

CONSTRAINT REDUCTION FOR LINEAR PROGRAMS WITH MANY INEQUALITY CONSTRAINTS*

ANDRÉ L. TITS†, P.-A. ABSIL‡, AND WILLIAM P. WOESSNER§

Abstract. Consider solving a linear program in standard form where the constraint matrix A is $m \times n$, with $n \gg m \gg 1$. Such problems arise, for example, as the result of finely discretizing a semi-infinite program. The cost per iteration of typical primal-dual interior-point methods on such problems is $O(m^2n)$. We propose to reduce that cost by replacing the normal equation matrix, AD^2A^T , where D is a diagonal matrix, with a “reduced” version (of same dimension), $A_QD_Q^2A_Q^T$, where Q is an index set including the indices of M most nearly active (or most violated) dual constraints at the current iterate, with $M \geq m$ a prescribed integer. This can result in a speedup of close to $n/|Q|$ at each iteration. Promising numerical results are reported for constraint-reduced versions of a dual-feasible affine-scaling algorithm and of Mehrotra’s predictor-corrector method [S. Mehrotra, *SIAM J. Optim.*, 2 (1992), pp. 575–601]. In particular, while it could be expected that neglecting a large portion of the constraints, especially at early iterations, may result in a significant deterioration of the search direction, it appears that the total number of iterations typically remains essentially constant as the size of the reduced constraint set is decreased down to some threshold. In some cases this threshold is a small fraction of the total set. In the case of the affine-scaling algorithm, global convergence and local quadratic convergence are proved.

Key words. linear programming, constraint reduction, column generation, primal-dual interior-point methods, affine scaling, Mehrotra’s predictor-corrector

AMS subject classifications. 90C05, 65K05, 90C06, 90C34, 90C51

DOI. 10.1137/050633421

1. Introduction. Consider a primal-dual linear programming pair in standard form, i.e.,

$$(1.1) \quad \min c^T x \text{ subject to } Ax = b, x \geq 0,$$

$$(1.2) \quad \max b^T y \text{ subject to } A^T y + s = c, s \geq 0,$$

where A has dimensions $m \times n$. The dual problem is equivalently written as

$$(1.3) \quad \max b^T y \text{ subject to } A^T y \leq c.$$

*Received by the editors June 10, 2005; accepted for publication (in revised form) December 9, 2005; published electronically April 21, 2006. The work of the first and third authors was supported in part by the National Science Foundation under grants DMI-9813057 and DMI-0422931, and by the US Department of Energy under grant DEFG0204ER25655. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation or those of the US Department of Energy.

<http://www.siam.org/journals/siopt/17-1/63342.html>

†Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742 (andre@umd.edu).

‡Department of Mathematical Engineering, Université catholique de Louvain, Belgium (<http://www.inma.ucl.ac.be/~absil/>). The work of this author was supported in part by Microsoft Research through a Research Fellowship at Peterhouse, Cambridge. Part of this work was done while this author was a Research Fellow with the Belgian National Fund for Scientific Research (Aspirant du F.N.R.S.) at the University of Liège, and while he was a Postdoctoral Associate with the School of Computational Science of Florida State University.

§Department of Computer Science and Institute for Systems Research, University of Maryland, College Park, MD 20742.

Most algorithms that have been proposed for the numerical solution of such problems belong to one of two classes: simplex methods and interior-point methods. For background on such methods, see, e.g., [NS96] and [Wri97]. In both classes of algorithms, the main computational task at each iteration is the solution of a linear system of equations. In the simplex case, the system has dimension m ; in the interior-point case it has dimensions $2n + m$, but can readily be reduced (“normal equations”) to one of size m at the cost of forming the matrix $H := AS^{-1}XA^T$. Here S and X are diagonal but vary from iteration to iteration, and the cost of forming H , when A is dense, is of the order of m^2n operations at each iteration.

The focus of the present paper is the solution of (1.1)–(1.2) when $n \gg m \gg 1$, i.e., when there are many more variables than equality constraints in the primal, many more inequality constraints than variables in the dual. This includes fine discretizations of “semi-infinite” problems of the form

$$(1.4) \quad \max b^T y \text{ subject to } a(\omega)^T y \leq c(\omega) \quad \forall \omega \in \Omega,$$

where, in the simplest cases, Ω is an interval of the real line. Network problems may also have a disproportionately large number of inequality constraints: For many network problems in dual form, there is one variable for each node of the network and one constraint for each arc or link, so that a linear program associated with a network with m nodes could have up to $O(m^2)$ constraints. Clearly, for such problems one iteration of a standard interior-point method would be computationally much more costly than one iteration of a simplex method. On the other hand, given the large number of vertices in the polyhedral feasible set of (1.3), the number of iterations needed to approach a solution with an interior-point method is likely to be significantly smaller than that needed when a simplex method is used.

Intuitively, when $n \gg m$, most of the constraints in (1.3) are of little or no relevance. Conceivably, if an interior-point search direction were computed based on a much smaller problem, with only a small subset of the constraints, significant progress could still be made toward a solution, provided this subset were astutely selected. Motivated by such consideration, in the present paper we aim at devising interior-point methods for the solution of (1.1)–(1.2) with $n \gg m \gg 1$, with drastically reduced computational cost per iteration. In a sense, such an algorithm would combine the best aspects of simplex methods and interior-point methods in the context of problems for which $n \gg m \gg 1$: each iteration would be effected at low computational cost, yet the iterates would follow an “interior” trajectory rather than being constrained to proceed along edges.

The issue of computing search directions for linear programs of the form (1.3) with $n \gg m$ —or for semi-infinite linear programs (with a continuum of inequality constraints)—based on a small subset of the constraints has been an active area of research for many years. In most cases, the proposed schemes are based on logarithmic barrier (“primal”) interior-point methods. In one approach, known as “column generation” (for the A matrix) or “build-up” (see, e.g., [Ye92, dHRT92, GLY94, Ye97]), constraints are added to (but never deleted from) the constraint set iteratively as they are deemed critical. In particular, the scheme studied in [Ye97] allows for more than one constraint (column) to be added at each step, and it is proved that the algorithm terminates in polynomial time with a bound whose dependence on the constraints is limited to those that are eventually included in the constraint set. In [Ye92, GLY94, Ye97], in the spirit of cutting-plane methods, the successive iterates are infeasible for (1.3) and the algorithm stops as soon as a feasible point is achieved;

while in the approach proposed in [dHRT92] all iterates are feasible for (1.3). Another approach is the “build-down” process (e.g., [Ye90]) by which columns of A are discarded when it is determined that the corresponding constraints are guaranteed not to be active at the solution. Both build-up [dHRT92] and build-down [Ye90] approaches were subsequently combined in [dHRT94], and a complexity analysis for the semi-infinite case was carried out in [LRT99].

In the present paper, a constraint reduction scheme is proposed in the context of *primal-dual* interior-point methods. Global and local quadratic convergence are proved in the case of a primal-dual affine-scaling (PDAS) method. (An early version of this analysis appeared in [Tit99].) Distinctive merits of the proposed scheme are its simplicity and the fact that it can be readily incorporated into other primal-dual interior-point methods. In the scheme’s simplest embodiment, the constraint set is determined “from scratch” at the beginning of each iteration, rather than being updated in a build-up/build-down fashion. Promising numerical results are reported with constraint-reduced versions of the PDAS method and of Mehrotra’s predictor-corrector (MPC) algorithm [Meh92]. Strikingly, while (consistent with conventional wisdom) the unreduced version of MPC significantly outperformed that of PDAS in our random experiments, the reduced version of PDAS performed essentially at the same level, in terms of CPU time, as that of MPC.

The remainder of the paper is organized as follows. In section 2, the basics of primal-dual interior-point methods are reviewed and the computational cost per iteration is analyzed, with special attention paid to possible gains to be achieved in certain steps by ignoring most constraints. Section 3 contains the heart of this paper’s contribution. There, a dual-feasible PDAS algorithm is proposed that features a constraint-reduction scheme. Global and local quadratic convergence of this algorithm are proved, and numerical results are reported that suggest that, even with a simplistic implementation, the constraint-reduction scheme may lead to significant speedup. In section 4, promising numerical results are reported for a similarly reduced MPC algorithm, both with a dual-feasible initial point and with an infeasible initial point. Finally, section 5 is devoted to concluding remarks.

2. Preliminaries. Let

$$\mathbf{n} := \{1, 2, \dots, n\};$$

for $i \in \mathbf{n}$, let $a_i \in \mathbb{R}^m$ denote the i th column of A ; let F be the feasible set for (1.3), i.e.,

$$F := \{y : A^T y \leq c\},$$

and let $F^\circ \subseteq \mathbb{R}^m$ denote the dual strictly feasible set

$$F^\circ := \{y : A^T y < c\}.$$

Also, given $y \in F$, let $I(y)$ denote the index set of active constraints at y , i.e.,

$$I(y) := \{i \in \mathbf{n} : a_i^T y = c_i\}.$$

Given any index set $Q \subseteq \mathbf{n}$, let A_Q denote the $m \times |Q|$ matrix obtained from A by deleting all columns a_i with $i \notin Q$; similarly let x_Q and s_Q denote the vectors of size $|Q|$ obtained from x and s by deleting all entries x_i and s_i with $i \notin Q$. Further, following standard practice, let X denote $\text{diag}(x_i, i \in \mathbf{n})$, and S denotes $\text{diag}(s_i, i \in \mathbf{n})$.

When subscripts, superscripts, or diacritical signs are attached to x and s , they are inherited by x_Q , s_Q , X , and S . The rest of the notation is standard. In particular, $\|\cdot\|$ denotes the Euclidean norm.

Primal-dual interior-point algorithms use search directions based on the Newton step for the solution of the equalities in the Karush–Kuhn–Tucker (KKT) conditions for (1.2), or a perturbation thereof, while maintaining positivity of x and s . Given $\mu > 0$, the perturbed KKT conditions of interest are

$$(2.1a) \quad A^T y + s - c = 0,$$

$$(2.1b) \quad Ax - b = 0,$$

$$(2.1c) \quad Xs = \mu e,$$

$$(2.1d) \quad x, s \geq 0,$$

with $\mu = 0$ yielding the true KKT conditions. Given a current guess (x, y, s) , the Newton step of interest is the solution to the linear system

$$(2.2) \quad \begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_c \\ -r_b \\ -Xs + \mu e \end{bmatrix},$$

where

$$r_b := Ax - b, \quad r_c := A^T y + s - c$$

are the primal and dual *residuals*. Applying block Gaussian elimination to eliminate Δs yields the system (usually referred to as “augmented system”)

$$(2.3a) \quad \begin{bmatrix} 0 & A \\ XA^T & -S \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \end{bmatrix} = \begin{bmatrix} -r_b \\ -Xr_c + Xs - \mu e \end{bmatrix},$$

$$(2.3b) \quad \Delta s = -A^T \Delta y - r_c.$$

With $s > 0$, further elimination of Δx results in the “normal equations”

$$(2.4a) \quad AS^{-1}XA^T \Delta y = -r_b + A(-S^{-1}Xr_c + x - \mu S^{-1}e),$$

$$(2.4b) \quad \Delta s = -A^T \Delta y - r_c,$$

$$(2.4c) \quad \Delta x = -x + \mu S^{-1}e - S^{-1}X\Delta s.$$

Note that (2.4a) is equivalently written as

$$AS^{-1}XA^T \Delta y = b - AS^{-1}(Xr_c + \mu e).$$

For ease of reference, define the Jacobian and “augmented” Jacobian

$$(2.5) \quad J(A, x, s) := \begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix}, \quad J_a(A, x, s) := \begin{bmatrix} 0 & A \\ XA^T & -S \end{bmatrix}.$$

The following result is proven in the appendix.¹

¹Concerning the second claim of Lemma 1, only sufficiency is used in the convergence analysis, but the fact that the listed conditions are in fact necessary and sufficient may be of independent interest. We could not find this result (or even the sufficiency portion) in the literature, so are providing a proof for completeness. We would be grateful to anyone who would point us to a reference for the result.

LEMMA 1. $J_a(A, x, s)$ is nonsingular if and only if $J(A, x, s)$ is. Further, suppose that $x \geq 0$ and $s \geq 0$.² Then $J(A, x, s)$ is nonsingular if and only if the following three conditions hold: (i) $|x_i| + |s_i| > 0$ for all i , (ii) $\{a_i : s_i = 0\}$ is linear independent, and (iii) $\{a_i : x_i \neq 0\}$ spans \mathbb{R}^m .

In the next two sections, two types of primal-dual interior-point methods are considered: first, a dual-feasible (but primal-infeasible) PDAS algorithm, then a version of MPC. In the former, at each iteration the normal equations (2.4) are solved once, with $\mu = 0$, and $r_c = 0$. In the latter, the normal equations are solved twice per iteration with different right-hand sides.

We assume that A is dense. For large m and $n \gg m$, the bulk of the CPU cost is consumed by the solution of the normal equations (2.4). Indeed, the number of operations (per iteration) in other computations amounts to at most a small multiple of n . As for the operations involved in solving the normal equations, the operation count is roughly as follows:

- Forming $H := AS^{-1}XA^T$: m^2n ;
- Forming $v := b - AS^{-1}(Xr_c + \mu e)$: $2mn$;
- Solving $H\Delta y = v$ (Cholesky factorization): $m^3/3$;
- Computing $\Delta s := -A^T\Delta y - r_c$: $2mn$;
- Computing $\Delta x := -x + S^{-1}(-X\Delta s + \mu e)$: $2n$.

(Because both algorithms we consider update y and s by taking a common step \hat{t} along Δy and Δs , r_c is updated at no cost: the new value is $(1 - \hat{t})$ times the old value.)

The above suggests that maximum CPU savings should be obtained by replacing, in the definition of H , matrix A by its submatrix A_Q , corresponding to a suitably chosen index set Q . The cost of forming H would then be reduced to $m^2|Q|$ operations. In this paper, we investigate the effect of making that modification only, and leaving all else unchanged, so as to least “perturb” the original algorithms.

A central issue is then the choice of Q . Given $y \in \mathbb{R}^m$ and $M \geq m$, let $\mathcal{Q}_M(y)$ be the set of all subsets of \mathbf{n} that contain the indexes of M leftmost components of $c - A^T y$. More precisely (some components of $c - A^T y$ may be equal, so “ M leftmost” may not be uniquely defined), let

$$(2.6) \quad \mathcal{Q}_M(y) := \{Q \subseteq \mathbf{n} : \exists Q' \subseteq Q \text{ s.t. } |Q'| = M \text{ and } c_i - a_i^T y \leq c_j - a_j^T y \forall i \in Q', j \notin Q'\}.$$

Consequently, the statement “ $Q \in \mathcal{Q}_M(y)$,” to be used later, means that the index set Q contains the indices of M components of the vector $c - A^T y$ that are smaller or equal to all other components of $c - A^T y$. The convergence analysis of section 3.2 guarantees that our reduced PDAS algorithm will perform appropriately (under certain assumptions involving M) as long as Q is in $\mathcal{Q}_M(y)$. Given that $n \gg m$, this leaves a lot of leeway in choosing Q . We have two competing goals. On the one hand, we want $|Q|$ to be small enough that the iterations are significantly faster than when $Q = \mathbf{n}$. On the other hand, we want to include enough well chosen constraints that the iteration count remains low. In the numerical experiments we report toward the end of this paper, we restrict ourselves to a very simple scheme: we let Q be precisely the set of indexes of M leftmost components of $c - A^T y$. Note that the “ M leftmost” rule is inexpensive to apply: it takes at most $O(n \log n)$ operations—comparisons,

²The result still holds, and the same proof applies, under the milder but less intuitive assumption “ $x_i s_i \geq 0$ for all i .” The result as stated is sufficient for our present purpose.

which are faster than additions or multiplications. (For small M , it takes even fewer comparisons.)

The following assumption will be needed in order for the proposed algorithms to be well defined.

ASSUMPTION 1. *All $m \times M$ submatrices of A have full row rank.*

LEMMA 2. *Suppose Assumption 1 holds. Let $x > 0$, $s > 0$, and $Q \subseteq \mathbf{n}$ with $|Q| \geq M$. Then $A_Q S_Q^{-1} X_Q A_Q^T$ is positive definite.*

Proof. Follows from positive definiteness of $S_Q^{-1} X_Q$ and full row rank of A_Q . \square

3. A reduced, dual-feasible PDAS algorithm.

3.1. Algorithm statement. The proposed reduced primal-dual interior-point affine scaling (rPDAS) iteration is strongly inspired from the iteration described in [TZ94], a dual-feasible primal-dual iteration based on the Newton system discussed above, with $\mu = 0$ and $r_c = 0$. In particular, the normal equations for the algorithm of [TZ94] are given by

$$(3.1a) \quad AS^{-1}XA^T\Delta y = b,$$

$$(3.1b) \quad \Delta s = -A^T\Delta y,$$

$$(3.1c) \quad \Delta x = -x - S^{-1}X\Delta s.$$

The iteration focuses on the dual variables. Note that the iteration requires the availability of an initial $y^0 \in F^\circ$.

Iteration rPDAS.

Parameters. $\beta \in (0, 1)$, $x_{\max} > 0$, $\underline{x} > 0$, integer M satisfying $m \leq M \leq n$.

Data. $y \in F^\circ$, $s := c - A^T y$, $x > 0$, with $x_i \leq x_{\max}$, $i = 1, \dots, n$, $Q \in \mathcal{Q}_M(y)$.

Step 1. Compute search direction:

$$(3.2a) \quad \text{Solve } A_Q S_Q^{-1} X_Q A_Q^T \Delta y = b,$$

$$(3.2b) \quad \text{and compute } \Delta s := -A^T \Delta y,$$

$$(3.2c) \quad \Delta x := -x - S^{-1} X \Delta s.$$

Set $\tilde{x} := x + \Delta x$ and, for $i \in \mathbf{n}$, set

$$(\tilde{x}_-)_i := \min\{\tilde{x}_i, 0\}.$$

Step 2. Updates:

(i) Compute the largest dual feasible step size

$$(3.3) \quad \bar{t} := \begin{cases} \infty & \text{if } \Delta s_i \geq 0 \quad \forall i \in \mathbf{n}, \\ \min\{-s_i/\Delta s_i : \Delta s_i < 0, i \in \mathbf{n}\} & \text{otherwise.} \end{cases}$$

Set

$$(3.4) \quad \hat{t} := \min\{\max\{\beta\bar{t}, \bar{t} - \|\Delta y\|\}, 1\}.$$

Set $y^+ := y + \hat{t}\Delta y$, $s^+ := s + \hat{t}\Delta s$.

(ii) Set

$$(3.5) \quad x_i^+ := \min\{\max\{\min\{\|\Delta y\|^2 + \|\tilde{x}_-\|^2, \underline{x}\}, \tilde{x}_i\}, x_{\max}\} \quad \forall i \in \mathbf{n}.$$

(iii) Pick $Q^+ \in \mathcal{Q}_M(y)$.

It should be noted that $(\Delta x_Q, \Delta y, \Delta s_Q)$ constructed by Iteration rPDAS also satisfies

$$(3.6a) \quad \Delta s_Q = -A_Q^T \Delta y,$$

$$(3.6b) \quad \Delta x_Q = -x_Q - S_Q^{-1} X_Q \Delta s_Q,$$

i.e., it satisfies the full set of normal equations associated with the constraint-reduced system. Equivalently, they satisfy the Newton system (with $\mu = 0$ and $r_c = 0$)

$$(3.7) \quad \begin{bmatrix} 0 & A_Q^T & I \\ A_Q & 0 & 0 \\ S_Q & 0 & X_Q \end{bmatrix} \begin{bmatrix} \Delta x_Q \\ \Delta y \\ \Delta s_Q \end{bmatrix} = \begin{bmatrix} 0 \\ b - A_Q x_Q \\ -X_Q s_Q \end{bmatrix}.$$

Remark 1. Primal update rule (3.5) is identical to the “dual” update rule used in [AT06] in the context of indefinite quadratic programming. (The “primal” problem in [AT06] can be viewed as a direct generalization of the *dual* (1.3).) As explained in [AT06], imposing the lower bound $\|\Delta y\|^2 + \|\tilde{x}_-\|^2$, which is a key to our global convergence analysis, precludes updating of x by means of a step in direction Δx ; further, the specific form of this lower bound simplifies the global convergence analysis while, together with the bound $\bar{t} - \|\Delta y\|$ in (3.4), allowing for a quadratic convergence rate. Also key to our global convergence analysis (though in our experience not needed in practice) is the upper bound x_{\max} imposed on all components of the primal variable x ; it should be stressed that global convergence of the sequence of vectors \tilde{x} to a solution is guaranteed regardless of the value of $x_{\max} > 0$. Finally, replacing in (3.5) $\min\{\|\Delta y\|^2 + \|\tilde{x}_-\|^2, \underline{x}\}$ simply with $\|\Delta y\|^2 + \|\tilde{x}_-\|^2$ would not affect the theoretical convergence properties of the algorithm. However, allowing small values of x_i^+ even when $\|\Delta y\|^2 + \|\tilde{x}_-\|^2$ is large proved beneficial in practice, especially in early iterations.

3.2. Convergence analysis. Before embarking on a convergence analysis, we introduce two more definitions. First, let $F^* \subseteq \mathbb{R}^m$ be the set of solutions of (1.3), i.e.,

$$F^* := \{y^* \in F : b^T y^* \geq b^T y \ \forall y \in F\}.$$

Of course, F^* is the set of y for which (2.1) holds with $\mu = 0$ for some $x, s \in \mathbb{R}^n$. Second, given $y \in F$, we will say that y is *stationary* for (1.3) whenever there exists $x \in \mathbb{R}^n$ such that

$$(3.8) \quad Ax = b$$

and

$$(3.9) \quad X(c - A^T y) = 0,$$

with no sign constraint imposed on x ; equivalently, with $s := c - A^T y (\geq 0)$, (2.1a)–(2.1c) hold with $\mu = 0$ for some $x \in \mathbb{R}^n$. We will refer to such x as a *multiplier vector* associated with y . Clearly, every point in F^* is stationary, but not all stationary points are in F^* : in particular, all vertices of F are stationary.

3.2.1. Global convergence. We now show that, under certain nondegeneracy assumptions, the sequence of dual iterates generated by Iteration rPDAS converges to F^* . First, on the basis of Lemma 2, it is readily verified that, under Assumption 1, Iteration rPDAS is well defined. That it can be repeated *ad infinitum* then follows from the next proposition.

PROPOSITION 3. *Suppose Assumption 1 holds. Then Iteration rPDAS generates quantities with the following properties: (i) $\Delta y \neq 0$ if and only if $b \neq 0$; (ii) $\hat{t} > 0$, $y^+ \in F^\circ$, $s^+ = c - A^T y^+ > 0$, and $x^+ > 0$.*

Proof. The first claim is a direct consequence of Lemma 2 and (3.2a); the other claims are immediate. \square

Now, let $y^0 \in F^\circ$, let $s^0 = c - A^T y^0$, let $x^0 > 0$, $Q^0 \subseteq \mathbf{n}$ with $|Q^0| \geq M$, and let $\{(x^k, y^k, s^k)\}$, $\{Q^k\}$, $\{\Delta y^k\}$, $\{\tilde{x}^k\}$, $\{\tilde{t}^k\}$, and $\{\hat{t}^k\}$ be generated by successive applications of Iteration rPDAS starting at (x^0, y^0, s^0) . Our analysis focuses on the dual sequence $\{y^k\}$.

In view of Proposition 3, $s^k = c - A^T y^k > 0$ for all k , so $y^k \in F^\circ$ for all k . We first note that, under no additional assumptions, the sequence of dual objective values is monotonic nondecreasing, strictly so if $b \neq 0$. This fact plays a central role in our global convergence analysis.

LEMMA 4. *Suppose Assumption 1 holds. If $b \neq 0$, then $b^T \Delta y^k > 0$ for all k . In particular, $\{b^T y^k\}$ is nondecreasing.*

Proof. The claim follows from (3.2a), Lemma 2, Proposition 3, and Step 2(i) of Iteration rPDAS. \square

The remainder of the global convergence analysis is carried out under two additional assumptions. The first one implies that $\{y^k\}$ is bounded.

ASSUMPTION 2. *The dual solution set F^* is nonempty and bounded.* Equivalently, the superlevel sets $\{y \in F : b^T y \geq \alpha\}$ are bounded for all α . Boundedness of $\{y^k\}$ then follows from its feasibility and monotonicity of $\{b^T y^k\}$ (Lemma 4 and Step 2(i) of Iteration rPDAS).

LEMMA 5. *Suppose Assumptions 1 and 2 hold; then $\{y^k\}$ is bounded.*

Our final nondegeneracy assumption ensures that small values of $\|\Delta y^k\|$ indicate that a stationary point of (1.3) is being approached (Lemma 6).

ASSUMPTION 3. *For all $y \in F$, $\{a_i : i \in I(y)\}$ is a linear independent set of vectors.*

LEMMA 6. *Suppose Assumptions 1 and 3 hold. Let $y^* \in \mathbb{R}^m$ and suppose that K , an infinite index set, is such that $\{y^k\}$ converges to y^* on K . If $\{\Delta y^k\}$ converges to zero on K , then y^* is stationary and $\{\tilde{x}^k\}$ converges to x^* on K , where x^* is the unique multiplier vector associated with y^* .*

Proof. Suppose $\{\Delta y^k\} \rightarrow 0$ as $k \rightarrow \infty$, $k \in K$. Without loss of generality (by going down to a further subsequence if necessary), assume that, for some Q^* , $Q_k = Q^*$ for all $k \in K$. Equation (3.7) implies that

$$(3.10) \quad A_{Q^*} \tilde{x}_{Q^*}^k - b = 0 \quad \forall k \in K,$$

and (3.2b)–(3.2c) yield

$$(3.11) \quad x_i^k a_i^T \Delta y^k - s_i^k \tilde{x}_i^k = 0 \quad \forall i, \forall k.$$

Let $s^* = c - A^T y^*$, so $s^k \rightarrow s^*$ as $k \rightarrow \infty$, $k \in K$. Since $\{x^k\}$ is bounded ($x_i^k \in [0, x_{\max}] \forall i$ by construction), it follows from (3.11) that for all $i \notin I(y^*)$ (i.e., all i for which $s_i^* > 0$), $\{\tilde{x}_i^k\} \rightarrow 0$ as $k \rightarrow \infty$, $k \in K$. Now, in view of Assumption 3 (linear

independence of the active constraints), $|I(y^*)| \leq m$ and, since $Q^k \in \mathcal{Q}_M(y^k)$ for all k , $M \geq m$, $I(y^*) \subseteq Q^*$. Hence, (3.10) yields

$$\sum_{i \in I(y^*)} \tilde{x}_i^k a_i - b \rightarrow 0 \text{ as } k \rightarrow \infty, \quad k \in K,$$

and Assumption 3 implies that, for all $i \in I(y^*)$, $\{\tilde{x}_i^k\}$ converges on K , say, to x_i^* . Taking limits in (3.10)–(3.11) then yields

$$Ax^* - b = 0,$$

$$X^* s^* = 0,$$

implying that y^* is stationary, with multiplier vector x^* . Uniqueness of x^* again follows from Assumption 3. \square

Proving that $\{y^k\}$ converges to F^* will be achieved in two main steps. The first objective is to show that $\{y^k\}$ converges to the set of *stationary points* of (1.3) (Lemma 9). This will be proved via a contradiction argument: if, for some infinite index set K , $\{y^k\}$ were to converge on K to a *nonsolution* point—for instance, to a nonstationary point—then $\{\Delta y^k\}$ would have to go to zero on K (Lemma 8), in contradiction with Lemma 6. The heart of the argument lies in the following lemma.

LEMMA 7. *Suppose Assumptions 1, 2, and 3 hold. Let K be an infinite index set such that*

$$\inf\{\|\Delta y^{k-1}\|^2 + \|\tilde{x}_-^{k-1}\|^2 : k \in K\} > 0.$$

Then $\{\Delta y^k\} \rightarrow 0$ as $k \rightarrow \infty$, $k \in K$.

Proof. In view of (3.5), for all $i \in \mathbf{n}$, x_i^k is bounded away from zero on K . Proceeding by contradiction, assume that, for some infinite index set $K' \subseteq K$, $\inf_{k \in K'} \|\Delta y^k\| > 0$. Since $\{y^k\}$ (see Lemma 5) and $\{x^k\}$ (see (3.5)) are bounded, we may assume, without loss of generality, that for some y^* and x^* , with $x_i^* > 0$ for all i , and some Q^* with $|Q^*| \geq M$,

$$\{y^k\} \rightarrow y^* \text{ as } k \rightarrow \infty, \quad k \in K',$$

$$\{x^k\} \rightarrow x^* \text{ as } k \rightarrow \infty, \quad k \in K',$$

and

$$Q^k = Q^* \quad \forall k \in K'.$$

Let $s^* := c - A^T y^*$; since $s^k = c - A^T y^k$ for all k , it follows that $\{s^k\} \rightarrow s^*$, $k \in K'$. Since in view of Lemma 1, of Assumptions 1 and 3, and of the fact that $x_i^* > 0$ for all i , the matrix $J(A_{Q^*}, x_{Q^*}^*, s_{Q^*}^*)$ is nonsingular, it follows from (3.7) that, for some v^* and \tilde{x}_i^* , $i \in Q^*$, with $v^* \neq 0$ (since $\inf_{k \in K'} \|\Delta y^k\| > 0$),

$$\{\Delta y^k\} \rightarrow v^* \text{ as } k \rightarrow \infty, \quad k \in K',$$

$$\{x_i^k\} \rightarrow \tilde{x}_i^* \text{ as } k \rightarrow \infty, \quad k \in K', \quad i \in Q^*.$$

In view of linear independence Assumption 3, since $\{s^k\} \rightarrow s^* = c - A^T y^*$, and since, by definition of \mathcal{Q}_M , $I(y^*) \subseteq Q^*$, it follows that s_i^k is bounded away from zero when $i \notin Q^*$, $k \in K'$. It then follows from (3.2b) and (3.2c) that, for some \tilde{x}^* ,

$$(3.12) \quad \{\tilde{x}^k\} \rightarrow \tilde{x}^* \text{ as } k \rightarrow \infty, \quad k \in K'.$$

Now, Step 2(i) of Iteration rPDAS and (3.2c) yield

$$\bar{t}^k = -\frac{s_{i_k}^k}{\Delta s_{i_k}^k} = \frac{x_{i_k}^k}{\tilde{x}_{i_k}^k} \quad \text{for some } i_k,$$

for all $k \in K'$ such that $\bar{t}^k < \infty$. Since the components of $\{x^k\}$ are bounded away from zero on K' (since $x_i^* > 0$ for all i), it follows from (3.12) that \bar{t}^k is bounded away from zero on K' , and from Step 2(i) in Iteration rPDAS that the same holds for \hat{t}^k . Thus, for some $\underline{t} > 0$, $\hat{t}^k \geq \underline{t}$ for all $k \in K'$. Also, (3.2b)–(3.2c) yield, for all k ,

$$(3.13) \quad \tilde{x}^k = (S^k)^{-1} X^k A^T \Delta y^k$$

which together with (3.2)(a) yields

$$(3.14) \quad b^T \Delta y^k = (\Delta y^k)^T A_{Q^k} X_{Q^k}^k (S_{Q^k}^k)^{-1} A_{Q^k}^T \Delta y^k = (\tilde{x}_{Q^k}^k)^T A_{Q^k}^T \Delta y^k \quad \forall k.$$

In view of Lemma 4, it follows that

$$(3.15) \quad b^T y^{k+1} = b^T (y^k + \hat{t}^k \Delta y^k) \geq b^T y^k + \underline{t} b^T \Delta y^k = b^T y^k + \underline{t} (A_{Q^k} \tilde{x}_{Q^k}^k)^T \Delta y^k \quad \forall k \in K'.$$

Now, since $v^* \neq 0$ and $|Q^*| \geq M$, it follows from Assumption 1 that $A_{Q^*}^T v^* \neq 0$. Further, taking limits in (3.13) as $k \rightarrow \infty$, $k \in K'$, we get

$$X_{Q^*}^* A_{Q^*}^T v^* - S_{Q^*}^* \tilde{x}_{Q^*}^* = 0.$$

Positivity of x_i^* and nonnegativity of s_i^* for all i then imply that $(\tilde{x}_{Q^*}^*)_i$ and $(A_{Q^*}^T v^*)_i$ have the same sign whenever the latter is nonzero, in which case the former is nonzero as well. It follows that $(\tilde{x}_{Q^*}^*)^T A_{Q^*}^T v^* > 0$. Thus there exists $\delta > 0$ such that $(A_{Q^*} \tilde{x}_{Q^*}^k)^T (\Delta y^k) > \delta$ for k large enough, $k \in K'$. Since, in view of Lemma 4, $\{b^T y^k\}$ is monotonic nondecreasing, it follows from (3.15) that $b^T y^k \rightarrow \infty$ as $k \rightarrow \infty$, a contradiction since y^k is bounded. \square

LEMMA 8. *Suppose Assumptions 1, 2, and 3 hold. Suppose there exists an infinite index set K such that $\{y^k\}$ is bounded away from F^* on K . Then $\{\Delta y^k\}$ goes to zero on K .*

Proof. Let us again proceed by contradiction, i.e., suppose $\{\Delta y^k\}$ does not converge to zero as $k \rightarrow \infty$, $k \in K$. In view of Lemma 7, there exists an infinite index set $K' \subseteq K$ such that

$$(3.16) \quad \tilde{x}_-^{k-1} \rightarrow 0 \text{ as } k \rightarrow \infty, \quad k \in K'$$

and

$$(3.17) \quad \Delta y^{k-1} \rightarrow 0 \text{ as } k \rightarrow \infty, \quad k \in K'.$$

Further, since $\{y^k\}$ is bounded, and bounded away from F^* , there is no loss of generality in assuming that, for some $y^* \notin F^*$, $\{y^k\} \rightarrow y^*$ as $k \rightarrow \infty$, $k \in K'$.

Since $\|y^k - y^{k-1}\| = \|\hat{t}^{k-1}\Delta y^{k-1}\| \leq \|\Delta y^{k-1}\|$, it follows that $\{y^{k-1}\} \rightarrow y^*$ as $k \rightarrow \infty$, $k \in K'$ which implies, in view of (3.17) and of Lemma 6, that y^* is stationary and $\{\tilde{x}^{k-1}\} \rightarrow x^*$ as $k \rightarrow \infty$, $k \in K'$, where x^* is the corresponding multiplier vector. From (3.16) it follows that $x^* \geq 0$, thus that $y^* \in F^*$, a contradiction. \square

LEMMA 9. *Suppose Assumptions 1, 2, and 3 hold. Then $\{y^k\}$ converges to the set of stationary points of (1.3).*

Proof. Suppose the claim does not hold. Because $\{y^k\}$ is bounded, there exist an infinite index set K and some nonstationary y^* such that $y^k \rightarrow y^*$ as $k \rightarrow \infty$, $k \in K$. By Lemma 6, $\{\Delta y^k\}$ does not converge to zero on K . This contradicts Lemma 8. \square

We are now ready to embark on the final step in the global convergence analysis: prove convergence of $\{y^k\}$ to the *solution set* for (1.3). The key to this result is Lemma 11, which establishes that the multiplier vectors associated with all limit points of $\{y^k\}$ are the same. Thus, let

$$L := \{y \in \mathbb{R}^m : y \text{ is a limit point of } \{y^k\}\}.$$

(In view of Lemma 9, every $y \in L$ is a stationary point of (P) .) The set L is bounded (since $\{y^k\}$ is bounded) and, as a limit set, it is closed, and thus compact. We first prove an auxiliary lemma.

LEMMA 10. *Suppose Assumptions 1, 2, and 3 hold. If $\{y^k\}$ is bounded away from F^* , then L is connected.*

Proof. Suppose the claim does not hold. Since L is compact, there must exist compact sets $E_1, E_2 \subset \mathbb{R}^n$, both nonempty, such that $L = E_1 \cup E_2$ and $E_1 \cap E_2 = \emptyset$. Thus $\delta := \min_{y \in E_1, y' \in E_2} \|y - y'\| > 0$. A simple contradiction argument based on the fact that $\{y^k\}$ is bounded shows that, for k large enough, $\min_{y \in L} \|y^k - y\| \leq \delta/3$, i.e., either $\min_{y \in E_1} \|y^k - y\| \leq \delta/3$ or $\min_{y \in E_2} \|y^k - y\| \leq \delta/3$. Moreover, since both E_1 and E_2 are nonempty (i.e., contain limit points of $\{y^k\}$), each of these situations occurs infinitely many times. Thus $K := \{k : \min_{y \in E_1} \|y^k - y\| \leq \delta/3, \min_{y \in E_2} \|y^{k+1} - y\| \leq \delta/3\}$ is an infinite index set and $\|\Delta y^k\| \geq \delta/3 > 0$ for all $k \in K$. In view of Lemma 8, this is a contradiction. \square

LEMMA 11. *Suppose Assumptions 1, 2, and 3 hold. Suppose $\{y^k\}$ is bounded away from F^* . Let $y, y' \in L$. Let x and x' be the associated multiplier vectors. Then $x = x'$.*

Proof. Given any $y \in L$, let $x(y)$ be the multiplier vector associated with y , let $s(y) = c - A^T y$, and let $J(y)$ be the index set of “binding” constraints at y , i.e.,

$$J(y) = \{i \in \mathbf{n} : x_i(y) \neq 0\}.$$

We first show that, if $y, y' \in L$ are such that $J(y) = J(y')$, then $x(y) = x(y')$. Indeed, from (2.1b),

$$\sum_{j \in J(y)} x_j(y) a_j = b = \sum_{j \in J(y')} x_j(y') a_j,$$

and the claim follows from linear independence Assumption 3. To conclude the proof, we show that, for any $y, y' \in L$, $J(y) = J(y')$. Let $\tilde{y} \in L$ be arbitrary and let $E_1 := \{y \in L : J(y) = J(\tilde{y})\}$ and $E_2 := \{y \in L : J(y) \neq J(\tilde{y})\}$. We show that both E_1 and E_2 are closed. Let $\{\xi^\ell\} \subseteq L$ be a convergent sequence, say to $\hat{\xi}$, such that $J(\xi^\ell) = J$ for all ℓ , for some J . It follows from the first part of this proof that $x(\xi^\ell) = x$ for all ℓ for some x . Now, for all ℓ , $s_j(\xi^\ell) = 0$ for all j such that $x_j \neq 0$, so

that $s_j(\hat{\xi}) = 0$ for all j such that $x_j \neq 0$. Thus $J \subseteq I(\hat{\xi})$, and from linear independence Assumption 3 it follows that $x(\hat{\xi}) = x$ and thus $J(\hat{\xi}) = J$. Also, since L is closed, $\hat{\xi} \in L$. Thus, if $\{\xi^\ell\} \subseteq E_1$, then $\hat{\xi} \in E_1$ and, if $\{\xi^\ell\} \subseteq E_2$, then $\hat{\xi} \in E_2$, proving that both E_1 and E_2 are closed. Since E_1 is nonempty (it contains \tilde{y}), connectedness of L (Lemma 10) implies that E_2 is empty. Thus $J(y) = J(\tilde{y})$ for all $y \in L$, and the proof is complete. \square

With all the tools in hand, we present the final theorem of this section. The essence of its proof is that if $\{y_k\}$ does not converge to F^* , complementary slackness will not be satisfied.

THEOREM 12. *Suppose Assumptions 1, 2, and 3 hold. Then $\{y^k\}$ converges to F^* .*

Proof. Proceeding again by contradiction, suppose that some limit point of $\{y^k\}$ is not in F^* and thus, since $y^k \in F$ for all k and since, in view of the monotonicity of $\{b^T y^k\}$ (Lemma 4), $b^T y^k$ takes on the same value at all limit points of $\{y^k\}$, that $\{y^k\}$ is bounded away from F^* . In view of Lemma 8, $\{\Delta y^k\} \rightarrow 0$. Let x^* be the common multiplier vector associated with all limit points of $\{y^k\}$ (see Lemma 11). A simple contradiction argument shows that Lemma 6 then implies that $\{\tilde{x}^k\} \rightarrow x^*$. Since $\{y^k\}$ is bounded away from F^* , $x^* \not\geq 0$. Let i_0 be such that $x_{i_0}^* < 0$. Then $\tilde{x}_{i_0}^k < 0$ for all k large enough. The definition of \tilde{x}^k in Step 1 of Iteration rPDAS, together with (3.2c), then implies that $\Delta s_{i_0}^k > 0$ for k large enough, and it then follows from the update rule for s^k in Step 2(i) that, for k large enough,

$$0 < s_{i_0}^k < s_{i_0}^{k+1} < \dots$$

On the other hand, since $x_{i_0}^* < 0$, complementary slackness (3.9) implies that $(c - A^T \hat{y})_{i_0} = 0$ for all limit points \hat{y} of $\{y^k\}$ and thus, since $\{y^k\}$ is bounded, $\{s_{i_0}^k\} \rightarrow 0$. This is a contradiction. \square

3.2.2. Local rate of convergence. We prove q-quadratic convergence of the pair (x^k, y^k) (when x_{\max} is large enough) under one additional assumption, which supersedes Assumption 2. (A sequence $\{z^k\}$ is said to converge q-quadratically to z^* if it converges to z^* and there exists a constant θ such that $\|z^{k+1} - z^*\| \leq \theta \|z^k - z^*\|^2$ for all k large enough.)

ASSUMPTION 4. *The dual solution set F^* is a singleton.*

Let y^* denote the unique solution to (1.3), i.e., $F^* = \{y^*\}$, let $s^* := c - A^T y^*$, and let x^* be the corresponding multiplier vector (unique in view of Assumption 3). Of course, under Assumptions 1, 3, and 4, it follows from Theorem 12 that $\{y^k\} \rightarrow y^*$ as $k \rightarrow \infty$. Further, under Assumption 3, Assumption 4 implies that strict complementarity holds, i.e.,

$$(3.18) \quad x_i^* > 0 \quad \forall i \in I(y^*).$$

Moreover, Assumption 4 implies that

$$(3.19) \quad \text{span}(\{a_i : i \in I(y^*)\}) = \mathbb{R}^m.$$

LEMMA 13. *Suppose Assumptions 1, 3, and 4 hold and let $Q \supseteq I(y^*)$ and $Q^c = \mathbf{n} \setminus Q$. Then $J_a(A_Q, x_Q^*, s_Q^*)$ and $J(A_Q, x_Q^*, s_Q^*)$ are nonsingular and $A_{Q^c}^T y^* < c_{Q^c}$.*

Proof. The last claim is immediate. Since $s^* = c - A^T y^*$, it follows from linear independence Assumption 3 that $\{a_i : s_i = 0\}$ is linear independent, hence its subset

retaining only the columns whose indices are in Q is also linear independent. The first two claims now follow directly from Lemma 1. \square

The following preliminary result is inspired from [PTH88, Proposition 4.2].

LEMMA 14. *Suppose Assumptions 1, 3, and 4 hold. Then (i) $\{\Delta y^k\} \rightarrow 0$; (ii) $\{\tilde{x}^k\} \rightarrow x^*$; and (iii) if $x_i^* \leq x_{\max}$ for all $i \in \mathbf{n}$, then $\{x^k\} \rightarrow x^*$.*

Proof. To prove Claim (i), proceed by contradiction. Specifically, suppose that, for some infinite index set K , $\inf_{k \in K} \|\Delta y^k\| > 0$. Without loss of generality, assume that, for some Q^* , $Q^k = Q^*$ for all $k \in K$. Since $Q^* \in \mathcal{Q}_M(y^k)$ for all $k \in K$, since $\{y^k\} \rightarrow y^*$ as $k \rightarrow \infty$, and since, in view of Assumption 3, $|I(y^*)| \leq m \leq M$, it must hold that $Q^* \supseteq I(y^*)$. On the other hand, Lemma 7 implies that there exists an infinite index set $K' \subseteq K$ such that $\{\Delta y^{k-1}\}_{k \in K'}$ and $\{\tilde{x}^{k-1}\}_{k \in K'}$ go to zero. In view of Lemma 6 it follows that $\{\tilde{x}^{k-1}\}_{k \in K'} \rightarrow x^*$. It then follows from (3.5) that, for all i , $\{x_i^k\}_{k \in K'} \rightarrow \xi_i^* := \min\{x_i^*, x_{\max}\}$. Since $Q^* \supseteq I(y^*)$, it follows from Lemma 1, Assumption 3, (3.18), and (3.19) that $J(A_{Q^*}, \xi_{Q^*}^*, s_{Q^*}^*)$ is nonsingular. Now note that, in view of (3.7), it holds that (see (2.5))

$$(3.20) \quad J(A_{Q^k}, x_{Q^k}^k, s_{Q^k}^k) \begin{bmatrix} \tilde{x}_{Q^k}^k \\ \Delta y^k \\ \Delta s_{Q^k}^k \end{bmatrix} = \begin{bmatrix} 0 \\ b \\ 0 \end{bmatrix} \quad \forall k \in K'.$$

On the other hand, by feasibility of x^* and complementarity slackness,

$$(3.21) \quad J(A_{Q^*}, \xi_{Q^*}^*, s_{Q^*}^*) \begin{bmatrix} x_{Q^*}^* \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ b \\ 0 \end{bmatrix}.$$

From (3.20) and (3.21), nonsingularity of $J(A_{Q^*}, \xi_{Q^*}^*, s_{Q^*}^*)$ thus implies that $\{\Delta y^k\} \rightarrow 0$ as $k \rightarrow \infty$, $k \in K'$, a contradiction since $K' \subseteq K$. Claim (i) is thus proved. Claim (ii) then directly follows from Lemma 6 and Claim (iii) follows from (3.5). \square

To prove q-quadratic convergence of $\{(y^k, x^k)\}$, the following property of Newton's method will be used. It is borrowed from [TZ94, Proposition 3.10].

PROPOSITION 15. *Let $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be twice continuously differentiable and let $\hat{z} \in \mathbb{R}^n$ be such that $\Phi(\hat{z}) = 0$ and $\frac{\partial \Phi}{\partial z}(\hat{z})$ is nonsingular. Let $\rho > 0$ be such that $\frac{\partial \Phi}{\partial z}(z)$ is nonsingular whenever $z \in B(\hat{z}, \rho) := \{z : \|z - \hat{z}\| \leq \rho\}$. Let $d^N : B(\hat{z}, \rho) \rightarrow \mathbb{R}^n$ be the Newton increment $d^N(z) := -(\frac{\partial \Phi}{\partial z}(z))^{-1} \Phi(z)$. Then given any $c_1 > 0$ there exists $c_2 > 0$ such that the following statement holds:*

For all $z \in B(\hat{z}, \rho)$ and $z^+ \in \mathcal{R}^n$ such that, for each $i \in \{1, \dots, n\}$, either

$$(i) \quad |z_i^+ - \hat{z}_i| \leq c_1 \|d^N(z)\|^2$$

or

$$(ii) \quad |z_i^+ - (z_i + d_i^N(z))| \leq c_1 \|d^N(z)\|^2,$$

it holds that

$$(3.22) \quad \|z^+ - \hat{z}\| \leq c_2 \|z - \hat{z}\|^2.$$

We will apply this proposition to the equality portion of the KKT conditions (2.1) (with $\mu = 0$). Eliminating s from this system of equations yields $\Phi(x, y) = 0$, with Φ given by

$$(3.23) \quad \Phi(x, y) := \begin{bmatrix} Ax - b \\ X(c - A^T y) \end{bmatrix}.$$

It is readily verified that (2.3a), with $\mu = 0$, $r_c = 0$, and s replaced by $c - A^T y$, is the Newton iteration for the solution of $\Phi(x, y) = 0$. In particular, $J_a(A, x, c - A^T y)$ is the Jacobian of $\Phi(x, y)$.

Of course, Iteration rPDAS does not make use of the Newton direction for Φ , since it is based on a reduced set of constraints. Lemma 16 below relates the direction computed in Step 1 of Iteration rPDAS to the Newton direction.

In what follows, we use z (possibly with subscripts, superscripts, or diacritical signs) to denote (x, y) (with the same subscripts, superscripts, or diacritical signs on both x and y). Also, let

$$G^o := \{z : x > 0, y \in F^o\}$$

and, given $z \in G^o$ and $Q \in \mathcal{Q}_M(y)$, let $\Delta x(z, Q)$, $\Delta y(z, Q)$, $x^+(z, Q)$, $y^+(z, Q)$, $\tilde{x}(z, Q)$, $\bar{t}(z, Q)$, and $\hat{t}(z, Q)$ denote the quantities defined by Iteration rPDAS, and let $\Delta z(z, Q) := (\Delta x(z, Q), \Delta y(z, Q))$ and $z^+(z, Q) := (x^+(z, Q), y^+(z, Q))$. Further, let $Q^c := \mathbf{n} \setminus Q$, let $d^N(z) := \Delta z(z, \mathbf{n})$ denote the Newton increment for Φ , and, given $\rho > 0$, let $B(z^*, \rho) := \{z : \|z - z^*\| \leq \rho\}$. The following lemma was inspired from an idea of O'Leary [O'L04]. (Existence of $\rho > 0$ follows from Lemma 13.)

LEMMA 16. *Suppose Assumptions 1, 3, and 4 hold. Let $\rho > 0$ be such that, for all $(x, y) \in B(z^*, \rho) \cap G^o$ and for all $Q \in \mathcal{Q}_M(y)$, $J_a(A_Q, x_Q, c_Q - A_Q^T y)$ is nonsingular and $A_{Q^c}^T y < c_{Q^c}$. Then there exists $\gamma > 0$ such that, for all $z \in B(z^*, \rho) \cap G^o$, $Q \in \mathcal{Q}_M(y)$,*

$$\|\Delta z(z, Q) - d^N(z)\| \leq \gamma \|z - z^*\| \cdot \|d^N(z)\|.$$

Proof. Let $z \in B(z^*, \rho) \cap G^o$, $Q \in \mathcal{Q}_M(y)$ and let $s := c - A^T y$. Then, $\Delta y(z, Q)$ and $\Delta x_Q(z, Q)$ satisfy (direct consequence of (3.7))

$$(3.24) \quad \begin{bmatrix} 0 & A_Q \\ X_Q A_Q^T & -S_Q \end{bmatrix} \begin{bmatrix} \Delta y(z, Q) \\ \Delta x_Q(z, Q) \end{bmatrix} = \begin{bmatrix} b - A_Q x_Q \\ X_Q s_Q \end{bmatrix}.$$

On the other hand, $\Delta y(z, \mathbf{n})$ and $\Delta x(z, \mathbf{n})$ satisfy (see (2.3a), with $\mu = 0$ and $r_c = 0$)

$$\begin{bmatrix} 0 & A \\ X A^T & -S \end{bmatrix} \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x(z, \mathbf{n}) \end{bmatrix} = \begin{bmatrix} b - Ax \\ Xs \end{bmatrix},$$

and eliminating $\Delta x_{Q^c}(z, \mathbf{n})$ in the latter yields

$$(3.25) \quad \begin{bmatrix} A_{Q^c} S_{Q^c}^{-1} X_{Q^c} A_{Q^c}^T & A_Q \\ X_Q A_Q^T & -S_Q \end{bmatrix} \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix} = \begin{bmatrix} b - A_Q x_Q \\ X_Q s_Q \end{bmatrix}.$$

Equating the left-hand sides of (3.24) and (3.25) yields

$$(3.26) \quad \begin{bmatrix} \Delta y(z, Q) \\ \Delta x_Q(z, Q) \end{bmatrix} = J_a(A_Q, x_Q, c_Q - A_Q^T y)^{-1} \begin{bmatrix} A_{Q^c} S_{Q^c}^{-1} X_{Q^c} A_{Q^c}^T & A_Q \\ X_Q A_Q^T & -S_Q \end{bmatrix} \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix}.$$

Expressing $\begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix}$ as $J_a(A_Q, x_Q, c_Q - A_Q^T y)^{-1} J_a(A_Q, x_Q, c_Q - A_Q^T y) \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix}$ and subtracting from (3.26) then yields

$$\begin{bmatrix} \Delta y(z, Q) \\ \Delta x_Q(z, Q) \end{bmatrix} - \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix}$$

$$= J_a(A_Q, x_Q, c_Q - A_Q^T y)^{-1} \begin{bmatrix} A_{Q^c} S_{Q^c}^{-1} X_{Q^c} A_{Q^c}^T & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta y(z, \mathbf{n}) \\ \Delta x_Q(z, \mathbf{n}) \end{bmatrix}.$$

Further, it follows from (3.1b) and (3.1c) and from (3.2b) and (3.2c) that

$$\Delta x_{Q^c}(z, Q) - \Delta x_{Q^c}(z, \mathbf{n}) = S_{Q^c}^{-1} X_{Q^c} A_{Q^c}^T (\Delta y(z, Q) - \Delta y(z, \mathbf{n})).$$

Since S_{Q^c} and $J_a(A_Q, x_Q, c_Q - A_Q^T y)$ are continuous and nonsingular over the closed ball $B(z^*, \rho)$, and since $\|X_{Q^c}\| = \|X_{Q^c} - X_{Q^c}^*\| \leq \|z - z^*\|$ (since $c_{Q^c} - A_{Q^c}^T z^* > 0$, i.e., $I(y^*) \cap Q^c$ is empty), in view of the fact that $\{Q^c : Q \in \mathcal{Q}_M(y)\}$ is finite (since $\mathcal{Q}_M(y)$ is) the claim follows. \square

We are now ready to prove q-quadratic convergence.

THEOREM 17. *Suppose Assumptions 1, 3, and 4 hold. If $x_i^* < x_{\max}$ for all $i \in \mathbf{n}$, then $\{(x^k, y^k)\}$ converges to (x^*, y^*) q-quadratically.*

Proof. We aim at establishing that the conditions in Proposition 15 hold for Φ given by (3.23) and with $\hat{z} := z^* (= (x^*, y^*))$, $z^+ := z^+(z, Q)$, $Q \in \mathcal{Q}_M(y)$, and $\rho > 0$ small enough. First, note that

$$\frac{\partial \Phi}{\partial z}(z) = J_a(A, x, c - A^T y),$$

so, in view of (3.18), (3.19), and linear independence Assumption 3, it follows from Lemma 1 that $\frac{\partial \Phi}{\partial z}(z^*)$ is nonsingular. Next, let $i \in I(y^*)$ and consider Step 2(ii) in Iteration rPDAS. Let $Q \supseteq I(y^*)$. From Lemma 13, we know that $J(A_Q, x_Q^*, s_Q^*)$ is nonsingular. Since, by feasibility of x^* and complementarity slackness,

$$J(A_Q, x_Q^*, s_Q^*) \begin{bmatrix} x_Q^* \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ b \\ 0 \end{bmatrix},$$

it follows from (3.7), (3.2), and the fact that $s_j := c_j - a_j^T y$ is bounded away from zero in a neighborhood of z^* for $j \notin Q$ (Lemma 13), that

$$(3.27) \quad \Delta y(z, Q) \rightarrow 0 \text{ as } z \rightarrow z^*$$

and

$$(3.28) \quad \tilde{x}(z, Q) \rightarrow x^* \text{ as } z \rightarrow z^*.$$

Note that, for $z \in G^o$ close enough to z^* , $Q \supseteq I(y^*)$ for all $Q \in \mathcal{Q}_M(y)$. Since $x_i^* > 0$ (from (3.18)) and $x^* \geq 0$, it follows that for $z \in G^o$ close enough to z^* ,

$$\|\Delta y(z, Q)\|^2 + \|\tilde{x}_-(z, Q)\|^2 < \tilde{x}_i(z, Q) \quad \forall Q \in \mathcal{Q}_M(y),$$

which, in view of the update rule for x in Step 2(ii) of Iteration rPDAS, since $x_i^* < x_{\max}$, implies that, for $z \in G^o$ close enough to z^* ,

$$x_i^+(z, Q) = \tilde{x}_i(z, Q) = x_i + \Delta x_i(z, Q) \quad \forall Q \in \mathcal{Q}_M(y),$$

yielding, for $z \in G^o$ close enough to z^* ,

$$(3.29) \quad x_i^+(z, Q) - (x_i + \Delta x_i(z, \mathbf{n})) = \Delta x_i(z, Q) - \Delta x_i(z, \mathbf{n}) \quad \forall Q \in \mathcal{Q}_M(y).$$

In view of Lemma 16, it follows that, for z close enough to z^* , $z \in G^o$,

$$(3.30) \quad |x_i^+(z, Q) - (x_i + \Delta x_i(z, \mathbf{n}))| \leq \gamma \|z - z^*\| \cdot \|d^N(z)\| \quad \forall Q \in \mathcal{Q}_M(y).$$

Next, let $i \notin I(y^*)$, so that $x_i^* = 0$, and again consider Step 2(ii) in Iteration rPDAS. Then for every $z \in G^o$ close enough to z^* , and every $Q \in \mathcal{Q}_M(y)$, either again

$$x_i^+(z, Q) = x_i + \Delta x_i(z, Q),$$

yielding again (3.29) and (3.30), or

$$x_i^+(z, Q) = \|\Delta y(z, Q)\|^2 + \|\tilde{x}_-(z, Q)\|^2, \leq \|\Delta z(z, Q)\|^2$$

yielding, since $x_i^* = 0$,

(3.31)

$$\|x_i^+(z, Q) - x_i^*\| \leq \|\Delta z(z, Q) - d^N(z) + d^N(z)\|^2 \leq (\gamma \|z - z^*\| \cdot \|d^N(z)\| + \|d^N(z)\|)^2$$

for every $z \in G^o$ close enough to z^* , $Q \in \mathcal{Q}_M(y)$. Finally, consider the “ y ” components of z . From (3.2b)–(3.2c) we know that, for $z \in G^o$ close enough to z^* , for all $Q \in \mathcal{Q}_M(y)$, and for all i such that $\Delta s_i(z, Q) := -a_i^T \Delta y(z, Q) \neq 0$,

$$\frac{s_i}{a_i^T \Delta y(z, Q)} = -\frac{s_i}{\Delta s_i(z, Q)} = \frac{x_i}{\tilde{x}_i(z, Q)}.$$

In view of Lemma 14(i), we conclude that, for $i \notin I(y^*)$,

$$\frac{|x_i|}{|\tilde{x}_i(z, Q)|} \rightarrow \infty \quad \text{as } z \rightarrow z^*, Q \in \mathcal{Q}_M(y).$$

Step 2(i) in Iteration rPDAS then yields

$$\bar{t}(z, Q) = \min \left\{ \frac{x_i}{\tilde{x}_i(z, Q)} : i \in I(y^*) \right\}$$

for $z \in G^o$ close enough to z^* , $Q \in \mathcal{Q}_M(y)$. Step 2(i) in Iteration rPDAS further yields, for $z \in G^o$ close enough to z^* (using (3.27)), $Q \in \mathcal{Q}_M(y)$,

$$\hat{t}(z, Q) = \min \left\{ 1, \frac{x_{i(z, Q)}}{\tilde{x}_{i(z, Q)}(z, Q)} - \|\Delta y(z, Q)\| \right\},$$

for some $i(z, Q) \in I(y^*)$. (Nonemptiness of $I(y^*)$ is insured by Assumption 4.) Thus, for $z \in G^o$ close enough to z^* , $Q \in \mathcal{Q}_M(y)$, and some $i(z, Q) \in I(y^*)$,

$$\begin{aligned} \|y^+(z, Q) - (y + \Delta y(z, Q))\| &= |\hat{t}(z, Q) - 1| \|\Delta y(z, Q)\| \\ &\leq \left| \|\Delta y(z, Q)\| + \frac{\tilde{x}_{i(z, Q)}(z, Q) - x_{i(z, Q)}}{\tilde{x}_{i(z, Q)}(z, Q)} \right| \|\Delta y(z, Q)\|. \end{aligned}$$

Since $x_i^* > 0$ for all $i \in I(y^*)$, it follows that for some $c_0 > 0$ and $z \in G^o$ close enough to z^* ,

$$\begin{aligned} \|y^+(z, Q) - (y + \Delta y(z, Q))\| &\leq (\|\Delta y(z, Q)\| + c_0 \|\Delta x(z, Q)\|) \|\Delta y(z, Q)\| \\ &\leq (1 + c_0) \|\Delta z(z, Q)\|^2 \\ &\leq (1 + c_0) (\|\Delta z(z, Q) - d^N(z)\| + \|d^N(z)\|)^2 \\ &\leq (1 + c_0) (\gamma \|z - z^*\| \cdot \|d^N(z)\| + \|d^N(z)\|)^2 \end{aligned}$$

for all $Q \in \mathcal{Q}_M(y)$. It follows from Lemma 16 that

$$\begin{aligned} \|y^+(z, Q) - (y + \Delta y(z, \mathbf{n}))\| &\leq (1 + c_0)(\gamma \|z - z^*\| \cdot \|d^N(z)\| + \|d^N(z)\|)^2 \\ &\quad + \gamma \|z - z^*\| \cdot \|d^N(z)\| \\ (3.32) \qquad \qquad \qquad &\leq c_1 \max\{\|d^N(z)\|^2, \|z - z^*\|^2\} \end{aligned}$$

for some $c_1 > 0$ independent of z for all $z \in G^o$ close enough to z^* , $Q \in \mathcal{Q}_M(y)$. Equations (3.30), (3.31), and (3.32) are the key to the completion of the proof.

For $z \in G^o$ (close enough to z^*) such that $\|z - z^*\| \leq \|d^N(z)\|$, in view of Proposition 15, (3.30), (3.31), and (3.32) imply that, for some $c_2 > 0$ independent of z , and for all $Q \in \mathcal{Q}_M(y)$,

$$\|z^+(z, Q) - z^*\| \leq c_2 \|z - z^*\|^2.$$

On the other hand, for $z \in G^o$ close enough to z^* such that $\|d^N(z)\| < \|z - z^*\|$, (3.30) yields, for some $c_3 > 0$ independent of z and for all $Q \in \mathcal{Q}_M(y)$,

$$|x_i^+(z, Q) - x_i^*| \leq \gamma \|z - z^*\| \cdot \|d^N(z)\| + \|x_i + \Delta x_i(z, \mathbf{n}) - x_i^*\| \leq c_3 \|z - z^*\|^2$$

for all $i \in I(y^*)$, where we have invoked quadratic convergence of the Newton iteration; (3.31) yields, for some $c_4 > 0$ independent of z ,

$$|x_i^+(z, Q) - x_i^*| \leq c_4 \|z - z^*\|^2$$

for all $i \notin I(y^*)$; and (3.32) yields, for some $c_5 > 0$ independent of z ,

$$\|y^+(z, Q) - y^*\| \leq c_1 \|z - z^*\|^2 + \|y_i + \Delta y(z, \mathbf{n}) - y^*\| \leq c_5 \|z - z^*\|^2.$$

In particular, they together imply again that, for all $z \in G^o$ close enough to z^* , $Q \in \mathcal{Q}_M(y)$,

$$\|z^+(z, Q) - z^*\| \leq c_2 \|z - z^*\|^2$$

for some $c_2 > 0$ independent of z . Since, in view of Lemma 14, z^k converges to z^* , it follows that z^k converges to z^* q-quadratically. \square

3.3. Numerical results. Algorithm rPDAS was implemented in MATLAB and run on an Intel(R) Pentium(R) III CPU 733MHz machine with 256 KB cache, 512 MB RAM, Linux kernel 2.6.11, and MATLAB 7 (R14).³

Parameters were chosen as $\beta := 0.99$, $x_{\max} := 10^{15}$, $\underline{x} := 10^{-4}$. The code was supplied with strictly feasible initial dual points (i.e., $y^0 \in F^o$). The initial primal vector, x^0 , was chosen using the heuristic in [Meh92, p. 589] modified to accommodate dual feasibility. Specifically, $x^0 := \hat{x}^0 + \hat{\delta}_x$, where \hat{x}^0 is the minimum norm solution of $Ax = b$, $\hat{\delta}_x := \delta_x + (\hat{x}^0 + \delta_x e)^T s^0 / (2 \sum_{i=1}^n s_i^0)$, $\delta_x := \max\{-1.5 \cdot \min(x), 0\}$, where $s^0 := x - A^T y^0$ and e is the vector of all ones. The “ M most active” heuristic (Q consists of the indexes of the M leftmost components of $c - A^T y$) was used to select the index set Q . The code uses MATLAB’s “Cholesky-Infinity” factorization (`cholinc` function) to solve the normal equations (3.2a). A safeguard $s_i := \max\{10^{-14}, s_i\}$ was

³The code is available from the authors.

applied before Step 1; this prevents the matrix $A_Q S_Q^{-1} X_Q A_Q^T$ in (3.2a) from being excessively ill-conditioned and avoids inaccuracies in the ratios $s_i/\Delta s_i$ involved in (3.3) that could lead to unnecessarily small steps. We used a stopping criterion, adapted from [Meh92, p. 592], based on the error in the primal-dual equalities (2.1b)–(2.1a) and the duality gap. Specifically, convergence was declared when

$$\frac{\|b - Ax\|}{1 + \|x\|} + \frac{\|c - A^T y - s\|}{1 + \|s\|} + \frac{|c^T x - b^T y|}{1 + |b^T y|} < tol,$$

where tol was set to 10^{-8} . Notice that, in the case of iteration rPDAS, $\|c - A^T y - s\|$ vanishes throughout, up to numerical errors.

Execution times strongly depend on how the computation of $H^Q := A_Q D_Q^2 A_Q^T$ (where $D_Q^2 := S_Q^{-1} X_Q$ is diagonal), involved in (3.2a), is implemented in MATLAB. Storing the $|Q| \times |Q|$ matrix D_Q^2 as a full matrix is inefficient, or even impossible (even when $|Q|$ is much smaller than n , it may still be large). We are left with two options: Either (i) compute an auxiliary matrix $D_Q^2 A_Q^T$ using a “for” loop over the $|Q|$ rows of A_Q^T , then compute $A_Q * D_Q^2 A_Q^T$, or (ii) create a sparse matrix D_Q^2 using the `spdiags` function, then compute $A_Q * D_Q^2 * A_Q^T$. If the matrix A is in sparse form with sufficiently few nonzero elements, then the second option is (much) more efficient. If the matrix A is dense, then both options have comparable speed. Consequently, we used the second option in all experiments.

Numerical results obtained with algorithm rPDAS on several types of problems (to be discussed below) are presented in Figures 1 through 4. The points on the plots, as well as those on the plots of Figures 5 through 12 discussed in section 4, correspond to different runs on the same problem. The runs differ only by the number of constraints M that are retained in Q ; this information is indicated on the horizontal axis in relative value. The rightmost point thus corresponds to the experiment without constraint reduction, while the points on the extreme left correspond to the most drastic constraint reduction. The plots are built as follows: the execution script picks progressively smaller values of M from a predefined list of values until it reaches the end of the list, or early, abnormal termination occurs. The latter was always caused by either (i) the number of iterations reaching a predefined limit of 100 (this is an ad hoc choice to stop the execution script when it reaches values of M where the number of iterations become high), or (ii) MATLAB generating NaN values, which happens when the normal matrix becomes numerically singular. Abnormal termination did occur in the numerical experiment presented in Figure 3 due to (ii) and in a few other instances due to (i).

In the lower plot of each figure, the vertical axis indicates CPU times to solution (total time, as well as time expended in the computation of H^Q and time used for the solution of the normal equation) as returned by the MATLAB function `cputime`. We emphasize that these results are valid only for the specific MATLAB implementation described above. Results could vary widely, depending on the programming language, the possible use of the BLAS, and the hardware. In contrast, the number of iterations shown on the upper plot has more meaning.

The first test problem is of the finely discretized semi-infinite type: the dual feasible set F is a polytope whose faces are tangent to the unit sphere. Contact points on the sphere were selected from the uniform distribution by first generating vectors of numbers distributed according to $\mathcal{N}(0, 1)$ —normal distribution with mean zero and standard deviation one—and then normalizing these vectors. These points form the columns of A , and c was selected as the vector of all ones. Each entry of the objective vector, b , was chosen from $\mathcal{N}(0, 1)$, and y^0 was selected as the zero vector.

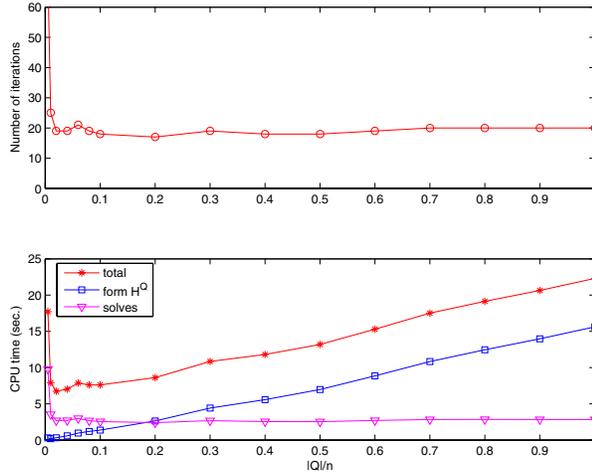


FIG. 1. *rPDAS* on the problem with constraints tangent to the unit sphere.

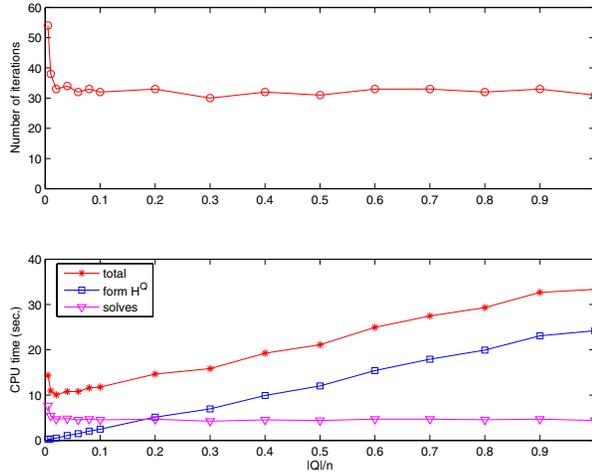
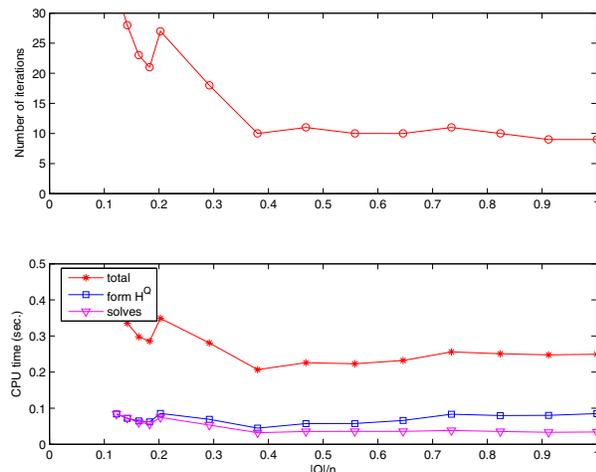


FIG. 2. *rPDAS* on the “fully random” problem.

This yields a problem that lends itself nicely to constraint reduction, since $n \gg m$ (we chose $m = 50$ and $n = 20000$), A is dense, and Assumption 1 on the full rank of submatrices of A holds for M as low as m . Numerical results are presented in Figure 1.

Arguably the most remarkable result in this paper is that observed on the upper plot of Figure 1 (and again in other figures discussed below): the number of iterations shows little variation over a significant range of values of $|Q|$. We tested the algorithm on several problems randomly generated, as explained above, and always observed that only very low values of $|Q|$ produce a significant increase in the number of iterations.

The second test problem is “fully random.” The entries of A and b were generated from $\mathcal{N}(0, 1)$. To ensure a dual-feasible initial point, y^0 and s^0 were chosen from a uniform distribution on $(0, 1)$ and the vector c was generated by taking $c := A^T y^0 + s^0$. We again chose $m = 50$ and $n = 20000$. Results are displayed in Figure 2.

FIG. 3. *rPDAS* on *SCSD1*.

Note that these results are qualitatively similar to those of Figure 1. Here again, the number of iterations is stable over a wide range of values of $|Q|$. Experiments conducted on other test problems drawn from the same distribution produced similar results.

Next, we searched the Netlib LP library for problems where n is significantly greater than m and Assumption 1 is satisfied for reasonably small M . This left us with the SCSD problems. These problems, however, are very sparse. The computation of the normal matrix AD^2A^T involves only sparse matrix multiplications that can be performed efficiently and account only for a small portion of the total execution time. Therefore, the constraint reduction strategy, which focuses on reducing the cost of forming the normal matrix, has little effect on the overall execution time. (If the computation of H^Q is done with a `for` loop as explained above, then an important speedup is observed.) We tested algorithm *rPDAS* on *SCSD1* ($m = 77$, $n = 760$) and *SCSD6* ($m = 147$, $n = 1350$). For both problems, we set y^0 to $0 \in F^o$. Results are displayed in Figures 3 and 4. Here again, the number of iterations is quite stable over a wide range of values of $|Q|$.

4. A reduced MPC algorithm.

4.1. Algorithm statement. We consider a constraint-reduced version of Mehrotra’s predictor-corrector (MPC) method [Meh92]—or rather of the simplified version of that algorithm found in [Wri97].

Iteration *rMPC*.

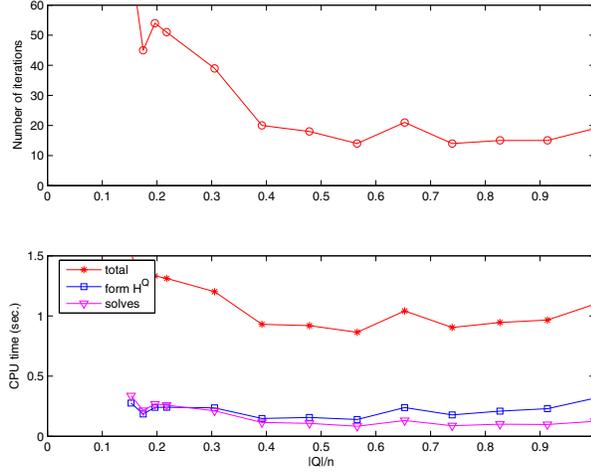
Parameters. $\beta \in (0, 1)$, integer M satisfying $m \leq M \leq n$.

Data. $y \in \mathbb{R}^m$, $s > 0$, $x > 0$, $Q \in \mathcal{Q}_M(y)$, $\mu := x^T s/n$.

Step 1. Compute affine scaling direction:

Solve

$$A_Q S_Q^{-1} X_Q A_Q^T \Delta y = -r_b + A(-S^{-1} X r_c + x)$$


 FIG. 4. *rPDAS* on *SCSD6*.

and compute

$$\begin{aligned}\Delta s &:= -A^T \Delta y - r_c, \\ \Delta x &:= -x - S^{-1} X \Delta s,\end{aligned}$$

and let

$$\begin{aligned}t_{\text{aff}}^{\text{pri}} &:= \arg \max\{t \in [0, 1] \mid x + t\Delta x \geq 0\}, \\ t_{\text{aff}}^{\text{dual}} &:= \arg \max\{t \in [0, 1] \mid s + t\Delta s \geq 0\}.\end{aligned}$$

Step 2. Compute centering parameter:

$$\begin{aligned}\mu_{\text{aff}} &:= (x + t_{\text{aff}}^{\text{pri}} \Delta x)^T (s + t_{\text{aff}}^{\text{dual}} \Delta s) / n, \\ \sigma &:= (\mu_{\text{aff}} / \mu)^3.\end{aligned}$$

Step 3. Compute centering/corrector direction:

Solve

$$A_Q S_Q^{-1} X_Q A_Q^T \Delta y^{\text{cc}} = -A S^{-1} (\sigma \mu e - \Delta X \Delta s)$$

and compute

$$\begin{aligned}\Delta s^{\text{cc}} &:= -A^T \Delta y^{\text{cc}}, \\ \Delta x^{\text{cc}} &:= S^{-1} (\sigma \mu e - \Delta X \Delta s) - S^{-1} X \Delta s^{\text{cc}}.\end{aligned}$$

Step 4. Compute MPC step:

$$\begin{aligned}\Delta x^{\text{mpc}} &:= \Delta x + \Delta x^{\text{cc}}, \\ \Delta y^{\text{mpc}} &:= \Delta y + \Delta y^{\text{cc}}, \\ \Delta s^{\text{mpc}} &:= \Delta s + \Delta s^{\text{cc}},\end{aligned}$$

$$t_{\max}^{\text{pri}} := \arg \max\{t \in [0, 1] \mid x + t\Delta x^{\text{mpc}} \geq 0\},$$

$$t_{\max}^{\text{dual}} := \arg \max\{t \in [0, 1] \mid s + t\Delta s^{\text{mpc}} \geq 0\},$$

$$t^{\text{pri}} := \min\{\beta t_{\max}^{\text{pri}}, 1\},$$

$$t^{\text{dual}} := \min\{\beta t_{\max}^{\text{dual}}, 1\}.$$

Step 5. Updates:

$$x^+ := x + t^{\text{pri}}\Delta x^{\text{mpc}},$$

$$y^+ := y + t^{\text{dual}}\Delta y^{\text{mpc}},$$

$$s^+ := s + t^{\text{dual}}\Delta s^{\text{mpc}}.$$

Pick $Q^+ \in \mathcal{Q}_M(y^+)$.

As compared with the case of Iteration rPDAS, the speed-up per iteration achieved by rMPC over MPC is not as striking. This is due to the presence of two additional matrix-vector products in the iteration (see Step 3) for a total of three matrix-vector products per iteration. Further, these products involve the full A matrix and require $O(mn)$ flops, which can be substantial.

4.2. Numerical results: Dual-feasible initial point. We report on numerical results obtained with a MATLAB implementation of the reduced MPC (rMPC) algorithm.⁴ The hardware, software, test problems, initial points, and presentation of the results are the same as in section 3.3. Figures 5, 6, 7, and 8 are the counterparts of Figures 1, 2, 3, and 4, respectively.

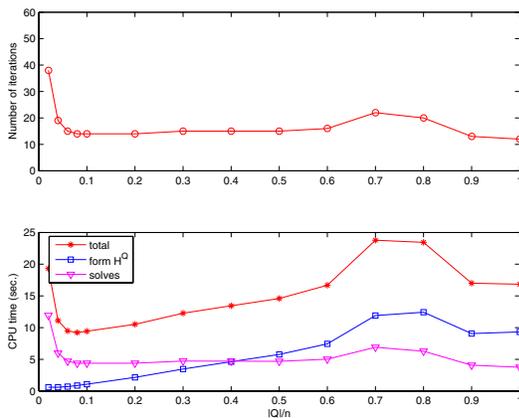


FIG. 5. rMPC on the problem with constraints tangent to the unit sphere, with dual-feasible initial point.

4.3. Numerical results: Infeasible initial point. We now report on numerical experiments that differ from the ones in section 4.2 only by the choice of the initial

⁴The code is available from the authors.

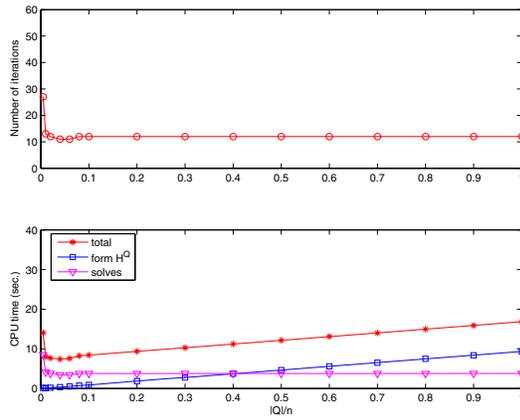


FIG. 6. $rMPC$ on the “fully random” problem, with dual-feasible initial point.

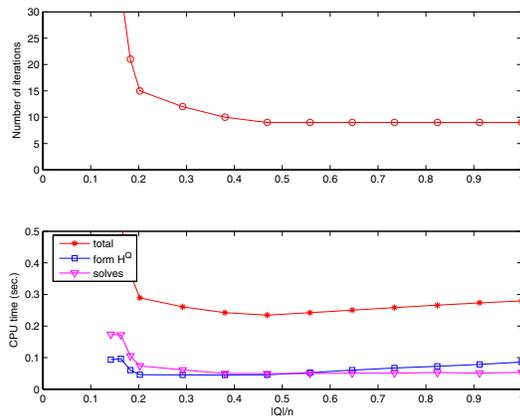


FIG. 7. $rMPC$ on $SCSD1$, with dual-feasible initial point.

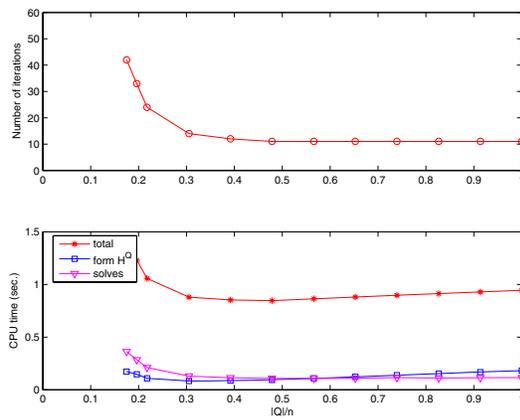


FIG. 8. $rMPC$ on $SCSD6$, with dual-feasible initial point.

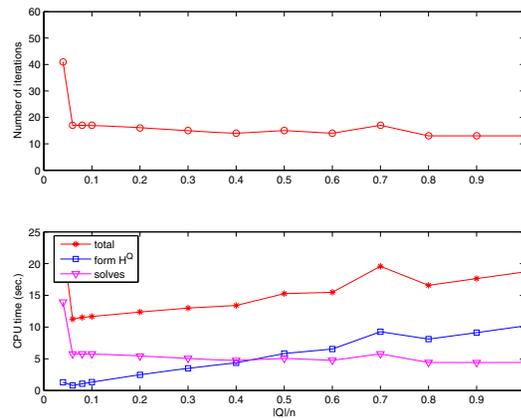


FIG. 9. *rMPC on the problem with constraints tangent to the unit sphere, with infeasible initial point.*

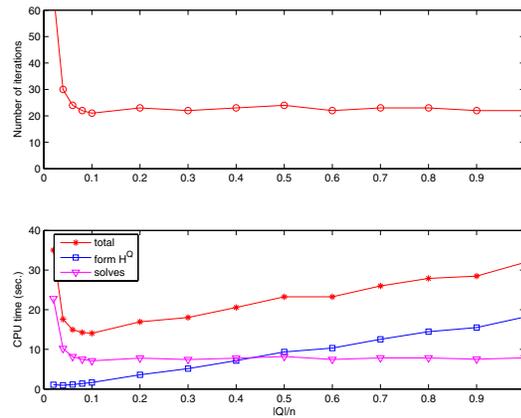


FIG. 10. *rMPC on the “fully random” problem, with infeasible initial point.*

variables. Here we select the initial variables as in [Meh92, p. 589], without modification. Consequently, there is no guarantee that the initial point will be dual-feasible; and indeed, in most experiments, the initial point was dual-infeasible (in addition to being primal-infeasible, as in all the previous experiments). Figures 9, 10, 11, and 12 are the counterparts of Figures 5, 6, 7, 8, respectively.

5. Discussion. In the context of primal-dual interior-point methods for linear programming, a scheme was proposed, aimed at significantly decreasing the computational effort at each iteration when solving problems which, when expressed in dual standard form, have many more constraints than (dual) variables. The core idea is to compute the dual search direction based only on a small subset of the constraints, carefully selected in an attempt to preserve the quality of the search direction. Global and local quadratic convergence was proved for a class of schemes in the case of a simple dual-feasible affine scaling algorithm. Promising numerical results were reported both on this “reduced” affine scaling algorithm and a similarly “reduced” version of the MPC algorithm, using a rather simplistic heuristic: for a prescribed $M > m$, keep only the M most nearly active (or most violated) constraints. In particular,

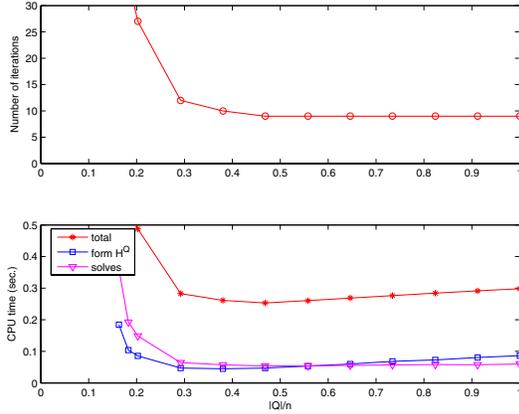


FIG. 11. *rMPC on SCSD1, with infeasible initial point.*

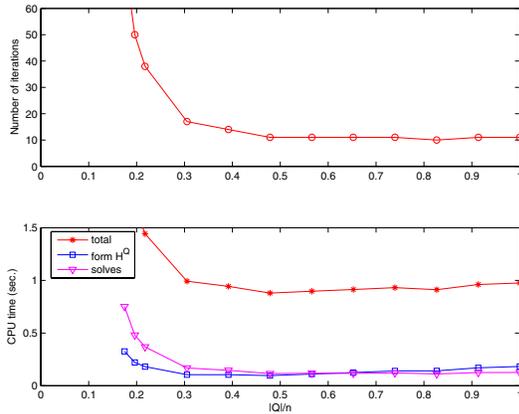


FIG. 12. *rMPC on SCSD6, with infeasible initial point.*

rather unexpectedly, it was observed that, on a number of problems, the number of iterations to solutions did not increase when the size of the reduced constraint set was decreased, down to a small fraction of the total number of constraints! Accordingly, all savings in computational effort per iteration directly translate to savings in total computational effort for the solution of the problem. Another interesting finding is that, in our MATLAB implementation, the reduced affine scaling algorithm rPDAS works as well as the reduced MPC algorithm on random problems in terms of CPU time; however, CPU times may vary widely over implementations.

The (unreduced) MPC algorithm has remarkable invariance properties. Let $\{(x^k, y^k, s^k)\}$ be a sequence generated by MPC on the problem defined by (A, b, c) . Let P be an invertible $m \times m$ matrix, let R be a diagonal positive-definite $n \times n$ matrix, let v belong to \mathbb{R}^m , and define $\mathbf{A} := PAR$, $\mathbf{b} := Pb$, $\mathbf{c} := Rc + RA^T P^T v$, $\mathbf{x}^0 := R^{-1}x^0$, $\mathbf{y}^0 := P^{-T}y^0 + v$, and $\mathbf{s}^0 := Rs^0$. Then the sequence $\{(\mathbf{x}^k, \mathbf{y}^k, \mathbf{z}^k)\}$ generated by MPC on the problem defined by $(\mathbf{A}, \mathbf{b}, \mathbf{c})$ satisfies $\mathbf{x}^k = R^{-1}x^k$, $\mathbf{y}^k = P^{-T}y^k + v$, and $\mathbf{s}^k = Rs^k$.⁵ The reduced algorithm rMPC is still invariant under the action of

⁵Note, however, that the procedure recommended in [Meh92] for generating x^0 and s^0 , while

P and v , but it is no longer invariant under the action of R , because the relation $\mathbf{s} = R\mathbf{s}$ affects the choice of the set Q . A simple way to recover invariance under R is to redefine \mathcal{Q}_M based on $(c_i - a_i^T y)/s_i^0$ instead of $c_i - a_i^T y$.

The rPDAS and PDAS algorithms (rPDAS with $Q = \mathbf{n}$) have weaker invariance properties than MPC. While they are invariant under the action of v and of orthogonal P (that is, Euclidean transformations of the dual space), they are neither invariant under the action of nonorthogonal P , because of the presence of $\|\Delta y\|$ in (3.4) and (3.5), nor under the action of R , because of (3.5) containing the quantity $\|\tilde{x}_-\|$ and fixed bounds on x (and also, for rPDAS, because of the way \mathcal{Q}_M is defined).⁶ Algorithms rPDAS and PDAS can be modified to achieve other invariance properties. If $\|\Delta y\|$ is replaced⁷ by $\|(\Delta Y^0)^{-1}\Delta y\|$, where $\Delta Y^0 = \text{diag}(\Delta y_i^0, i = 1, \dots, m)$, then the algorithms are invariant under v and nonsingular diagonal P . If instead $\|\Delta y\|$ is replaced by $\|\Delta y\|/\|\Delta y^0\|$, then the algorithms are invariant under Euclidean transformation and uniform scaling of the dual (i.e., P is a nonzero scalar multiple of an orthogonal matrix). If (3.5) is replaced by

$$x_i^+ := \min\{\max\{\min\{(\|\Delta y\|^2 + \|(X^0)^{-1}\tilde{x}_-\|^2)x_i^0, \underline{x}x_i^0\}, \tilde{x}_i\}, x_{\max}x_i^0\} \quad \forall i \in \mathbf{n},$$

then PDAS is invariant under R ; if, moreover, \mathcal{Q}_M is redefined based on $(c_i - a_i^T y)/s_i^0$ instead of $c_i - a_i^T y$, then rPDAS becomes invariant under R , too.

We have focused on a constraint selection rule that requires that, at each iteration, the M “most nearly active” (or “most violated”) constraints all be included in the reduced set. It should be clear, however, that nearness to activity can be measured differently for each constraint, and indeed differently at each iteration. In fact, only two conditions must be satisfied in order for our convergence analysis to go through: (i) A_Q must have full row rank at each iteration, which is required in order for the algorithm to be well defined, and (ii) constraints must be included in the reduced set whenever y is “close enough” to the corresponding constraint boundary.

Appendix. Proof of Lemma 1.

The first claim is a direct consequence of the equivalence between (2.2) and (2.3). Let us now prove the sufficiency portion of the second claim. Thus suppose conditions (i) through (iii) hold, and let $(\xi, \eta, \sigma)^T$ be in the nullspace of $J(A, x, s)$. We show that it must be identically zero. We have

$$(A.1) \quad A^T \eta + \sigma = 0,$$

$$(A.2) \quad A\xi = 0,$$

$$(A.3) \quad S\xi + X\sigma = 0.$$

Equation (A.1) yields

$$(A.4) \quad \xi^T A^T \eta + \xi^T \sigma = 0,$$

which, in view of (A.2), yields

$$(A.5) \quad \xi^T \sigma = 0.$$

invariant under the action of P and v , is not invariant under that of R .

⁶Note that simpler versions of PDAS that do not aim at superlinear convergence enjoy v , P , and R invariance as defined above (see, e.g., [MAR90]).

⁷Assuming that no component of Δy^0 vanishes.

Also, (A.3) yields

$$\sigma_i = -\frac{s_i}{x_i}\xi_i$$

whenever $x_i \neq 0$, and $\xi_i = 0$ otherwise (since $|x_i| + |s_i| > 0$), so that (A.5) yields

$$-\sum_{i:x_i \neq 0} \frac{s_i}{x_i}\xi_i^2 = 0.$$

Since $s_i/x_i \geq 0$ whenever $x_i \neq 0$, it follows that $s_i\xi_i = 0$ whenever $x_i \neq 0$. It then follows from (A.3) that $x_i\sigma_i = 0$ whenever $x_i \neq 0$, yielding

$$(A.6) \quad X\sigma = 0$$

and, from (A.3),

$$(A.7) \quad S\xi = 0.$$

Since $\{a_i : s_i = 0\}$ is linear independent, it follows from (A.2) and (A.7) that $\xi = 0$. Equation (A.1) and (A.6) now yield $XA^T\eta = 0$, so that $a_i^T\eta = 0$ whenever $x_i \neq 0$. Since $\{a_i : x_i \neq 0\}$ spans \mathbb{R}^m , we conclude that $\eta = 0$. Finally, it now follows from (A.1) that $\sigma = 0$, concluding the proof of the sufficiency portion of the second claim.

As for the necessity portion of the second claim, first, inspection of the last n rows, then of the first n columns of $J(A, x, s)$, shows that the first two conditions are needed in order for $J(A, x, s)$ to be nonsingular. As for the third condition, suppose it does not hold, i.e., suppose that $\{a_i : x_i \neq 0\}$ does not span \mathbb{R}^m . Then there exists $\eta \neq 0$ such that $a_i^T\eta = 0$ for all i such that $x_i \neq 0$. Further, let $\xi := 0$ and let $\sigma := -A^T\eta$, so that $\sigma_i = 0$ for all i such that $x_i \neq 0$. It is readily checked that (ξ, η, σ) is in the nullspace of $J(A, x, s)$. Since $\eta \neq 0$, $J(A, x, s)$ must be singular. This completes the proof of the necessity portion of the second claim.

Acknowledgments. The authors wish to thank Dianne O’Leary for helpful discussions. Further, they wish to thank two anonymous referees for their careful reading of the paper and for their many helpful comments.

REFERENCES

- [AT06] P.-A. ABSIL AND A. L. TITS, *Newton-KKT interior-point methods for indefinite quadratic programming*, *Comput. Optim. Appl.*, 2006, to appear.
- [dHRT92] D. DEN HERTOOG, C. ROOS, AND T. TERLAKY, *A build-up variant of the logarithmic barrier method for LP*, *Oper. Res. Lett.*, 12 (1992), pp. 181–186.
- [dHRT94] D. DEN HERTOOG, C. ROOS, AND T. TERLAKY, *Adding and deleting constraints in the logarithmic barrier method for LP*, *Advances in Optimization and Approximation*, D. Z. Du and J. Sun, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 166–185.
- [GLY94] J.-L. GOFFIN, Z.-Q. LUO, AND Y. YE, *On the complexity of a column generation algorithm for convex or quasiconvex feasibility problems*, in *Large Scale Optimization*, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 182–191.
- [LRT99] Z.-Q. LUO, K. ROOS, AND T. TERLAKY, *Complexity analysis of logarithmic barrier decomposition methods for semi-infinite linear programming*, *Appl. Numer. Math.*, 29 (1999), pp. 379–394.
- [MAR90] R. D. C. MONTEIRO, I. ADLER, AND M. G. C. RESENDE, *A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension*, *Math. Oper. Res.*, 15 (1990), pp. 191–214.

- [Meh92] S. MEHROTRA, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.
- [NS96] S. G. NASH AND A. SOFER, *Linear and Nonlinear Programming*, McGraw-Hill, New York, 1996.
- [O’L04] D. P. O’LEARY, *private communication*, 2004.
- [PTH88] E. R. PANIER, A. L. TITS, AND J. N. HERSKOVITS, *A QP-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization*, SIAM J. Control Optim., 26 (1988), pp. 788–811.
- [Tit99] A. L. TITS, *An Interior Point Method for Linear Programming, with an Active Set Flavor*, Technical report TR-99-47, Institute for Systems Research, University of Maryland, College Park, MD, 1999.
- [TZ94] A. L. TITS AND J. L. ZHOU, *A simple, quadratically convergent algorithm for linear programming and convex quadratic programming*, Large Scale Optimization, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 411–427.
- [Wri97] S. J. WRIGHT, *Primal-dual Interior-point Methods*, SIAM, Philadelphia, 1997.
- [Ye90] Y. YE, *A “build-down” scheme for linear programming*, Math. Programming, 46 (1990), pp. 61–72.
- [Ye92] Y. YE, *A potential reduction algorithm allowing column generation*, SIAM J. Optim., 2 (1992), pp. 7–20.
- [Ye97] Y. YE, *Complexity analysis of the analytic center cutting plane method that uses multiple cuts*, Math. Programming, 78 (1997), pp. 85–104.