ACTIVE-SET METHODS FOR CONVEX QUADRATIC PROGRAMMING

Anders Forsgren* Philip E. Gill† Elizabeth Wong†

UCSD Center for Computational Mathematics
Technical Report CCoM-15-02
March 28, 2015

Abstract

Computational methods are proposed for solving a convex quadratic program (QP). Active-set methods are defined for a particular primal and dual formulation of a QP with general equality constraints and simple lower bounds on the variables. In the first part of the paper, two methods are proposed, one primal and one dual. These methods generate a sequence of iterates that are feasible with respect to the equality constraints associated with the optimality conditions of the primal-dual form. The primal method maintains feasibility of the primal inequalities while driving the infeasibilities of the dual inequalities to zero. In contrast, the dual method maintains feasibility of the dual inequalities while moving to satisfy the infeasibilities of the primal inequalities. In each of these methods, the search directions satisfy a KKT system of equations formed from Hessian and constraint components associated with an appropriate column basis. The composition of the basis is specified by an active-set strategy that guarantees the nonsingularity of each set of KKT equations. Each of the proposed methods is a conventional active-set method in the sense that an initial primal- or dual-feasible point is required. In the second part of the paper, it is shown how the quadratic program may be solved as coupled pair of primal and dual quadratic programs created from the original by simultaneously shifting the simple-bound constraints and adding a penalty term to the objective function. Any conventional column basis may be made optimal for such a primal-dual pair of shifted-penalized problems. The shifts are then updated using the solution of either the primal or the dual shifted problem. An obvious application of this approach is to solve a shifted dual QP to define an initial feasible point for the primal (or *vice versa*). The computational performance of each of the proposed methods is evaluated on a set of convex problems from the CUTEst test collection.

*Optimization and Systems Theory, Department of Mathematics, Royal Institute of Technology, SE-100 44 Stockholm, Sweden (andersf@kth.se). Research supported by the Swedish Research Council (VR).

†Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112 (pgill@ucsd.edu, elwong@ucsd.edu). Research supported in part by Northrop Grumman Aerospace Systems, and National Science Foundation grants DMS-1318480 and DMS-1361421.

1. Introduction

We consider the formulation and analysis of active-set methods for a convex quadratic program (QP) of the form

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n,\, y\in\mathbb{R}^m}{\text{minimize}} & c^Tx + \tfrac{1}{2}x^THx + \tfrac{1}{2}y^TMy \\ \text{subject to} & Ax + My = b, \quad x \ge 0, \end{array} \tag{1.1}$$

where A, b, c, H and M are constant, with H and M symmetric positive semidefinite. It is assumed throughout that the matrix $\begin{pmatrix} A & M \end{pmatrix}$ associated with the equality constraints has full row rank. (This assumption can be made without loss of generality, as shown in Proposition A.8.)

In order to simplify the theoretical discussion, the inequalities of (1.1) involve nonnegativity constraints only. However, the methods to be described are easily extended to treat all forms of linear constraints. (Numerical results are given for problems with constraints in the form $x_L \le x \le x_U$ and $b_L \le Ax \le b_U$, for fixed vectors x_L, x_U, b_L and b_U.) If $M = 0$, the QP (1.1) is a conventional convex quadratic program with constraints defined in standard form. A regularized quadratic program may be obtained by defining $M = \mu I$ for some small positive parameter μ.

Active-set methods for quadratic programming solve a sequence of linear equations that involve the y-variables and a subset of the x-variables. Each set of equations constitutes the optimality conditions associated with an equality-constrained quadratic subproblem. The goal is to predict the optimal active set, i.e., the set of constraints that are satisfied with equality, at the solution of the problem. A conventional active-set method has two phases. In the first phase, a feasible point is found while ignoring the objective function; in the second phase, the objective is minimized while feasibility is maintained. A useful feature of active-set methods is that they are well-suited for "warm starts", where a good estimate of the optimal active set is used to start the algorithm. This is particularly useful in applications where a sequence of quadratic programs is solved, e.g., in a sequential quadratic programming method or in an ODE- or PDE-constrained problem with mesh refinement. Other applications of active-set methods for quadratic programming include mixed-integer nonlinear programming, portfolio analysis, structural analysis, and optimal control.

In Section 2, the primal and dual forms of a convex quadratic program with constraints in standard form are generalized to include general lower bounds on both the primal and dual variables. These problems constitute a primal-dual pair that includes problem (1.1) and its associated dual as a special case. In Sections 3 and 4, an active-set method is proposed for each of the primal and dual forms associated with the generalized problem of Section 2. Both of these methods generate a sequence of iterates that are feasible with respect to the equality constraints associated with the optimality conditions of the primal-dual problem pair. The primal method maintains feasibility of the primal inequalities while driving the infeasibilities of the dual inequalities to zero. By contrast, the dual method maintains feasibility of the dual inequalities while moving to satisfy the infeasibilities of the primal inequalities. In each of these methods, the search directions satisfy a KKT system of equations formed from Hessian and constraint components associated with an appropriate column basis. The composition of the basis is specified by an active-set strategy that guarantees the nonsingularity of each set of KKT equations.

The methods formulated in Sections 2–4 define conventional active-set methods in the sense that an initial feasible point is required. In Section 5, a method is proposed that solves a pair of coupled quadratic programs created from the original by simultaneously shifting the simple-bound constraints and adding a penalty term to the objective function. Any conventional column basis can be made optimal for such a primal-dual pair of shifted-penalized problems. The shifts are then updated using the solution of either the primal or the dual shifted problem. An obvious application of this idea is to solve a shifted dual QP to define an initial feasible point for the primal, or *vice-versa*. In addition to the obvious benefit of using the objective function while getting feasible, this approach provides an effective method for finding a dual-feasible point when H is positive semidefinite and $M = 0$. Finding a dual-feasible point is relatively straightforward for the strictly convex case, i.e., when H is positive definite. However, in the general case, the dual constraints for the phase-one linear program involve entries from H as well as A, which complicates the formulation of the phase-one method considerably.

Finally, in Section 7 some numerical experiments are presented for a simple Matlab implementation of the coupled primal-dual method applied to a set of convex problems from the CUTEst test collection [33].

There are a number of alternative active-set methods available for solving a QP with constraints written in the format of problem (1.1). Broadly speaking, these methods fall into three classes defined here in the order of increasing generality: (i) methods for strictly convex quadratic programming (H symmetric positive definite) [1,22,31,41,44]; (ii) methods for convex quadratic programming (H symmetric positive semidefinite) [6,26,38,39,45]; and (iii) methods for general quadratic programming (no assumptions on H other than symmetry) [2,3,9,17,19,23,28–30,32,35–37,45]. Of the methods specifically designed for convex quadratic programming, only the methods of Boland [6] and Wong [45, Chapter 4] are dual active-set methods. Some existing active-set quadratic programming solvers include QPOPT [24], QPSchur [1], SQOPT [26], SQIC [30] and QPA (part of the GALAHAD software library) [34].

The primal active-set method proposed in Section 3 is motivated by the methods of Fletcher [17], Gould [32], and Gill and Wong [30], which may be viewed as methods that extend the properties of the simplex method to general quadratic programming. At each iteration, a direction is computed that satisfies a *nonsingular* system of linear equations based on an estimate of the active set at a solution. The equations may be written in symmetric form and involve both the primal and dual variables. In this context, the purpose of the active-set strategy is not only to obtain a good estimate of the optimal active set, but also to ensure that the systems of linear equations that must be solved at each iteration are nonsingular. This strategy allows the application of any convenient linear solver for the computation of the iterates. In this paper, these ideas are applied to convex quadratic programming. The resulting sequence of iterates is the same as that generated by an algorithm for general QP, but the structure of the iteration is different, as is the structure of the linear equations that must be solved. Similar ideas are used to formulate the new dual active-set method proposed in Sections 4.

The proposed primal, dual, and coupled primal-dual methods use a "conventional" active-set approach in the sense that the constraints remain unchanged during the solution of a given QP. Alternative approaches that use a parametric active-set method have been proposed by Best [4, 5], Ritter [42, 43], Ferreau, Bock and Diehl [16], Potschka et al. [40] and implemented in the `qpOASES` package by Ferreau et al. [15]. The use of shifts for the bounds have been suggested by Cartis and Gould [11] in the context of interior methods for linear programming. Another class of active-set methods that are shown to be convergent for strictly convex quadratic programs have been considered by Curtis, Han, and Robinson [12].

Notation and terminology. Given vectors a and b with the same dimension, $\min(a, b)$ is a vector with components $\min(a_i, b_i)$. The vectors e and e_j denote, respectively, the column vector of ones and the jth column of the identity matrix I. The dimensions of e, e_i and I are defined by the context. Given vectors x and y, the column vector consisting of the components of x augmented by the components of y is denoted by (x, y).

2. Background

Although the purpose of this paper is the solution of quadratic programs of the form (1.1), for reasons that will become evident in Section 5, the analysis will focus on the properties of a pair of problems that may be interpreted as a primal-dual pair of QPs associated with problem (1.1).

2.1. Formulation of the primal and dual problems

For given constant vectors q and r, consider the pair of convex quadratic programs

$$(\text{PQP}_{q,r}) \qquad \begin{array}{ll} \underset{x,y}{\text{minimize}} & \tfrac{1}{2}x^THx + \tfrac{1}{2}y^TMy + c^Tx + r^Tx \\ \text{subject to} & Ax + My = b, \qquad\qquad x \ge -q, \end{array}$$

and

$$(\text{DQP}_{q,r}) \qquad \begin{array}{ll} \underset{x,y,z}{\text{maximize}} & -\tfrac{1}{2}x^THx - \tfrac{1}{2}y^TMy + b^Ty - q^Tz \\ \text{subject to} & -Hx + A^Ty + z = c, \qquad z \ge -r. \end{array}$$

The following result gives joint optimality conditions for the triple (x, y, z) such that $(x,\, y)$ is optimal for $(\text{PQP}_{q,r})$, and (x, y, z) is optimal for $(\text{DQP}_{q,r})$. If q and r are zero, then $(\text{PQP}_{0,0})$ and $(\text{DQP}_{0,0})$ are the primal and dual problems associated with (1.1). For arbitrary q and r, $(\text{PQP}_{q,r})$ and $(\text{DQP}_{q,r})$ are essentially the dual of each other, the difference is only an additive constant in the value of the objective function.

Proposition 2.1. *Let q and r denote constant vectors in $\mathbb{R}^n$. If $(x,\, y,\, z)$ is a given triple in $\mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, then $(x,\, y)$ is optimal for $(PQP_{q,r})$ and $(x,\, y,\, z)$ is optimal* *for* ($DQP_{q,r}$) *if and only if*

$$Ax + My - b = 0, \tag{2.1a}$$

$$Hx + c - A^T y - z = 0, \tag{2.1b}$$

$$x + q \geq 0, \tag{2.1c}$$

$$z + r \geq 0, \tag{2.1d}$$

$$(x + q)^T (z + r) = 0. \tag{2.1e}$$

In addition, it holds that $\operatorname{optval}(PQP_{q,r}) - \operatorname{optval}(DQP_{q,r}) = -q^T r$. *Finally,* (2.1) *has a solution if and only if the sets*

$$\big\{(x, y, z) : -Hx + A^T y + z = c,\ z \geq -r\big\} \quad \textit{and} \quad \big\{x : Ax + My = b,\ x \geq -q\big\}$$

are both nonempty.

Proof. Let the vector of Lagrange multipliers for the constraints $Ax + My - b = 0$ be denoted by $\widetilde{y}$. Without loss of generality, the Lagrange multipliers for the bounds $x + q \geq 0$ of ($\text{PQP}_{q,r}$) may be written in the form $z + r$, where r is the given fixed vector r. With these definitions, a Lagrangian function $L(x, y, \widetilde{y}, z)$ associated with ($\text{PQP}_{q,r}$) is given by

$$L(x, y, \widetilde{y}, z) = \tfrac{1}{2} x^T H x + (c + r)^T x + \tfrac{1}{2} y^T M y - \widetilde{y}^T (Ax + My - b) - (z + r)^T (x + q),$$

with $z + r \geq 0$. Stationarity of the Lagrangian with respect to x and y implies that

$$Hx + c + r - A^T \widetilde{y} - z - r = Hx + c - A^T \widetilde{y} - z = 0, \tag{2.2a}$$

$$My - M\widetilde{y} = 0. \tag{2.2b}$$

The optimality conditions for ($\text{PQP}_{q,r}$) are then given by: (i) the feasibility conditions (2.1a) and (2.1c); (ii) the nonnegativity conditions (2.1d) for the multipliers associated with the bounds $x + q \geq 0$; (iii) the stationarity conditions (2.2); and (iv) the complementarity conditions (2.1e). The vector y appears only in the term My of (2.1a) and (2.2b). In addition, (2.2b) implies that $My = M\widetilde{y}$, in which case we may choose $y = \widetilde{y}$. This common value of y and $\widetilde{y}$ must satisfy (2.2a), which is then equivalent to (2.1b). The optimality conditions (2.1) for ($\text{PQP}_{q,r}$) follow directly.

With the substitution $\widetilde{y} = y$, the expression for the primal Lagrangian may be rearranged so that

$$L(x, y, y, z) = -\tfrac{1}{2} x^T H x - \tfrac{1}{2} y^T M y + b^T y - q^T z + (Hx + c - A^T y - z)^T x - q^T r. \tag{2.3}$$

Taking into account (2.2) for $y = \widetilde{y}$, the dual objective is given by (2.3) as $-\frac{1}{2} x^T H x - \frac{1}{2} y^T M y + b^T y - q^T z - q^T r$, and the dual constraints are $Hx + c - A^T y - z = 0$ and $z + r \geq 0$. It follows that ($\text{DQP}_{q,r}$) is equivalent to the dual of ($\text{PQP}_{q,r}$), the only difference is the constant term $-q^T r$ in the objective, which is a consequence of the shift $z + r$ in the dual variables. Consequently, strong duality for convex quadratic programming implies $\operatorname{optval}(\text{PQP}_{q,r}) - \operatorname{optval}(\text{DQP}_{q,r}) = -q^T r$. In addition, the variables x, y and z satisfying (2.1) are feasible for ($\mathrm{PQP}_{q,r}$) and ($\mathrm{DQP}_{q,r}$) with the difference in the objective function value being $-q^Tr$. It follows that (x, y, z) is optimal for ($\mathrm{DQP}_{q,r}$) as well as ($\mathrm{PQP}_{q,r}$). Finally, feasibility of both ($\mathrm{PQP}_{q,r}$) and ($\mathrm{DQP}_{q,r}$) is both necessary and sufficient for the existence of optimal solutions. ∎

2.2. Optimality conditions and the KKT equations

The joint optimality conditions (2.1) may be written in terms of the index sets of fixed and free primal variables. At an arbitrary point x, consider the index sets

$$\mathcal{A}(x) = \{\, i : x_i = -q_i \,\} \quad \text{and} \quad \mathcal{F}(x) = \{1, 2, \dots, n\} \setminus \mathcal{A}(x).$$

The set $\mathcal{A}(x)$ is the *active set* of bound constraints at the point x. The variables with indices in $\mathcal{F}(x)$ are referred to as the *free variables*. If z is any n-vector, the n_F-vector z_F and n_A-vector z_A denote the vectors of components of z associated with $\mathcal{F}(x)$ and $\mathcal{A}(x)$. For the symmetric Hessian H, the matrices H_{FF} and H_{AA} denote the subset of rows and columns of H associated with the sets $\mathcal{F}(x)$ and $\mathcal{A}(x)$, respectively. The unsymmetric matrix of components h_{ij} with $i \in \mathcal{F}(x)$ and $j \in \mathcal{A}(x)$ will be denoted by H_{FA}. Similarly, A_F and A_A denote the matrices of free and active columns of A. With these definitions, the joint optimality conditions (2.1) may be written in the form

$$A_F x_F + A_A x_A + My - b = 0, \tag{2.4a}$$
$$H_{FF} x_F + H_{FA} x_A + c_F - A_F^T y - z_F = 0, \quad x_F + q_F \ge 0, \quad z_F + r_F = 0, \tag{2.4b}$$
$$H_{FA}^T x_F + H_{AA} x_A + c_A - A_A^T y - z_A = 0, \quad x_A + q_A = 0, \quad z_A + r_A \ge 0. \tag{2.4c}$$

Eliminating x_A and z_F using the equalities $z_F + r_F = 0$ and $x_A + q_A = 0$ yields the equations

$$\begin{aligned} A_F x_F + My &= b + A_A q_A, \\ H_{FF} x_F - A_F^T y &= H_{FA} q_A - c_F - r_F, \\ H_{FA}^T x_F - A_A^T y - z_A &= H_{AA} q_A - c_A, \end{aligned}$$

which may be expressed in matrix form as

$$\begin{pmatrix} H_{FF} & A_F^T \\ A_F & -M \end{pmatrix} \begin{pmatrix} x_F \\ -y \end{pmatrix} = \begin{pmatrix} H_{FA} q_A - c_F - r_F \\ A_A q_A + b \end{pmatrix}, \tag{2.5}$$

with $z_A = H_{FA}^T x_F + c_A - A_A^T y - H_{AA} q_A$. If a solution of ($\mathrm{PQP}_{q,r}$) or ($\mathrm{DQP}_{q,r}$) exists, then the equations (2.5) are compatible, but not necessarily nonsingular.

The proposed methods are based on maintaining index sets $\mathcal{B}$ and $\mathcal{N}$ that approximate the free and active sets $\mathcal{F}$ and $\mathcal{A}$ at a solution. The sets $\mathcal{B}$ and $\mathcal{N}$ define a partition of the index set $\mathcal{I} = \{1,\ 2,\ \dots,\ n\}$, i.e., $\mathcal{I} = \mathcal{B} \cup \mathcal{N}$ with $\mathcal{B} \cap \mathcal{N} = \emptyset$. Following standard terminology, we refer to the subvectors x_B and x_N associated with an arbitrary x as the basic and nonbasic variables, respectively. Let H_{BB}, H_{NN},

H_{BN}, A_B, and A_N denote submatrices of H and A analogous to H_{FF}, H_{AA}, H_{FA}, A_F, and A_A, respectively. The crucial distinction between $\mathcal{B}$ and $\mathcal{F}$ is that the basic set $\mathcal{B}$ is defined in such a way that the KKT matrix

$$K_B = \begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix}$$

is nonsingular (cf. equation (2.5)). As in Gill and Wong [30], any set $\mathcal{B}$ such that K_B is nonsingular is referred to as a *second-order consistent basis*. Methods that impose restrictions on the eigenvalues of K_B are known as inertia-controlling methods. (For a description of inertia-controlling methods for general quadratic programming, see, e.g., Gill et al. [29], and Gill and Wong [30].) Given a point (x, y, z) satisfying the optimality conditions (2.4), it is always possible to define a second-order consistent basis $\mathcal{B}$ such that (x, y, z) satisfies the conditions

$$A_B x_B + A_N x_N + My = b, \tag{2.6a}$$

$$H_{BB} x_B + H_{BN} x_N + c_B - A_B^T y - z_B = 0, \quad z_B + r_B = 0, \quad x_B + q_B \geq 0, \tag{2.6b}$$

$$H_{BN}^T x_B + H_{NN} x_N + c_N - A_N^T y - z_N = 0, \quad z_N + r_N \geq 0, \quad x_N + q_N = 0. \tag{2.6c}$$

(For simplicity, it is assumed that $\mathcal{B}$ and $\mathcal{N}$ can be defined so that $\mathcal{N} \subseteq \mathcal{A}(x)$. In practice it may be necessary to include indices in $\mathcal{N}$ that correspond to variables that are temporarily fixed at their current values, see, e.g., Gill and Wong [30, Section 6].) Eliminating x_N and z_B from the equality conditions of (2.6a) and (2.6b) gives (x_B, y) as the unique solution of the equations

$$\begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \begin{pmatrix} x_B \\ -y \end{pmatrix} = \begin{pmatrix} H_{BN} q_N - c_B - r_B \\ A_N q_N + b \end{pmatrix}. \tag{2.7}$$

Once x_B and y have been defined, the z_N-variables may be computed as

$$z_N = H_{BN}^T x_B - H_{NN} q_N + c_N - A_N^T y. \tag{2.8}$$

The two methods proposed in this paper generate a sequence of iterates that satisfy the equality conditions of (2.6) for some partition $\mathcal{B}$ and $\mathcal{N}$. The primal method of Section 3 imposes the restriction that $x_B + q_B \geq 0$, which implies that the sequence of iterates is primal feasible. The dual method of Section 4 imposes dual feasibility via the bounds $z_N + r_N \geq 0$.

The primal and dual methods are derived in terms of a common framework that serves to emphasize the similarities in the methods. In particular, the methods require the solution of a common set of equations defined in terms of a partition of the index set $\mathcal{I} = \{1, 2, \ldots, n\}$. At the start of each iteration, $\mathcal{I}$ is partitioned into disjoint sets $\mathcal{B}$ and $\mathcal{N}$ such that $\mathcal{B} \cup \mathcal{N} = \mathcal{I}$. This initial partition has the property that (x, y, z) is uniquely defined via the equations

$$\left.\begin{aligned} Hx + c - A^T y - z &= 0, \\ Ax + My - b &= 0, \end{aligned}\right\} \text{ with } x_N + q_N \text{ and } z_B + r_B \text{ fixed.} \tag{2.9}$$

During an iteration, $x_B + q_B$ and $z_N + r_N$ are free to move. In a primal iteration, x is primal feasible, with $x_N + q_N = 0$ and $x_B + q_B \geq 0$. In a dual iteration, z is dual feasible, with $z_N + r_N \geq 0$ and $z_B + r_B = 0$.

An iteration of the primal method begins with the identification of a nonbasic primal variable x_l with an infeasible dual value, i.e., $z_l + r_l < 0$. The index l is removed from the nonbasic set, i.e., $\mathcal{N} \leftarrow \mathcal{N} \setminus \{l\}$, and there follows a finite number of *subiterations* during which $z_l + r_l$ is driven to zero from below. During these subiterations, the index l remains distinct from $\mathcal{B}$ and $\mathcal{N}$ while indices are moved from $\mathcal{B}$ to $\mathcal{N}$ in order to maintain primal feasibility. On completion of the final subiteration, $z_l + r_l = 0$ and the index l is added to the basic set, i.e., $\mathcal{B} \leftarrow \mathcal{B} \cup \{l\}$, which defines a conventional partition $\mathcal{B} \cup \mathcal{N} = \{1, 2, \ldots, n\}$ for the next iteration. An iteration of the dual method is defined in an analogous way. The iteration begins with the identification of a basic dual variable z_l associated with an infeasible primal variable, i.e., $x_l + q_l < 0$. The index l is removed from the basic set, and the dual subiterations drive $x_l + q_l$ to zero while maintaining dual feasibility. On completion of the final dual subiteration, $x_l + q_l = 0$ and the index l is added to the nonbasic set, giving $\mathcal{B} \cup \mathcal{N} = \{1, 2, \ldots, n\}$ as in the primal case.

At each subiteration of both the primal and dual methods, an index l and index sets $\mathcal{B}$ and $\mathcal{N}$ are known such that $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1, 2, \ldots, n\}$. A search direction $(\Delta x, \Delta y, \Delta z)$ is defined that satisfies the identities

$$H\Delta x - A^T\Delta y - \Delta z = 0, \tag{2.10a}$$

$$A\Delta x + M\Delta y = 0, \tag{2.10b}$$

$$\Delta x_N = 0, \tag{2.10c}$$

$$\Delta z_B = 0. \tag{2.10d}$$

As $l \notin \mathcal{B}$ and $l \notin \mathcal{N}$, these identities allow both Δx_l and Δz_l to be nonzero. The identities (2.10) may be expressed in the partitioned matrix form

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & H_{Nl}^T & a_l^T & 1 & & \\ h_{Bl} & H_{BB} & H_{BN} & A_B^T & & I & \\ h_{Nl} & H_{BN}^T & H_{NN} & A_N^T & & & I \\ a_l & A_B & A_N & -M & & & \\ & & I & & & & \\ & & & & & I & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ \Delta x_N \\ -\Delta y \\ -\Delta z_l \\ -\Delta z_B \\ -\Delta z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

which implies that the quantities Δx_l, Δx_B, Δy and Δz_l satisfy the homogeneous equations

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T & 1 \\ h_{Bl} & H_{BB} & A_B^T & \\ a_l & A_B & -M & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \\ -\Delta z_l \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \tag{2.11a}$$

with Δz_N given by

$$\Delta z_N = h_{Nl}\Delta x_l + H_{BN}^T\Delta x_B - A_N^T\Delta y. \tag{2.11b}$$

The properties of these equations are established in the next section.

2.3. The linear algebra framework

This section establishes the linear algebra framework that serves to emphasize the underlying symmetry between the primal and dual methods. It is shown that the search direction for the primal and the dual method is a nonzero solution of the homogeneous equations (2.11a), i.e., every direction is a nontrivial null vector of the matrix of (2.11a). In particular, it is shown that the null-space of (2.11a) has dimension one, which implies that every solution of (2.11a) is unique up to a scalar multiple. The length of the direction is then completely determined by fixing either $\Delta x_l = 1$ or $\Delta z_l = 1$. The choice of which component to fix depends on whether or not the corresponding component in a null vector of (2.11a) is nonzero. The conditions are stated precisely in Propositions 2.3 and 2.4 below.

The first result shows that the components Δx_l and Δz_l of any direction (Δx, Δy, Δz) satisfying the identities (2.10) must be such that $\Delta x_l \Delta z_l \geq 0$.

Proposition 2.2. *If the vector* $(\Delta x, \Delta y, \Delta z)$ *satisfies the identities*

$$\begin{aligned} H\Delta x - A^T\Delta y - \Delta z &= 0, \\ A\Delta x + M\Delta y &= 0, \end{aligned}$$

then $\Delta x^T \Delta z \geq 0$. *Moreover, given an index* l *and index sets* $\mathcal{B}$ *and* $\mathcal{N}$ *such that* $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1, 2, \ldots, n\}$ *with* $\Delta x_N = 0$ *and* $\Delta z_B = 0$, *then* $\Delta x_l \Delta z_l \geq 0$.

Proof. Premultiplying the first identity by Δx^T and the second by Δy^T gives

$$\Delta x^T H \Delta x - \Delta x^T A^T \Delta y - \Delta x^T \Delta z = 0, \quad \text{and} \quad \Delta y^T A \Delta x + \Delta y^T M \Delta y = 0.$$

Eliminating the term $\Delta x^T A^T \Delta y$ gives $\Delta x^T H \Delta x + \Delta y^T M \Delta y = \Delta x^T \Delta z$. By definition, H and M are symmetric positive semidefinite, which gives $\Delta x^T \Delta z \geq 0$. In particular, if $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1, 2, \ldots, n\}$, with $\Delta x_N = 0$ and $\Delta z_B = 0$, it must hold that $\Delta x^T \Delta z = \Delta x_l \Delta z_l \geq 0$. ∎

The set of vectors $(\Delta x_l, \Delta x_B, -\Delta y, -\Delta z_l)$ satisfying the homogeneous equations (2.11a) is completely characterized by the properties of the matrices K_B and K_l such that

$$K_B = \begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \quad \text{and} \quad K_l = \begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix}. \tag{2.12}$$

The properties are summarized by the results of the following two propositions.

Proposition 2.3. *If* K_B *is nonsingular, and* Δx_l *is a given nonnegative scalar, then the quantities* Δx_B, Δy, Δz_l *and* Δz_N *of* (2.11) *are unique and satisfy the* *equations*

$$\begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_B \\ -\Delta y \end{pmatrix} = - \begin{pmatrix} h_{Bl} \\ a_l \end{pmatrix} \Delta x_l, \tag{2.13a}$$

$$\Delta z_l = h_{ll} \Delta x_l + h_{Bl}^T \Delta x_B - a_l^T \Delta y, \tag{2.13b}$$

$$\Delta z_N = h_{Nl} \Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y. \tag{2.13c}$$

If $\Delta x_l = 0$, then $\Delta x_B = 0$, $\Delta y = 0$, $\Delta z_l = 0$ and $\Delta z_N = 0$. Otherwise, $\Delta x_l > 0$, and either

(i) *K_l is nonsingular and $\Delta z_l > 0$, or*

(ii) *K_l is singular and $\Delta z_l = 0$, in which case it holds that $\Delta y = 0$, $\Delta z_N = 0$, and the multiplicity of the zero eigenvalue of K_l is one, with corresponding eigenvector $(\Delta x_l, \Delta x_B, 0)$.*

Proof. The second and third blocks of the equations (2.11a) imply that

$$\begin{pmatrix} h_{Bl} \\ a_l \end{pmatrix} \Delta x_l + \begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{2.14}$$

As K_B is nonsingular by assumption, the vectors Δx_B and Δy must constitute the unique solution of (2.14) for a given value of Δx_l. Furthermore, given Δx_B and Δy, the quantities Δz_l and Δz_N of (2.13) are also uniquely defined. The specific value $\Delta x_l = 0$, gives $\Delta x_B = 0$ and $\Delta y = 0$, so that $\Delta z_l = 0$ and $\Delta z_N = 0$. It follows that Δx_l must be nonzero for at least one of the vectors Δx_B, Δy, Δz_l or Δz_N to be nonzero.

Next it is shown that if $\Delta x_l > 0$, then either (i) or (ii) must hold. For (i), it is necessary to show that if $\Delta x_l > 0$ and K_l is nonsingular, then $\Delta z_l > 0$. If K_l is nonsingular, the homogeneous equations (2.11a) may be written in the form

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \Delta z_l, \tag{2.15}$$

which implies that Δx_l, Δx_B and Δy are unique for a given value of Δz_l. In particular, if $\Delta z_l = 0$ then $\Delta x_l = 0$, which would contradict the assumption that $\Delta x_l > 0$. If follows that Δz_l must be nonzero. Finally, Proposition 2.2 implies that if Δz_l is nonzero and $\Delta x_l > 0$, then $\Delta z_l > 0$ as required.

For the first part of (ii), it must be shown that if K_l is singular, then $\Delta z_l = 0$. If K_l is singular, it must have a nontrivial null vector $(p_l,\, p_B,\, -u)$. Moreover, every null vector must have a nonzero p_l, because otherwise $(p_B,\, -u)$ would be a nontrivial null vector of K_B, which contradicts the assumption that K_B is nonsingular. A fixed value of p_l uniquely defines p_B and u, which indicates that the multiplicity of the zero eigenvalue must be one. A simple substitution shows that $(p_l,\, p_B,\, -u,\, v_l)$ is a nontrivial solution of the homogeneous equation (2.11a) such that $v_l = 0$. As the subspace of vectors satisfying (2.11a) is of dimension one, it follows that every solution is unique up to a scalar multiple. Given the properties of the known solution $(p_l,\, p_B,\, -u,\, 0)$, it follows that every solution $(\Delta x_l,\, \Delta x_B,\, -\Delta y,\, -\Delta z_l)$ of (2.11a) is an eigenvector associated with the zero eigenvalue of K_l, with $\Delta z_l = 0$.

For the second part of (ii), if $\Delta z_l = 0$, the homogeneous equations (2.11a) become

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.16}$$

As K_l is singular in (2.16), Proposition A.1 implies that

$$\begin{pmatrix} h_{ll} & h_{Bl}^T \\ h_{Bl} & H_{BB} \\ a_l & A_B \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad \text{and} \quad \begin{pmatrix} a_l^T \\ A_B^T \\ -M \end{pmatrix} \Delta y = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.17}$$

The nonsingularity of K_B implies that $\begin{pmatrix} A_B & -M \end{pmatrix}$ has full row rank, in which case the second equation of (2.17) gives $\Delta y = 0$. It follows that every eigenvector of K_l associated with the zero eigenvalue has the form $(\Delta x_l,\, \Delta x_B,\, 0)$. It remains to show that $\Delta z_N = 0$. If Proposition A.2 is applied to the first equation of (2.17), then it must hold that

$$\begin{pmatrix} h_{ll} & h_{Bl}^T \\ h_{Bl} & H_{BB} \\ h_{Nl} & H_{BN}^T \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

It follows from (2.13c) that $\Delta z_N = h_{Nl}\Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y = 0$, which completes the proof. ∎

Proposition 2.4. *If K_l is nonsingular, and Δz_l is a given nonnegative scalar, then the quantities Δx_l, Δx_B, Δy and Δz_N of (2.11) are unique and satisfy the equations*

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \Delta z_l, \tag{2.18a}$$

$$\Delta z_N = H_{Nl}\Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y. \tag{2.18b}$$

If $\Delta z_l = 0$, then $\Delta x_l = 0$, $\Delta x_B = 0$, $\Delta y = 0$ and $\Delta z_N = 0$. Otherwise, $\Delta z_l > 0$ and either

(i) *K_B is nonsingular and $\Delta x_l > 0$, or*

(ii) *K_B is singular and $\Delta x_l = 0$, in which case, it holds that $\Delta x_B = 0$ and the multiplicity of the zero eigenvalue of K_B is one, with corresponding eigenvector $(0, \Delta y)$.*

Proof. In Proposition 2.2 it is established that $\Delta x_l \geq 0$ if $\Delta z_l > 0$, which implies that the statement of the proposition includes all possible values of Δx_l.

It follows from (2.11a) that Δx_l, Δx_B, and Δy must satisfy the equations

$$\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} \Delta z_l \\ 0 \\ 0 \end{pmatrix}. \tag{2.19}$$

Under the given assumption that K_l is nonsingular, the vectors Δx_l, Δx_B and Δy are uniquely determined by (2.19) for a fixed value of Δz_l. In addition, once Δx_l, Δx_B and Δy are defined, Δz_N is uniquely determined by (2.18b). It follows that if $\Delta z_l = 0$, then $\Delta x_l = 0$, $\Delta x_B = 0$, $\Delta y = 0$ and $\Delta z_N = 0$.

It remains to show that if $\Delta z_l > 0$, then either (i) or (ii) must hold. If K_B is singular, then Proposition A.1 implies that there must exist u and v such that

$$\begin{pmatrix} H_{BB} \\ A_B \end{pmatrix} u = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} A_B^T \\ -M \end{pmatrix} v = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Proposition A.2 implies that the vector u must also satisfy

$$\begin{pmatrix} h_{Bl}^T \\ H_{BB} \\ A_B \end{pmatrix} u = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

If u is nonzero, then $(0, u, 0)$ is a nontrivial null vector for K_l, which contradicts the assumption that K_l is nonsingular. It follows that $\begin{pmatrix} H_{BB} & A_B^T \end{pmatrix}$ has full row rank and the singularity of K_B must be caused by dependent rows in $\begin{pmatrix} A_B & -M \end{pmatrix}$. The nonsingularity of K_l implies that $\begin{pmatrix} a_l & A_B & -M \end{pmatrix}$ has full row rank and there must exist a vector v such that $v^T a_l \neq 0$, $v^T A_B = 0$ and $v^T M = 0$. If v is scaled so that $v^T a_l = -\Delta z_l$, then $(0, 0, -v)$ must be a solution of (2.19). It follows that $\Delta x_l = 0$, $v = \Delta y$, and $(0, \Delta y)$ is an eigenvector of K_B associated with a zero eigenvalue. The nonsingularity of K_l implies that v is unique given the value of the scalar Δz_l, and hence the zero eigenvalue has multiplicity one.

Conversely, $\Delta x_l = 0$ implies that $(\Delta x_B, \Delta y)$ is a null vector K_B. However, if K_B is nonsingular, then the vector is zero, contradicting (2.18a). It follows that K_B must be singular. ∎

3. A Primal Active-Set Method for Convex QP

In this section a primal-feasible method for convex QP is formulated. Each iteration begins and ends with a point (x, y, z) that satisfies the conditions

$$\begin{aligned} A_B x_B + A_N x_N + My &= b, & x_N + q_N &= 0, & \text{(3.1a)} \\ H_{BB} x_B + H_{BN} x_N + c_B - A_B^T y - z_B &= 0, & z_B + r_B &= 0, & \text{(3.1b)} \\ H_{BN}^T x_B + H_{NN} x_N + c_N - A_N^T y - z_N &= 0, & x_B + q_B &\geq 0, & \text{(3.1c)} \end{aligned}$$

for appropriate second-order consistent bases. The purpose of the iterations is to drive (x, y, z) to optimality by driving the dual variables to feasibility (i.e., by driving the negative components of $z_N + r_N$ to zero). Methods for finding $\mathcal{B}$ and $\mathcal{N}$ at the initial point are discussed in Section 5.

An iteration consists of a group of one or more consecutive *subiterations* during which a specific dual variable is made feasible. The first subiteration is called the *base* subiteration. In some cases only the base iteration is performed, but, in general, additional *intermediate* subiterations are required.

At the start of the base subiteration, an index l in the nonbasic set $\mathcal{N}$ is identified such that $z_l + r_l < 0$. The idea is to remove the index l from $\mathcal{N}$ (i.e., $\mathcal{N} \leftarrow \mathcal{N} \setminus \{l\}$) and attempt to increase the value of $z_l + r_l$ by taking a step along a primal-feasible direction (Δx_l, Δx_B, Δy, Δz_l). The removal of l from $\mathcal{N}$ implies that $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1,\, 2,\, \ldots,\, n\}$ with $\mathcal{B}$ second-order consistent. This implies that K_B is nonsingular and the (unique) search direction may be computed as in (2.13) with $\Delta x_l = 1$.

If $\Delta z_l > 0$, the step $\alpha_* = -(z_l + r_l)/\Delta z_l$ gives $z_l + \alpha_* \Delta z_l + r_l = 0$. Otherwise, $\Delta z_l = 0$, and there is no finite value of α that will drive $z_l + \alpha \Delta z_l + r_l$ to its bound, and α_* is defined to be $+\infty$. Proposition A.7 implies that the case $\Delta z_l = 0$ corresponds to the primal objective function being linear and decreasing along the search direction.

Even in the case that Δz_l is positive, it is not always possible to take the step α_* and remain primal feasible. A positive step in the direction (Δx_l, Δx_B, Δy, Δz_l) must increase x_l from its bound, but may decrease some of the basic variables. This makes it necessary to limit the step to ensure that the primal variables remain feasible. The largest step length that maintains primal feasibility is given by

$$\alpha_{\max} = \min_{i:\Delta x_i < 0} \frac{x_i + q_i}{-\Delta x_i}.$$

If $\alpha_{\max}$ is finite, this value gives $x_k + \alpha_{\max}\Delta x_k + q_k = 0$, where the index k is given by $k = \operatorname{argmin}_{i:\Delta x_i < 0}\ (x_i + q_i)/(-\Delta x_i)$. The overall step length is then given by

$$\alpha = \min\big(\alpha_*, \alpha_{\max}\big).$$

An infinite value of α indicates that the primal problem ($\text{PQP}_{q,r}$) is unbounded, or, equivalently, that the dual problem ($\text{DQP}_{q,r}$) is infeasible. In this case, the algorithm is terminated. If the step $\alpha = \alpha_*$ is taken, then $z_l + \alpha \Delta z_l + r_l = 0$, the subiterations are terminated with no intermediate subiterations and $\mathcal{B} \leftarrow \mathcal{B} \cup \{l\}$. Otherwise, $\alpha = \alpha_{\max}$, and the basic and nonbasic sets are updated as $\mathcal{B} \leftarrow \mathcal{B} \setminus \{k\}$ and $\mathcal{N} \leftarrow \mathcal{N} \cup \{k\}$ giving an updated partition $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1,\, 2,\, \ldots,\, n\}$ as before. In order to show that the equations associated with the new partition are well-defined, it is necessary to show that allowing z_k to move does not give a singular K_l. Proposition A.5 shows that the submatrix K_l associated with the updated $\mathcal{B}$ and $\mathcal{N}$ is nonsingular for the cases $\Delta z_l > 0$ and $\Delta z_l = 0$.

Because the removal of k from $\mathcal{B}$ does not alter the nonsingularity of K_l, it is possible to add l to $\mathcal{B}$ and thereby define a unique solution of the system (2.9). However, if $z_l + r_l < 0$, additional intermediate subiterations are required to drive $z_l + r_l$ to zero. In each of these subiterations, the search direction is computed by choosing $\Delta z_l = 1$ in Proposition 2.4. The step length α_* is given by $\alpha_* = -(z_l +$

$r_l)/\Delta z_l$ as in the base iteration above, but now α_* is always finite because $\Delta z_l = 1$. Similar to the base subiteration, if no constraint is added, then $z_l + \alpha_* \Delta z_l + r_l = 0$. Otherwise, the index of another blocking variable k is moved from $\mathcal{B}$ to $\mathcal{N}$. Proposition A.5 implies that the updated matrix K_l is nonsingular at the end of an intermediate subiteration. As a consequence, the intermediate subiterations can be repeated until $z_l + r_l$ is driven to zero.

At the end of the base subiteration or after the intermediate subiterations are completed, it must hold that $z_l + r_l = 0$ and the final K_l is nonsingular. This implies that a new iteration may be initiated with the new basic set $\mathcal{B} \cup \{l\}$ defining a nonsingular K_B.

The primal active-set method is summarized in Algorithm 1 below. Its convergence properties are given in Theorem 3.1.

Theorem 3.1. *Assume that problem* $(\mathrm{PQP}_{q,r})$ *is nondegenerate. Given an initial point* (x, y, z) *satisfying conditions* (3.1) *for a second-order consistent basis* $\mathcal{B}$*, then* Algorithm 1 *finds a solution of* $(\mathrm{PQP}_{q,r})$ *or determines that* $(\mathrm{DQP}_{q,r})$ *is infeasible in a finite number of iterations.*

Proof. Algorithm 1 is a special case of Algorithm 3 of Section 5, which describes a primal QP method defined as part of a primal-dual strategy for choosing appropriate nonzero shifts q and r. The convergence of Algorithm 3 is established in Theorem 5.1. ∎

4. A Dual Active-Set Method for Convex QP

Each iteration of the dual active-set method begins and ends with a point (x, y, z) that satisfies the conditions

$$Hx + c - A^T y - z = 0, \tag{4.1a}$$

$$Ax + My - b = 0, \tag{4.1b}$$

$$z_N + r_N \geq 0, \qquad z_B + r_B = 0, \tag{4.1c}$$

$$x_N + q_N = 0, \tag{4.1d}$$

for appropriate second-order consistent bases. For the dual method, the purpose is to drive the primal variables to feasibility (i.e., by driving the negative components of $x + q$ to zero).

An iteration begins with a base subiteration in which an index l in the basic set $\mathcal{B}$ is identified such that $x_l + q_l < 0$. The corresponding dual variable z_l may be increased from its current value $z_l = -r_l$ by removing the index l from $\mathcal{B}$, and defining $\mathcal{B} \leftarrow \mathcal{B} \setminus \{l\}$. Once l is removed from $\mathcal{B}$, it holds that $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1, 2, \ldots, n\}$. The resulting matrix K_l is nonsingular, and the unique direction $(\Delta x_l, \Delta x_B, \Delta y)$ may be computed as in Proposition 2.4 with the particular value $\Delta z_l = 1$.

If $\Delta x_l > 0$, the step $\alpha_* = -(x_l + q_l)/\Delta x_l$ gives $x_l + \alpha_* \Delta x_l + q_l = 0$. Otherwise, $\Delta x_l = 0$, and there is no finite value of α that will drive $x_l + \alpha \Delta x_l + q_l$ to its bound. In this case, the result of Proposition A.7 implies that the dual objective function

Algorithm 1 A primal active-set method for convex QP.

```
Find $(x, y, z)$ satisfying conditions (3.1) for some second-order consistent basis $\mathcal{B}$;
while $\exists\, l : z_l + r_l < 0$ do
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{l\}$;
    PRIMAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);                 [returns $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$]
    while $z_l + r_l < 0$ do
        PRIMAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);     [returns $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$]
    end while
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{l\}$;
end while
function PRIMAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$)
    $\Delta x_l \leftarrow 1$;   Solve $\begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_B \\ -\Delta y \end{pmatrix} = -\begin{pmatrix} h_{Bl} \\ a_l \end{pmatrix}$;
    $\Delta z_N \leftarrow h_{Nl} \Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y$;
    $\Delta z_l \leftarrow h_{ll} \Delta x_l + h_{Bl}^T \Delta x_B - a_l^T \Delta y$;                 [$\Delta z_l \geq 0$]
    $\alpha_* \leftarrow -(z_l + r_l)/\Delta z_l$;                             [$\alpha_* \leftarrow +\infty$ if $\Delta z_l = 0$]
    $\alpha_{\max} \leftarrow \min_{i:\Delta x_i < 0} (x_i + q_i)/(-\Delta x_i)$;   $k \leftarrow \operatorname{argmin}_{i:\Delta x_i < 0} (x_i + q_i)/(-\Delta x_i)$;
    $\alpha \leftarrow \min(\alpha_*, \alpha_{\max})$;
    if $\alpha = +\infty$ then
        stop;                                                  [(DQP$_{q,r}$) is infeasible]
    end if
    $x_l \leftarrow x_l + \alpha \Delta x_l$;   $x_B \leftarrow x_B + \alpha \Delta x_B$;
    $y \leftarrow y + \alpha \Delta y$;   $z_l \leftarrow z_l + \alpha \Delta z_l$;   $z_N \leftarrow z_N + \alpha \Delta z_N$;
    if $z_l + r_l < 0$ then
        $\mathcal{B} \leftarrow \mathcal{B} \setminus \{k\}$;   $\mathcal{N} \leftarrow \mathcal{N} \cup \{k\}$;
    end if
    return $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$;
end function
function PRIMAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$)
    $\Delta z_l \leftarrow 1$;   Solve $\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$;        [$\Delta x_l \geq 0$]
    $\Delta z_N \leftarrow H_{Nl} \Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y$;
    $\alpha_* \leftarrow -(z_l + r_l)$;
    $\alpha_{\max} \leftarrow \min_{i:\Delta x_i < 0} (x_i + q_i)/(-\Delta x_i)$;   $k \leftarrow \operatorname{argmin}_{i:\Delta x_i < 0} (x_i + q_i)/(-\Delta x_i)$;
    $\alpha \leftarrow \min(\alpha_*, \alpha_{\max})$;
    $x_l \leftarrow x_l + \alpha \Delta x_l$;   $x_B \leftarrow x_B + \alpha \Delta x_B$;
    $y \leftarrow y + \alpha \Delta y$;   $z_l \leftarrow z_l + \alpha \Delta z_l$;   $z_N \leftarrow z_N + \alpha \Delta z_N$;
    if $z_l + r_l < 0$ then
        $\mathcal{B} \leftarrow \mathcal{B} \setminus \{k\}$;   $\mathcal{N} \leftarrow \mathcal{N} \cup \{k\}$;
    end if
    return $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$;
end function
```

is linear and increasing along the search direction and α_* is defined to be $+\infty$. As $x_l + q_l$ is increased towards its bound, the associated dual variable z_l increases from its current value, but other nonbasic dual variables may decrease and violate their bounds. This makes it necessary to limit the step by

$$\alpha_{\max} = \min_{i:\Delta z_i<0} \frac{z_i + r_i}{-\Delta z_i}$$

to maintain dual feasibility. If $\alpha_{\max}$ is finite, then $z_k + \alpha_{\max}\Delta z_k + r_k = 0$, where the index k is given by $k = \operatorname{argmin}_{i:\Delta z_i<0}\ (z_i + r_i)/(-\Delta z_i)$. The overall step length is then $\alpha = \min\big(\alpha_*, \alpha_{\max}\big)$, where an infinite value of α implies that the dual problem is unbounded, or, equivalently, that the primal problem $(\mathrm{PQP}_{q,r})$ is infeasible. If $\alpha = \alpha_*$, then $x_l + \alpha\Delta x_l + q_l = 0$. Otherwise $\alpha = \alpha_{\max}$, and $\mathcal{N}$ and $\mathcal{B}$ are updated as $\mathcal{N} = \mathcal{N} \setminus \{k\}$ and $\mathcal{B} = \mathcal{B} \cup \{k\}$. Regardless of the definition of the step, the partition at the new point satisfies $\mathcal{B} \cup \{l\} \cup \mathcal{N} = \{1, 2, \ldots, n\}$. To ensure nonsingularity, it is necessary to show that allowing the variable x_k to move does not cause singularity. In Proposition A.6 it is established that K_B is nonsingular for the two possible cases $\Delta x_l > 0$ and $\Delta x_l = 0$..

As K_B is nonsingular, moving l into $\mathcal{N}$ would provide second-order consistent sets $\mathcal{B}$ and $\mathcal{N}$ such that $\mathcal{B} \cup \mathcal{N} = \{1, 2, \ldots, n\}$ with the current x a unique solution of (2.9). However, if $x_l + q_l < 0$, additional intermediate subiterations are necessary to drive $x_l + q_l$ to zero. The nonsingularity of K_B implies that the search direction may be computed as in Proposition 2.3, with the definition $\Delta x_l = 1$. The step length is computed as $\alpha_* = -(x_l + q_l)/\Delta x_l$ as above, but in this case α_* is always finite because $\Delta x_l = 1$. As in the case of a base subiteration, if no constraint index is added to $\mathcal{B}$, then $x_l + \alpha\Delta x_l + q_l = 0$. Otherwise, the index k of a blocking variable is moved from $\mathcal{N}$ to $\mathcal{B}$. In Proposition A.6 it is shown that the updated K_B is nonsingular at the end of an intermediate subiteration. Consequently, the intermediate subiterations can be repeated until the final $x_l + q_l$ is zero.

Once a zero value of $x_l + q_l$ is obtained at the end of the base subiteration or after intermediate subiterations, the resulting K_B matrix is nonsingular. At this point, the iteration is complete and the index l is moved to $\mathcal{N}$. The new K_B matrix is nonsingular, and a new iteration may be initiated.

The dual active-set method is summarized in Algorithm 2 below. Its convergence properties are given in Theorem 4.1.

Theorem 4.1. *Assume that problem* $(\mathrm{DQP}_{q,r})$ *is nondegenerate. Then, given an initial point* (x, y, z) *satisfying conditions* (4.1) *for some second-order consistent basis* $\mathcal{B}$*,* Algorithm 2 *either solves* $(\mathrm{DQP}_{q,r})$ *or concludes that* $(\mathrm{PQP}_{q,r})$ *is infeasible in a finite number of iterations.*

Proof. The proof mirrors that of the primal active-set method of Section 5 (see Theorem 5.1). ∎

Algorithm 2 A dual active-set method for convex QP.

```
Find $(x, y, z)$ satisfying conditions (4.1) for some second-order consistent basis $\mathcal{B}$;
while $\exists\, l : x_l + q_l < 0$ do
    $\mathcal{B} \leftarrow \mathcal{B} \setminus \{l\}$;
    DUAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);                                  [Base subiteration]
    while $x_l + q_l < 0$ do
        DUAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);          [Intermediate subiteration]
    end while
    $\mathcal{N} \leftarrow \mathcal{N} \cup \{l\}$;
end while
function DUAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$)
    $\Delta z_l \leftarrow 1$;  Solve $\begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$;      [$\Delta x_l \geq 0$]
    $\Delta z_N \leftarrow h_{Nl} \Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y$;
    $\alpha_* \leftarrow -(x_l + q_l)/\Delta x_l$;                                  [$\alpha_* \leftarrow +\infty$ if $\Delta x_l = 0$]
    $\alpha_{\max} \leftarrow \min_{i:\Delta z_i < 0} \, (z_i + r_i)/(-\Delta z_i)$;   $k \leftarrow \operatorname{argmin}_{i:\Delta z_i < 0} \, (z_i + r_i)/(-\Delta z_i)$;
    $\alpha \leftarrow \min\left(\alpha_*, \alpha_{\max}\right)$;
    if $\alpha = +\infty$ then
        stop;                                                   [(PQP$_{q,r}$) is infeasible]
    end if
    $x_l \leftarrow x_l + \alpha \Delta x_l$;   $x_B \leftarrow x_B + \alpha \Delta x_B$;
    $y \leftarrow y + \alpha \Delta y$;   $z_l \leftarrow z_l + \alpha \Delta z_l$;   $z_N \leftarrow z_N + \alpha \Delta z_N$;
    if $x_l + q_l < 0$ then
        $\mathcal{B} \leftarrow \mathcal{B} \cup \{k\}$;   $\mathcal{N} \leftarrow \mathcal{N} \setminus \{k\}$;
    end if
    return $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$;
end function
function DUAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$)
    $\Delta x_l \leftarrow 1$;  Solve $\begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix} \begin{pmatrix} \Delta x_B \\ -\Delta y \end{pmatrix} = -\begin{pmatrix} h_{Bl} \\ a_l \end{pmatrix}$;
    $\Delta z_l \leftarrow h_{ll} \Delta x_l + h_{Bl}^T \Delta x_B - a_l^T \Delta y$;                                 [$\Delta z_l \geq 0$]
    $\Delta z_N \leftarrow h_{Nl} \Delta x_l + H_{BN}^T \Delta x_B - A_N^T \Delta y$;
    $\alpha_* \leftarrow -(x_l + q_l)$;
    $\alpha_{\max} \leftarrow \min_{i:\Delta z_i < 0} \, (z_i + r_i)/(-\Delta z_i)$;   $k \leftarrow \operatorname{argmin}_{i:\Delta z_i < 0} \, (z_i + r_i)/(-\Delta z_i)$;
    $\alpha \leftarrow \min\left(\alpha_*, \alpha_{\max}\right)$;
    $x_l \leftarrow x_l + \alpha \Delta x_l$;   $x_B \leftarrow x_B + \alpha \Delta x_B$;
    $y \leftarrow y + \alpha \Delta y$;   $z_l \leftarrow z_l + \alpha \Delta z_l$;   $z_N \leftarrow z_N + \alpha \Delta z_N$;
    if $x_l + q_l < 0$ then
        $\mathcal{B} \leftarrow \mathcal{B} \cup \{k\}$;   $\mathcal{N} \leftarrow \mathcal{N} \setminus \{k\}$;
    end if
    return $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$;
end function
```

5. Combining Primal and Dual Active-Set Methods

The primal active-set method proposed in Section 3 may be used to solve (PQP$_{q,r}$) for a given initial second-order consistent basis satisfying the conditions (3.1). An appropriate initial point may be found by solving a conventional phase-1 linear program. Alternatively, the dual active-set method of Section 4 may be used in conjunction with an appropriate phase-1 procedure to solve the quadratic program (PQP$_{q,r}$) for a given initial second-order consistent basis satisfying the conditions (4.1). In this section a method is proposed that provides an alternative to the conventional phase-1/phase-2 approach. It is shown that a pair of coupled quadratic programs may be created from the original by simultaneously shifting the bound constraints. Any second-order consistent basis can be made optimal for such a primal-dual pair of shifted problems. The shifts are then updated using the solution of either the primal or the dual shifted problem. An obvious application of this approach is to solve a shifted dual QP to define an initial feasible point for the primal, or *vice-versa*. This strategy provides an alternative to the conventional phase-1/phase-2 approach that utilizes the QP objective function while finding a feasible point.

5.1. Finding an initial second-order-consistent basis

For the methods described in Section 5.2 below, it is possible to define a simple procedure for finding the initial second-order consistent basis $\mathcal{B}$. The required basis must define a nonsingular KKT matrix K_B such that

$$K_B = \begin{pmatrix} H_{BB} & A_B^T \\ A_B & -M \end{pmatrix}. \tag{5.1}$$

The initial basis may be obtained by a finding a symmetric permutation $\mathit{\Pi}$ of the "full" KKT matrix K such that

$$\mathit{\Pi}^T K \mathit{\Pi} = \mathit{\Pi}^T \begin{pmatrix} H & A \\ A & -M \end{pmatrix} \mathit{\Pi} = \begin{pmatrix} H_{BB} & A_B^T & H_{BN} \\ A_B & -M & A_N \\ H_{BN}^T & A_N^T & H_{NN} \end{pmatrix}, \tag{5.2}$$

where the leading principal block 2×2 submatrix is of the form (5.1). The full row-rank assumption on $\begin{pmatrix} A & -M \end{pmatrix}$ ensures that the partition (5.2) is well defined, see [20, Section 6]. In practice, the permutation may be determined using any method for finding a symmetric indefinite factorization of K, see, e.g., [8, 10, 18]. Such methods use symmetric interchanges that implicitly form the nonsingular matrix K_B by deferring singular pivots. In this case, K_B may be defined as any submatrix of the largest nonsingular principal submatrix obtained by the factorization (There may be further permutations within $\mathit{\Pi}$ that are not relevant to this discussion; for further details, see, e.g., [13, 14, 20, 21].)

The permutation $\mathit{\Pi}$ defines the initial $\mathcal{B}$-$\mathcal{N}$ partition of the columns of A. Once $\mathit{\Pi}$ has been determined, the variables with indices in $\mathcal{N}$ are set on their bounds and the shifts are initialized.

5.2. Initializing the shifts

Given a second-order consistent basis, it is straightforward to create a $(q^{(0)}, r^{(0)})$-pair and corresponding (x, y, z) so that $q^{(0)} \geq 0$, $r^{(0)} \geq 0$ and (x, y, z) are optimal for $(\mathrm{PQP}_{q^{(0)},r^{(0)}})$ and $(\mathrm{DQP}_{q^{(0)},r^{(0)}})$. First, choose nonnegative vectors $q_N^{(0)}$ and $r_B^{(0)}$. (Obvious choices are $q_N^{(0)} = 0$ and $r_B^{(0)} = 0$.) Define $z_B = -r_B^{(0)}$, $x_N = -q_N^{(0)}$, and solve the nonsingular KKT-system (2.7) to obtain x_B and y, and compute z_N from (2.8). Finally, let $q_B^{(0)} \geq \max\{-x_B, 0\}$ and $r_N^{(0)} \geq \max\{-z_N, 0\}$. Then, it follows from Proposition 2.1 that x, y and z are optimal for the problems $(\mathrm{PQP}_{q^{(0)},r^{(0)}})$ and $(\mathrm{DQP}_{q^{(0)},r^{(0)}})$, with $q^{(0)} \geq 0$ and $r^{(0)} \geq 0$. If $q^{(0)}$ and $r^{(0)}$ are zero, then x, y and z are optimal for the original problem.

5.3. Solving the original problem by removing the shifts

The original problem may now be solved as a pair of shifted quadratic programs. Two alternative strategies are proposed. The first is a "primal first" strategy in which a shifted primal quadratic program is solved, followed by a dual. The second is an analogous "dual first" strategy.

The "primal-first" strategy is summarized as follows.

(0) Find $\mathcal{B}$, $\mathcal{N}$, $q^{(0)}$, $r^{(0)}$, x, y, z, as described in Sections 5.1 and 5.2.

(1) Set $q^{(1)} = q^{(0)}$, $r^{(1)} = 0$. Solve $(\mathrm{PQP}_{q,0})$ using the primal active-set method.

(2) Set $q^{(2)} = 0$, $r^{(2)} = 0$. Solve $(\mathrm{DQP}_{0,0})$ using the dual active-set method.

In steps (1) and (2), the initial $\mathcal{B}$–$\mathcal{N}$ partition and initial values of x, y, and z are defined as the final $\mathcal{B}$–$\mathcal{N}$ partition and final values of x, y, and z from the preceding step.

The "dual-first" strategy is defined in an analogous way.

(0) Find $\mathcal{B}$, $\mathcal{N}$, $q^{(0)}$, $r^{(0)}$, x, y, z, as described in Section 5.1 and 5.2.

(1) Set $q^{(1)} = 0$, $r^{(1)} = r^{(0)}$. Solve $(\mathrm{DQP}_{0,r})$ using the dual active-set method.

(2) Set $q^{(2)} = 0$, $r^{(2)} = 0$. Solve $(\mathrm{PQP}_{0,0})$ using the primal active-set method.

As in the "primal-first" strategy, the initial $\mathcal{B}$–$\mathcal{N}$ partition and initial values of x, y, and z for steps (1) and (2), are defined as the final $\mathcal{B}$–$\mathcal{N}$ partition and final values of x, y, and z from the preceding step.

In order for these approaches to be well-defined, a simple generalization of the primal and dual active-set methods is needed.

5.4. Relaxed initial conditions for the primal QP method.

For Algorithm 1, the initial values of $\mathcal{B}$, $\mathcal{N}$, q, r, x, y, and z must satisfy conditions (3.1). However, the choice of $r = r^{(1)} = 0$ in Step (1) of the primal-first strategy may give some negative components in the vector $z_B + r_B$. This possibility may be handled by defining a simple generalization of Algorithm 1 that allows initial points satisfying the conditions

$$\begin{aligned} Hx + c - A^T y - z &= 0, && (5.3a)\\ Ax + My - b &= 0, && (5.3b)\\ x_B + q_B &\geq 0, \qquad x_N + q_N = 0, && (5.3c)\\ z_B + r_B &\leq 0, && (5.3d) \end{aligned}$$

instead of the conditions (3.1). In Algorithm 1, the index l identified at the start of the primal base iteration is selected from the nonbasic indices such that $z_j + r_j < 0$. In the generalized algorithm, the set of eligible indices for l is extended to include indices associated with negative values of $z_B + r_B$. If the index l is deleted from $\mathcal{B}$, the associated matrix K_l is nonsingular, and intermediate subiterations are executed until the updated value satisfies $z_l + r_l = 0$. At this point, the index l is returned $\mathcal{B}$. The method is summarized in Algorithm 3.

Algorithm 3 A primal active-set method for convex QP.

```
Find $(x, y, z)$ satisfying conditions (5.3) for some second-order consistent basis $\mathcal{B}$;
while $\exists\, l : z_l + r_l < 0$ do
    if $l \in \mathcal{N}$ then
        $\mathcal{N} \leftarrow \mathcal{N} \setminus \{l\}$;
        PRIMAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);          [returns $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$]
    else
        $\mathcal{B} \leftarrow \mathcal{B} \setminus \{l\}$;
    end if
    while $z_l + r_l < 0$ do
        PRIMAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);  [returns $\mathcal{B}$, $\mathcal{N}$, $x$, $y$, $z$]
    end while
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{l\}$;
end while
```

Theorem 5.1. *Assume that problem* $(\mathrm{PQP}_{q,r})$ *is nondegenerate. Given an initial point* (x, y, z) *satisfying conditions* (5.3) *for a second-order consistent basis* $\mathcal{B}$, *then* Algorithm 3 *finds a solution of* $(\mathrm{PQP}_{q,r})$ *or determines that* $(\mathrm{DQP}_{q,r})$ *is infeasible in a finite number of iterations.*

Proof. Assume that (x, y, z) satisfies the conditions (5.3) for the second-order consistent basis $\mathcal{B}$. Let $\mathcal{B}^<$ denote the index set $\mathcal{B}^< = \{i \in \mathcal{B} : z_i + r_i < 0\}$, and let $\widetilde{r}$ be the vector $\widetilde{r}_i = r_i$, $i \notin \mathcal{B}^<$, and $\widetilde{r}_i = -z_i$, $i \in \mathcal{B}^<$. These definitions imply that $\widetilde{r}_i = -z_i > -z_i + z_i + r_i = r_i$, for every $i \in \mathcal{B}^<$. It follows that $\widetilde{r} \geq r$, and the feasible region of $(\mathrm{DQP}_{q,r})$ is a subset of the feasible region of $(\mathrm{DQP}_{q,\widetilde{r}})$. In addition, if r is replaced by $\widetilde{r}$ in (3.1), the only difference is that $z_B + \widetilde{r}_B = 0$, i.e., the initial point for (5.3) is a stationary point with respect to $(\mathrm{PQP}_{q,\widetilde{r}})$.

The first step of the proof is to show that after a finite number of iterations of Algorithm 3, one of three possible events must occur: (i) the cardinality of the set $\mathcal{B}^<$ is decreased by at least one; (ii) a solution of problem ($\mathrm{PQP}_{q,r}$) is found; or (iii) ($\mathrm{DQP}_{q,r}$) is declared infeasible. The proof will also establish that if (i) does not occur, then either (ii) or (iii) must hold after a finite number of iterations.

Assume that (i) never occurs. This implies that the index l selected in the base iteration can never be an index in $\mathcal{B}^<$ because at the end of such an iteration, it would belong to $\mathcal{B}$ with $z_l + r_l = 0$, contradicting the assumption that the cardinality of $\mathcal{B}^<$ never decreases. For the same reason, it must hold that $k \notin \mathcal{B}^<$ for every index k selected to be moved from $\mathcal{B}$ to $\mathcal{N}$ in any subiteration, because an index can only be moved from $\mathcal{N}$ to $\mathcal{B}$ by being selected in the base iteration. These arguments imply that $z_i = -\widetilde{r}_i$, with $i \in \mathcal{B}^<$, throughout the iterations. It follows that the iterates may be interpreted as being members of a sequence constructed for solving ($\mathrm{PQP}_{q,\widetilde{r}}$) with a fixed $\widetilde{r}$, where the initial stationary point is given, and each iteration gives a new stationary point. The nondegeneracy assumption implies that the objective value of ($\mathrm{PQP}_{q,\widetilde{r}}$) is strictly decreasing at each base subiteration, and nonincreasing at each intermediate subiteration. The number of intermediate subiterations is finite, which implies that a strict improvement of the objective value of ($\mathrm{PQP}_{q,\widetilde{r}}$) is obtained at each iteration. As there are only a finite number of stationary points, Algorithm 3 either solves ($\mathrm{PQP}_{q,\widetilde{r}}$) or concludes that ($\mathrm{DQP}_{q,\widetilde{r}}$) is infeasible after a finite number of iterations. If ($\mathrm{PQP}_{q,\widetilde{r}}$) is solved, then $z_N + r_N \geq 0$, because $\widetilde{r}_j = r_j$ for $j \in \mathcal{N}$. Hence, Algorithm 3 can not proceed further by selecting an $l \in \mathcal{N}$, and the only way to reduce the objective is to select an l in $\mathcal{B}$ such that $z_j + r_j < 0$. Under the assumption that (i) does not occur, it must hold that no eligible indices exist and $\mathcal{B}^< = \emptyset$. However, in this case ($\mathrm{PQP}_{q,r}$) has been solved with $\widetilde{r} = r$, and (ii) must hold. If Algorithm 3 declares ($\mathrm{DQP}_{q,\widetilde{r}}$) to be infeasible, then ($\mathrm{DQP}_{q,r}$) must also be infeasible because the feasible region of ($\mathrm{DQP}_{q,r}$) is contained in the feasible region of ($\mathrm{DQP}_{q,\widetilde{r}}$). In this case ($\mathrm{DQP}_{q,r}$) is infeasible and (iii) occurs.

Finally, if (i) occurs, there is an iteration at which the cardinality of $\mathcal{B}^<$ decreases and an index is removed from $\mathcal{B}^<$. There may be more than one such index, but there is at least one l moved from $\mathcal{B}^<$ to $\mathcal{B}\backslash\mathcal{B}^<$, or one k moved from $\mathcal{B}^<$ to $\mathcal{N}$. In either case, the cardinality of $\mathcal{B}^<$ is decreased by at least one. After such an iteration, the argument given above may be repeated for the new set $\mathcal{B}^<$ and new shift $\widetilde{r}$. Applying this argument repeatedly gives the result the situation (i) can occur only a finite number of times.

It follows that, (ii) or (iii) must occur after a finite number of iterations, which is the required result. ∎

5.5. Relaxed initial conditions for the dual QP method.

Analogous to the primal case, the choice of $q = q^{(1)} = 0$ in Step (1) of the dual-first strategy may give some negative components in the vector $x_N + q_N$. In the case, the conditions on the initial values of $\mathcal{B}$, $\mathcal{N}$, q, r, x, y, and z are generalized so that

$$Hx + c - A^T y - z = 0, \tag{5.4a}$$
$$Ax + My - b = 0, \tag{5.4b}$$
$$z_N + r_N \geq 0, \qquad z_B + r_B = 0, \tag{5.4c}$$
$$x_N + q_N \leq 0. \tag{5.4d}$$

Similarly, the set of eligible indices may be extended to include indices associated with negative values of $x_N + q_N$. If the index l is from $\mathcal{N}$, the associated matrix K_B is nonsingular, and intermediate subiterations are executed until the updated value satisfies $x_l + q_l = 0$. At this point, the index l is returned $\mathcal{N}$. The method is summarized in Algorithm 4.

The strategies of solving two consecutive quadratic programs may be generalized to a sequence of more than two quadratic programs, where we alternate between primal and dual active-set methods, and eliminate the shifts in more than two steps.

Algorithm 4 A dual active-set method for convex QP.

```
Find $(x, y, z)$ satisfying conditions (5.4) for some second-order consistent $\mathcal{B}$;
while $\exists\ l : x_l + q_l < 0$ do
    if $l \in \mathcal{B}$ then
        $\mathcal{B} \leftarrow \mathcal{B} \setminus \{l\}$;
        DUAL_BASE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);                [Base subiteration]
    else
        $\mathcal{N} \leftarrow \mathcal{N} \setminus \{l\}$;
    end if
    while $x_l + q_l < 0$ do
        DUAL_INTERMEDIATE($\mathcal{B}$, $\mathcal{N}$, $l$, $x$, $y$, $z$);     [Intermediate subiteration]
    end while
    $\mathcal{N} \leftarrow \mathcal{N} \cup \{l\}$;
end while
```

6. Practical Issues

6.1. Quadratic programs with upper and lower bounds

As stated, the primal quadratic program has lower bound zero on the x-variables. This is for notational convenience. This form may be generalized in a straightforward manner to a form where the x-variables has both lower and upper bounds on the primal variables, i.e., $b_L \leq x \leq b_U$, where components of b_L can be $-\infty$ and components of b_U can be $+\infty$. Given primal shifts q_L and q_U, and dual shifts r_L and r_U, we have the primal-dual pair

$$(\text{PQP}_{q,r}) \qquad \begin{array}{ll} \underset{x,y}{\text{minimize}} & \tfrac{1}{2}x^T Hx + \tfrac{1}{2}y^T My + c^T x + (r_L - r_U)^T x \\ \text{subject to} & Ax + My = b, \qquad\qquad b_L - q_L \leq x \leq b_U + q_U, \end{array}$$

and

$$(\mathrm{DQP}_{q,r})\qquad \begin{array}{ll} \underset{x,y,z_L,z_U}{\text{maximize}} & -\frac{1}{2}x^THx - \frac{1}{2}y^TMy + b^Ty + (b_L - q_L)^Tz_L - (b_U + q_U)^Tz_U \\ \text{subject to} & -Hx + A^Ty + z_L - z_U = c, \quad z_L \geq -r_L, \quad z_U \geq -r_U. \end{array}$$

An infinite bound has neither a shift nor a corresponding dual variable. For example, if x_j is a free variable, then the corresponding components of b_L and b_U are infinite. In the procedure given in Section 5.1 for finding the first second-order consistent basis $\mathcal{B}$, it is assumed that variables with indices not selected for $\mathcal{B}$ are initialized at one of their bounds. As a free variable has no finite bounds, any index j associated with a free variable should be selected for $\mathcal{B}$. However, this cannot be guaranteed in practice, and in the next section it is shown that the primal and dual QP methods may be extended to allow a free variable to be fixed temporarily at some value.

6.2. Temporary bounds

If the QP is defined in the general problem format of Section 6.1, then any free variable not selected for $\mathcal{B}$ has no upper or lower bound and must be temporarily fixed at some value $x_j = \bar{x}_j$ (say). The treatment of such "temporary bounds" involves some additional modifications to the primal and dual methods of Sections 5.4 and 5.5.

Each temporary bound $x_j = \bar{x}_j$ defines an associated dual variable z_j with initial value $\bar{z}_j$. Since the bound is temporary, it is treated as an equality constraint, and the desired value of z_j is zero. Initially, an index j corresponding to a temporary bound is assigned a primal shift $q_j = 0$ and a dual shift $r_j = -\bar{z}_j$, making $\bar{x}_j$ and $\bar{z}_j$ feasible for the shifted problem. In both the primal-first and dual-first approaches, the idea is to drive the z_j-variables associated with temporary bounds to zero in the primal and leave them unchanged in the dual.

In a primal problem, regardless of whether it is solved before or after the dual problem, an index j corresponding to a temporary bound for which $z_j \neq 0$ is considered eligible for selection as l in the base subiteration, i.e., the index can be selected regardless of the sign of z_j. Once selected, z_j is driven to zero and j belongs to $\mathcal{B}$ after such an iteration. In addition, since x_j is unbounded, j will remain in $\mathcal{B}$ throughout the iterations. Hence, at termination of a primal problem, any index j corresponding to a temporarily bounded variable must have $z_j = 0$. If the maximum step length at a base subiteration is infinite, the dual problem is infeasible, as in the case of a regular bound.

In a dual problem, the dual method is modified so that the dual variables associated with temporary bounds remain fixed throughout the iterations. At any subiteration, if it holds that $\Delta z_j \neq 0$ for some temporary bound, then no step is taken and one such index j is moved from $\mathcal{N}$ to $\mathcal{B}$. Consequently, a move is made only if $\Delta z_j = 0$ for every temporary bound j. It follows that the dual variables for the temporary bounds will remain unaltered throughout the dual iterations. Note that an index j corresponding to a temporary bound is moved at most once from $\mathcal{N}$ to $\mathcal{B}$, and never moved back since the corresponding x_j-variable is unbounded. If the maximum step length at a base subiteration is infinite, it must hold that $\Delta z_j = 0$ for all temporary bounds j, and the primal problem is infeasible.

The discussion above implies that a pair of primal and dual problems solved consecutively will terminate with $z_j = 0$ for all indices j associated with temporary bounds. This is because z_j is unchanged in the dual problem and driven to zero in the primal problem.

7. Numerical Examples

In this section we describe a particular formulation of the primal-dual shifted method of Section 5. In addition, some numerical experiments are presented for a simple Matlab implementation applied to a set of convex problems from the CUTEst test collection (see Bongartz et al. [7], and Gould, Orban and Toint [33]).

7.1. The coupled primal-dual algorithm PDQP

For illustrative purposes, a primal-dual shifted method is used in which either a "primal-first" or "dual-first" strategy is selected based on the initial point. In particular, if the point is dual feasible, then the "dual-first" strategy. is used. Otherwise, the "primal-first" strategy is selected.

7.2. The implementation

Each CUTEst QP problem may be written in the form

$$\underset{x}{\text{minimize}}\;\; c^Tx + \tfrac{1}{2}x^THx \quad \text{subject to}\;\; \ell \le \begin{pmatrix} x \\ \widetilde{A}x \end{pmatrix} \le u, \tag{7.1}$$

where ℓ and u are constant vectors of lower and upper bounds and $\widetilde{A}$ has dimension $m \times n$. In this format, a fixed variable or equality constraint has the same value for its upper and lower bound. Each problem was converted to the equivalent form

$$\underset{x,s}{\text{minimize}}\;\; c^Tx + \tfrac{1}{2}x^THx \quad \text{subject to}\;\; \widetilde{A}x - s = 0, \quad \ell \le \begin{pmatrix} x \\ s \end{pmatrix} \le u, \tag{7.2}$$

where s is a vector of slack variables. With this formulation, the QP problem involves simple upper and lower bounds instead of nonnegativity constraints. It follows that the matrix M is zero, but the full row-rank assumption on the constraint matrix is satisfied because the constraint matrix A takes the form $\begin{pmatrix} \widetilde{A} & -I \end{pmatrix}$ and has rank m. In this situation, a nonsingular $m \times m$ submatrix A_B of A may be identified using a so-called "crash" procedure. One algorithm for doing this has been proposed by Gill, Murray and Saunders [25], who use a sparse LU factorization of A^T to identify a square nonsingular subset of the columns of A_B. These factors give a matrix Z whose columns form a basis for the null space of A as

$$Z = \begin{pmatrix} -A_B^{-1}A_N \\ I \end{pmatrix}.$$

This Z may be used to form $Z^T HZ$, and a partial Cholesky factorization with interchanges may be used to find an upper-triangular matrix R that is the factor of the largest nonsingular leading submatrix of $Z^T HZ$. Let Z_R denote the columns of Z corresponding to R, and let Z be partitioned as $Z = \begin{pmatrix} Z_R & Z_A \end{pmatrix}$. Then, the set $\mathcal{B}$ given by the nonsingular A_B may be augmented by the indices corresponding to Z_R, giving a final $\mathcal{B}$ for which the corresponding KKT-matrix K_B is nonsingular.

7.3. Numerical results

Numerical results from a simple MATLAB implementation of Algorithm PDQP were obtained for a set of 121 convex QPs in standard interface format (SIF). The problems were selected based on the dimension of the constraint matrix A in (7.2). In particular, the test set includes all QP problems for which the smaller of m and n is of the order of 500 or less. This gave 121 QPs ranging in size from BQP1VAR (one variable and one constraint) to LINCONT (1257 variables and 419 constraints).

In order to judge how the proposed method compares to a conventional two-phase active-set method, the same 121 problems were solved using the convex QP solver SQOPT [26], which is a Fortran implementation of a two-phase (primal) reduced-gradient active-set method for large-scale QP. All SQOPT runs were made using the default parameter options.

Both PDQP and SQOPT are terminated at a point (x, y, z) that satisfies the equations conditions of (2.6) modified to conform to the constraint format of (7.2). The feasibility and optimality tolerances are given by $\epsilon_{\mathtt{fea}} = 10^{-6}$ and $\epsilon_{\mathtt{opt}} = 10^{-6}$, respectively. For a given $\epsilon_{\mathtt{opt}}$, PDQP and SQOPT terminate when

$$\max_{i\in\mathcal{B}} |z_i| \le \epsilon_{\mathtt{opt}} \|y\|_\infty, \quad \text{and} \quad \begin{cases} z_i \ge -\epsilon_{\mathtt{opt}} \|y\|_\infty & \text{if } x_i \ge -\ell_i,\ i \in \mathcal{N}; \\ z_i \le \epsilon_{\mathtt{opt}} \|y\|_\infty & \text{if } x_i \le u_i,\ i \in \mathcal{N}. \end{cases} \tag{7.3}$$

Both PDQP and SQOPT use the EXPAND procedure of Gill et al. [27] to allow the variables (x, s) to move outside their bounds by as much as $\epsilon_{\mathtt{fea}}$.

A summary of the results is given in Table 1. The first four columns give the name of the problem, the number of linear constraints m, the number of variables n, and the optimal objective value Objective. The next two columns summarize the SQOPT result for the given problem, with Phs1 and Itn giving the phase-one iterations and iteration total, respectively. The last four columns summarize the results for the PDQP implementation. The first column gives the total number of primal and dual iterations Itn. The second column gives the order in which the primal and dual algorithms were applied, with PD indicating the “primal-first” strategy, and DP the “dual-first” strategy. The final two columns, headed by p-Itn, and d-Itn, give the iterations required for the primal method and the dual method, respectively.

Of the 121 problems tested, two (LINCONT and NASH) are known to be infeasible. This infeasibility was identified correctly by both SQOPT and PDQP. In total, SQOPT solved 117 of the remaining 119 problems, but declared (incorrectly) that problems RDW2D51U and RDW2D52U are unbounded. PDQP solved the same number of problems, but failed to achieve the required accuracy for the problems RDW2D51B and RDW2D52F.

In these two cases, the final objective values computed by PDQP were 1.0947648E-02 and 1.0491239E-02 respectively, instead of the optimal values 1.0947332e-02 and 1.0490828e-02. (The five RDW2D5* problems in the test set are known to be difficult to solve, see Gill and Wong [30].)

If the failed and infeasible runs are excluded, Algorithm PDQP required the same or fewer number of iterations than SQOPT on 73 of the 115 QP problems solved to optimality by both methods. This constitutes 64% of the problems solved. Of this 64%, the "dual-first" strategy made up 44% of the cases and 56% of the improvements were associated with the "primal-first" strategy.

Table 1: Results for PDQP and SQOPT on 121 CUTEst QPs.

				SQOPT		PDQP			
Name	m	n	Objective	Phs1	Itn	Itn	Order	P-Itn	D-Itn
ALLINQP	50	100	-9.1592833E+00	0	45	65	PD	63	2
AUG2DQP	100	220	1.7797215E+02	8	116	440	PD	326	114
AUG3D	27	156	8.3333333E-02	0	45	45	DP	0	45
AVGASA	10	8	-4.6319255E+00	5	8	5	DP	0	5
AVGASB	10	8	-4.4832193E+00	5	8	7	DP	0	7
BIGGSB1	1	100	1.5000000E-02	0	103	101	PD	101	0
BQP1VAR	1	1	0.0000000E+00	0	1	1	DP	0	1
BQPGABIM	1	50	-3.7903432E-05	0	36	7	PD	7	0
BQPGASIM	1	50	-5.5198140E-05	0	40	8	PD	8	0
CHENHARK	1	100	-2.0000000E+00	0	132	32	DP	0	32
CVXBQP1	1	100	2.2725000E+02	0	100	119	DP	2	117
CVXQP1	50	100	1.1590718E+04	5	67	91	DP	1	90
CVXQP2	25	100	8.1209404E+03	2	82	85	DP	2	83
CVXQP3	75	100	1.1943432E+04	17	46	113	DP	2	111
DEGENQP	1005	10	0.0000000E+00	0	6	18	PD	18	0
DTOC3	18	29	2.2459038E+02	1	10	17	DP	0	17
DUAL1	1	85	3.5012967E-02	0	88	88	PD	88	0
DUAL2	1	96	3.3733671E-02	0	99	99	PD	99	0
DUAL3	1	111	1.3575583E-01	0	106	106	PD	106	0
DUAL4	1	75	7.4609064E-01	0	61	61	PD	61	0
DUALC1	215	9	6.1552516E+03	1	9	4	DP	0	4
DUALC2	229	7	3.5513063E+03	2	4	4	DP	0	4
DUALC5	278	8	4.2723256E+02	1	7	6	DP	0	6
DUALC8	503	8	1.8309361E+04	4	6	8	DP	0	8
GENHS28	8	10	9.2717369E-01	0	3	5	DP	0	5
GMNCASE2	1050	175	-9.9444495E-01	18	99	91	DP	0	91
GMNCASE3	1050	175	1.5251466E+00	31	100	86	DP	0	86
GMNCASE4	350	175	5.9468849E+03	74	171	175	DP	0	175
GOULDQP2	199	399	9.0045697E-06	0	213	419	DP	0	419
GOULDQP3	199	399	5.6732908E-02	0	200	406	PD	205	201
GRIDNETA	100	180	9.5242163E+01	5	35	134	PD	81	53
GRIDNETB	100	180	4.7268237E+01	0	81	97	DP	0	97
GRIDNETC	100	180	4.8352347E+01	6	93	153	DP	0	153
HS3	1	2	0.0000000E+00	0	2	1	PD	1	0
HS3MOD	1	2	1.2325951E-32	0	2	1	PD	1	0
HS21	1	2	-9.9960000E+01	0	1	0	PD	0	0
HS28	1	3	1.2325951E-32	0	2	0	PD	0	0
HS35	1	3	1.1111111E-01	0	5	1	DP	0	1
HS35I	1	3	1.1111111E-01	0	5	1	DP	0	1
HS35MOD	1	3	2.5000000E-01	0	1	0	PD	0	0
HS44	6	4	-1.5000000E+01	0	2	4	PD	4	0
HS44NEW	6	4	-1.5000000E+01	0	4	9	PD	9	0
HS51	3	5	-8.8817841E-16	0	2	0	DP	0	0
HS52	3	5	5.3266475E+00	0	2	1	DP	0	1
HS53	3	5	4.0930232E+00	0	2	1	DP	0	1
HS76	3	4	-4.6818181E+00	0	4	4	DP	0	4

Table 1: Results for PDQP and SQOPT on 121 CUTEst QPs.
(continued)

				SQOPT		PDQP			
Name	m	n	Objective	Phs1	Itn	Itn	Order	P-Itn	D-Itn
HS76I	3	4	-4.6818181E+00	0	4	4	DP	0	4
HS118	17	15	6.6482045E+02	0	21	23	DP	0	23
HS268	5	5	7.2759576E-12	0	8	0	PD	0	0
HUES-MOD	2	100	3.4829823E+07	1	103	7	DP	0	7
HUESTIS	2	100	3.4829823E+09	1	103	7	DP	0	7
JNLBRNG1	1	529	-1.8004556E-01	0	292	82	PD	82	0
JNLBRNG2	1	529	-4.1023852E+00	0	252	42	PD	42	0
JNLBRNGA	1	529	-3.0795806E-01	0	292	292	PD	292	0
JNLBRNGB	1	529	-6.5067871E+00	0	247	247	PD	247	0
KSIP	1001	20	5.7579792E-01	0	2847	36	DP	0	36
LINCONT	419	1257	infeasible	138	138^{i}	304^{i}	DP	0	304
LISWET1	100	106	2.6072632E-01	0	52	401	DP	0	401
LISWET2	100	106	2.5876398E-01	0	63	378	DP	0	378
LISWET3	100	106	2.5876398E-01	0	64	378	DP	0	378
LISWET4	100	106	2.5876399E-01	0	61	378	DP	0	378
LISWET5	100	106	2.5876410E-01	0	58	378	DP	0	378
LISWET6	100	106	2.5876390E-01	0	67	378	DP	0	378
LISWET7	100	106	2.5895785E-01	0	68	378	DP	0	378
LISWET8	100	106	2.5747454E-01	0	94	417	DP	0	417
LISWET9	100	103	2.1543892E+01	0	28	263	DP	0	263
LISWET10	100	106	2.5874831E-01	0	68	378	DP	0	378
LISWET11	100	106	2.5704145E-01	0	68	379	DP	0	379
LISWET12	100	106	9.1994948E+00	0	37	460	DP	0	460
LOTSCHD	7	12	2.3984158E+03	4	8	16	DP	0	16
MOSARQP1	10	100	-1.5420010E+02	0	102	52	DP	0	52
MOSARQP2	10	100	-2.0651670E+02	0	100	33	DP	0	33
NASH	24	72	infeasible	5	5^{i}	24^{i}	DP	0	24
OBSTCLAE	1	529	1.6780270E+00	0	605	178	DP	0	178
OBSTCLAL	1	529	1.6780270E+00	0	263	263	PD	263	0
OBSTCLBL	1	529	6.5193252E+00	0	469	469	PD	469	0
OBSTCLBM	1	529	6.5193252E+00	0	484	189	DP	0	189
OBSTCLBU	1	529	6.5193252E+00	0	303	303	PD	303	0
OSLBQP	1	8	6.2500000E+00	0	6	0	PD	0	0
PENTDI	1	500	-7.5000000E-01	0	2	2	PD	2	0
POWELL20	100	100	5.2703125E+04	49	52	99	DP	0	99
PRIMAL1	85	325	-3.5012967E-02	0	217	70	PD	70	0
PRIMAL2	96	649	-3.3733671E-02	0	407	97	PD	97	0
PRIMAL3	111	745	-1.3575583E-01	0	1223	102	PD	102	0
PRIMAL4	75	1489	-7.4609064E-01	0	1264	63	PD	63	0
PRIMALC1	9	230	-6.1552516E+03	0	18	5	PD	5	0
PRIMALC2	7	231	-3.5513063E+03	0	3	5	PD	5	0
PRIMALC5	8	287	-4.2723256E+02	0	10	6	PD	6	0
PRIMALC8	8	520	-1.8309432E+04	0	30	6	PD	6	0
QPCBLEND	74	83	-7.8425425E-03	0	111	182	PD	182	0
QPCBOEI1	351	384	1.1503952E+07	415	1055	793	PD	395	398
QPCBOEI2	166	143	8.1719635E+06	142	315	340	PD	163	177
QPCSTAIR	356	467	6.2043917E+06	210	433	970	PD	645	325
QUDLIN	1	420	-8.8290000E+06	0	419	419	PD	419	0
RDW2D51F	225	578	1.1209939E-03	29	29	217	DP	0	217
RDW2D51U	225	578	8.3930032E-04	14	16^{f}	219	DP	0	219
RDW2D52B	225	578	1.0947648E-02	349	488	316^{f}	DP	0	314
RDW2D52F	225	578	1.0491239E-02	29	191	414^{f}	DP	0	414
RDW2D52U	225	578	1.0455316E-02	15	318^{f}	219	DP	0	219
S268	5	5	7.2759576E-12	0	8	0	PD	0	0
SIM2BQP	1	2	0.0000000E+00	0	1	1	PD	1	0
SIMBQP	1	2	6.0185310E-31	0	2	1	PD	1	0
STCQP1	30	65	4.9452085E+02	8	53	20	DP	0	20
STCQP2	128	257	1.4294017E+03	80	215	73	DP	0	73
STEENBRA	108	432	1.6957674E+04	14	89	177	PD	2	175

Table 1: Results for PDQP and SQOPT on 121 CUTEst QPs.
(continued)

				SQOPT		PDQP			
Name	m	n	Objective	Phs1	Itn	Itn	Order	P-Itn	D-Itn
TAME	1	2	3.0814879E-33	0	1	1	PD	1	0
TORSION1	1	484	-4.5608771E-01	0	256	256	PD	256	0
TORSION2	1	484	-4.5608771E-01	0	544	144	DP	0	144
TORSION3	1	484	-1.2422498E+00	0	112	112	PD	112	0
TORSION4	1	484	-1.2422498E+00	0	689	288	DP	0	288
TORSION5	1	484	-2.8847068E+00	0	40	40	PD	40	0
TORSION6	1	484	-2.8847068E+00	0	708	360	DP	0	360
TORSIONA	1	484	-4.1611287E-01	0	272	272	PD	272	0
TORSIONB	1	484	-4.1611287E-01	0	529	128	DP	0	128
TORSIONC	1	484	-1.1994864E+00	0	120	120	PD	120	0
TORSIOND	1	484	-1.1994864E+00	0	681	280	DP	0	280
TORSIONE	1	484	-2.8405962E+00	0	40	40	PD	40	0
TORSIONF	1	484	-2.8405962E+00	0	761	360	DP	0	360
UBH1	60	99	1.1473520E+00	11	40	112	DP	0	112
YAO	20	22	2.3988296E+00	0	2	20	DP	0	20
ZECEVIC2	2	2	-4.1250000E+00	0	4	5	PD	5	0

i = infeasible, f = failed

Algorithm PDQP would not be recommended for solving a linear program (LP). Nevertheless, it was applied to 16 LPs from the CUTEst test set. The results are summarized in Table 2.

Table 2: Results for PDQP and SQOPT on 16 CUTEst LPs.

				SQOPT		PDQP			
Name	m	n	Objective	Phs1	Itn	Itn	Order	P-Itn	D-Itn
AGG	488	163	-3.5991767e+07	84	124	200	PD	156	44
DEGENLPA	15	20	-7.6798369e+01	9	11	52	PD	26	26
DEGENLPB	15	20	-3.0742351e+01	9	10	69	PD	37	32
EXTRASIM	1	2	1.0000000e+00	0	0	0	PD	0	0
GOFFIN	50	51	-1.2612134e-13	3	25	100	PD	100	0
MAKELA4	40	21	0.0000000e+00	0	1	42	PD	42	0
MODEL	38	1542	infeasible	12	12^i	44^i	DP	0	44
OET1	1002	3	5.3824312e-01	1	107	22	DP	0	22
OET3	1002	4	4.5049718e-03	1	319	18	DP	0	18
PT	501	2	1.7839423e-01	1	135	20	DP	0	20
S277-280	4	4	5.0761905e+00	1	6	10	DP	0	10
SIMPLLPA	2	2	1.0000000e+00	1	2	5	DP	1	4
SIMPLLPB	3	2	1.1000000e+00	1	1	6	DP	0	6
SSEBLIN	72	194	1.6170600e+07	26	136	262	DP	8	254
SUPERSIM	2	2	6.6666667e-01	1	1	2	DP	0	2
TFI2	101	3	6.4903111e-01	0	34	48	DP	46	2

i = infeasible

8. Summary and Conclusions

A general framework has been proposed for solving a convex quadratic program with general equality constraints and simple lower bounds on the variables. This framework allows the definition of two methods, one primal and one dual, that generate a sequence of iterates that are feasible with respect to the equality constraints associated with the optimality conditions of a general primal-dual form. The primal method maintains feasibility of the primal inequalities while driving the infeasibilities of the dual inequalities to zero. The dual method maintains feasibility of the dual inequalities while moving to satisfy the infeasibilities of the primal inequalities. In each of these methods, the search directions satisfy a KKT system of equations formed from Hessian and constraint components associated with an appropriate column basis. The composition of the basis is specified by an active-set strategy that guarantees the nonsingularity of each set of KKT equations.

Each of the proposed methods is a conventional active-set method in the sense that an initial primal- or dual-feasible point is required. In addition, it has been shown how the bounds of the primal and dual problems may be shifted so as to give a strategy for solving the original problem by solving a pair of coupled quadratic programs, one primal and one dual. An application of this approach is to solve a shifted dual QP for a feasible point for the primal (or *vice versa*), thereby avoiding the need for a traditional feasibility phase that ignores the properties of the objective function.

The numerical results indicate that the proposed primal, dual, and coupled primal-dual QP methods can be efficient relative to existing two-phase active-set methods. Future work will focus on the application of the proposed methods to situations in which a series of related QPs must be solved, for example, in sequential quadratic programming methods and methods for mixed-integer nonlinear programming.

References

[1] R. A. Bartlett and L. T. Biegler. QPSchur: a dual, active-set, Schur-complement method for large-scale and structured convex quadratic programming. *Optim. Eng.*, 7(1):5–32, 2006. 3

[2] E. M. L. Beale. An introduction to Beale's method of quadratic programming. In J. Abadie, editor, *Nonlinear programming*, pages 143–153, Amsterdam, the Netherlands, 1967. North Holland. 3

[3] E. M. L. Beale and R. Benveniste. Quadratic programming. In H. J. Greenberg, editor, *Design and Implementation of Optimization Software*, pages 249–258. Sijthoff and Noordhoff, The Netherlands, 1978. 3

[4] M. J. Best. An algorithm for the solution of the parametric quadratic programming problem. CORR 82-14, Department of Combinatorics and Optimization, University of Waterloo, Canada, 1982. 4

[5] M. J. Best. An algorithm for the solution of the parametric quadratic programming problem. In H. Fischer, B. Riedmüller, and S. Schäffler, editors, *Applied Mathematics and Parallel Computing: Festschrift for Klaus Ritter*, pages 57–76. Physica, Heidelberg, 1996. 4

[6] N. L. Boland. A dual-active-set algorithm for positive semi-definite quadratic programming. *Math. Programming*, 78(1, Ser. A):1–27, 1997. 3

[7] I. Bongartz, A. R. Conn, N. I. M. Gould, and P. L. Toint. CUTE: Constrained and unconstrained testing environment. *ACM Trans. Math. Software*, 21(1):123–160, 1995. 24

[8] J. R. Bunch and L. Kaufman. Some stable methods for calculating inertia and solving symmetric linear systems. *Math. Comput.*, 31:163–179, 1977. 18

[9] J. R. Bunch and L. Kaufman. A computational method for the indefinite quadratic programming problem. *Linear Algebra Appl.*, 34:341–370, 1980. 3

[10] J. R. Bunch and B. N. Parlett. Direct methods for solving symmetric indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 8:639–655, 1971. 18

[11] C. C. Cartis and N. I. M. Gould. Finding a point in the relative interior of a polyhedron. Report RAL-TR-2006-016, Rutherford Appleton Laboratory, Oxon, UK, December 2006. 4

[12] F. E. Curtis, Z. Han, and D. P. Robinson. A Globally Convergent Primal-Dual Active-Set Framework for Large-Scale Convex Quadratic Optimization. *Computational Optimization and Applications*, DOI: 10.1007/s10589-014-9681-9, 2014. 4

[13] I. S. Duff. MA57—a code for the solution of sparse symmetric definite and indefinite systems. *ACM Trans. Math. Software*, 30(2):118–144, 2004. 18

[14] I. S. Duff and J. K. Reid. The multifrontal solution of indefinite sparse symmetric linear equations. *ACM Trans. Math. Software*, 9:302–325, 1983. 18

[15] H. Ferreau, C. Kirches, A. Potschka, H. Bock, and M. Diehl. qpOASES: a parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, pages 1–37, 2014. 4

[16] H. J. Ferreau, H. G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. *Internat. J. Robust Nonlinear Control*, 18(8):816–830, 2008. 4

[17] R. Fletcher. A general quadratic programming algorithm. *J. Inst. Math. Applics.*, 7:76–91, 1971. 3

[18] R. Fletcher. Factorizing symmetric indefinite matrices. *Linear Algebra Appl.*, 14:257–272, 1976. 18

[19] R. Fletcher. Stable reduced Hessian updates for indefinite quadratic programming. *Math. Program.*, 87(2, Ser. B):251–264, 2000. Studies in algorithmic optimization. 3

[20] A. Forsgren. Inertia-controlling factorizations for optimization algorithms. *Appl. Num. Math.*, 43:91–107, 2002. 18

[21] A. Forsgren and W. Murray. Newton methods for large-scale linear equality-constrained minimization. *SIAM J. Matrix Anal. Appl.*, 14:560–587, 1993. 18

[22] P. E. Gill, N. I. M. Gould, W. Murray, M. A. Saunders, and M. H. Wright. A weighted Gram-Schmidt method for convex quadratic programming. *Math. Program.*, 30:176–195, 1984. 3

[23] P. E. Gill and W. Murray. Numerically stable methods for quadratic programming. *Math. Program.*, 14:349–372, 1978. 3

[24] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for QPOPT 1.0: a Fortran package for quadratic programming. Report SOL 95-4, Department of Operations Research, Stanford University, Stanford, CA, 1995. 3

[25] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Rev.*, 47:99–131, 2005. 24

[26] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for SQOPT Version 7: Software for large-scale linear and quadratic programming. Numerical Analysis Report 06-1, Department of Mathematics, University of California, San Diego, La Jolla, CA, 2006. 3, 25

[27] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. A practical anti-cycling procedure for linearly constrained optimization. *Math. Program.*, 45:437–474, 1989. 25

[28] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. A Schur-complement method for sparse quadratic programming. In M. G. Cox and S. J. Hammarling, editors, *Reliable Numerical Computation*, pages 113–138. Oxford University Press, 1990. 3

[29] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. Inertia-controlling methods for general quadratic programming. *SIAM Rev.*, 33(1):1–36, 1991. 3, 7

[30] P. E. Gill and E. Wong. Methods for convex and general quadratic programming. *Math. Prog. Comp.*, 7(1):71–112, 2015. 3, 7, 26

[31] D. Goldfarb and A. Idnani. A numerically stable dual method for solving strictly convex quadratic programs. *Math. Programming*, 27(1):1–33, 1983. 3

[32] N. I. M. Gould. An algorithm for large-scale quadratic programming. *IMA J. Numer. Anal.*, 11(3):299–324, 1991. 3

[33] N. I. M. Gould, D. Orban, and P. L. Toint. CUTEr and SifDec: A constrained and unconstrained testing environment, revisited. *ACM Trans. Math. Software*, 29(4):373–394, 2003. 3, 24

[34] N. I. M. Gould, D. Orban, and P. L. Toint. GALAHAD, a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization. *ACM Trans. Math. Software*, 29(4):353–372, Dec 2003. 3

[35] N. I. M. Gould and P. L. Toint. An iterative working-set method for large-scale nonconvex quadratic programming. *Appl. Numer. Math.*, 43(1-2):109–128, 2002. 19th Dundee Biennial Conference on Numerical Analysis (2001). 3

[36] N. I. M. Gould and P. L. Toint. Numerical methods for large-scale non-convex quadratic programming. In *Trends in industrial and applied mathematics (Amritsar, 2001)*, volume 72 of *Appl. Optim.*, pages 149–179. Kluwer Acad. Publ., Dordrecht, 2002. 3

[37] S. C. Hoyle. *A Single-Phase Method for Quadratic Programming*. PhD thesis, Report SOL 86-9, Department of Operations Research, Stanford University, Stanford, CA, 1986. 3

[38] H. M. Huynh. *A Large-Scale Quadratic Programming Solver Based on Block-LU Updates of the KKT System*. PhD thesis, Program in Scientific Computing and Computational Mathematics, Stanford University, Stanford, CA, 2008. 3

[39] C. M. Maes. *A Regularized Active-Set Method for Sparse Convex Quadratic Programming*. PhD thesis, Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, August 2010. 3

[40] A. Potschka, C. Kirches, H. Bock, and J. Schlöder. Reliable solution of convex quadratic programs with parametric active set methods. Technical report, Interdisciplinary Center for Scientific Computing, Heidelberg University, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany, November 2010. 4

[41] M. J. D. Powell. On the quadratic programming algorithm of Goldfarb and Idnani. *Math. Programming Stud.*, 25:46–61, 1985. 3

[42] K. Ritter. A method for solving nonlinear maximum problems depending on parameters. *Naval Research Logistics Quarterly*, 14:147–162, 1967. 4

[43] K. Ritter. On parametric linear and quadratic programming. MRC Technical report 2197, University of Wisconsin at Madison, Wisconsin, USA, 1981. 4

[44] J. Stoer. On the realization of the quadratic programming algorithm of Goldfarb and Idnani. In *Vistas in applied mathematics*, Transl. Ser. Math. Engrg., pages 167–180. Optimization Software, New York, 1986. 3

[45] E. Wong. *Active-Set Methods for Quadratic Programming*. PhD thesis, Department of Mathematics, University of California San Diego, La Jolla, CA, 2011. 3

A. Appendix

The appendix concerns some basic results used in previous sections. The first result shows that the nonsingularity of a KKT matrix may be established by checking that the two row blocks $\begin{pmatrix} H & A^T \end{pmatrix}$ and $\begin{pmatrix} A & -M \end{pmatrix}$ have full row rank.

Proposition A.1. *Assume that H and M are symmetric, positive semidefinite matrices. The vectors u and v satisfy*

$$\begin{pmatrix} H & A^T \\ A & -M \end{pmatrix} \begin{pmatrix} u \\ -v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{A.1}$$

if and only if

$$\begin{pmatrix} H \\ A \end{pmatrix} u = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \textit{and} \quad \begin{pmatrix} A^T \\ -M \end{pmatrix} v = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{A.2}$$

Proof. If (A.2) holds, then (A.1) holds, which establishes the "if" direction. Now assume that u and v are vectors such that (A.1) holds. Then,

$$u^T H u - u^T A^T v = 0, \quad \text{and} \quad v^T A u + v^T M v = 0.$$

Adding these equations gives the identity $u^T H u + v^T M v = 0$. But then, the symmetry and semidefiniteness of H and M imply $u^T H u = 0$ and $v^T M v = 0$. This can hold only if $Hu = 0$ and $Mv = 0$. If $Hu = 0$ and $Mv = 0$, (A.1) gives $A^T v = 0$ and $Au = 0$, which implies that (A.2) holds. ∎

The next result shows that when checking a subset of the columns of a symmetric positive semidefinite matrix for linear dependence, it is only the diagonal block that is of importance. The off-diagonal block may be ignored.

Proposition A.2. *Let H be a symmetric, positive semidefinite matrix partitioned as*

$$H = \begin{pmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{pmatrix}.$$

Then,

$$\begin{pmatrix} H_{11} \\ H_{12}^T \end{pmatrix} u = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \textit{if and only if} \;\; H_{11} u = 0.$$

Proof. If H is positive semidefinite, then H_{11} is positive semidefinite, and it holds that

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} H_{11} \\ H_{12}^T \end{pmatrix} u = \begin{pmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{pmatrix} \begin{pmatrix} u \\ 0 \end{pmatrix}$$

if and only if

$$0 = \begin{pmatrix} u^T & 0 \end{pmatrix} \begin{pmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{pmatrix} \begin{pmatrix} u \\ 0 \end{pmatrix} = u^T H_{11} u$$

if and only if $H_{11} u = 0$, as required. ∎

In the following propositions, the distinct integers k and l, together with integers from the index sets $\mathcal{B}$ and $\mathcal{N}$ define a partition of $\mathcal{I} = \{1,\ 2,\ \ldots,\ n\}$, i.e., $\mathcal{I} = \mathcal{B} \cup \{k\} \cup \{l\} \cup \mathcal{N}$. If w is any n-vector, the n_B-vector w_B and w_N-vector w_N denote the vectors of components of w associated with $\mathcal{B}$ and $\mathcal{N}$. For the symmetric Hessian H, the matrices H_{BB} and H_{NN} denote the subset of rows and columns of H associated with the sets $\mathcal{B}$ and $\mathcal{N}$ respectively. The unsymmetric matrix of components h_{ij} with $i \in \mathcal{B}$ and $j \in \mathcal{N}$ will be denoted by H_{BN}. Similarly, A_B and A_N denote the matrices of columns associated with $\mathcal{B}$ and $\mathcal{N}$.

The next result concerns the row rank of the $\begin{pmatrix} A & -M \end{pmatrix}$ block of the KKT matrix.

Proposition A.3. *If the matrix* $\begin{pmatrix} a_l & a_k & A_B & -M \end{pmatrix}$ *has full row rank, and there exist* Δx_l, Δx_k, Δx_B, *and* Δy *such that* $a_l \Delta x_l + a_k \Delta x_k + A_B \Delta x_B + M \Delta y = 0$ *with* $\Delta x_k \neq 0$, *then* $\begin{pmatrix} a_l & A_B & -M \end{pmatrix}$ *has full row rank.*

Proof. It must be established that $u^T \begin{pmatrix} a_l & A_B & -M \end{pmatrix} = 0$ implies that $u = 0$. For a given u, let $\gamma = -u^T a_k$, so that

$$\begin{pmatrix} u^T & \gamma \end{pmatrix} \begin{pmatrix} a_l & a_k & A_B & -M \\ & 1 & & \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}.$$

Then,

$$0 = \begin{pmatrix} u^T & \gamma \end{pmatrix} \begin{pmatrix} a_l & a_k & A_B & -M \\ & 1 & & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_k \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \gamma\, \Delta x_k.$$

As $\Delta x_k \neq 0$, it must hold that $\gamma = 0$, in which case

$$u^T \begin{pmatrix} a_l & a_k & A_B & -M \end{pmatrix} = 0.$$

As $\begin{pmatrix} a_l & a_k & A_B & -M \end{pmatrix}$ has full row rank by assumption, it follows that $u = 0$ and $\begin{pmatrix} a_l & A_B & -M \end{pmatrix}$ must have full row rank. ∎

An analogous result holds concerning the $\begin{pmatrix} H & A^T \end{pmatrix}$ block of the KKT matrix.

Proposition A.4. *If* $\begin{pmatrix} H_{BB} & A_B^T \end{pmatrix}$ *has full row rank, and there exist quantities* Δx_N, Δx_B, Δy, *and* Δz_k *such that*

$$\begin{pmatrix} h_{Nk}^T & h_{Bk}^T & a_k^T & 1 \\ h_{BN} & H_{BB} & A_B^T & \end{pmatrix} \begin{pmatrix} \Delta x_N \\ \Delta x_B \\ -\Delta y \\ -\Delta z_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{A.3}$$

with $\Delta z_k \neq 0$, *then the matrix*

$$\begin{pmatrix} h_{kk} & h_{Bk}^T & a_k^T \\ h_{Bk} & H_{BB} & A_B^T \end{pmatrix}$$

has full row rank.

Proof. Let $\begin{pmatrix} \mu & v^T \end{pmatrix}$ be any vector such that

$$\begin{pmatrix} \mu & v^T \end{pmatrix} \begin{pmatrix} h_{Nk}^T & h_{Bk}^T & a_k^T \\ h_{BN} & H_{BB} & A_B^T \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}.$$

The assumed identity (A.3) gives

$$0 = \begin{pmatrix} \mu & v^T \end{pmatrix} \begin{pmatrix} h_{Nk}^T & h_{Bk}^T & a_k^T \\ h_{BN} & H_{BB} & A_B^T \end{pmatrix} \begin{pmatrix} \Delta x_N \\ \Delta x_B \\ -\Delta y \end{pmatrix} = \mu \, \Delta z_k.$$

As $\Delta z_k \neq 0$ by assumption, it must hold that $\mu = 0$. The full row rank of $\begin{pmatrix} H_{BB} & A_B^T \end{pmatrix}$ then gives $v = 0$ and

$$\begin{pmatrix} h_{Nk}^T & h_{Bk}^T & a_k^T \\ h_{BN} & H_{BB} & A_B^T \end{pmatrix}$$

must have full row rank. Proposition A.1 implies that this is equivalent to

$$\begin{pmatrix} h_{kk} & h_{Bk}^T & a_k^T \\ h_{Bk} & H_{BB} & A_B^T \end{pmatrix}$$

having full row rank. ∎

The next proposition concerns the primal subiterations when the constraint index k is moved from $\mathcal{B}$ to $\mathcal{N}$. In particular, it is shown that the K_l matrix is nonsingular after a subiteration.

Proposition A.5. *Assume that* $(\Delta x_l,\, \Delta x_k,\, \Delta x_B,\, -\Delta y,\, -\Delta z_l)$ *is the unique solution of the equations*

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{Bl}^T & a_l^T & 1 \\ h_{kl} & h_{kk} & h_{Bk}^T & a_k^T & \\ h_{Bl} & h_{Bk} & H_{BB} & A_B^T & \\ a_l & a_k & A_B & -M & \\ 1 & & & & -1 \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_k \\ \Delta x_B \\ -\Delta y \\ -\Delta z_l \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \tag{A.4}$$

and that $\Delta x_k \neq 0$. *Then, the matrices* K_l *and* K_k *are nonsingular, where*

$$K_l = \begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix} \quad \text{and} \quad K_k = \begin{pmatrix} h_{kk} & h_{Bk}^T & a_k^T \\ h_{Bk} & H_{BB} & A_B^T \\ a_k & A_B & -M \end{pmatrix}.$$

Proof. By assumption, the equations (A.4) have a unique solution with $\Delta x_k \neq 0$. This implies that there is no solution of the overdetermined equations

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{Bl}^T & a_l^T & 1 \\ h_{kl} & h_{kk} & h_{Bk}^T & a_k^T & \\ h_{Bl} & h_{Bk} & H_{BB} & A_B^T & \\ a_l & a_k & A_B & -M & \\ 1 & & & & -1 \\ & 1 & & & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_k \\ \Delta x_B \\ -\Delta y \\ -\Delta z_l \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}. \tag{A.5}$$

Given an arbitrary matrix D and nonzero vector f, the fundamental theorem of linear algebra implies that if $Dw = f$ has no solution, then there exists a vector v such that $v^Tf \neq 0$. The application of this result to (A.5) implies the existence of a nontrivial vector $(\Delta\widetilde{x}_l,\, \Delta\widetilde{x}_k,\, \Delta\widetilde{x}_{\scriptscriptstyle B},\, -\Delta\widetilde{y},\, -\Delta\widetilde{z}_l,\, -\Delta\widetilde{z}_k)$ such that

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{{\scriptscriptstyle B}l}^T & a_l^T & 1 & \\ h_{kl} & h_{kk} & h_{{\scriptscriptstyle B}k}^T & a_k^T & & 1 \\ h_{{\scriptscriptstyle B}l} & h_{{\scriptscriptstyle B}k} & H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T & & \\ a_l & a_k & A_{\scriptscriptstyle B} & -M & & \\ 1 & & & & -1 & \end{pmatrix} \begin{pmatrix} \Delta\widetilde{x}_l \\ \Delta\widetilde{x}_k \\ \Delta\widetilde{x}_{\scriptscriptstyle B} \\ -\Delta\widetilde{y} \\ -\Delta\widetilde{z}_l \\ -\Delta\widetilde{z}_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{A.6}$$

with $\Delta\widetilde{z}_l \neq 0$. The last equation of (A.6) gives $\Delta\widetilde{x}_l + \Delta\widetilde{z}_l = 0$, in which case $\Delta\widetilde{x}_l\Delta\widetilde{z}_l = -\Delta\widetilde{z}_l^2 < 0$ because $\Delta\widetilde{z}_l \neq 0$. Any solution of (A.6) may be viewed as a solution of the equations $H\Delta\widetilde{x} - A^T\Delta\widetilde{y} - \Delta\widetilde{z} = 0$, $A\Delta\widetilde{x} + M\Delta\widetilde{y} = 0$, $\Delta\widetilde{z}_{\scriptscriptstyle B} = 0$, and $\Delta\widetilde{x}_i = 0$ for $i \in \{1, 2, \dots, n\} \setminus \{l\} \setminus \{k\}$. An argument similar to that used to establish Proposition 2.2 gives

$$\Delta\widetilde{x}_l\Delta\widetilde{z}_l + \Delta\widetilde{x}_k\Delta\widetilde{z}_k \geq 0,$$

which implies that $\Delta\widetilde{x}_k\Delta\widetilde{z}_k > 0$, with $\Delta\widetilde{x}_k \neq 0$ and $\Delta\widetilde{z}_k \neq 0$.

As the search direction is unique, it follows from (A.4) that $\begin{pmatrix} h_{{\scriptscriptstyle B}l} & H_{{\scriptscriptstyle B}k} & H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T \end{pmatrix}$ has full row rank, and Proposition A.2 implies that $\begin{pmatrix} H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T \end{pmatrix}$ has full row rank. Hence, as $\Delta\widetilde{z}_l \neq 0$, it follows from (A.6) and Proposition A.4 that the matrix

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{{\scriptscriptstyle B}l}^T & a_l^T \\ h_{{\scriptscriptstyle B}l} & h_{{\scriptscriptstyle B}k} & H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T \end{pmatrix}$$

has full row rank, which is equivalent to the matrix

$$\begin{pmatrix} h_{ll} & h_{{\scriptscriptstyle B}l}^T & a_l^T \\ h_{{\scriptscriptstyle B}l} & H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T \end{pmatrix}$$

having full row rank by Proposition A.2,

Again, the search direction is unique and (A.4) implies that $\begin{pmatrix} a_l & a_k & A_{\scriptscriptstyle B} & -M \end{pmatrix}$ has full row rank. As $\Delta\widetilde{x}_k \neq 0$, Proposition A.3 implies that $\begin{pmatrix} a_l & A_{\scriptscriptstyle B} & -M \end{pmatrix}$ must have full row rank. Consequently, Proposition A.1 implies that K_l is nonsingular.

As $\Delta\widetilde{x}_k$, $\Delta\widetilde{x}_l$, $\Delta\widetilde{z}_k$ and $\Delta\widetilde{z}_l$ are all nonzero, the roles of k and l may be reversed to give the result that K_k is nonsingular. ∎

The next proposition concerns the situation when a constraint index k is moved from $\mathcal{N}$ to $\mathcal{B}$ in a dual subiteration. In particular, it is shown that the resulting matrix $K_{\scriptscriptstyle B}$ defined after a subiteration is nonsingular.

Proposition A.6. *Assume that there is a unique solution of the equations*

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{{\scriptscriptstyle B}l}^T & a_l^T & 1 & \\ h_{kl} & h_{kk} & h_{{\scriptscriptstyle B}k}^T & a_k^T & & 1 \\ h_{{\scriptscriptstyle B}l} & h_{{\scriptscriptstyle B}k} & H_{\scriptscriptstyle BB} & A_{\scriptscriptstyle B}^T & & \\ a_l & a_k & A_{\scriptscriptstyle B} & -M & & \\ 1 & & & & -1 & \\ & 1 & & & & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_k \\ \Delta x_{\scriptscriptstyle B} \\ -\Delta y \\ -\Delta z_l \\ -\Delta z_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \tag{A.7}$$

with $\Delta z_k \neq 0$. Then, the matrices K_l and K_k are nonsingular, where

$$K_l = \begin{pmatrix} h_{ll} & h_{Bl}^T & a_l^T \\ h_{Bl} & H_{BB} & A_B^T \\ a_l & A_B & -M \end{pmatrix}, \quad \textit{and} \quad K_k = \begin{pmatrix} h_{kk} & h_{Bk}^T & a_k^T \\ h_{Bk} & H_{BB} & A_B^T \\ a_k & A_B & -M \end{pmatrix}.$$

Proof. As (A.7) has a unique solution with $\Delta z_k \neq 0$, there is no solution of

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{Bl}^T & a_l^T & 1 \\ h_{kl} & h_{kk} & h_{Bk}^T & a_k^T & \\ h_{Bl} & h_{Bk} & H_{BB} & A_B^T & \\ a_l & a_k & A_B & -M & \\ 1 & & & & -1 \\ & 1 & & & \end{pmatrix} \begin{pmatrix} \Delta x_l \\ \Delta x_k \\ \Delta x_B \\ -\Delta y \\ -\Delta z_l \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}. \tag{A.8}$$

The fundamental theorem of linear algebra applied to (A.8) implies the existence of a solution of

$$\begin{pmatrix} h_{ll} & h_{kl} & h_{Bl}^T & a_l^T & 1 & \\ h_{kl} & h_{kk} & h_{Bk}^T & a_k^T & & 1 \\ h_{Bl} & h_{Bk} & H_{BB} & A_B^T & & \\ a_l & a_k & A_B & -M & & \\ 1 & & & & -1 & \end{pmatrix} \begin{pmatrix} \Delta\widetilde{x}_l \\ \Delta\widetilde{x}_k \\ \Delta\widetilde{x}_B \\ -\Delta\widetilde{y} \\ -\Delta\widetilde{z}_l \\ -\Delta\widetilde{z}_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{A.9}$$

with $\Delta\widetilde{z}_l \neq 0$. It follows from (A.9) that $\Delta\widetilde{x}_l + \Delta\widetilde{z}_l = 0$. As $\Delta\widetilde{z}_l \neq 0$, this implies $\Delta\widetilde{x}_l\Delta\widetilde{z}_l < 0$. The solution of (A.9) may be regarded as a solution of the homogeneous equations $H\Delta x - A^T\Delta y - \Delta z = 0$, $A\Delta x + M\Delta y = 0$, with $\Delta z_i = 0$, for $i \in \mathcal{B}$, and $\Delta x_i = 0$, for $i \in \{1, \ldots, n\} \setminus \{k\} \setminus \{l\}$. Hence, Proposition 2.2 gives

$$\Delta\widetilde{x}_l\Delta\widetilde{z}_l + \Delta\widetilde{x}_k\Delta\widetilde{z}_k \geq 0,$$

so that $\Delta\widetilde{x}_k\Delta\widetilde{z}_k > 0$. Hence, it must hold that $\Delta\widetilde{x}_k \neq 0$ and $\Delta\widetilde{z}_k \neq 0$.

As $\Delta\widetilde{x}_k \neq 0$, $\Delta\widetilde{x}_l \neq 0$, $\Delta\widetilde{z}_k \neq 0$ and $\Delta\widetilde{z}_l \neq 0$, the remainder of the proof is analogous to that of Proposition A.5. ∎

The next result gives expressions for the primal and dual objective functions in terms of the computed search directions.

Proposition A.7. *Assume that (x, y, z) satisfies the primal and dual equality constraints*

$$Hx + c - A^Ty - z = 0, \quad \textit{and} \quad Ax + My - b = 0.$$

Consider the partition $\{1, 2, \ldots, n\} = \mathcal{B} \cup \{l\} \cup \mathcal{N}$ such that $x_N + q_N = 0$ and $z_B + r_B = 0$. If the components of the direction $(\Delta x, \Delta y, \Delta z)$ satisfy (2.10), then the primal and dual objective functions for $(\mathrm{PQP}_{q,r})$ and $(\mathrm{DQP}_{q,r})$, i.e.,

$$\begin{aligned} f_P(x, y) &= \tfrac{1}{2}x^THx + \tfrac{1}{2}y^TMy + c^Tx + r^Tx \\ f_D(x, y, z) &= -\tfrac{1}{2}x^THx - \tfrac{1}{2}y^TMy + b^Ty - q^Tz, \end{aligned}$$

satisfy the identities

$$\begin{aligned} f_P(x+\alpha\Delta x, y+\alpha\Delta y) &= f_P(x,y) + \Delta x_l(z_l+r_l)\alpha + \tfrac{1}{2}\Delta x_l\Delta z_l\alpha^2,\\ f_D(x+\alpha\Delta x, y+\alpha\Delta y, z+\alpha\Delta z) &= f_D(x,y,z) - \Delta z_l(x_l+q_l)\alpha - \tfrac{1}{2}\Delta x_l\Delta z_l\alpha^2. \end{aligned}$$

Proof. The directional derivative of the primal objective function is given by

$$\begin{aligned} \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\nabla f_P(x,y) &= \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\begin{pmatrix}Hx+c+r\\ My\end{pmatrix}\\ &= \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\begin{pmatrix}A^Ty+z+r\\ My\end{pmatrix} \qquad\qquad \text{(A.10a)}\\ &= (A\Delta x + M\Delta y)^Ty + \Delta x^T(z+r)\\ &= \Delta x_l(z_l+r_l), \qquad\qquad \text{(A.10b)} \end{aligned}$$

where the identity $Hx+c = A^Ty+z$ has been used in (A.10a) and the identities $A\Delta x + M\Delta y = 0$, $\Delta x_N = 0$ and $z_B + r_B = 0$ have been used in (A.10b).

The curvature in the direction $(\Delta x, \Delta y)$ is given by

$$\begin{aligned} \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\nabla^2 f_P(x,y)\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix} &= \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\begin{pmatrix}H & \\ & M\end{pmatrix}\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}\\ &= \begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\begin{pmatrix}A^T\Delta y + \Delta z\\ M\Delta y\end{pmatrix} \qquad\qquad \text{(A.11a)}\\ &= (A\Delta x + M\Delta y)^T\Delta y + \Delta x^T\Delta z\\ &= \Delta x_l\Delta z_l, \qquad\qquad \text{(A.11b)} \end{aligned}$$

where the identity $H\Delta x - A^T\Delta y - \Delta z = 0$ has been used in (A.11a) and the identities $A\Delta x + M\Delta y = 0$, $\Delta x_N = 0$ and $\Delta z_B = 0$ have been used in (A.11b).

The directional derivative of the dual objective function is given by

$$\begin{aligned} \begin{pmatrix}\Delta x^T & \Delta y^T & \Delta z^T\end{pmatrix}\nabla f_D(x,y,z) &= \begin{pmatrix}\Delta x^T & \Delta y^T & \Delta z^T\end{pmatrix}\begin{pmatrix}-Hx\\ -My+b\\ -q\end{pmatrix} \qquad \text{(A.12a)}\\ &= -\Delta x^THx + \Delta y^T(-My+b) - \Delta z^Tq \qquad \text{(A.12b)}\\ &= -(A^T\Delta y + \Delta z)^Tx + \Delta y^T(-My+b)\\ &\qquad - \Delta z^Tq \qquad\qquad \text{(A.12c)}\\ &= -\Delta y^T(Ax+My-b) - \Delta z^T(x+q) \qquad \text{(A.12d)}\\ &= -\Delta z_l(x_l+q_l), \qquad\qquad \text{(A.12e)} \end{aligned}$$

where the identity $H\Delta x - A^T\Delta y - \Delta z = 0$ has been used in (A.12c) and the identities $Ax + My - b = 0$, $x_N + q_N = 0$ and $\Delta z_B = 0$ have been used in (A.12e).

As z only appears linearly in the dual objective function, it follows from the structure of the Hessian matrices of $f_P(x,y)$ and $f_D(x,y,z)$ that

$$\begin{pmatrix}\Delta x^T & \Delta y^T & \Delta z^T\end{pmatrix}\nabla^2 f_D(x,y,z)\begin{pmatrix}\Delta x\\ \Delta y\\ \Delta z\end{pmatrix} = -\begin{pmatrix}\Delta x^T & \Delta y^T\end{pmatrix}\nabla^2 f_P(x,y)\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}.$$

Consequently, (A.11) gives

$$\begin{pmatrix} \Delta x^T & \Delta y^T & \Delta z^T \end{pmatrix} \nabla^2 f_D(x,y,z) \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -\Delta x_l \Delta z_l.$$

▮

Finally, it is shown that there is no loss of generality in assuming that $\begin{pmatrix} A & M \end{pmatrix}$ has full row rank in $(\mathrm{PQP}_{q,r})$.

Proposition A.8. *There is no loss of generality in assuming that* $\begin{pmatrix} A & M \end{pmatrix}$ *has full row rank in* $(\mathrm{PQP}_{q,r})$.

Proof. Let $(x,\ y,\ z)$ be any vector satisfying (2.1a) and (2.1b). Assume that $\begin{pmatrix} A & M \end{pmatrix}$ has linearly dependent rows, and that $\begin{pmatrix} A & M \end{pmatrix}$ and b may be partitioned conformally such that

$$\begin{pmatrix} A & M \end{pmatrix} = \begin{pmatrix} A_1 & M_{11} & M_{12} \\ A_2 & M_{12}^T & M_{22} \end{pmatrix}, \quad \text{and} \quad b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$

with $\begin{pmatrix} A_1 & M_{11} & M_{12} \end{pmatrix}$ of full row rank, and

$$\begin{pmatrix} A_2 & M_{12}^T & M_{22} \end{pmatrix} = N \begin{pmatrix} A_1 & M_{11} & M_{12} \end{pmatrix}, \tag{A.13}$$

with $A_1 \in \mathbb{R}^{m_1 \times n}$ and $A_2 \in \mathbb{R}^{m_2 \times n}$ for some matrix $N \in \mathbb{R}^{m_2 \times m_1}$. From the linear dependence of the rows of $\begin{pmatrix} A & M \end{pmatrix}$, it follows that x, y and z satisfy (2.1a) and (2.1b) if and only if

$$\begin{aligned} &A_1 x + M_{11} y_1 + M_{12} y_2 - b_1 = 0 \quad \text{and} \quad b_2 = N b_1, \\ &Hx + c - A_1^T y_1 - A_2^T y_2 - z = 0. \end{aligned}$$

It follows from (A.13) that $M_{12} = M_{11} N^T$ and $A_2^T = A_1^T N^T$, so that x, y and z satisfy (2.1a) and (2.1b) if and only if

$$\begin{aligned} &A_1 x + M_{11}(y_1 + N^T y_2) - b_1 = 0 \ \text{ and } \ b_2 = N b_1, \\ &Hx + c - A_1^T (y_1 + N^T y_2) - z = 0. \end{aligned}$$

We may now define $\widetilde{y}_1 = y_1 + N^T y_2$ and replace (2.1a) and (2.1b) by the system

$$\begin{aligned} A_1 x + M_{11} \widetilde{y}_1 - b_1 &= 0, \\ Hx + c - A_1^T \widetilde{y}_1 - z &= 0. \end{aligned}$$

By assumption, $\begin{pmatrix} A_1 & M_{11} & M_{12} \end{pmatrix}$ has full row rank. Proposition A.2 implies that $\begin{pmatrix} A_1 & M_{11} \end{pmatrix}$ has full row rank. This gives an equivalent problem for which $\begin{pmatrix} A_1 & M_{11} \end{pmatrix}$ has full row rank. ▮