advantageous over ($\mathrm{PC_s}$) for instances of types $n^0$ and $n^-$ with small cardinality ($K = 6, 8, 10$); indeed, ($\mathrm{PC_l}$) is at least one order faster than ($\mathrm{PC_s}$) for these instances. This is consistent with the trends of improvement ratios of the continuous bounds of ($\mathrm{PC_l}$) over those of ($\mathrm{PC_s}$) (see Table 1). It can be also noticed from Table 2 that the average computation time of the "large" SDP problem ($\mathrm{SDP_l}$) is larger than that of the "small" SDP problem ($\mathrm{SDP_s}$). Nevertheless, the computation time of ($\mathrm{SDP_l}$) is no more than

Table 2: Comparison results of P/C reformulations for (MV)

| Problem | $K$ | $\text{time}_\text{s}$ | $\text{time}_\text{l}$ | $(\text{PC}_\text{s})$ | | | $(\text{PC}_\text{l})$ | | | $(\text{PC}_\text{c})$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | gap | time | nodes | gap | time | nodes | gap | time | nodes |
| $200^+$ | 6 | 2.88 | 8.55 | 0.00 | 3.49 | 34 | 0.00 | 3.41 | 27 | 0.00 | 2.37 | 18 |
| | 8 | 2.54 | 8.76 | 0.00 | 2.71 | 23 | 0.00 | 3.16 | 24 | 0.00 | 2.15 | 12 |
| | 10 | 2.57 | 8.49 | 0.00 | 2.66 | 33 | 0.01 | 2.62 | 40 | 0.00 | 1.97 | 24 |
| | 12 | 2.52 | 8.62 | 0.00 | 4.00 | 247 | 0.01 | 3.83 | 192 | 0.00 | 2.88 | 136 |
| | nc | 2.68 | 7.80 | 0.01 | 38.21 | 5199 | 0.01 | 14.09 | 1264 | 0.01 | 60.57 | 7286 |
| $200^0$ | 6 | 2.62 | 8.16 | 0.00 | 43.19 | 1278 | 0.00 | 5.72 | 86 | 0.00 | 13.72 | 351 |
| | 8 | 2.57 | 8.00 | 0.00 | 73.80 | 2706 | 0.00 | 5.63 | 77 | 0.00 | 17.71 | 525 |
| | 10 | 2.55 | 7.74 | 0.00 | 66.59 | 2922 | 0.00 | 4.38 | 100 | 0.00 | 13.86 | 550 |
| | 12 | 2.58 | 7.92 | 0.00 | 30.88 | 2560 | 0.00 | 16.31 | 1329 | 0.01 | 24.28 | 1783 |
| | nc | 2.66 | 7.59 | 0.01 | 44.93 | 3913 | 0.01 | 25.26 | 1983 | 0.01 | 29.33 | 2318 |
| $200^-$ | 6 | 2.52 | 7.99 | 0.00 | 153.15 | 3936 | 0.00 | 16.15 | 308 | 0.00 | 58.02 | 1425 |
| | 8 | 2.45 | 7.94 | 0.01 | 286.64 | 8119 | 0.00 | 15.56 | 286 | 0.00 | 72.43 | 1941 |
| | 10 | 2.52 | 8.15 | 0.01 | 315.68 | 10870 | 0.00 | 9.74 | 260 | 0.00 | 52.64 | 1915 |
| | 12 | 2.46 | 7.80 | 0.01 | 182.20 | 17084 | 0.00 | 86.12 | 10545 | 0.00 | 142.59 | 11989 |
| | nc | 2.52 | 7.65 | 0.01 | 452.40 | 49213 | 0.01 | 136.47 | 19029 | 0.01 | 477.37 | 45470 |
| $300^+$ | 6 | 7.57 | 20.82 | 0.00 | 8.24 | 62 | 0.00 | 8.70 | 41 | 0.00 | 5.30 | 22 |
| | 8 | 7.53 | 21.09 | 0.00 | 6.68 | 67 | 0.00 | 7.31 | 32 | 0.00 | 4.79 | 23 |
| | 10 | 7.54 | 20.86 | 0.00 | 7.57 | 95 | 0.01 | 5.23 | 27 | 0.00 | 4.77 | 35 |
| | 12 | 7.53 | 22.22 | 0.00 | 5.80 | 64 | 0.00 | 7.29 | 105 | 0.00 | 4.69 | 50 |
| | nc | 7.54 | 18.73 | 0.01 | 101.04 | 6202 | 0.01 | 160.85 | 7762 | 0.00 | 69.20 | 3411 |
| $300^0$ | 6 | 7.63 | 21.05 | 0.00 | 151.36 | 2611 | 0.00 | 16.68 | 143 | 0.00 | 45.55 | 669 |
| | 8 | 7.90 | 20.00 | 0.01 | 321.13 | 6738 | 0.00 | 18.12 | 152 | 0.00 | 76.65 | 1239 |
| | 10 | 7.77 | 20.18 | 0.00 | 226.42 | 4906 | 0.00 | 9.30 | 97 | 0.00 | 35.30 | 721 |
| | 12 | 7.61 | 20.82 | 0.00 | 44.17 | 1566 | 0.00 | 22.25 | 622 | 0.00 | 48.56 | 1442 |
| | nc | 7.63 | 19.23 | 0.01 | 289.32 | 12310 | 0.01 | 118.34 | 5627 | 0.01 | 258.30 | 9386 |
| $300^-$ | 6 | 7.48 | 22.20 | 0.00 | 609.64 | 9233 | 0.00 | 32.67 | 322 | 0.01 | 202.38 | 2807 |
| | 8 | 7.79 | 22.24 | 0.01 | 1549.83 | 26204 | 0.00 | 31.38 | 327 | 0.01 | 324.25 | 4825 |
| | 10 | 7.70 | 20.83 | 0.01 | 972.67 | 20578 | 0.00 | 24.43 | 391 | 0.01 | 170.06 | 3353 |
| | 12 | 7.49 | 21.88 | 0.01 | 47.09 | 1458 | 0.00 | 13.12 | 420 | 0.00 | 20.66 | 650 |
| | nc | 7.47 | 20.79 | 0.01 | 323.40 | 12473 | 0.01 | 125.27 | 6381 | 0.01 | 360.12 | 13360 |
| $400^+$ | 6 | 18.60 | 47.87 | 0.00 | 20.44 | 147 | 0.00 | 20.42 | 63 | 0.00 | 10.54 | 48 |
| | 8 | 18.13 | 48.08 | 0.00 | 16.74 | 157 | 0.00 | 17.43 | 68 | 0.00 | 10.51 | 58 |
| | 10 | 18.09 | 45.42 | 0.00 | 10.44 | 49 | 0.01 | 11.25 | 48 | 0.00 | 6.36 | 20 |
| | 12 | 19.88 | 52.63 | 0.00 | 8.88 | 78 | 0.00 | 11.34 | 127 | 0.00 | 6.89 | 63 |
| | nc | 17.94 | 38.60 | 0.01 | 795.78 | 27427 | 0.01 | 1060.14 | 28868 | 0.01 | 579.53 | 16495 |
| $400^0$ | 6 | 17.87 | 44.07 | 0.01 | 355.55 | 3901 | 0.00 | 42.59 | 246 | 0.00 | 127.33 | 1128 |
| | 8 | 18.18 | 44.19 | 0.01 | 570.61 | 7860 | 0.00 | 34.64 | 188 | 0.00 | 150.44 | 1423 |
| | 10 | 18.12 | 43.61 | 0.01 | 996.21 | 12005 | 0.00 | 26.67 | 216 | 0.01 | 122.02 | 1442 |
| | 12 | 18.41 | 45.75 | 0.00 | 67.06 | 1088 | 0.00 | 40.35 | 635 | 0.00 | 52.67 | 790 |
| | nc | 17.65 | 49.76 | 0.03(1) | 1750.72 | 41886 | 0.01 | 781.63 | 23308 | 0.06(1) | 2017.53 | 38985 |
| $400^-$ | 6 | 18.42 | 44.47 | 0.01 | 2258.11 | 21906 | 0.00 | 95.40 | 652 | 0.01 | 656.19 | 5793 |
| | 8 | 18.06 | 42.69 | 0.22(2) | 3487.72 | 39109 | 0.01 | 81.72 | 547 | 0.01 | 1367.03 | 12509 |
| | 10 | 17.94 | 40.88 | 0.17(2) | 4292.37 | 46773 | 0.00 | 44.12 | 377 | 0.01 | 632.66 | 7397 |
| | 12 | 18.11 | 44.26 | 0.01 | 248.93 | 4102 | 0.00 | 30.17 | 416 | 0.00 | 46.22 | 724 |
| | nc | 17.73 | 44.20 | 0.02(1) | 4623.92 | 104261 | 0.06(1) | 1887.66 | 60738 | 0.12(2) | 4564.43 | 88673 |

60 seconds even for the large-size instances with $n = 400$, which is often neglectable when compared with the computing time of the branch-and-cut method. Figure 1 further displays the trends of total computing time and the number of nodes of the three reformulations for (MV), where the total computing time is the sum of the average time for solving the SDP problem and the average time for solving the corresponding P/C reformulation. As expected, the performance of $(\text{PC}_\text{c})$ falls in between $(\text{PC}_\text{s})$ and $(\text{PC}_\text{l})$ for most instances in terms of the computation time and the number of nodes used. Interestingly, we observe that $(\text{PC}_\text{c})$ performs better than both $(\text{PC}_\text{s})$ and $(\text{PC}_\text{l})$ for instances of type $300^+$ and $400^+$ when there

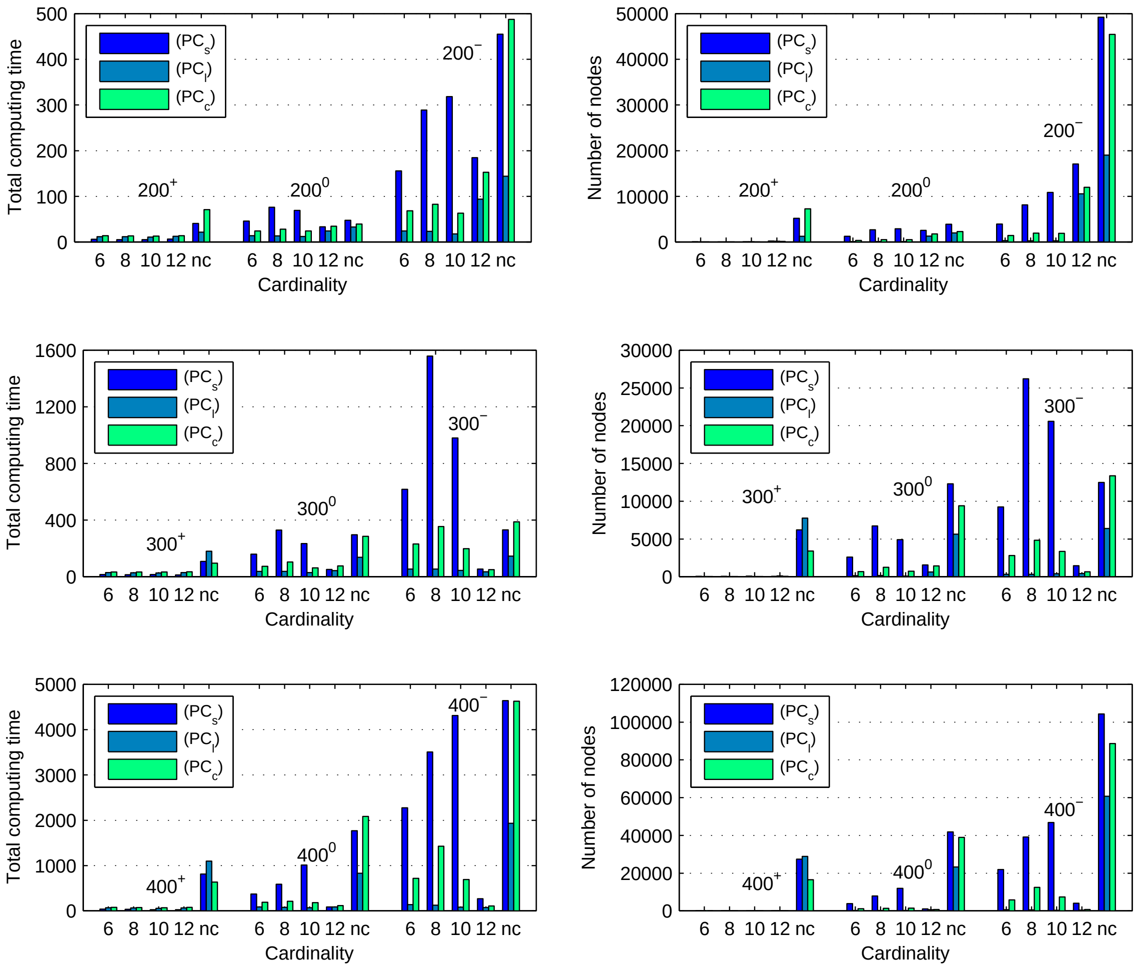


Figure 1: Total computing time and number of nodes for (MV)

is no cardinality constraint. For instances of $n^+$ type with cardinality constraint, ($\mathrm{PC_l}$) does not have advantage over ($\mathrm{PC_s}$) in terms of the total computing time.

Next, we consider 5 instances of (MV) from OR-Library, where the mean vector $\mu$ and the covariance matrix $Q$ were estimated using real data from DAX 100 (Germany), FTSE 100 (UK), S&P 100 (USA), Nikkei 225 (Japan) and S&P 500 (USA). The data files of $\mu$ and $Q$ are available at `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`. In our test, we set $\alpha_i = 0.075$ and $u_i = 0.4$ for each $i$. As in Cesarone et al. (2009), the parameter $\rho$ is set in the following manner: Let $\rho_{\min} = \mu^T x^*$ with $x^*$ being the optimal solution to the minimum-risk problem: $\min\{x^T Qx \mid e^T x = 1, 0 \le x \le u\}$. Let $\rho_{\max} = \max\{\mu^T x \mid e^T x = 1, 0 \le x \le u\}$. Set $\rho = \rho_{\min} + 0.3(\rho_{\max} - \rho_{\min})$.

Numerical results of the three P/C reformulations for the 5 instances with different

Table 3: Numerical results of P/C reformulations for (MV) with real data

| Problem | $n$ | $K$ | $\text{time}_\text{s}$ | $\text{time}_\text{l}$ | $(\text{PC}_\text{s})$ gap | time | nodes | $(\text{PC}_\text{l})$ gap | time | nodes | $(\text{PC}_\text{c})$ gap | time | nodes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DAX 100 | 85 | 5 | 0.62 | 1.67 | 0.00 | 48.19 | 2533 | 0.01 | 56.05 | 4556 | 0.00 | 54.51 | 3057 |
| | | 7 | 0.46 | 1.82 | 0.00 | 28.39 | 1394 | 0.00 | 22.56 | 1524 | 0.00 | 26.52 | 1235 |
| | | 9 | 0.42 | 1.76 | 0.00 | 7.74 | 458 | 0.00 | 4.77 | 355 | 0.00 | 6.68 | 384 |
| | | nc | 0.51 | 1.58 | 0.00 | 0.55 | 126 | 0.00 | 0.55 | 152 | 0.00 | 0.62 | 139 |
| FTSE 100 | 89 | 7 | 0.46 | 1.80 | 0.01 | 28.33 | 1191 | 0.01 | 42.53 | 2554 | 0.00 | 27.33 | 1160 |
| | | 9 | 0.56 | 1.87 | 0.00 | 10.22 | 548 | 0.00 | 7.30 | 508 | 0.00 | 9.95 | 534 |
| | | 11 | 0.50 | 1.75 | 0.00 | 0.90 | 87 | 0.00 | 0.34 | 25 | 0.00 | 0.50 | 36 |
| | | nc | 0.48 | 1.69 | 0.00 | 0.64 | 94 | 0.00 | 0.34 | 61 | 0.00 | 0.50 | 75 |
| S&P 100 | 98 | 8 | 0.51 | 2.16 | 0.01 | 457.47 | 14249 | 0.01 | 449.75 | 17776 | 0.01 | 415.60 | 12919 |
| | | 10 | 0.54 | 2.29 | 0.01 | 301.16 | 10918 | 0.01 | 201.29 | 10212 | 0.01 | 264.77 | 9743 |
| | | 12 | 0.58 | 2.33 | 0.00 | 45.44 | 3677 | 0.00 | 19.42 | 1722 | 0.00 | 31.54 | 2341 |
| | | nc | 0.53 | 2.03 | 0.01 | 20.90 | 3456 | 0.01 | 9.91 | 2342 | 0.01 | 13.59 | 2409 |
| Nikkei 225 | 225 | 6 | 3.86 | 9.71 | 0.00 | 206.83 | 13760 | 0.01 | 202.11 | 16731 | 0.01 | 323.86 | 20888 |
| | | 8 | 3.91 | 10.15 | 0.01 | 72.99 | 4397 | 0.01 | 47.41 | 3437 | 0.01 | 52.04 | 3290 |
| | | 10 | 3.98 | 10.34 | 0.00 | 5.84 | 372 | 0.00 | 3.60 | 229 | 0.00 | 4.73 | 265 |
| | | nc | 3.91 | 10.36 | 0.00 | 0.98 | 108 | 0.00 | 1.06 | 133 | 0.00 | 0.86 | 116 |
| S&P 500 | 458 | 7 | 25.01 | 56.50 | 0.01 | 2407.66 | 38950 | 0.01 | 2235.88 | 61697 | 0.01 | 1454.79 | 27136 |
| | | 9 | 25.02 | 60.21 | 0.01 | 513.04 | 8586 | 0.01 | 414.25 | 12448 | 0.01 | 445.35 | 7968 |
| | | 11 | 25.01 | 63.75 | 0.01 | 125.85 | 2687 | 0.00 | 92.96 | 2655 | 0.01 | 84.21 | 1792 |
| | | nc | 25.04 | 57.37 | 0.00 | 25.65 | 1203 | 0.00 | 29.05 | 1731 | 0.00 | 24.18 | 1242 |

cardinality $K$ and the case without cardinality constraint are summarized in Table 3. We see from Table 3 that the reformulation $(\text{PC}_\text{l})$ slightly outperforms $(\text{PC}_\text{s})$ and $(\text{PC}_\text{c})$ for most of the instances, while $(\text{PC}_\text{s})$ and $(\text{PC}_\text{c})$ can be also solved faster than $(\text{PC}_\text{l})$ by `CPLEX` for some instances.

## 4.3. Subset selection problems

In this subsection, we compare the performance of P/C reformulations for subset selection problems in multivariate regression (see Arthanari and Dodge (1993); Miller (2002)). As discussed in Section 1, for given $m$ data points $(a_i, b_i)$ with $a_i \in \Re^n$ and $b_i \in \Re$, the optimization model for the subset selection problem has the following form:

$$(\text{SSP}) \qquad \min\{\|Ax - b\|^2 \mid |\text{supp}(x)| \le K\},$$

where $A^T = (a_1, \ldots, a_m)$ and $b \in \Re^m$. We also consider a useful variant of (SSP) where $x$ is required to be nonnegative (see Breiman (1995) and Yuan and Lin (2007)), i.e.,

$$(\text{SSP}^+) \qquad \min\{\|Ax - b\|^2 \mid |\text{supp}(x)| \le K,\ x \ge 0\}.$$

In our test, the data in (SSP) and $(\text{SSP}^+)$ are randomly generated in the following fashion. For a fixed $n$, we generate $m = 2n$ data points $(a_i, b_i)$, $i = 1, \ldots, m$. The elements of $a_i$ are generated from the normal distribution $N(0, 1)$ and $b = A\bar{x} + \epsilon$, where $\epsilon = (\epsilon_1, \ldots, \epsilon_m)^T$ and

Table 4: Comparison of continuous bounds for (SSP) and ($SSP^+$)

| (SSP) | | | | | ($SSP^+$) | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $n$ | $K$ | $b_s$ | $b_l$ | imp.ratio | $n$ | $K$ | $b_s$ | $b_l$ | imp.ratio |
| 50 | 5 | 8.23 | 9.61 | 18.15 | 50 | 5 | 12.55 | 13.29 | 6.32 |
| 50 | 10 | 7.43 | 8.62 | 17.17 | 50 | 10 | 19.98 | 20.55 | 2.83 |
| 50 | 15 | 6.64 | 7.48 | 12.73 | 50 | 15 | 23.26 | 23.53 | 1.01 |
| 50 | 20 | 5.79 | 6.62 | 16.11 | 50 | 20 | 33.33 | 33.48 | 0.42 |
| 100 | 5 | 15.14 | 17.54 | 15.96 | 100 | 5 | 27.28 | 29.21 | 7.32 |
| 100 | 10 | 15.66 | 18.08 | 15.63 | 100 | 10 | 49.09 | 51.09 | 4.25 |
| 100 | 15 | 15.78 | 17.59 | 11.28 | 100 | 15 | 62.06 | 64.19 | 3.46 |
| 100 | 20 | 13.09 | 15.11 | 15.69 | 100 | 20 | 68.33 | 70.17 | 2.76 |

$\epsilon_i$ is taken from the normal distribution $N(0, 1)$, $i = 1, \ldots, m$. The elements of the vector $\bar{x}$ are from the uniform distribution in $[-1, 1]$. In order to apply the P/C reformulations to the above two models, we need to set some sufficiently large bounds for $x_i$. In our test, we set $-100 \leq x_i \leq 100$ $(i = 1, \ldots, n)$ in (SSP) and $0 \leq x_i \leq 100$ $(i = 1, \ldots, n)$ in ($SSP^+$). Similar parameters settings were used in Bertsimas and Shioda (2009). Using the above data generation method, we build 80 instances of (SSP) and ($SSP^+$), 5 instances for each $n$ and $K$ with $n = 50, 100$ and $K = 5, 10, 15, 20$, respectively.

We first report in Table 4 the comparison results of the continuous bounds of (PC($d$)) with $d = d_s$ and $d = d_l$ for the 80 instances of (SSP) and ($SSP^+$), where the results are the average of the 5 instances for each $n$ and $K$. We observe from Table 4 that the average improvement ratio of $b_l$ over $b_s$ ranges between 10% and 20% for instances of (SSP) but is less than 8% for all instances of ($SSP^+$). From Table 4, we also see that there is a tendency for the improvement ratio to decrease as the cardinality $K$ increases, which is particularly notable for instances of ($SSP^+$). Table 5 summarizes the numerical results of the P/C reformulations for the 80 instances of (SSP) and ($SSP^+$), where the time limit of `CPLEX` is set as 10000 seconds. The results in Table 5 are average of the 5 instances for each $n$ and $K$ and the notations are the same as those in Table 2. We see from Table 5 that ($PC_l$) clearly outperforms ($PC_s$) in terms of the CPU time and the number of nodes for all instances of (SSP) and ($SSP^+$). The superior performance of ($PC_l$) over ($PC_s$) becomes more notable for the hard instances of (SSP) and ($SSP^+$) with $n = 100$ and $K = 15, 20$. Interestingly, we observe from Table 5 that ($PC_c$) can be solved faster than both ($PC_s$) and ($PC_l$) on average for all the instances of (SSP). Figure 2 further illustrates the total computing time and the number of nodes of the three P/C reformulations for instances of (SSP) and ($SSP^+$) with $n = 100$.

Table 5: Numerical results of P/C reformulations for (SSP) and (SSP+)

| Problem | $n$ | $K$ | time$_s$ | time$_l$ | (PC$_s$) gap | time | nodes | (PC$_l$) gap | time | nodes | (PC$_c$) gap | time | nodes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (SSP) | 50 | 5 | 0.24 | 1.45 | 0.00 | 1.75 | 97 | 0.00 | 2.50 | 96 | 0.00 | 1.59 | 78 |
| | 50 | 10 | 0.22 | 1.45 | 0.00 | 8.23 | 201 | 0.00 | 6.37 | 139 | 0.00 | 3.96 | 119 |
| | 50 | 15 | 0.20 | 1.32 | 0.00 | 7.68 | 180 | 0.00 | 7.90 | 153 | 0.00 | 5.33 | 135 |
| | 50 | 20 | 0.21 | 1.33 | 0.00 | 37.36 | 386 | 0.00 | 22.25 | 288 | 0.00 | 17.12 | 243 |
| | 100 | 5 | 0.59 | 5.15 | 0.00 | 28.34 | 463 | 0.00 | 15.66 | 200 | 0.00 | 10.38 | 168 |
| | 100 | 10 | 0.63 | 4.90 | 0.00 | 12.07 | 204 | 0.00 | 11.29 | 147 | 0.00 | 5.91 | 106 |
| | 100 | 15 | 0.66 | 4.84 | 0.01 | 3897.18 | 7963 | 0.00 | 1022.45 | 2119 | 0.00 | 748.02 | 2130 |
| | 100 | 20 | 0.59 | 4.67 | 0.29(3) | 6208.10 | 9075 | 0.33(2) | 4192.52 | 3334 | 0.10(1) | 2763.31 | 3871 |
| (SSP$^+$) | 50 | 5 | 0.25 | 1.10 | 0.00 | 0.39 | 53 | 0.00 | 0.36 | 46 | 0.00 | 0.23 | 40 |
| | 50 | 10 | 0.22 | 1.04 | 0.00 | 1.03 | 79 | 0.00 | 0.49 | 42 | 0.00 | 0.54 | 48 |
| | 50 | 15 | 0.21 | 1.02 | 0.00 | 0.25 | 28 | 0.00 | 0.22 | 25 | 0.00 | 0.20 | 24 |
| | 50 | 20 | 0.23 | 1.08 | 0.00 | 0.23 | 25 | 0.00 | 0.23 | 22 | 0.00 | 0.21 | 20 |
| | 100 | 5 | 0.74 | 3.62 | 0.00 | 14.77 | 415 | 0.00 | 9.06 | 232 | 0.00 | 7.89 | 232 |
| | 100 | 10 | 0.54 | 2.67 | 0.01 | 191.85 | 2149 | 0.00 | 35.18 | 577 | 0.00 | 62.23 | 808 |
| | 100 | 15 | 0.56 | 2.78 | 0.07(1) | 2150.20 | 5750 | 0.01 | 195.89 | 1338 | 0.01 | 322.43 | 1863 |
| | 100 | 20 | 0.58 | 2.79 | 0.19(1) | 3365.37 | 9169 | 0.12(1) | 2308.19 | 4315 | 0.09(1) | 2299.63 | 6018 |

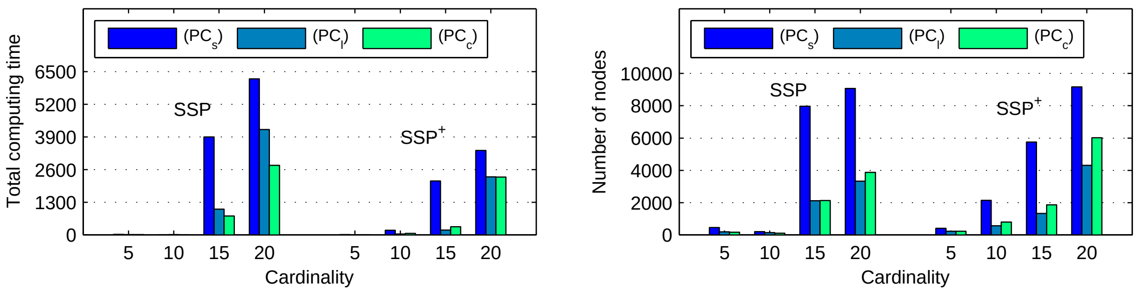


Figure 2: Total computing time and number of nodes for (SSP) and (SSP$^+$) with $n = 100$

# 5. Concluding Remarks

We have presented in this paper a semidefinite program approach to improve the performance of MIQP solvers for quadratic programs with cardinality and minimum threshold constraints. This SDP approach is based on computing the diagonal decomposition that generates the tightest continuous relaxation in the perspective reformulation of the problem. The algorithmic implication of this SDP approach is that continuous-relaxation based branch-and-cut methods could be more efficient when applied to the perspective reformulations using the parameter vector $d$ found by the new SDP formulation. Although the size of the new SDP problem is larger than that of the “small” SDP problem proposed in Frangioni and Gentile (2007), it can be efficiently computed via SDP solvers based on interior-point methods due to the simple structure of the problem. Stimulated by the new SDP problem, we have also proposed an alternative way of constructing the perspective reformulation, which can be viewed as a "dual" method to the method of convex envelope using the perspective function on semi-continuous variables. Our preliminary comparison results indicate that the proposed SDP formulation can help improve the performance of the MIQP solvers for the perspective cut reformulation of the problem.

## Acknowledgments

We are grateful to the two anonymous reviewers for their constructive suggestions and insightful comments which have helped us to improve the paper substantially. We are particularly indebted to one of the anonymous reviewers who provided us detailed instructions on the implementation of the perspective cut reformulation. This research was supported by the NSFC grants 11101092, 11371103 and the Joint NSFC/RGC grant 71061160506, and by the RGC grants of CUHK414808 and CUHK414610.

## References

Aktürk, M., A. Atamtürk, S. Gürel, et al. 2009. A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Oper. Res. Lett.* **37** 187–191.

Arthanari, T.S., Y. Dodge. 1993. *Mathematical Programming in Statistics*. John Wiley & Sons, New York.

Ben-Tal, A., A.S. Nemirovski. 2001. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. MPS-SIAM Series on Optimization, Society for Industrial Mathematics.

Bertsimas, D., R. Shioda. 2009. Algorithm for cardinality-constrained quadratic optimization. *Comput. Optimi. Appl.* **43** 1–22.

Bienstock, D. 1996. Computational study of a family of mixed-integer quadratic programming problems. *Math. Program.* **74** 121–140.

Blog, B., G. Van der Hoek, A.H.G. Rinnooy Kan, G.T. Timmer. 1983. The optimal selection of small portfolios. *Manage. Sci.* **29** 792–798.

Bonami, P., M.A. Lejeune. 2009. An exact solution approach for portfolio optimization problems under stochastic and integer constraints. *Oper. Res.* **57** 650–670.

Breiman, L. 1995. Better subset regression using the nonnegative garrote. *Technometrics* 373–384.

Cesarone, F., A. Scozzari, F. Tardella. 2009. Efficient algorithms for mean-variance portfolio optimization with hard real-world constraints. *Giornale dell'Istituto Italiano degli Attuari* **72** 37–56.

Chang, T.J., N. Meade, J.E. Beasley, Y.M. Sharaiha. 2000. Heuristics for cardinality constrained portfolio optimisation. *Comput. Oper. Res.* **27** 1271–1302.

Cui, X. T., X. J. Zheng, S. S. Zhu, X. L. Sun. 2013. Convex relaxations and MIQCQP reformulations for a class of cardinality-constrained portfolio selection problems. *J. Global Optim.* **56** 1409–1423.

Frangioni, A., C. Gentile. 2006. Perspective cuts for a class of convex 0–1 mixed integer programs. *Math. Program.* **106** 225–236.

Frangioni, A., C. Gentile. 2007. SDP diagonalizations and perspective cuts for a class of nonseparable MIQP. *Oper. Res. Lett.* **35** 181–185.

Frangioni, A., C. Gentile. 2009. A computational comparison of reformulations of the perspective relaxation: SOCP vs. cutting planes. *Oper. Res. Lett.* **37** 206–210.

Gao, J.J., D. Li. 2013. Optimal cardinality constrained portfolio selection. *Oper. Res.* **61** 745–761.

Grant, M., S. Boyd. 2009. CVX: Matlab software for disciplined convex programming (web page and software).

Guignard, M., S. Kim. 1987. Lagrangean decomposition: A model yielding stronger Lagrangean bounds. *Math. Program.* **39** 215–228.

Günlük, O., J. Linderoth. 2010. Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Math. Program.* **124** 183–205.

Hiriart-Urruty, J.B., C. Lemaréchal. 1993. *Convex Analysis and Minimization Algorithms: Fundamentals*. Springer Verlag, New York.

Jacob, N.L. 1974. A limited-diversification portfolio selection model for the small investor. *J. Finance* **29** 847–856.

Jobst, N.J., M.D. Horniman, C.A. Lucas, G. Mitra. 2001. Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints. *Quant. Finance* **1** 489–501.

Li, D., X.L. Sun, J. Wang. 2006. Optimal lot solution to cardinality constrained mean-variance formulation for portfolio selection. *Math. Finance* **16** 83–101.

Maringer, D., H. Kellerer. 2003. Optimization of cardinality constrained portfolios with a hybrid local search algorithm. *OR Spectrum* **25** 481–495.

Michelon, P., N. Maculan. 1991. Lagrangian decomposition for integer nonlinear programming with linear constraints. *Math. Program.* **52** 303–313.

Miller, A.J. 2002. *Subset Selection in Regression*. Chapman and Hall. Second edition.

Mitra, G., F. Ellison, A. Scowcroft. 2007. Quadratic programming for portfolio planning: Insights into algorithmic and computational issues. part ii: Processing of portfolio planning models with discrete constraints. *J. Asset Manage.* **8** 249–258.

Pólik, I., T. Terlaky. 2007. A survey of the $\mathcal{S}$-lemma. *SIAM Rev.* **49** 371–418.

Shaw, D.X., S. Liu, L. Kopman. 2008. Lagrangian relaxation procedure for cardinality-constrained portfolio optimization. *Optim. Methods Softw.* **23** 411–420.

Tawarmalani, M., N.V. Sahinidis. 2001. Semidefinite relaxations of fractional programs via novel convexification techniques. *J. Global Optim.* **20** 133–154.

Vandenberghe, L., S. Boyd. 1996. Semidefinite programming. *SIAM Rev.* **38** 49–95.

Xie, J., S. He, S. Zhang. 2008. Randomized portfolio selection, with constraints. *Pacific J. Optim.* **4** 89–112.

Yuan, M., Y. Lin. 2007. On the non-negative garrotte estimator. *J. Roy. Statist. Soc. Ser. B* **69** 143–161.