ADAPTIVE CONSTRAINT REDUCTION FOR CONVEX QUADRATIC PROGRAMMING*

JIN HYUK JUNG†, DIANNE P. O'LEARY‡, AND ANDRÉ L. TITS§

Abstract. We propose an adaptive, constraint-reduced, primal-dual interior-point algorithm for convex quadratic programming with many more inequality constraints than variables. We reduce the computational effort by assembling, instead of the exact normal-equation matrix, an approximate matrix from a well chosen index set which includes indices of constraints that seem to be most critical. Starting with a large portion of the constraints, our proposed scheme excludes more unnecessary constraints at later iterations. We provide proofs for the global convergence and the quadratic local convergence rate of an affine-scaling variant. Numerical experiments on random problems, on a data-fitting problem, and on a problem in array pattern synthesis show the effectiveness of the constraint reduction in decreasing the time per iteration without significantly affecting the number of iterations. We note that a similar constraint-reduction approach can be applied to algorithms of Mehrotra's predictor-corrector type, although no convergence theory is supplied.

Keywords: Convex Quadratic Programming, Constraint Reduction, Column Generation, Primal-Dual Interior-Point Method

1. Introduction. Consider the convex quadratic programming (CQP) problem in standard form

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &:= \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{c}^T\mathbf{x}, \\ \text{s.t. } \mathbf{A}\mathbf{x} &\geq \mathbf{b}, \end{aligned} \tag{1.1}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{c} \in \mathbb{R}^n$, and $\mathbf{b} \in \mathbb{R}^m$, and where $\mathbf{H} \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite. The associated dual problem is

$$\begin{aligned} \max_{\mathbf{x},\boldsymbol{\lambda}} f_D(\mathbf{x},\boldsymbol{\lambda}) &:= -\frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{b}^T\boldsymbol{\lambda}, \\ \text{s.t. } \mathbf{H}\mathbf{x} + \mathbf{c} - \mathbf{A}^T\boldsymbol{\lambda} &= \mathbf{0}, \\ \boldsymbol{\lambda} &\geq \mathbf{0}, \end{aligned} \tag{1.2}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of Lagrange multipliers.

In the present paper, we are mainly concerned with the case when (1.1) has many more constraints than variables, i.e., $m \gg n$. A large portion of the constraints are not active at the solution and thus do not contribute much to deciding the search direction of the later iterations of an interior-point method (IPM) used to solve the problem. Since the major work in computing

*This work was supported by the US Department of Energy under Grant DEFG0204ER25655, and by the National Science Foundation under Grant DMI0422931 (A.L.T). 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.

†Department of Computer Science, University of Maryland, College Park, MD 20742 salbang@gmail.com

‡Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 oleary@cs.umd.edu

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

the search direction involves forming a matrix at a cost proportional to the number of constraints, computing the matrix without irrelevant constraints reduces the computational burden.

Reducing computational cost by computing search directions using only a fraction of the constraints has been actively studied. The most prominent approach is "column generation". Ye [30] used this approach with a "build-down" scheme for linear programming (LP). He proposed a rule which can safely eliminate inequality constraints that will not be active at the optimum. He applied the rule to Karmarkar's method [15] and the simplex method [5]. Dantzig and Ye [6] proposed a "build-up" interior-point method of dual affine-scaling form. Starting from a strictly dual-feasible point, it uses a subset of constraints to determine the search direction at each iteration. The direction is accepted if taking the step violates no constraint. If some constraints are violated, the algorithm adds them to the working set and retries. Ye [32] proposed a potential reduction algorithm allowing column generation for linear feasibility problems (to which LP problems can be converted). Starting with a polytope including the feasible domain, at every iteration, the scheme builds a cutting plane when a violated inequality is encountered. Luo and Sun [16] proposed a similar scheme for convex quadratic feasibility problems (to which CQP problems can be transformed). Tone [25] proposed an active set strategy for the dual potential reduction algorithm for LP proposed by Ye [31]. The strategy finds the search direction using constraints associated with small dual slack variables.

Den Hertog et al. proposed a "build-up" path-following method for LP [9], which follows the central path corresponding to a fraction of constraints until the iterate violates or almost violates some of the other constraints. After adding the constraints to the working set, the algorithm restarts from the previous iterate. The process is continued until the iterate is close enough to an optimum. The same authors later added "build-down" strategies to the path-following method [10].

The primal-dual LP constraint-reduction algorithms of Tits et al. [23] (affine scaling) and Winternitz et al. [28] (Mehrotra's predictor-corrector) choose constraints at each iteration without building up. Convergence of these algorithms was proven, and experiments demonstrated good performance. An attractive aspect of the constraint-reduction scheme considered in these papers is its easy applicability to the state-of-the-art primal-dual IPMs (PDIPMs) such as variants of Mehrotra's predictor-corrector algorithm [17, 29, 18]. The purpose of this paper is to extend the affine scaling algorithm and convergence results of [23] from LP to CQP.

In this paper, we present a primal-dual affine-scaling constraint-reduction algorithm for CQP that inherits the good properties of the constraint-reduction algorithm [23] for LP and of the primal-dual affine-scaling IPM [24, 1] for QP. In addition, we propose an adaptive scheme for reducing the number of constraints involved in finding the search direction. Since it becomes more obvious which constraints would be active as the iterate gets closer to a solution, eliminating more seemingly inactive constraints in later iterations should not impair the quality of the search direction. In our new scheme for CQP, the size of the constraint set is determined by how close the current point is to the solution. Either the duality gap or a complementarity measure[1] provides a good criterion. This paper complements the results in [14], where we formulate (without convergence proof) a related algorithm, a constraint-reduced CQP version of Mehrotra's predictor-corrector algorithm which is

[1] This is often called the *duality measure*.

specialized to the problem of training linear and nonlinear support-vector machines. The focus in that paper is on application-specific issues such as kernalization.

In Section 2, we present a constraint-reduction algorithm for CQP. In particular, we provide a new interpretation for constraint reduction in Section 2.2. In Section 3, we discuss the case when a strictly feasible starting point is not readily available. In Section 4, numerical results are presented. Concluding remarks are provided in Section 5. Finally, a convergence analysis is provided in Appendix A.

2. Adaptive Constraint Reduction for Convex Quadratic Programming.

2.1. Notation. Let $M := \{1, ..., m\}$, and let $\mathbf{a}_i$ be the transpose of the ith row of $\mathbf{A}$. For an index set $Q \subseteq M$, let $\mathbf{A}_Q$ be a submatrix of $\mathbf{A}$ constructed by deleting rows $\mathbf{a}_i^T$ for $i \notin Q$. The same notation $\mathbf{v}_Q$ is applied to a column vector $\mathbf{v} \in \mathbb{R}^m$. For an $m \times m$ matrix $\mathbf{B}$, we let $\mathbf{B}_{Q^2}$ denote a submatrix of $\mathbf{B}$ constructed by deleting both rows and columns indexed by $i \notin Q$. The complement Q^c of an index set Q is defined as $Q^c := M \setminus Q$. We define the feasible set $\mathcal{F}_P := \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} \geq \mathbf{b}\}$, the strictly feasible set $\mathcal{F}_P^o := \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} > \mathbf{b}\}$, and the solution set $\mathcal{F}_P^* := \{\mathbf{x}^* \in \mathcal{F}_P : f(\mathbf{x}^*) \leq f(\mathbf{x}),\ \forall \mathbf{x} \in \mathcal{F}_P\}$. We use $\mathcal{N}(\mathbf{A}) := \{\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{0}\}$ to denote the null space of $\mathbf{A}$. We define the index set $\mathcal{A}(\mathbf{x})$ of active constraints at $\mathbf{x} \in \mathcal{F}_P$ as

$$\mathcal{A}(\mathbf{x}) := \{i \in M : \mathbf{a}_i^T \mathbf{x} = b_i\}. \tag{2.1}$$

With the help of a slack vector $\mathbf{s}$, the Karush-Kuhn-Tucker (KKT) conditions for (1.1) are written as

$$\mathbf{H}\mathbf{x} - \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{c} = \mathbf{0}, \tag{2.2}$$
$$\mathbf{A}\mathbf{x} - \mathbf{b} - \mathbf{s} = \mathbf{0}, \tag{2.3}$$
$$\mathbf{S}\boldsymbol{\lambda} = \mathbf{0}, \tag{2.4}$$
$$\mathbf{s}, \boldsymbol{\lambda} \geq \mathbf{0}, \tag{2.5}$$

where $\mathbf{S} := \text{diag}(\mathbf{s})$. Since the objective function is convex and the constraints are linear, the KKT conditions are necessary and sufficient for the optimality of $\mathbf{x}$ and $\boldsymbol{\lambda}$ for (1.1) and (1.2) (see [29, Appendix A] for a proof).

Starting with a point $(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda})$ satisfying $\mathbf{s} > \mathbf{0}$ and $\boldsymbol{\lambda} > \mathbf{0}$, variants of Newton's method can be applied to the equalities (2.2)-(2.4), keeping $\mathbf{s}$ and $\boldsymbol{\lambda}$ inside the positive orthant. Such methods, which involve (2.4) rather than a perturbed version thereof, are termed "affine-scaling"-type methods. We define $\boldsymbol{\Lambda} := \text{diag}(\boldsymbol{\lambda})$, $\mathbf{r_c} := \mathbf{H}\mathbf{x} + \mathbf{c} - \mathbf{A}^T\boldsymbol{\lambda}$, and $\mathbf{r_b} := \mathbf{A}\mathbf{x} - \mathbf{b} - \mathbf{s}$.

2.2. Adaptive constraint reduction. In this section, we present a primal-feasible, adaptive constraint-reduction method based on the constraint-reduced dual-feasible primal-dual affine-scaling algorithm for LP proposed in [23]. Indeed the primal (1.1) with $\mathbf{H} := \mathbf{0}$ (and $\mathbf{b} := -\mathbf{c}$, $\mathbf{c} := -\mathbf{b}$, $\mathbf{A} := -\mathbf{A}^T$) corresponds to the *dual* formulation of [23].

Under primal feasibility, the primal-dual affine-scaling search direction $(\Delta\mathbf{x}, \Delta\mathbf{s}, \Delta\boldsymbol{\lambda})$ is the

Newton direction for (2.2)-(2.4) with $\mathbf{r_b} := \mathbf{0}$:

$$\begin{bmatrix} \mathbf{H} & \mathbf{A}^T & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{\Lambda} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\boldsymbol{\lambda} \\ \Delta\mathbf{s} \end{bmatrix} = - \begin{bmatrix} \mathbf{r_c} \\ \mathbf{r_d} \\ \mathbf{S\lambda} \end{bmatrix}. \tag{2.6}$$

Using two steps of block Gauss elimination it can be seen that this direction solves the normal equation system

$$\mathbf{M}\Delta\mathbf{x} = -(\mathbf{Hx} + \mathbf{c}), \tag{2.7}$$

$$\Delta\mathbf{s} = \mathbf{A}\Delta\mathbf{x}, \tag{2.8}$$

$$\Delta\boldsymbol{\lambda} = -\boldsymbol{\lambda} - \mathbf{S}^{-1}\mathbf{\Lambda}\Delta\mathbf{s}, \tag{2.9}$$

where

$$\mathbf{M} := \mathbf{H} + \mathbf{A}^T\mathbf{S}^{-1}\mathbf{\Lambda}\mathbf{A} = \mathbf{H} + \sum_{i=1}^{m} s_i^{-1}\lambda_i \mathbf{a}_i\mathbf{a}_i^T. \tag{2.10}$$

The dominant work in computing $(\Delta\mathbf{x}, \Delta\mathbf{s}, \Delta\boldsymbol{\lambda})$ is forming $\mathbf{M}$, which requires approximately $mn^2/2$ multiplications if $\mathbf{A}$ is dense.

Now suppose that we have prior knowledge of which constraints are active at the solution $\mathbf{x}^*$, and that the solution is unique and strictly complementary. Then if we have any set Q such that $\mathcal{A}(\mathbf{x}^*) \subseteq Q$, we can find $\mathbf{x}^*$ by solving the reduced minimization problem

$$\begin{aligned} &\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^T\mathbf{Hx} + \mathbf{c}^T\mathbf{x}, \\ &\text{s.t. } \mathbf{A}_Q\mathbf{x} \geq \mathbf{b}_Q. \end{aligned} \tag{2.11}$$

The affine-scaling search direction for this problem is obtained by solving the reduced normal equations

$$\mathbf{M}_{(Q)}\Delta\mathbf{x} = -(\mathbf{Hx} + \mathbf{c}), \tag{2.12}$$

in place of (2.7), with

$$\mathbf{M}_{(Q)} := \mathbf{H} + \mathbf{A}_Q^T\mathbf{S}_{Q^2}^{-1}\mathbf{\Lambda}_{Q^2}\mathbf{A}_Q,$$

generalizing to the CQP context the idea used in [23] in the context of linear programming.

In reality, we do not know which constraints will be active until we obtain the solution. But recall that if the jth constraint is inactive at $\mathbf{x}^*$, then complementary slackness (2.4) implies that the corresponding KKT multiplier λ_j^* vanishes. Therefore, as we approach the solution, the term $s_j^{-1}\lambda_j\mathbf{a}_j\mathbf{a}_j^T$ makes almost no contribution to $\mathbf{M}$ in (2.10). This suggests that approximate Newton search directions can be found without involving constraints that are unlikely to be active at the optimal solution, and that we can recognize such constraints by small values of λ_j. Similarly, we can recognize candidates for constraints that are active at the optimal solution by small values of s_j.

Therefore, our strategy is to try to include in Q constraints that seem likely to be active, and we can vary Q iteration by iteration. This reduces the cost of matrix formation from $mn^2/2$ to $|Q|n^2/2$ multiplications. Following [23], we leave (2.8) and (2.9) as-is; the former guarantees primal feasibility whenever identical steps are taken along $\Delta\mathbf{x}$ and $\Delta\mathbf{s}$.

More formally, following [23], we choose Q to include indices of q smallest components of $\mathbf{Ax}-\mathbf{b}$, breaking ties in an arbitrary way; i.e.,

$$Q \in \mathcal{Q}(\mathbf{Ax}-\mathbf{b}, q), \tag{2.13}$$

where

$$\begin{aligned}\mathcal{Q}(\mathbf{s}, q) := \{Q \subseteq M : \text{rank}([\mathbf{H},\ \mathbf{A}_Q^T]) = n \text{ and } \exists Q' \subseteq Q \text{ s.t.} \\ |Q'| = q \text{ and } s_i \le s_j,\ \forall i \in Q',\ \forall j \notin Q',\}.\end{aligned} \tag{2.14}$$

Finding Q in $\mathcal{Q}(\mathbf{s}, q)$ requires sorting ($O(m \log m)$ operations), which is negligible additional work compared to the matrix formation. Note that $\text{rank}([\mathbf{H},\ \mathbf{A}_Q^T]) = n$ if and only if $\mathcal{N}(\mathbf{H}) \cap \mathcal{N}(\mathbf{A}_Q) = \{\mathbf{0}\}$.

To guarantee a successful iteration, we need to ensure that the matrix $\mathbf{M}_{(Q)}$ is positive definite.

LEMMA 2.1. *(Corresponds to Lemma 2 of [23]) Let $\boldsymbol{\lambda} > \mathbf{0}$, $\mathbf{s} > \mathbf{0}$, and $Q \subseteq M$ such that* $\text{rank}([\mathbf{H},\ \mathbf{A}_Q^T]) = n$. *Then $\mathbf{M}_{(Q)}$ is positive definite.*

Proof. If $[\mathbf{H},\ \mathbf{A}_Q^T]$ has full rank, then $\mathcal{N}(\mathbf{H}) \cap \mathcal{N}(\mathbf{A}_Q) = \{\mathbf{0}\}$. Since both $\mathbf{H}$ and $\mathbf{A}_Q^T \mathbf{S}_{Q^2}^{-1} \boldsymbol{\Lambda}_{Q^2} \mathbf{A}_Q$ are positive semidefinite, it immediately follows that their sum is positive definite. ∎

Although using a very small index set Q greatly reduces the cost of matrix assembly, it makes it more likely that Q misses important constraints in early iterations. As a result, the quality of the search direction could be impaired [14], resulting in an increase in the iteration count. To keep the iteration count low, we use a large number of appropriately selected constraints in early iterations, but exclude more constraints in later iterations as the complementary measure $\mu := \frac{\mathbf{s}^T\boldsymbol{\lambda}}{m}$ becomes smaller. Specifically, based on two user-selected parameters q_U (an upper bound for q) and β, with $n \le q_U \le m$ and $\beta \ge 0$, we set

$$q := \begin{cases} n & \text{, if } \mu^\beta m \le n, \\ \lceil \mu^\beta m \rceil & \text{, if } n < \mu^\beta m \le q_U, \\ q_U & \text{, if } q_U < \mu^\beta m. \end{cases} \tag{2.15}$$

This leads to Algorithm 1, based on the algorithm from [24] with the addition of constraint reduction and a slight modification in the specifics of the dual update (2.23) as in [1, 23]. Notice that q is determined at each iteration.

Algorithm 1 Primal-Feasible Primal-Dual Affine-Scaling Quadratic Programming Algorithm

---

Parameters. $\eta \in (0,1)$, $\beta \ge 0$, $\lambda_{\max}$ and $\underline{\lambda}$ satisfying $\lambda_{\max} > 0$, $\underline{\lambda} > 0$, $q_U \in \{n, \ldots, m\}$, $tol > 0$.
Data. $\mathbf{x}^0 \in \mathcal{F}_P^o$ and $\boldsymbol{\lambda}^0 > \mathbf{0}$.
Set

$$\mathbf{s}^0 := \mathbf{Ax}^0 - \mathbf{b}. \tag{2.16}$$

for $k = 0,...$ **do**

Compute $\mu^k := \mathbf{s}^{kT}\boldsymbol{\lambda}^k/m$.

Terminate if

$$\frac{\|\mathbf{H}\mathbf{x}^k + \mathbf{c} - \mathbf{A}^T\boldsymbol{\lambda}^k\|_\infty}{\max(\|\mathbf{A}\|_\infty, \|\mathbf{H}\|_\infty, \|\mathbf{c}\|_\infty)} \leq tol, \frac{\|\mathbf{A}\mathbf{x}^k - \mathbf{b} - \mathbf{s}^k\|_\infty}{\max(\|\mathbf{A}\|_\infty, \|\mathbf{b}\|_\infty)} \leq tol, \text{ and } \mu^k \leq tol. \tag{2.17}$$

Step 1. Choose the index set:
Choose q such that $n \leq q \leq q_U$ using (2.15) and choose $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x}^k - \mathbf{b}, q)$.

Step 2. Compute a feasible descent direction $\Delta\mathbf{x}^k$, $\Delta\mathbf{s}^k$, and $\Delta\boldsymbol{\lambda}^k$ satisfying the reduced normal equations (2.12) and (2.8)-(2.9).

Set $\tilde{\boldsymbol{\lambda}}^k := \boldsymbol{\lambda}^k + \Delta\boldsymbol{\lambda}$ and

$$\tilde{\boldsymbol{\lambda}}^k_- := \min\{\tilde{\boldsymbol{\lambda}}, \mathbf{0}\}. \tag{2.18}$$

Step 3. Updates:
Compute the largest feasible primal step length.

$$\bar{\alpha}^k := \begin{cases} \infty & \text{if } \Delta\mathbf{s} \geq \mathbf{0}, \\ \min_i\{-\frac{s_i}{\Delta s_i} \,|\, \Delta s_i < 0,\, i \in M\} & \text{otherwise.} \end{cases} \tag{2.19}$$

Set

$$\alpha^k := \begin{cases} \eta\bar{\alpha}^k & \text{, if } \bar{\alpha}^k - \|\Delta\mathbf{x}^k\| \leq \eta\bar{\alpha}^k < 1, \\ \bar{\alpha}^k - \|\Delta\mathbf{x}^k\| & \text{, if } \eta\bar{\alpha}^k < \bar{\alpha}^k - \|\Delta\mathbf{x}^k\| < 1, \\ 1 & \text{, otherwise.} \end{cases} \tag{2.20}$$

Take the step

$$\mathbf{x}^{k+1} := \mathbf{x}^k + \alpha^k\Delta\mathbf{x}^k, \tag{2.21}$$

$$\mathbf{s}^{k+1} := \mathbf{s}^k + \alpha^k\Delta\mathbf{s}^k. \tag{2.22}$$

Notice $\mathbf{s}^{k+1} = \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}$ and $\mathbf{s}^{k+1} > \mathbf{0}$, since, when $\Delta\mathbf{s} \ngeq \mathbf{0}$, $\alpha^k < \bar{\alpha}^k$ and $\mathbf{s} + \bar{\alpha}^k\Delta\mathbf{s} \geq 0$.

For $i = 1, ..., m$, set

$$\lambda_i^{k+1} = \min\{\max\{\min\{\|\Delta\mathbf{x}^k\|^2 + \|\tilde{\boldsymbol{\lambda}}^k_-\|^2, \underline{\lambda}\}, \tilde{\lambda}_i^k\}, \lambda_{\max}\}, \forall i \in M. \tag{2.23}$$

end for

For a motivation for the update rules for λ^k and α^k, see Remark 1 in [23].

2.3. Practicalities. The success of the algorithm depends on choice of scalar parameters, scaling for numerical stability, and choice of Q.

2.3.1. Scalar Parameters. The most critical algorithm parameter should be q_U, the upper bound on q. However, the numerical results of Section 4 show that the performance of the algorithm is remarkably insensitive to this choice, as long as it is at least as big as the number of constraints active at the optimal solution. Choosing $q_U = 3n$ is a good heuristic.

We had success using the parameter $\beta := 1/4$ to control constraint reduction speed (see (2.15)), and $\eta := .98$. Upper bound $\lambda_{\max}$ on the components of λ solves a technical problem in the convergence proof. The bound $\underline{\lambda}$ is unnecessary in theory (it could be set to ∞), but choosing it small improves the performance of the algorithm. Other parameter settings might be better.

2.3.2. Scaling. Rows or columns of the coefficient matrix $\mathbf{A}$ are often associated with different measurement units, making the scales quite different. This can cause the condition number of the matrix to be unnecessarily large, creating numerical instability. We used the following heuristic to scale the problem: we normalized every row of $\mathbf{A}$ to length 1, and scaled $\mathbf{b}$ accordingly. In other words, we used the equivalent constraints $(\mathbf{DA})\mathbf{x} \geq \mathbf{Db}$, with $d_{ii} := \frac{1}{\|\mathbf{a}_i\|}$ for $i = 1, \ldots, m$. This diagonal scaling has the advantage of making $s_i^{-1}\lambda_i$ a reasonable indicator of the term's contribution to (2.10).

2.3.3. Selecting Q and Assembling $\mathbf{M}_Q$. At each iteration, we choose Q to contain the q smallest s_i values, where q is defined by (2.15). Then this Q is in $\mathcal{Q}(\mathbf{s}, q)$ if $\operatorname{rank}([\mathbf{H},\ \mathbf{A}_Q]) = n$. To determine whether this condition is satisfied, we compute the Cholesky factor of $\mathbf{M}_Q$, which will have a zero row if $\mathbf{M}$ is rank-deficient [12, 11]. If this happens, we could repeatedly update the factor, using the next most active constraints [28], until $\mathbf{M}_Q$ becomes full rank. However, for ease of implementation, we instead repeatedly doubled q until the matrix became nonsingular.

Following [23], we also enforced a "safeguard" on $\mathbf{s}$, $s_i := \max(10^{-14}, s_i)$, for the purpose of assembling $\mathbf{M}_{(Q)}$. This keeps $\mathbf{M}_{(Q)}$ from being too ill-conditioned. In addition, when the Cholesky factorization routine chol failed to factor $\mathbf{M}_{(Q)}$ for $Q = M$ due to numerical difficulty, we used the Cholesky infinity factorization cholinc(·,'inf') instead [33].

2.3.4. Choosing a Starting Point. Choosing a good starting point is problem dependent. Several options are illustrated in the experiments in Section 4.

2.4. Convergence of the Adaptive Constraint-Reduction Algorithm. In this section we summarize the convergence properties of our algorithm. The proof of global convergence is in Appendix A.1, and that for local convergence is in Appendix A.2.

Four assumptions ensure global convergence of the algorithm. The first assumption guarantees that $\mathcal{Q}(\mathbf{s}, q)$ is nonempty for all q and all $\mathbf{x} \in \mathcal{F}_P$.

ASSUMPTION 2.1. $[\mathbf{H},\ \mathbf{A}^T]$ *has full row rank.*

In view of Lemma 2.1 and this assumption, the iteration of Algorithm 1 is well defined and constructs an infinite sequence if the termination criteria are ignored.

To guarantee that a starting point for Algorithm 1 and a solution for the problem exist, and that the sequence $\{\mathbf{x}^k\}$ is bounded, we make the following two assumptions.

ASSUMPTION 2.2. $\mathcal{F}_P^o \neq \emptyset$.

ASSUMPTION 2.3. $\mathcal{F}_P^*$ *is nonempty and bounded.*

We impose a constraint qualification.

ASSUMPTION 2.4. *$\forall \mathbf{x} \in \mathcal{F}_P$, $\{\mathbf{a}_i^T : i \in \mathcal{A}(\mathbf{x})\}$ is a linearly independent set.*

If $\{\mathbf{a}_i^T : i \in \mathcal{A}(\mathbf{x})\}$ is a linearly independent set, then $\mathbf{A}_{\mathcal{A}(\mathbf{x})}$ has full row rank and $|\mathcal{A}(\mathbf{x})| \leq n$. Accordingly, Assumption 2.4 guarantees that $\mathcal{A}(\mathbf{x}) \subseteq Q$ for every $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x}-\mathbf{b}, q)$ with $q \geq n$ and $\mathbf{x} \in \mathcal{F}_P$. This is a key property of Q required for the convergence proof. Under these assumptions, we can prove convergence of the algorithm.

THEOREM 2.2. *$\{\mathbf{x}^k\}$ converges to $\mathcal{F}_P^*$.*

In outline, we prove this result in several steps:

- We show in Proposition A.2 that $\alpha^k > 0$, $\mathbf{s}^{k+1} > \mathbf{0}$, $\boldsymbol{\lambda}^{k+1} > 0$, $\mathbf{x}^{k+1} \in \mathcal{F}_P^o$, and $\Delta\mathbf{x}^k \neq \mathbf{0}$ as long as $\mathbf{H}\mathbf{x}^k + \mathbf{c} \neq \mathbf{0}$.
- Proposition A.4 establishes descent properties at step k of the algorithm when $\Delta\mathbf{x}^k \neq \mathbf{0}$.
- This, in conjunction with the boundedness of the level sets (a consequence of Assumption 2.3), shows that the sequence of $\mathbf{x}$ iterates is bounded (Cor. A.5).
- For large k, the set Q at iteration k always contains the indices of the constraints that are active at the optimal solution (Lemma A.6).
- A study of the limit points of the bounded sequence leads to the convergence result.

To establish a q-quadratic local convergence rate, we impose two more assumptions.

ASSUMPTION 2.5. *$\mathcal{F}_P^*$ is a singleton.*

ASSUMPTION 2.6. *The Lagrange multipliers $\boldsymbol{\lambda}^*$ associated with the optimal solution $\mathbf{x}^*$ are strictly complementary to $\mathbf{s}^* := \mathbf{A}^T\mathbf{x}^* - \mathbf{b}$, i.e., $\lambda_i^* s_i^* = 0$ and $\lambda_i^* + s_i^* > 0$ for all $i \in M$.*

Notice Assumption 2.5 implies that $\mathcal{N}(\mathbf{A}_{\mathcal{A}(\mathbf{x}^*)}) \cap \mathcal{N}(\mathbf{H}) = \{\mathbf{0}\}$, or equivalently $[\mathbf{H},\ \mathbf{A}_{\mathcal{A}(\mathbf{x}^*)}]$ spans $\mathbb{R}^n$ for the optimal solution $\mathbf{x}^*$.

With these additional assumptions, we can establish a rate of convergence.

THEOREM 2.3. *Let $\boldsymbol{\lambda}^*$ be the Lagrange multipliers associated with the optimal solution $\mathbf{x}^*$. If $\lambda_i^* < \lambda_{\max}$ for all $i \in M$, then $\{(\mathbf{x}^k, \boldsymbol{\lambda}^k)\}$ converges to the primal and dual optimal solution pair $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ q-quadratically, i.e., there exist a nonnegative integer k' and a constant c such that, for all $k \geq k'$,*

$$\|(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) - (\mathbf{x}^*, \boldsymbol{\lambda}^*)\| \leq c\|(\mathbf{x}^k, \boldsymbol{\lambda}^k) - (\mathbf{x}^*, \boldsymbol{\lambda}^*)\|^2. \tag{2.24}$$

3. Extended Problems: Infeasible Starting Point. Algorithm 1 requires the availability of a strictly feasible starting point $\mathbf{x}^0 \in \mathcal{F}_P^o$. Such point is not always readily available. One application where it is not is considered in section 4.3 below. We deal with such situations by adding a nonnegative relaxation variable y_i to each constraint and a penalty for violations, which leads to the following extended problem:

$$\min_{\mathbf{x},\mathbf{y}} f_E(\mathbf{x}, \mathbf{y}) := \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{c}^T\mathbf{x} + \mathbf{d}^T\mathbf{y} \tag{3.1}$$

$$\text{s.t. } \mathbf{A}\mathbf{x} + \mathbf{y} \geq \mathbf{b}, \tag{3.2}$$

$$\mathbf{y} \geq \mathbf{0}, \tag{3.3}$$

where $\mathbf{y} := [y_1, \ldots, y_m]^T$, $\mathbf{d} \in \mathbb{R}^m$, and $\mathbf{d} > \mathbf{0}$[2]. In this problem, d_i drives y_i to zero and $\mathbf{Ax} \geq \mathbf{b}$ to feasibility. If each component of $\mathbf{d}$ is large enough (relative to the magnitude of the dual solution), then we obtain the same optimal solution $\mathbf{x}^*$ from the original and extended problem (3.1)-(3.3) [8, Sec. 16.5]. Note that it is standard to increase, if necessary, the magnitude of $\mathbf{d}$ during the course of the algorithm.

We focus in the next subsection on the very careful way the normal equations need to be formed in order to take full advantage of constraint reduction for these problems.

3.1. Constraint Reduction for Extended Problems. We can convert the extended problem (3.1)-(3.3) to the standard form (1.1) by defining

$$\mathbf{z} := \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \hat{\mathbf{b}} := \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}, \hat{\mathbf{c}} := \begin{bmatrix} \mathbf{c} \\ \mathbf{d} \end{bmatrix}, \hat{\mathbf{A}} := \begin{bmatrix} \mathbf{A} & \mathbf{I}_m \\ \mathbf{0} & \mathbf{I}_m \end{bmatrix}, \text{ and } \hat{\mathbf{H}} := \begin{bmatrix} \mathbf{H} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{3.4}$$

where $\mathbf{I}_m$ is the $m \times m$ identity matrix, $\hat{\mathbf{A}}$ is $2m \times (m+n)$ and $\hat{\mathbf{H}}$ is $(m+n) \times (m+n)$. By introducing slack variables $\mathbf{s}$ for (3.2) and $\mathbf{w}$ for (3.3), and defining $\mathbf{t} := [\mathbf{s}^T,\ \mathbf{w}^T]^T$, $\boldsymbol{\phi} := [\boldsymbol{\lambda}^T,\ \boldsymbol{\pi}^T]^T$, $\mathbf{Y} := \text{diag}(\mathbf{y})$, $\mathbf{W} := \text{diag}(\mathbf{w})$, $\boldsymbol{\Pi} := \text{diag}(\boldsymbol{\pi})$, $\mathbf{T} := \text{diag}(\mathbf{t})$, and $\boldsymbol{\Phi} := \text{diag}(\boldsymbol{\phi})$, we obtain the KKT conditions for the converted standard form (see (2.2)-(2.5))

$$\hat{\mathbf{H}}\mathbf{z} - \hat{\mathbf{A}}^T\boldsymbol{\phi} + \hat{\mathbf{c}} = \mathbf{0}, \tag{3.5}$$

$$\hat{\mathbf{A}}\mathbf{z} - \hat{\mathbf{b}} - \mathbf{t} = \mathbf{0}, \tag{3.6}$$

$$\mathbf{T}\boldsymbol{\phi} = \mathbf{0}, \tag{3.7}$$

$$\mathbf{t},\ \boldsymbol{\phi} \geq \mathbf{0}. \tag{3.8}$$

The corresponding normal equations, of size $(m+n) \times (m+n)$, are (see (2.7), (2.10))

$$(\hat{\mathbf{H}} + \hat{\mathbf{A}}^T\mathbf{T}^{-1}\boldsymbol{\Phi}\hat{\mathbf{A}})\Delta\mathbf{z} = -(\hat{\mathbf{H}}\mathbf{z} + \hat{\mathbf{c}}), \tag{3.9}$$

and we compute the other variables by solving (see (2.8)-(2.9))

$$\Delta\mathbf{t} = \hat{\mathbf{A}}\Delta\mathbf{z}, \tag{3.10}$$

$$\Delta\boldsymbol{\phi} = -\boldsymbol{\phi} - \mathbf{T}^{-1}\boldsymbol{\Phi}\Delta\mathbf{t}. \tag{3.11}$$

Similarly to the reduced normal equations (2.12) for the original standard form (1.1), we can derive reduced normal equations of size $(m+n) \times (m+n)$ with an index set, for $\hat{q} \geq m+n$,

$$\hat{Q} \in \hat{\mathcal{Q}}(\hat{\mathbf{A}}\mathbf{z} - \hat{\mathbf{b}}, \hat{q}), \tag{3.12}$$

where

$$\begin{aligned} \hat{\mathcal{Q}}(\mathbf{t}, \hat{q}) := \{\hat{Q} \subseteq \{1, \ldots, 2m\} :\ &\text{rank}([\hat{\mathbf{H}},\ \hat{\mathbf{A}}_{\hat{Q}}^T]) = m+n, \text{ and } \exists \hat{Q}' \subseteq \hat{Q} \text{ s.t.} \\ &|\hat{Q}'| = \hat{q}, \text{ and } t_i \leq t_j, \forall i \in \hat{Q}', \forall j \notin \hat{Q}'\}. \end{aligned} \tag{3.13}$$

[2]Problem (3.1)-(3.3) can also be used as a relaxation to problem (1.1), when (1.1) is infeasible. Applications include the soft margin support vector machine (SVM) [4, 3, 21, 14] and support vector regression [21, chap. 1].

So, consistent with (2.14), $\hat{Q}$ includes the indices of $\hat{q}$ most nearly active constraints in (3.2) and (3.3). We begin with the reduced normal equations obtained from (3.9):

$$\left(\hat{\mathbf{H}} + \hat{\mathbf{A}}_{\hat{Q}}^T \mathbf{T}_{\hat{Q}^2}^{-1} \mathbf{\Phi}_{\hat{Q}^2} \hat{\mathbf{A}}_{\hat{Q}}\right) \Delta \mathbf{z} = -(\hat{\mathbf{H}}\mathbf{z} + \hat{\mathbf{c}}). \tag{3.14}$$

This system of equations is of size $(m+n)\times(m+n)$, but well structured. It would cost $O(|\hat{Q}|(m+n)^2)$ multiplications to naively form the matrix on the left-hand side of (3.14) (if we do not exploit the structure of $\hat{\mathbf{A}}$), and the gain we could achieve through the constraint reduction would not be impressive.

Next we consider how that cost can be reduced to $O(|Q_3|n^2)$ when the structure of $\hat{\mathbf{A}}$ is considered. We define three index sets related to $\hat{Q}$:

$$\begin{aligned} Q_1 &:= \hat{Q} \cap M, \\ Q_2 &:= \{i > 0 : m + i \in \hat{Q}\}, \text{ and} \\ Q_3 &:= Q_1 \cap Q_2. \end{aligned} \tag{3.15}$$

The indices of the nearly active constraints in (3.2) are contained in Q_1. The indices of the nearly active constraints in (3.3) are contained in Q_2. From (3.4) we see that the last m rows of $\hat{\mathbf{H}}$ are zero. Therefore, in order to have $\text{rank}([\hat{\mathbf{H}},\ \hat{\mathbf{A}}_{\hat{Q}}^T]) = m + n$ as required, for each value of i we must include in $[\hat{\mathbf{H}},\ \hat{\mathbf{A}}_{\hat{Q}}^T]$ either the ith column of $\hat{\mathbf{A}}^T$ or the $(i + m)$th column. Therefore, for each i, $i \in Q_1 \cup Q_2$, so $Q_1 \cup Q_2 = M$. Therefore,

$$|Q_3| = |Q_1 \cap Q_2| = (|Q_1| + |Q_2|) - |Q_1 \cup Q_2| = |\hat{Q}| - m. \tag{3.16}$$

Using this partitioning in applying a sequence of block eliminations to the Newton system

$$\hat{\mathbf{H}}\Delta\mathbf{z} - \hat{\mathbf{A}}_{\hat{Q}}^T \Delta\mathbf{\phi}_{\hat{Q}} = -(\hat{\mathbf{H}}\mathbf{z} + \hat{\mathbf{c}} - \hat{\mathbf{A}}_{\hat{Q}}^T \mathbf{\phi}_{\hat{Q}}), \tag{3.17}$$

$$\hat{\mathbf{A}}_{\hat{Q}}\Delta\mathbf{z} - \Delta\mathbf{t}_{\hat{Q}} = \mathbf{0}, \tag{3.18}$$

$$\mathbf{T}_{\hat{Q}^2}\Delta\mathbf{\phi}_{\hat{Q}} + \mathbf{\Phi}_{\hat{Q}^2}\Delta\mathbf{t}_{\hat{Q}} = -\mathbf{T}_{\hat{Q}^2}\mathbf{\phi}_{\hat{Q}}, \tag{3.19}$$

corresponding to the reduced version of (3.5)-(3.7), leads to the reduced normal equations

$$\begin{aligned} &\left(\mathbf{H} + \mathbf{A}_{Q_1}^T \mathbf{I}_{Q_1} \Theta\ \mathbf{I}_{Q_1}^T \mathbf{A}_{Q_1}\right) \Delta\mathbf{x} \\ &\qquad = -\mathbf{H}\mathbf{x} - \mathbf{c} + \mathbf{A}_{Q_1}^T \mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} (\mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} + (\mathbf{I}_{Q_2}^T \mathbf{W}_{Q_2^2}^{-1} \mathbf{\Pi}_{Q_2^2} \mathbf{I}_{Q_2})_{Q_1^2})^{-1} \mathbf{d}_{Q_1}, \end{aligned} \tag{3.20}$$

where Θ is a diagonal $m \times m$ matrix given by

$$\mathbf{\Theta} := \mathbf{I}_{Q_1}^T \left(\mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} - \mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} \left(\mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} + (\mathbf{I}_{Q_2}^T \mathbf{W}_{Q_2^2}^{-1} \mathbf{\Pi}_{Q_2^2} \mathbf{I}_{Q_2})_{Q_1^2} \right)^{-1} \mathbf{S}_{Q_1^2}^{-1} \mathbf{\Lambda}_{Q_1^2} \right) \mathbf{I}_{Q_1}.$$

Simple algebra shows that

$$\mathbf{\Theta} = \mathbf{I}_{Q_3}^T \left(\mathbf{S}\mathbf{\Lambda}^{-1} + \mathbf{W}\mathbf{\Pi}^{-1} \right)_{Q_3}^{-1} \mathbf{I}_{Q_3},$$

in particular, that all entries of Θ vanish except for the diagonal entries with index in Q_3. Accordingly, (3.20) reduces to

$$\begin{aligned}\left(\mathbf{H}+\mathbf{A}_{Q_3}^T\left(\mathbf{S}\mathbf{\Lambda}^{-1}+\mathbf{W}\mathbf{\Pi}^{-1}\right)_{Q_3^2}^{-1}\mathbf{A}_{Q_3}\right)\Delta\mathbf{x}\\ =-\mathbf{H}\mathbf{x}-\mathbf{c}+\mathbf{A}_{Q_1}^T\mathbf{S}_{Q_1^2}^{-1}\mathbf{\Lambda}_{Q_1^2}(\mathbf{S}_{Q_1^2}^{-1}\mathbf{\Lambda}_{Q_1^2}+(\mathbf{I}_{Q_2}^T\mathbf{W}_{Q_2^2}^{-1}\mathbf{\Pi}_{Q_2^2}\mathbf{I}_{Q_2})_{Q_1^2})^{-1}\mathbf{d}_{Q_1}.\end{aligned} \tag{3.21}$$

to be solved for $\Delta\mathbf{x}$. We then get $\Delta\mathbf{y}$ via

$$\Delta\mathbf{y}=-(\mathbf{I}_{Q_1}^T\mathbf{S}_{Q_1^2}^{-1}\mathbf{\Lambda}_{Q_1^2}\mathbf{I}_{Q_1}+\mathbf{I}_{Q_2}^T\mathbf{W}_{Q_2^2}^{-1}\mathbf{\Pi}_{Q_2^2}\mathbf{I}_{Q_2})^{-1}(\mathbf{d}+\mathbf{I}_{Q_1}^T\mathbf{S}_{Q_1^2}^{-1}\mathbf{\Lambda}_{Q_1^2}\mathbf{A}_{Q_1}\Delta\mathbf{x}), \tag{3.22}$$

in which only diagonal matrices are inverted. To obtain the other variables we simply use (3.10) and (3.11).

Now (3.21) is uniquely solvable if and only if $\operatorname{rank}[\mathbf{H},\mathbf{A}_{Q_3}^T]=n$, and the inverse in the right-hand side of (3.22) is well-defined if and only if $Q_1\cup Q_2=M$. In other words, $\Delta\mathbf{x}$ and $\Delta\mathbf{y}$ are uniquely determined if and only if $\operatorname{rank}([\hat{\mathbf{H}},\ \hat{\mathbf{A}}_{\hat{Q}}^T])=m+n$, guaranteed by our definition of $\hat{Q}$.

In terms of computational cost, without constraint reduction (i.e., $Q_1=Q_2=Q_3=M$), forming the "normal" matrix in the left-hand side of (3.21) dominates: it takes $O(mn^2)$ floating point operations. Forming the right-hand side of (3.21) takes $O(mn)$ operations, and so does forming the right-handside of (3.22). With constraint reduction, the cost of forming the normal matrix is reduced to $O(|Q_3|n^2)$ operations. Since $|Q_3|$ can be selected as small as $n-\operatorname{rank}(\mathbf{H})$, the savings are the same as in the "feasible" case of section 2 (see subsection 2.3.3), in spite of the fact that at least $\hat{q}\geq m+n$ indexes must be included in $\hat{Q}$.

So we can define a constraint-reduced affine-scaling primal-dual interior-point algorithm for the extended problem (3.1)-(3.3) by using $\hat{Q}$, Q_1, Q_2 and Q_3 as defined in (3.13) and (3.15); by solving (3.21), (3.22), (3.10), and (3.11); and by substituting $\mathbf{z}$ for $\mathbf{x}$, $\mathbf{t}$ for $\mathbf{s}$, and $\boldsymbol{\phi}$ for $\boldsymbol{\lambda}$ in Algorithm 1, noticing that $\mathbf{w}$ and $\Delta\mathbf{w}$ are always the same as $\mathbf{y}$ and $\Delta\mathbf{y}$.

Since extending Algorithm 1 is very straightforward, we only discuss how to choose $\hat{Q}$. At each iteration we choose the set as follows. If every component of $\mathbf{d}$ is sufficiently large, all of the constraints (3.3) will be active at the solution, so we should define $Q_2:=M$. We choose Q_1' to include the q smallest s_i values, where the definition of q exactly follows (2.15) with $\mu:=\frac{\mathbf{s}^T\boldsymbol{\lambda}+\mathbf{w}^T\boldsymbol{\pi}}{2m}$. In addition to these basic indices required for the convergence, inspired by [14], we then choose Q_3' to contain the q largest $\bar{\theta}_i:=(s_i/\lambda_i+w_i/\pi_i)^{-1}$ values to improve performance. Finally $\hat{Q}:=Q_1'\cup Q_3'\cup\{m+1,\ldots,2m\}$.

3.2. Convergence Analysis for Extended Problems. By imposing Assumptions 2.1-2.6 on the standard form converted from (3.1)-(3.3), we can extend the convergence analysis to the algorithm for the extended problem. Assumption 2.2 is not necessary because problem (3.1)-(3.3) is always strictly feasible.

Define

$$\tilde{\mathcal{F}}_P:=\{(\mathbf{x},\mathbf{y})\in\mathbb{R}^{n+m}:\mathbf{A}\mathbf{x}+\mathbf{y}\geq\mathbf{b}\text{ and }\mathbf{y}\geq\mathbf{0}\}, \tag{3.23}$$

$$\tilde{\mathcal{F}}_P^o:=\{(\mathbf{x},\mathbf{y})\in\mathbb{R}^{n+m}:\mathbf{A}\mathbf{x}+\mathbf{y}>\mathbf{b}\text{ and }\mathbf{y}>\mathbf{0}\},\text{ and} \tag{3.24}$$

$$\tilde{\mathcal{F}}_P^*:=\{(\mathbf{x}^*,\mathbf{y}^*)\in\tilde{\mathcal{F}}_P:f_E(\mathbf{x}^*,\mathbf{y}^*)\leq f_E(\mathbf{x},\mathbf{y}),\ \forall(\mathbf{x},\mathbf{y})\in\tilde{\mathcal{F}}_P\}. \tag{3.25}$$

We define $\hat{\mathcal{A}}(\mathbf{z}) \subseteq \{1, \ldots, 2m\}$, the index set of active constraints at $\mathbf{z} := [\mathbf{x}^T,\ \mathbf{y}^T]^T$, similarly to (2.1). We also define its byproducts

$$\begin{aligned} \mathcal{A}_1(\mathbf{x}, \mathbf{y}) &:= \hat{\mathcal{A}}(\mathbf{z}) \cap M, \\ \mathcal{A}_2(\mathbf{x}, \mathbf{y}) &:= \{i > 0 : m + i \in \hat{\mathcal{A}}(\mathbf{z})\}, \text{ and} \\ \mathcal{A}_3(\mathbf{x}, \mathbf{y}) &:= \mathcal{A}_1(\mathbf{x}, \mathbf{y}) \cap \mathcal{A}_2(\mathbf{x}, \mathbf{y}). \end{aligned} \tag{3.26}$$

ASSUMPTION 3.1. *For all* $(\mathbf{x}, \mathbf{y})$ *in* $\tilde{\mathcal{F}}_P$, $\{\mathbf{a}_i^T : i \in \mathcal{A}_3(\mathbf{x}, \mathbf{y})\}$ *is a linearly independent set.*
The following proposition provides the connection between Assumption 2.4 (with $\mathbf{x}$, $\mathbf{a}_i$, $\mathcal{A}(\mathbf{x})$, $\mathcal{F}_P$ replaced by $\mathbf{z}$, $\hat{\mathbf{a}}_i$, $\hat{\mathcal{A}}(\mathbf{z})$, $\tilde{\mathcal{F}}_P$) and Assumption 3.1. The proof is routine and therefore omitted.

PROPOSITION 3.1. *Let* $\mathbf{z} := [\mathbf{x}^T,\ \mathbf{y}^T]^T$. *If* $\{\mathbf{a}_i^T : i \in \mathcal{A}_3(\mathbf{x}, \mathbf{y})\}$ *is a linearly independent set, then* $\{\hat{\mathbf{a}}_i^T : i \in \hat{\mathcal{A}}(\mathbf{z})\}$ *is a linearly independent set.*

THEOREM 3.2. *Let* $\{(\mathbf{x}^k, \mathbf{y}^k)\}$ *(or* $\{\mathbf{z}^k\}$*) be the sequence constructed by Algorithm 1 applied to the converted standard form. Under Assumptions 2.1, 2.3 (with* $\mathcal{F}_P^*$ *replaced by* $\tilde{\mathcal{F}}_P^*$*) and 3.1, the sequence* $(\mathbf{x}^k, \mathbf{y}^k)$ *converges to* $\tilde{\mathcal{F}}_P^*$.

Proof. Assumption 2.1 implies that $[\hat{\mathbf{H}},\ \hat{\mathbf{A}}^T]$ has full row rank. $\tilde{\mathcal{F}}_P^o$ is trivially nonempty. Due to Proposition 3.1, Assumption 3.1 implies that for all $\mathbf{z} \in \tilde{\mathcal{F}}_P$, $\{\hat{\mathbf{a}}_i^T : i \in \hat{\mathcal{A}}(\mathbf{z})\}$ is a linearly independent set. Hence Assumptions 2.1-2.4 hold with $\mathbf{H}$, $\mathbf{A}$, $\mathbf{a}_i$, $\mathbf{x}$, $\mathcal{A}(\mathbf{x})$, $\mathcal{F}_P$, $\mathcal{F}_P^o$, and $\mathcal{F}_P^*$ replaced by $\hat{\mathbf{H}}$, $\hat{\mathbf{A}}$, $\hat{\mathbf{a}}_i$, $\mathbf{z}$, $\hat{\mathcal{A}}(\mathbf{z})$, $\tilde{\mathcal{F}}_P$, $\tilde{\mathcal{F}}_P^o$, and $\tilde{\mathcal{F}}_P^*$. Since the set of index sets $\hat{\mathcal{Q}}(\hat{\mathbf{A}}\mathbf{z} - \hat{\mathbf{b}}, \hat{q})$ is consistent with $\mathcal{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}, q)$, the claim follows from Theorem 2.2. □

ASSUMPTION 3.2. *(Corresponds to Assumption 2.6) Strict complementarity holds at the optimal solution* $(\mathbf{x}^*, \mathbf{y}^*)$.

THEOREM 3.3. *Let* $\{(\mathbf{x}^k, \mathbf{y}^k)\}$ *(or* $\{\mathbf{z}^k\}$*) be the sequence constructed by Algorithm 1 applied to the converted standard form. Suppose that Assumptions 2.1, 2.3 (with* $\mathcal{F}_P^*$ *replaced by* $\tilde{\mathcal{F}}_P^*$*), 3.1, 2.5 (with* $\mathcal{F}_P^*$ *replaced by* $\tilde{\mathcal{F}}_P^*$*), and 3.2 hold. If* $\lambda_i < \phi_{\max}$ *and* $\pi_i < \phi_{\max}$ *for all* $i \in M$*, then the sequence* $\{(\mathbf{x}^k, \mathbf{y}^k, \boldsymbol{\lambda}^k, \boldsymbol{\pi}^k)\}$ *(or* $\{(\mathbf{z}^k, \boldsymbol{\phi}^k)\}$*) converges to the primal-dual solution pair* $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\lambda}^*, \boldsymbol{\pi}^*)$ *(or* $(\mathbf{z}^*, \boldsymbol{\phi}^*)$*) q-quadratically.*

Proof. Assumptions 2.1-2.6 hold with $\mathbf{H}$, $\mathbf{A}$, $\mathbf{a}_i$, $\mathbf{b}$, $\mathbf{x}$, $\mathcal{A}(\mathbf{x})$, $\mathcal{F}_P$, $\mathcal{F}_P^o$, $\mathcal{F}_P^*$, $\boldsymbol{\lambda}^*$, $\mathbf{s}^*$, $\mathbf{x}^*$, and M replaced by $\hat{\mathbf{H}}$, $\hat{\mathbf{A}}$, $\hat{\mathbf{a}}_i$, $\hat{\mathbf{b}}$, $\mathbf{z}$, $\hat{\mathcal{A}}(\mathbf{z})$, $\tilde{\mathcal{F}}_P$, $\tilde{\mathcal{F}}_P^o$, $\tilde{\mathcal{F}}_P^*$, $\boldsymbol{\phi}^*$, $\mathbf{t}^*$, $\mathbf{z}^*$, and $\{1, \ldots, 2m\}$. Then the claim follows directly from Theorem 2.3. □

It is possible to extend the convergence results to problems that also include an ℓ_2 penalty term in the objective function. A term $\mathbf{y}^T \operatorname{diag}(\mathbf{g})\mathbf{y}$ would then be added in the objective function (3.1), where $\mathbf{g} \in \mathbb{R}^m$, $\mathbf{d} \geq \mathbf{0}$, $\mathbf{g} \geq \mathbf{0}$, and $\mathbf{d} + \mathbf{g} > \mathbf{0}$. The same index set choice as defined in (3.13) can be used. By following the same steps, a constraint reduced algorithm can be devised. Problems of this type include support vector machine training with ℓ_2 penalty [21, 7].

4. Numerical Results. We implemented Algorithm 1 and its extended version in MATLAB R2009a with a dense direct solver for the normal equations in order to concentrate on the action of the reduction. The algorithms were tested on a machine with an AMD Atholon 64 X2 4600+(2.4 GHz) dual core processor with 2×64 KB L1 cache, 2×512 KB L2 cache, and 2.5 GB DDR2-400MHz configured as dual channel. The machine ran Windows 7 64 bit edition.

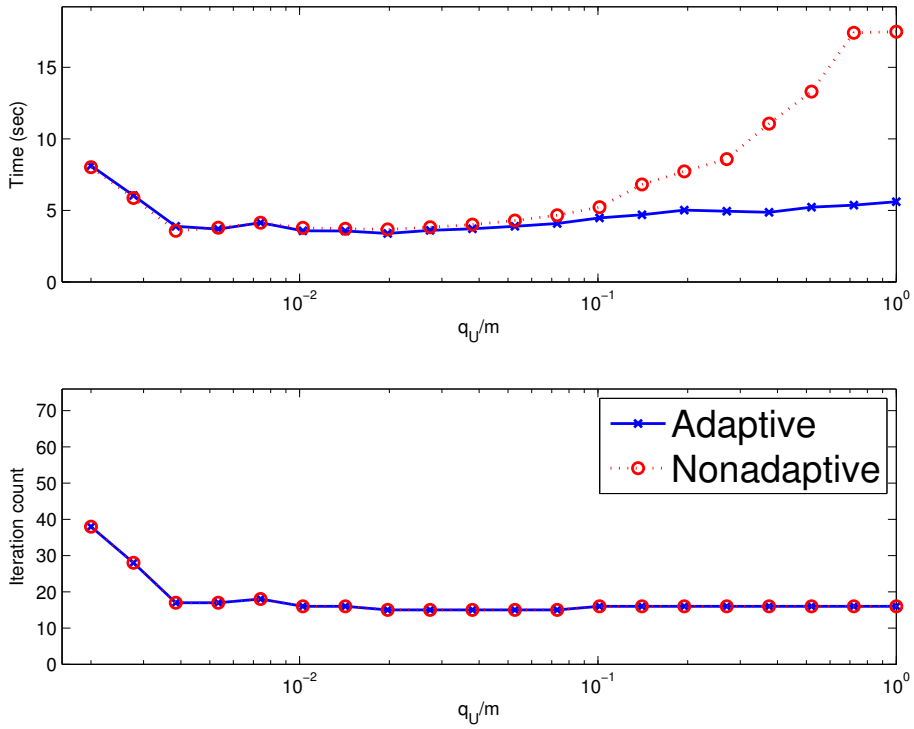


Fig. 4.1: Adaptive reduction is compared with nonadaptive reduction on the fully random problem (4.2). The rightmost red circle corresponds to (unreduced) affine scaling.

We set the algorithms to terminate when either convergence was detected or more than 200 iterations were performed. We set $\underline{\lambda} := 10^{-10}$, $\lambda_{\max} := 10^{30}$, $\eta := .98$, and $tol := 10^{-8}$. We set the adaptation parameter β (see (2.15)) to $1/4$. We also tested a non-adaptive version of our scheme by setting β to zero, resulting in $q = q_U$. When $q_U = m$, the latter becomes the "unreduced" affine scaling algorithm of [1]. We varied q_U to see how our algorithm would behave depending on it.

We discuss our results on three types of problems: random, data fitting, and semi-infinite quadratic optimization problems.

4.1. Random Problems. We compared the adaptive reduction with the nonadaptive reduction on random problems of size $m = 50000$ and $n = 100$. We generated $\mathbf{A}$ and $\mathbf{c}$ by taking random numbers drawn from a $N(0, 1)$ distribution. Diagonal components of $\mathbf{H}$ are taken from $U(0, 1)$, uniformly distributed random numbers in $(0, 1)$. We set $\mathbf{s}^0$ by taking numbers from $U(1, 2)$ and $\mathbf{x}^0$ from $U(0, 1)$. We set $\mathbf{b} := \mathbf{A}\mathbf{x}^0 - \mathbf{s}^0$. This is a slight modification of the random problem in [23], which uses different ranges of initial $\mathbf{s}^0$.

Timing and iteration counts are presented in Figure 4.1 with the horizontal axis in log scale. When q_U is very small (less than or equal to $10^{-1}m$), adaptive shrinking of q takes place only for a few of the final iterations after the iterate is close enough to the solution. This prevents the

adaptive reduction scheme from showing advantages for small q_U. It is noticeable that, for a wide range of q_U, the iteration count of both adaptive and nonadaptive reduction is near constant. In the same range of q_U, the timing of the adaptive reduction is near constant, while that of nonadaptive reduction decreases as q_U decreases.

4.2. Data Fitting. Data fitting is a problem of finding a model approximating time series data $\bar{b}_1, \ldots, \bar{b}_{\bar{m}}$ measured at times $\bar{t}_1, \ldots, \bar{t}_{\bar{m}}$. We build a model with a set of basis functions $\psi_1(\bar{t}), \ldots, \psi_{\bar{n}}(\bar{t})$:

$$u(\bar{t}) := \sum_{j=1}^{\bar{n}} \bar{x}_j \psi_j(\bar{t}).$$

To find good coefficients $\bar{x}_1, \ldots, \bar{x}_n$, we can use Chebyshev approximation, minimizing the maximal error of the model [2, 27]. Let $\bar{\mathbf{A}}$ be the $\bar{m}n \times \bar{n}$ matrix with entries $\bar{a}_{kj} = \psi_j(\bar{t}_k)$. If we form a vector $\bar{\mathbf{b}}$ from the values $\bar{b}_k$ and a vector $\bar{\mathbf{x}}$ from the coefficients $\bar{x}_j$, then the maximal error, to be minimized over all choices of $\bar{\mathbf{x}}$, can be written as

$$\max_{k \in \{1,\ldots,\bar{m}\}} \left|\bar{b}_k - u(\bar{t}_k)\right| = \left\|\bar{\mathbf{b}} - \bar{\mathbf{A}}\bar{\mathbf{x}}\right\|_\infty .$$

The solution to this *min-max* problem can be very sensitive to noise in the measurements. To reduce the sensitivity, we use a regularization method, first introduced by Tikhonov; see, e.g., [22]. We instead solve the regularized min-max problem

$$\min_{\bar{\mathbf{x}}} \|\bar{\mathbf{b}} - \bar{\mathbf{A}}\bar{\mathbf{x}}\|_\infty + \frac{1}{2}\alpha\|\bar{\mathbf{x}}\|^2_{\bar{\mathbf{H}}}, \tag{4.1}$$

where $\bar{\mathbf{H}}$ is an $\bar{n} \times \bar{n}$ symmetric positive semidefinite matrix, $\|\bar{\mathbf{x}}\|_{\bar{\mathbf{H}}} := \sqrt{\bar{\mathbf{x}}^T\bar{\mathbf{H}}\bar{\mathbf{x}}}$ and α is a positive value. This can be written as

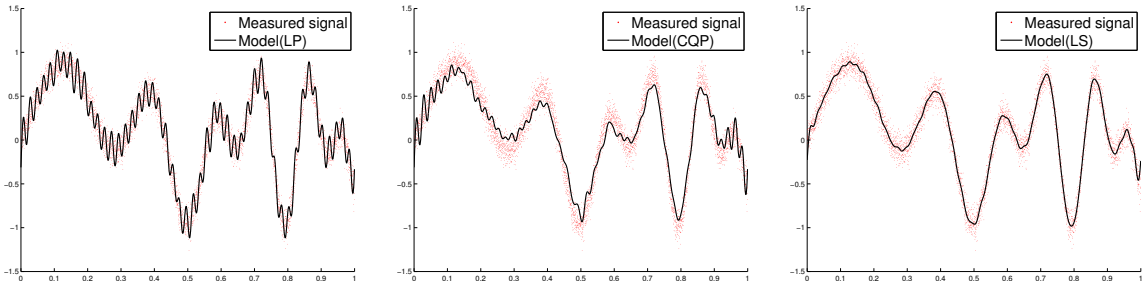
$$\begin{aligned} \min_{\bar{\mathbf{x}},\bar{t}} \ & \bar{t} + \frac{1}{2}\alpha\|\bar{\mathbf{x}}\|^2_{\bar{\mathbf{H}}} \\ \text{s.t. } & \bar{\mathbf{A}}\bar{\mathbf{x}} - \bar{\mathbf{b}} \geq -\bar{t}\mathbf{e} \\ & -\bar{\mathbf{A}}\bar{\mathbf{x}} + \bar{\mathbf{b}} \geq -\bar{t}\mathbf{e}, \end{aligned} \tag{4.2}$$

which is an instance of the standard form (1.1), with $\mathbf{x} := [\bar{\mathbf{x}}^T, \bar{t}]^T$, $\mathbf{b} := [\bar{\mathbf{b}}^T, -\bar{\mathbf{b}}^T]^T$, $\mathbf{c} := [0, \ldots, 0, 1]^T$, $m := 2\bar{m}$, $n := \bar{n} + 1$,

$$\mathbf{A} := \begin{bmatrix} \bar{\mathbf{A}} & \mathbf{e} \\ -\bar{\mathbf{A}} & \mathbf{e} \end{bmatrix} \text{ and } \mathbf{H} := \begin{bmatrix} \alpha\bar{\mathbf{H}} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}.$$

We used the problem setting of [28]. For convenience, we restate it here. For basis functions, we used cosine and sine functions:

$$\begin{aligned} &\psi_i(\bar{t}) = \cos(2(i-1)\pi\bar{t}) \text{ for } i = 1, \ldots, \bar{n}+1, \text{ and} \\ &\psi_i(\bar{t}) = \sin(2(i-\bar{n}-1)\pi\bar{t}), \text{ for } i = \bar{n}+2, \ldots, 2\bar{n}+1. \end{aligned}$$



(a) Model is generated from an LP problem by setting $\alpha = 0$ in (4.2).

(b) Model is generated from the CQP problem (4.2) with the diagonal components of $\bar{\mathbf{H}}$ proportional to the frequency of the corresponding basis function.

(c) Model is generated from a least squares (LS) problem, $\min_{\bar{\mathbf{x}}} \|\bar{\mathbf{b}} - \bar{\mathbf{A}}\bar{\mathbf{x}}\|_2^2$.

Fig. 4.2: Measured signal fit by various methods.

The sampling points were, for $i = 1, \ldots, \bar{m}$, $\bar{t}_i := (i-1)/\bar{m}$. For observed data, we used the signal function

$$g(\bar{t}) := \sin(10\bar{t})\cos(25\bar{t}^2),$$

and set

$$\bar{b}_i := g(\bar{t}_i) + \epsilon_i, \text{ for } i = 1, \ldots, \bar{m},$$

where $\epsilon_i \sim N(0, .09)$ denotes independent random noise following a normal distribution with 0 mean and 0.09 variance.

Figure 4.2 compares models obtained from linear programming ((4.2) using $\alpha := 0$), the regularized min-max ((4.2) with $\alpha := 10^{-6}$ and $\bar{\mathbf{H}}$ defined below), and least squares ((4.1) with $\alpha = 0$ and 2-norm instead of infinity norm) problems. Without the regularization term, the model obtained from (4.2) tends to be too oscillatory as seen in Figure 4.2a. This tendency is caused by giving too much weight to basis functions with high frequency. To suppress high frequency components, in the CQP model (4.2) (Figure 4.2b), we used a penalty weight proportional to the frequency of the basis functions. We let $\bar{\mathbf{H}}$ be a diagonal matrix with $\bar{h}_{jj} := 2(j-1)\pi$ for $j = 1, \ldots, \bar{n}+1$, and $\bar{h}_{jj} := 2(j-\bar{n}-1)\pi$ for $j = \bar{n}+2, \ldots, 2\bar{n}+1$. Further, we set $\alpha := 10^{-6}$ and $\bar{n} := 99$, resulting in $n = 200$.

For the numerical testing of our adaptive reduction and nonadaptive reduction schemes, we set $\bar{m} := 20000$ ($m = 40000$). For a strictly feasible initial point, we used $\bar{\mathbf{x}}_0 := \mathbf{0}$ and $\bar{t}_0 := \|\bar{\mathbf{b}}\|_\infty + 1$, and we used that same α and $\bar{\mathbf{H}}$ as above. Timing and iteration counts on varying q_U are presented in Figure 4.3. The nonadaptive algorithm with $q_U := m$ corresponds to a standard PDAS IPM algorithm. When q_U is small (less than $10^{-1}m$), adaptive shrinking of q does not take place until

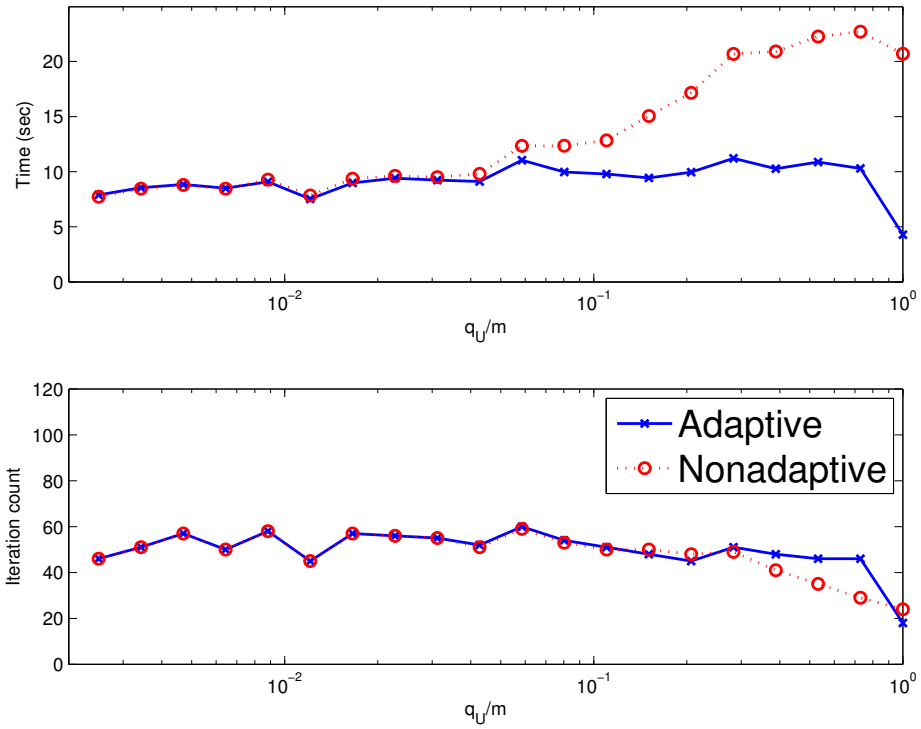


Fig. 4.3: Adaptive reduction is compared with nonadaptive reduction on the data-fitting problem (4.2). The rightmost red circle corresponds to (unreduced) affine scaling.

μ becomes sufficiently small. Up to this point, both algorithms compute the same primal-dual iterates. Once μ becomes sufficiently small, implying the iterate is close to the solution, shrinking the index set size does not affect the search direction as much as it does in early iterations, because the normal matrix is dominated by the diverging λ_i/s_i. On the other hand, when q_U is large, the adaptive algorithm may use fewer constraints than the nonadaptive even in early iterations. This affects the search direction at early iterations and may result in a different iteration count.

4.3. Semi-infinite Quadratic Optimization: Array Pattern Synthesis. In this section we experiment with the array pattern synthesis problem discussed in [19]. In this problem, $\hat{m}$ sensors are evenly distributed around a circle. We want to minimize the sidelobe energy of the array response by adjusting the weights $\tilde{w}_k$ for the sensors in the array. The problem is formulated as

$$\min_{\tilde{\mathbf{w}}} \ \frac{1}{2}\tilde{\mathbf{w}}^T\mathbf{Q}\tilde{\mathbf{w}}, \tag{4.3}$$

$$\mathbf{a}^T(\varphi,\theta)\tilde{\mathbf{w}} \le \sigma(\varphi), \ \text{ for } \varphi \in \Phi_s := [\varphi_s,\, 2\pi-\varphi_s] \text{ and } \theta \in [0, 2\pi), \tag{4.4}$$

$$\mathbf{P}\tilde{\mathbf{w}} = \mathbf{p}, \tag{4.5}$$

where

$$\mathbf{Q} = 2 \begin{bmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix}, \tag{4.6}$$

$r_{kl} = J_0(4\pi f_0 \frac{r}{c} \sin((k-l)\pi/\hat{m}))$ for $k, l = 0, \ldots, \hat{m}-1$, and J_0 is the Bessel function of the first kind and of order zero. The scalar parameter f_0 is the frequency of the wave, r is the radius of sensor array circle, and c is the speed of wave propagation. The parameter $\varphi_s \in [0,\, \pi]$ is the sidelobe angle.

The constraints are specified by

$$\mathbf{a}(\varphi, \theta) = \mathrm{Re}\left(e^{i\theta}\tilde{\mathbf{d}}(\varphi)\right), \tag{4.7}$$

$$\tilde{\mathbf{d}}(\varphi) = \begin{bmatrix} \mathbf{d}(\varphi) \\ -i\mathbf{d}(\varphi) \end{bmatrix}, \tag{4.8}$$

where $i = \sqrt{-1}$ and each component of $\mathbf{d}(\varphi)$ is defined as $d_m(\varphi) := e^{-i\mathbf{k}^T\mathbf{r}_m}$ for $m = 0, \ldots, \hat{m}-1$. The vectors $\mathbf{k}$ and $\mathbf{r}_m$ are defined as $\mathbf{k}^T = \frac{-2\pi f_0}{c}[\cos\varphi,\, \sin\varphi]$ and $\mathbf{r}_m^T := r[\cos\frac{2\pi m}{\hat{m}},\, \sin\frac{2\pi m}{\hat{m}}]$ for $m = 0, \ldots, \hat{m}-1$. Finally, $\mathbf{p} := [1, 0]^T$ and

$$\mathbf{P} := \begin{bmatrix} \mathrm{Re}(\tilde{\mathbf{d}}^T(0)) \\ \mathrm{Im}(\tilde{\mathbf{d}}^T(0)) \end{bmatrix} \tag{4.9}$$

To solve this problem, we discretize the functional constraint (4.4) for φ and θ. Then the inequality constraints (4.4) become

$$\mathbf{A}\tilde{\mathbf{w}} \leq \sigma\mathbf{e}. \tag{4.10}$$

Since our algorithm does not support equality constraints, we eliminate constraints (4.5). To do this, we first find $\tilde{\mathbf{w}}_0$ satisfying $\mathbf{P}\tilde{\mathbf{w}}_0 = \mathbf{p}$. (In the numerical tests, we used the least squares solution.) Secondly we find an orthonormal basis $\mathbf{Z}$ for the nullspace of $\mathbf{P}$ using QR factorization. Then all $\tilde{\mathbf{w}}$ that satisfy $\mathbf{P}\tilde{\mathbf{w}} = \mathbf{p}$ are of the form $\tilde{\mathbf{w}} = \tilde{\mathbf{w}}_0 + \mathbf{Z}\mathbf{v}$, where $\mathbf{v}$ is unconstrained. The inequality constraints (4.10) become

$$\mathbf{A}\mathbf{Z}\mathbf{v} \leq \sigma\mathbf{e} - \mathbf{A}\tilde{\mathbf{w}}_0. \tag{4.11}$$

In terms of $\mathbf{v}$ the objective function (4.3) becomes

$$\frac{1}{2}\mathbf{v}^T\mathbf{Z}^T\mathbf{Q}\mathbf{Z}\mathbf{v} + (\tilde{\mathbf{w}}_0^T\mathbf{Q}\mathbf{Z})\mathbf{v} + \frac{1}{2}\tilde{\mathbf{w}}_0^T\mathbf{Q}\tilde{\mathbf{w}}_0,$$

where the last constant term can be omitted.

After this modification, we make each row of $\mathbf{AZ}$ have length 1 with the right hand side scaled accordingly. So indeed, we solve

$$\begin{aligned} &\min_{\mathbf{v}} \ \frac{1}{2}\mathbf{v}^T\tilde{\mathbf{Q}}\mathbf{v} + \mathbf{c}^T\mathbf{v}, \\ &\tilde{\mathbf{A}}\mathbf{v} \geq \mathbf{b}, \end{aligned} \tag{4.12}$$

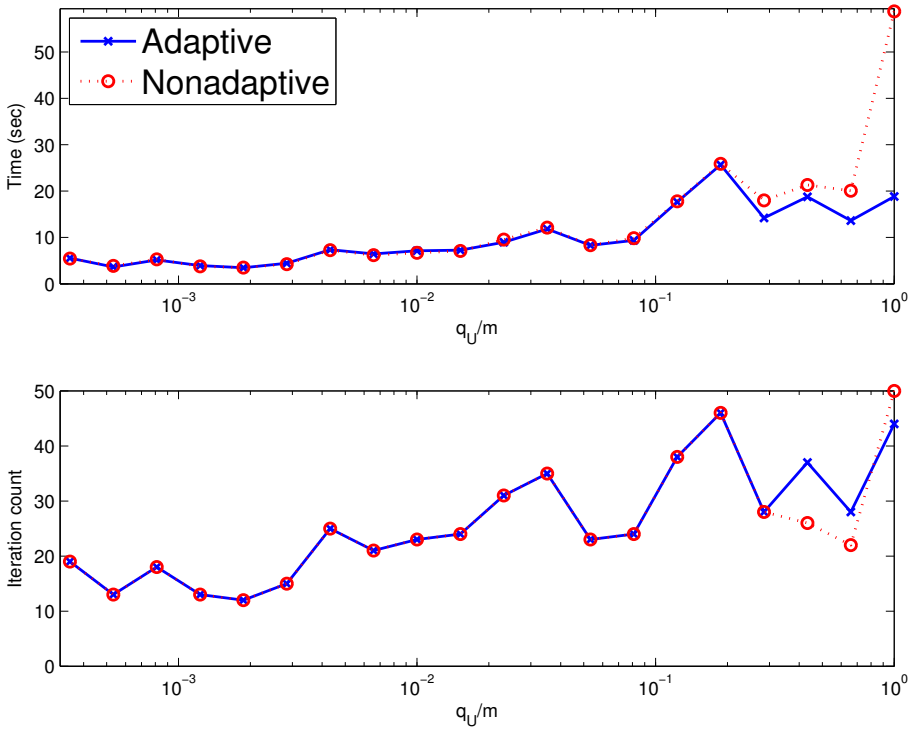


Fig. 4.4: Adaptive reduction is compared with nonadaptive reduction on the array pattern synthesis problem. The rightmost red circle corresponds to (unreduced) affine scaling.

where $\tilde{\mathbf{Q}} := \mathbf{Z}^T\mathbf{Q}\mathbf{Z}$, $\mathbf{c} := \tilde{\mathbf{w}}_0^T\mathbf{Q}\mathbf{Z}$, $\tilde{\mathbf{A}}$ is the normalized version of $-\mathbf{A}\mathbf{Z}$, and $\mathbf{b}$ is the negated right hand side of (4.11) scaled accordingly to the normalization.

Following [19], we set $\hat{m} = 20$, $\varphi_s = 30°$, $r = 1$, $f_0 = 50$, $c = 100$, and $\sigma = 10^{-17.5/20}$ (i.e., -17.5dB). We discretized φ and θ at one degree intervals, i.e., $\varphi_j = 30°, 31°, 32°, \ldots, 330°$ and $\theta_k = 0, 1°, 2°, \ldots, 359°$. Then each row of $\mathbf{A}$ becomes $\mathbf{a}_{360(j-1)+k}^T = \mathbf{a}(\varphi_j, \theta_k)^T$. Originally, the problem with this setting has $301 \times 360 = 108360$ inequality and 2 equality constraints with 40 variables. After the removal of the equality constraints, the number of variables reduces to 38.

Because no feasible initial point is readily available, we use the extension described in Section 3. The initial point for this problem is set as $\mathbf{v}_0 := [0, \ldots, 0]$ and $\mathbf{y}_0 := \max\{\mathbf{b}, \mathbf{0}\} + \mathbf{e}$. We set $\mathbf{d} := [100, \ldots, 100]^T$, $\boldsymbol{\pi}_0 := \mathbf{y}_0$, and $\boldsymbol{\lambda}_0 := \tilde{\mathbf{A}}\mathbf{v}_0 - \mathbf{b} + \mathbf{y}_0$.

The results are shown in Figure 4.4. It is worth noting that, in overall, the iteration count steadily decreases as q_U decreases.

4.4. Summary of Numerical Results. Figure 4.5 summarizes our results, comparing the time for the adaptive constraint-reduced algorithm, the algorithm without reduction, and MATLAB's quadprog. The scalar parameter q_U is set to m. Our algorithm has proved to be efficient

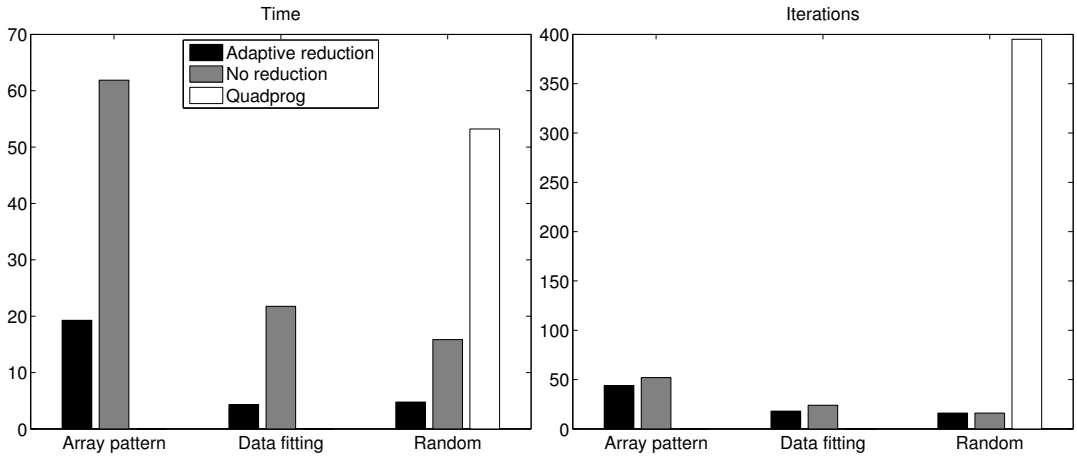


Fig. 4.5: Adaptive constraint-reduced algorithm is compared with non-reduced algorithm and MATLAB's quadprog (which uses an active set algorithm). The scalar parameter q_U is set to m, corresponding to the rightmost point in Figures 4.1, 4.3 and 4.4. quadprog failed to solve the data fitting and array pattern synthesis problems; the large-scale option in quadprog is not available for the problems we tested.

and reliable.

5. Conclusions. We proposed an affine-scaling algorithm which significantly reduces computational effort in solving convex quadratic programming problems having many more constraints than variables. We established global convergence and a quadratic local convergence rate, and proposed an extension to handle the case when a strictly feasible initial point is not readily available. We demonstrated algorithms' effectiveness when a direct solver is used for the normal equations. The constraint reduction also reduces computational time required for matrix-vector products in solving the normal equations, which would benefit iterative solvers such as the preconditioned conjugate gradient method [20, 26]. Thus the algorithm of [23] has been effectively extended from LP to CQP. We also demonstrated how the addition of adaptiveness to the constraint reduction can save computational time while keeping the algorithm quite stable (in solution time and convergence) over a wide range of the scalar parameter q_U.

We established global convergence and a quadratic local convergence rate. We showed how the method can be applied to the extended problems that explicitly include nonnegative slack variables such as the training of support vector machines with soft margins. In addition, we deduced the convergence properties of the algorithm for the extended problems from those of the standard one, thus simplifying the extension of other convergence-proven and possibly constraint-reduced IPMs to the extended problems.

Applying constraint reduction to competing IPM algorithms such as Mehrotra's predictor-corrector variants for general CQP with many inequality constraints is also possible. Introducing the reduction scheme to those competitors can also notably save computational time as demonstrated in [23, 28] and [14]. Establishing convergence properties of such algorithms is a topic for future research.

A. Convergence Proof for the Constraint-Reduced Affine-Scaling PDIPM. The following proofs for global convergence and the local rate of convergence are adapted from the proofs provided in [24] and [23]. Many parts are identical to [23] except for the action of the Hessian matrix.

A.1. Global Convergence Proof. Throughout this section, we use a superscript * to denote a limit point of a sequence, not necessarily the solution to (1.1). We will need the reduced Newton system, analogous to (2.6), that is satisfied by the components $(\Delta\mathbf{x}, \Delta\mathbf{s}_Q, \Delta\boldsymbol{\lambda}_Q)$ of the affine-scaling search direction (and from which the reduced normal equations are defined):

$$\mathbf{H}\Delta\mathbf{x} - \mathbf{A}_Q^T\Delta\boldsymbol{\lambda}_Q = -(\mathbf{H}\mathbf{x} + \mathbf{c} - \mathbf{A}_Q^T\boldsymbol{\lambda}_Q), \tag{A.1}$$

$$\mathbf{A}_Q\Delta\mathbf{x} - \Delta\mathbf{s}_Q = \mathbf{0}, \tag{A.2}$$

$$\mathbf{S}_{Q^2}\Delta\boldsymbol{\lambda}_Q + \boldsymbol{\Lambda}_{Q^2}\Delta\mathbf{s}_Q = -\mathbf{S}_{Q^2}\boldsymbol{\lambda}_Q, \tag{A.3}$$

and the Jacobian $\mathbf{J}$ and augmented Jacobian $\mathbf{J}_a$ matrices associated with the unreduced system (2.2)-(2.5)

$$\mathbf{J}(\mathbf{A},\mathbf{s},\boldsymbol{\lambda}) := \begin{bmatrix} \mathbf{H} & \mathbf{0} & -\mathbf{A}^T \\ \mathbf{A} & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda} & \mathbf{S} \end{bmatrix}, \text{ and } \mathbf{J}_a(\mathbf{A},\mathbf{s},\boldsymbol{\lambda}) := \begin{bmatrix} \mathbf{H} & -\mathbf{A}^T \\ \boldsymbol{\Lambda}\mathbf{A} & \mathbf{S} \end{bmatrix}. \tag{A.4}$$

LEMMA A.1. *(Corresponds to Lemma 1 of [23]) For* $\mathbf{s},\boldsymbol{\lambda} \geq \mathbf{0}$, $\mathbf{J}(\mathbf{A},\mathbf{s},\boldsymbol{\lambda})$ *is nonsingular if and only if*

(*i*) $\forall i \in M : s_i + \lambda_i > 0$,
(*ii*) *Rows of* $\mathbf{A}_{\{i:s_i=0\}}$ *are linearly independent,*
(*iii*) $\operatorname{rank}([\mathbf{H},\ \mathbf{A}_{\{i:\lambda_i\neq 0\}}]) = n$, *i.e.,*

$$\{\mathbf{x} : \mathbf{A}_{\{i:\lambda_i\neq 0\}}\mathbf{x} = \mathbf{0}\} \cap \{\mathbf{x} : \mathbf{H}\mathbf{x} = \mathbf{0}\} = \{\mathbf{0}\}. \tag{A.5}$$

Proof. See the proof of Lemma B.1 in [13]. □

For the following propositions, lemmas, corollaries, and theorems, unless explicitly stated, it is assumed that Assumptions 2.1, 2.2, 2.3, and 2.4 hold, although some of the earlier results do not require all of them.

PROPOSITION A.2. *(Corresponds to Proposition 3 of [23]) The points generated by the iteration*

of Algorithm 1 satisfy:

$$
\begin{aligned}
&(i)\ \Delta\mathbf{x}^k \neq \mathbf{0} \ \textit{iff}\ \mathbf{H}\mathbf{x}^k + \mathbf{c} \neq \mathbf{0},\\
&(ii)\ \alpha^k > 0,\\
&(iii)\ \mathbf{s}^{k+1} = \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b} > \mathbf{0}\ \textit{and}\ \mathbf{x}^{k+1} \in \mathcal{F}_P^o,\\
&(iv)\ \boldsymbol{\lambda}^{k+1} > 0.
\end{aligned}
$$

Proof. The first claim is a direct consequence of Lemma 2.1 and (2.12). The second and the third claims are true due to (2.8), (2.20), (2.21), and (2.22). The fourth is true due to (2.23); specifically, $\|\Delta\mathbf{x}^k\|^2 + \|\tilde{\boldsymbol{\lambda}}_-^k\|^2 > 0$, $\underline{\lambda} > 0$, $\lambda_{\max} > 0$, and $\tilde{\lambda}_i^k$ is taken only when $0 < \min\{\|\Delta\mathbf{x}^k\|^2 + \|\tilde{\boldsymbol{\lambda}}_-^k\|^2, \underline{\lambda}\} \leq \tilde{\lambda}_i^k \leq \lambda_{\max}$. □

From the initial point, which is strictly primal feasible with $\boldsymbol{\lambda}$ positive, the algorithm generates the next point which is also strictly primal feasible with $\boldsymbol{\lambda}$ positive. Hence the iteration can be repeated. In other words, the sequence generated by Algorithm 1 is well defined and valid so that the interior-point method generates an infinite sequence of points unless a point $\mathbf{x}^k$ such that $\mathbf{H}\mathbf{x}^k + \mathbf{c} = \mathbf{0}$ is found, in which case $\mathbf{x}^k$ is the solution of the unconstrained problem and also in $\mathcal{F}_P^o$. In the sequel it is assumed that Algorithm 1 generates an infinite sequence of primal-dual points.

LEMMA A.3. *For any $\mathbf{x} \in \mathcal{F}_P^o$, $q \geq n$, $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}, q)$ and for every $\tilde{\boldsymbol{\lambda}}$ and $\Delta\mathbf{x}$ generated by Algorithm 1, $\tilde{\boldsymbol{\lambda}}_Q^T \mathbf{A}_Q \Delta\mathbf{x} \leq 0$, where the equality holds only when $\tilde{\boldsymbol{\lambda}}_Q^T = \mathbf{0}$ and $\mathbf{A}_Q\Delta\mathbf{x} = \mathbf{0}$.*

Proof. From the definition of $\tilde{\boldsymbol{\lambda}}$ and (2.8), we know that

$$\tilde{\boldsymbol{\lambda}}_Q^T \mathbf{A}_Q \Delta\mathbf{x} = \tilde{\boldsymbol{\lambda}}_Q^T \Delta\mathbf{s}_Q.$$

From (2.9),

$$\tilde{\boldsymbol{\lambda}}_Q^T \Delta\mathbf{s}_Q = -\tilde{\boldsymbol{\lambda}}_Q^T \mathbf{S}_{Q^2} \boldsymbol{\Lambda}_{Q^2}^{-1} (\boldsymbol{\lambda}_Q + \Delta\boldsymbol{\lambda}_Q) = -\tilde{\boldsymbol{\lambda}}_Q^T \mathbf{S}_{Q^2} \boldsymbol{\Lambda}_{Q^2}^{-1} \tilde{\boldsymbol{\lambda}}_Q \leq 0.$$

since $\mathbf{S}_{Q^2}$ and $\boldsymbol{\Lambda}_{Q^2}$ are diagonal and positive definite. Since $\boldsymbol{\Lambda}_{Q^2}$ and $\mathbf{S}_{Q^2}$ are nonsingular, the equality holds only if $\tilde{\boldsymbol{\lambda}}_Q = \mathbf{0}$. In view of (2.9), $\Delta\mathbf{s}_Q = \mathbf{0}$ if and only if $\tilde{\boldsymbol{\lambda}}_Q = \mathbf{0}$. Therefore, from (2.8), we conclude that $\mathbf{A}_Q\Delta\mathbf{x} = \mathbf{0}$ if and only if $\tilde{\boldsymbol{\lambda}}_Q = \mathbf{0}$. □

As a first step in the proof of global convergence, we show that the objective function decreases monotonically on the sequence of points generated by Algorithm 1.

PROPOSITION A.4. *(Corresponds to Lemma 4 of [23]) If $\Delta\mathbf{x} \neq \mathbf{0}$, then*

(i) $f(\mathbf{x} + \alpha\Delta\mathbf{x}) < f(\mathbf{x})$ *for all* $\alpha \in (0, 2)$,
(ii) $\frac{\mathrm{d}}{\mathrm{d}\alpha} f(\mathbf{x} + \alpha\Delta\mathbf{x}) < 0$ *for all* $0 \leq \alpha < 1$,
(iii) $f(\mathbf{x} + \alpha\Delta\mathbf{x}) < f(\mathbf{x} + \underline{\alpha}\Delta\mathbf{x})$ *for all* α *and* $\underline{\alpha}$ *such that* $0 \leq \underline{\alpha} < \alpha \leq 1$.

Proof. Since f is quadratic, $f(\mathbf{x} + \alpha\Delta\mathbf{x})$ can be exactly expressed by the second order Taylor

expansion

$$\begin{aligned} f(\mathbf{x}+\alpha\Delta\mathbf{x}) &= f(\mathbf{x}) + \alpha\nabla f(\mathbf{x})^T\Delta\mathbf{x} + \frac{1}{2}\alpha^2\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} \\ &= f(\mathbf{x}) + \alpha\Delta\mathbf{x}^T\left(\mathbf{H}\mathbf{x}+\mathbf{c}\right) + \frac{1}{2}\alpha^2\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} \\ &= f(\mathbf{x}) + \alpha\Delta\mathbf{x}^T\left(-\mathbf{H}\Delta\mathbf{x} + \mathbf{A}_Q^T\tilde{\boldsymbol{\lambda}}_Q\right) + \frac{1}{2}\alpha^2\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} \text{ (by } (A.1)) \\ &= f(\mathbf{x}) - \alpha(1-\frac{1}{2}\alpha)\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} + \alpha\Delta\mathbf{x}^T\mathbf{A}_Q^T\tilde{\boldsymbol{\lambda}}_Q. \end{aligned} \tag{A.6}$$

By Lemma A.3, $\Delta\mathbf{x}^T\mathbf{A}_Q^T\tilde{\boldsymbol{\lambda}}_Q$ is nonpositive and, since $\mathbf{H}$ is positive semidefinite, so is $-\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x}$. By Assumption 2.1 and Lemma A.3, they cannot be zero at the same time unless $\Delta\mathbf{x}=\mathbf{0}$. Since α and $1-\frac{\alpha}{2}$ are both positive when $\alpha\in(0,2)$, the first claim holds.

Now let us consider the second claim. From (A.6) we derive

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}f(\mathbf{x}+\alpha\Delta\mathbf{x}) = -(1-\alpha)\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} + \tilde{\boldsymbol{\lambda}}_Q^T\mathbf{A}_Q\Delta\mathbf{x}. \tag{A.7}$$

Since $-(1-\alpha)<0$ for all $0\le\alpha<1$, the claim does hold.

Let us consider the third claim. Since $\frac{\mathrm{d}}{\mathrm{d}\alpha}f(\mathbf{x}+\alpha\Delta\mathbf{x})<0$ for $0\le\alpha<1$, $f(\mathbf{x}+\alpha\Delta\mathbf{x})$ strictly decreases with respect to $\alpha\in[0,1]$. Then the claim immediately follows. □

COROLLARY A.5. *(Corresponds to Corollary of Proposition 3.1 [24]) The sequence* $\{\mathbf{x}^k\}$ *is bounded.*

Proof. Since, in view of Assumption 2.3, $\mathcal{F}_P^*$ is bounded, the level set $\{\mathbf{x}\in\mathcal{F}_P : f(\mathbf{x})<f(\mathbf{x}^0)\}$ is bounded [13]. By Proposition A.4, $f(\mathbf{x}^k)$ decreases monotonically, so the claim holds. □

A point $\mathbf{x}$ is said to be *stationary* for (1.1) if it satisfies the KKT conditions without nonnegativity constraints on $\boldsymbol{\lambda}$, i.e.,

$$\mathbf{H}\mathbf{x}+\mathbf{c}-\mathbf{A}^T\boldsymbol{\lambda}=\mathbf{0}, \tag{A.8}$$

$$\mathbf{A}\mathbf{x}-\mathbf{b}-\mathbf{s}=\mathbf{0}, \tag{A.9}$$

$$\mathbf{S}\boldsymbol{\lambda}=\mathbf{0}, \tag{A.10}$$

$$\mathbf{s}\ge\mathbf{0}. \tag{A.11}$$

A stationary point $\mathbf{x}$ is a solution to (1.1) if all the components of its associated multiplier vector $\boldsymbol{\lambda}$ are nonnegative.

We proceed by showing that the sequence generated by Algorithm 1 approaches the set of stationary points. In the following lemma, it will be shown that, if the sequence converges to some point, the limit point is stationary and the sequence of modified Newton steps $\{\Delta\mathbf{x}^k\}$ converges to $\mathbf{0}$.

LEMMA A.6. *(Corresponds to Lemma 3.5 of [28]) Suppose that* $\{\mathbf{x}^k\}\to\mathbf{x}^*$ *on an infinite index set* K *and* $q^k\ge n$. *Then there exists* k' *such that* $\mathcal{A}(\mathbf{x}^*)\subseteq Q$ *for all* $Q\in\mathcal{Q}(\mathbf{A}\mathbf{x}^k-\mathbf{b},q^k)$ *for all* $k\in K$ *and* $k>k'$.

Proof. Under Assumption 2.4, it follows that $|\mathcal{A}(\mathbf{x}^*)|\le n$. Since $q^k\ge n$ and $\{\mathbf{x}^k\}\to\mathbf{x}^*$ on K, the claim does hold by the definition of $\mathcal{Q}(\mathbf{A}\mathbf{x}^k-\mathbf{b},q^k)$ (2.14). □

In *Step 2* of Algorithm 1, by replacing $\Delta\mathbf{s}$ in (2.9) with (2.8), we can rewrite (2.9) as

$$\tilde{\boldsymbol{\lambda}}^k = -(\mathbf{S}^k)^{-1}\boldsymbol{\Lambda}^k\mathbf{A}\Delta\mathbf{x}^k, \tag{A.12}$$

or equivalently

$$\tilde{\lambda}_i^k = -\frac{\lambda_i^k}{s_i^k}\mathbf{a}_i^T\Delta\mathbf{x}^k. \tag{A.13}$$

We use this modified form in the following lemmas.

LEMMA A.7. *(Corresponds to Lemma 6 of [23]) Suppose* $\{\mathbf{x}^k\}$ *converges to some point* $\mathbf{x}^*$ *on an infinite index set* K*. If* $\{\Delta\mathbf{x}^k\}$ *converges to zero on* K*, then* $\mathbf{x}^*$ *is stationary and* $\{\tilde{\boldsymbol{\lambda}}^k\}$ *converges on* K *to* $\boldsymbol{\lambda}^*$*, which is the unique multiplier associated with* $\mathbf{x}^*$*.*

Proof. Suppose $\{\Delta\mathbf{x}^k\} \to \mathbf{0}$ as $k \to \infty$, $k \in K$. Since $\{\boldsymbol{\lambda}^k\}$ is bounded by construction of Algorithm 1 and $\{s_i^k\}$ is bounded away from 0 for $i \notin \mathcal{A}(\mathbf{x}^*)$, it follows from (A.13) that

$$\forall i \in \mathcal{A}(\mathbf{x}^*)^{\mathrm{c}}, \quad \{\tilde{\lambda}_i^k\} \to 0, \text{ as } k \to \infty,\, k \in K. \tag{A.14}$$

We have shown the convergence of $\boldsymbol{\lambda}^k_{\mathcal{A}(\mathbf{x}^*)^{\mathrm{c}}}$ on K so far.

Now we need to show the convergence of $\boldsymbol{\lambda}_{\mathcal{A}(\mathbf{x}^*)}$. At iteration k, the system of equations (A.1) can be written as

$$\mathbf{H}\mathbf{x}^k + \mathbf{c} - \mathbf{A}_{Q^k}^T\tilde{\boldsymbol{\lambda}}^k_{Q^k} = -\mathbf{H}\Delta\mathbf{x}^k. \tag{A.15}$$

By Lemma A.6, there exists k' such that $\mathcal{A}(\mathbf{x}^*) \subseteq Q^k$ for $k > k'$ and $k \in K$. Then, since $\{\mathbf{x}^k\}$ converges to $\mathbf{x}^*$ on K and $\{\Delta\mathbf{x}^k\}$ converges to zero on K (by assumption), the equations (A.14) and (A.15) yield

$$\mathbf{H}\mathbf{x}^k + \mathbf{c} - \mathbf{A}^T_{\mathcal{A}(\mathbf{x}^*)}\tilde{\boldsymbol{\lambda}}^k_{\mathcal{A}(\mathbf{x}^*)} \to 0 \text{ as } k \to \infty,\, k \in K. \tag{A.16}$$

Since the rows of $\mathbf{A}_{\mathcal{A}(\mathbf{x}^*)}$ are linearly independent by Assumption 2.4, in view of (A.14), there exists a unique $\boldsymbol{\lambda}^*$ to which $\{\tilde{\boldsymbol{\lambda}}^k\}$ converges on K. By taking limits in (A.12), (A.14), and (A.16) and by using boundedness of $\{\boldsymbol{\lambda}^k\}$ due to construction of Algorithm 1, it follows that

$$\mathbf{H}\mathbf{x}^* + \mathbf{c} - \mathbf{A}^T\boldsymbol{\lambda}^* = \mathbf{0}, \tag{A.17}$$

$$\mathbf{S}^*\boldsymbol{\lambda}^* = \mathbf{0}. \tag{A.18}$$

This implies that $\mathbf{x}^*$ is stationary with the unique associated multiplier vector $\boldsymbol{\lambda}^*$. □

LEMMA A.8. *(Corresponds to Lemma 7 of [23]) Let* K *be an infinite index set such that*

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

Then $\{\Delta\mathbf{x}^k\} \to \mathbf{0}$ *as* $k \to \infty$*,* $k \in K$*.*

Proof. By contradiction. Suppose not. First, by (2.23) and by the condition that $\inf\{\|\Delta\mathbf{x}^{k-1}\|^2 + \|\tilde{\boldsymbol{\lambda}}_-^{k-1}\|^2 : k \in K\}$ is greater than 0 (by the assumption on K), λ_i^k $(i \in M)$ is bounded away from

zero on K. Since $\{\mathbf{x}^k\}$ (by Corollary A.5) and $\{\boldsymbol{\lambda}^k\}$ (by construction) are bounded, we may conclude that there is a convergent subsequence $\{\mathbf{x}^k\}$ and $\{\boldsymbol{\lambda}^k\}$. So there exists some infinite index set $K' \subseteq K$, a point $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$, and an index set Q^* such that

$$\begin{aligned} &\inf_{k \in K'} \|\Delta \mathbf{x}^k\| > 0, \\ &\{\mathbf{x}^k\} \to \mathbf{x}^* \text{ as } k \to \infty,\ k \in K', \\ &\{\boldsymbol{\lambda}^k\} \to \boldsymbol{\lambda}^* > \mathbf{0} \text{ as } k \to \infty,\ k \in K', \end{aligned} \tag{A.19}$$

and

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

By Assumptions 2.1 and 2.4 and the fact that $\boldsymbol{\lambda}^* > 0$, Lemma A.1 tells us that $\mathbf{J}(\mathbf{A}_{Q^*}, \mathbf{s}^*_{Q^*}, \boldsymbol{\lambda}^*_{Q^*})$ is nonsingular. It then follows from (A.1)-(A.3) and continuity of $\mathbf{J}(\mathbf{A}_{Q^*}, \mathbf{s}_{Q^*}, \boldsymbol{\lambda}_{Q^*})$ with respect to $\mathbf{s}_{Q^*}$ and $\boldsymbol{\lambda}_{Q^*}$ that, for some $\Delta \mathbf{x}^* \neq \mathbf{0}$ and $\tilde{\boldsymbol{\lambda}}^*$,

$$\{\Delta \mathbf{x}^k\} \to \Delta \mathbf{x}^*,\ \text{as } k \to \infty,\ k \in K', \tag{A.20}$$

$$\{\tilde{\boldsymbol{\lambda}}^k_{Q^*}\} \to \tilde{\boldsymbol{\lambda}}^*_{Q^*},\ \text{as } k \to \infty,\ k \in K'. \tag{A.21}$$

Define $\mathbf{s}^* := \mathbf{A}\mathbf{x}^* - \mathbf{b}$. Since $\mathbf{s}^k = \mathbf{A}\mathbf{x}^k - \mathbf{b}$ and $\{\mathbf{x}^k\} \to \mathbf{x}^*$ on K', we know that $\{\mathbf{s}^k\} \to \mathbf{s}^*$ on K' and $\mathbf{s}^* \geq \mathbf{0}$ by construction of Algorithm 1. In addition, since $Q^* = Q^k \in \mathcal{Q}(\mathbf{A}\mathbf{x}^k - \mathbf{b}, q^k)$ for all $k \in K'$ with $q^k \geq n$, it follows from Lemma A.6 that $\mathcal{A}(\mathbf{x}^*) \subseteq Q^*$. Therefore s^k_i is bounded away from zero for $k \in K'$, $i \notin Q^*$.

Therefore, since $\{\Delta \mathbf{x}^k\} \to \Delta \mathbf{x}^*$ as $k \to \infty$ and since s^k_i is bounded away from zero for $i \notin Q^*$ on K', we know from (A.13) that

$$\forall i \notin Q^*, \quad \{\tilde{\lambda}^k_i\} \to \tilde{\lambda}^*_i \text{ as } k \to \infty,\ k \in K'. \tag{A.22}$$

It then immediately follows from (A.21) and (A.22) that

$$\{\tilde{\boldsymbol{\lambda}}^k\} \to \tilde{\boldsymbol{\lambda}}^* \text{ as } k \to \infty,\ k \in K', \tag{A.23}$$

implying $\{\tilde{\boldsymbol{\lambda}}^k\}$ is bounded on K'.

Up to this point we have shown that $\{\Delta \mathbf{x}^k\} \to \Delta \mathbf{x}^*$ and $\{\tilde{\boldsymbol{\lambda}}^k\} \to \tilde{\boldsymbol{\lambda}}^*$ as $k \to \infty$ on K'. With these facts, we will show that $f(\mathbf{x}^k) \to \infty$ as $k \to \infty$ on K', contradicting Corollary A.5. For this we will first show that α^k is bounded below on K'. Indeed, using (A.13), we can restate (2.19) as

$$\bar{\alpha}^k = \begin{cases} \infty & \text{if } \tilde{\boldsymbol{\lambda}} \leq \mathbf{0}, \\ \min_i \left\{ -\frac{s^k_i}{\Delta s^k_i} = \frac{\lambda^k_i}{\tilde{\lambda}^k_i} \;\middle|\; \text{s.t. } \tilde{\lambda}_i > 0,\ i \in M \right\} & \text{otherwise.} \end{cases} \tag{A.24}$$

Since $\{\tilde{\boldsymbol{\lambda}}^k\}$ is bounded, and each $\{\lambda^k_i\}$ is bounded away from zero for $k \in K'$, it follows that $\bar{\alpha}^k$ is bounded away from zero on K' and, since $\alpha^k \geq \eta \bar{\alpha}^k$ by (2.20), so is α^k. That is, there exists $\underline{\alpha} > 0$ such that $\alpha^k > \underline{\alpha}\ \forall k \in K'$.

We now combine the lower bound on α with the monotonicity of the objective function we showed in Proposition A.4. From the second and third claim of Proposition A.4 (and the expansion of $f(\mathbf{x}^k + \underline{\alpha}\Delta\mathbf{x}^k)$ similar to (A.6)) it immediately follows, since $\alpha^k > \underline{\alpha} > 0$ and $\alpha^k \leq 1$ on K', that

$$\begin{aligned} f(\mathbf{x}^k + \alpha^k \Delta\mathbf{x}^k) &< f(\mathbf{x}^k + \underline{\alpha}\Delta\mathbf{x}^k) \\ &= f(\mathbf{x}^k) - \underline{\alpha}(1 - \frac{1}{2}\underline{\alpha})\Delta\mathbf{x}^{kT}\mathbf{H}\Delta\mathbf{x}^k + \underline{\alpha}\tilde{\boldsymbol{\lambda}}_{Q^k}^{kT}\mathbf{A}_{Q^k}\Delta\mathbf{x}^k. \end{aligned} \tag{A.25}$$

Our focus from now is showing that $\underline{\alpha}(1 - \frac{1}{2}\underline{\alpha})\Delta\mathbf{x}^{kT}\mathbf{H}\Delta\mathbf{x}^k - \underline{\alpha}\tilde{\boldsymbol{\lambda}}_{Q^k}^{kT}\mathbf{A}_{Q^k}\Delta\mathbf{x}^k$ is bounded below on K', which immediately implies that $f(\mathbf{x}^k) \to -\infty$ on K', since by Proposition A.4 $\{f(\mathbf{x}^k)\}$ is monotonically decreasing. Taking limits in (A.2)-(A.3) on K' yields

$$\mathbf{S}_{Q^{*2}}^* \tilde{\boldsymbol{\lambda}}_{Q^*}^* = -\boldsymbol{\Lambda}_{Q^{*2}}^* \mathbf{A}_{Q^*}\Delta\mathbf{x}^* \text{ as } k \to \infty,\ k \in K'. \tag{A.26}$$

So, since $\boldsymbol{\lambda}^* > \mathbf{0}$ by (A.19) and $\mathbf{s}^* \geq \mathbf{0}$, we know that $\mathbf{S}_{Q^2}^* \tilde{\boldsymbol{\lambda}}_{Q^*}^* \geq \mathbf{0}$ and $\mathbf{A}_{Q^*}\Delta\mathbf{x}^* \leq \mathbf{0}$. So, for $i \in Q^*$, it follows that $\mathbf{a}_i^T\Delta\mathbf{x}^* < 0$ if $\mathbf{a}_i^T\Delta\mathbf{x}^* \neq 0$. Thus we know that $\tilde{\boldsymbol{\lambda}}_{Q^*}^{*T}\mathbf{A}_{Q^*}\Delta\mathbf{x}^* = \sum_{i\in Q^*} \tilde{\lambda}_i^* \mathbf{a}_i^T\Delta\mathbf{x}^* < 0$ if $\mathbf{A}_{Q^*}\Delta\mathbf{x}^* \neq 0$. From this, since $\mathcal{N}(\mathbf{A}_{Q^*}) \cap \mathcal{N}(\mathbf{H}) = \{\mathbf{0}\}$ under Assumption 2.1 and $\mathbf{H}$ is positive semidefinite, we conclude that $\Delta\mathbf{x}^{*T}\mathbf{H}\Delta\mathbf{x}^* > 0$ or $-\tilde{\boldsymbol{\lambda}}_{Q^*}^{*T}\mathbf{A}_{Q^*}\Delta\mathbf{x}^* > 0$, similarly to Lemma A.3. So, since $\Delta\mathbf{x}^{kT}\mathbf{H}\Delta\mathbf{x}^k \to \Delta\mathbf{x}^{*T}\mathbf{H}\Delta\mathbf{x}^*$ and $-\tilde{\boldsymbol{\lambda}}_{Q^k}^{kT}\mathbf{A}_{Q^k}\Delta\mathbf{x}^k \to -\tilde{\boldsymbol{\lambda}}_{Q^*}^{*T}\mathbf{A}_{Q^*}\Delta\mathbf{x}^*$ as $k \to \infty$ on K' and $0 < \underline{\alpha} < 1$, there exists $\delta > 0$ such that, for large enough $k \in K'$,

$$\underline{\alpha}(1 - \frac{1}{2}\underline{\alpha})\Delta\mathbf{x}^{kT}\mathbf{H}\Delta\mathbf{x}^k - \underline{\alpha}\tilde{\boldsymbol{\lambda}}_{Q^*}^{kT}\mathbf{A}_{Q^*}\Delta\mathbf{x}^k > \delta. \tag{A.27}$$

Since $f(\mathbf{x}^k)$ is monotonic decreasing by Proposition A.4, (A.25) yields that for some large k'

$$f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k) - \delta,\ \forall k \geq k' \text{ and } k \in K', \tag{A.28}$$

which implies that $f(\mathbf{x}^k) \to -\infty$ as $k \to \infty$. This contradicts the boundedness of $\{\mathbf{x}^k\}$. □

The remainder of the analysis is a direct transposition from [24] and [23] to the present setting. We omit the details (which can be found in [13]). First, using Corollary A.5, Lemma A.7 and Lemma A.8, the following lemma can be readily proven.

LEMMA A.9. *(Corresponds to Lemma 8 of [23]) Suppose $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$ on some infinite index set K. Then $\{\Delta\mathbf{x}^k\}$ goes to $\mathbf{0}$ on K.*

This fact, together with Corollary A.5 and Lemma A.7, leads to the following lemma.

LEMMA A.10. *(Corresponds to Lemma 9 of [23]) $\{\mathbf{x}^k\}$ approaches the set of stationary points of (1.1), i.e., for any $\epsilon > 0$ there exists k' so that $\mathbf{x}^k$ is ϵ-close to a stationary point for $k > k'$.*

As a result of this and of Lemma A.9, when the sequence $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$, it follows that the set of all limit points of the sequence,

$$L := \{\mathbf{x} : \mathbf{x} \text{ is a limit point of } \{\mathbf{x}^k\}\},$$

is connected as the following lemma establishes.

LEMMA A.11. *(Corresponds to Lemma A.5 under Lemma 3.6 of [24]) If $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$, then L is connected.*

In addition, from Lemma A.10 it can be shown that the line going through any two points $\mathbf{x}, \mathbf{x}' \in L$ associated with the same active constraints is parallel to the null space of $\mathbf{H}$ as claimed in the following lemma.

LEMMA A.12. *(Corresponds to Lemma A.3 under Lemma 3.6 of [24]) Let $\mathbf{x}, \mathbf{x}' \in L$ be such that $\mathcal{A}(\mathbf{x}) = \mathcal{A}(\mathbf{x}')$. Then $\mathbf{H}(\mathbf{x} - \mathbf{x}') = 0$.*

This result, together with Lemma A.11, implies that, when the sequence $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$, the smallest affine space containing L is parallel to the null space of $\mathbf{H}$ as the following lemma shows.

LEMMA A.13. *(Corresponds to Lemma A.4 under Lemma 3.6 of [24]) If $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$, then for all $\mathbf{x}, \mathbf{x}' \in L$, $\mathbf{H}(\mathbf{x}' - \mathbf{x}) = 0$.*

Proof. Since there are only finitely many possible combinations of binding constraints, in view of Lemma A.12, L is a finite union of affine sets of the form $L \cap (\mathbf{x} + \mathcal{N}(\mathbf{H}))$ with $\mathbf{x} \in L$.

Suppose that there are N such distinct affine sets A_i which are in the form $L \cap (\mathbf{x} + \mathcal{N}(\mathbf{H}))$ with $\mathbf{x} \in L$. Then $L = \cup_{i=1}^N A_i$. Notice each A_i is a subset of an affine subspace $\boldsymbol{\ell}_i + \mathcal{N}(\mathbf{H})$ for $\boldsymbol{\ell}_i \in A_i$. So, for any distinct $i, j \in \{1, \ldots, N\}$, A_i and A_j lie on either the same affine subspace $(\boldsymbol{\ell}_i + \mathcal{N}(\mathbf{H}) = \boldsymbol{\ell}_j + \mathcal{N}(\mathbf{H}))$ or parallel affine subspaces $(\boldsymbol{\ell}_i + \mathcal{N}(\mathbf{H}) \neq \boldsymbol{\ell}_j + \mathcal{N}(\mathbf{H}))$. However, since L is connected in view of Lemma A.11 and there are finitely many A_i's, they can not lie on distinct parallel affine subspaces. Therefore all A_i's lie on the same affine subspace which is parallel to $\mathcal{N}(\mathbf{H})$. This proves the claim. □

From this and Lemmas A.10, A.11 and A.12, we can conclude the following.

LEMMA A.14. *(Corresponds to Lemma 3.6 of [24]) Suppose $\{\mathbf{x}^k\}$ is bounded away from $\mathcal{F}_P^*$. Let $\mathbf{x}^*, \mathbf{x}'^* \in L$. Let $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ be the associated multiplier vectors. Then $\boldsymbol{\lambda} = \boldsymbol{\lambda}'$.*

With Lemmas A.7, A.9,A.10 and A.14 and Proposition A.4 in hand, the proof of Theorem 2.2 can be completed along the lines of the global convergence analysis in [23, Theorem 12]. See [13] for full details and proofs not provided in this paper.

A.2. Local Rate of Convergence. In this section, we will show that, under Assumptions 2.1-2.6, $\{\mathbf{x}^k, \boldsymbol{\lambda}^k\}$ converges to the primal-dual solution $\{\mathbf{x}^*, \boldsymbol{\lambda}^*\}$ q-quadratically. For this result, suppose Assumptions 2.1-2.6 hold.

LEMMA A.15. *(Corresponds to Lemma 1 of [23]) $\mathbf{J}_a(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ in (A.4) is nonsingular if and only if $\mathbf{J}(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ is nonsingular.*

Proof. Assume that $\mathbf{J}_a(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ is singular. Then there exists a nonzero vector $[\mathbf{u}^T,\ \mathbf{w}^T]^T \neq \mathbf{0}$ such that

$$\mathbf{H}\mathbf{u} - \mathbf{A}^T\mathbf{w} = \mathbf{0}, \tag{A.29}$$

$$\boldsymbol{\Lambda}\mathbf{A}\mathbf{u} + \mathbf{S}\mathbf{w} = \mathbf{0}. \tag{A.30}$$

Now letting $\mathbf{v} := \mathbf{A}\mathbf{u}$, it immediately follows from (A.30) that

$$\boldsymbol{\Lambda}\mathbf{v} + \mathbf{S}\mathbf{w} = \mathbf{0}, \tag{A.31}$$

which implies that $\mathbf{J}(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ is also singular since $[\mathbf{u}^T,\ \mathbf{v}^T,\ \mathbf{w}^T] \neq \mathbf{0}$.

Now assume that $\mathbf{J}(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ is singular. Then there exists a nonzero vector $[\mathbf{u}^T,\ \mathbf{v}^T,\ \mathbf{w}^T]^T \neq \mathbf{0}$ such that

$$\mathbf{H}\mathbf{u} - \mathbf{A}^T\mathbf{w} = \mathbf{0}, \tag{A.32}$$

$$\mathbf{A}\mathbf{u} - \mathbf{v} = \mathbf{0}, \tag{A.33}$$

$$\boldsymbol{\Lambda}\mathbf{v} + \mathbf{S}\mathbf{w} = \mathbf{0}. \tag{A.34}$$

Then $\mathbf{u}$ and $\mathbf{w}$ naturally satisfy (A.29) and (A.30). If $\mathbf{u} = \mathbf{0}$ and $\mathbf{w} = \mathbf{0}$ at the same time, then $\mathbf{v}$ is also a zero vector, a contradiction. Thus $J_a(\mathbf{A}, \mathbf{s}, \boldsymbol{\lambda})$ is also singular. □

From here on, we denote by $\mathbf{x}^*$ the optimal solution to (1.1) (By Assumption 2.5, it exists and is unique) and by $\boldsymbol{\lambda}^*$ its associated Lagrange multiplier. We define $\mathbf{s}^* := \mathbf{A}\mathbf{x}^* - \mathbf{b}$.

LEMMA A.16. *(Corresponds to Lemma 13 of [23]) If $\mathcal{A}(\mathbf{x}^*) \subseteq Q$ then $\mathbf{J}(\mathbf{A}_Q, \mathbf{s}^*_Q, \boldsymbol{\lambda}^*_Q)$ and $\mathbf{J}_a(\mathbf{A}_Q, \mathbf{s}^*_Q, \boldsymbol{\lambda}^*_Q)$ are nonsingular.*

Proof. Let us verify the assumptions of Lemma A.1. First, $\mathbf{s}^*_Q + \boldsymbol{\lambda}^*_Q > \mathbf{0}$ due to strict complementarity Assumption 2.6. Second, the rows of $\mathbf{A}_{\mathcal{A}(\mathbf{x}^*)}$ are linearly independent by Assumption 2.4, and $s^*_i = 0$ for $i \in \mathcal{A}(\mathbf{x}^*) \subseteq Q$ and $s^*_i > 0$ for $i \notin \mathcal{A}(\mathbf{x}^*)$. Third, $\mathbf{A}_{\mathcal{A}(\mathbf{x}^*)}$ and $\mathbf{H}$ share the trivial nullspace due to Assumptions 2.5 and 2.6 which implies that $\{j : \lambda^*_j \neq 0\} = \mathcal{A}(\mathbf{x}^*)$. Thus the conclusion follows from Lemmas A.1 and A.15. □

LEMMA A.17. *(Corresponds to Lemma 14 of [23]) Under our assumptions,*

(i) $\{\Delta\mathbf{x}^k\} \to \mathbf{0}$,

(ii) $\{\tilde{\boldsymbol{\lambda}}^k\} \to \boldsymbol{\lambda}^*$,

(iii) If $\lambda^*_i \leq \lambda_{\max}$ for all $i \in M$, then $\{\boldsymbol{\lambda}^k\} \to \boldsymbol{\lambda}^*$.

Proof. Since $\{\mathbf{x}^k\} \to \mathbf{x}^*$ (Theorem 2.2), the first claim immediately follows. The second claim follows by Lemma A.7 and the third claim by (2.23). □

The q-quadratic convergence will be shown using the following property of Newton's method, which is adapted from Proposition 3.10 of [24].

PROPOSITION A.18. *(Corresponds to Proposition 3.10 of [24]) Let $\Psi : \mathbb{R}^n \to \mathbb{R}^n$ be twice continuously differentiable and let $\mathbf{t}^* \in \mathbb{R}^n$ be a zero of Ψ, i.e., $\Psi(\mathbf{t}^*) = \mathbf{0}$. Suppose there exists $\epsilon > 0$ such that $\frac{\partial\Psi}{\partial\mathbf{t}}(\mathbf{t})$ is nonsingular for all $\mathbf{t} \in B(\hat{\mathbf{s}}, \epsilon) := \{\mathbf{t} : \|\mathbf{t} - \mathbf{t}^*\| \leq \epsilon\}$. Define $\Delta^{\mathrm{N}}\mathbf{t}$ to be the Newton increment at $\mathbf{t}$, i.e., $\Delta^{\mathrm{N}}\mathbf{t} := -(\frac{\partial\Psi}{\partial\mathbf{t}}(\mathbf{t}))^{-1}\Psi(\mathbf{t})$. Then, given any $c > 0$, for all $\mathbf{t} \in B(\mathbf{t}^*, \epsilon)$, if $\mathbf{t}^+ \in \mathbb{R}^n$ satisfies, for each $i \in \{1, \ldots, n\}$, either*

(i) $|t^+_i - t^*_i| \leq c\|\Delta^{\mathrm{N}}\mathbf{t}\|^2$
or

(ii) $|t^+_i - (t_i + \Delta^{\mathrm{N}}t_i)| \leq c\|\Delta^{\mathrm{N}}\mathbf{t}\|^2$,

then there exists $\nu > 0$ such that

$$\|\mathbf{t}^+ - \mathbf{t}^*\| \leq \nu\|\mathbf{t} - \mathbf{t}^*\|^2. \tag{A.35}$$

Proof. See [24]. □

To use this proposition, we write the first three conditions of the KKT system (2.2)-(2.4) ($\mu = 0$)

as $\Psi(\mathbf{x}, \boldsymbol{\lambda}) = 0$, where

$$\Psi(\mathbf{x}, \boldsymbol{\lambda}) := \begin{bmatrix} \mathbf{H}\mathbf{x} - \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{c} \\ \boldsymbol{\Lambda}(\mathbf{A}\mathbf{x} - \mathbf{b}) \end{bmatrix}. \tag{A.36}$$

Then

$$\begin{bmatrix} \mathbf{H} & -\mathbf{A}^T \\ \boldsymbol{\Lambda}\mathbf{A} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{r_c} \\ -\mathbf{S}\boldsymbol{\lambda} + \sigma\mu\mathbf{e} - \boldsymbol{\Lambda}\mathbf{r_b} \end{bmatrix}, \tag{A.37}$$

$$\Delta\mathbf{s} = \mathbf{A}\Delta\mathbf{x} + \mathbf{r_b}. \tag{A.38}$$

is equivalent to the Newton direction for the solution of $\Psi(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}$, and $\mathbf{J}_a(\mathbf{A}, \mathbf{A}\mathbf{x} - \mathbf{b}, \boldsymbol{\lambda})$ is the Jacobian of $\Psi(\mathbf{x}, \boldsymbol{\lambda})$. We use $\mathbf{t}$ to denote the vector containing both $\mathbf{x}$ and $\boldsymbol{\lambda}$, i.e., $\mathbf{t}^k = [\mathbf{x}^{kT},\ \boldsymbol{\lambda}^{kT}]^T$. Also, we define a strictly feasible set E^o for Ψ:

$$E^o := \{\mathbf{t} : \mathbf{x} \in \mathcal{F}_P^o,\ \boldsymbol{\lambda} > 0\}. \tag{A.39}$$

Hence, $E^o \cap B(\mathbf{t}^*, \epsilon)$ denotes the set of strictly feasible points in a ball around $\mathbf{t}^*$. Given $\mathbf{t} \in E^o$ and $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}, q)$, $\Delta\mathbf{t} := [\Delta\mathbf{x}^T, \Delta\boldsymbol{\lambda}^T]^T$ denotes the composite direction at $\mathbf{t}$ generated by Algorithm 1.

The direction generated by Algorithm 1 is not the same as the Newton direction for Ψ. The following lemma relates the two directions.

LEMMA A.19. *(Corresponds to Lemma 16 of [23].) Let ϵ be such that, for all $\mathbf{t} \in E^o \cap B(\mathbf{t}^*, \epsilon)$ and for all $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}, q)$, $\mathbf{J}_a(\mathbf{A}_Q, \mathbf{A}_Q\mathbf{x} - \mathbf{b}, \boldsymbol{\lambda}_Q)$ is nonsingular and $\mathbf{A}_{Q^c}\mathbf{x} > \mathbf{b}_{Q^c}$. Then there exists a positive constant ξ such that, for all $\mathbf{t} \in E^o \cap B(\mathbf{t}^*, \epsilon)$ and for any $Q \in \mathcal{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}, q)$,*

$$\|\Delta\mathbf{t} - \Delta^{\mathrm{N}}\mathbf{t}\| \le \xi\|\mathbf{t} - \mathbf{t}^*\|\|\Delta^{\mathrm{N}}\mathbf{t}\|.$$

Again, with this result, Proposition A.18, and Lemmas A.15 and A.17 and with the help of [24], the proof of Theorem 2.3 can be completed by inserting the Hessian $\mathbf{H}$ in appropriate places of [23]. See [13] for full details and proofs not provided in this paper.

Acknowledgement. We are grateful to the referees for their careful reading and suggestions.

REFERENCES

[1] P.-A. Absil and A. L. Tits. Newton-KKT interior-point methods for indefinite quadratic programming. *Comput. Optim. Appl.*, 36(1):5–41, 2007.
[2] K. E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, Washington, D.C. 20005, 1989.
[3] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
[4] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Sep 1995.
[5] G. B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, N.J., 1963.
[6] G. B. Dantzig and Y. Ye. A build–up interior method for linear programming : Affine scaling form. Technical report, University of Iowa, Iowa City, IA 52242, USA, July 1991.

[7] M. C. Ferris and T. S. Munson. Interior-point methods for massive support vector machines. *SIAM J. Optim.*, 13(3):783–804, 2002.

[8] I. Griva, S. G. Nash, and A. Sofer. *Linear and Nonlinear Optimization, Second Edition*. SIAM Press, Philadelphia, PA, 2009.

[9] D. Hertog, C. Roos, and T. Terlaky. A build-up variant of the path-following method for LP. Technical Report DUT-TWI-91-47, Delft University of Technology, Delft, The Netherlands, 1991.

[10] D. Hertog, C. Roos, and T. Terlaky. Adding and deleting constraints in the path–following method for linear programming. In *Advances in Optimization and Approximation (Nonconvex Optimization and Its Applications)*, volume 1, pages 166–185. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994.

[11] N. J. Higham. Analysis of the Cholesky decomposition of a semi-definite matrix. In M. G. Cox and S. J. Hammarling, editors, *Reliable Numerical Computation*, pages 161–185, Walton Street, Oxford OX2 6DP, UK, 1990. Oxford University Press.

[12] A. S. Householder. *The Theory of Matrices in Numerical Analysis*. Blaisdell, New York, 1964. Reprinted by Dover, New York, 1975.

[13] J. H. Jung. *Adaptive Constraint Reduction for Convex Quadratic Programming and Training Support Vector Machines*. PhD thesis, University of Maryland, 2008. Available at http://hdl.handle.net/1903/8020.

[14] J. H. Jung, D. P. O'Leary, and A. L. Tits. Adaptive constraint reduction for training support vector machines. *Electronic Transactions on Numerical Analysis*, 31:156–177, 2008.

[15] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[16] Z.-Q. Luo and J. Sun. An analytic center based column generation algorithm for convex quadratic feasibility problems. *SIAM J. Optim.*, 9(1):217–235, 1998.

[17] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM J. Optim.*, 2(4):575–601, Nov. 1992.

[18] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2000.

[19] S. Nordebo, Z. Zang, and I. Claesson. A semi-infinite quadratic programming algorithm with applications to array pattern synthesis. *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, 48(3):225–232, Mar 2001.

[20] Y. Saad. *Iterative Methods for Sparse Linear Systems, 2nd edition*, chapter 9. Preconditioned Iterations, pages 261–281. SIAM, Philadelphia, PA, 2nd edition, 2003.

[21] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[22] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems*. John Wiley & Sons, Washington, D.C. 20005, 1977.

[23] A. L. Tits, P.-A. Absil, and W. P. Woessner. Constraint reduction for linear programs with many inequality constraints. *SIAM J. Optim.*, 17(1):119–146, 2006.

[24] A. L. Tits and J. L. Zhou. A simple, quadratically convergent interior point algorithm for linear programming and convex quadratic programming. In W. W. Hager, D. W. Hearn, and P. M. Pardalos, editors, *Large Scale Optimization: State of the Art*, pages 411–427. Kluwer Academic Publishers, 1994.

[25] K. Tone. An active-set strategy in an interior point method for linear programming. *Math. Program.*, 59(3):345–360, 1993.

[26] W. Wang and D. P. O'Leary. Adaptive use of iterative methods in predictor-corrector interior point methods for linear programming. *Numerical Algorithms*, 25(1–4):387–406, 2000.

[27] G. Watson. Choice of norms for data fitting and function approximation. *Acta Numerica*, pages 337–377, 1998.

[28] L. Winternitz, S. O. Nicholls, A. Tits, and D. O'Leary. A constraint reduced variant of Mehrotra's Predictor-Corrector Algorithm, 2007. Submitted for publication. `http://www.optimization-online.org/DB_FILE/2007/07/1734.pdf`.

[29] S. J. Wright. *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.

[30] Y. Ye. A "build-down" scheme for linear programming. *Math. Program.*, 46(1):61–72, 1990.

[31] Y. Ye. An $O(n^3L)$ potential reduction algorithm for linear programming. *Math. Program.*, 50:239–258, 1991.

[32] Y. Ye. A potential reduction algorithm allowing column generation. *SIAM J. Optim.*, 2(1):7–20, Feb. 1992.

[33] Y. Zhang. Solving large–scale linear programs by interior–point methods under the MATLAB environment. Technical Report 96–01, Dept. of Mathematics and Statistics, Univ. of Maryland Baltimore County, 1996.