KU LEUVEN

STADIUS
Center for Dynamical Systems, Signal Processing and Data Analytics

Citation/Reference	Quoc Tran Dinh, Ion Necoara, Carlo Savorgnan, Moritz Diehl (2013), **An inexact perturbed path-following method for Lagrangian decomposition in large-scale separable convex optimization** SIAM J. Optim. Vol. 23, no 1, 95-125.
Archived version	Final publisher's version / pdf
Published version	insert link to the published version of your paper http://dx.doi.org/10.1137/11085311X
Journal homepage	
Author contact	your email quoc.trandinh@epfl.ch Klik hier als u tekst wilt invoeren.
IR	url in Lirias https://lirias.kuleuven.be/handle/123456789/361656

(article begins on next page)

SIAM J. OPTIM.
Vol. 23, No. 1, pp. 95–125
© 2013 Society for Industrial and Applied Mathematics

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

AN INEXACT PERTURBED PATH-FOLLOWING METHOD FOR LAGRANGIAN DECOMPOSITION IN LARGE-SCALE SEPARABLE CONVEX OPTIMIZATION*

QUOC TRAN DINH†, ION NECOARA‡, CARLO SAVORGNAN§, AND MORITZ DIEHL§

Abstract. This paper studies an inexact perturbed path-following algorithm in the framework of Lagrangian dual decomposition for solving large-scale separable convex programming problems. Unlike the exact versions considered in the literature, we propose solving the primal subproblems inexactly up to a given accuracy. This leads to an inexactness of the gradient vector and the Hessian matrix of the smoothed dual function. Then an inexact perturbed algorithm is applied to minimize the smoothed dual function. The algorithm consists of two phases, and both make use of the inexact derivative information of the smoothed dual problem. The convergence of the algorithm is analyzed, and the worst-case complexity is estimated. As a special case, an exact path-following decomposition algorithm is obtained and its worst-case complexity is given. Implementation details are discussed, and preliminary numerical results are reported.

Key words. smoothing technique, self-concordant barrier, Lagrangian decomposition, inexact perturbed Newton-type method, separable convex optimization, parallel algorithm

AMS subject classifications. 90C25, 49M27, 90C06, 49M15, 90C51

DOI. 10.1137/11085311X

1. Introduction.

Many optimization problems arising in networked systems, image processing, data mining, economics, distributed control, and multistage stochastic optimization can be formulated as separable convex optimization problems; see, e.g., [5, 11, 8, 14, 20, 24, 25, 28] and the references quoted therein. For a centralized setup and problems of moderate size there exist many standard iterative algorithms to solve them, such as Newton, quasi-Newton, or projected gradient-type methods. But in many applications, we encounter separable convex programming problems which may not be easy to solve by standard optimization algorithms due to the high dimensionality; the hierarchical, multistage, or dynamical structure; the existence of multiple decision-makers; or the distributed locations of data and devices. Decomposition methods can be an appropriate choice for solving these problems. Moreover, decomposition approaches also benefit if the primal subproblems generated from the

*Received by the editors October 26, 2011; accepted for publication (in revised form) October 15, 2012; published electronically January 29, 2013. This research was supported by Research Council KUL: CoE EF/05/006 Optimization in Engineering (OPTEC), IOF-SCORES4CHEM, GOA/10/009 (MaNet), GOA/10/11, several PhD/postdoc and fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects G.0452.04, G.0499.04, G.0211.05, G.0226.06, G.0321.06, G.0302.07, G.0320.08, G.0558.08, G.0557.08, G.0588.09, G.0377.09, G.0712.11, research communities (ICCoS, ANMMM, MLDM); IWT: PhD Grants, Belgian Federal Science Policy Office: IUAP P6/04; EU: ERNSI; FP7-HDMPC, FP7-EMBOCON no. 248940, ERC-HIGHWIND, Contract Research: AMINAL. Other: Helmholtz-viCERP, COMET-ACCM, CNCS-UEFISCDI (TE, no. 19/11.08.2010); CNCS (PN II, no. 80EU/2010); POSDRU (no. 89/1.5/S/62557).

http://www.siam.org/journals/siopt/23-1/85311.html

†Department of Electrical Engineering (ESAT-SCD) and Optimization in Engineering Center (OPTEC), K.U. Leuven, B-3001 Leuven, Belgium, and Department of Mathematics-Mechanics-Informatics, VNU University of Science, Hanoi, Vietnam (quoc.trandinh@esat.kuleuven.be).

‡Automation and Systems Engineering Department, University Politehnica of Bucharest, 060042 Bucharest, Romania (ion.necoara@acse.pub.ro).

§Department of Electrical Engineering (ESAT-SCD) and Optimization in Engineering Center (OPTEC), K.U. Leuven, B-3001 Leuven, Belgium (carlo.savorgnan@esat.kuleuven.be, moritz.diehl@esat.kuleuven.be).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

components of the problem can be solved in a closed form or lower computational cost than the full problem.

In this paper, we are interested in the following separable convex programming problem (SCPP):

$$\text{(SCPP)} \qquad \phi^* := \left\{ \begin{array}{ll} \max\limits_{x\in\mathbb{R}^n} & \left\{\phi(x) := \sum_{i=1}^M \phi_i(x_i)\right\} \\ \text{s.t.} & \sum_{i=1}^M (A_i x_i - b_i) = 0, \\ & x_i \in X_i,\ i = 1, \ldots, M, \end{array} \right.$$

where $x := (x_1^T, \ldots, x_M^T)^T$ with $x_i \in \mathbb{R}^{n_i}$ is a vector of decision variables, each $\phi_i : \mathbb{R}^{n_i} \to \mathbb{R}$ is concave, X_i is a nonempty, closed convex subset in $\mathbb{R}^{n_i}$, $A_i \in \mathbb{R}^{m\times n_i}$, $b_i \in \mathbb{R}^m$ for all $i = 1, \ldots, M$, and $n_1 + n_2 + \cdots + n_M = n$. The first constraint is usually referred to as a *linear coupling constraint*.

Several methods have been proposed for solving problem (SCPP) by decomposing it into smaller subproblems that can be solved separately by standard optimization techniques; see, e.g., [2, 4, 13, 19, 22]. One standard technique for treating separable programming problems is Lagrangian dual decomposition [2]. However, using such a technique generally leads to a nonsmooth optimization problem. There are several approaches to overcoming this difficulty by smoothing the dual function. One can add an augmented Lagrangian term [19] or a proximal term [4] to the objective function of the primal problem. Unfortunately, the first approach breaks the separability of the original problem due to the cross terms between the components. The second approach is a more tractable way to solve this type of problem.

Recently, smoothing techniques in convex optimization have attracted increasing interest and have found many applications [16]. In the framework of the Lagrangian dual decomposition, there are two relevant approaches. The first is regularization. By adding a regularization term such as a proximal term to the objective function, the primal subproblems become strongly convex. Consequently, the dual master problem is smooth, which allows one to apply smoothing optimization techniques [4, 13, 22]. The second approach is using barrier functions. This technique is suitable for problems with conic constraints [7, 10, 12, 14, 21, 27, 28]. Several methods in this direction used a fundamental property that, by smoothing via self-concordant log-barriers, the family of the dual functions depending on a penalty parameter is strongly self-concordant in the sense of Nesterov and Nemirovskii [17]. Consequently, path-following methods can be applied to solve the dual master problem. Up to now, the existing methods required a crucial assumption that the primal subproblems are solved exactly. In practice, solving the primal subproblems exactly to construct the dual function is only conceptual. Any numerical optimization method provides an approximate solution, and, consequently, the dual function is also approximated. In this paper, we study an inexact perturbed path-following decomposition method for solving (SCPP) which employs approximate gradient vectors and approximate Hessian matrices of the smoothed dual function.

Contribution. The contribution of this paper is as follows:

1. By applying a smoothing technique via self-concordant barriers, we construct a local and a global smooth approximation to the dual function and estimate the approximation error.
2. A new two-phase inexact perturbed path-following decomposition algorithm is proposed for solving (SCPP). Both phases allow one to solve the primal subproblems approximately. The overall algorithm is highly parallelizable.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

3. The convergence and the worst-case complexity of the algorithm are investigated under standard assumptions used in any interior point method.
4. As a special case, an exact path-following decomposition algorithm studied in [12, 14, 21, 28] is obtained. However, for this variant we obtain better values for the radius of the neighborhood of the central path compared to those from existing methods.

Let us emphasize some differences between the proposed method and existing similar methods. First, although smoothing techniques via self-concordant barriers are not new [12, 14, 21, 28], in this paper we prove a new local and global estimate for the dual function. These estimates are based only on the convexity of the objective function, which is not necessarily smooth. Since the smoothed dual function is continuously differentiable, smooth optimization techniques can be used to minimize such a function. Second, the new algorithm allows us to solve the primal subproblems inexactly, where the inexactness in the early iterations of the algorithm can be high, resulting in significant time saving when the solution of the primal subproblems requires a high computational cost. Note that the proposed algorithm is different from that considered in [26] for linear programming, where the inexactness of the primal subproblems was defined in a different way. Third, by analyzing directly the convergence of the algorithm based on a recent monograph [15], the theory in this paper is self-contained. Moreover, it also allows us to optimally choose the parameters and to trade off between the convergence rate of the dual master problem and the accuracy of the primal subproblems. Fourth, we also show how to recover the primal solution of the original problem. This step was usually ignored in the previous methods. Finally, in the exact case, the radius of the neighborhood of the central path is $(3-\sqrt{5})/2 \approx 0.38197$, which is larger than $2-\sqrt{3} \approx 0.26795$ of previous methods [12, 14, 21, 28]. Moreover, since the performance of an interior point algorithm crucially depends on the parameters of the algorithm, we analyze directly the path-following iteration to select these parameters in an appropriate way.

The rest of this paper is organized as follows. In the next section, we briefly recall the Lagrangian dual decomposition method in separable convex optimization. Section 3 is devoted to constructing smooth approximations for the dual function via self-concordant barriers and investigates the main properties of these approximations. Section 4 presents an inexact perturbed path-following decomposition algorithm and investigates its convergence and its worst-case complexity. Section 5 deals with an exact variant of the algorithm presented in section 4. Section 6 discusses implementation details, and section 7 presents preliminary numerical tests. The proofs of the technical statements are given in Appendix A.

Notation and terminology. Throughout the paper, we shall consider the Euclidean space $\mathbb{R}^n$ endowed with an inner product $x^T y$ for $x, y \in \mathbb{R}^n$ and the Euclidean norm $\|x\| = \sqrt{x^T x}$. The notation $x = (x_1, \ldots, x_M)$ defines a vector in $\mathbb{R}^n$ formed from M subvectors $x_i \in \mathbb{R}^{n_i}$, $i = 1, \ldots, M$, where $n_1 + \cdots + n_M = n$.

For a given symmetric real matrix P, the expression $P \succeq 0$ (resp., $P \succ 0$) means that P is positive semidefinite (resp., positive definite); $P \preceq Q$ means that $Q - P \succeq 0$.

For a proper, lower semicontinuous convex function f, $\mathrm{dom}(f)$ denotes the domain of f, $\overline{\mathrm{dom}}(f)$ is the closure of $\mathrm{dom}(f)$, and $\partial f(x)$ denotes the subdifferential of f at x. For a concave function f we also denote by $\partial f(x)$ the "superdifferential" of f at x, i.e., $\partial f(x) := -\partial\{-f(x)\}$. Let f be twice continuously differentiable and convex on $\mathbb{R}^n$. For a given vector u, the local norm of u w.r.t. f at x, where $\nabla^2 f(x) \succ 0$, is defined as $\|u\|_x := \left[u^T \nabla^2 f(x) u\right]^{1/2}$ and its dual norm is $\|u\|_x^* := \max\{u^T v \mid \|v\|_x \leq$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

$1\} = \left[u^T\nabla^2 f(x)^{-1}u\right]^{1/2}$. Clearly, $\left|u^T v\right| \le \|u\|_x\|v\|_x^*$. The set $\mathcal{N}_X(x) := \{w \in \mathbb{R}^n \mid w^T(x-u) \ge 0,\ u \in X\}$ if $x \in X$ and $\mathcal{N}_X(x) := \emptyset$, otherwise, is called the normal cone of a closed convex set X at x.

The notation $\mathbb{R}_+$ (resp., $\mathbb{R}_{++}$) defines the set of nonnegative (resp., positive) real numbers. The function $\omega : \mathbb{R}_+ \to \mathbb{R}$ is defined by $\omega(t) := t - \ln(1+t)$, and its dual $\omega_* : [0,1] \to \mathbb{R}$ is defined by $\omega_*(t) := -t - \ln(1-t)$. Note that both functions are convex, nonnegative, and increasing. For a real number x, $\lfloor x \rfloor$ denotes the largest integer number which is less than or equal to x, and ":=" means "equal by definition."

2. Lagrangian dual decomposition in convex optimization. A classical technique for addressing coupling constraints in SCPP is Lagrangian dual decomposition [2]. We briefly recall such a technique in this section.

Let $A := [A_1, \dots, A_M]$ and $b := \sum_{i=1}^M b_i$. The linear coupling constraint $\sum_{i=1}^M (A_i x_i - b_i) = 0$ can be written as $Ax = b$. The Lagrange function associated with the constraint $Ax = b$ for problem (SCPP) is defined by $\mathcal{L}(x,y) := \phi(x) + y^T(Ax - b) = \sum_{i=1}^M \left[\phi_i(x_i) + y^T(A_i x_i - b_i)\right]$, where $y \in \mathbb{R}^m$ is the corresponding Lagrange multiplier. The dual problem of (SCPP) is formulated as

$$d_0^* := \min_{y \in \mathbb{R}^m} d_0(y), \tag{2.1}$$

where d_0 is the dual function defined by

$$d_0(y) := \max_{x \in X} \mathcal{L}(x,y) = \max_{x \in X} \left\{ \sum_{i=1}^M \left[\phi_i(x_i) + y^T(A_i x_i - b_i)\right] \right\}. \tag{2.2}$$

We say that problem (SCPP) satisfies the Slater condition if $\mathrm{ri}(X) \cap \{x \in \mathbb{R}^n \mid Ax = b\} \ne \emptyset$, where $\mathrm{ri}(X)$ is the relative interior of the convex set X [3]. Let us denote by X^* and Y^* the solution sets of (SCPP) and (2.1), respectively. Throughout the paper, we assume that the following fundamental assumptions hold; see [19].

Assumption A1.

(a) The solution set X^* of (SCPP) is nonempty, and either the Slater condition for (SCPP) is satisfied or X is polyhedral.

(b) For $i = 1, \dots, M$, the function ϕ_i is proper, upper semicontinuous, and concave on X_i.

(c) The matrix A is *full-row rank*.

Note that Assumptions A1(a) and A1(b) are standard in convex optimization, which guarantees the solvability of the primal-dual problems and strong duality. Assumption A1(c) is not restrictive since it can be guaranteed by applying standard linear algebra techniques to eliminate redundant constraints.

Under Assumption A1, the solution set Y^* of the dual problem (2.1) is nonempty, convex, and bounded. Moreover, *strong duality* holds, i.e.,

$$\begin{aligned} d_0^* = d_0(y_0^*) &= \min_{y \in \mathbb{R}^m} d_0(y) = \max_{x \in X} \{\phi(x) \mid Ax = b\} \\ &= \phi(x_0^*) = \phi^* \quad \forall (x_0^*, y_0^*) \in X^* \times Y^*. \end{aligned}$$

Finally, we note that the dual function $d_0(\cdot)$ can be computed separately by

$$\begin{aligned} d_0(y) &= \sum_{i=1}^M d_{0,i}(y), \text{ where} \\ d_{0,i}(y) &:= \max_{x_i \in X_i} \left\{\phi_i(x_i) + y^T(A_i x_i - b_i)\right\}, \quad i = 1, \dots, M. \end{aligned} \tag{2.3}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

We denote by $x^*_{0,i}(y)$ a solution of the maximization problem in (2.3) for $i = 1, \ldots, M$ and $x^*_0(y) := (x^*_{0,1}(y), \ldots, x^*_{0,M}(y))$.

3. Smoothing via self-concordant barriers. Let us assume that the feasible set X_i possesses a ν_i-self-concordant barrier F_i for $i = 1, \ldots, M$; see [17, 15]. In other words, we make the following assumption.

Assumption A2. For each $i \in \{1, \ldots, M\}$, the feasible set X_i is bounded in $\mathbb{R}^{n_i}$ with $\mathrm{int}(X_i) \neq \emptyset$ and possesses a self-concordant barrier F_i with a parameter $\nu_i > 0$.

The assumption on the boundedness of X_i is not restrictive. In principle, we can bound the set of desired solutions by a sufficiently large compact set such that all the sample points generated by a given optimization algorithm belong to this set.

Let us denote by x^c_i the *analytic center* of X_i, which is defined as

$$x^c_i := \mathrm{argmin}\left\{F_i(x_i) \mid x_i \in \mathrm{int}(X_i)\right\}, \quad i = 1, \ldots, M.$$

Under Assumption A2, $x^c := (x^c_1, \ldots, x^c_M)$ is well-defined due to [18, Corollary 2.3.6]. To compute x^c, one can apply the algorithms proposed in [15, pp. 204–205]. Moreover, the following estimates hold:

$$F_i(x_i) - F_i(x^c_i) \geq \omega(\|x_i - x^c_i\|_{x^c_i}) \quad \text{and} \quad \|x_i - x^c_i\|_{x^c_i} \leq \nu_i + 2\sqrt{\nu_i} \tag{3.1}$$

for all $x_i \in \overline{\mathrm{dom}}(F_i)$ and $i = 1, \ldots, M$; see [15, Theorems 4.1.13 and 4.2.6].

3.1. A smooth approximation of the dual function. Let us define the following function:

$$\begin{aligned} d(y;t) &:= \sum_{i=1}^{M} d_i(y;t), \\ d_i(y;t) &:= \max_{x_i \in \mathrm{int}(X_i)} \left\{\phi_i(x_i) + y^T(A_i x_i - b_i) - t[F_i(x_i) - F_i(x^c_i)]\right\}, \end{aligned} \tag{3.2}$$

where $t > 0$ is referred to as a *smoothness* or *penalty* parameter for $i = 1, \ldots, M$. Similarly as in [10, 14, 21, 28], we can show that $d(\cdot; t)$ is well-defined and smooth due to strict convexity of F_i. We denote by $x_i(y;t)$ the unique solution of the maximization problems in (3.2) for $i = 1, \ldots, M$ and $x^*(y;t) := (x^*_1(y;t), \ldots, x^*_M(y;t))$. We refer to $d(\cdot;t)$ as a *smoothed dual function* of d_0 and to the maximization problems in (3.2) as *primal subproblems*. The optimality condition for the primal subproblem (3.2) is

$$0 \in \partial\phi_i(x^*_i(y;t)) + A^T_i y - t\nabla F_i(x^*_i(y;t)), \quad i = 1, \ldots, M, \tag{3.3}$$

where $\partial\phi_i(x^*_i(y;t))$ is the superdifferential of ϕ_i at $x^*_i(y;t)$. Since problem (3.2) is unconstrained and convex, the condition (3.3) is necessary and sufficient for optimality.

Associated with $d(\cdot;t)$, we consider the following smoothed dual *master problem*:

$$d^*(t) := \min_{y \in Y} d(y;t). \tag{3.4}$$

We denote by $y^*(t)$ a solution of (3.4) if it exists and $x^*(t) := x^*(y^*(t);t)$.

Let $F(x) := \sum_{i=1}^{M} F_i(x_i)$. Then the function F is also a self-concordant barrier of X with a parameter $\nu := \sum_{i=1}^{M} \nu_i$; see [17, Proposition 2.3.1(iii)]. For a given $\beta \in (0,1)$, we define a neighborhood in $\mathbb{R}^m$ w.r.t. F and $t > 0$ as

$$\mathcal{N}^F_t(\beta) := \left\{y \in \mathbb{R}^m \mid \lambda_{F_i}(x^*_i(y;t)) := \|\nabla F_i(x^*_i(y;t))\|^*_{x^*_i(y;t)} \leq \beta,\ i = 1, \ldots, M\right\}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Since $x^c \in \mathcal{N}_t^F(\beta)$, if $\partial\phi(x^c) \bigcap \text{range} A^T \neq \emptyset$, then $\mathcal{N}_t^F(\beta)$ is nonempty. Let

$$\underline{\omega}(x^*(y;t)) := \sum_{i=1}^{M} \omega\left(\|x_i^*(y;t) - x_i^c\|_{x_i^c}\right)$$

and

$$\bar{\omega}(x^*(y;t)) := \sum_{i=1}^{M} \nu_i \omega^{-1}(\nu_i^{-1}\omega_*(\lambda_{F_i}(x_i^*(y;t)))).$$

The following lemma provides a local estimate for d_0, whose proof can be found in section A.1.

LEMMA 3.1. *Suppose that Assumptions* A1 *and* A2 *are satisfied and* $\beta \in (0,1)$. *Suppose further that* $\partial\phi(x^c) \bigcap \text{range} A^T \neq \emptyset$. *Then the function* $d(\cdot;t)$ *defined by* (3.2) *satisfies*

$$0 \leq t\underline{\omega}(x^*(y;t)) \leq d_0(y) - d(y;t) \leq t\left[\bar{\omega}(x^*(y;t)) + \nu\right] \tag{3.5}$$

for all $y \in \mathcal{N}_t^F(\beta)$. *Consequently, one has*

$$0 \leq d_0(y) - d(y;t) \leq t\left[\bar{\omega}_\beta + \nu\right] \quad \forall y \in \mathcal{N}_t^F(\beta),$$

where $\bar{\omega}_\beta := \sum_{i=1}^{M} \nu_i \omega^{-1}(\nu_i^{-1}\omega_*(\beta))$ *and* ω^{-1} *is the inverse function of* ω.

Lemma 3.1 implies that, for a given $\varepsilon_d > 0$, if we choose $t_f := (\bar{\omega}_\beta + \nu)^{-1}\varepsilon_d$, then $d(y;t_f) \leq d_0(y) \leq d(y;t_f) + \varepsilon_d$ for all $y \in \mathcal{N}_t^F(\beta)$.

Under Assumption A1, the solution set Y^* of the dual problem (2.1) is bounded. Let Y be a compact set in $\mathbb{R}^m$ such that $Y^* \subseteq Y$. We define

$$K_i := \max_{y \in Y} \max_{\xi_i \in \partial\phi_i(x_i^c)} \left\{\|\xi_i + A_i^T y\|_{x_i^c}^*\right\} \in [0, +\infty), \quad i = 1, \ldots, M. \tag{3.6}$$

The following lemma provides a global estimate of the dual function d_0. The proof of this lemma can also be found in section A.2.

LEMMA 3.2. *Suppose that Assumptions* A1 *and* A2 *are satisfied and the constants* K_i, $i = 1, \ldots, M$, *are defined by* (3.6). *Then, for any* $t > 0$, *we have*

$$t\underline{\omega}(x^*(y;t)) \leq d_0(y) - d(y;t) \leq tD_X(t) \quad \forall y \in Y, \tag{3.7}$$

where $D_X(t) := \sum_{i=1}^{M} \bar{\zeta}(K_i;\nu_i,t)$ *and* $\bar{\zeta}(\tau;a,b) := a\left(1 + \max\left\{0, \ln\left(\frac{\tau}{ab}\right)\right\}\right)$.

Consequently, for a given tolerance $\varepsilon_d > 0$ *and a constant* $\kappa \in (0,1)$ (*e.g.,* $\kappa = 0.5$), *if*

$$0 < t \leq \bar{t} := \min_{1 \leq i \leq M} \left\{ \frac{K_i}{\nu_i}\kappa^{1/\kappa}, \left(\frac{\varepsilon_d}{\sum_{i=1}^{M}[\nu_i + \nu_i^{1-\kappa}K_i^\kappa]}\right)^{1/(1-\kappa)} \right\}, \tag{3.8}$$

then $d(y;t) \leq d_0(y) \leq d(y;t) + \varepsilon_d$ *for all* $y \in Y$.

If we choose $\kappa = 0.5$, then the estimate (3.8) becomes

$$0 < t \leq \bar{t} := \min_{1 \leq M} \left\{ 0.25\nu_i^{-1}K_i, \left[\varepsilon_d\left(\sum_{i=1}^{M}(\nu_i + \sqrt{\nu_i K_i})\right)^{-1}\right]^2 \right\}.$$

Lemma 3.2 shows that if we fix $t_f \in (0, \bar{t}]$ and minimize $d(\cdot;t_f)$ over Y, then the obtained solution $y^*(t_f)$ is an ε_d-solution of (2.1). Since $d(\cdot;t_f)$ is continuously differentiable, smooth optimization techniques such as gradient-based methods can be applied to minimize $d(\cdot;t_f)$ over Y.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

3.2. The self-concordance of the smoothed dual function. If the function $-\phi_i$ is self-concordant on $\mathrm{dom}(-\phi_i)$ with a parameter κ_{ϕ_i}, then the family of the functions $\phi_i(\cdot;t) := tF(\cdot) - \phi_i(\cdot)$ is also self-concordant on $\mathrm{dom}(-\phi_i) \cap \mathrm{dom}(F_i)$. Consequently, the smooth dual function $d(\cdot;t)$ is self-concordant due to Legendre transformation, as stated in the following lemma; see, e.g., [12, 14, 21, 28].

LEMMA 3.3. *Suppose that Assumptions* A1 *and* A2 *are satisfied. Suppose further that $-\phi_i$ is κ_{ϕ_i}-self-concordant. Then, for $t > 0$, the function $d_i(\cdot;t)$ defined by* (3.2) *is self-concordant with the parameter $\kappa_{d_i} := \max\{\kappa_{\phi_i}, 2/\sqrt{t}\}$, $i = 1, \ldots, M$. Consequently, $d(\cdot;t)$ is self-concordant with the parameter $\kappa_d := \max_{1\le i\le M} \kappa_{d_i}$.*

Similarly as in standard path-following methods [17, 15], in the following discussion, we assume that ϕ_i is linear, as stated in Assumption A3.

Assumption A3. The function ϕ_i is linear, i.e., $\phi_i(x_i) := c_i^T x_i$ for $i = 1, \ldots, M$.

Let $c := (c_1, \ldots, c_M)$ be a column vector formed from c_i $(i = 1, \ldots, M)$. Assumption A3 and Lemma 3.3 imply that $d(\cdot;t)$ is $\frac{2}{\sqrt{t}}$-self-concordant. Since ϕ_i is linear, the optimality condition (3.3) is rewritten as

$$c + A^T y - t\nabla F(x^*(y;t)) = 0. \tag{3.9}$$

The following lemma provides explicit formulas for computing the derivatives of $d(\cdot;t)$. The proof can be found in [14, 28].

LEMMA 3.4. *Suppose that Assumptions* A1, A2, *and* A3 *are satisfied. Then the gradient vector and the Hessian matrix of $d(\cdot,t)$ on Y are given, respectively, as*

$$\nabla d(y;t) = Ax^*(y;t) - b \quad \text{and} \quad \nabla^2 d(y;t) = t^{-1}A\nabla^2 F(x^*(y;t))^{-1}A^T, \tag{3.10}$$

where $x^*(y;t)$ is the solution vector of the primal subproblem (3.2).

Note that since A is full-row rank and $\nabla^2 F(x^*(y;t)) \succ 0$, we can see that $\nabla^2 d(y;t) \succ 0$ for any $y \in Y$. Now, since $d(\cdot;t)$ is $\frac{2}{\sqrt{t}}$ self-concordant, if we define

$$\tilde{d}(y;t) := t^{-1}d(y;t), \tag{3.11}$$

then $\tilde{d}(\cdot;t)$ is standard self-concordant, i.e., $\kappa_{\tilde{d}} = 2$, due to [15, Corollary 4.1.2]. For a given vector $v \in \mathbb{R}^m$, we define the local norm $\|v\|_y$ of v w.r.t. $\tilde{d}(\cdot;t)$ as $\|v\|_y := [v^T\nabla^2\tilde{d}(y;t)v]^{1/2}$.

3.3. Optimality and feasibility recovery. It remains to show the relations between the master problem (3.4), the dual problem (2.1), and the original primal problem (SCPP). We first prove the following lemma.

LEMMA 3.5. *Let Assumptions* A1, A2, *and* A3 *be satisfied. Then the following hold:*

(a) *For a given $y \in Y$, $d(y;\cdot)$ is nonincreasing in $\mathbb{R}_{++}$.*
(b) *The function d^* defined by* (3.4) *is nonincreasing and differentiable in $\mathbb{R}_{++}$. Moreover, $d^*(t) \le d_0^* = \phi^*$ and $\lim_{t\downarrow 0^+} d^*(t) = \phi^*$.*
(c) *The point $x^*(t) := x^*(y^*(t);t)$ is feasible to* (SCPP) *and $\lim_{t\downarrow 0^+} x^*(t) = x_0^* \in X^*$.*

Proof. Since the function $\xi(x,y;t) := \phi(x) + y^T(Ax-b) - t[F(x) - F(x^c)]$ is strictly concave in x and linear in t, it is well known that $d(y;t) = \max\{\xi(x,y;t) \mid x \in \mathrm{int}(X)\}$ is differentiable w.r.t. t and its derivative is given by $\frac{\partial d(y;t)}{\partial t} = -[F(x^*(y;t)) - F(x^c)] \le -\omega(\|x^*(y;t) - x^c\|_{x^c}) \le 0$ due to (3.1). Thus $d(y,\cdot)$ is nonincreasing in t, as stated in

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(a). From the definitions of d^*, $d(y,\cdot)$, and y^* in (3.4) and strong duality, we have

$$\begin{aligned} d^*(t) &= \min_{y\in Y} d(y;t) \overset{\text{strong duality}}{=} \max_{x\in \text{int}(X)} \min_{y\in Y} \left\{\phi(x) + y^T(Ax-b) - t[F(x)-F(x^c)]\right\} \\ &= \max_{x\in \text{int}(X)} \{\phi(x) - t[F(x)-F(x^c)] \mid Ax = b\} \\ &= \phi(x^*(t)) - t[F(x^*(t)) - F(x^c)]. \end{aligned} \tag{3.12}$$

It follows from the second line of (3.12) that d^* is differentiable and nonincreasing in $\mathbb{R}_{++}$. From the second line of (3.12), we also deduce that $x^*(t)$ is feasible to (SCPP). The limit in (c) was proved in [28, Proposition 2]. Since $x^*(t)$ is feasible to (SCPP) and $F(x^*(t)) - F(x^c) \geq 0$, the last line of (3.12) implies that $d^* \leq d_0^*$. We also obtain the limit $\lim_{t\downarrow 0^+} d^*(t) = d_0^* = \phi^*$. ∎

Let us define the Newton decrement of $\tilde{d}(\cdot;t)$ as follows:

$$\lambda = \lambda_{\tilde{d}(\cdot;t)}(y) := \|\nabla \tilde{d}(y;t)\|_y^* = \left[\nabla \tilde{d}(y;t) \nabla^2 \tilde{d}(y;t)^{-1} \nabla \tilde{d}(y;t)\right]^{1/2}. \tag{3.13}$$

The following lemma shows the gap between $d(y;t)$ and $d^*(t)$.

LEMMA 3.6. *Suppose that Assumptions* A1, A2, *and* A3 *are satisfied. Then, for any $y \in Y$ and $t > 0$ such that $\lambda_{\tilde{d}(\cdot;t)}(y) \leq \beta < 1$, we have*

$$0 \leq t\omega(\lambda_{\tilde{d}(\cdot;t)}(y)) \leq d(y;t) - d^*(t) \leq t\omega_*(\lambda_{\tilde{d}(\cdot;t)}(y)). \tag{3.14}$$

Moreover, it holds that

$$(c + A^T y)^T (u - x^*(y;t)) \leq t\nu \quad \text{and} \quad \|Ax^*(y;t) - b\|_y^* \leq t\beta \tag{3.15}$$

for all $u \in X$.

Proof. Since $\tilde{d}(\cdot;t)$ is standard self-concordant and $y^*(t) = \text{argmin}\{\tilde{d}(y;t) \mid y \in Y\}$, for any $y \in Y$ such that $\lambda \leq \beta < 1$, by applying [15, Theorem 4.1.13, inequality 4.1.17], we have $0 \leq \omega(\lambda) \leq \tilde{d}(y;t) - \tilde{d}(y^*(t);t) \leq \omega_*(\lambda)$. By (3.11), these inequalities are equivalent to (3.14). It follows from the optimality condition (3.9) that $c + A^T y = t\nabla F(x^*(y;t))$. Hence, by [15, Theorem 4.2.4], we have $(c + A^T y)^T (u - x^*(y;t)) = t\nabla F(x^*(y;t))^T (u - x^*(y;t)) \leq t\nu$ for any $u \in \text{dom} F$. Since $X \subseteq \text{dom} F$, the last inequality implies the first condition in (3.15). Furthermore, from (3.10) we have $\nabla d(y;t) = Ax^*(y;t) - b$. Therefore, $\|Ax^*(y;t) - b\|_y^* = t\|\nabla \tilde{d}(y^*(t);t)\|_y^* = t\lambda_{\tilde{d}(\cdot;t)}(y) \leq t\beta$. ∎

Let us recall the optimality condition for the primal-dual problems (SCPP) and (2.1) as

$$0 \in c + A^T y_0^* - \mathcal{N}_X(x_0^*) \quad \text{and} \quad Ax_0^* - b = 0 \;\; \forall (x_0^*, y_0^*) \in \mathbb{R}^n \times \mathbb{R}^m, \tag{3.16}$$

where $\mathcal{N}_X(x)$ is the normal cone of X at x. Here, since X^* is nonempty, the first inclusion also covers implicitly that $x_0^* \in X$. Moreover, if $x_0^* \in X$, then (3.16) can be expressed equivalently as $(c + A^T y_0^*)^T (u - x_0^*) \leq 0$ for all $u \in X$. Now, we define an approximate solution of (SCPP) and (2.1) as follows.

DEFINITION 3.7. *For a given tolerance $\varepsilon_p \in [0,1)$, a point $(\tilde{x}^*, \tilde{y}^*) \in X \times \mathbb{R}^m$ is said to be an ε_p-solution of* (SCPP) *and* (2.1) *if $(c + A^T \tilde{y}^*)^T (u - \tilde{x}^*) \leq \varepsilon_p$ for all $u \in X$ and $\|A\tilde{x}^* - b\|_{\tilde{y}^*}^* \leq \varepsilon_p$.*

It is clear that for any point $x \in \text{int}(X)$, $\mathcal{N}_X(x) = \{0\}$. Furthermore, according to (3.16), the conditions in Definition 3.7 are well-defined.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Finally, we note that $\nu \geq 1$, $\beta < 1$, and $x^*(y;t) \in \mathrm{int}(X)$. By (3.15), if we choose the tolerance $\varepsilon_p := \nu t$, then $(x^*(y;t), y)$ is an ε_p-solution of (SCPP) and (2.1) in the sense of Definition 3.7. We denote the feasibility gap by $\mathcal{F}(y;t) := \|Ax^*(y;t) - b\|_y^*$ for further references.

4. Inexact perturbed path-following method. This section presents an inexact perturbed path-following decomposition algorithm for solving (2.1).

4.1. Inexact solution of the primal subproblems. First, we define an inexact solution of (3.2) by using local norms. For a given $y \in Y$ and $t > 0$, suppose that we solve (3.2) approximately up to a given accuracy $\bar{\delta} \geq 0$. More precisely, we define this approximation as follows.

DEFINITION 4.1. *For given $\bar{\delta} \geq 0$, a vector $\bar{x}_{\bar{\delta}}(y;t)$ is said to be a $\bar{\delta}$-approximate solution of $x^*(y;t)$ if*

$$\|\bar{x}_{\bar{\delta}}(y;t) - x^*(y;t)\|_{x^*(y;t)} \leq \bar{\delta}. \tag{4.1}$$

Associated with $\bar{x}_{\bar{\delta}}(\cdot)$, we define the function

$$d_{\bar{\delta}}(y;t) := c^T\bar{x}_{\bar{\delta}}(y;t) + y^T(A\bar{x}_{\bar{\delta}}(y;t) - b) - t[F(\bar{x}_{\bar{\delta}}(y;t)) - F(x^c)]. \tag{4.2}$$

This function can be considered as an *inexact version* of d. Next, we introduce two quantities

$$\nabla d_{\bar{\delta}}(y;t) := A\bar{x}_{\bar{\delta}}(y;t) - b \quad \text{and} \quad \nabla^2 d_{\bar{\delta}}(y;t) := t^{-1}A\nabla^2 F(\bar{x}_{\bar{\delta}}(y;t))^{-1}A^T. \tag{4.3}$$

Since $x^*(y;t) \in \mathrm{dom}(F)$, we can choose an appropriate $\bar{\delta} \geq 0$ such that $\bar{x}_{\bar{\delta}}(y;t) \in \mathrm{dom}(F)$. Hence, $\nabla^2 F(\bar{x}_{\bar{\delta}}(y;t))$ is positive definite, which means that $\nabla^2 d_{\bar{\delta}}$ is well-defined. Note that $\nabla d_{\bar{\delta}}$ and $\nabla^2 d_{\bar{\delta}}$ are not the gradient vector and Hessian matrix of $d_{\bar{\delta}}(\cdot;t)$. However, due to Lemma 3.4 and (4.1), we can consider these quantities as an approximate gradient vector and Hessian matrix of $d(\cdot;t)$, respectively.

Let

$$\tilde{d}_{\bar{\delta}}(y;t) := t^{-1}d_{\bar{\delta}}(y;t), \tag{4.4}$$

and let $\bar{\lambda}$ be the inexact Newton decrement of $\tilde{d}_\delta$ which is defined by

$$\bar{\lambda} = \bar{\lambda}_{\tilde{d}_{\bar{\delta}}(\cdot;t)}(y) := \|\!|\nabla\tilde{d}_{\bar{\delta}}(y;t)|\!\|_y^* = \left[\nabla\tilde{d}_{\bar{\delta}}(y;t)\nabla^2\tilde{d}_{\bar{\delta}}(y;t)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y;t)\right]^{1/2}. \tag{4.5}$$

Here, we use the norm $\|\!|\cdot|\!\|_y$ to distinguish it from $\|\cdot\|_y$.

4.2. The algorithmic framework. From Lemma 3.6 we see that if we can generate a sequence $\{(y^k, t_k)\}_{k\geq 0}$ such that $\lambda_k := \lambda_{\tilde{d}(\cdot,t_k)}(y^k) \leq \beta < 1$, then

$$d(y^k;t_k) \uparrow d_0^* = \phi^* \quad \text{and} \quad \mathcal{F}(y^k;t_k) \to 0 \quad \text{as } t_k \downarrow 0^+.$$

Therefore, the aim of the algorithm is to generate $\{(y^k, t_k)\}_{k\geq 0}$ such that $\lambda_k \leq \beta < 1$ and $t_k \downarrow 0^+$. First, we fix $t = t_0 > 0$ and find a point $y^0 \in Y$ such that $\lambda_{\tilde{d}(\cdot;t_0)}(y^0) \leq \beta$. Then we simultaneously update y^k and t_k to control $t_k \downarrow 0^+$. The algorithmic framework is presented as follows.

INEXACT-PERTURBED PATH-FOLLOWING ALGORITHMIC FRAMEWORK.

Initialization. Choose an appropriate $\beta \in (0,1)$ and a tolerance $\varepsilon_d > 0$. Fix $t := t_0 > 0$.

Phase 1. (*Determine a starting point $y^0 \in Y$ such that $\lambda_{\tilde{d}(\cdot;t_0)}(y^0) \leq \beta$*).

Choose an initial vector $y^{0,0} \in Y$.

For $j = 0, 1, \ldots, j_{\max}$, perform the following steps:

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

1. If $\lambda_j := \lambda_{\tilde{d}(\cdot;t_0)}(y^{0,j}) \leq \beta$, then set $y^0 := y^{0,j}$ and terminate.
2. Solve (3.2) *in parallel* to obtain an approximation solution of $x^*(y^{0,j}, t_0)$.
3. Evaluate $\nabla d_{\bar{\delta}}(y^{0,j}, t_0)$ and $\nabla^2 d_{\bar{\delta}}(y^{0,j}, t_0)$ by using (4.3).
4. Perform the inexact perturbed damped Newton step: $y^{0,j+1} := y^{0,j} - \alpha_j \nabla^2 d_{\bar{\delta}}(y^{0,j}, t_0)^{-1} \nabla d_{\bar{\delta}}(y^{0,j}, t_0)$, where $\alpha_j \in (0, 1]$ is a given step size.

End For.

Phase 2. (*Path-following iterations*).

Compute an appropriate value $\sigma \in (0, 1)$.

For $k = 0, 1, \ldots, k_{\max}$, perform the following steps:

1. If $t_k \leq \varepsilon_d / \omega_*(\beta)$, then terminate.
2. Update $t_{k+1} := (1 - \sigma) t_k$.
3. Solve (3.2) *in parallel* to obtain an approximation solution of $x^*(y^k; t_{k+1})$.
4. Evaluate the quantities $\nabla d_{\bar{\delta}}(y^k; t_{k+1})$ and $\nabla^2 d_{\bar{\delta}}(y^k; t_{k+1})$ as in (4.3).
5. Perform the inexact perturbed full-step Newton step as $y^{k+1} := y^k - \nabla^2 d_{\bar{\delta}}(y^k; t_{k+1})^{-1} \nabla d_{\bar{\delta}}(y^k, t_{k+1})$.

End For.

Output. An ε_d-approximate solution y^k of (3.4), i.e., $0 \leq d(y^k; t_k) - d^*(t_k) \leq \varepsilon_d$.

END.

This algorithm is still conceptual. In the following subsections, we shall discuss each step of this algorithmic framework in detail. We note that the proposed algorithm provides an ε_d-approximate solution y^k such that $t_k \leq \varepsilon_t := \omega_*(\beta)^{-1} \varepsilon_d$. Now, by solving the primal subproblem (3.2), we obtain $x^*(y^k; t_k)$ as an ε_p-solution of (SCPP) in the sense of Definition 3.7, where $\varepsilon_p := \nu \varepsilon_t$.

4.3. Computing inexact solutions. The condition (4.1) cannot be used in practice to compute $\bar{x}_{\bar{\delta}}$ since $x^*(y; t)$ is unknown. We need to show how to compute $\bar{x}_{\bar{\delta}}$ practically such that (4.1) holds.

For notational simplicity, we denote $\bar{x}_{\bar{\delta}} := \bar{x}_{\bar{\delta}}(y; t)$ and $x^* := x^*(y; t)$. The error of the approximate solution $\bar{x}_{\bar{\delta}}$ to x^* is defined as

$$\delta(\bar{x}_{\bar{\delta}}, x^*) := \|\bar{x}_{\bar{\delta}}(y; t) - x^*(y; t)\|_{x^*(y;t)}. \tag{4.6}$$

The following lemma gives a criterion to ensure that the condition (4.1) holds.

LEMMA 4.2. *Let $\delta(\bar{x}_{\bar{\delta}}, x^*)$ be defined by* (4.6) *such that $\delta(\bar{x}_{\bar{\delta}}, x^*) < 1$. Then*

$$0 \leq t\omega(\delta(\bar{x}_{\bar{\delta}}, x^*)) \leq d(y; t) - d_{\bar{\delta}}(y; t) \leq t\omega_*(\delta(\bar{x}_{\bar{\delta}}, x^*)). \tag{4.7}$$

Moreover, if

$$E_{\bar{\delta}}^c := \|c + A^T y - t\nabla F(\bar{x}_{\bar{\delta}})\|_{x^c}^* \leq \varepsilon_d := \left[(\nu + 2\sqrt{\nu})(1 + \bar{\delta})\right]^{-1} \bar{\delta} t, \tag{4.8}$$

then $\bar{x}_{\bar{\delta}}(y; t)$ satisfies (4.1). *Consequently, if $t \leq \omega_*(\beta)^{-1} \varepsilon_d$ and $\bar{\delta} < 1$, then*

$$|d_{\bar{\delta}}(y; t) - d^*(t)| \leq \left[1 + \omega_*(\beta)^{-1} \omega_*(\bar{\delta})\right] \varepsilon_d. \tag{4.9}$$

Proof. It follows from the definitions of $d(\cdot; t)$ and $d_{\bar{\delta}}(\cdot; t)$ and (3.9) that

$$\begin{aligned} d(y; t) - d_{\bar{\delta}}(y; t) &= [c + A^T y](x^* - \bar{x}_{\bar{\delta}}) - t[F(x^*) - F(\bar{x}_{\bar{\delta}})] \\ &= -t[F(x^*) + \nabla F(x^*)^T (\bar{x}_{\bar{\delta}} - x^*) - F(\bar{x}_{\bar{\delta}})]. \end{aligned}$$

Since F is self-concordant, by applying [15, Theorems 4.1.7 and 4.1.8] and the definition of $\delta(\bar{x}_{\bar{\delta}}, x^*)$, the above equality implies

$$0 \leq t\omega(\delta(\bar{x}_{\bar{\delta}}, x^*)) \leq d(y; t) - d_{\bar{\delta}}(y; t) \leq t\omega_*(\delta(\bar{x}_{\bar{\delta}}, x^*)),$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

which is indeed (4.7).

Next, by again using (3.9) and the definition of $E^c_{\bar{\delta}}$ we have

$$E^c_{\bar{\delta}} \overset{(3.9)}{=} t\|\nabla F(\bar{x}_{\bar{\delta}}) - \nabla F(x^*)\|^*_{x^c} \geq (\nu + 2\sqrt{\nu})^{-1} t\|\nabla F(\bar{x}_{\bar{\delta}}) - \nabla F(x^*)\|^*_{x^*},$$

where the last inequality follows from [15, Corollary 4.2.1]. Combining this inequality and [15, Theorem 4.1.7], we obtain

$$\begin{aligned}\frac{\delta(\bar{x}_{\bar{\delta}}, x^*)^2}{1+\delta(\bar{x}_{\bar{\delta}}, x^*)} &\leq \left[\nabla F(\bar{x}_{\bar{\delta}}) - \nabla F(x^*)\right]^T (\bar{x}_{\bar{\delta}} - x^*) \leq \|\nabla F(\bar{x}_{\bar{\delta}}) - \nabla F(x^*)\|^*_{x^*}\|\bar{x}_{\bar{\delta}} - x^*\|_{x^*} \\ &\leq t^{-1}(\nu + 2\sqrt{\nu})E^c_{\bar{\delta}}\delta(\bar{x}_{\bar{\delta}}, x^*).\end{aligned}$$

Hence, we get

$$\delta(\bar{x}_{\bar{\delta}}, x^*) \leq \left[t - (\nu + 2\sqrt{\nu})E^c_{\bar{\delta}}\right]^{-1}(\nu + 2\sqrt{\nu})E^c_{\bar{\delta}}, \tag{4.10}$$

provided that $t > (\nu+2\sqrt{\nu})E^c_{\bar{\delta}}$. Let us define an accuracy ε_p for the primal subproblem (3.2) as $\varepsilon_p := [(\nu + 2\sqrt{\nu})(1 + \bar{\delta})]^{-1}\bar{\delta}t \geq 0$. Then it follows from (4.10) that if $E^c_{\bar{\delta}} \leq [(\nu+2\sqrt{\nu})(1+\bar{\delta})]^{-1}\bar{\delta}t$, then $\bar{x}_{\bar{\delta}}(y;t)$ satisfies (4.1). It remains to consider the distance from d_δ to $d^*(t)$ when t is sufficiently small. Suppose that $t \leq \omega_*(\beta)^{-1}\varepsilon_d$. Then, by combining (3.14) and (4.7) we obtain (4.9). □

Remark 1. Since $E_{\bar{\delta}} := \|c + A^T y - t\nabla F(\bar{x}_{\bar{\delta}})\|^*_{\bar{x}_{\bar{\delta}}} \geq (1-\bar{\delta})\|c + A^T y - t\nabla F(\bar{x}_{\bar{\delta}})\|^*_{x^*}$, by the same argument as in the proof of Lemma 4.2, we can show that if $E_{\bar{\delta}} \leq \hat{\varepsilon}_p$, where $\hat{\varepsilon}_p := \frac{\bar{\delta}(1-\bar{\delta})t}{1+\delta}$, then (4.1) holds. This condition can be used to terminate the algorithm presented in the next section.

4.4. Phase 2: The path-following scheme with inexact perturbed full-step Newton iterations. Now, we analyze steps 2 to 5 in Phase 2 of the algorithmic framework. In the path-following fashion, we perform only one inexact perturbed full-step Newton (IPFNT) iteration for each value of the parameter t. This iteration of this scheme is specified as follows:

$$\begin{cases} t_+ := t - \Delta_t, \\ y_+ := y - \nabla^2 d_{\bar{\delta}}(y; t_+)^{-1}\nabla d_{\bar{\delta}}(y; t_+). \end{cases} \tag{4.11}$$

Since the Newton-type method is invariant under linear transformations, by (4.2), the second line of (4.11) is equivalent to

$$y_+ := y - \nabla^2 \tilde{d}_{\bar{\delta}}(y; t_+)^{-1}\nabla \tilde{d}_{\bar{\delta}}(y; t_+). \tag{4.12}$$

For the sake of notational simplicity, we denote all the functions at (y_+, t_+) and $(y; t_+)$ by the subindexes "$_+$" and "$_1$," respectively, and at $(y; t)$ without index in the following analysis. More precisely, we denote

$$\begin{array}{llll} \bar{\lambda}_+ & := \bar{\lambda}_{\tilde{d}_{\bar{\delta}}(\cdot;t_+)}(y_+), & \delta_+ & := \|\bar{x}_{\bar{\delta}}(y_+, t_+) - x^*(y_+, t_+)\|_{x^*(y_+,t_+)}, \\ \bar{\lambda}_1 & := \bar{\lambda}_{\tilde{d}_{\bar{\delta}}(\cdot;t_+)}(y), & \delta_1 & := \|\bar{x}_{\bar{\delta}}(y; t_+) - x^*(y; t_+)\|_{x^*(y;t_+)}, \\ \bar{\lambda} & := \bar{\lambda}_{\tilde{d}_{\bar{\delta}}(\cdot;t)}(y), & \delta & := \|\bar{x}_{\bar{\delta}}(y; t) - x^*(y; t)\|_{x^*(y;t)} \end{array} \tag{4.13}$$

and

$$\Delta := \|\bar{x}_{\bar{\delta}}(y; t_+) - \bar{x}_{\bar{\delta}}(y; t)\|_{\bar{x}_{\bar{\delta}}(y;t)} \quad \text{and} \quad \Delta^* := \|x^*(y; t_+) - x^*(y; t)\|_{x^*(y;t)}. \tag{4.14}$$

Note that the above notation does not cause any confusion since it can be recognized from the context.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

4.4.1. The main estimate. Now, by using the notation in (4.13) and (4.14), we provide a main estimate which will be used to analyze the convergence of the algorithm in subsection 4.4.4. The proof of this lemma can be found in section A.3.

LEMMA 4.3. *Let $y \in Y$ and $t > 0$ be given and (y_+, t_+) be a pair generated by* (4.11). *Let $\xi := \frac{\Delta+\bar{\lambda}}{1-\delta_1-2\Delta-\bar{\lambda}}$. Suppose that $\delta_1 + 2\Delta + \bar{\lambda} < 1$, $\delta_+ < 1$. Then*

$$\bar{\lambda}_+ \le (1-\delta_+)^{-1}\Big\{\delta_+ + \delta_1 + \xi^2 + \delta_1\left[(1-\delta_1)^{-2} + 2(1-\delta_1)^{-1}\right]\xi\Big\}. \tag{4.15}$$

Moreover, the right-hand side of (4.15) *is nondecreasing w.r.t. all variables δ_+, δ_1, Δ, and $\bar{\lambda}$.*

In particular, if we set $\delta_+ = 0$ and $\delta_1 = 0$, i.e., the primal subproblem (3.2) *is assumed to be solved exactly, then $\bar{\lambda}_+ = \lambda_+$, $\bar{\lambda} = \lambda$, and* (4.15) *reduces to*

$$\lambda_+ \le (1 - 2\Delta^* - \lambda)^{-2}(\lambda + \Delta^*)^2, \tag{4.16}$$

provided that $\lambda + 2\Delta^* < 1$.

4.4.2. Maximum neighborhood of the central path. The key point of the path-following algorithm is to determine the maximum neighborhood $(\beta_*, \beta^*) \subseteq (0,1)$ of the central path such that for any $\beta \in (\beta_*, \beta^*)$, if $\bar{\lambda} \le \beta$, then $\bar{\lambda}_+ \le \beta$. Now, we analyze the estimate (4.15) to find $\bar{\delta}$ and Δ such that the last condition holds.

Suppose that $\bar{\delta} \ge 0$ as in Definition 4.1. First, we construct the following parametric cubic polynomial:

$$\mathcal{P}_{\bar{\delta}}(\beta) := c_0(\bar{\delta}) + c_1(\bar{\delta})\beta + c_2(\bar{\delta})\beta^2 + c_3(\bar{\delta})\beta^3, \tag{4.17}$$

where the coefficients are given by $c_0(\bar{\delta}) := -2\bar{\delta}(1-\bar{\delta})^2 \le 0$, $c_1(\bar{\delta}) := (1-\bar{\delta})^{-1}[1 - 3\bar{\delta} + \bar{\delta}^4]$, $c_2(\bar{\delta}) := \bar{\delta}[(1-\bar{\delta})^{-2} + 2(1-\bar{\delta})^{-1}] - 3 + 2\bar{\delta}(1-\bar{\delta})$, and $c_3(\bar{\delta}) := 1 - \bar{\delta} > 0$. Then we define

$$p := \bar{\delta}[(1-\bar{\delta})^{-2} + 2(1-\bar{\delta})^{-1}], \quad q := (1-\bar{\delta})\beta - 2\bar{\delta}, \quad \theta := 0.5(\sqrt{p^2+4q} - p). \tag{4.18}$$

The following theorem provides the conditions such that if $\bar{\lambda} \le \beta$, then $\bar{\lambda}_+ \le \beta$.

THEOREM 4.4. *Let $\bar{\delta}_{\max} := 0.043286$. Suppose that $\bar{\delta} \in [0, \bar{\delta}_{\max}]$ is fixed and θ is defined by* (4.18). *Then the polynomial $\mathcal{P}_{\bar{\delta}}$ defined by* (4.17) *has three nonnegative real roots $0 \le \beta_* < \beta^* < \beta_3$. Moreover, if we choose $\beta \in (\beta_*, \beta^*)$ and compute $\bar{\Delta} := \frac{\theta(1-\bar{\delta}-\beta)-\beta}{1+2\theta}$, then $\bar{\Delta} > 0$ and, for $0 \le \delta_+ \le \bar{\delta}$, $0 \le \delta_1 \le \bar{\delta}$, and $0 \le \Delta \le \bar{\Delta}$, the condition $\bar{\lambda} \le \beta$ implies $\bar{\lambda}_+ \le \beta$.*

The proof of this theorem is postponed to section A.3. Now, we illustrate the variation of the values of β_*, β^*, and $\bar{\Delta}$ w.r.t. $\bar{\delta}$ in Figure 4.1. The left figure shows the values of β_* (solid) and β^* (dashed), and the right one plots the value of $\bar{\Delta}$ when β is chosen by $\beta := \frac{\beta_*+\beta^*}{2}$ (dashed) and $\beta := \frac{\beta^*}{4}$ (solid), respectively.

4.4.3. The update rule of the penalty parameter. It remains to quantify the decrement Δ_t of the penalty parameter t in (4.11). The following lemma shows how to update t.

LEMMA 4.5. *Let $\bar{\delta}$ and $\bar{\Delta}$ be defined as in Theorem* 4.4, *and let*

$$\bar{\Delta}^* := \frac{1}{2}\Big[(1-\bar{\delta})\bar{\Delta} - \bar{\delta} + 1 - \sqrt{((1-\bar{\delta})\bar{\Delta} - \bar{\delta} - 1)^2 + 4\bar{\delta}}\Big]. \tag{4.19}$$

Then $\bar{\Delta}^* > 0$ and the penalty parameter t can be decreased linearly as $t_+ := (1-\sigma)t$, where $\sigma := [\sqrt{\nu} + \bar{\Delta}^*(\sqrt{\nu}+1)]^{-1}\bar{\Delta}^* \in (0,1)$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

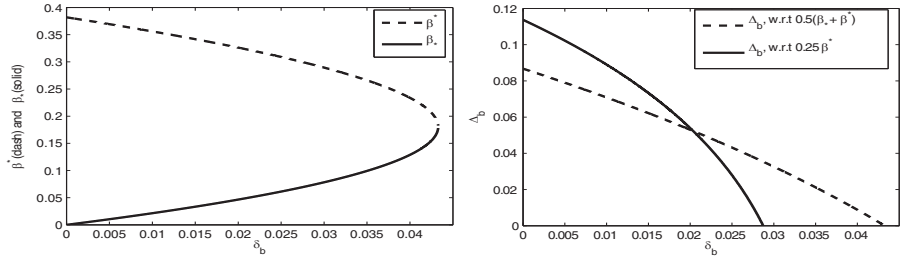

FIG. 4.1. *The values of β_*, β^*, and $\bar{\Delta}$ varying w.r.t. $\bar{\delta}$.*

Proof. It follows from (3.9) that $c+A^Ty-t\nabla F(x^*)=0$ and $c+A^Ty-t_+\nabla F(x_1^*)=0$, where $x^* := x^*(y;t)$ and $x_1^* := x^*(y;t_+)$. Subtracting these equalities and then using $t_+ = t-\Delta_t$, we have $t_+[\nabla F(x_1^*)-\nabla F(x^*)] = \Delta_t \nabla F(x^*)$. Using this relation together with [15, Theorem 4.1.7] and $\|\nabla F(x^*)\|_{x^*}^* \le \sqrt{\nu}$ (see [15, inequality 4.2.4]), we have

$$\begin{aligned}\frac{t_+\|x_1^*-x^*\|_{x^*}^2}{1+\|x_1^*-x^*\|_{x^*}} &\le t_+[\nabla F(x_1^*)-\nabla F(x^*)]^T(x_1^*-x^*) = \Delta_t\nabla F(x^*)^T(x_1^*-x^*)\\ &\le \Delta_t\|\nabla F(x^*)\|_{x^*}^*\|x_1^*-x^*\|_{x^*} \le \Delta_t\sqrt{\nu}\|x_1^*-x^*\|_{x^*}.\end{aligned}$$

By the definition of Δ^* in (4.14), if $t > (\sqrt{\nu}+1)\Delta_t$, then the above inequality leads to

$$\Delta^* \le \bar{\Delta}^* := \big[t-(\sqrt{\nu}+1)\Delta_t\big]^{-1}\sqrt{\nu}\Delta_t. \tag{4.20}$$

Therefore,

$$\Delta_t = t\big[\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}^*\big]^{-1}\bar{\Delta}^*. \tag{4.21}$$

On the other hand, using the definitions of Δ and δ, we have

$$\begin{aligned}\Delta := \|\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}\|_{\bar{x}_{\bar{\delta}}} &\overset{(A.10)}{\le} (1-\delta)^{-1}\Big[\|\bar{x}_{\bar{\delta}1}-x_1^*\|_{x^*}+\|x_1^*-x^*\|_{x^*}+\|x^*-\bar{x}_{\bar{\delta}}\|_{x^*}\Big]\\ &\le (1-\delta)^{-1}\left[(1-\Delta^*)^{-1}\delta_1+\Delta^*+\delta\right] \qquad (4.22)\\ &\overset{(4.20),\delta,\delta_1\le\bar{\delta}}{\le} (1-\bar{\delta})^{-1}\left[(1-\bar{\Delta}^*)^{-1}\bar{\delta}+\bar{\Delta}^*+\bar{\delta}\right].\end{aligned}$$

Now, we need to find a condition such that $\Delta \le \bar{\Delta}$, where $\bar{\Delta}$ is given in Theorem 4.4. It follows from (4.22) that $\Delta \le \bar{\Delta}$ if $\frac{\bar{\delta}}{1-\Delta^*}+\Delta^* \le (1-\bar{\delta})\bar{\Delta}-\bar{\delta}$. The last condition holds if

$$0 \le \bar{\Delta}^* \le \frac{1}{2}\Big[(1-\bar{\delta})\bar{\Delta}-\bar{\delta}+1-\sqrt{((1-\bar{\delta})\bar{\Delta}-\bar{\delta}-1)^2+4\bar{\delta}}\Big]. \tag{4.23}$$

Moreover, by the choice of $\bar{\Delta}$ and $\bar{\delta}$, we have $(1-\bar{\delta})\bar{\Delta}-\bar{\delta}>0$. This implies $\bar{\Delta}^*>0$. Since $\bar{\Delta}^*$ satisfies (4.20), we can fix $\bar{\Delta}^*$ at the upper bound as defined in (4.19) and compute Δ_t according to $\bar{\Delta}^*$ as (4.21). Therefore, (4.21) gives us an update rule for the penalty parameter t, i.e., $t_+ := t-\sigma t = (1-\sigma)t$, where $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu}+\Delta^*(\sqrt{\nu}+1)} \in (0,1)$. ∎

Finally, we show that the conditions given in Theorem 4.4 and Lemma 4.5 are well-defined. Indeed, let us fix $\bar{\delta} := 0.01$. Then we can compute the values of β_* and β^* as $\beta_* \approx 0.021371 < \beta^* \approx 0.356037$. Therefore, if we choose $\beta := \frac{\beta^*}{4} \approx 0.089009 > \beta_*$, then $\bar{\Delta} \approx 0.089012$ and $\bar{\Delta}^* \approx 0.067399$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

4.4.4. The algorithm and its convergence. Before presenting the algorithm, we need to find a stopping criterion. By using Lemma A.1(c) with $\Delta \leftarrow \delta$, we have

$$\lambda \leq (1-\delta)^{-1}(\bar{\lambda}+\delta), \tag{4.24}$$

provided that $\delta < 1$ and $\bar{\lambda} \leq \beta < 1$. Consequently, if $\bar{\lambda} \leq (1-\bar{\delta})\beta - \bar{\delta}$, then $\lambda \leq \beta$. Let us define $\vartheta := (1-\bar{\delta})\beta - \bar{\delta}$, where $0 < \bar{\delta} < \beta/(\beta+1)$. It follows from Lemma 3.6 that if $t\omega_*(\vartheta) \leq \varepsilon_d$ for a given tolerance $\varepsilon_d > 0$, then y is an ε_d-solution of (3.4).

The second phase of the algorithmic framework presented in subsection 4.2 is now described in detail as follows.

ALGORITHM **1**. (*Path-following algorithm with* IPFNT *iterations*).
Initialization: Choose $\bar{\delta} \in [0, \bar{\delta}_{\max}]$, and compute β_* and β^* as in Theorem 4.4.
Phase 1. Apply Algorithm 2 presented in subsection 4.5 below to find $y^0 \in Y$ such that $\lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t_0)}(y^0) \leq \beta$.
Phase 2.
Initialization of Phase 2: Perform the following steps:

1. Given a tolerance $\varepsilon_d > 0$.
2. Compute $\bar{\Delta}$ as in Theorem 4.4. Then compute $\bar{\Delta}^*$ by (4.19).
3. Compute $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}^*}$ and the accuracy factor $\gamma := \frac{\bar{\delta}}{(\nu+2\sqrt{\nu})(1+\bar{\delta})}$.

Iteration: For $k = 0, 1, \ldots, k_{\max}$ perform the following steps:

1. If $t_k \leq \frac{\varepsilon_d}{\omega_*(\vartheta)}$, where $\vartheta := (1-\bar{\delta})\beta - \bar{\delta}$, then terminate.
2. Compute an accuracy $\varepsilon_k := \gamma t_k$ for the primal subproblems.
3. Update $t_{k+1} := (1-\sigma)t_k$.
4. Solve approximately (2.2) *in parallel* up to the accuracy ε_k to obtain $\bar{x}_{\bar{\delta}}(y^k; t_{k+1})$.
5. Compute $\nabla d_{\bar{\delta}}(y^k; t_{k+1})$ and $\nabla^2 d_{\bar{\delta}}(y^k; t_{k+1})$ as in (4.3).
6. Update y^{k+1} as $y^{k+1} := y^k - \nabla^2 d_{\bar{\delta}}(y^k; t_{k+1})^{-1} \nabla d_{\bar{\delta}}(y^k; t_{k+1})$.

End For.
END.

The core steps of Phase 2 in Algorithm 1 are steps 4 and 6, where we need to solve M convex primal subproblems *in parallel* and to compute the IPFNT direction, respectively. Note that step 6 requires one to solve a system of linear equations. In addition, the quantity $\nabla^2 F(\bar{x}_{\bar{\delta}}(y^k, t_{k+1}))$ can also be computed *in parallel*.

The parameter t at step 3 can be updated adaptively as $t_{k+1} := (1-\sigma_k)t_k$, where $\sigma_k := \frac{\bar{\Delta}^*}{R_{\bar{\delta}}+(R_{\bar{\delta}}+1)\bar{\Delta}^*}$ and $R_{\bar{\delta}} := (1-\bar{\delta})^{-1}\big[\bar{\delta}(1-\bar{\delta})^{-1} + \|\nabla F(\bar{x}_{\bar{\delta}})\|^*_{\bar{x}_{\bar{\delta}}}\big]$. The stopping criterion at step 1 can be replaced by $\omega_*(\vartheta_k)t_k \leq \varepsilon_d$, where $\vartheta_k := (1-\bar{\delta})^{-1}[\lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t_k)}(y^k) + \bar{\delta}]$ due to Lemma 3.6 and (4.24).

Let us define $\lambda_{k+1} := \lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t_{k+1})}(y^{k+1})$ and $\lambda_k := \lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t_k)}(y^k)$. Then the local convergence of Algorithm 1 is stated in the following theorem.

THEOREM 4.6. *Let* $\{(y^k; t_k)\}$ *be a sequence generated by Algorithm* 1. *Then the number of iterations to obtain an* ε_d*-solution of* (3.4) *does not exceed*

$$k_{\max} := \left\lfloor \ln\left(\frac{t_0\omega_*(\vartheta)}{\varepsilon_d}\right)\left[\ln\left(1+\frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}\right)\right]^{-1}\right\rfloor + 1, \tag{4.25}$$

where $\vartheta := (1-\bar{\delta})\beta - \bar{\delta} \in (0,1)$ *and* $\bar{\Delta}^*$ *is defined by* (4.19).

Proof. Note that y^k is an ε_d-solution of (3.4) if $t_k \leq \frac{\varepsilon_d}{\omega_*(\vartheta)}$ due to Lemma 3.6, where $\vartheta = (1-\bar{\delta})\beta - \bar{\delta}$. Since $t_k = (1-\sigma)^k t_0$ due to step 3, we require $(1-\sigma)^k \leq \frac{\varepsilon_d}{t_0\omega_*(\vartheta)}$. Moreover, since $(1-\sigma)^{-1} = 1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$, the two last expressions imply (4.25). ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Remark 2 (the worst-case complexity). Since $\ln(1+\frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}) \approx \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$, it follows from Theorem 4.6 that the complexity of Algorithm 1 is $O(\sqrt{\nu}\ln\frac{t_0}{\varepsilon_d})$.

Remark 3 (linear convergence). The sequence $\{t_k\}$ linearly converges to zero with a contraction factor not greater than $1-\sigma$. When $\lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t)}(y) \le \beta$, it follows from (3.11) that $\lambda_{d_{\bar{\delta}}(\cdot;t)}(y) \le \beta\sqrt{t}$. Thus the sequence of Newton decrements $\{\lambda_{d(\cdot;t_k)}(y^k)\}_k$ of d also converges linearly to zero with a contraction factor at most $\sqrt{1-\sigma}$.

Remark 4 (the inexactness of the IPFNT direction). In implementations we can also apply an inexact method to solve the linear system for computing an IPFNT direction in (4.11). For more details of this method, one can refer to [23].

Finally, as a consequence of Theorem 4.6, the following corollary shows how to recover the optimality and feasibility of the original primal-dual problems (SCPP) and (2.1).

COROLLARY 4.7. *Suppose that* $(y^k;t_k)$ *is the output of Algorithm* 1 *and* $x^*(y^k;t_k)$ *is the solution of the primal subproblem* (3.2). *Then* $(x^*(y^k;t_k),y^k)$ *is an* ε_p*-solution of* (SCPP) *and* (2.1), *where* $\varepsilon_p := \nu\omega_*(\beta)^{-1}\varepsilon_d$.

4.5. Phase 1: Finding a starting point. Phase 1 of the algorithmic framework aims to find $y^0 \in Y$ such that $\lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t)}(y^0) \le \beta$. In this subsection, we apply an inexact perturbed damped Newton (IPDNT) method for finding such a point y^0.

4.5.1. IPDNT iteration. For a given $t = t_0 > 0$ and an accuracy $\bar{\delta} \ge 0$, let us assume that the current point $y \in Y$ is given, and we compute the new point y_+ by applying the IPDNT iteration as follows:

$$y_+ := y - \alpha(y)\nabla^2 d_{\bar{\delta}}(y;t_0)^{-1}\nabla d_{\bar{\delta}}(y;t_0), \tag{4.26}$$

where $\alpha := \alpha(y) > 0$ is the step size which will be defined appropriately. Note that since (4.26) is invariant under linear transformations, we can write

$$y_+ := y - \alpha(y)\nabla^2 \tilde{d}_{\bar{\delta}}(y;t_0)^{-1}\nabla \tilde{d}_{\bar{\delta}}(y;t_0). \tag{4.27}$$

It follows from (3.11) that $\tilde{d}(\cdot;t_0)$ is standard self-concordant, and by [15, Theorem 4.1.8], we have

$$\tilde{d}(y_+,t_0) \le \tilde{d}(y;t_0) + \nabla\tilde{d}(y;t_0)^T(y_+-y) + \omega_*(\|y_+-y\|_y), \tag{4.28}$$

provided that $\|y_+-y\|_y < 1$. On the other hand, (4.7) implies

$$0 \le \omega(\delta(\bar{x}_{\bar{\delta}},x^*)) \le \tilde{d}(y;t_0) - \tilde{d}_{\bar{\delta}}(y;t_0) \le \omega_*(\delta(\bar{x}_{\bar{\delta}},x^*)), \tag{4.29}$$

which gives bounds for the difference between $\tilde{d}(\cdot;t_0)$ and $\tilde{d}_{\bar{\delta}}(\cdot;t_0)$. In order to analyze the convergence of the IPDNT iteration (4.26) we denote

$$\begin{aligned} \hat{\delta}_+ &:= \|\bar{x}_{\bar{\delta}}(y_+,t_0) - x^*(y_+,t_0)\|_{x^*(y_+,t_0)},\\ \hat{\delta} &:= \|\bar{x}_{\bar{\delta}}(y;t_0) - x^*(y;t_0)\|_{x^*(y;t_0)},\\ \bar{\lambda}_0 &:= \lambda_{\tilde{d}_{\bar{\delta}}(\cdot;t_0)}(y) = \alpha(y)\|\!|y_+-y|\!\|_y, \end{aligned} \tag{4.30}$$

the solution differences of $d(\cdot;t_0)$ and $d_{\bar{\delta}}(\cdot;t_0)$ and the Newton decrement of $\tilde{d}_{\bar{\delta}}(\cdot;t_0)$, respectively. The next subsection shows how to update the step size $\alpha(y)$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

4.5.2. Finding the step size. The following lemma provides a formula to update the step size $\alpha(y)$ in (4.26).

LEMMA 4.8. *Let* $0 < \bar{\hat{\delta}} < \hat{\delta}^* := \beta(2+\beta+2\sqrt{\beta+1})^{-1}$, *and let* $\underline{\eta}$ *be defined as*

$$\underline{\eta} := \beta\Big[(1+\bar{\hat{\delta}})\beta + \sqrt{(1-\bar{\hat{\delta}})^2\beta^2 - 4\bar{\hat{\delta}}\beta}\Big]^{-1}\Big[(1-\bar{\hat{\delta}})\beta - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\beta^2 - 4\bar{\hat{\delta}}\beta}\Big]. \tag{4.31}$$

Then $\underline{\eta} \in (0,1)$. *Furthermore, if we choose the step size* $\alpha(y)$ *as*

$$\alpha(y) := \big[2\bar{\lambda}_0(1+\bar{\lambda}_0)\big]^{-1}\Big[(1-\bar{\hat{\delta}})\bar{\lambda}_0 - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\bar{\lambda}_0^2 - 4\bar{\hat{\delta}}\bar{\lambda}_0}\Big], \tag{4.32}$$

then $\alpha(y) \in (0,1)$ *and*

$$\tilde{d}_{\bar{\delta}}(y_+, t_0) \le \tilde{d}_{\bar{\delta}}(y; t_0) - \omega(\underline{\eta}). \tag{4.33}$$

As a consequence, if $\bar{\hat{\delta}} = 0$, *then* $\underline{\eta} = \beta$ *and* $\alpha(y) = (1+\bar{\lambda}_0)^{-1}$.

The asymptotic behavior of the functions $\eta(\cdot)$ and $\alpha(\cdot)$ w.r.t. $\bar{\hat{\delta}}$ is plotted in Figure 4.2. We can observe that α depends almost linearly on $\bar{\hat{\delta}}$.

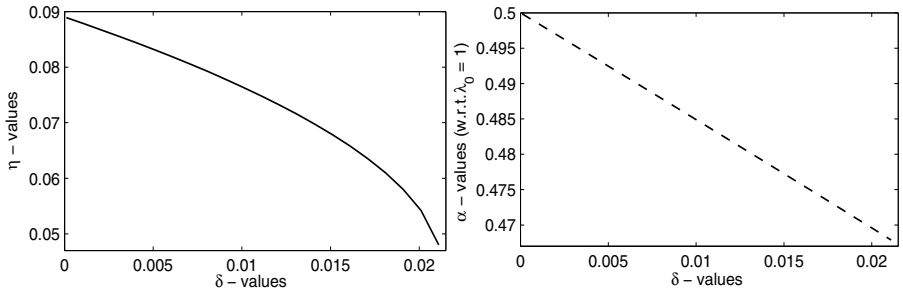

FIG. 4.2. *The asymptotic behavior of* η *and* α *w.r.t.* $\bar{\hat{\delta}}$ *at* $\lambda_0 = 1$ *and* $\beta = 0.089009$.

Proof. Let $p := y_+ - y$. From (4.28) and (4.29), we have

$$\begin{aligned}
\tilde{d}_{\bar{\delta}}(y_+, t_0) &\overset{(4.29)}{\le} \tilde{d}(y_+, t_0) \overset{(4.28)}{\le} \tilde{d}(y; t_0) + \nabla\tilde{d}(y; t_0)^T(y_+ - y) + \omega_*(\|y_+ - y\|_y) \\
&\overset{(4.29)}{\le} \tilde{d}_{\bar{\delta}}(y; t_0) + \nabla\tilde{d}(y; t_0)^T(y_+ - y) + \omega_*(\|y_+ - y\|_y) + \omega_*(\hat{\delta}) \\
&= \tilde{d}_{\bar{\delta}}(y; t_0) + \nabla\tilde{d}_{\bar{\delta}}(y; t_0)^T p + \big[\nabla\tilde{d}(y; t_0) - \nabla\tilde{d}_{\bar{\delta}}(y, t_0)\big]^T p + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}) \\
&\overset{(4.26)}{\le} \tilde{d}_{\bar{\delta}}(y; t_0) - \alpha\bar{\lambda}_0^2 + \|\nabla\tilde{d}(y; t_0) - \nabla\tilde{d}_{\bar{\delta}}(y; t_0)\|_y^*\|p\|_y + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}) \\
&\overset{(A.9)}{\le} \tilde{d}_{\bar{\delta}}(y; t_0) - \alpha\bar{\lambda}_0^2 + \hat{\delta}\|p\|_y + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}).
\end{aligned} \tag{4.34}$$

Furthermore, from (A.11) and the definition of $\nabla^2\tilde{d}$ and $\nabla^2\tilde{d}_{\bar{\delta}}$, we have

$$(1-\hat{\delta})\nabla^2\tilde{d}_{\bar{\delta}}(y; t_0) \preceq \nabla^2\tilde{d}(y; t_0) \preceq (1-\hat{\delta})^{-2}\nabla^2\tilde{d}_{\bar{\delta}}(y; t_0).$$

These inequalities imply $(1-\hat{\delta})\|\!|p|\!\|_y \le \|p\|_y \le (1-\hat{\delta})^{-1}\|\!|p|\!\|_y$. Combining the previous inequalities, (4.27), and the definition of $\bar{\lambda}_0$ in (4.30) we get

$$\alpha(1-\hat{\delta})\bar{\lambda}_0 \le \|p\|_y \le \alpha(1-\hat{\delta})^{-1}\bar{\lambda}_0.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Let us assume that $\alpha\bar{\lambda}_0 + \hat{\delta} < 1$. By substituting the second inequality into (4.34) and observing that the right-hand side of (4.34) is nondecreasing w.r.t. $\|p\|_y$, we get

$$\tilde{d}_{\bar{\delta}}(y_+, t_0) \le \tilde{d}_{\bar{\delta}}(y; t_0) - \alpha\bar{\lambda}_0^2 + (1-\hat{\delta})^{-1}\alpha\bar{\lambda}\hat{\delta} + \omega_*\left((1-\hat{\delta})^{-1}\alpha\bar{\lambda}_0\right) + \omega_*(\hat{\delta}). \tag{4.35}$$

Now, let us simplify the last four terms of (4.35) as follows:

$$\begin{aligned} &- \alpha\bar{\lambda}_0^2 + (1-\hat{\delta})^{-1}\alpha\bar{\lambda}_0\hat{\delta} + \omega_*\left((1-\hat{\delta})^{-1}\alpha\bar{\lambda}_0\right) + \omega_*(\hat{\delta}) \\ &= -\alpha\bar{\lambda}_0^2 - (\alpha\bar{\lambda}_0 + \hat{\delta}) - \ln\left[1 - (\alpha\bar{\lambda}_0 + \hat{\delta})\right] \\ &= -\alpha\bar{\lambda}_0^2 + \omega_*(\alpha\bar{\lambda}_0 + \hat{\delta}). \end{aligned} \tag{4.36}$$

Suppose that we can choose $\eta > 0$ such that $\alpha\bar{\lambda}_0^2 - \omega_*(\alpha\bar{\lambda}_0 + \hat{\delta}) = \omega(\eta)$. This condition leads to $\alpha\bar{\lambda}_0^2 = (\alpha\bar{\lambda}_0 + \hat{\delta})[\alpha(\bar{\lambda}_0 + \bar{\lambda}_0) + \hat{\delta}]$, which implies

$$\alpha = \left[2\bar{\lambda}_0(1+\bar{\lambda}_0)\right]^{-1}\left[(1-\hat{\delta})\bar{\lambda}_0 - 2\hat{\delta} + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}\right], \tag{4.37}$$

provided that $0 \le \hat{\delta} < \bar{\hat{\delta}} := \frac{2+\bar{\lambda}_0 - 2\sqrt{1+\bar{\lambda}_0}}{\bar{\lambda}_0}$. Consequently, we deduce

$$\eta = \bar{\lambda}_0\left[(1+\hat{\delta})\bar{\lambda}_0 + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}\right]^{-1}\left[(1-\hat{\delta})\bar{\lambda}_0 - 2\hat{\delta} + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}\right].$$

We assume that $\bar{\lambda}_0 \ge \beta$ for a given $\beta \in (0,1)$. Let us fix $\bar{\hat{\delta}}$ such that

$$0 < \bar{\hat{\delta}} < \hat{\delta}^* := \beta^{-1}\left[2 + \beta - 2\sqrt{1+\beta}\right] = \left[2 + \beta + 2\sqrt{1+\beta}\right]^{-1}\beta.$$

If we choose the step size $\alpha(y)$ as in (4.32) for the IPDNT iteration (4.26), then we obtain (4.33) with $\underline{\eta}$ defined by (4.31). □

Finally, we estimate the constant $\underline{\eta}$ for the case $\beta \approx 0.089009$. We first obtain $\hat{\delta}^* \approx 0.021314$. Let $\bar{\hat{\delta}} = \frac{1}{2}\hat{\delta}^* \approx 0.010657$. Then we get $\underline{\eta} \approx 0.075496$ and $\omega(\underline{\eta}) \approx 0.003002$.

4.5.3. The algorithm and its worst-case complexity. In summary, the algorithm for finding $y^0 \in Y$ is presented in detail as follows.

ALGORITHM **2**. (*Finding a starting point* $y^0 \in Y$).

Initialization: Perform the following steps:

1. Select $\beta \in (\beta_*, \beta^*)$ and $t_0 > 0$ as desired (e.g., $\beta = \frac{1}{4}\beta^* \approx 0.089009$).
2. Take an arbitrary point $y^{0,0} \in Y$.
3. Compute $\hat{\delta}^* := \beta[2 + \beta + 2\sqrt{1+\beta}]^{-1}$, and fix $\bar{\hat{\delta}} \in (0, \hat{\delta}^*)$ (e.g., $\bar{\hat{\delta}} = 0.5\hat{\delta}^*$).
4. Compute an accuracy $\varepsilon_0 := \frac{t_0\bar{\hat{\delta}}}{(\nu+2\sqrt{\nu})(1+\hat{\delta})}$.

Iteration: For $j = 0, 1, \ldots, j_{\max}$, perform the following steps:

1. Solve (3.2) approximately *in parallel* up to the accuracy ε_0 to obtain $\bar{x}_{\bar{\delta}}(y^{0,j}, t_0)$.
2. Compute $\bar{\lambda}_j := \bar{\lambda}_{\tilde{d}_{\bar{\delta}}(\cdot;t_0)}(y^{0,j})$.
3. If $\bar{\lambda}_j \le \beta$, then set $y^0 := y^{0,j}$ and terminate.
4. Update $y^{0,j+1}$ as $y^{0,j+1} := y^{0,j} - \alpha_j\nabla^2 d_{\bar{\delta}}(y^{0,j}, t_0)^{-1}\nabla d_{\bar{\delta}}(y^{0,j}, t_0)$, where $\alpha_j := \left[2\bar{\lambda}_j(1+\bar{\lambda}_j)\right]^{-1}[(1-\bar{\hat{\delta}})\bar{\lambda}_j - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\bar{\lambda}_j^2 - 4\bar{\hat{\delta}}\bar{\lambda}_j}] \in (0,1)$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

End For.
End.

The convergence of this algorithm is stated in the following theorem.

Theorem 4.9. *The number of iterations required in Algorithm* 2 *does not exceed*

$$j_{\max} := \Big\lfloor [t_0\omega(\underline{\eta})]^{-1}\big[d_{\bar\delta}(y^{0,0},t_0) - d^*(t_0) + \omega_*(\bar{\bar\delta})\big]\Big\rfloor + 1, \tag{4.38}$$

where $d^*(t_0) = \min_{y\in Y} d(y;t_0)$ *and* $\underline{\eta}$ *is given by* (4.31).

Proof. Summing up (4.33) from $j = 0$ to $j = l$ and then using (4.29) we have $0 \le \tilde d(y^{0,l},t_0) - \tilde d^*(t_0) \le \tilde d_{\bar\delta}(y^{0,l},t_0) - \tilde d^*(t_0) + \omega_*(\bar{\bar\delta}) \le \tilde d_{\bar\delta}(y^{0,0},t_0) - \tilde d^*(t_0) + \omega_*(\bar{\bar\delta}) - l\omega(\underline{\eta})$. This inequality together with (3.11) and (4.4) implies

$$j \le [t_0\omega(\underline{\eta})]^{-1}\big[d_{\bar\delta}(y^{0,0},t_0) - d^*(t_0) + \omega_*(\bar{\bar\delta})\big].$$

Hence, the maximum iteration number in Algorithm 2 does not exceed $j_{\max}$ defined by (4.38). ∎

Since $d^*(t_0)$ is unknown, the constant $j_{\max}$ in (4.38) gives only an upper bound for Algorithm 2. However, in Algorithm 2, we do not use $j_{\max}$ as a stopping criterion.

5. Path-following decomposition algorithm with exact Newton iterations. If we set $\bar\delta = 0$, then Algorithm 1 reduces to those considered in [10, 14, 21, 27, 28] as a special case. Note that, in [10, 14, 21, 27, 28], the primal subproblem (3.2) is assumed to be solved exactly so that the family $\{d(\cdot;t)\}_{t>0}$ of the smoothed dual functions is strongly self-concordant due to the Legendre transformation. Consequently, the standard theory of interior point methods in [17] can be applied to minimize such a function. In contrast to those methods, in this section we analyze directly the path-following iterations to select appropriate parameters for implementation. Moreover, the radius of the neighborhood of the central path in Algorithm 3 below is $(3-\sqrt{5})/2 \approx 0.381966$ compared to that in the literature, $2-\sqrt{3} \approx 0.267949$.

5.1. The exact path-following iteration. Let us assume that the primal subproblem (3.2) is solved exactly; i.e., $\bar\delta = 0$ in Definition 4.1. Then we have $\bar x_{\bar\delta} \equiv x^*$ and $\delta(\bar x_{\bar\delta}, x^*) = 0$ for all $y \in Y$ and $t > 0$. Moreover, it follows from (4.20) that $\Delta = \Delta^* = \|x^*(y;t_+) - x^*(y;t)\|_{x^*(y;t)}$. We consider one step of the path-following scheme with exact full-step Newton iterations:

$$\begin{cases} t_+ := t - \Delta_t,\ \Delta_t > 0, \\ y_+ := y - \nabla^2 d(y;t_+)^{-1}\nabla d(y;t_+) \equiv y - \nabla^2\tilde d(y;t_+)^{-1}\nabla\tilde d(y;t_+). \end{cases} \tag{5.1}$$

For the sake of notational simplicity, we denote $\tilde\lambda := \lambda_{\tilde d(\cdot;t)}(y)$, $\tilde\lambda_1 := \lambda_{\tilde d(\cdot,t_+)}(y)$, and $\tilde\lambda_+ := \lambda_{\tilde d(\cdot;t_+)}(y_+)$. It follows from (4.16) of Lemma 4.3 that

$$\tilde\lambda_+ \le \big(1 - 2\Delta^* - \tilde\lambda\big)^{-2}\big(\tilde\lambda + \Delta^*\big)^2. \tag{5.2}$$

Now, we fix $\beta \in (0,1)$ and assume that $\tilde\lambda \le \beta$. We need to find a condition on Δ^* such that $\tilde\lambda_+ \le \beta$. Indeed, since the right-hand side of (5.2) is nondecreasing w.r.t. $\tilde\lambda$, it implies that $\tilde\lambda_+ \le (1-2\Delta^*-\beta)^{-2}(\Delta^*+\beta)^2$. Thus if $\frac{\Delta^*+\beta}{1-2\Delta^*-\beta} \le \sqrt{\beta}$, then $\tilde\lambda_+ \le \beta$. The last condition leads to

$$0 \le \Delta^* \le \bar\Delta^* := (1+2\sqrt{\beta})^{-1}\sqrt{\beta}(1-\sqrt{\beta}-\beta), \tag{5.3}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

provided that

$$0 < \beta < \beta^* := (3 - \sqrt{5})/2 \approx 0.381966. \tag{5.4}$$

In particular, if we choose $\beta = \frac{\beta^*}{4} \approx 0.095492$, then $\bar{\Delta}^* \approx 0.113729$. Since $\bar{\Delta} \equiv \bar{\Delta}^*$, according to (4.21) and (5.1), we can update t as

$$t_+ := (1 - \sigma)t, \quad \text{where } \sigma := \left[\sqrt{\nu} + (\sqrt{\nu} + 1)\bar{\Delta}^*\right]^{-1}\bar{\Delta}^* \in (0, 1). \tag{5.5}$$

5.2. The algorithm and its convergence. The exact variant of Algorithms 1 and 2 is presented in detail as follows.

ALGORITHM **3**. (*Path-following algorithm with exact Newton iterations*).
Initialization: Given a tolerance $\varepsilon_d > 0$ and choose an initial value $t_0 > 0$. Fix a constant $\beta \in (0, \beta^*)$, where $\beta^* = \frac{3-\sqrt{5}}{2} \approx 0.381966$. Then compute $\bar{\Delta}^* := \frac{\sqrt{\beta}(1-\sqrt{\beta}-\beta)}{1+2\sqrt{\beta}}$ and $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}^*}$.
Phase 1. (*Finding a starting point*).
Choose an arbitrary starting point $y^{0,0} \in Y$.
For $j = 0, 1, \ldots, \tilde{j}_{\max}$, perform the following steps:

1. Solve the primal subproblem (3.2) *exactly in parallel* to obtain $x^*(y^{0,j}, t_0)$.
2. Evaluate $\nabla d(y^{0,j}, t_0)$ and $\nabla^2 d(y^{0,j}, t_0)$ as in (3.10). Then compute the Newton decrement $\tilde{\lambda}_j = \lambda_{\tilde{d}(\cdot;t_0)}(y^{0,j})$.
3. If $\tilde{\lambda}_j \le \beta$, then set $y^0 := y^{0,j}$ and terminate.
4. Update $y^{0,j+1}$ as $y^{0,j+1} := y^{0,j} - (1 + \tilde{\lambda}_j)^{-1}\nabla^2 d(y^{0,j}, t_0)^{-1}\nabla d(y^{0,j}, t_0)$.

End For.
Phase 2. (*Path-following iterations*).
For $k = 0, 1, \ldots, \tilde{k}_{\max}$ perform the following steps:

1. If $t_k \le \frac{\varepsilon_d}{\omega_*(\beta)}$, then terminate.
2. Update t_k as $t_{k+1} := (1 - \sigma)t_k$.
3. Solve the primal subproblem (3.2) *exactly in parallel* to obtain $x^*(y^k; t_{k+1})$.
4. Evaluate $\nabla d(y^k; t_{k+1})$ and $\nabla^2 d(y^k; t_{k+1})$ as in (3.10).
5. Update y^{k+1} as $y^{k+1} := y^k + \Delta y^k = y^k - \nabla^2 d(y^k; t_{k+1})^{-1}\nabla d(y^k; t_{k+1})$.

End For.
END.

Since $\tilde{d}(\cdot; t_0)$ is standard self-concordant due to Lemma 3.3, by [15, Theorem 4.1.12], the number of iterations required in Phase 1 does not exceed

$$\tilde{j}_{\max} := \left\lfloor \left[\tilde{d}(y^{0,0}; t_0) - \tilde{d}^*(t_0)\right]\omega(\beta)^{-1} \right\rfloor + 1 = \left\lfloor [d(y^{0,0}; t_0) - d^*(t_0)][t_0\omega(\beta)]^{-1} \right\rfloor + 1. \tag{5.6}$$

The convergence of Phase 2 in Algorithm 3 is stated in the following theorem.

THEOREM 5.1. *The maximum number of iterations needed in Phase* 2 *of Algorithm* 3 *to obtain an* ε_d*-solution of* (3.4) *does not exceed*

$$\tilde{k}_{\max} := \left\lfloor \ln\left(\frac{t_0\omega_*(\beta)}{\varepsilon_d}\right)\left[\ln\left(1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^* + 1)}\right)\right]^{-1} \right\rfloor + 1, \tag{5.7}$$

where $\bar{\Delta}^*$ *is defined by* (5.3).

Proof. From step 2 of Algorithm 3, we have $t_k = (1 - \sigma)^k t_0$. Hence, if $t_k \le \frac{\varepsilon_d}{\omega_*(\beta)}$, then $k \ge \ln(\frac{\omega_*(\beta)t_0}{\varepsilon_d})[\ln(1 - \sigma)^{-1}]^{-1}$. However, since $(1 - \sigma)^{-1} = 1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$, it

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

follows from the previous relation that $k \geq \ln(\frac{\omega_*(\beta)t_0}{\varepsilon_d})\ln(1+\frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)})^{-1}$, which leads to (5.7). ∎

Remark 5 (the worst-case complexity). Since $\ln(1+\frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}) \approx \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$, the worst-case complexity of Algorithm 3 is still $O\big(\sqrt{\nu}\ln\big(\frac{t_0}{\varepsilon_d}\big)\big)$.

Remark 6 (damped Newton iteration). Note that, at step 5 of Algorithm 3, we can use the damped Newton iteration $y^{k+1} := y^k - \alpha_k \nabla^2 d(y^k; t_{k+1})^{-1}\nabla d(y^k, t_{k+1})$ instead of the full-step Newton iteration, where $\alpha_k = (1+\lambda_{\tilde{d}(\cdot;t_{k+1})}(y^k))^{-1}$. In this case, with the same argument as before, we can compute $\beta^* = 0.5$ and $\Delta^* = \frac{\sqrt{0.5\beta}-\beta}{1+\sqrt{0.5\beta}}$.

6. Discussion on the implementation. In this section, we further discuss the implementation issues of the proposed algorithms.

6.1. Handling nonlinear objective function and local equality constraints. If the objective function ϕ_i in (SCPP) is nonlinear and concave and its epigraph is endowed with a self-concordant log-barrier for some $i \in \{1,\dots,M\}$, then we propose using a slack variable to move the objective function into the constraints and reformulate it as an optimization problem with linear objective function. More precisely, the reformulation becomes

$$\min_{x,s} \{s \mid Ax - b = 0,\ x \in X,\ \phi(x) \leq s\}. \tag{6.1}$$

By elimination of variables, it is not difficult to show that the optimality condition of the resulting problem collapses to the optimality condition of the original problem, i.e.,

$$\nabla\phi_i(x_i) + A_i^T y - t\nabla F_i(x_i) = 0.$$

The algorithms developed in the previous sections can be applied to solve such a problem without reformulating it as (6.1).

We also note that, in Algorithms 1 and 2, we need to solve the primal subproblems in (3.2) approximately up to a desired accuracy. Instead of solving these primal subproblems directly, we can treat them from the optimality condition (3.3). Since the objective function associated with this optimality condition is self-concordant, Newton-type methods can be applied to solve such a problem; see, e.g., [3, 15].

If, for some $i \in \{1,\dots,M\}$, local equality constraints $E_i x_i = f_i$ are considered in (SCPP), then the KKT condition of the primal subproblem i becomes

$$\begin{cases} c_i + A_i^T y + E_i^T z_i - t\nabla F_i(x_i) = 0, \\ E_i x_i - f_i = 0. \end{cases} \tag{6.2}$$

Instead of the full KKT system (6.2), we consider a reduced KKT condition as follows:

$$Z_i^T(c_i + A_i^T y) - tZ_i^T \nabla F_i(Z_i x_i^z + R_i^{-T} f_i) = 0.$$

Here, (Q_i, R_i) is a QR-factorization of E_i^T and $Q_i = [Y_i, Z_i]$ is a basis of the range space and the null space of E_i^T, respectively. Due to the invariance of the norm $\|\cdot\|_{x^*}$, we can show that $\|\bar{x}_{\bar{\delta}} - x^*\|_{x^*} = \|\bar{x}^z_{\bar{\delta}} - x^{z*}\|_{x^{z*}}$. Therefore, the condition (4.1) coincides with $\|\bar{x}^z_{\bar{\delta}} - x^{z*}\|_{x^{z*}} \leq \bar{\delta}$. However, the last condition is satisfied if

$$\|Z_i^T(c_i + A_i^T y) - tZ_i^T\nabla F_i(Z_i x_i^z + R_i^{-T} f_i)\|^*_{x_i^{zc}} \leq \varepsilon_i(t), \quad i = 1,\dots,M.$$

Note that the QR-factorization of E_i^T is computed only one time, a priori.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

6.2. Computing the inexact perturbed Newton direction. Regarding the Newton direction in Algorithms 1 and 2, one has to solve the linear system

$$\nabla^2 d_{\bar\delta}(y^k;t)\Delta y^k = -\nabla d_{\bar\delta}(y^k;t). \tag{6.3}$$

Here, the gradient vector $\nabla d_{\bar\delta}(y^k;t) = A\bar x_{\bar\delta}(y^k;t) - b = \sum_{i=1}^M (A_i\bar x_i(y^k;t) - b_i) := g_k$ and the Hessian matrix $\nabla^2 d_{\bar\delta}(y^k;t)$ is obtained from

$$\nabla^2 d_{\bar\delta}(y^k;t) = t^{-1}\sum_{i=1}^M A_i\nabla^2 F_i(\bar x_i(y^k;t))^{-1}A_i^T.$$

Note that each block $G_i^k := t^{-1}A_i\nabla^2 F_i(\bar x_i(y^k;t))^{-1}A_i^T$ can be computed *in parallel*. Then, the linear system (6.3) can be written as

$$\left(\sum_{i=1}^M A_i G_i^k A_i^T\right)\Delta y^k = -g_k. \tag{6.4}$$

Since matrix $G^k := \sum_{i=1}^M A_i G_i^k A_i^T \succ 0$, one can apply either Cholesky-type factorizations or conjugate gradient (CG)–type methods to solve (6.4). Note that the CG method requires only matrix-vector operations. More details on parallel solution of (6.4) can be found, e.g., in [14, 28].

7. Numerical tests. In this paper, we test the algorithms developed in the previous sections by solving a routing problem with congestion cost. This problem appears in many areas, including telecommunications, networks, and transportation [9].

Let $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ be a network of $n_{\mathcal{N}}$ nodes and $n_{\mathcal{A}}$ links, and let $\mathcal{C}$ be a set of $n_{\mathcal{C}}$ commodities to be sent through the network $\mathcal{G}$, where each commodity $k \in \mathcal{C}$ has a source $s_k \in \mathcal{N}$, a destination $d_k \in \mathcal{N}$, and a certain amount of demand r_k. The optimization model of the routing problem with congestion (RPC) can be formulated as follows; see, e.g., [9] for more details:

$$\begin{aligned}
\min_{u_{ijk},v_{ij}} \quad & \textstyle\sum_{k\in\mathcal{C}}\sum_{(i,j)\in\mathcal{A}} c_{ij}u_{ijk} + \sum_{(i,j)\in\mathcal{A}} w_{ij}g_{ij}(v_{ij}) \\
\text{s.t.} \quad & \textstyle\sum_{j:(i,j)\in\mathcal{A}} u_{ijk} - \sum_{j:(j,i)\in\mathcal{A}} u_{jik} = \begin{cases} r_k & \text{if } i = s_k, \\ -r_k & \text{if } i = d_k, \\ 0 & \text{otherwise,} \end{cases} \\
& \textstyle\sum_{k\in C} u_{ijk} - v_{ij} = b_{ij},\ (i,j) \in \mathcal{A}, \\
& u_{ijk} \ge 0,\ v_{ij} \ge 0,\ (i,j) \in \mathcal{A},
\end{aligned} \tag{7.1}$$

where $w_{ij} \ge 0$ is the weighting of the additional cost function g_{ij} for $(i,j) \in \mathcal{A}$.

In this example we assume that the additional cost function g_{ij} is given by either (a) $g_{ij}(v_{ij}) = -\ln(v_{ij})$, the logarithmic function, or (b) $g_{ij}(v_{ij}) = v_{ij}\ln(v_{ij})$, the entropy function. It was shown in [15] that the epigraph of g_{ij} possesses a standard self-concordant barrier (a) $F_{ij}(v_{ij}, s_{ij}) = -\ln v_{ij} - \ln(\ln v_{ij} + s_{ij})$ or (b) $F_{ij}(v_{ij}, s_{ij}) = -\ln v_{ij} - \ln(s_{ij} - v_{ij}\ln v_{ij})$, respectively.

By using slack variables s_{ij}, we can move the nonlinear terms of the objective function to the constraints. The objective function of the resulting problem becomes

$$f(u,v,s) := \sum_{k\in\mathcal{C}}\sum_{(i,j)\in\mathcal{A}} c_{ij}u_{ijk} + \sum_{(i,j)\in\mathcal{A}} w_{ij}s_{ij}, \tag{7.2}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

with additional constraints $g_{ij}(v_{ij}) \leq s_{ij}$, $(i,j) \in \mathcal{A}$. It is clear that problem (7.1) is separable and convex. Let

$$X_{ij} := \left\{ v_{ij} \geq 0, \sum_{k \in \mathcal{C}} u_{ijk} - v_{ij} = b_{ij}, g_{ij}(v_{ij}) \leq s_{ij}, (i,j) \in \mathcal{A}, k \in \mathcal{C} \right\}, \ (i,j) \in \mathcal{A}. \tag{7.3}$$

Then problem (7.1) can be reformulated in the form of (SCPP) with linear objective function (7.2) and local constraint set (7.3). Moreover, the resulting problem has $M := n_{\mathcal{A}}$ components; $n := n_{\mathcal{C}} n_{\mathcal{A}} + 2n_{\mathcal{A}}$ variables, including u_{ijk}, v_{ij}, and s_{ij}; and $m := n_{\mathcal{C}} n_{\mathcal{N}}$ coupling constraints. Each primal subproblem (3.2) has $n_i := n_{\mathcal{C}} + 2$ variables and one local linear equality constraint.

The aim is to compare the effect of the inexactness on the performance of the algorithms. We consider two variants of Algorithm 1, where we set $\bar{\delta} = 0.5\bar{\delta}^*$ and $\bar{\delta} = 0.25\bar{\delta}^*$ in Phase 1 and $\bar{\delta} = 0.01$ and $\bar{\delta} = 0.005$ in Phase 2, respectively. We denote these variants by `A1-v1` and `A1-v2`, respectively. For Algorithm 3, we also consider two cases. In the first case we set the tolerance of the primal subproblems to $\varepsilon_p = 10^{-6}$, and the second to 10^{-10}, where we denote them by `A3-v1` and `A3-v2`, respectively. All variants are terminated with the same tolerance $\varepsilon_d = 10^{-4}$. The initial penalty parameter value is set to $t_0 := 0.25$.

We benchmarked four variants with performance profiles [6]. Recall that a performance profile is built based on a set $\mathcal{S}$ of n_s algorithms (solvers) and a collection $\mathcal{P}$ of n_p problems. Suppose that we build a profile based on computational time. We denote $T_{p,s} :=$ *computational time required to solve problem* p *by solver* s. We compare the performance of algorithm s on problem p with the best performance of any algorithm on this problem; that is, we compute the performance ratio $r_{p,s} := \frac{T_{p,s}}{\min\{T_{p,\hat{s}} \mid \hat{s} \in \mathcal{S}\}}$. Now, let $\tilde{\rho}_s(\tilde{\tau}) := \frac{1}{n_p} \text{size}\{p \in \mathcal{P} \mid r_{p,s} \leq \tilde{\tau}\}$ for $\tilde{\tau} \in \mathbb{R}_+$. The function $\tilde{\rho}_s : \mathbb{R} \to [0,1]$ is the probability for solver s that a performance ratio is within a factor $\tilde{\tau}$ of the best possible ratio. We use the term "performance profile" for the distribution function $\tilde{\rho}_s$ of a performance metric. We can also plot the performance profiles in log-scale, i.e., $\rho_s(\tau) := \frac{1}{n_p} \text{size}\{p \in \mathcal{P} \mid \log_2(r_{p,s}) \leq \tau := \log_2 \tilde{\tau}\}$.

All the algorithms have been implemented in C++ running on an Intel Core TM2, Quad-Core Processor Q6600 (2.4GHz) PC Desktop with 3Gb RAM and have been paralellized by using `OpenMP`. The input data was generated randomly, where the nodes of the network were generated in a rectangle $[0,100] \times [0,300]$, the demand r_k was in $[50,500]$, the weighting vector w was set to 10, the congestion b_{ij} was in $[10,100]$, and the linear cost c_{ij} was the Euclidean length of the link $(i,j) \in \mathcal{A}$. The nonlinear cost function g_{ij} was chosen randomly between two functions in (a) and (b) defined above with the same probability.

We tested the algorithms on a collection of 108 random problems. The size of these problems varied from $M = 6$ to $14,280$ components, $n = 84$ to $77,142$ variables, and $m = 15$ to 500 coupling constraints. The performance profiles of the four algorithms in terms of computational time are shown in Figure 7.1, where the horizontal axis is the factor τ (not more than 2^τ-times worse than the best one) and the vertical axis is the probability function values $\rho_s(\tau)$ (problems ratio).

As we can see from Figure 7.1, Algorithm 1 performs better than Algorithm 3 both in the total computational time and the time for solving the primal subproblems. This provides evidence of the effect of the inexactness on the performance of the algorithm. We also observed that the numbers of iterations for solving the master problem in Phase 1 of all variants are similar, while they are different in Phase 2. However, since

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

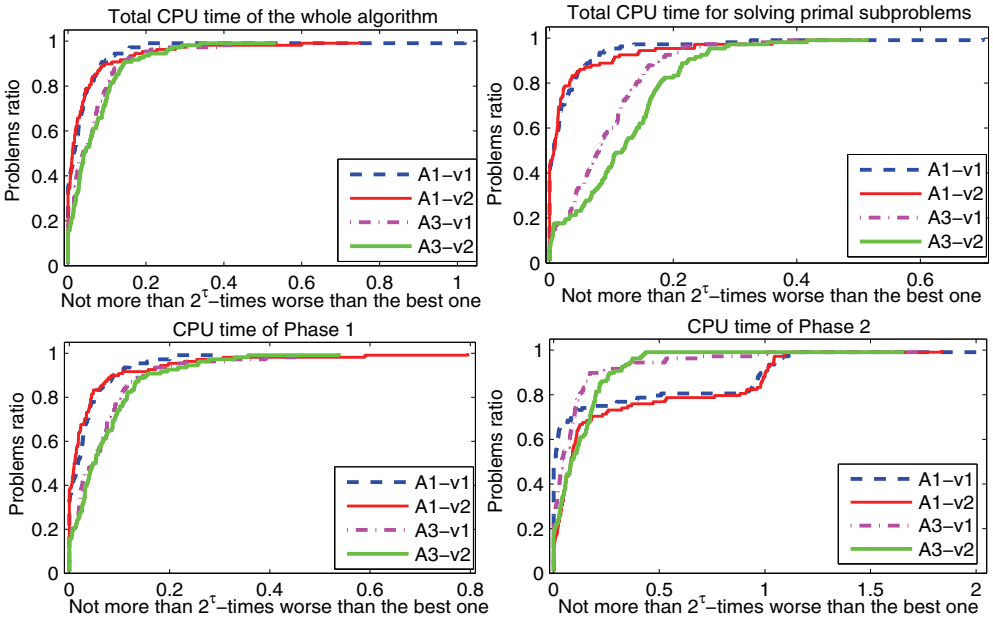

FIG. 7.1. *The performance profiles of the four variants in terms of computational time.*

Phase 2 is performed when the approximate point is in the quadratic convergence region, it requires a few iterations toward the desired approximate solution. Therefore, the computational time of Phase 1 dominates that of Phase 2. We notice that, in this particular example, the structure of the master problem is almost dense, and we did not use any sparse linear algebra solver.

We also compared the total number of iterations for solving the primal subproblems in Figure 7.2. It shows that Algorithm 1 is superior to Algorithm 3 in terms of number of iterations, although the accuracy of solving the primal subproblem in Algorithm 3 is set only to 10^{-6}, which is not exact as theoretically required. This performance profile also reveals the effect of the inexactness on the number of iterations. In our numerical results, the inexact version `A1-v1` saves 22% (resp., 23%) of the total number of iterations to solve the primal subproblems compared to `A3-v1` (resp., `A3-v2`), while the variant `A1-v2` saves 20% (resp., 21%) compared to `A3-v1` (resp., `A3-v2`).

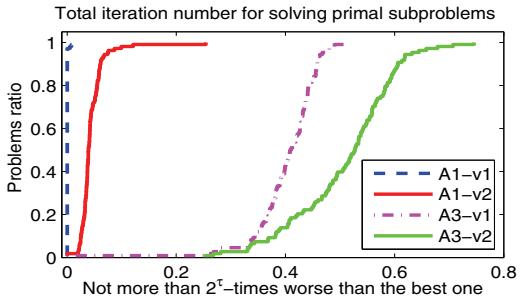

FIG. 7.2. *The performance profile of the four variants in terms of iteration number.*

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

8. Concluding remarks. We have proposed a smoothing technique for Lagrangian decomposition using self-concordant barriers in separable convex optimization. We have proved a global and a local approximation for the dual function. Then we have proposed a path-following algorithm with inexact perturbed Newton iterations. The convergence of the algorithm has been analyzed, and its worst-case complexity has been estimated. The theory presented in this paper is significant in practice since it allows us to solve the primal subproblems inexactly. Moreover, we can trade off between the accuracy of solving the primal subproblem and the convergence rate of the path-following phase. As a special case, we have again obtained the path-following methods studied by Mehrotra and Ozevin [12] and Shida [21] with some additional advantages. Preliminary numerical tests confirm the advantages of the inexact methods. Extensions to distributed implementation of linear algebra in the master problem are an interesting and significant future research direction.

Appendix A. The proof of the technical statements. In this appendix, we provide a complete proof of Lemmas 3.1, 3.2, and 4.3 and Theorem 4.4.

A.1. The proof of Lemma 3.1.

Proof. For notational simplicity, we denote $x_i^* := x_i^*(y;t)$. The left-hand side of (3.5) follows from $F_i(x_i) - F_i(x_i^c) \geq \omega(\|x_i - x_i^c\|_{x_i^c}) \geq 0$ due to (3.1). We prove the right-hand side of (3.5). Since F_i is standard self-concordant and $x_i^c = \operatorname{argmin}_{x_i \in \operatorname{int}(X_i)} F_i(x_i)$, according to [15, Theorem 4.1.13] we have

$$F_i(x_i^*) - F_i(x_i^c) \leq \omega_*(\lambda_{F_i}(x_i^*)), \tag{A.1}$$

provided that $\lambda_{F_i}(x_i^*) < 1$. Now, we prove (3.5). Let $x_i(\alpha) := x_i^* + \alpha(x_{0i}^*(y) - x_i^*)$ for $\alpha \in [0,1)$. Since $x_i^* \in \operatorname{int}(X_i)$ and $\alpha < 1$, $x_i(\alpha) \in \operatorname{int}(X_i)$. By applying [17, inequality 2.3.3], we have $F_i(x_i(\alpha)) \leq F_i(x_i^*) - \nu_i \ln(1-\alpha)$, which is equivalent to

$$F_i(x_i(\alpha)) - F_i(x_i^c) \leq F_i(x_i^*) - F_i(x_i^c) - \nu_i \ln(1-\alpha). \tag{A.2}$$

From the definition of $d_i(\cdot;t)$ and $d_{0i}(\cdot)$, the concavity of ϕ_i, and (A.1) we have

$$\begin{aligned} d_i(y;t) &\geq \max_{\alpha\in[0,1)} \left\{\phi_i(x_i(\alpha)) + y^T(A_i x_i(\alpha) - b_i) - t[F_i(x_i(\alpha)) - F_i(x_i^c)]\right\} \\ &\geq \max_{\alpha\in[0,1)} \Big\{\alpha[\phi_i(x_{0i}^*(y)) + y^T(A_i x_{0i}^*(y) - b_i)] + (1-\alpha)[\phi_i(x_i^*) + y^T(A_i x_i^* - b_i)] \\ &\qquad - t[F_i(x_i^*) - F_i(x_i^c)] + \nu_i t \ln(1-\alpha) \mid \alpha \in [0,1)\Big\} \qquad \text{(A.3)} \\ &\overset{\text{(A.1)}}{\geq} \max_{\alpha\in[0,1)} \Big\{\alpha d_{0i}(y) + (1-\alpha) d_i(y;t) + t\nu_i \ln(1-\alpha) - t\omega_*(\lambda_{F_i}(x_i^*))\Big\}. \end{aligned}$$

By solving the last maximization problem in (A.3) we obtain the solution $\alpha^* = 0$ if $d_{0i}(y) - d_i(y;t) \leq t\nu_i$ and $\alpha^* = 1 - [d_{0i}(y) - d_i(y;t)]^{-1}\nu_i t$ otherwise. Substituting this solution into (A.3) we get

$$d_{0i}(y) - d_i(y;t) \leq t\nu_i\Big\{1 + \ln\left[(d_{0i}(y) - d_i(y;t))/(t\nu_i)\right] + \omega_*(\lambda_{F_i}(x_i^*))/\nu_i\Big\}, \tag{A.4}$$

provided that $d_{0i}(y) - d_i(y;t) > t\nu_i$. By rearranging (A.4) we obtain $d_{0i}(y) - d_i(y;t) \leq t\nu_i\big(1 + \omega^{-1}(\omega_*(\lambda_{F_i}(x_i^*))/\nu_i)\big)$. Summing up the last inequalities from $i = 1$ to M we obtain the right-hand side of (3.5). ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

A.2. The proof of Lemma 3.2.

Proof. The first inequality in (3.7) was proved in Lemma 3.1. We now prove the second one. Let us denote $x_i^\tau(y) := x_i^c + \tau(x_{0i}^*(y) - x_i^c)$, where $\tau \in [0,1]$ and $d_i^c(y) := \phi_i(x_i^c) + y^T(A_i x_i^c - b_i)$. Since F_i is ν_i-self-concordant, it follows from [17, inequality 2.3.3] that

$$F_i(x_i^\tau(y)) \le F_i(x_i^c) - \nu_i \ln(1-\tau),\ \tau \in [0,1).$$

Combining this inequality and the concavity of ϕ_i and then using the definitions of d_i^c and $d_i(\cdot)$ we have

$$\begin{aligned} d_i(y;t) &\ge \max_{\tau\in[0,1)} \left\{\phi_i(x_i^\tau(y)) + y^T A_i(x_i^\tau(y) - b_i) - t[F_i(x_i^\tau(y)) - F_i(x_i^c)]\right\} \\ &\ge \max_{\alpha\in[0,1)} \Big\{(1-\tau)[\phi_i(x_i^c) + y^T(A_i x_i^c - b_i)] + \tau[\phi_i(x_{0i}^*(y)) + y^T(A_i x_{0i}^*(y) - b_i)] \\ &\qquad + t\nu_i \ln(1-\tau)\Big\} \\ &= \max_{\tau\in[0,1)} \left\{(1-\tau)d_i^c(y) + \tau d_{0i}(y) + t\nu_i \ln(1-\tau)\right\}. \end{aligned} \tag{A.5}$$

Now, we maximize the function $\xi(\tau) := (1-\tau)d_i^c(y) + \tau d_{0i}(y) + t\nu_i \ln(1-\tau)$ in the last line of (A.5) w.r.t. $\tau \in [0,1)$ to obtain $\tau^* = \left[1 - \frac{t\nu_i}{d_{0i}(y) - d_i^c(y)}\right]_+$, where $[a]_+ := \max\{0, a\}$. If $d_{0i}(y) - d_i^c(y) \le t\nu_i$, i.e., $\tau^* = 0$, then $d_{0i}(y) - d_i^c(y) \le t\nu_i$. Otherwise, by substituting τ^* into the last line of (A.5), we obtain

$$d_{0i}(y) \le d_i(y;t) + t\nu_i\Big(1 + \left[\ln((t\nu_i)^{-1}(d_{0i}(y) - d_i^c(y)))\right]_+\Big). \tag{A.6}$$

Furthermore, we note that $d_{0i}(y) - d_i^c(y) = \max_{x_i\in X_i}\left\{\phi_i(x_i) + y^T(A_i x_i - b_i)\right\} - \left[\phi_i(x_i^c) + y^T(A_i x_i^c - b_i)\right] \ge 0$ for all $y \in Y$ and

$$\begin{aligned} d_{0i}(y) - d_i^c(y) &\overset{\phi \text{ is concave}}{\le} \max_{x_i\in X_i}\left\{\max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\left[\xi_i + A_i^T y\right]^T(x_i - x_i^c)\right\}\right\} \\ &\le \max_{x_i\in X_i}\left\{\max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\|\xi_i + A_i^T y\|_{x_i^c}^* \|x_i - x_i^c\|_{x_i^c}\right\}\right\} \\ &\overset{(3.1)}{\le} (\nu_i + 2\sqrt{\nu_i}) \max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\|\xi_i + A_i^T y\|_{x_i^c}^*\right\} \\ &\le K_i < +\infty \quad \forall y \in Y. \end{aligned} \tag{A.7}$$

Summing up the inequalities (A.6) for $i = 1, \ldots, M$ and then using (A.7) we get (3.7).

Finally, for fixed $\kappa \in (0,1)$, since $\ln(x^{-1}) \le x^{-\kappa}$ for $0 < x \le \kappa^{1/\kappa}$, we have

$$\begin{aligned} \nu_i t\left(1 + \left[\ln\frac{K_i}{\nu_i t}\right]_+\right) &\le \nu_i t\left[1 + \left(\frac{K_i}{\nu_i t}\right)^\kappa\right] \\ &\le \left[\nu_i + K_i^\kappa \nu_i^{1-\kappa}\right] t^{1-\kappa} \quad \forall t \le \nu_i^{-1} K_i \kappa^{1/\kappa}. \end{aligned}$$

Consequently, if $t \le \min\{\frac{K_i}{\nu_i}\kappa^{1/\kappa}, (\frac{\varepsilon_d}{\sum_{i=1}^M[\nu_i + \nu_i^{1-\kappa}K_i^\kappa]})^{1/(1-\kappa)},\ i = 1, \ldots, M\}$, then $D_X(t) \le \varepsilon_d$, where $D_X(t)$ is defined as in Lemma 3.2. Combining this condition and (3.7) we get the last conclusion of Lemma 3.2. ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

A.3. The proof of Lemma 4.3. First, we prove the following lemma, which will be used to prove the main inequality in Lemma 4.3.

LEMMA A.1. *Suppose that Assumptions* A1, A2, *and* A3 *are satisfied. Then the following hold:*

(a) $\nabla^2\tilde{d}$ *and* $\nabla^2\tilde{d}_{\bar{\delta}}$ *defined by* (3.10) *and* (4.3), *respectively, guarantee*

$$(1-\delta_+)^2\nabla^2\tilde{d}(y_+,t_+) \preceq \nabla^2\tilde{d}_{\bar{\delta}}(y_+,t_+) \preceq (1-\delta_+)^{-2}\nabla^2\tilde{d}(y_+,t_+), \tag{A.8}$$

where $\delta_+ < 1$ *defined by* (4.6).

(b) *Moreover, one has*

$$\|\nabla\tilde{d}_{\bar{\delta}}(y;t) - \nabla\tilde{d}(y;t)\|_y^* \le \|\bar{x}_{\bar{\delta}} - x^*\|_{x^*}. \tag{A.9}$$

(c) *If* $\Delta < 1$, *then* $\bar{\lambda}_1 \le \frac{\Delta+\bar{\lambda}}{1-\Delta}$.

Proof. Since F is standard self-concordant, for any $x \in \mathrm{dom}(F)$ and z such that $\|z-x\|_x < 1$, it follows from [15, Theorem 4.1.6] that

$$(1-\|z-x\|_x)^2\nabla^2F(x) \preceq \nabla^2F(z) \preceq (1-\|z-x\|_x)^{-2}\nabla^2F(x). \tag{A.10}$$

Since $\nabla^2F(x)$ is symmetric positive definite, by applying [1, Proposition 8.6.6] to two matrices $(1-\|z-x\|_x)^{-2}\nabla^2F(x)$ and $\nabla^2F(z)$, and then to two matrices $(1-\|z-x\|_x)^2\nabla^2F(x)$ and $\nabla^2F(z)$, we obtain

$$(1-\|z-x\|_x)^2A\nabla^2F(x)^{-1}A^T \preceq A\nabla^2F(z)^{-1}A^T \preceq (1-\|z-x\|_x)^{-2}A\nabla^2F(x)^{-1}A^T. \tag{A.11}$$

Again using [1, Proposition 8.6.6] for (A.11) we get

$$\begin{aligned}&(1-\|z-x\|_x)^2A^T[A\nabla^2F(x)^{-1}A^T]^{-1}A \preceq A^T[A\nabla^2F(z)^{-1}A^T]^{-1}A\\ &\preceq (1-\|z-x\|_x)^{-2}A^T[A\nabla^2F(x)^{-1}A^T]^{-1}A.\end{aligned} \tag{A.12}$$

Now, using (3.10) and (3.11), we have $\nabla^2\tilde{d}(y;t) = t^{-2}A\nabla^2F(x^*)^{-1}A^T$. Alternatively, using (4.3) and (4.4), we get $\nabla^2\tilde{d}_{\bar{\delta}}(y;t) = t^{-2}A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T$. Substituting these relations with $x = x_+^*$ and $z = \bar{x}_{\bar{\delta}+}$ into (A.11) and noting that $\delta_+ = \delta(\bar{x}_+, x_+^*)$ defined by (4.6), we obtain (A.8).

Next, we prove (b). For any $x \in \mathrm{dom}(F)$, we have $\nabla^2F(x) \succ 0$. It is not difficult to show that the matrix $M(x) := \begin{bmatrix}\nabla^2F(x) & A^T\\ A & A\nabla^2F(x)^{-1}A^T\end{bmatrix}$ is symmetric positive semidefinite. Since A has full-row rank, $A\nabla^2F(x)^{-1}A^T \succ 0$. By applying Schur's complement to $M(x)$ [1], we obtain

$$A^T[A\nabla^2F(x)^{-1}A^T]^{-1}A \preceq \nabla^2F(x). \tag{A.13}$$

To prove (A.9) we note that $\nabla d_{\bar{\delta}}(y;t) - \nabla d(y;t) = A(\bar{x}_{\bar{\delta}} - x^*)$. Thus $\nabla\tilde{d}_{\bar{\delta}}(y;t) - \nabla\tilde{d}(y;t) = \frac{1}{t}A(\bar{x}_{\bar{\delta}} - x^*)$. This implies

$$\begin{aligned}\left[\|\nabla\tilde{d}_{\bar{\delta}}(y;t) - \nabla\tilde{d}(y;t)\|_y^*\right]^2 &= t^{-2}(\bar{x}_{\bar{\delta}} - x^*)^TA^T\nabla^2\tilde{d}(y;t)^{-1}A(\bar{x}_{\bar{\delta}} - x^*)\\ &\overset{(3.10),(3.11)}{=} (\bar{x}_{\bar{\delta}} - x^*)^TA^T[A\nabla^2F(x^*)^{-1}A^T]^{-1}A(\bar{x}_{\bar{\delta}} - x^*)\\ &\overset{(A.13)}{\le} (\bar{x}_{\bar{\delta}} - x^*)^T\nabla^2F(x^*)(\bar{x}_{\bar{\delta}} - x^*)\\ &= \|\bar{x}_{\bar{\delta}} - x^*\|_{x^*}^2,\end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

which is equivalent to (A.9).

Finally, we prove (c). By using the definitions of $\nabla\tilde{d}_{\bar{\delta}}(\cdot;t_+)$ and $\nabla^2\tilde{d}_{\bar{\delta}}(\cdot;t_+)$ in (4.3), and of $\tilde{d}_{\bar{\delta}}(\cdot;t_+)$ in (4.4), for any feasible point $\hat{x}$ to (SCPP), it follows from the definition of $\bar{\lambda}_1$ in (4.5) and $A\hat{x}=b$ that

$$\begin{aligned}\bar{\lambda}_1^2 &= \left[\|\nabla\tilde{d}_{\bar{\delta}}(y;t_+)\|_y^*\right]^2 \overset{(4.5)}{=} \nabla\tilde{d}_{\bar{\delta}}(y;t_+)\nabla^2\tilde{d}_{\bar{\delta}}(y;t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y;t_+)\\ &\overset{(4.4)}{=} t_+^{-1}\nabla d_{\bar{\delta}}(y;t_+)\nabla^2 d_{\bar{\delta}}(y;t_+)^{-1}\nabla d_{\bar{\delta}}(y;t_+)\\ &\overset{(4.3)}{=} (\bar{x}_{\bar{\delta}1}-\hat{x})^T A^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}1})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}1}-\hat{x}).\end{aligned} \tag{A.14}$$

Since $\Delta = \|\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}\|_{\bar{x}_{\bar{\delta}}} < 1$ by assumption, we can apply the right-hand side of (A.12) with $x=\bar{x}_{\bar{\delta}}$ and $z=\bar{x}_{\bar{\delta}1}$ to obtain

$$\bar{\lambda}_1^2 \le (1-\Delta)^{-2}(\bar{x}_{\bar{\delta}1}-\hat{x})^T A^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}1}-\hat{x}). \tag{A.15}$$

Now, for any symmetric positive semidefinite matrix Q in $\mathbb{R}^{n\times n}$ and $u,v\in\mathbb{R}^n$, one can easily show that

$$(u+v)^TQ(u+v) \le \left[(u^TQu)^{1/2}+(v^TQv)^{1/2}\right]^2. \tag{A.16}$$

Since $H_{\bar{\delta}} := A^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A \succeq 0$, by applying (A.16) with $Q=H_{\bar{\delta}}$, $u=\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}$, and $v=\bar{x}_{\bar{\delta}}-\hat{x}$, we have

$$\bar{\lambda}_1^2 \le (1-\Delta)^{-2}\left\{\left[(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}})\right]_{[1]}^{1/2}+\left[(\bar{x}_{\bar{\delta}}-\hat{x})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}}-\hat{x})\right]_{[2]}^{1/2}\right\}^2. \tag{A.17}$$

Note that $H_{\bar{\delta}} \preceq \nabla^2F(\bar{x}_{\bar{\delta}})$ due to (A.13). The first term $[\cdot]_{[1]}$ in (A.17) satisfies

$$[\cdot]_{[1]} \le (\bar{x}_{\bar{\delta}+}-\bar{x}_{\bar{\delta}})^T\nabla^2F(\bar{x}_{\bar{\delta}})(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}) = \Delta^2. \tag{A.18}$$

On the other hand, by substituting $\bar{x}_{\bar{\delta}1}$ by $\bar{x}_{\bar{\delta}}$ into (A.14), we get

$$\bar{\lambda}^2 = (\bar{x}_{\bar{\delta}}-\hat{x})^TA^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}}-\hat{x}) = (\bar{x}_{\bar{\delta}}-\hat{x})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}}-\hat{x}). \tag{A.19}$$

Combining (A.17), (A.18), and (A.19), we obtain $\bar{\lambda}_1 \le \frac{(\Delta+\bar{\lambda})}{(1-\Delta)}$, which is indeed (c). ∎

The proof of Lemma 4.3. Since $\delta_1+2\Delta+\bar{\lambda}<1$, it implies that $\delta_1<1$, $\Delta<1/2$, and $\bar{\lambda}<1$. The proof of Lemma 4.3 is divided into several steps as follows.

Step 1. First, letting $p := y_+ - y$, we prove the following inequality:

$$\bar{\lambda}_+ \le (1-\delta_+)^{-1}\left\{\delta_+ + (1-\|p\|_y)^{-1}\left[\delta_1+\frac{(2\delta_1-\delta_1^2)}{(1-\delta_1)^2}\|p\|_y+\frac{\|p\|_y^2}{1-\|p\|_y}\right]\right\}. \tag{A.20}$$

Indeed, it follows from (A.8) that

$$\begin{aligned}\bar{\lambda}_+ &= \|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|_{y_+}^* = \left[\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\nabla^2\tilde{d}_{\bar{\delta}}(y_+,t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\right]^{1/2}\\ &\overset{(A.8)}{\le} (1-\delta_+)^{-1}\left[\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\nabla^2\tilde{d}(y_+,t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\right]^{1/2}\\ &\le (1-\delta_+)^{-1}\|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|_{y_+}^*.\end{aligned} \tag{A.21}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Furthermore, by using (A.9) we have

$$\begin{aligned}\|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|^*_{y_+} &\le \|\nabla\tilde{d}(y_+,t_+)\|^*_{y_+} + \|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+) - \nabla\tilde{d}(y_+,t_+)\|^*_{y_+}\\ &\overset{(A.9)}{\le} \|\nabla\tilde{d}(y_+,t_+)\|^*_{y_+} + \delta_+.\end{aligned} \tag{A.22}$$

Since $\tilde{d}(\cdot;t_+)$ is standard self-concordant due to Lemma 3.3, one has

$$\|\nabla\tilde{d}(y_+,t_+)\|^*_{y_+} \le (1-\|y_+-y\|_y)^{-1}\|\nabla\tilde{d}(y_+,t_+)\|^*_y = (1-\|p\|_y)^{-1}\|\nabla\tilde{d}(y_+,t_+)\|^*_y. \tag{A.23}$$

Plugging (A.23) and (A.22) into (A.21) we obtain

$$\bar{\lambda}_+ \le (1-\delta_+)^{-1}\left[(1-\|p\|_y)^{-1}\|\nabla\tilde{d}(y_+,t_+)\|^*_y + \delta_+\right]. \tag{A.24}$$

On the other hand, from (4.12), we have

$$\begin{aligned}\nabla\tilde{d}(y_+,t_+) &\overset{(4.12)}{=} \nabla\tilde{d}(y_+,t_+) - \left[\nabla\tilde{d}_{\bar{\delta}}(y,t_+) + \nabla^2\tilde{d}_{\bar{\delta}}(y;t_+)(y_+-y)\right]\\ &= \left[\nabla\tilde{d}(y;t_+) - \nabla\tilde{d}_{\bar{\delta}}(y;t_+)\right]_{[1]} + \left\{[\nabla^2\tilde{d}(y;t_+) - \nabla^2\tilde{d}_{\bar{\delta}}(y;t_+)](y_+-y)\right\}_{[2]}\\ &\quad + \left[\nabla\tilde{d}(y_+,t_+) - \nabla\tilde{d}(y;t_+) - \nabla^2\tilde{d}(y;t_+)(y_+-y)\right]_{[3]}.\end{aligned} \tag{A.25}$$

By substituting t by t_+ into (A.9), we obtain an estimate for $[\cdot]_{[1]}$ of (A.25) as

$$\|\nabla\tilde{d}(y;t_+) - \nabla\tilde{d}_{\bar{\delta}}(y;t_+)\|^*_y \le \|\bar{x}_{\bar{\delta}1} - x^*_1\|_{x^*_1} = \delta_1. \tag{A.26}$$

Next, we consider the second term $[\cdot]_{[2]}$ of (A.25). It follows from (A.8) that

$$\left[(1-\delta_1)^2 - 1\right]\nabla^2\tilde{d}(y;t_+) \preceq \nabla^2\tilde{d}_{\bar{\delta}}(y;t_+) - \nabla^2\tilde{d}(y;t_+) \preceq \left[(1-\delta_1)^{-2} - 1\right]\nabla^2\tilde{d}(y;t_+). \tag{A.27}$$

If we define $G := \left[\nabla^2\tilde{d}_{\bar{\delta}}(y;t_+) - \nabla^2\tilde{d}(y;t_+)\right]$ and $H := \nabla^2\tilde{d}(y;t_+)^{-1/2}G\nabla^2\tilde{d}(y;t_+)^{-1/2}$, then

$$\|[\nabla^2\tilde{d}(y;t) - \nabla^2\tilde{d}_{\bar{\delta}}(y;t_+)](y_+-y)\|^*_y = \|Gp\|^*_y \le \|H\|\|p\|_y. \tag{A.28}$$

By virtue of (A.27) and the condition $\delta_1 < 1$, one has

$$\|H\| \le \max\left\{1-(1-\delta_1)^2, (1-\delta_1)^{-2}-1\right\} = (1-\delta_1)^{-2}(2\delta_1-\delta_1^2).$$

Hence, (A.28) leads to

$$\|[\nabla^2\tilde{d}(y;t) - \nabla^2\tilde{d}_{\bar{\delta}}(y;t_+)](y_+-y)\|^*_y \le (1-\delta_1)^{-2}(2\delta_1-\delta_1^2)\|p\|_y. \tag{A.29}$$

Furthermore, since $\tilde{d}(\cdot;t)$ is standard self-concordant, similarly as in the proof of [15, Theorem 4.1.14], the third term $[\cdot]_{[3]}$ of (A.25) is estimated as

$$\|\nabla\tilde{d}(y_+,t_+) - \nabla\tilde{d}(y;t_+) - \nabla^2\tilde{d}(y;t_+)(y_+-y)\|^*_y \le (1-\|p\|_y)^{-1}\|p\|^2_y. \tag{A.30}$$

Now, we apply the triangle inequality $\|a+b+c\|^*_y \le \|a\|^*_y + \|b\|^*_y + \|c\|^*_y$ to (A.25) and then plug (A.26), (A.29), and (A.30) into the resulting inequality to obtain

$$\|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|^*_y \le \delta_1 + (1-\delta_1)^{-2}(2\delta_1-\delta_1^2)\|p\|_y + (1-\|p\|_y)^{-1}\|p\|^2_y.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Finally, by substituting the last inequality into (A.24) we get (A.20).

Step 2. Next, we estimate (A.20) in terms of $\bar{\lambda}_1$ to obtain

(A.31)
$$\bar{\lambda}_+ \le (1-\delta_+)^{-1}\left[\left(\frac{\bar{\lambda}_1}{1-\delta_1-\bar{\lambda}_1}\right)^2 + \frac{(2\delta_1-\delta^2)}{(1-\delta_1)^2}\left(\frac{\bar{\lambda}_1}{1-\delta_1-\bar{\lambda}_1}\right) + \frac{(1-\delta_1)\delta_1}{1-\delta_1-\bar{\lambda}_1} + \delta_+\right].$$

Indeed, by using (A.11) with $x = \bar{x}_{\bar{\delta}1}$ and $z = x_1^*$ and then (3.10) we have

$$(1-\delta_1)^2\nabla^2\tilde{d}_{\bar{\delta}}(y;t_+) \preceq \nabla^2\tilde{d}(y;t_+) \preceq (1-\delta_1)^{-2}\nabla^2\tilde{d}_{\bar{\delta}}(y,t_+).$$

These inequalities together with the definition of $\|\!|\cdot|\!\|_y$ imply

$$(1-\delta_1)\|\!|p|\!\|_y \le \|p\|_y = \left[p^T\nabla^2 d(y;t_+)p\right]^{1/2} \le (1-\delta_1)^{-1}\|\!|p|\!\|_y.$$

Moreover, since $\|\!|p|\!\|_y = \|\!|\nabla\tilde{d}_{\bar{\delta}}(y;t_+)|\!\|_y^* = \bar{\lambda}_1$ due to (4.12), the last inequality is equivalent to

$$\|p\|_y \le (1-\delta_1)^{-1}\bar{\lambda}_1. \tag{A.32}$$

Note that the right-hand side of (A.20) is nondecreasing w.r.t. $\|p\|_y$ in $[0,1)$. Substituting (A.32) into (A.20) we finally obtain (A.31).

Step 3. We further estimate (A.31) in terms of Δ and $\bar{\lambda}$. First, we can easily check that the right-hand side of (A.31) is nondecreasing w.r.t. $\bar{\lambda}_1$, δ_1, and δ_+. Now, by using the definitions of Δ and $\bar{\lambda}$, it follows from Lemma A.1(c) that

$$\bar{\lambda}_1 \le (1-\Delta)^{-1}(\bar{\lambda}+\Delta).$$

Since $\delta_+ < 1$ and $\delta_1 + 2\Delta + \bar{\lambda} < 1$, substituting this inequality into (A.31), we obtain

$$\bar{\lambda}_+ \le (1-\delta_+)^{-1}\left[\delta_+ + \left(\frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}\right)^2 + \frac{(2\delta_1-\delta_1^2)}{(1-\delta_1)^2}\left(\frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}\right) + \frac{\delta_1(1-\delta_1)(1-\Delta)}{1-\delta_1-2\Delta-\bar{\lambda}}\right]. \tag{A.33}$$

The right-hand side of (A.33) is well-defined and nondecreasing w.r.t. all variables.

Step 4. Finally, we facilitate the right-hand side of (A.33) to obtain (4.15). Since $\bar{\lambda} \ge 0$, we have

$$\begin{aligned}(1-\delta_1)(1-\Delta) &= [1-\delta_1-2\Delta-\bar{\lambda}] + (\bar{\lambda}+\Delta) + \delta_1\Delta\\ &\le [1-\delta_1-2\Delta-\bar{\lambda}] + (1+\delta_1)(\bar{\lambda}+\Delta).\end{aligned}$$

The last inequality implies

$$\frac{\delta_1(1-\delta_1)(1-\Delta)}{1-\delta_1-2\Delta-\bar{\lambda}} \le \delta_1 + \delta_1(1+\delta_1)\left(\frac{\Delta+\bar{\lambda}}{1-\delta_1-2\Delta-\bar{\lambda}}\right). \tag{A.34}$$

Alternatively, since $0 \le \delta_1 < 1$, we have $1+\delta_1 \le \frac{1}{1-\delta_1}$. Thus

$$\begin{aligned}(1-\delta_1)^{-2}(2\delta_1-\delta_1^2) + \delta_1(1+\delta_1) &= \delta_1\left[(1-\delta_1)^{-2} + (1-\delta_1)^{-1} + (1+\delta_1)\right]\\ &\le \delta_1\left[(1-\delta_1)^{-2} + 2(1-\delta_1)^{-1}\right].\end{aligned} \tag{A.35}$$

Substituting inequality (A.34) into (A.33) and then using (A.35) and $\xi := \frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}$, we obtain (4.15).

Step 5. The nondecrease of the right-hand side of (4.15) is obvious. The inequality (4.16) follows directly from (4.15) by noting that $\bar{\lambda} \equiv \lambda$ and $\bar{x}_{\bar{\delta}} \equiv x^*$. ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

A.4. The proof of Theorem 4.4.

Proof. Let us define $\bar{\xi} := \frac{\Delta+\beta}{1-\bar{\delta}-\beta-2\Delta}$ and

$$\varphi(\beta,\bar{\delta},\Delta) := (1-\bar{\delta})^{-1}\big\{2\bar{\delta}+\bar{\xi}^2+\bar{\delta}[(1-\bar{\delta})^{-2}+2(1-\bar{\delta})^{-1}]\bar{\xi}\big\}.$$

By the assumption that $\bar{\lambda} \le \beta$, it follows from Lemma 4.3 that if $\varphi(\beta,\bar{\delta},\Delta) \le \beta$, then $\bar{\lambda}_+ \le \beta$. This condition holds if (a) $0 \le \bar{\xi} \le (\sqrt{p^2+4q}-p)/2 \equiv \theta$ and (b) $0 \le \bar{\delta} \le \beta/(\beta+2)$, where p and q are defined by (4.18). The condition (a) is equivalent to $(1+2\theta)\Delta \le \theta(1-\bar{\delta}-\beta)-\beta$. Because $\Delta > 0$, we need $\theta > (1-\bar{\delta}-\beta)^{-1}\beta$. This is guaranteed if $\mathcal{P}_{\bar{\delta}}(\beta) > 0$, where $\mathcal{P}_{\bar{\delta}}$ is defined in (4.17). By a well-known characteristic of a cubic polynomial, $\mathcal{P}_{\bar{\delta}}(\beta)$ has three real roots if $18c_0c_1c_2c_3 - 4c_2^3c_0 + c_2^2c_1^2 - 4c_3c_1^3 - 27c_3^2c_0^2 \ge 0$. By numerically checking the last condition, we can show that if $0 \le \bar{\delta} \le \bar{\delta}_{\max} := 0.043286$, then the three roots satisfy $0 \le \beta_* < \beta^* < \beta_3$ and $\mathcal{P}_{\bar{\delta}}(\beta) > 0$ for all $\beta \in (\beta_*, \beta^*)$. With such values of $\bar{\delta}$ and β we have $\theta > (1-\bar{\delta}-\beta)^{-1}\beta$ and the condition (b) is also satisfied. Eventually, if we define $\bar{\Delta} := \frac{\theta(1-\bar{\delta}-\beta)-\beta}{1+2\theta} > 0$ and choose $\bar{\delta}$, β, and Δ such that $0 \le \bar{\delta} \le \bar{\delta}_{\max}$, $\beta \in (\beta_*, \beta^*)$ and $0 \le \Delta \le \bar{\Delta}$, then $\bar{\lambda} \le \beta$ implies $\bar{\lambda}_+ \le \beta$. ∎

Acknowledgments. The authors would like to thank the associate editor and the two anonymous referees for their insightful comments and suggestions that helped us to improve this paper.

REFERENCES

[1] D. S. Bernstein, *Matrix Mathematics: Theory, Facts and Formulas with Application to Linear Systems Theory*, Princeton University Press, Princeton, NJ, Oxford, UK, 2005.
[2] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice–Hall, Englewood Cliffs, NJ, 1989.
[3] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.
[4] G. Chen and M. Teboulle, *A proximal-based decomposition method for convex minimization problems*, Math. Programming, 64 (1994), pp. 81–101.
[5] A. J. Connejo, R. Mínguez, E. Castillo, and R. García-Bertrand, *Decomposition Techniques in Mathematical Programming: Engineering and Science Applications*, Springer-Verlag, Berlin, 2006.
[6] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), pp. 201–213.
[7] M. Fukuda, M. Kojima, and M. Shida, *Lagrangian dual interior-point methods for semidefinite programs*, SIAM J. Optim., 12 (2002), pp. 1007–1031.
[8] N. Garg and J. Könemann, *Faster and simpler algorithms for multicommodity flow and other fractional packing problems*, SIAM J. Comput., 37 (2007), pp. 630–652.
[9] K. Holmberg and K. C. Kiwiel, *Mean value cross decomposition for nonlinear convex problem*, Optim. Methods Softw., 21 (2006), pp. 401–417.
[10] M. Kojima, N. Megiddo, S. Mizuno, and S. Shindoh, *Horizontal and Vertical Decomposition in Interior Point Methods for Linear Programs*, Technical report, Information Sciences, Tokyo Institute of Technology, Tokyo, 1993.
[11] N. Komodakis, N. Paragios, and G. Tziritas, *MRF energy minimization & beyond via dual decomposition*, IEEE Trans Pattern Anal. Mach. Intell., 33 (2011), pp. 531–552.
[12] S. Mehrotra and M. G. Ozevin, *Decomposition based interior point methods for two-stage stochastic convex quadratic programs with recourse*, Oper. Res., 57 (2009), pp. 964–974.
[13] I. Necoara and J. A. K. Suykens, *Applications of a smoothing technique to decomposition in convex optimization*, IEEE Trans. Automat. Control, 53 (2008), pp. 2674–2679.
[14] I. Necoara and J. A. K. Suykens, *Interior-point Lagrangian decomposition method for separable convex optimization*, J. Optim. Theory Appl., 143 (2009), pp. 567–588.
[15] Y. Nesterov, *Introductory Lectures on Convex Optimization*, Kluwer, Boston, 2004.
[16] Y. Nesterov, *Smooth minimization of nonsmooth functions*, Math. Program., 103 (2005), pp. 127–152.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[17] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.
[18] J. Renegar, *A Mathematical View of Interior-Point Methods in Convex Optimization*, SIAM, Philadelphia, 2001.
[19] A. Ruszczyński, *On convergence of an augmented Lagrangian decomposition method for sparse convex optimization*, Math. Oper. Res., 20 (1995), pp. 634–656.
[20] S. Samar, S. Boyd, and D. Gorinevsky, *Distributed estimation via dual decomposition*, in Proceedings of the European Control Conference (ECC), Kos, Greece, 2007, pp. 1511–1516.
[21] M. Shida, *An interior-point smoothing technique for Lagrangian relaxation in large-scale convex programming*, Optimization, 57 (2008), pp. 183–200.
[22] Q. Tran Dinh, C. Savorgnan, and M. Diehl, *Combining Lagrangian decomposition and excessive gap smoothing technique for solving large-scale separable convex optimization problems*, Comput. Optim. Appl., DOI 10.1007/s10589-012-9515-6, 2012.
[23] E. Wei, A. Ozdaglar, and A. Jadbabaie, *A distributed Newton method for network utility maximization*, IEEE Trans. Automat. Control, to appear.
[24] A. Venkat, I. Hiskens, J. Rawlings, and S. Wright, *Distributed MPC strategies with application to power system automatic generation control*, IEEE Trans. Control Syst. Technol., 16 (2008), pp. 1192–1206.
[25] L. Xiao, M. Johansson, and S. Boyd, *Simultaneous routing and resource allocation via dual decomposition*, IEEE Trans. Commun., 52 (2004), pp. 1136–1144.
[26] G. Zhao, *Interior point methods with decomposition for solving large-scale linear programs*, J. Optim. Theory Appl., 102 (1999), pp. 169–192.
[27] G. Zhao, *A log-barrier with Benders decomposition for solving two-stage stochastic programs*, Math. Program., 90 (2001), pp. 507–536.
[28] G. Zhao, *A Lagrangian dual method with self-concordant barriers for multistage stochastic convex programming*, Math. Program., 102 (2005), pp. 1–24.

Downloaded 03/17/15 to 128.179.132.207. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.