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. In this paper, we propose an inexact perturbed path-following algorithm in the framework of Lagrangian dual decomposition for solving large-scale structured convex optimization problems. Unlike the exact versions considered in literature, we propose to solve the primal problem inexactly up to a given accuracy. The inexact perturbed algorithm allows to use both approximate Hessian matrices and approximate gradient vectors to compute Newton-type directions for the dual problem. The algorithm is divided into two phases. The first phase computes an initial point which makes use of inexact perturbed damped Newton-type iterations, while the second one performs the path-following algorithm with inexact perturbed full-step Newton-type iterations. We analyze the convergence of both phases and estimate the worst-case complexity. As a special case, an exact path-following algorithm for Lagrangian relaxation is derived and its worst-case complexity is estimated. This variant possesses some differences compared to previously known methods. Implementation details are discussed and numerical results are reported.

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

1. Introduction.

Many optimization problems arising in networked systems, image processing, data mining, economics, distributed model predictive control and multi-stage stochastic optimization can be formulated as a separable convex optimization problem, see, e.g. [6, 11, 13, 15, 20, 25, 26]. If the optimization problem is moderate size or possesses a sparsity structure, then it can be solved efficiently by standard optimization methods. In many practical situations, we can encounter problems which may not be easy to solve by standard optimization algorithms due to the high dimensionality or the distributed locations of the data and devices. However, many problems can be reformulated as separable convex optimization problems such that the subproblems generated from their components can be solved in a closed form or easier than the full problem.

In this paper, we are interested in the following convex separable optimization problem:

$$
\begin{cases}
\max\limits_{x\in\mathbb{R}^n} & \Big\{\phi(x) := \sum\limits_{i=1}^{M}\phi_i(x_i)\Big\}\\
\text{s.t.} & x_i \in X_i,\ (i = 1,\cdots,M),\\
& \sum\limits_{i=1}^{M} A_i x_i = b,
\end{cases}
\tag{1.1}
$$

where $x = (x_1^T,\ldots,x_M^T)^T$ with $x_i\in\mathbb{R}^{n_i}$ is a vector of decision variables, $\phi_i : \mathbb{R}^{n_i}\to\mathbb{R}$ is concave, X_i is a nonempty, closed convex subset in $\mathbb{R}^{n_i}$, $A_i\in\mathbb{R}^{m\times n_i}$, $b\in\mathbb{R}^m$ for all $i = 1,\ldots,M$, and $n_1 + n_2 + \cdots + n_M = n$. The last constraint is usually referred to as a *linear coupling constraint*. Problems of the form (1.1) were considered

[*]Department of Electrical Engineering (ESAT-SCD) and Optimization in Engineering Center (OPTEC), K.U. Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium ({quoc.trandinh, carlo.savorgnan, moritz.diehl}@esat.kuleuven.be).

[†] Department of Mathematics-Mechanics-Informatics, Hanoi University of Science, Hanoi, Vietnam.

[‡] Automation and Systems Engineering Department, University Politehnica Bucharest, 060042 Bucharest, Romania; (ion.necoara@upb.ro)

in many research papers, see, e.g. [2, 14, 15, 23]. Note that linear inequality coupling constraints of the form $\sum_{i=1}^{M} B_i x_i \leq d$ can also be formulated into (1.1) by using slack variables, see, e.g. [15].

Several methods can solve problem (1.1) by decomposing it into smaller subproblems that can be solved separately by standard optimization techniques. For instance, by applying Lagrangian relaxation, the coupling constraint can be brought into the objective function and, by the separability, we can decompose the dual function into smaller subproblems [2]. However, using such a Lagrangian relaxation technique generally leads to a nonsmooth optimization problem. There are several attempts to overcome this difficulty by smoothing the dual function. One can add an augmented Lagrangian term or a proximal term to the objective function of the problem. Unfortunately, the first approach breaks the separability of the original problem due to the cross terms between the components. Therefore, the second approach is more suitable for this type of problems.

Recently, smoothing techniques in convex optimization have attracted increasing interest and have found many applications [18]. In the framework of the Lagrangian dual decomposition, there are two popular approaches. The first approach is regularization. By adding a regularization term as a proximal term to the objective function, the primal subproblem becomes strongly convex. Consequently, the master dual problem is smooth which allows one to apply smoothing optimization techniques [3, 5, 14, 23]. The second approach is using barrier functions, this technique is suitable for problems with conic constraints [7, 10, 12, 15, 22, 27, 28, 29]. Several methods in this direction are based on the fact that by using a self-concordant log-barrier function, the family of the dual functions which depend on a barrier parameter is strongly self-concordant in the sense of Nesterov and Nemirovski [16] under certain assumptions. Consequently, path-following methods can be used to solve the master dual problem. Note that this technique is only applicable to the cases where either the objective function is linear, quadratic and self-concordant or the problem is compatible in the sense that it possesses a property that makes the smooth objective function of the dual self-concordant. Several methods in this direction require the crucial assumption that the primal subproblems are solved exactly. In practice, solving exactly the primal subproblems to compute the dual function is only conceptual. Any numerical optimization method provides an approximate solution and, consequently, the dual function is also approximated. This paper studies an inexact perturbed path-following method in the framework of Lagrangian decomposition for solving (1.1).

Contribution. The contribution of this paper is fivefold.

1. By applying smoothing technique via self-concordant barrier functions, we provide a local and a global smooth approximation to the dual function and estimate the approximation error.
2. A new inexact perturbed path-following decomposition algorithm is proposed for solving (1.1). The algorithm consists of two phases. Both phases allow the primal subproblems to be solved approximately. Moreover, the algorithm is highly parallelizable.
3. The convergence theory is investigated under standard assumptions used in any interior point method and the worst-case complexity is estimated.
4. When the primal problem is assumed to be solved exactly, our method reduces to the path-following method for Lagrangian decomposition considered in [12, 15, 22, 29]. However, the variants presented in this papers possess a larger neighborhood of the analytic center where convergence is guaranteed.

5. The implementation details are discussed and numerical experiments are implemented to confirm the theoretical development.

Let us emphasize some differences between the method presented in this paper and previously known methods:

1. Even though smoothing techniques based on self-concordant barriers are not new, in this paper we do not only apply smoothing techniques to the dual problem but also provide some properties of the smooth function. The smooth approximation of the dual function only requires that the objective function is convex (not necessarily smooth). Moreover, since the dual function is smooth, it is possible to apply derivative based, smooth optimization techniques such as gradient-based methods or sequential quadratic programming-based (SQP) methods to solve the master problem.
2. The new algorithm allows us to solve the primal subproblems inexactly where we can control the accuracy up to $\delta^* \approx 0.043286$ (see Section 4 for more details) such that at the early steps of the path-following algorithm, they can be solved very inexactly. This point is significant if the primal subproblems require high computational cost. Note that the algorithm developed in this paper is different from the one considered in [27] for linear programming, where the inexactness of the primal subproblems is defined in a different way.
3. Based on a recent monograph [17], we directly analyze the convergence of the algorithm. This makes our theory self-contained. Moreover, it also allows us to optimally choose the parameters and to trade-off between the convergence rate of the master problem and the accuracy of the primal subproblems.
4. In the exact case, the variant in this paper still has some advantages compared to the previous ones. Firstly, the radius of the neighborhood of the analytic center is $(3-\sqrt{5})/2 \approx 0.38197$ which is larger than $2-\sqrt{3} \approx 0.26795$ of previous methods. Secondly, 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 optimal way.

The rest of this paper is organized as follows. In the next section, we briefly describe the Lagrangian dual decomposition method applied to separable convex optimization. Section 3 deals with a smoothing technique for the dual function via self-concordant barriers and investigates the main properties of the smooth dual function. Section 4 presents an inexact perturbed path-following decomposition algorithm. The convergence of the algorithm is analyzed and the worst-case complexity is estimated. Section 5 considers an exact variant of the algorithm presented in Section 4. Section 6 discusses implementation details of the algorithms. Section 7 presents numerical tests and a comparison. Concluding remarks are included in the last section. The proofs of the technical statements are given in the appendix.

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

For a proper, lower semi-continuous convex function f, the notation $\mathrm{dom}(f)$ 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 $\partial f(x)$ as the "superdifferential" of f at x, where $\partial f(x) := -\partial\{-f(x)\}$. Let f be twice continuously differentiable and convex on $\mathbb{R}^n$. For a given vector u, the local norm of u with respect to f at x, where $\nabla^2 f(x)$ is positive definite, is defined as $\|u\|_x := \left[u^T\nabla^2 f(x)u\right]^{1/2}$ and its dual norm is $\|u\|_x^* := \max\{u^Tv \mid \|v\|_x \leq 1\} = \left[u^T\nabla^2 f(x)^{-1}u\right]^{1/2}$. Clearly, $\left|u^Tv\right| \leq \|u\|_x\|v\|_x^*$. Let F be a standard self-concordant function, $W^0(x,r) := \{z \in \mathbb{R}^n \mid \|z-x\|_x < r\}$ defines the *Dikin* ellipsoid of F at x, where $\|z-x\|_x = [(z-x)^T\nabla^2 F(x)(z-x)]^{1/2}$.

For a given symmetric matrix P in $\mathbb{R}^{n\times n}$, the expression $P \succeq 0$ (resp. $P \succ 0$) means that P is positive semi-definite (resp. positive definite); $P \succeq Q$ and $P \preceq Q$ (resp. $P \succ Q$ and $P \prec Q$) mean that $P-Q$ and $Q-P$ are positive semidefinite (resp. positive definite), respectively.

The notation $\mathbb{R}_+$ and $\mathbb{R}_{++}$ define the set of non-negative and positive numbers, respectively. 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.

2. Lagrangian dual relaxation in convex optimization. A classical technique to address coupling constraints in separable convex optimization is based on Lagrangian relaxation [2]. We briefly review such a technique in this section.

Without loss of generality we consider problem (1.1) with $M = 2$. The separable convex optimization problem (1.1), with $M = 2$, can be expressed as:

$$\phi^* := \begin{cases} \max\limits_{x:=(x_1,x_2)} & \{\phi(x) := \phi_1(x_1) + \phi_2(x_2)\} \\ \text{s.t.} & A_1x_1 + A_2x_2 = b, \\ & x \in X := X_1 \times X_2. \end{cases} \tag{2.1}$$

Let us define $A := [A_1, A_2]$ and $n := n_1 + n_2$. The linear coupling constraint $A_1x_1 + A_2x_2 = b$ can be written as $Ax = b$. The Lagrange function for problem (2.1) with respect to the coupling constraint $A_1x_1 + A_2x_2 = b$ is defined as:

$$L(x,y) := \phi(x) + y^T(Ax-b) = \phi_1(x_1) + \phi_2(x_2) + y^T(A_1x_1 + A_2x_2 - b),$$

where $y \in \mathbb{R}^m$ is the Lagrange multiplier associated with the coupling constraint. A pair $(x_0^*, y_0^*) \in X \times \mathbb{R}^m$ is called a saddle point of L if

$$L(x, y_0^*) \leq L(x_0^*, y_0^*) \leq L(x_0^*, y),\ \forall x \in X,\ \forall y \in \mathbb{R}^m.$$

The dual problem of (2.1) is

$$d_0^* := \min_{y\in\mathbb{R}^m} d_0(y), \tag{2.2}$$

where d_0 is the dual function which is defined as

$$d_0(y) := \max_{x\in X}\left\{\phi_1(x_1) + \phi_2(x_2) + y^T(A_1x_1 + A_2x_2 - b)\right\}. \tag{2.3}$$

If *strong duality* holds at (x_0^*, y_0^*) with $x_0^* := (x_{0,1}^*, x_{0,2}^*) \in X$ and $y_0^* \in \mathbb{R}^m$, then we have [4]:

$$d_0^* = d_0(y_0^*) = \min_{y\in\mathbb{R}^m} d_0(y) = \max_{x\in X}\{\phi(x) \mid Ax = b\} = \phi(x_0^*) = \phi^*.$$

Let us denote by X^* the solution set of (2.1) and by Y^* the solution set of the dual problem (2.2). It is well-known that if either the Slater condition holds, i.e.

$\mathrm{ri}(X) \cap \{x \in \mathbb{R}^n \mid Ax = b\} \neq \emptyset$, where $\mathrm{ri}(X)$ is the relative interior of the convex set X, or X is polyhedral, then Y^* is bounded [4].

Finally, it is important to notice that the dual function $d_0(\cdot)$ can be computed separately by

$$\text{(2.4)} \qquad \text{where} \quad \begin{aligned} & d_0(y) = d_{0,1}(y) + d_{0,2}(y) - b^T y, \\ & d_{0,i}(y) := \max_{x_i \in X_i} \left\{ \phi_i(x_i) + y^T A_i x_i \right\}, \ i = 1, 2. \end{aligned}$$

Let $x^*_{0,i}(y)$ be a solution of the maximization problem in (2.3) $(i = 1, 2)$, and $x^*_0(y) := (x^*_{0,1}(y), x^*_{0,2}(y))$. Lagrangian relaxation generally leads to a nonsmooth optimization problem in the dual form. Consequently, a numerical solution to the dual problem encounters many drawbacks.

3. Smoothing technique via self-concordant barriers. Let us assume that the feasible set X_i is convex, has nonempty interiors and possesses a ν_i-self-concordant barrier F_i for $i = 1, 2$. Theory of self-concordant functions and self-concordant barriers can be found in [8, 16, 17]. Throughout the paper, we use the following assumptions.

Assumption A.1.

(a) *The solution set X^* of* (2.1) *is nonempty. Either the Slater condition for* (2.1) *is satisfied or X is polyhedral.*

(b) *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 parameter ν_i for $i = 1, 2$.*

(c) *The function ϕ_i is proper, upper semicontinuous and concave on X_i for $i = 1, 2$.*

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

Note that Assumptions A.1.a) and A.1.c) are standard in convex optimization, which guarantee the solvability of the problem and strong duality. Assumption A.1.b) can be satisfied by assuming that the set of the sample points generated by such an optimization algorithm is bounded. Assumption A.1.d) is not restrictive since it can be satisfied by applying standard linear algebra techniques to eliminate redundant constraints.

Remark 1. *As we can see in Section 6, the convex feasible set X_i can be given as follows*

$$X_i := X_i^c \cap X_i^a, \quad X_i^a := \{x_i \in \mathbb{R}^{n_i} \mid E_i x_i = f_i\},$$

where $\mathrm{int}(X_i^c)$ is nonempty and X_i^c possesses a ν_i-self-concordant barrier F_i. Let $E = [E_1, E_2]$ be a matrix formed from E_i and A/E be a reduced form of $\begin{bmatrix} A \\ E \end{bmatrix}$ and $\mathrm{int}(X_i) := \mathrm{int}(X_i) \cap X_i^a$ for $i = 1, 2$. In this case, the theory developed in the next sections can be extended to the problem with this constraint, see, e.g. [15].

Let us denote by x_i^c the *analytic center* of X_i, which is defined as:

$$x_i^c := \operatorname*{argmin}_{x_i \in \mathrm{ri}(X_i)} F_i(x_i), \ i = 1, 2.$$

Under Assumption A.1.b), $x^c := (x_1^c, x_2^c)$ is well-defined due to [19, Corollary 2.3.6]. To compute x^c, one can apply the algorithms proposed in [17, pp. 204–205]. Moreover, the following estimates hold:

$$\text{(3.1)} \qquad F_i(x_i) - F_i(x_i^c) \geq \omega(\|x_i - x_i^c\|_{x_i^c}) \text{ and } \|x_i - x_i^c\|_{x_i^c} \leq \nu_i + 2\sqrt{\nu_i},$$

for all $x_i \in \overline{\mathrm{dom}}(F_i)$ and $i = 1, 2$ [17, Theorems 4.1.13 and 4.2.6].

3.1. A smooth approximation of the dual function. Similarly to [10, 15, 22, 29], we construct a smooth approximation of the nonsmooth dual function d_0 defined by (2.3) via self-concordant barriers.

Let us define the following functions:

$$
\begin{aligned}
&d_i(y,t) := \max_{x_i\in\mathrm{int}(X_i)}\left\{\phi_i(x_i)+y^TA_ix_i-t[F_i(x_i)-F_i(x_i^c)]\right\},\ i=1,2,\\
&\text{and}\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (3.2)\\
&d(y,t) := d_1(y,t)+d_2(y,t)-b^Ty,
\end{aligned}
$$

where $t > 0$ is referred to as a smoothness or barrier parameter. Note that, due to the strict concavity of the objective function, the maximization problem in (3.2) has a unique solution, which is denoted by $x_i^*(y,t)$. Consequently, the functions $d_i(\cdot,t)$ $(i = 1,2)$ and $d(\cdot,t)$ are well-defined and smooth on $\mathbb{R}^m$ for any $t > 0$. As in [29] we refer to d as a *smooth dual approximation* of d_0 and to the maximization problem in (3.2) as a *primal subproblem*.

If we denote by $x^*(y,t) := (x_1^*(y,t), x_2^*(y,t))$, then we can write

$$
d(y,t) = \phi(x^*(y,t)) + y^T(Ax^*(y,t)-b) - t[F(x^*(y,t)) - F(x^c)].
$$

The optimality condition for (3.2) is

$$
0 \in \partial\phi_i(x_i^*(y,t)) + A_i^Ty - t\nabla F_i(x_i^*(y,t)),\ i=1,2, \tag{3.3}
$$

where $\partial\phi_i(x_i^*(y,t))$ is the super-differential of ϕ_i at $x_i^*(y,t)$ $(i = 1,2)$. Since problem (3.2) is convex, this condition is necessary and sufficient.

Associated with the smooth dual function $d(\cdot,t)$, we consider the following *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 by $x^*(t) := x^*(y^*(t),t)$.

For a given $\beta \in (0,1)$, we define a neighbourhood in $\mathbb{R}^m$ with respect to F_i and $t > 0$ as

$$
\mathcal{N}_t^{F_i}(\beta) := \left\{y\in\mathbb{R}^m \mid \lambda_{F_i}(x_i^*(y,t)) := \|\nabla F_i(x_i^*(y,t))\|^*_{x_i^*(y,t)} \le \beta\right\}.
$$

The following lemma provides a local estimate for $d_0(\cdot)$, whose proof can be found in the appendix.

Lemma 3.1. *Under Assumption A.1 and $\beta \in (0,1)$, the function $d(\cdot,t)$ defined by* (3.2) *satisfies:*

$$
0\le t\left[\sum_{i=1}^{2}\omega\big(\|x_i^*(y,t)-x_i^c\|_{x_i^c}\big)\right]\le d_0(y)-d(y,t)\le t\sum_{i=1}^{2}[\omega_*(\lambda_{F_i}(x_i^*(y,t)))+\nu_i], \tag{3.5}
$$

for all $y \in \mathcal{N}_t^{F_1}(\beta)\cap\mathcal{N}_t^{F_2}(\beta)$.

From Lemma 3.1, we see that

$$
0 \le d_0(y) - d(y,t) \le t[2\omega_*(\beta)+\nu_1+\nu_2],\quad \forall y\in\mathcal{N}_t^{F_1}(\beta)\cap\mathcal{N}_t^{F_2}(\beta).
$$

Hence, for $t = t_f > 0$ sufficiently small, $d(\cdot,t_f)$ is a local approximation to $d_0(\cdot)$.

Under Assumption A.1, the dual optimal solution set Y^* is bounded. Without loss of generality, we can assume that Y is bounded such that $Y^* \subset Y$. Let

$$d^c(y) := \phi(x^c) + y^T(Ax^c - b),$$

where x^c is the analytic center of X. From (2.3) we have:

$$d_0(y) - d^c(y) = \max_{x\in X}\left\{\phi(x) + y^T(Ax-b)\right\} - \left[\phi(x^c) + y^T(Ax^c - b)\right] \geq 0,\ \forall y \in Y.$$

Furthermore,

$$\begin{aligned}
0 \leq d_0(y) - d^c(y) &= \max_{x\in X}\{\phi(x) - \phi(x^c) + y^T A(x - x^c)\} \\
&\overset{\phi \text{ is concave}}{\leq} \sum_{i=1}^{2} \max_{x_i\in X_i}\left\{\max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\left[\xi_i + A_i^T y\right]^T(x_i - x_i^c)\right\}\right\} \\
&\leq \sum_{i=1}^{2} \max_{x_i\in X_i}\left\{\max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\|\xi_i + A_i^T y\|^*_{x_i^c}\|x_i - x_i^c\|_{x_i^c}\right\}\right\} \\
&\overset{(3.1)}{\leq} \sum_{i=1}^{2}(\nu_i + 2\sqrt{\nu_i}) \max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\|\xi_i + A_i^T y\|^*_{x_i^c}\right\} \\
&\leq K_1 + K_2 < +\infty,\ \forall y \in Y,
\end{aligned} \tag{3.6}$$

where $K_i := (\nu_i + 2\sqrt{\nu_i})\max_{\xi_i\in\partial\phi_i(x_i^c)}\left\{\|\xi_i + A_i^T y\|^*_{x_i^c}\right\}$ $(i = 1, 2)$. The following lemma shows that $d(\cdot, t)$ is a global approximation to $d_0(\cdot)$. The proof can be found in the appendix.

Lemma 3.2. *Suppose that Assumption A.1 is satisfied. Then, for any $t > 0$ and $y \in Y$, the following estimate holds:*

$$0 \leq t\sum_{i=1}^{2}\omega(\|x_i^*(y,t) - x_i^c\|_{x_i^c}) \leq d_0(y) - d(y,t) \leq t[\bar{\zeta}(K_1;\nu_1,t) + \bar{\zeta}(K_2;\nu_2,t)], \tag{3.7}$$

where $\bar{\zeta}(\tau; a, b) := a\left(1 + \max\left\{0, \ln\left(\frac{\tau}{ab}\right)\right\}\right)$ and K_1 and K_2 are two constants given in (3.6)*.*

The proof of the following statement can also be found in the appendix.

Lemma 3.3. *For a given tolerance $\varepsilon_d > 0$, if we choose $t > 0$ such that*

$$0 < t \leq \bar{t} := \min\left\{\frac{K_1}{\nu_1}\kappa^{1/\kappa},\ \frac{K_2}{\nu_2}\kappa^{1/\kappa},\ \varepsilon_d^{1/(1-\kappa)}\left(\sum_{i=1}^{2}\nu_i + (K_i/\nu_i)^\kappa\right)^{-1/(1-\kappa)}\right\}, \tag{3.8}$$

for a fixed $\kappa \in (0, 1)$, then it follows from Lemma 3.2 that

$$d(y,t) \leq d_0(y) \leq d(y,t) + \varepsilon_d.$$

In other words, if we fix $t_f \in (0, \bar{t})$ and minimize $d(\cdot, t_f)$ over Y, then $y^*(t_f)$ is an ε_d-solution of (2.2)*.*

Since $d(\cdot, t)$ is continuously differentiable, smooth optimization techniques such as gradient-based or SQP-based methods can be applied to solve problem (3.4). If we choose $t_f > 0$ sufficiently small, then according to Lemmas 3.1 and 3.2, we can obtain an approximate solution of (2.2) with a desired accuracy.

3.2. The self-concordance of the smooth dual function. If the function $-\phi_i$ is self-concordant on $\mathrm{dom}(-\phi_i)$ with parameter M_{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 as stated in the following lemma. The proof of this lemma can be found, for instance, in [12, 15, 22, 29].

LEMMA 3.4. *Suppose that Assumption A.1 is satisfied. Suppose further that $-\phi_i$ is M_{ϕ_i}-self-concordant. Then, the function $d_i(\cdot,t)$ defined by* (3.2) *is self-concordant with the parameter $M_{d_i} := \max\{M_{\phi_i}, \frac{2}{\sqrt{t}}\}$ for any $t > 0$ and $i = 1,2$. Consequently, $d(\cdot,t)$ is self-concordant with the parameter $M_d = \max\{M_{\phi_1}, M_{\phi_2}, \frac{2}{\sqrt{t}}\}$.*

Similar to standard path-following methods [16, 17], in the following discussion, we assume that ϕ_i is linear as stated in Assumption A.2 below.

ASSUMPTION A.2. *The function ϕ_i is linear, i.e. $\phi_i(x_i) := c_i^T x_i$ for $i = 1,2$.*
Let $c := (c_1, c_2)$ be the vector formed from c_i $(i = 1,2)$. Assumption A.2 implies that $tF - \phi$ is $\frac{2}{\sqrt{t}}$-self-concordant. According to Lemma 3.4, $d_i(\cdot,t)$ is $\frac{2}{\sqrt{t}}$-self-concordant. Since ϕ_i is linear, if we denote by $F(x) := F_1(x_1) + F_2(x_2)$ the self-concordant barrier of X with the parameter $\nu := \nu_1 + \nu_2$, then the optimality condition (3.9) is reduced to

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

The following lemma provides an explicit formula for the derivatives of $d(\cdot,t)$. The proof can be found in [15, 29].

LEMMA 3.5. *Suppose that Assumptions* A.1 *and* A.2 *are satisfied. Then the first and second order derivatives of $d(\cdot,t)$ on Y are respectively given as*

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

where $x^*(y,t) = (x_1^*(y,t), x_2^*(y,t))$ is the solution of the primal subproblem in (3.2).

Note that since A is full-row rank and $\nabla^2 F(x^*(y,t))$ is positive definite, matrix $\nabla^2 d(y,t)$ is nonsingular for any $y \in Y$. Moreover, since $F(x)$ and ϕ are separable, the Hessian matrix $\nabla^2 F$ is block diagonal and they can also be evaluated *in parallel*, see Section 6 for more details about implementation issues.

Now, since $d(\cdot,t)$ is $\frac{2}{\sqrt{t}}$ self-concordant, if we define

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

then $\tilde{d}(\cdot,t)$ is standard self-concordant, i.e. $M_{\tilde{d}} = 2$, due to [17, Corollary 4.1.2]. For a given vector $v \in \mathbb{R}^m$, we define the norm $\|v\|_y$ with respect to $\tilde{d}(\cdot,t)$ as $\|v\|_y := [v^T \nabla^2 \tilde{d}(y,t) v]^{1/2}$.

3.3. Recovering the optimality and the feasibility. It remains to show the relations between the master problem (3.4), the dual problem (2.2) and the original primal problem (2.1). We first prove the following lemma.

LEMMA 3.6. *Let Assumption A.1 be satisfied. Then:*

a) *$d(y,\cdot)$ is non-increasing in $\mathbb{R}_{++}$ for a given $y \in Y$.*
b) *$d^*(\cdot)$ defined by* (3.4) *is differentiable and non-increasing in $\mathbb{R}_{++}$.*
c) *It holds that $d^*(t) \le d_0^*$ and $\lim_{t\downarrow 0^+} d^*(t) = d_0^* = \phi^*$. Moreover, $x^*(t)$ is feasible for problem* (2.1).

Proof. Since the function $\xi(x,y,t) := \phi(x)+y^T(Ax-b)-t[F(x)-F(x^c)]$ is strictly concave and linear in t, it is well-known that $d(y,t) = \max_{x\in\mathrm{int}(X)} \xi(x,y,t)$ is differentiable with respect to t and its derivative is given by $\eta'(t) = -[F(x^*(y,t)) - F(x^c)] \leq -\omega(\|x^*(y,t)-x^c\|_{x^c}) \leq 0$ by (3.1). Thus $d(y,\cdot)$ is nonincreasing in t which proves a).

Now, we prove b) and c). From the definitions of $d^*(\cdot)$, $d(y,\cdot)$ and $y^*(\cdot)$ in (3.4), by using strong duality, we have

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

It follows from the fourth line of (3.12) that $d^*(\cdot)$ is differentiable and nonincreasing in $\mathbb{R}_{++}$. Moreover, since x^c is the analytic center of X, we have $F(x^*(t)) - F(x^c) \geq \omega(\|x^*(t)-x^c\|_{x^c})$ due to (3.1). This inequality implies that $d^*(t) \leq \phi(x^*(t)) \leq \phi^* = d_0^*$. On the other hand, from the fourth line of (3.12), we also deduce that $x^*(t)$ is feasible to (2.1). Furthermore, since $d^*(\cdot)$ is continuous on $\mathbb{R}_{++}$, we have $\lim_{t\downarrow 0^+} d^*(t) = d_0^*$ which proves c). □

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 $d(\cdot,t)$ and $d^*(t)$.

Lemma 3.7. *Suppose that Assumption A.1 is satisfied. Then, for any $y\in Y$ and $t>0$ such that $\lambda_{\tilde{d}(\cdot,t)}(y) < 1$, one has*

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

Consequently, it holds that

$$
d(y,t) - d_0^* = d(y,t) - \phi^* \leq t\omega_*(\lambda_{\tilde{d}(\cdot,t)}(y)). \tag{3.15}
$$

Proof. Since $\tilde{d}(\cdot,t)$ is standard self-concordant, for any $y\in Y$ such that $\lambda_{\tilde{d}(\cdot,t)}(y) < 1$, and $y^*(t) = \operatorname{argmin}_{y\in Y}\tilde{d}(y,t)$, by applying [17, Theorem 4.1.13, inequality 4.1.17], we have

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

This inequality is indeed (3.14) due to (3.11). To prove (3.15), we note that $d^*(t) - d_0^* \leq 0$ by Lemma 3.5 c), adding this inequality to (3.14) and noting that $d_0^* = \phi^*$ we obtain (3.15). □

We can also estimate a lower bound for $d^*(t) - d_0^*$. Since F is convex, by using (3.1), we have

$$
F(x) - F(x^c) \leq \nabla F(x)^T(x-x^c) \leq \|\nabla F(x)\|_{x^c}^*\|x-x^c\|_{x^c} \leq (\nu+2\sqrt{\nu})\|\nabla F(x)\|_{x^c}^*.
$$

Since X is bounded and ∇F is continuous, using the above inequality, we have $c_X^F := \max_{x\in X}\|\nabla F(x)\|_{x^c}^* < +\infty$. Thus it follows from the last inequality that $\max_{x\in X}\{F(x) - F(x^c)\} \le (\nu + 2\sqrt{\nu})c_X^F < +\infty$. Moreover, for any functions u, v on Z, we have $\max\limits_{z\in Z}\{u(z) - v(z)\} \ge \max\limits_{z\in Z} u(z) - \max\limits_{z\in Z} v(z)$. Finally, we estimate $d^*(t) - d_0^*$ as

$$
\begin{aligned}
d^*(t) = \min_{y\in Y} d(y,t) &= \min_{y\in Y}\left\{\max_{x\in\mathrm{int}(X)}\{L(x,y) - t[F(x) - F(x^c)]\}\right\}\\
&\ge \min_{y\in Y}\left\{\max_{x\in\mathrm{int}(X)}\{L(x,y)\} - t\max_{x\in\mathrm{int}(X)}\{F(x) - F(x^c)\}\right\}\\
&\ge \min_{y\in Y}\max_{x\in X} L(x,y) - t\max_{x\in\mathrm{int}(X)}\{F(x) - F(x^c)\}\\
&\ge d_0^* - t(\nu + 2\sqrt{\nu})c_X^F.
\end{aligned}
$$

Combining this inequality with (3.14) we obtain

$$
d(y,t) - d_0^* \ge t\left[\omega(\lambda_{\tilde{d}(\cdot,t)}(y)) - (\nu + 2\sqrt{\nu})c_X^F\right].
$$

Now, we define an approximate solution of the dual problem (2.2) as follows:

Definition 3.8. *For a given tolerance $\varepsilon_d > 0$, a point $y^*(t)$ is said to be an ε_d-solution of* (2.2) *if $0 \le d_0^* - d^*(t) \le \varepsilon_d$.*

Let $y^*(t)$ be an ε_d-solution of (2.2) and $y \in Y$ such that $\lambda = \lambda_{\tilde{d}(\cdot,t)}(y) \le \beta$ for a fixed $\beta \in (0,1)$. We have

$$
0 \le d_0^* - d(y,t) \le |d(y,t) - d^*(t)| + |d^*(t) - d_0^*| \le \varepsilon_d + t\omega_*(\lambda_{\tilde{d}(\cdot,t)}(y)) \le \varepsilon_d + \omega_*(\beta)t.
$$

Consequently, if we choose t such that $t \le \omega_*(\beta)^{-1}\varepsilon_d$ then

$$
0 \le d_0^* - d(y,t) = \phi^* - d(y,t) \le 2\varepsilon_d. \tag{3.16}
$$

The algorithms presented in the next sections aim to find a $2\varepsilon_d$-approximate solution of the dual problem (2.2) in the sense of (3.16). Thus $d(y,t)$ is a $2\varepsilon_d$-approximation of the optimal value ϕ^*.

It remains to quantify the feasibility gap of the original problem (2.1) with respect to the coupling equality constraint $Ax = b$. We define this *feasibility gap* with respect to $x^*(y,t)$ as follows:

$$
\mathcal{G}_{\mathrm{feas}}(y,t) := \|Ax^*(y,t) - b\|_y^*. \tag{3.17}
$$

Here, $x^*(y,t) \in \mathrm{int}(X)$. From (3.17), (3.11) and (3.13) and noting that $\lambda \le \beta$, we have:

$$
\mathcal{G}_{\mathrm{feas}}(y,t) = \|\nabla d(y,t)\|_y^* = t\lambda \le t\beta.
$$

Therefore, with $t \le \omega_*(\beta)^{-1}\varepsilon_d$ the feasibility gap reaches:

$$
\mathcal{G}_{\mathrm{feas}}(y,t) \le \beta\omega_*(\beta)^{-1}\varepsilon_d.
$$

4. Inexact perturbed path-following method for Lagrangian decomposition. This section presents an inexact perturbed path-following algorithm for approximately solving (2.2).

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

DEFINITION 4.1. *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 following 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 smooth dual version* of d_0. Next, we introduce two quantities:

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

Since $x^*(y,t) \in \mathrm{dom}(F) = \mathrm{int}(X)$, 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 3.5 and (4.1), we can consider these quantities as an approximate gradient vector and Hessian matrix of $d(\cdot,t)$, respectively.

Let

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

and $\bar{\lambda}$ 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 3.7 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

$$d(y^k,t_k) \uparrow d_0^* = \phi^* \text{ and } \mathcal{G}_{\mathrm{feas}}(y^k,t_k) \to 0, \text{ as } t_k \downarrow 0^+.$$

The aim of the algorithm is to generate $\{(y^k,t_k)\}_{k\geq 0}$ such that $\lambda_k \leq \beta < 1$. 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 and t such that t tends to zero. 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 $y^{0,0} \in Y$. Set $j = 0$.

`For` $j = 0, 1, \ldots$ `perform`

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 the primal subproblems (3.2) *in parallel* to obtain an approximation 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 (4.3).
4. Perform an inexact-perturbed damped Newton iteration: $y^{0,j+1} := y^{0,j} - \lambda_j(1+\lambda_j)^{-1}\nabla^2 d_{\bar\delta}(y^{0,j},t_0)^{-1}\nabla d_{\bar\delta}(y^{0,j},t_0)$.

`End For`

Phase 2. *Path-following iterations*

Compute $\sigma \in (0,1)$. Set $k := 0$.

`For` $k = 0,1,\ldots$ `perform:`

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

`End For`

Output. A $2\varepsilon_d$-approximate solution y^k of (2.2).

This algorithm is still conceptual. In the following subsections, we shall extract each step of this algorithmic framework in detail.

Let us emphasize an important point. In order to compute $d(y,t)$ we have to solve exactly the maximization problem in (3.2) or equivalently, to solve the system of nonlinear equations (3.9). This requirement is impractical. In practice, we can only solve this problem up to a desired accuracy $\bar\delta > 0$. Therefore, the theory of the path-following algorithm presented in [10, 12, 15, 22, 29] for solving (3.4) may no longer be satisfied. Here, we propose an inexact perturbed path-following algorithm for solving (3.4). This algorithm allows us to solve inexactly the primal subproblem (3.2). Consequently, inexact-perturbed Newton-type iterations are performed, which means that not only inexact gradient but also inexact Hessian of $d(\cdot,t)$ are used.

4.3. Computing inexact solution $\bar x_{\bar\delta}$. Note that condition (4.1) can not be used in practice to compute $\bar x_{\bar\delta}$ since $x^*(y,t)$ is unknown. We show how to compute $\bar x_{\bar\delta}$ such that (4.1) holds based on the optimality condition (3.9).

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

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

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

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

Since F is self-concordant, by applying [17, Theorems 4.1.7 and 4.1.8], and the definition of $\delta(\bar x_{\bar\delta}, x^*)$, the above equality implies that

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

Here, the last inequality holds if $\delta(\bar x_{\bar\delta}, x^*) < 1$.

Next, using again the optimality condition (3.9) we have

$$\begin{aligned} E^c_{\bar\delta} := \|c + A^Ty - t\nabla F(\bar x_{\bar\delta})\|^*_{x^c} &\overset{(3.9)}{=} t\|\nabla F(\bar x_{\bar\delta}) - \nabla F(x^*)\|^*_{x^c} \\ &\ge \frac{t}{\nu + 2\sqrt{\nu}}\|\nabla F(\bar x_{\bar\delta}) - \nabla F(x^*)\|^*_{x^*}, \end{aligned}$$

where the last inequality follows from [17, Corollary 4.2.1]. Combining this inequality and [17, Theorem 4.1.7], we obtain

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

Hence, we get

$$\delta(\bar{x}_{\bar{\delta}}, x^*) \leq \frac{(\nu+2\sqrt{\nu})E_{\bar{\delta}}^c}{t-(\nu+2\sqrt{\nu})E_{\bar{\delta}}^c}, \tag{4.8}$$

provided that $t > (\nu+2\sqrt{\nu})E_{\bar{\delta}}^c$. Let us define an accuracy ε_p for the primal subproblem (3.2) as

$$\varepsilon_p := \frac{\bar{\delta}t}{(\nu+2\sqrt{\nu})(1+\bar{\delta})} \geq 0. \tag{4.9}$$

Then it follows from (4.8) that if

$$E_{\bar{\delta}}^c = \|c + A^T y - t\nabla F(\bar{x}_{\bar{\delta}})\|_{x^c}^* \leq \frac{\bar{\delta}t}{(\nu+2\sqrt{\nu})(1+\bar{\delta})} \tag{4.10}$$

then $\bar{x}_{\bar{\delta}}(y,t)$ satisfies (4.1).

It remains to consider the distance from d_δ to d_0^* when t is sufficiently small. Suppose that $t \leq \omega_*(\beta)^{-1}\varepsilon_d$. Then, by combining (3.16) and (4.7) we have

$$|d_{\bar{\delta}}(y,t) - \phi^*| = |d_{\bar{\delta}}(y,t) - d_0^*| \leq 2\left[1+\omega_*(\beta)^{-1}\omega_*(\bar{\delta})\right]\varepsilon_d, \tag{4.11}$$

provided that $\bar{\delta} < 1$.

Remark 2. *Since* $E_{\bar{\delta}} := \|c+A^Ty-t\nabla F(\bar{x}_{\bar{\delta}})\|_{\bar{x}_{\bar{\delta}}}^* \geq (1-\bar{\delta})\|c+A^Ty-t\nabla F(\bar{x}_{\bar{\delta}})\|_{x^*}^*$. *By the same argument as before, we can show that if* $E_{\bar{\delta}} \leq \hat{\varepsilon}_p$, *where* $\hat{\varepsilon}_p := \frac{\bar{\delta}(1-\bar{\delta})t}{1+\bar{\delta}}$ *then* (4.1) *holds. This rule can be used to terminate the algorithms presented in the next sections.*

4.4. Phase 2 - The path-following scheme with inexact-perturbed full-step Newton iterations. Now, we analyze Steps 2-5 in Phase 2 of the algorithmic framework. In the path-following fashion, we only perform one inexact-perturbed full-step Newton (IPFNT) iteration for each value of parameter t. In other words, the IPFNT iteration and the update of t are simultaneously carried out. The parameter t is decreased by $t_+ := t - \Delta t$, where $\Delta t > 0$. Hence, one step of the path-following method is performed as follows:

$$\begin{cases} t_+ := t - \Delta t,\\ y_+ := y - \nabla^2 d_{\bar{\delta}}(y,t_+)^{-1}\nabla d_{\bar{\delta}}(y,t_+). \end{cases} \tag{4.12}$$

Since the Newton method is invariant under linear transformations, by (4.2), the second line of (4.12) is equivalent to:

$$y_+ := y - \nabla^2 \tilde{d}_{\bar{\delta}}(y,t_+)^{-1}\nabla \tilde{d}_{\bar{\delta}}(y,t_+). \tag{4.13}$$

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

$$\begin{aligned}
&\bar\lambda_+ := \bar\lambda_{\tilde d_{\bar\delta}(\cdot,t_+)}(y_+), && \delta_+ := \delta(\bar x_{\bar\delta +}, x^*_+) = \|\bar x_{\bar\delta}(y_+,t_+) - x^*(y_+,t_+)\|_{x^*(y_+,t_+)},\\
&\bar\lambda_1 := \bar\lambda_{\tilde d_{\bar\delta}(\cdot,t_+)}(y), && \delta_1 := \delta(\bar x_{\bar\delta 1}, x^*_1) = \|\bar x_{\bar\delta}(y,t_+) - x^*(y,t_+)\|_{x^*(y,t_+)},\\
&\bar\lambda := \bar\lambda_{\tilde d_{\bar\delta}(\cdot,t)}(y), && \delta := \delta(\bar x_{\bar\delta}, x^*) = \|\bar x_{\bar\delta}(y,t) - x^*(y,t)\|_{x^*(y,t)},
\end{aligned}$$

and by

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

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

4.4.1. The main estimate. Using the above notation, we provide a main estimate which will be used to analyze the convergence of the algorithm presented in Subsection 4.4.4. The proof of this result is postponed to Subsection 4.6.

LEMMA 4.2. *Let $y \in Y$ be given and $t > 0$. Let (y_+, t_+) be a pair generated by* (4.12). *Suppose that $\delta_1 + 2\Delta + \bar\lambda < 1$, $\delta_+ < 1$ and $\xi := \frac{\Delta+\bar\lambda}{1-\delta_1-2\Delta-\bar\lambda}$. Then*

$$\bar\lambda_+ \le \frac{1}{(1-\delta_+)}\left\{\delta_+ + \delta_1 + \xi^2 + \delta_1\left[(1-\delta_1)^{-2} + 2(1-\delta_1)^{-1}\right]\xi\right\}. \tag{4.14}$$

Moreover, the right-hand side of (4.14) *is nondecreasing with respect to all variables δ_+, δ_1, Δ and $\bar\lambda$.*

In particular, if we set $\delta_+ = 0$ and $\delta_1 = 0$, i.e. (3.2) *is solved exactly, then $\bar\lambda_+ = \lambda_+$, $\bar\lambda = \lambda$ and* (4.14) *reduces to*

$$\lambda_+ \le \left(\frac{\lambda + \Delta^*}{1 - 2\Delta^* - \lambda}\right)^2, \tag{4.15}$$

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

4.4.2. Finding the maximum centering parameter β^*. The key point of the path-following algorithm is to determine the maximum value of $\beta \in (0, \beta^*) \subseteq (0,1)$ and appropriate values of $\bar\delta$ and Δ such that if $\bar\lambda \le \beta$, then $\bar\lambda_+ \le \beta$. We analyze the estimate (4.14) to find these parameters.

First, let $\beta \in (0,1)$ such that $\bar\lambda \le \beta$. Since the right-hand side of (4.14) is nondecreasing with respect to all variables, if we define

$$\varphi_{\bar\delta}(\bar\xi) := \frac{1}{1-\bar\delta}\left\{2\bar\delta + \bar\xi^2 + \bar\delta[(1-\bar\delta)^{-2} + 2(1-\bar\delta)^{-1}]\bar\xi\right\},$$

and $\bar\xi := \frac{\Delta+\beta}{1-\bar\delta-\beta-2\Delta}$, then $\bar\lambda_+ \le \beta$ if $\varphi_{\bar\delta}(\bar\xi) \le \beta$. This condition leads to $0 \le \bar\xi \le \frac{\sqrt{p^2+4q}-p}{2}$ and $0 \le \bar\delta \le \frac{\beta}{\beta+2}$, where $p := \bar\delta[(1-\bar\delta)^{-2} + 2(1-\bar\delta)^{-1}]$ and $q := (1-\bar\delta)\beta - 2\bar\delta$.

Now, let $\theta := \frac{\sqrt{p^2+4q}-p}{2} > 0$. Since $\bar\xi = \frac{\beta+\Delta}{1-\bar\delta-\beta-2\Delta} \le \theta$, we have $(1+2\theta)\Delta \le \theta(1-\bar\delta-\beta) - \beta$. Thus, in order to ensure $\Delta > 0$, we require that $\theta = \frac{\sqrt{p^2+4q}-p}{2} > \frac{\beta}{1-\bar\delta-\beta}$. This condition leads to

$$\mathcal{P}(\beta) := c_0 + c_1\beta + c_2\beta^2 + c_3\beta^3 > 0, \tag{4.16}$$

where $c_0 := -2\bar{\delta}(1-\bar{\delta})^2 \leq 0$, $c_1 := (1-\bar{\delta})[(1+\bar{\delta})^2-p] \geq 0$, $c_2 := p-3-2\bar{\delta}^2+2\bar{\delta} \leq 0$ and $c_3 := 1-\bar{\delta} > 0$. By well-known characteristics of the cubic polynomial, we know that $\mathcal{P}(\beta)$ has three real roots if $18c_0c_1c_2c_3 - 4c_2^3c_0 + c_2^2c_1^2 - 4c_3c_1^3 - 27c_3^2c_0^2 \geq 0$. By numerical solution, the last condition leads to $0 \leq \bar{\delta} \leq \bar{\delta}_{\max}$, where $\bar{\delta}_{\max} \approx 0.0432863855$.

Finally, we summarize the above analysis into the following theorem.

THEOREM 4.3. *Let* $\bar{\delta}_{\max} = 0.0432863855$ *and* $0 \leq \bar{\delta} \leq \bar{\delta}_{\max}$. *Then* $\mathcal{P}$ *defined by* (4.16) *has three nonnegative real roots* $0 \leq \beta_* < \beta^* < \beta_3$. *Suppose that* $\beta \in (\beta_*, \beta^*)$ *and* $\bar{\Delta} := \frac{\theta(1-\bar{\delta}-\beta)-\beta}{1+2\theta} > 0$ *where* $\theta := \frac{\sqrt{p^2+4q}-p}{2}$, *and* p *and* q *are defined as above. Then, for* $0 \leq \delta_+ \leq \bar{\delta}$, $0 \leq \delta_1 \leq \bar{\delta}$ *and* $0 \leq \Delta \leq \bar{\Delta}$, *if* $\bar{\lambda} \leq \beta$ *then* $\bar{\lambda}_+ \leq \beta$.

Proof. Note that the cubic polynomial $\mathcal{P}(\beta)$ has three real roots if $18c_0c_1c_2c_3 - 4c_2^3c_0 + c_2^2c_1^2 - 4c_3c_1^3 - 27c_3^2c_0^2 \geq 0$. Numerically, this condition leads to $0 \leq \bar{\delta} \leq \bar{\delta}_{\max} = 0.0432863855$. Moreover, one can show that three roots $\beta_* < \beta^* < 1 < \beta_3$ of $\mathcal{P}$ are nonnegative and $\mathcal{P}(\beta) > 0$ if $\beta \in (\beta_*, \beta^*)$. However, $\mathcal{P}(\beta) > 0$ implies $\theta(1-\bar{\delta}-\beta)-\beta > 0$, where $\theta := \frac{\sqrt{p^2+4q}-p}{2}$. Thus, from the definition of $\bar{\xi}$, we have $0 \leq \Delta \leq \bar{\Delta} := \frac{\theta(1-\bar{\delta}-\beta)-\beta}{1+2\theta} > 0$. □

In order to see the values of β_*, β^* and $\bar{\Delta}$ varying with respect to the accuracy $\bar{\delta}$, we illustrate them in Figure 4.1, where the left-hand side shows the values of β_* (solid) and β^* (dash) and the right-hand side shows the value of $\bar{\Delta}$ varying with respect to $\bar{\delta}$ when β is chosen by $\beta := \frac{\beta_*+\beta^*}{2}$ (dash) and $\beta := \frac{\beta^*}{4}$ (solid), respectively.

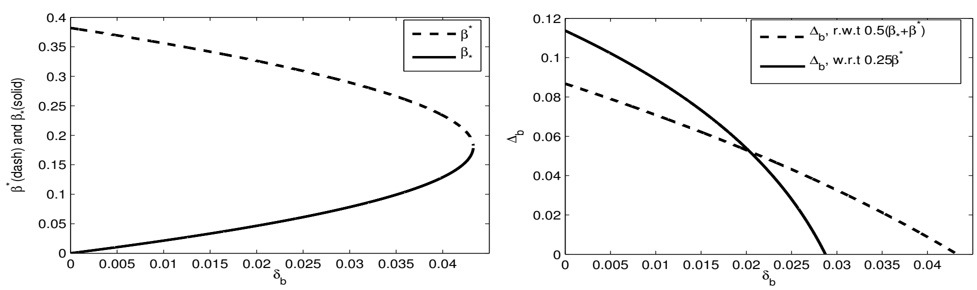

FIG. 4.1. *The values of* β_*, β^* *and* $\bar{\Delta}$ *varying w.r.t* $\bar{\delta}$.

4.4.3. The update rule for the barrier parameter t. It remains to quantify the decrement Δt of the barrier parameter t. From (3.9) we have

$$c + A^T y - t\nabla F(x^*) = 0 \quad \text{and} \quad c + A^T y - t_+\nabla F(x_1^*) = 0,$$

where $x^* := x^*(y,t)$ and $x_1^* := x^*(y,t_+)$ are defined as before. 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 [17, Theorem 4.1.7] and $\|\nabla F(x^*)\|_{x^*}^* \leq \sqrt{\nu}$ (see [17, inequality 4.2.4]), we have

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

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

$$\Delta^* \leq \bar{\Delta}^* := \frac{\sqrt{\nu}\Delta t}{t-(\sqrt{\nu}+1)\Delta t}. \tag{4.17}$$

Note that (4.17) implies

$$\Delta t = \frac{\bar{\Delta}^* t}{\sqrt{\nu} + (\sqrt{\nu}+1)\bar{\Delta}^*}. \tag{4.18}$$

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

$$\begin{aligned}
\Delta := \|\bar{x}_{\bar{\delta}1} - \bar{x}_{\bar{\delta}}\|_{\bar{x}_{\bar{\delta}}} &\overset{(4.41)}{\leq} \frac{1}{(1-\delta)}\left[\|\bar{x}_{\bar{\delta}1} - x_1^*\|_{x^*} + \|x_1^* - x^*\|_{x^*} + \|x^* - \bar{x}_{\bar{\delta}}\|_{x^*}\right]\\
&\leq \frac{1}{(1-\delta)}\left[\frac{\delta_1}{1-\Delta^*} + \Delta^* + \delta\right]\\
&\overset{(4.17)}{\leq} \frac{1}{(1-\delta)}\left[\frac{\delta_1}{1-\bar{\Delta}^*} + \bar{\Delta}^* + \delta\right]\\
&\overset{\delta,\delta_1\leq\bar{\delta}}{\leq} \frac{1}{(1-\bar{\delta})}\left[\frac{\bar{\delta}}{1-\bar{\Delta}^*} + \bar{\Delta}^* + \bar{\delta}\right].
\end{aligned} \tag{4.19}$$

Now, we need to find a condition such that $\Delta \leq \bar{\Delta}$, where $\bar{\Delta}$ is given in Theorem 4.3. This condition holds if $\frac{\bar{\delta}}{1-\Delta^*} + \Delta^* \leq (1-\bar{\delta})\bar{\Delta} - \bar{\delta}$ due to (4.19). Since $\Delta^* \leq \bar{\Delta}^*$ due to (4.17), we impose a more relaxed condition

$$0 \leq \bar{\Delta}^* \leq \frac{1}{2}\left[(1-\bar{\delta})\bar{\Delta} - \bar{\delta} + 1 - \sqrt{((1-\bar{\delta})\bar{\Delta} - \bar{\delta} - 1)^2 + 4\bar{\delta}}\right], \tag{4.20}$$

provided that $\bar{\delta} \leq \frac{\bar{\Delta}}{1+\bar{\Delta}}$. Thus, we can fix $\bar{\Delta}^*$ at

$$\bar{\Delta}^* = \frac{1}{2}\left[(1-\bar{\delta})\bar{\Delta} - \bar{\delta} + 1 - \sqrt{((1-\bar{\delta})\bar{\Delta} - \bar{\delta} - 1)^2 + 4\bar{\delta}}\right]. \tag{4.21}$$

The update rule for the barrier parameter t becomes

$$t_+ := (1-\sigma)t = \left(1 - \frac{\bar{\Delta}^*}{\sqrt{\nu} + (\sqrt{\nu}+1)\bar{\Delta}^*}\right) = \frac{\sqrt{\nu}(\bar{\Delta}^*+1)t}{\sqrt{\nu}(\bar{\Delta}^*+1) + \bar{\Delta}^*},$$

where $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu}+\bar{\Delta}^*(\sqrt{\nu}+1)} \in (0,1)$.

Finally, we show that the conditions given in Theorem 4.3, (4.20) and (4.21) 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

$$\bar{\Delta} \approx 0.089012, \text{ and } \bar{\Delta}^* \approx 0.067399.$$

4.4.4. The algorithm and its convergence. Now, we are at the point to present the algorithm and its convergence. Before presenting the algorithm, we need to find a stopping criterion for it. By using Lemma 4.6c), we have:

$$\lambda \leq (1-\delta)^{-1}(\bar{\lambda} + \delta), \tag{4.22}$$

provided that $\delta < 1$ and $\bar{\lambda} \leq \beta < 1$. Consequently, if $\bar{\lambda} \leq (1-\bar{\delta})^{-1}(\beta + \bar{\delta})$ then $\lambda \leq \beta$. Let us define $\vartheta := (1-\bar{\delta})^{-1}(\beta + \bar{\delta})$. It follows from Lemma 3.7 that if $t\omega_*(\vartheta) \leq \varepsilon_d$ for a given tolerance $\varepsilon_d > 0$, then y is a $2\varepsilon_d$-solution of (2.2).

The algorithmic framework presented in Subsection 4.2 is now described in detail as follows.

Algorithm 1. (*Path-following algorithm with* IPFNT *iterations*)

Initialization: Perform the following steps:

1. Choose $\bar{\delta} \in [0, \bar{\delta}_{\max}]$ and compute β_* and β^* as the first and second roots of $\mathcal{P}$ defined by (4.16), respectively.
2. Fix some $\beta \in (\beta_*, \beta^*)$ (e.g. $\beta = \frac{1}{4}\beta^*$).
3. Choose an initial value $t = t_0 > 0$.

Phase 1. Apply Algorithm 2 presented in Subsection 4.5 to find $y^0 \in Y$ such that $\lambda_{\tilde{d}_{\bar{\delta}}(\cdot, t_0)}(y^0) \le \beta$.

Phase 2.

Initialization: Perform the following steps:

1. Given a tolerance $\varepsilon_d > 0$.
2. Compute $\bar{\Delta}$ as in Theorem 4.3. Then, compute $\bar{\Delta}^*$ by (4.21).
3. Compute the factor $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu} + (\sqrt{\nu}+1)\bar{\Delta}^*}$.
4. Compute the accuracy factor $\gamma := \frac{\bar{\delta}}{(\nu + 2\sqrt{\nu})(1+\bar{\delta})}$.

Iteration: Perform the following loop.

`For` $k = 0, 1, \cdots$ `do`

1. If $t_k \le \frac{\varepsilon_d}{\omega_*(\vartheta)}$, where $\vartheta := (1-\bar{\delta})^{-1}(\beta + \bar{\delta})$, then terminate.
2. Compute an accuracy for the primal subproblem $\varepsilon_k := \gamma t_k$.
3. Update $t_{k+1} := (1-\sigma)t_k$.
4. Solve approximately (2.3) *in parallel* up to the given tolerance 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 $\nabla^2 d_{\bar{\delta}}(y^k, t_{k+1})$ according to (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 of For`.

The core step of Phase 2 in Algorithm 1 is Step 4, where we need to solve two convex optimization problems to compute the gradient vector and the Hessian matrix of $d_{\bar{\delta}}(\cdot, t_{k+1})$ at Step 5. These quantities require an approximate solution $\bar{x}_{\bar{\delta}}(y^k, t_{k+1})$, the gradient vector $\nabla F(\bar{x}_{\bar{\delta}}(y^k, t_{k+1}))$ and the Hessian matrix $\nabla^2 F(\bar{x}_{\bar{\delta}}(y^k, t_{k+1}))$, which can also be computed *in parallel*. Note that Step 4 actually requires to solve a system of nonlinear equations (3.9) (see Section 6 for more details). The update rule of t at Step 3 can be done in an adaptive way, where we can use $\|\nabla F(\bar{x}_{\bar{\delta}})\|^*_{\bar{x}_{\bar{\delta}}}$ instead of its upper bound $\sqrt{\nu}$. For example, we can use $\Delta t := \frac{\bar{\Delta}^* t}{R_{\bar{\delta}} + (R_{\bar{\delta}}+1)\bar{\Delta}^*}$ instead of (4.18), where $R_{\bar{\delta}} := (1-\bar{\delta})^{-1}\left[\bar{\delta}(1-\bar{\delta})^{-1} + \|\nabla F(\bar{x}_{\bar{\delta}})\|^*_{\bar{x}_{\bar{\delta}}}\right]$. The stopping criterion at Step 1 can be replaced by $\omega_*(\vartheta_k)t_k \le \varepsilon_d$, where $\vartheta_k := (1-\bar{\delta})^{-1}[\lambda_{\tilde{d}_{\bar{\delta}}(\cdot, t_k)}(y^k) + \bar{\delta}]$ due to Lemma 3.7.

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 4.4. *Let $\{(y^k, t_k)\}$ be a sequence generated by Algorithm 1. Then the number of iterations $k_{\max}$ to obtain a $2\varepsilon_d$-solution of* (2.2) *does not exceed*

$$k_{\max} := \left\lfloor \frac{\ln\left(\frac{\varepsilon_d}{t_0 \omega_*(\vartheta)}\right)}{\ln(1-\sigma)} \right\rfloor + 1, \tag{4.23}$$

where $\sigma = \frac{\bar{\Delta}^*}{\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}^*} \in (0,1)$ *and* $\vartheta = (1-\bar{\delta})\beta - \bar{\delta} \in (0,1)$.

Proof. Note that y^k is a $2\varepsilon_d$-solution of (2.2) if $t_k \le \frac{\varepsilon_d}{\omega_*(\vartheta)}$ due to Lemma 3.7, where $\vartheta = (1-\bar{\delta})\beta - \bar{\delta}$. Since $t_k = (1-\sigma)^k t_0$ due to Step 3, we require $(1-\sigma)^k \le \frac{\varepsilon_d}{t_0\omega_*(\vartheta)}$. Consequently, we obtain (4.23). □

REMARK 3 (**The worst-case complexity**). *Since* $(1-\sigma) = \left[1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}\right]^{-1}$ *which implies that* $-\ln(1-\sigma) \sim \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$. *It follows from Theorem 4.4 that the complexity of Algorithm 1 is* $O(\sqrt{\nu}\ln\frac{t_0}{\varepsilon_d})$.

REMARK 4 (**Linear convergence**). *The rate of convergence of the sequence* $\{t_k\}$ *is linear and the contraction rate is no greater than* $1-\sigma$. *Note that if* $\lambda_{\tilde{d}_{\bar{\delta}}(\cdot,t)}(y) \le \beta$, *then it follows from* (3.11) *that* $\lambda_{d_{\bar{\delta}}(\cdot,t)}(y) \le \beta\sqrt{t}$. *Therefore, the sequence of Newton decrements* $\{\lambda_{d(\cdot,t_k)}(y^k)\}_k$ *of* d *also converges linearly to zero with the contraction factor less than or equal to* $\sqrt{1-\sigma}$.

REMARK 5 (**Recovering the feasibility**). *Since* $\nabla d_{\bar{\delta}}(y,t) = A\bar{x}_{\bar{\delta}}(y,t) - b = t\nabla\tilde{d}_{\bar{\delta}}(y,t)$, *we have* $\|A\bar{x}_{\bar{\delta}}(y,t) - b\|_y^* = t\|\nabla\tilde{d}_{\bar{\delta}}(y,t)\| = t\bar{\lambda} \le t\beta$. *If we define the inexact feasibility gap at* $\bar{x}_{\bar{\delta}}(y,t)$ *as*

$$\bar{\mathcal{G}}_{\rm feas}(y,t) := \|A\bar{x}_{\bar{\delta}}(y,t) - b\|_y^*,$$

then $\bar{\mathcal{G}}_{\rm feas}(y,t) \le t\beta$, *which shows that* $\bar{\mathcal{G}}_{\rm feas}(y,t)$ *converges linearly to zero with the same rate as* t.

REMARK 6 (**The inexactness in the IPFNT direction** (4.12)). *Note that we can apply an inexact method to solve the linear system* (4.12). *Under appropriate assumptions of the inexact term, we can still prove the convergence of the algorithm. For more detail on inexact Newton methods, one can refer to [24].*

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 consider an inexact perturbed damped Newton (IPDNT) method for finding such a point y^0.

4.5.1. Inexact perturbed damped Newton iteration. Let us fix $t = t_0 > 0$ and choose an accuracy $\bar{\delta} \ge 0$. We assume that the current iterate $y \in Y$ is given, and we compute the next iterate y_+ by applying the IPDNT iteration to $d_{\bar{\delta}}(\cdot,t_0)$ as

$$y_+ := y - \alpha(y)\nabla^2 d_{\bar{\delta}}(y,t_0)^{-1}\nabla d_{\bar{\delta}}(y,t_0), \tag{4.24}$$

where $\alpha := \alpha(y) > 0$ is the step size which will be chosen appropriately. Note that since (4.24) is invariant under linear transformation, it is equivalent to

$$y_+ := y - \alpha(y)\nabla^2\tilde{d}_{\bar{\delta}}(y,t_0)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y,t_0), \tag{4.25}$$

It follows from (3.11) that $\tilde{d}(\cdot,t_0)$ is standard self-concordant, by [17, Theorem 4.1.8], we have

$$\tilde{d}(y_+,t_0) \le \tilde{d}(y,t_0) + \nabla\tilde{d}(y,t_0)^T(y_+ - y) + \omega_*(\|y_+ - y\|_y), \tag{4.26}$$

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

$$0 \le \omega(\delta(\bar{x}_{\bar{\delta}},x^*)) \le \tilde{d}(y,t_0) - \tilde{d}_{\bar{\delta}}(y,t_0) \le \omega_*(\delta(\bar{x}_{\bar{\delta}},x^*)), \tag{4.27}$$

which is an approximation between $\tilde{d}(\cdot, t_0)$ and $\tilde{d}_{\bar{\delta}}(\cdot, t_0)$. In order to analyze the convergence of the IPDNT iteration (4.24) we denote by

$$\begin{aligned} \hat{\delta}_+ &:= \|\bar{x}_{\bar{\delta}}(y_+, t_0) - x^*(y_+, t_0)\|_{x^*(y_+, t_0)}, \\ \hat{\delta} &:= \|\bar{x}_{\bar{\delta}}(y, t_0) - x^*(y, t_0)\|_{x^*(y, t_0)}, \\ \bar{\lambda}_0 &:= \lambda_{\tilde{d}_{\bar{\delta}}(\cdot, t_0)}(y) = \alpha(y)\|\!|y_+ - y|\!\|_y, \end{aligned} \tag{4.28}$$

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.

4.5.2. Finding the step size $\alpha(y)$**.** Now, we find an appropriate step size $\alpha(y) \in (0, 1]$ such that the sequence generated by (4.25) converges to y^0. Let $p := y_+ - y$. From (4.26) and (4.27), we have

$$\begin{aligned} \tilde{d}_{\bar{\delta}}(y_+, t_0) &\overset{(4.26)}{\leq} \tilde{d}(y_+, t_0) \overset{(4.27)}{\leq} \tilde{d}(y, t_0) + \nabla\tilde{d}(y, t_0)^T(y_+ - y) + \omega_*(\|y_+ - y\|_y) \\ &\overset{(4.26)}{\leq} \tilde{d}_{\bar{\delta}}(y, t_0) + \nabla\tilde{d}(y, t_0)^T(y_+ - y) + \omega_*(\|y_+ - y\|_y) + \omega_*(\hat{\delta}) \\ &= \tilde{d}_{\bar{\delta}}(y, t_0) + \nabla\tilde{d}_{\bar{\delta}}(y, t_0)^T p + \left[\nabla\tilde{d}(y, t_0) - \nabla\tilde{d}_{\bar{\delta}}(y, t_0)\right]^T p + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}) \\ &\overset{(4.24)}{\leq} \tilde{d}_{\bar{\delta}}(y, t_0) - \alpha\bar{\lambda}_0^2 + \|\nabla\tilde{d}(y, t_0) - \nabla\tilde{d}_{\bar{\delta}}(y, t_0)\|_y^* \|p\|_y + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}) \\ &\overset{(4.39)}{\leq} \tilde{d}_{\bar{\delta}}(y, t_0) - \alpha\bar{\lambda}_0^2 + \hat{\delta}\|p\|_y + \omega_*(\|p\|_y) + \omega_*(\hat{\delta}). \end{aligned} \tag{4.29}$$

Furthermore, from (4.42) and the definition of $\nabla^2\tilde{d}$ and $\nabla^2\tilde{d}_{\bar{\delta}}$, we have

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

This inequality implies that

$$(1 - \hat{\delta})\|\!|p|\!\|_y \leq \|p\|_y \leq (1 - \hat{\delta})^{-1}\|\!|p|\!\|_y.$$

Combining this inequality, (4.25) and the definition of $\bar{\lambda}_0$ in (4.28) we get:

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

Let us assume that $\alpha\bar{\lambda}_0 + \hat{\delta} < 1$. By substituting the right-hand side of this inequality into (4.29) and observing that the right hand side of (4.29) is nondecreasing with respect to $\|p\|_y$, we get

$$\tilde{d}_{\bar{\delta}}(y_+, t_0) \leq \tilde{d}_{\bar{\delta}}(y, t_0) - \alpha\bar{\lambda}_0^2 + \frac{\alpha\bar{\lambda}\hat{\delta}}{1 - \hat{\delta}} + \omega_*\left(\frac{\alpha\bar{\lambda}_0}{1 - \hat{\delta}}\right) + \omega_*(\hat{\delta}). \tag{4.30}$$

Now, let us simplify the last terms of (4.30) which we denote by T as follows:

$$\begin{aligned} T &:= -\alpha\bar{\lambda}_0^2 + \frac{\alpha\bar{\lambda}_0\hat{\delta}}{1 - \hat{\delta}} + \omega_*\left(\frac{\alpha\bar{\lambda}_0}{1 - \hat{\delta}}\right) + \omega_*(\hat{\delta}) \\ &= -\alpha\bar{\lambda}_0^2 + \frac{\alpha\bar{\lambda}_0\hat{\delta}}{1 - \hat{\delta}} - \frac{\alpha\bar{\lambda}_0}{1 - \hat{\delta}} - \ln\left(1 - \frac{\alpha\bar{\lambda}_0}{1 - \hat{\delta}}\right) - \hat{\delta} - \ln(1 - \hat{\delta}) \\ &= -\alpha\bar{\lambda}_0^2 - (\alpha\bar{\lambda}_0 + \hat{\delta}) - \ln\left[1 - (\alpha\bar{\lambda}_0 + \hat{\delta})\right] \\ &= -\alpha\bar{\lambda}_0^2 + \omega_*(\alpha\bar{\lambda}_0 + \hat{\delta}). \end{aligned} \tag{4.31}$$

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

$$\alpha = \frac{(1-\hat{\delta})\bar{\lambda}_0 - 2\hat{\delta} + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}}{2\bar{\lambda}_0(1+\bar{\lambda}_0)}, \tag{4.32}$$

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

$$\eta = \frac{\bar{\lambda}_0\left[(1-\hat{\delta})\bar{\lambda}_0 - 2\hat{\delta} + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}\right]}{(1+\hat{\delta})\bar{\lambda}_0 + \sqrt{(1-\hat{\delta})^2\bar{\lambda}_0^2 - 4\hat{\delta}\bar{\lambda}_0}}. \tag{4.33}$$

Note that if $\hat{\delta} = 0$, then $\alpha = \frac{1}{1+\bar{\lambda}_0}$ and $\eta = \bar{\lambda}_0$. The IPDNT iteration (4.24) becomes the exact damped Newton iteration as in [17].

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

$$0 < \bar{\hat{\delta}} < \hat{\delta}^* := \frac{2+\beta-2\sqrt{1+\beta}}{\beta} = \frac{\beta}{2+\beta+2\sqrt{1+\beta}}. \tag{4.34}$$

Next, we choose the step size α as

$$\alpha(y) := \frac{(1-\bar{\hat{\delta}})\bar{\lambda}_0 - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\bar{\lambda}_0^2 - 4\bar{\hat{\delta}}\bar{\lambda}_0}}{2\bar{\lambda}_0(1+\bar{\lambda}_0)} \in (0,1). \tag{4.35}$$

Then the IPDNT iteration (4.24) with $\alpha(y)$ given as (4.35) generates a new point y_+ such that

$$\tilde{d}_{\bar{\delta}}(y_+, t_0) \leq \tilde{d}_{\bar{\delta}}(y, t_0) - \omega(\underline{\eta}), \tag{4.36}$$

where

$$\underline{\eta} := \frac{\beta\left[(1-\bar{\hat{\delta}})\beta - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\beta^2 - 4\bar{\hat{\delta}}\beta}\right]}{(1+\bar{\hat{\delta}})\beta + \sqrt{(1-\bar{\hat{\delta}})^2\beta^2 - 4\bar{\hat{\delta}}\beta}} \in (0,1). \tag{4.37}$$

Finally, let us estimate the constant $\underline{\eta}$ for the case $\beta \approx 0.089009$. We first obtain $\hat{\delta}^* \approx 0.02131$. Let $\bar{\hat{\delta}} = \frac{1}{2}\hat{\delta}^* \approx 0.010657$. Then we get $\underline{\eta} \approx 0.0754963$. Consequently, $\omega(\underline{\eta}) \approx 0.003002$.

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 $\hat{\delta}^* := \frac{\beta}{2+\beta+2\sqrt{1+\beta}}$ and fixed $\bar{\hat{\delta}} \in (0, \hat{\delta}^*)$ (e.g. $\bar{\hat{\delta}} = 0.5\hat{\delta}^*$).

4. Compute an accuracy $\varepsilon_p := \frac{t_0\bar{\hat{\delta}}}{2(\nu+2\sqrt{\nu})(1+\bar{\hat{\delta}})}$.

Iteration: Perform the following loop.

`For` $j = 0, 1, \cdots$ `do`

1. Solve approximately the primal subproblem (3.2) *in parallel* up to the accuracy ε_p 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 $\bar{\lambda}_j \leq \beta$ 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 $\alpha_j \in (0, 1]$ is computed by

$$\alpha_j := \frac{(1-\bar{\hat{\delta}})\bar{\lambda}_j - 2\bar{\hat{\delta}} + \sqrt{(1-\bar{\hat{\delta}})^2\bar{\lambda}_j^2 - 4\bar{\hat{\delta}}\bar{\lambda}_j}}{2\bar{\lambda}_j(1+\bar{\lambda}_j)}.$$

`End of For`.

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

THEOREM 4.5. *The number of iterations to terminate Algorithm 2 does not exceed*

$$J_{\max} := \left\lfloor \frac{d_{\bar{\delta}}(y^{0,0}, t_0) - d^*(t_0) + \omega_*(\bar{\hat{\delta}})}{t_0\omega(\underline{\eta})} \right\rfloor + 1, \tag{4.38}$$

where $d^*(t_0) = \min_{y\in Y} d(y, t_0)$ *and* $\underline{\eta}$ *is given by* (4.37).

Proof. Summing up (4.36) from $j = 0$ to $j = k$ and then using (4.27) we have

$$\begin{aligned} 0 &\leq \tilde{d}(y^{0,k}, t_0) - \tilde{d}^*(t_0) \leq \tilde{d}_{\bar{\delta}}(y^{0,k}, t_0) + \omega_*(\bar{\hat{\delta}}) - \tilde{d}^*(t_0) \\ &\leq \tilde{d}_{\bar{\delta}}(y^{0,0}, t_0) + \omega_*(\bar{\hat{\delta}}) - \tilde{d}^*(t_0) - k\omega(\underline{\eta}). \end{aligned}$$

This inequality together with (3.11) and (4.4) imply

$$k \leq \frac{d_{\bar{\delta}}(y^{0,0}, t_0) - d^*(t_0) + \omega_*(\bar{\hat{\delta}})}{t_0\omega(\underline{\eta})}.$$

Hence, the maximum number of iterations in Algorithm 2 does not exceed $J_{\max}$ defined by (4.38). □

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

4.6. The proof of Lemma 4.2. First, we prove the following lemma which will be used to prove the main inequality in Lemma 4.2.

LEMMA 4.6. *Suppose that Assumptions* A.1 *and* A.2 *are satisfied. Then*

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

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

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

b) *Moreover, one has*

$$\|\nabla\tilde{d}_{\bar{\delta}}(y, t) - \nabla\tilde{d}(y, t)\|_y^* \leq \|\bar{x}_{\bar{\delta}} - x^*\|_{x^*}. \tag{4.39}$$

c) *If* $\Delta < 1$ *then*

$$\bar{\lambda}_1 \le \frac{\Delta + \bar{\lambda}}{1 - \Delta}. \tag{4.40}$$

Proof. Since F is standard self-concordant, for any $z \in W^0(x,1)$, it follows from [17, Theorem 4.1.6] that

$$(1 - \|z - x\|_x)^2 \nabla^2 F(x) \preceq \nabla^2 F(z) \preceq \frac{1}{(1 - \|z - x\|_x)^2} \nabla^2 F(x). \tag{4.41}$$

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

$$\begin{aligned}(1 - \|z - x\|_x)^2 A\nabla^2 F(x)^{-1}A^T &\preceq A\nabla^2 F(z)^{-1}A^T\\ &\preceq (1 - \|z - x\|_x)^{-2} A\nabla^2 F(x)^{-1}A^T.\end{aligned} \tag{4.42}$$

Using again [1, Proposition 8.6.6] for (4.42) we get

$$\begin{aligned}&(1 - \|z - x\|_x)^2 A^T[A\nabla^2 F(x)^{-1}A^T]^{-1}A \preceq A^T[A\nabla^2 F(z)^{-1}A^T]^{-1}A\\ &\preceq (1 - \|z - x\|_x)^{-2} A^T[A\nabla^2 F(x)^{-1}A^T]^{-1}A.\end{aligned} \tag{4.43}$$

Now, using (3.10) and (3.11), we have $\nabla^2\tilde{d}(y,t) = \frac{1}{t^2}A\nabla^2 F(x^*)^{-1}A^T$. Alternatively, using (4.3) and (4.4), we get $\nabla^2\tilde{d}_{\bar{\delta}}(y,t) = \frac{1}{t^2}A\nabla^2 F(\bar{x}_{\bar{\delta}})^{-1}A^T$. Substituting these relations with $x = x^*_+$ and $z = \bar{x}_{\bar{\delta}+}$ into (4.42) and noting that $\delta_+ = \delta(\bar{x}_+, x^*_+)$ defined by (4.6), we obtain (4.39).

Next, we prove b). For any $x \in \mathrm{dom}(F)$, the Hessian matrix $\nabla^2 F(x)$ is symmetric positive definite. Let us define

$$M(x) := \begin{bmatrix} \nabla^2 F(x) & A^T \\ A & A\nabla^2 F(x)^{-1}A^T \end{bmatrix}.$$

First, we show that $M(x)$ is positive definite. Indeed, for any $z = (u, v) \in \mathbb{R}^n \times \mathbb{R}^m$, we have

$$\begin{aligned} z^T M(x) z &= u^T\nabla^2 F(x)u + u^T A^T v + v^T A u + v^T A\nabla^2 F(x)^{-1}A^T v\\ &= \|\nabla^2 F(x)^{1/2}u\|^2 + 2(\nabla^2 F(x)^{1/2}u)^T(\nabla^2 F(x)^{-1/2}A^T v) + \|\nabla^2 F(x)^{-1/2}A^T v\|^2\\ &= \|\nabla^2 F(x)^{1/2}u + \nabla^2 F(x)^{-1/2}A^T v\|^2 \ge 0,\end{aligned}$$

which shows that $M(x) \succeq 0$. Now, since A has full-row rank, $A\nabla^2 F(x)^{-1}A^T$ is also symmetric positive definite. By applying Schur's complement to $M(x)$ [1], we obtain

$$A^T[A\nabla^2 F(x)^{-1}A^T]^{-1}A \preceq \nabla^2 F(x). \tag{4.44}$$

To prove (4.39) 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

$$\begin{aligned}\left[\|\nabla\tilde{d}_{\bar{\delta}}(y,t) - \nabla\tilde{d}(y,t)\|^*_y\right]^2 &= \frac{1}{t^2}(\bar{x}_{\bar{\delta}} - x^*)^T A^T\nabla^2\tilde{d}(y,t)^{-1}A(\bar{x}_{\bar{\delta}} - x^*)\\ &\overset{(3.10),(3.11)}{=} (\bar{x}_{\bar{\delta}} - x^*)^T A^T[A\nabla^2 F(x^*)^{-1}A^T]^{-1}A(\bar{x}_{\bar{\delta}} - x^*)\\ &\overset{(4.44)}{\le} (\bar{x}_{\bar{\delta}} - x^*)^T\nabla^2 F(x^*)(\bar{x}_{\bar{\delta}} - x^*)\\ &= \|\bar{x}_{\bar{\delta}} - x^*\|^2_{x^*},\end{aligned}$$

which is indeed (4.39).

Finally, we prove (4.40). By using the definitions of $\nabla\tilde{d}_{\bar{\delta}}(\cdot,t_+)$ and $\nabla^2\tilde{d}_{\bar{\delta}}(\cdot,t_+)$ in (4.3), of $\tilde{d}_{\bar{\delta}}(\cdot,t_+)$ in (4.4), for any feasible point $\hat{x}$ of (2.1), it follows from the definition of $\bar{\lambda}_1$ in (4.5) and $A\hat{x}=b$ that

$$
\begin{aligned}
\bar{\lambda}_1^2 &= \left[\|\nabla\tilde{d}_{\bar{\delta}}(y,t_+)\|_y^*\right]^2\\
&\overset{(4.5)}{=} \nabla\tilde{d}_{\bar{\delta}}(y,t_+)\nabla^2\tilde{d}_{\bar{\delta}}(y,t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y,t_+)\\
&\overset{(4.4)}{=} \frac{1}{t_+}\nabla d_{\bar{\delta}}(y,t_+)\nabla^2 d_{\bar{\delta}}(y,t_+)^{-1}\nabla d_{\bar{\delta}}(y,t_+)\\
&\overset{(4.3)}{=} (\bar{x}_{\bar{\delta}1}-\hat{x})^TA^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}1})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}1}-\hat{x}).
\end{aligned}
\tag{4.45}
$$

Since $\Delta = \|\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}\|_{\bar{x}_{\bar{\delta}}} < 1$ by assumption, it implies that $\bar{x}_{\bar{\delta}1}\in W^0(\bar{x}_{\bar{\delta}},1)$. By applying the right-hand side of (4.43) with $x=\bar{x}_{\bar{\delta}}$ and $z=\bar{x}_{\bar{\delta}1}$, we get: that

$$
\bar{\lambda}_1^2 \le \frac{1}{(1-\Delta)^2}(\bar{x}_{\bar{\delta}1}-\hat{x})^TA^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}1}-\hat{x}). \tag{4.46}
$$

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

$$
(u+v)^TQ(u+v) \le \left[\sqrt{u^TQu}+\sqrt{v^TQv}\right]^2. \tag{4.47}
$$

Since $H_{\bar{\delta}} := A^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A$ is symmetric positive semidefinite, by applying (4.47) 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

$$
\bar{\lambda}_1^2 \le \frac{1}{(1-\Delta)^2}\left\{\left[(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}})\right]^{1/2}+\left[(\bar{x}_{\bar{\delta}}-\hat{x})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}}-\hat{x})\right]^{1/2}\right\}^2. \tag{4.48}
$$

Note that $H_{\bar{\delta}} \preceq \nabla^2F(\bar{x}_{\bar{\delta}})$ due to (4.44). The first term of the right-hand side of (4.48) satisfies

$$
[\cdots] \le (\bar{x}_{\bar{\delta}+}-\bar{x}_{\bar{\delta}})^T\nabla^2F(\bar{x}_{\bar{\delta}})(\bar{x}_{\bar{\delta}1}-\bar{x}_{\bar{\delta}}) = \Delta^2. \tag{4.49}
$$

On the other hand, by substituting $\bar{x}_{\bar{\delta}1}$ by $\bar{x}_{\bar{\delta}}$ into (4.45), we get

$$
\bar{\lambda}^2 = (\bar{x}_{\bar{\delta}}-\hat{x})^TA^T\left[A\nabla^2F(\bar{x}_{\bar{\delta}})^{-1}A^T\right]^{-1}A(\bar{x}_{\bar{\delta}}-\hat{x}) = (\bar{x}_{\bar{\delta}}-\hat{x})^TH_{\bar{\delta}}(\bar{x}_{\bar{\delta}}-\hat{x}). \tag{4.50}
$$

Combining (4.48), (4.49) and (4.50), we obtain

$$
\bar{\lambda}_1^2 \le \frac{(\Delta+\bar{\lambda})^2}{(1-\Delta)^2},
$$

which is equivalent to (4.40). □

The proof of Lemma 4.2. 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 4.2 is divided into several steps as follows.

Step 1. First, we prove the following inequality:

$$
\bar{\lambda}_+ \le \frac{1}{(1-\delta_+)}\left\{\delta_+ + \frac{1}{(1-\|p\|_y)}\left[\delta_1+\frac{(2\delta_1-\delta_1^2)}{(1-\delta_1)^2}\|p\|_y+\frac{\|p\|_y^2}{1-\|p\|_y}\right]\right\}, \tag{4.51}
$$

where $p := y_+ - y$. Indeed, it follows (4.39) that

$$\begin{aligned}\bar{\lambda}_+ &= \|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|_{y_+}^* = \left[\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\nabla^2\tilde{d}_{\bar{\delta}}(y_+,t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\right]^{1/2}\\ &\overset{(4.39)}{\leq} \frac{1}{(1-\delta_+)}\left[\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\nabla^2\tilde{d}(y_+,t_+)^{-1}\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\right]^{1/2}\\ &\leq \frac{1}{(1-\delta_+)}\|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|_{y_+}^*.\end{aligned} \tag{4.52}$$

Next, using (4.39) we have

$$\begin{aligned}\|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+)\|_{y_+}^* &\leq \|\nabla\tilde{d}(y_+,t_+)\|_{y_+}^* + \|\nabla\tilde{d}_{\bar{\delta}}(y_+,t_+) - \nabla\tilde{d}(y_+,t_+)\|_{y_+}^*\\ &\overset{(4.39)}{\leq} \|\nabla\tilde{d}(y_+,t_+)\|_{y_+}^* + \delta_+.\end{aligned} \tag{4.53}$$

Since $\tilde{d}(\cdot,t_+)$ is standard self-concordant according to Lemma 3.4, one has

$$\begin{aligned}\|\nabla\tilde{d}(y_+,t_+)\|_{y_+}^* &\leq \frac{1}{1-\|y_+-y\|_y}\|\nabla\tilde{d}(y_+,t_+)\|_y^*\\ &= \frac{1}{1-\|p\|_y}\|\nabla\tilde{d}(y_+,t_+)\|_y^*.\end{aligned} \tag{4.54}$$

Plugging (4.54) and (4.53) into (4.52) we obtain

$$\bar{\lambda}_+ \leq \frac{1}{(1-\delta_+)}\left[\frac{\|\nabla\tilde{d}(y_+,t_+)\|_y^*}{1-\|p\|_y} + \delta_+\right]. \tag{4.55}$$

On the other hand, from (4.13), we have

$$\begin{aligned}\nabla\tilde{d}(y_+,t_+) &\overset{(4.13)}{=} \nabla\tilde{d}(y_+,t_+) - \left[\nabla\tilde{d}_{\bar{\delta}}(y,t_+) + \nabla^2\tilde{d}_{\bar{\delta}}(y,t_+)(y_+-y)\right]\\ &= \left[\nabla\tilde{d}(y,t_+) - \nabla\tilde{d}_{\bar{\delta}}(y,t_+)\right]\\ &+ \left\{[\nabla^2\tilde{d}(y,t_+) - \nabla^2\tilde{d}_{\bar{\delta}}(y,t_+)](y_+-y)\right\}\\ &+ \left[\nabla\tilde{d}(y_+,t_+) - \nabla\tilde{d}(y,t_+) - \nabla^2\tilde{d}(y,t_+)(y_+-y)\right].\end{aligned} \tag{4.56}$$

By substituting t by t_+ in (4.39), we obtain an estimate of the first term of (4.56) as

$$\|\nabla\tilde{d}(y,t_+) - \nabla\tilde{d}_{\bar{\delta}}(y,t_+)\|_y^* \leq \|\bar{x}_{\bar{\delta}1} - x_1^*\|_{x_1^*} = \delta_1. \tag{4.57}$$

Next, we consider the second term of (4.56). It follows from (4.39) that

$$\begin{aligned}\left[(1-\delta_1)^2 - 1\right]\nabla^2\tilde{d}(y,t_+) &\preceq \nabla^2\tilde{d}_{\bar{\delta}}(y,t_+) - \nabla^2\tilde{d}(y,t_+)\\ &\preceq \left[(1-\delta_1)^{-2} - 1\right]\nabla^2\tilde{d}(y,t_+).\end{aligned} \tag{4.58}$$

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

$$\|[\nabla^2\tilde{d}(y,t) - \nabla^2\tilde{d}_{\bar{\delta}}(y,t_+)](y_+-y)\|_y^* = \|Gp\|_y^* \leq \|H\|\|p\|_y, \tag{4.59}$$

where, by virtue of (4.58) and the condition $\delta_1 < 1$, one has

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

Hence, (4.59) leads to

$$\|[\nabla^2\tilde{d}(y,t) - \nabla^2\tilde{d}_{\bar{\delta}}(y,t_+)](y_+-y)\|_y^* \leq \frac{(2\delta_1-\delta_1^2)}{(1-\delta_1)^2}\|p\|_y. \tag{4.60}$$

Furthermore, since $\tilde{d}(\cdot,t)$ is standard self-concordant, similar to the proof of [17, Theorem 4.1.14], we have

$$\|\nabla\tilde{d}(y_+,t_+) - \nabla\tilde{d}(y,t_+) - \nabla^2\tilde{d}(y,t_+)(y_+-y)\|_y^* \leq \frac{\|p\|_y^2}{1-\|p\|_y}. \tag{4.61}$$

Now, we apply the triangle inequality $\|a+b+c\|_y^* \leq \|a\|_y^* + \|b\|_y^* + \|c\|_y^*$ to (4.56) and then plugging (4.57), (4.60) and (4.61) into the resulting inequality, we obtain

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

Finally, by substituting this inequality into (4.55) we get (4.51).

Step 2. Next, we estimate (4.51) in terms of $\bar{\lambda}_1$ to obtain

$$\bar{\lambda}_+ \leq \frac{1}{(1-\delta_+)}\left[\left(\frac{\bar{\lambda}_1}{1-\delta_1-\bar{\lambda}_1}\right)^2 + \frac{(2\delta_1-\delta^2)}{(1-\delta_1)^2}\left(\frac{\bar{\lambda}_1}{1-\delta_1-\bar{\lambda}_1}\right) + \frac{(1-\delta_1)\delta_1}{1-\delta_1-\bar{\lambda}_1} + \delta_+\right]. \tag{4.62}$$

Indeed, by using (4.42) with $x = \bar{x}_{\bar{\delta}1}$ and $z = x_1^*$ and then (3.10) we have

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

This inequality together with the definition of $|\!|\!| \cdot |\!|\!|$ imply

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

Moreover, since $|\!|\!|p|\!|\!|_y = |\!|\!|\nabla\tilde{d}_{\bar{\delta}}(y,t_+)|\!|\!|_y^* = \bar{\lambda}_1$ due to (4.13), the last inequality is equivalent to

$$\|p\|_y \leq \frac{\bar{\lambda}_1}{1-\delta_1}. \tag{4.63}$$

Note that the right-hand side of (4.51) is nondecreasing w.r.t. $\|p\|_y$ in $[0,1)$. Substituting (4.63) into (4.51) we finally obtain (4.62).

Step 3. We further estimate (4.62) in terms of Δ and $\bar{\lambda}$. First, we can easily check that the right-hand side of (4.62) is nondecreasing with respect to $\bar{\lambda}_1$, δ_1 and δ_+. Now, by using the definitions of Δ and $\bar{\lambda}$, it follows from Lemma 4.6 c) that

$$\bar{\lambda}_1 \leq \frac{\bar{\lambda}+\Delta}{1-\Delta}.$$

Since $\delta_+ < 1$ and $\delta_1 + 2\Delta + \bar{\lambda} < 1$, substituting this inequality into (4.62), we obtain

$$\begin{aligned}\bar{\lambda}_+ \leq & \frac{1}{(1-\delta_+)}\left[\delta_+ + \left(\frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}\right)^2 + \frac{(2\delta_1-\delta_1^2)}{(1-\delta_1)^2}\left(\frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}\right)\right. \\ & \left. + \frac{\delta_1(1-\delta_1)(1-\Delta)}{1-\delta_1-2\Delta-\bar{\lambda}}\right]. \end{aligned} \tag{4.64}$$

The right-hand side of (4.64) is well-defined and also nondecreasing with respect to all variables.

Step 4. Finally, we facilitate the right-hand side of (4.64) to obtain (4.14). Since $\bar{\lambda} \geq 0$, we have

$$\begin{aligned}(1-\delta_1)(1-\Delta) &= 1-\Delta-\delta_1+\delta_1\Delta = [1-\delta_1-2\Delta-\bar{\lambda}] + (\bar{\lambda}+\Delta) + \delta_1\Delta \\ &\leq [1-\delta_1-2\Delta-\bar{\lambda}] + (\bar{\lambda}_1+\Delta) + \delta_1(\Delta+\bar{\lambda}) \\ &= [1-\delta_1-2\Delta-\bar{\lambda}] + (1+\delta_1)(\bar{\lambda}+\Delta).\end{aligned}$$

Therefore, this inequality implies

$$\frac{\delta_1(1-\delta_1)(1-\Delta)}{1-\delta_1-2\Delta-\bar{\lambda}} \leq \delta_1 + \delta_1(1+\delta_1)\left[\frac{\Delta+\bar{\lambda}}{1-\delta_1-2\Delta-\bar{\lambda}}\right]. \tag{4.65}$$

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

$$\begin{aligned}\frac{2\delta_1-\delta_1^2)}{(1-\delta_1)^2} + \delta_1(1+\delta_1) &= \delta_1\left[\frac{1}{(1-\delta_1)^2} + \frac{1}{(1-\delta_1)} + (1+\delta_1)\right] \\ &\leq \delta_1\left[\frac{1}{(1-\delta_1)^2} + \frac{2}{1-\delta_1}\right].\end{aligned}$$

Substituting inequality (4.65) into (4.64) and then using the last inequality and $\xi := \frac{\bar{\lambda}+\Delta}{1-\delta_1-2\Delta-\bar{\lambda}}$, we obtain (4.14).

Step 5. The nondecrease of the right-hand side of (4.14) is obvious. The inequality (4.15) follows directly from (4.14) by noting that $\bar{\lambda} \equiv \lambda$ and $\bar{x}_{\bar{\delta}} \equiv x^*$. □

5. Path-following decomposition algorithm with exact Newton iterations. In Algorithm 1, if we set $\bar{\delta} = 0$, then this algorithm reduces to the ones considered in [10, 15, 22, 28, 29]. However, we emphasize the following points.

1. We consider this variant as a special case of the algorithm presented in the previous sections which is called *path-following decomposition algorithm with exact Newton iterations*.
2. In [10, 15, 22, 28, 29], since the primal subproblem (3.2) is solved exactly, the family $\{d(\cdot,t)\}_{t>0}$ of the smooth dual functions is strongly self-concordant due to Legendre transformation. Consequently, the standard theory of interior point methods in [16] can be applied to minimize this function. In contrast to those methods, in this paper we analyze directly the path-following iterations to select appropriate parameters for implementation.

Note that the radius of the neighbourhood of the analytic center in Algorithm 3 below is $\beta^* = \frac{1}{2}(3-\sqrt{5}) \approx 0.381966$ compared to the one used in literature, $\beta^* = 2-\sqrt{3} \approx 0.26795$.

5.1. Analyzing the exact path-following iteration. Let us assume that the primal subproblem (3.2) is solved exactly, i.e. $\bar{\delta} = 0$. 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 (4.17) 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 sake of notational simplicity, we denote by $\tilde{\lambda} := \lambda_{\tilde{d}(\cdot,t)}(y)$, $\tilde{\lambda}_1 := \lambda_{\tilde{d}(\cdot,t_+)}(y)$ and $\tilde{\lambda}_+ := \lambda_{\tilde{d}(\cdot,t_+)}(y_+)$. It follows from (4.15) of Lemma 4.2 that

$$\tilde{\lambda}_+ \le \left(\frac{\tilde{\lambda} + \Delta^*}{1 - 2\Delta^* - \tilde{\lambda}}\right)^2. \tag{5.2}$$

Now, we fix $\beta \in (0,1)$ such 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 with respect to $\tilde{\lambda}$, it implies that $\tilde{\lambda}_+ \le \left(\frac{\Delta^*+\beta}{1-2\Delta^*-\beta}\right)^2$. Thus $\tilde{\lambda}_+ \le \beta$ if $\frac{\Delta^*+\beta}{1-2\Delta^*-\beta} \le \sqrt{\beta}$ which leads to

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

provided that

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

Since $\Delta \equiv \Delta^*$, according to (4.18), we can choose

$$\Delta t := \sigma t = \frac{\bar{\Delta}^* t}{\sqrt{\nu} + (\sqrt{\nu}+1)\bar{\Delta}^*}, \tag{5.5}$$

where $\sigma := \frac{\bar{\Delta}^*}{\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}^*}$. Therefore, t is updated by $t_+ := t - \Delta t = (1-\sigma)t$. Note that t decreases linearly with the contraction factor $(1-\sigma)$.

In particular, if we choose $\beta = \frac{\beta^*}{4} \approx 0.095492$ then $\bar{\Delta}^* \approx 0.113729$, which leads to $(1-\sigma) = \frac{\sqrt{\nu}(\bar{\Delta}^*+1)}{\sqrt{\nu}(\bar{\Delta}^*+1)+\bar{\Delta}^*} \approx \frac{1.1137\sqrt{\nu}}{1.1137\sqrt{\nu}+0.1137}$.

5.2. The algorithm and its convergence. Let us fix an initial value $t = t_0 > 0$ and $\beta \in (0, \beta^*)$, where β^* is given in (5.4). First, we apply Phase 1 to find a starting point $y^0 \in Y$ such that $\tilde{\lambda}_0 := \lambda_{\tilde{d}(\cdot,t_0)}(y^0) \le \beta$. This phase is carried out by applying the damped Newton iteration scheme proposed in [17]. Then we perform the path-following algorithm. From Definition 3.8, we can see that if $t_k \le \frac{\varepsilon_d}{\omega_*(\beta)}$ then y^k is a $2\varepsilon_d$-solution of (2.2). The algorithm is presented in detail as follows.

ALGORITHM 3. (*Path-following algorithm with exact Newton iterations*)

Initialization: Perform the following steps:

1. Fix a constant $\beta \in (0, \beta^*)$ (e.g. $\beta = \frac{1}{4}\beta^*$), where $\beta^* = \frac{3-\sqrt{5}}{2} \approx 0.381966$.
2. Compute $\bar{\Delta} := \frac{\sqrt{\beta}(1-\sqrt{\beta}-\beta}{1+2\sqrt{\beta}}$ and $\sigma := \frac{\bar{\Delta}}{\sqrt{\nu}+(\sqrt{\nu}+1)\bar{\Delta}}$.

3. Fix a tolerance $\varepsilon_d > 0$ and choose an initial value $t_0 > 0$.

Phase 1. (*Finding a starting point*).

1. Choose an arbitrary starting point $y^{0,0} \in Y$.

`For` $j = 0, 1, \cdots$ `do`

1. Solve exactly the primal subproblem (3.2) *in parallel* to obtain $x^*(y^{0,j}, t_0)$.
2. Evaluate $\nabla d(y^{0,j}, t_0)$ and $\nabla d(y^{0,j}, t_0)$ by (3.10) and (3.10), respectively.
3. Compute the Newton decrement $\tilde{\lambda}_j = \lambda_{\tilde{d}(\cdot,t_0)}(y^{0,j})$.
4. If $\tilde{\lambda}_j \leq \beta$ then set $y^0 := y^{0,j}$ and terminate.
5. Update $y^{0,j+1}$ as $y^{0,j+1} := y^{0,j} - \alpha_j \nabla^2 d(y^{0,j}, t_0)^{-1} \nabla d(y^{0,j}, t_0)$, where the step size $\alpha_j := \frac{1}{1+\tilde{\lambda}_j} \in (0, 1]$.

`End of For`.

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

`For` $k = 0, 1, \cdots$ `do`

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

As in Algorithm 1, the main task of this algorithm is Step 1 in Phase 1 and Step 3 in Phase 2, which can be carried out in parallel, and Step 5 in Phase 1 and Step 4 in Phase 2, which require a centralized computation to solve the linear system $\nabla^2 d(y^k, t_{k+1})\Delta y = -\nabla d(y^k, t_{k+1})$ (see Section 6). In an implementation, the primal subproblem can not be solved exactly but it must be solved up to a very high accuracy.

Since $\tilde{d}(\cdot, t_0)$ is standard self-concordant due to Lemma 3.4. By [17, Theorem 4.1.12], the number of iterations to obtain $y^0 \in Y$ such that $\lambda_{\tilde{d}(\cdot,t_0)}(y^0) \leq \beta$ does not exceed

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

The number $\bar{J}_{\max}$ not only depends on the distance $d(y^{0,0}, t_0) - d^*(t_0)$ but also on t_0. If we choose t_0 small then $\bar{J}_{\max}$ is large, while the number of iterations in Algorithm 3 is small. Therefore, in the implementation, we need to balance between these quantities to get a good performance.

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

THEOREM 5.1. *Let $t_0 > 0$ and $y^0 \in Y$ such that $\lambda_{\tilde{d}(\cdot,t_0)}(y^0) \leq \beta$. Then the maximum number of iterations k needed by Algorithm 3 to obtain a $2\varepsilon_d$ - solution y^k of* (2.2) *does not exceed*

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

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

Proof. From Step 2 of Algorithm 3, we have $t_k = (1-\sigma)^k t_0 = \left(1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}\right)^k t_0$. Algorithm 3 is terminated if $t_k \le \frac{\varepsilon_d}{\omega_*(\beta)}$. Thus $\left(1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}\right)^k \le \frac{\varepsilon_d}{t_0\omega_*(\beta)}$, which leads to (5.7). □

REMARK 7 (**The worst-case complexity**). *Since* $\ln\left(1 + \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}\right) \sim \frac{\bar{\Delta}^*}{\sqrt{\nu}(\bar{\Delta}^*+1)}$, *the worst-case complexity of Algorithm 3 is* $O(\sqrt{\nu}\ln(t_0/\varepsilon_d))$.

REMARK 8 (**Damped Newton iteration**). *Note that, at Step 5 of Algorithm 3, we can use a 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 implementation. In this section, we first show how to handle a general concave objective function. Next, we discuss the solution of the primal subproblem (3.2) including local equality constraints. Finally, we briefly describe a parallel method to compute the Newton-type direction for the master problem.

6.1. Handling general objective function. If ϕ_i is nonlinear, concave and its epigraph is endowed with a self-concordant barrier for some $i \in I_M := \{1, \dots, M\}$, then we propose to use slack variables to move the objective function into the constraints. Let us denote by $\hat{x}_i := (x_i^T, s_i)^T$ and

$$\hat{X}_i := \{(x_i, s_i) \mid x_i \in X_i,\ s_i \ge \underline{s}_i,\ \phi_i(x_i) \ge s_i\},$$

for a sufficiently small value $\underline{s}_i$ such that the constraint $s_i \ge \underline{s}_i$ is inactive. Let $\hat{F}_i$ be a self-concordant barrier of $\hat{X}_i$ and let $\hat{c}_i := (0^T, 1)^T \in \mathbb{R}^{n_i+1}$. Then problem (1.1) can be transformed into a convex separable optimization problem with linear objective function. In this case, the algorithms developed in the previous sections can be applied to solve the resulting problem.

If ϕ_i is concave quadratic then, according to [16, Theorem 3.3.1], we can construct a self-concordant barrier $G_i(\hat{x}_i) := -\ln(\phi_i(x_i) - s_i)$ for the epigraph of ϕ_i. Particularly, the optimality condition for this problem is $\hat{c} + \hat{A}^T y - t\nabla\hat{F}(\hat{x}) = 0$, which can be written as

$$\begin{cases} A^T y - t\nabla F(x) - t\mathrm{diag}(f_i(x_i) - s_i)^{-1}\nabla f(x) = 0, \\ t\mathrm{diag}(f_i(x_i) - s_i)^{-1} = 1. \end{cases}$$

By substituting the second line into the first line of the above expression, we obtain

$$A^T y - t\nabla F(x) + \nabla f(x) = 0.$$

However, this condition is indeed the optimality condition of the following problem

$$d(y,t) := \max_{x\in \mathrm{int}(X)} \left\{ f(x) + y^T(Ax - b) - t[F(x) - F(x^c)] \right\}. \tag{6.1}$$

Consequently, the algorithms developed in the previous sections can be applied to solve (1.1) without moving ϕ_i into the constraints.

Several examples of convex problems for which the logarithmic function $G_i(\hat{x}_i)$ is self-concordant can be found in [8]. Note that, in some problems, we may need to reformulate the epigraph of f_i to obtain a self-concordant barrier. For example, many optimization problems in network use an objective function of the form $\phi_i(x_i) = \frac{x_i}{1-x_i}$, where $0 \le x_i < 1$. The inequality presented the epigraph of ϕ_i is $\frac{x_i}{1-x_i} \le s_i$, which is equivalent to $\sqrt{(x_i + s_i)^2 + 4} \le x_i - s_i - 2$. The last inequality is indeed a second order cone constraint endowed with a 2-self-concordant barrier [16].

6.2. Solving the primal subproblems. Let us recall the primal subproblem in (6.1) with a nonlinear objective function. We need to solve this problem inexactly up to a desired accuracy $\varepsilon(t) > 0$, e.g. $\varepsilon(t) = \frac{\bar{\delta}t}{(\nu+2\sqrt{\nu})(1+\bar{\delta})}$. Note that the approximate optimality condition of (3.4) becomes

$$\|\nabla f(x) + A^T y - t\nabla F(\bar{x})\|^*_{x^c} \le \varepsilon(t). \tag{6.2}$$

By separability, this approximate problem can be solved in parallel as

$$\|\nabla f_i(x) + A_i^T y - t\nabla F_i(\bar{x}_i)\|^*_{x_i^c} \le \varepsilon_i(t),\ \varepsilon_i(t) \ge 0,\ i = 1, \dots, M, \tag{6.3}$$

where $\sum_{i=1}^M \varepsilon_i(t) = \varepsilon(t)$. In principle, we can choose $\varepsilon_i(t) = \frac{\varepsilon(t)}{M}$. However, in some practical situations, it is important to choose different $\varepsilon_i(t)$ for different components, especially, when some component problems can be solved analytically in a closed form.

Since F_i is standard self-concordant, the function $\psi_i(x_i; y, t) := F_i(x_i) - t^{-1}(f_i(x_i) + y^T A_i x_i)$ is also standard self-concordant. Moreover, $\nabla\psi_i(x_i; y, t) = \nabla F_i(x_i) - t^{-1}(\nabla f_i(x_i) + A_i^T y)$ and $\nabla^2\psi_i(x_i; y, t) = \nabla^2 F_i(x_i) - \nabla^2 f_i(x_i)$. Since $\nabla^2\psi_i(x_i; y, t) \succ 0$, we define

$$\lambda_{\psi_i}(x_i) := \left[\nabla\psi_i(x_i; y, t)\nabla\psi_i(x_i; y, t)^{-1}\nabla\psi_i(x_i; y, t)\right]^{-1/2},$$

the Newton decrement of ψ_i.

Now, let us apply Newton method to solve problem (6.2). First, we fix $\beta_i \in (0, \beta^*)$, where $\beta^* := \frac{1}{2}(3 - \sqrt{5})$, and choose $x_i^0 \in \mathrm{int}(X_i)$. Then, we generate a sequence $\{x_i^j\}_{j\ge 0}$ as

$$\text{where} \quad \begin{aligned} x_i^{j+1} &:= x_i^j + \alpha_{ij}\Delta x_i^j, \\ \Delta x_i^j &:= -\nabla^2\psi_i(x_i^j; y, t)^{-1}\nabla\psi_i(x_i; y, t) \text{ and } \alpha_{ij} \in (0, 1]. \end{aligned} \tag{6.4}$$

Theoretically, the step-size α_{ij} can be chosen as $\alpha_{ij} := 1$ if $\lambda_{\psi_i}(x_i^j) \le \beta_i$ and $\alpha_{ij} := (1 + \lambda_{\psi_i}(x_i^j))^{-1}$, otherwise. However, this choice is usually too conservative and not preferable in practice. Thus one can use an appropriate line-search procedure to select α_{ij}. Note that in linear programming, F_i is diagonal, e.g. $F_i(x_i) = \mathrm{diag}(-\ln(x_i))$, so that computing the Newton iteration (6.4) requires a low computational cost. In general, we have to solve a linear system of the form

$$\nabla^2\psi_i(x_i^j; y, t)\Delta x_i^j = -\nabla\psi_i(x_i; y, t)$$

to obtain a Newton direction Δx_i^j. The convergence of the Newton scheme (6.4) can be found in [17]. Note that in Algorithms 1 and 2, (6.3) is solved repeatedly at different t_k. It is important to warm-start the Newton iteration (6.4) by using the finally approximate solution of the previous iterate t_{k-1} as a starting point for the current one t_k.

Finally, if the local equality constraints $E_i x_i = f_i$ are available in (1.1) for some $i \in \{1, \dots, M\}$, then the KKT conditions of the primal subproblem i become

$$\begin{cases} c_i + A_i^T y + E_i^T z_i - t\nabla F_i(x_i) = 0, \\ E_i x_i - f_i = 0. \end{cases} \tag{6.5}$$

Instead of the full KKT system (6.5), we consider a reduced KKT condition as follows

$$Z_i^T(c_i + A_i^T y) - tZ_i^T\nabla F_i(Z_i x_i^z + R_i^{-T} f_i) = 0. \tag{6.6}$$

Here, (Q_i, R_i) is a QR-factorization of E_i^T and $[Y_i, Z_i] = Q_i$ 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 $\|\bar{x}_{\bar{\delta}} - x^*\|_{x^*} = \|\bar{x}_{\bar{\delta}}^z - x^{*z}\|_{x^{*z}}$. Therefore, the condition (4.1) coincides with $\|\bar{x}_{\bar{\delta}}^z - x^{*z}\|_{x^{*z}} \le \bar{\delta}$. However, the last condition is satisfied if

$$\|Z_i^T(c_i + A_i^T y) - tZ_i^T \nabla F_i(Z_i x_i^z + R_i^{-T} f_i)\|_{x_i^{cz}}^* \le \varepsilon_i, \tag{6.7}$$

where $\sum_{i=1}^M \varepsilon_i = \varepsilon_p$ and ε_p is defined by (4.9). Note that the QR-factorization of E_i^T can be computed one time, a priori.

6.3. Computing the inexact perturbed Newton direction. Let us rewrite the inexact-perturbed Newton direction in Algorithms 1 and 2 in a unified formula:

$$\Delta y^k := -\nabla^2 d_{\bar{\delta}}(y^k, t)^{-1} \nabla d_{\bar{\delta}}(y^k, t),$$

where t can be t_{k+1} or t_0. We discuss in this subsection how to compute Δy^k in an appropriate way by taking into account the specific structure of problem (1.1). Note that Δy^k is the solution of the following linear system:

$$\nabla^2 d_{\bar{\delta}}(y^k, t)\Delta y^k = -\nabla d_{\bar{\delta}}(y^k, t). \tag{6.8}$$

The gradient vector $\nabla d_{\bar{\delta}}(y^k, t)$ is computed as

$$\nabla d_{\bar{\delta}}(y^k, t) = A\bar{x}_{\bar{\delta}}(y^k, t) - b = \sum_{i=1}^M A_i \bar{x}_i(y^k, t) - b := g_k,$$

and the Hessian matrix $\nabla^2 d_{\bar{\delta}}(y^k, t)$ is obtained from

$$\nabla^2 d_{\bar{\delta}}(y^k, t) = \frac{1}{t}\sum_{i=1}^M A_i \nabla^2 F_i(\bar{x}_i(y^k, t))^{-1} A_i^T := \sum_{i=1}^M A_i G_i^k A_i^T.$$

Note that each block $A_i \bar{x}_i(y^k, t)$ as well as $A_i \nabla^2 F_i(\bar{x}_i(y^k, t))^{-1} A_i^T$ can be computed *in parallel*. Then, the linear system (6.8) can be written as

$$\left(\sum_{i=1}^M A_i G_i^k A_i^T\right) \Delta y^k = -g_k. \tag{6.9}$$

Sine matrix $G_i^k \succeq 0$ and $\sum_{i=1}^M A_i G_i^k A_i^T \succ 0$, one can apply either Cholesky-type factorizations or conjugate gradient (CG) methods to solve this problem. Note that the CG method only requires matrix-vector operations. More details on parallel solution of (6.8) can be found, e.g., in [15, 29]. One possibility to parallelize the centralized computation is using (quasi) Monte-Carlo methods.

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 the area of telecommunications and in other network flow problems such as transportation [9]. Let us consider a network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{A}$ is the set of links. Let $\mathcal{C}$ be a set of commodities to be sent through the network $\mathcal{G}$. 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 $d_k \ge 0$. Each link $(i, j) \in \mathcal{A}$ has a maximum capacity $b_{ij} \ge 0$ in which no congestion is assumed to be appeared, and a linear cost per unit

c_{ij}. The variable u_{ijk} denotes the amount of commodity k that is sent through the link (i,j). Flow exceeding b_{ij} may be sent through the link (i,j) but will then causes congestion with an additional nonlinear cost function g_{ij} depending on the exceeded value v_{ij} considered as a variable. We denote by $\mathcal{N}_s$ and $\mathcal{N}_d$ the sets of sources and destinations, respectively. Let $\mathcal{N}_c := \mathcal{N}\backslash(\mathcal{N}_s \cup \mathcal{N}_d)$ and assume that each node in $\mathcal{N}_c$ has at least one ingoing link and one outgoing link.

Mathematically, the optimization model of the routing problem with congestion (RPC) can be formulated as, see, e.g. [9]:

$$\begin{cases} \min\limits_{u_{ijk}, v_{ij}} & \sum\limits_{k\in\mathcal{C}}\sum\limits_{(i,j)\in\mathcal{A}} c_{ij}u_{ijk} + \sum\limits_{(i,j)\in\mathcal{A}} w_{ij}g_{ij}(v_{ij}) \\ \text{s.t.} & \sum\limits_{j:(i,j)\in\mathcal{A}} u_{ijk} - \sum\limits_{j:(j,i)\in\mathcal{A}} u_{jik} = \begin{cases} d_k & \text{if } i\in\mathcal{N}_s, \\ -d_k & \text{if } i\in\mathcal{N}_d, \\ 0 & \text{otherwise,} \end{cases} \\ & \sum\limits_{k\in C} u_{ijk} - v_{ij} = b_{ij},\ (i,j)\in\mathcal{A}, \\ & u_{ijk}\geq 0,\ v_{ij}\geq 0,\ (i,j)\in\mathcal{A}, \end{cases} \tag{7.1}$$

where $w_{ij} \geq 0$ 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 one of the following functions: 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. With these choices, it was shown in [17], the self-concordant barrier function corresponding to the epigraph

$$\mathcal{E}_{g_{ij}} := \{(v_{ij}, s)\in\mathbb{R}_+\times\mathbb{R} \mid g_{ij}(v_{ij})\leq s\}$$

of g_{ij} is given by: a) $F_{ij}(v_{ij}, s_{ij}) = -\ln v_{ij} - \ln(\ln v_{ij} + s_{ij})$ with parameter $\nu_{ij} = 2$ or b) $F_{ij}(v_{ij}, s_{ij}) = -\ln v_{ij} - \ln(s_{ij} - v_{ij}\ln v_{ij})$ with parameter $\nu_{ij} = 2$, respectively. Now, 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

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

with additional constraints $g_{ij}(v_{ij}) \leq s_{ij}$, $(i,j)\in\mathcal{A}$.

It is clear that problem (7.1) is separably convex with respect to M components, n variables u_{ijk}, v_{ij} and s_{ij} and m coupling constraints, where $M := n_{\mathcal{A}}$, $n := n_{\mathcal{C}}n_{\mathcal{A}} + 2n_{\mathcal{A}}$ and $m := n_{\mathcal{C}}n_{\mathcal{N}}$, where $n_{\mathcal{A}} := |\mathcal{A}|$, $n_{\mathcal{C}} := |\mathcal{C}|$ and $n_{\mathcal{N}} := |\mathcal{N}|$. 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 (1.1) with linear objective function (7.2) and the local constraint set (7.3). Note that each primal subproblem of the form (3.2) has $n_{\mathcal{C}} + 2$ variables and one equality constraint.

The aim is to compare the effect of the parameters 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 subproblem to $\varepsilon_p = 10^{-6}$, and the second one is 10^{-10}, where we call them as `A3-v1` and `A3-v2`, respectively. All variants are terminated with the same tolerance $\varepsilon_d = 10^{-4}$. The initial barrier parameter value is set to $t_0 := 0.25$.

The algorithms are implemented in C++ running on a PC Desktop Intel® Core(TM)2 Quad CPU Q6600 with 2.4GHz and 3Gb RAM. The algorithms are paralellized by using `OpenMP`. The input data is generated randomly, where the nodes of the network are generated in a rectangle $[0, 100] \times [0, 300]$, the demand d_k is in $[50, 500]$, the weighting vector w is set to 10, the congestion vector is in $[10, 100]$ and the linear cost c_{ij} is the Euclidean length of the link $(i, j) \in \mathcal{A}$. The nonlinear cost function g_{ij} is chosen randomly between two functions in a) and b) defined above.

We test the algorithms on a collection of 108 random problems. The size of these problems varies from $M = 6$ to 14.280 components, $n = 84$ to 77.142 variables and $m = 15$ to 500 coupling constraints.

The performance profiles are shown in Figures 7.1 and 7.2. The first figure shows the performance profile of 4 variants which consists of the total CPU time, the total time of solving the primal subproblems in two phases, the CPU time of Phase 1 and the CPU time of Phase 2 separately in second. As we can see from this figure

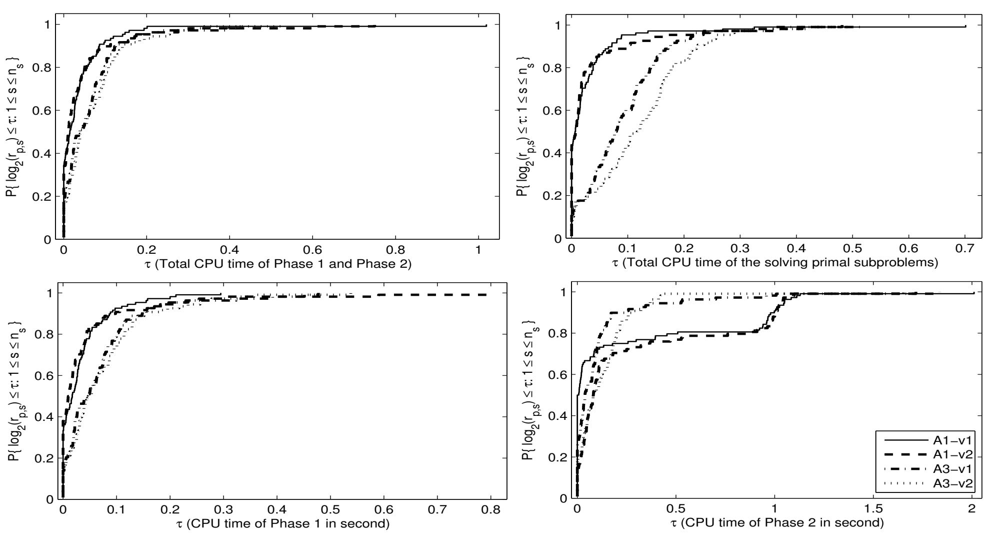

FIG. 7.1. *The CPU time performance profile of four variants.*

that Algorithm 1 performs better than Algorithm 3 both in the total computational time and the computational time for solving the primal subproblems. Moreover, the accuracy in solving the primal subproblems also affects the performance of the algorithms. We also observe that the number of iterations for solving the master problem in Phase 1 for all four variants are almost similar, while they are different in Phase 2. However, Phase 2 is performed when the iteration point is in the quadratic convergence region, it only takes few steps toward the desired approximate solution. Therefore, the computational time of Phase 1 dominates the one in Phase 2. Moreover, in this example, the structure of the master problem is almost dense and we do not use any sparse linear algebra solver. Consequently, the algorithms developed in this paper are recommended to the class of problems with many variables and few coupling constraints in the case the master dual problem possesses dense structure.

In other applications, the efficient methods for sparse linear algebra should be taken into account.

We also compare the total number of iterations for solving the primal subproblems in Figure 7.2. It can be seen from this figure that Algorithm 1 is superior in terms of iterations to Algorithm 3, although the accuracy of solving the primal subproblem in Algorithm 3 is set to 10^{-6}, which is not too high in interior point methods. The

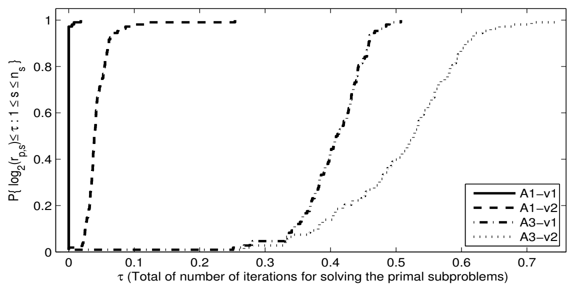

FIG. 7.2. *The iteration performance profile of four variants.*

performance profiles also reveal the effect of the parameters on the number of iterations and computational time. In our numerical results, the inexact version `A1-v1` saves 22% (resp. 23%) the total number of iterations to solve the primal subproblems compared to `A3-v1` (resp. `A3-v2`); while the variant `A1-v2` saves 20% (reps. 21%) compared to `A3-v1` (resp. `A3-v2`). Therefore, in practice, it is valuable to carefully choose appropriate parameters for a specific implementation.

8. Concluding remarks. We have proposed a smoothing technique for Lagrangian decomposition using self-concordant barriers in large-scale convex separable optimization. We provided global and local approximations for the dual function. Then, we proposed a path-following algorithm with inexact perturbed Newton iterations. The convergence of the algorithm has been analyzed and its complexity has been estimated. The theory presented in this paper is significant in practice, since it allows us to solve the primal subproblem inexactly. Moreover, we allow one to balance between the accuracy of solving the primal subproblem and the convergence rate of the path-following algorithm. Even in the exact case, we also obtained a direct analysis for the convergence of the path-following algorithm which was presented by Mehrotra [12] *et al* and Shida [22]. The details of implementation and numerical tests have also been presented. Extensions to the inexactness of linear algebra and to distributed implementation are an interesting and significant future research direction.

Acknowledgments. This research was supported by Research Council KUL: CoE EF/05/006 Optimization in Engineering(OPTEC), GOA AMBioRICS, IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; the Flemish Government via 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 (convex MPC), G.0558.08 (Robust MHE), G.0557.08, G.0588.09, research communities (ICCoS, ANMMM, MLDM) and via IWT: PhD Grants, McKnow-E, Eureka-Flite+EU: ERNSI; FP7-HD-MPC (Collaborative Project STREP-grantnr. 223854), Contract Research: AMINAL, HIGHWIND, and Helmholtz Gemeinschaft: viCERP; Austria: ACCM, and the Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011), European Union FP7-EMBOCON under grant agreement no 248940; CNCS-UEFISCDI (project TE231, no. 19/11.08.2010); ANCS (project PN II, no. 80EU/2010); Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of

Labor, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557.

Appendix A. The proof of the technical statements. In this appendix, we provide a complete proof of Lemmas 3.1, 3.2 and 3.3.

A.1. The proof of Lemma 3.1. *Proof.* Since F_i is standard self-concordant, according to [17, Theorem 4.1.7, inequality 4.1.8] we have

$$\begin{aligned} F_i(y_i) &\geq F_i(x_i) + \nabla F_i(x_i)^T(y_i - x_i) + \omega(\|y_i - x_i\|_{x_i}) \\ &\geq F_i(x_i) - \|\nabla F_i(x_i)\|_{x_i}^*\|y_i - x_i\|_{x_i} + \omega(\|y_i - x_i\|_{x_i}). \end{aligned}$$

This inequality implies

$$\begin{aligned} F_i(x_i) - F_i(y_i) &\leq \|\nabla F_i(x_i)\|_{x_i}^*\|y_i - x_i\|_{x_i} - \omega(\|y_i - x_i\|_{x_i}) \\ &\leq \max_{x_i \in \mathrm{dom}(F_i)} \left\{\|\nabla F_i(x_i)\|_{x_i}^*\|y_i - x_i\|_{x_i} - \omega(\|y_i - x_i\|_{x_i})\right\} \\ &\leq \max_{\xi := \|y_i - x_i\|_{x_i} \geq 0} \left\{\|\nabla F_i(x_i)\|_{x_i}^*\xi - \omega(\xi)\right\} \\ &= \omega_*(\|\nabla F_i(x_i)\|_{x_i}^*). \end{aligned}$$

Here, the last equality follows from [17, Lemma 4.1.4] and the assumption that $\lambda_{F_i}(x_i^*(y,t)) < 1$. Using the above inequality with $y_i = x_i^c$ and $x_i = x_i^*(y,t)$ we have

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

Now, we prove (3.5). Let $x_i(\alpha) := x_i^*(y,t) + \alpha(x_i^*(y) - x_i^*(y,t))$ with $\alpha \in [0,1)$. Since $x_i^*(y,t) \in \mathrm{int}(X_i)$ and $\alpha < 1$, $x_i(\alpha) \in \mathrm{int}(X_i)$. By applying [16, inequality 2.3.3], we have

$$F_i(x_i(\alpha)) \leq F_i(x_i^*(y,t)) - \nu_i \ln(1-\alpha),$$

which is equivalent to

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

Now, from the definition of $d_i(y,t)$, the concavity of ϕ_i and $d_i(y)$, and (A.1) we have

$$\begin{aligned} d_i(y,t) &= \max_{x_i \in \mathrm{int}(X_i)} \left\{\phi_i(x_i) + y^TA_ix_i - t[F_i(x_i) - F_i(x_i^c)]\right\} \\ &\geq \max_{\alpha\in[0,1)} \left\{\phi_i(x_i(\alpha)) + y^TA_ix_i(\alpha) - t[F_i(x_i(\alpha)) - F_i(x_i^c)]\right\} \\ &\geq \max_{\alpha\in[0,1)} \Big\{\alpha[\phi_i(x_i^*(y)) + y^TA_ix_i^*(y)] + (1-\alpha)[\phi_i(x_i^*(y,t)) + y^TA_ix_i^*(y,t)] \\ &\quad -t[F_i(x_i(y,t)) - F_i(x_i^c)] + \nu_i t \ln(1-\alpha)\Big\} \\ &= \max_{\alpha\in[0,1)} \Big\{\alpha d_i(y) + (1-\alpha)d_i(y,t) - \alpha t[F_i(x_i(y,t)) - F_i(x_i^c)] + \nu_i t\ln(1-\alpha)\Big\} \\ &\overset{(A.1)}{\geq} \max_{\alpha\in[0,1)} \Big\{\alpha d_i(y) + (1-\alpha)d_i(y,t) + t\nu_i\ln(1-\alpha) - \alpha t\omega_*(\lambda_{F_i}(x_i^*(y,t)))\Big\}. \end{aligned} \tag{A.3}$$

Rearranging (A.3), we obtain

$$d_i(y,t) \geq d_i(y) - t\omega_*(\lambda_{F_i}(x_i^*(y,t))) + t\nu_i\frac{\ln(1-\alpha)}{\alpha},\ \forall\alpha\in[0,1). \tag{A.4}$$

Since $\frac{\ln(1-\alpha)}{\alpha} \le -1$ for all $\alpha \in (0,1)$ and $\lim_{\alpha\to 0^+} \frac{\ln(1-\alpha)}{\alpha} = -1$. Inequality (A.4) implies that

$$d_i(y,t) - d_i(y) \ge -t[\omega_*(\lambda_{F_i}(x_i^*(y,t))) + \nu_i].$$

which is the right-hand side of (3.5). The left-hand side of (3.5) follows from the relation $F_i(x_i) - F_i(x_i^c) \ge \omega(\|x_i - x_i^c\|_{x_i^c}) \ge 0$ due to (3.1). □

A.2. The proof of Lemma 3.2. *Proof.* The second inequality in (3.7) is proved in Lemma 3.1. We now prove the third one. Let us denote by $x_i^\tau(y) := x_i^c + \tau(x_i^*(y) - x_i^c)$, where $\tau \in [0,1]$. Since F_i is ν_i-self-concordant, it follows from [16, 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 we have

$$\begin{aligned}
d_i(y,t) &= \max_{x_i \in \mathrm{int}(X_i)} \left\{\phi_i(x_i) + y^T A_i x_i - t[F_i(x_i) - F_i(x_i^c)]\right\}\\
&\ge \max_{\tau\in[0,1)} \left\{\phi_i(x_i^\tau(y)) + y^T A_i x_i^\tau(y) - t[F_i(x_i^\tau(y)) - F_i(x_i^c)]\right\}\\
&\ge \max_{\tau\in[0,1)} \left\{(1-\tau)[\phi_i(x_i^c) + y^T A_i x_i^c] + \tau[\phi_i(x_i^*(y)) + y^T A_i x_i^*(y)] + t\nu_1 \ln(1-\tau)\right\}\\
&= \max_{\tau\in[0,1)} \left\{(1-\tau) d_i^c(y) + \tau d_i(y) + t\nu_i \ln(1-\tau)\right\}.
\end{aligned} \tag{A.5}$$

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

$$d_i(y) \le d_i(y,t) + t\nu_i \left(1 + \left[\ln \frac{d_i(y) - d_i^c(y)}{t\nu_i}\right]_+\right).$$

Summing up this inequality for $i = 1,2$ we get (3.7). □

A.3. The proof of Lemma 3.3. *Proof.* Let us fix $\kappa \in (0,1)$, it is trivial that $\ln(x^{-1}) \le x^{-\kappa}$ for $0 < x \le \kappa^{1/\kappa}$. Therefore, we have

$$\nu_i t(1 + [\ln(K_i/(\nu_i t)]_+) \le \nu_i t(1 + (K_i/(\nu_i t))^{-\kappa}) \le [\nu_i + \nu_i(K_i/\nu_i)^\kappa] t^{1-\kappa},\ \ \forall t \le \frac{\nu_i}{K_i}\kappa^{1/\kappa}.$$

Consequently, if $t \le \min\left\{\frac{\nu_i}{K_i}\kappa^{1/\kappa}, \left(\frac{\varepsilon}{2[\nu_i + \nu_i(K_i/\nu_i)^\kappa}\right)^{1/(1-\kappa)}\right\}$ then $\nu_i t(1 + [\ln(K_i/(\nu_i t)]_+) \le 0.5\varepsilon$. Summing up this inequality for $i = 1,2$, we get (3.8) in Lemma 3.3. □

REFERENCES

[1] Bernstein, D.S.: *Matrix mathematics: Theory, facts and formulas with application to linear systems theory.* Princeton University Press, Princeton and Oxford (2005).

[2] Bertsekas, D.P., and Tsitsiklis, J.N.: *Parallel and Distributed Computation: Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall, (1989).

[3] Bertsekas, D.P.: Incremental proximal methods for large-scale convex optimization. Report LIDS - 2847 (2010).

[4] Boyd, S. and Vandenberghe, L.: *Convex Optimization*. University Press, Cambridge (2004).
[5] Chen, G., and Teboulle, M.: A proximal-based decomposition method for convex minimization problems. Math. Program., **64**, 81–101 (1994).
[6] Connejo, A. J., Mínguez, R., Castillo, E. and García-Bertrand, R.: *Decomposition Techniques in Mathematical Programming: Engineering and Science Applications*. Springer-Verlag, (2006).
[7] Fukuda, M., Kojima, M. and Shida, M.: Lagrangian dual interior-point methods for semidefinite programs. SIAM J. Optim. **12**, 1007–1031 (2002).
[8] Hertog, D.D.: *Interior point approach to linear, quadratic and convex programming: Algorithms and complexity*. PhD Thesis, Delf University, Netherland (1992).
[9] Holmberg, K. and Kiwiel, K.C.: Mean value cross decomposition for nonlinear convex problem. Optim. Methods and Softw. **21**(3), 401–417 (2006).
[10] Kojima, M., Megiddo, N. and Mizuno, S. et al: Horizontal and vertical decomposition in interior point methods for linear programs. Technical Report. Information Sciences, Tokyo Institute of Technology (1993).
[11] Komodakis, N., Paragios, N., and Tziritas, G.: MRF Energy Minimization & Beyond via Dual Decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence (in press).
[12] Mehrotra, S. and Ozevin, M. G.: Decomposition Based Interior Point Methods for Two-Stage Stochastic Convex Quadratic Programs with Recourse. Operation Research, **57**(4), 964–974 (2009).
[13] Neveen, G., Jochen, K.: Faster and simpler algorithms for multi-commodity flow and other fractional packing problems. SIAM J. Comput. **37**(2), 630–652 (2007).
[14] Necoara, I. and Suykens, J.A.K.: Applications of a smoothing technique to decomposition in convex optimization, IEEE Trans. Automatic control, **53**(11), 2674–2679 (2008).
[15] Necoara, I. and J.A.K. Suykens, J.A.K.: Interior-point Lagrangian decomposition method for separable convex optimization. J. Optim. Theory Appl., **143**, 567–588 (2009).
[16] Nesterov, Y. and Nemerovskii, A.: *Interior point polynomial methods in convex programming: Theory and applications*. SIAM, Philadenphia (1994).
[17] Nesterov, Y.: *Introductory Lectures on Convex Optimization*. Kluwer, Boston (2004).
[18] Nesterov, Y.: Smooth minimization of nonsmooth functions. Math. Program., **103**(1):127–152, (2005).
[19] Renegar, J.: *A Mathematical View of Interior-Point Methods in Convex Programming*. Society for Industrial and Applied Mathematics, Philadelphia, 2001.
[20] Samar, S., Boyd, S., and Gorinevsky,D.: Distributed Estimation via Dual Decomposition. Proceedings European Control Conference (ECC), 1511–1516, Kos, Greece, (2007).
[21] Shapiro, A., Dentcheva, D. and Ruszczyński, A: *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia (2009).
[22] Shida, M.: An interior-point smoothing technique for Lagrangian relaxation in large-scale convex programming. Optimization, **57**(1), 183–200 (2008).
[23] Tran Dinh, Q., Savorgnan, C. and Diehl, M.: Combining Lagrangian Decomposition and Excessive Gap Smoothing Technique for Solving Large-Scale Separable Convex Optimization Problems. `http://arxiv.org/abs/1105.5427`, (2011) (submitted).
[24] Wei, E, Ozdaglar, A. and Jadbabaie, A.: A Distributed Newton Method for Network Utility Maximization. LIDS report 2832, `http://web.mit.edu/asuman/www/publications.htm`, (2011) (submitted).
[25] Venkat, A., Hiskens, I., Rawlings, J., and Wright, S.: Distributed MPC strategies with application to power system automatic generation control. IEEE Trans. Control Syst. Technol. **16**(6), 1192–12-6 (2008).
[26] Xiao, L., Johansson, M. and Boyd, S.: Simultaneous routing and resource allocation via dual decomposition. IEEE Trans. Commun. **52**(7), 1136–1144 (2004).
[27] Zhao, G.: Interior point methods with decomposition for solving large-scale linear programs. J. Optim. Theory Appl. **102**, 169–192 (1999).
[28] Zhao, G.: A Log-barrier with Benders decomposition for solving two-stage stochastic programs. Math. Progam. **90**, 507–536 (2001).
[29] Zhao, G.: A Lagrangian dual method with self-concordant barriers for multistage stochastic convex programming. Math. Progam. **102**, 1–24 (2005).