An interior-point Lagrangian decomposition method for separable convex optimization

I. Necoara[1,2] and J.A.K. Suykens[1,2]

September 2008

[1] Katholieke Universiteit Leuven, Department of Electrical Engineering (ESAT), Kasteelpark Arenberg 10, B–3001 Leuven (Heverlee), Belgium.

[2] We acknowledge financial support from Research Council K.U. Leuven: GOA AMBioRICS, CoE EF/05/006, OT/03/12, PhD/postdoc & fellow grants; Flemish Government: FWO PhD/postdoc grants, FWO projects G.0499.04, G.0211.05, G.0226.06, G.0302.07; Research communities (ICCoS, ANMMM, MLDM); AWI: BIL/05/43, IWT: PhD Grants; Belgian Federal Science Policy Office: IUAP DYSCO.

Abstract. In this paper we propose a distributed algorithm for solving large-scale separable convex problems using Lagrangian dual decomposition and the interior-point framework. By adding self-concordant barrier terms to the ordinary Lagrangian we prove under mild assumptions that the corresponding family of augmented dual functions is self-concordant. This makes it possible to efficiently use the Newton method for tracing the central path. We show that the new algorithm is globally convergent and highly parallelizable and thus it is suitable for solving large-scale separable convex problems.

Keywords. Separable convex optimization, self-concordant function, interior-point methods, augmented Lagrangian decomposition, parallel computations.

1 Introduction

Can self-concordance and interior-point methods be incorporated in a Lagrangian dual decomposition framework? This paper presents a decomposition algorithm that incorporates the interior-point method into augmented Lagrangian decomposition technique for solving large-scale separable convex problems. Separable convex problems, i.e. optimization problems with separable convex objective function but with coupling constraints, arise in many fields: networks (communication networks, multicommodity network flows) [1,2], process system engineering (e.g. distributed model predictive control) [3], stochastic programming [4], etc. There has been considerable interest in parallel and distributed computation methods for solving this type of structured optimization problems and many methods have been proposed: dual subgradient methods [5,6], alternating direction methods [5,8,9], proximal method of multipliers [10], partial inverse methods [11], proximal center methods [12], interior-point based methods [2,4,13], etc.

The methods mentioned above belong to the class of augmented Lagrangian or multiplier methods [5], i.e. they can be viewed as techniques for maximizing an augmented dual function. For example in the alternating direction method a quadratic penalty term is added to the standard Lagrangian to obtain a smooth dual function. However, the quadratic term destroys the separability of the given problem. Hence such a method avoids inseparability by alternating minimization in a Gauss-Seidel fashion of the augmented Lagrangian followed by a steepest ascent update for the multipliers. The partial inverse method can be seen as the proximal point algorithm applied to a particular maximal monotone operator [11]. In general the performance of these two methods is very sensitive to the variations of their parameters and some rules for choosing these parameters were given e.g. in [7,9]. In the proximal center method we use smoothing techniques in order to obtain a well-behaved Lagrangian, i.e. we add a separable strongly convex term to the ordinary Lagrangian. This leads to a smooth dual function, i.e. with Lipschitz continuous gradient, which preserves separability of the problem, the corresponding parameter is selected optimally and moreover the multipliers are updated using an optimal gradient based scheme (see [12] for more details). In [4, 13] interior-point methods are proposed for solving special classes of separable convex problems, such as block-angular linear programs. In those papers the Newton direction is used to update the multipliers obtaining polynomial-time complexity for the proposed algorithms.

In the present paper we use a similar smoothing technique as in [12] in order to obtain a well-behaved augmented dual function. Although we relax the coupling constraints using the Lagrangian dual framework as in [12], the main difference here is that the smoothing term is a self-concordant barrier, while in [12] the main property of the smoothing term was strong convexity. Therefore, using the properties of self-concordant functions we show that the augmented dual function becomes under mild assumptions also self-concordant. Hence the Newton direction can be used instead of gradient based directions as it is done in most of the augmented Lagrangian methods. Furthermore, we develop a specialized interior-point method to maximize the augmented dual function which takes into account the special structure of our problem. We present a parallel algorithm for computing the Newton direction of the dual function and we also prove global convergence of the proposed method.

The main contributions of the paper are the following:

(i) We consider a more general model for separable convex problems that includes local equality and inequality constraints, and linear coupling constraints which generalizes the models in $[2, 4, 13]$ (mostly specialized for separable linear programs).

(ii) We derive sufficient conditions for self-concordance of augmented Lagrangian and we prove self-concordance for the corresponding family of augmented dual functions.

(iii) We provide an interior-point based algorithm for solving the dual problem with proofs of global convergence and polynomial-time complexity.

(iv) We propose a practical implementation of the algorithm based on solving approximatively the subproblems and on parallel computations of the Newton directions.

Note that item (ii) generalizes the result of [13] obtained for the case of linear programs. However, the consideration of general convex problems with local equality constraints requires new proofs with more involved arguments in order to prove self-concordance for the family of augmented dual functions.

The paper is organized as follows. In Section 2 we formulate the separable convex problem followed by a brief description of some of the existing decomposition methods for this problem. The main results are given in Sections 3 and 4. In Section 3 we show that the augmented Lagrangian obtained by adding self-concordant barrier terms to the ordinary Lagrangian forms a self-concordant family of dual functions. Then an interior-point Lagrangian decomposition algorithm with polynomial complexity is proposed in Section 4. The new algorithm makes use of the special structure of our problem so that it is highly parallelizable and it can be effectively implemented on parallel processors. We conclude the paper with some possible applications.

Throughout the paper we use the following notations. For a function ψ with two arguments, scalar parameter t and decision variable x, i.e. $\psi(t, x)$, we use “ $'$” for the partial derivative of $\psi(t, x)$ with respect to t and “∇” with respect to x: e.g. $\nabla\psi'(t, x) = \frac{\partial^2}{\partial t \partial x}\psi(t, x)$. For a function ϕ, three times differentiable, i.e. $\phi \in \mathcal{C}^3(\operatorname{dom}\phi)$,

$\nabla^3\phi(x)[h_1, h_2, h_3]$ denotes the third differential of ϕ at x along directions h_1, h_2 and h_3. We use the notation $A \preceq B$ if $B - A$ is positive semidefinite. We use D_A to denote the block diagonal matrix having on the main diagonal the matrices $A_1, \cdots, A_N$.

2 Problem formulation

We consider the following general *separable convex* optimization problem:

$$f^* = \min_{x_1 \in X_1 \cdots x_N \in X_N} \sum_{i=1}^{N} f_i(x_i) \tag{1}$$

$$\text{s.t.} \sum_{i=1}^{N} B_i x_i = a, \quad A_i x_i = a_i \ \forall i = 1 \cdots N, \tag{2}$$

where $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$ are convex functions, X_i are closed convex sets, $A_i \in \mathbb{R}^{m_i \times n_i}$, $B_i \in \mathbb{R}^{m \times n_i}$, $a_i \in \mathbb{R}^{m_i}$ and $a \in \mathbb{R}^m$. For simplicity of the exposition we define the vector $x := [x_1^T \cdots x_N^T]^T$, the function $f(x) : \;=\; \sum_{i=1}^N f_i(x_i)$, the set $X := \prod_{i=1}^N X_i$, the matrix $B := [B_1 \cdots B_N]$ and $n := \sum_{i=1}^N n_i$.

Remark 2.1. (i) Note that we do not assume strict/strong convexity of any function f_i. (ii) Coupling *inequality* constraints $\sum_{i=1}^N B_i x_i \leq a$ can be included in this framework by adding a slack variable x_{N+1}: $\sum_{i=1}^N B_i x_i + x_{N+1} = a$, i.e. $B_{N+1} = I$ and $X_{N+1} = \mathbb{R}^m_+$.

Let $\langle \cdot, \cdot \rangle$ denote the Euclidian inner product on $\mathbb{R}^n$. By forming the Lagrangian corresponding only to the coupling linear constraints (with the multipliers $\lambda \in \mathbb{R}^m$), i.e.

$$L_0(x, \lambda) = f(x) + \langle \lambda, Bx - a \rangle,$$

we can define the standard dual function

$$d_0(\lambda) = \min_x \{L_0(x, \lambda) : x_i \in X_i,\ A_i x_i = a_i \ \forall i = 1 \cdots N\}.$$

Since L_0 preserves the separability of our problem we can use the dual decomposition method [5] by solving in parallel N minimization problems and then updating the multipliers in some fashion. Note that the dual function d_0 is concave but, in general d_0 is not differentiable (e.g. when f is not strictly convex). Therefore, for maximizing d_0 one has to use involved nonsmooth optimization techniques [5,6]. From duality theory one knows that if (x^*, λ^*) is a saddle point for the Lagrangian L_0, then under appropriate conditions (constraint qualification), x^* is an optimal solution for the primal (1)–(2) and λ^* is an associated dual optimal multiplier for the dual problem: $\max_{\lambda \in \mathbb{R}^m} d_0(\lambda)$.

In order to obtain a smooth dual function we need to use smoothing techniques applied to the ordinary Lagrangian L_0. One approach is the *augmented Lagrangian* obtained e.g. by adding a quadratic penalty term to the Lagrangian L_0: $t\|Bx - a\|^2$. In the alternating direction method [5,8,9] the minimization of the augmented Lagrangian is performed by alternating minimization in a Gauss-Seidel fashion followed by a steepest ascent update for the multipliers.

In [12] we proposed the *proximal center method* in which we added to the standard Lagrangian a smoothing term $t\sum_{i=1}^{N} g_{X_i}(x_i)$, where each function g_{X_i} is strongly convex and depends on the set X_i so that the augmented Lagrangian takes the following form:

$$L_t^{\text{prox}}(x, \lambda) = \sum_{i=1}^{N} [f_i(x_i) + t g_{X_i}(x_i)] + \langle \lambda, Bx - a \rangle.$$

Therefore, the augmented Lagrangian L_t^{prox} is strongly convex, preserves separability of the problem like L_0 and the associated augmented dual function

$$d_t^{\text{prox}}(\lambda) = \min_x \{L_t^{\text{prox}}(x, \lambda) : x_i \in X_i,\ A_i x_i = a_i \ \forall i = 1 \cdots N\}$$

is differentiable and has also a Lipschitz continuous gradient. In [12] an accelerated gradient based method is used to maximize the augmented dual function d_t^{prox}, while the corresponding minimization problems are solved in parallel. Moreover, the smoothing parameter t is selected optimally.

Note that the methods discussed above use only the gradient directions of the augmented dual function in order to update the multipliers. Therefore, in the absence of more conservative assumptions like strong convexity, the global convergence rate of these methods is slow, in general sub-linear. In this paper we propose to smoothen the Lagrangian by adding instead of strongly convex terms g_{X_i}, self-concordant barrier terms ϕ_{X_i} associated with the sets X_i, in order to obtain the self-concordant Lagrangian:

$$L_t^{\mathrm{sc}}(x,\lambda) = \sum_{i=1}^{N} [f_i(x_i) + t\phi_{X_i}(x_i)] + \langle \lambda, Bx - a \rangle. \tag{3}$$

In the next section we show, using the theory of self-concordant barrier functions, that for a relatively large class of convex functions f_i (see also Section 5), we can obtain a self-concordant augmented dual function:

$$d^{\mathrm{sc}}(t,\lambda) = \min_x \{L_t^{\mathrm{sc}}(x,\lambda) : x_i \in \mathrm{int}(X_i),\ A_i x_i = a_i\ \forall i = 1 \cdots N\}. \tag{4}$$

This open the possibility of deriving an interior-point method using Newton directions for updating the multipliers to speed up the convergence rate of the proposed algorithm.

3 Sufficient conditions for self-concordance of the augmented dual function

In this section we derive sufficient conditions under which the family of augmented dual functions is self-concordant. A key property that allows to prove polynomial convergence for barrier type methods is the property of self-concordance:

Definition 3.1. [14] A closed convex function ϕ with open convex domain $X_\phi \subseteq \mathbb{R}^n$ is called *M_ϕ-self-concordant*, where $M_\phi \geq 0$, if ϕ is three times continuously differentiable on X_ϕ and if for all $x \in X_\phi$ and $h \in \mathbb{R}^n$ we have

$$\nabla^3 \phi(x)[h, h, h] \leq M_\phi \big(h^T \nabla^2 \phi(x) h\big)^{3/2}. \tag{5}$$

A function ϕ is called *N_ϕ-self-concordant barrier* for its domain X_ϕ if ϕ is 2-self-concordant function and for all $x \in X_\phi$ and $h \in \mathbb{R}^n$ we have

$$\langle \nabla \phi(x), h \rangle^2 \leq N_\phi \; h^T \nabla^2 \phi(x) h. \tag{6}$$

Note that (5) is equivalent to (see [14]):

$$|\nabla^3 \phi(x)[h_1, h_2, h_3]| \leq M_\phi \prod_{i=1}^{3} \big(h_i^T \nabla^2 \phi(x) h_i\big)^{1/2}. \tag{7}$$

Moreover, if Hessian $\nabla^2 \phi(x)$ is positive definite, then the inequality (6) is equivalent to

$$\nabla \phi(x)^T \nabla^2 \phi(x)^{-1} \nabla \phi(x) \leq N_\phi. \tag{8}$$

The next lemma provides some basic properties of self-concordant functions.

Proposition 3.1. *[14] Let ϕ be an M_ϕ-self-concordant function such that its domain X_ϕ does not contain straight lines. Then, the Hessian $\nabla^2 \phi(x)$ is positive definite for all $x \in X_\phi$ and ϕ is a barrier function for the closure of X_ϕ, denoted cl(X_ϕ).* □

Note that a self-concordant function which is also a barrier for its domain is called *strongly self-concordant*. The next lemma gives some helpful composition rules for self-concordant functions. We use $\mathrm{int}(X)$ to denote the interior of a set X.

Lemma 3.1. *(i) [14] Any linear or convex quadratic function is 0-self-concordant.*

(ii) [14] Let ϕ_i be M_i-self concordant and let $p_i > 0$, $i = 1, 2$. Then the function $p_1\phi_1 + p_2\phi_2$ is also M-self concordant, where $M = \max\{M_1/\sqrt{p_1}, M_2/\sqrt{p_2}\}$.

(iii) Let $X_{box} = \prod_{i=1}^n [l_i,\ u_i]$ such that $l_i < u_i$ and $\psi \in \mathcal{C}^3(int(X_{box}))$ be convex. If there exists $\beta > 0$ such that for all $x \in int(X_{box})$ and $h \in \mathbb{R}^n$ the following inequality holds

$$|\nabla^3\psi(x)[h,h,h]| \leq \beta\ h^T\nabla^2\psi(x)h\ \sqrt{\sum_{i=1}^n h_i^2/(u_i - x_i)^2 + h_i^2/(x_i - l_i)^2}, \tag{9}$$

then $\bar{\psi}_t(x) = \psi(x) - t\sum_{i=1}^n \ln(u_i - x_i)(x_i - l_i)$ is $2(1+\beta)/\sqrt{t}$-self concordant.

Proof. (i) and (ii) can be found in [14].

(iii) Denote $\phi_{\text{box}}(x) = -\sum_{i=1}^n \ln(u_i - x_i)(x_i - l_i)$. Note that

$$h^T\nabla^2\phi_{\text{box}}(x)h = \sum_{i=1}^n h_i^2/(u_i - x_i)^2 + h_i^2/(x_i - l_i)^2$$
$$\nabla^3\phi_{\text{box}}(x)[h,h,h] = 2\sum_{i=1}^n h_i^3/(u_i - x_i)^3 - h_i^3/(x_i - l_i)^3$$

and using Cauchy-Schwarz inequality it follows that ϕ_{box} is 2-self-concordant function on $\text{int}(X_{\text{box}})$. Let us denote $c = \sqrt{h^T\nabla^2\psi(x)h}$ and $d = \sqrt{\sum_{i=1}^n h_i^2/(u_i - x_i)^2 + h_i^2/(x_i - l_i)^2}$. Using hypothesis (9) and 2-self-concordance of ϕ_{box} we have the following inequalities:

$$|\nabla^3\bar{\psi}_t(x)[h,h,h]| \leq |\nabla^3\psi(x)[h,h,h]| + t|\nabla^3\phi_{\text{box}}(x)[h,h,h]| \leq \beta c^2 d + 2td^3.$$

With some computations we can observe that $(\beta c^2 d + 2td^3)^2 \leq \frac{4(1+\beta)^2}{t}(c^2 + td^2)^3$ and since $h^T\nabla^2\bar{\psi}_t(x)h = c^2 + td^2$, the proof is complete. □

Note that condition (9) is similar to ψ is β-*compatible* with ϕ_{box} on X_{box}, defined in [14].

The following assumptions will be valid throughout this section. We consider a given compact convex set X with nonempty interior and ϕ an associated N_ϕ-self-concordant barrier for X (whenever $X = X_{\text{box}}$ we consider $\phi = \phi_{\text{box}}$). Given a function

$f \in \mathcal{C}^3(\mathrm{int}(X))$, we also assume that it satisfies one of the properties (i)–(iii) of Lemma 3.1, i.e. f is either linear or convex quadratic or M_f-self-concordant or X is a box and f satisfies condition (9). Let $A \in \mathbb{R}^{p\times n}$ and $B \in \mathbb{R}^{m\times n}$ be full row rank matrices so that $p < n$ and $m < n$ and the set $\{x \in \mathbb{R}^n : Ax = a\} \cap \mathrm{int}(X) \neq \emptyset$.

We analyze the following prototype minimization problem:

$$\min_x \{f(x) + t\phi(x) + \langle \lambda, Bx\rangle : \ x \in \mathrm{int}(X), Ax = a\}. \tag{10}$$

Let us define the dual convex function:

$$d(t, \lambda) := \max_x \{-f(x) - t\phi(x) - \langle \lambda, Bx\rangle : \ Ax = a\}.$$

Boundedness of the set X and self-concordance property of the function $f + t\phi$ (which follow from the assumptions mentioned above) guarantee existence and uniqueness of the maximizer $x(t, \lambda)$ of (10). Therefore, we can consistently define the maximizer $x(t, \lambda)$ of (10) and the dual convex function $d(t, \lambda)$ for every $t > 0$ and $\lambda \in \mathbb{R}^m$.

In the following three lemmas we derive the main properties of the family of augmented dual functions $\{d(t, \cdot)\}_{t>0}$.

Lemma 3.2. *For any $t > 0$ the function $d(t, \cdot)$ is M_t-self-concordant, where M_t is either $2/\sqrt{t}$ or $\max\{M_f, 2/\sqrt{t}\}$ or $2(1 + \beta)/\sqrt{t}$.*

Proof. Since f is assumed to be either linear or convex quadratic or M_f-self-concordant or X is a box and f satisfies condition (9) it follows from Lemma 3.1 that $f + t\phi$ is also M_t-self concordant (where M_t is either $2/\sqrt{t}$ or $\max\{M_f, 2/\sqrt{t}\}$ or $2(1 + \beta)/\sqrt{t}$, respectively) and with positive definite Hessian (according to our assumptions and Proposition 3.1). Moreover, $f + t\phi$ is strongly self-concordant since ϕ is a barrier function for X. Since A has full row rank and $p < n$, then there exists some vectors $u_i, i = 1 \cdots n - p$, that form a basis of the null space of this matrix. Let U be the matrix having as columns the vectors u_i and x_0 a particular solution of $Ax = a$. Then, for a fixed t, the feasible set of (10) can be described as

$$Q = \{y \in \mathbb{R}^k : x_0 + Uy \in \mathrm{int}(X)\},$$

which is an open convex set. Using that self-concordance is affine invariant it follows that the functions $\bar{f}(y) = f(x_0 + Uy)$, $\bar{\phi}(y) = \phi(x_0 + Uy)$ have the same properties as the functions f, ϕ, respectively, that $\bar{f} + t\bar{\phi}$ is also M_t-self concordant and that

$$d(t, \lambda) = \max_{y \in Q} [-\bar{f}(y) - t\bar{\phi}(y) - \langle \lambda, B(x_0 + Uy) \rangle].$$

From our assumptions and Proposition 3.1 it follows that the Hessian of ϕ and $\bar{\phi}$ are positive definite. Since f is convex it follows that the Hessian of $\bar{f} + t\bar{\phi}$ is also positive definite and thus invertible. Since $d(t, \cdot)$ is basically the Legendre transformation of $\bar{f} + t\bar{\phi}$, in view of known properties of the Legendre transformation, it follows that if $\bar{f} + t\bar{\phi}$ is convex on X from $\mathcal{C}^3$ such that its Hessian is positive definite, then $d(t, \cdot)$ has the same properties on its domain $X_{d(t,\cdot)} := \{\lambda \in \mathbb{R}^m : -\bar{f}(y) - t\bar{\phi}(y) - \langle \lambda, B(x_0 + Uy) \rangle$ bounded above on $Q\}$. Moreover, from Theorem 2.4.1 in [14] it follows that $d(t, \cdot)$ is also M_t-self-concordant on the domain $X_{d(t,\cdot)}$. □

Lemma 3.3. *The inequality* $|\langle \nabla d'(t, \lambda), h \rangle| \leq (2\xi_t/\alpha_t)\sqrt{h^T \nabla^2 d(t, \lambda) h}$ *holds true for each* $t > 0$ *and* $\lambda, h \in \mathbb{R}^m$*, where* $\xi_t = (M_t/2)\sqrt{N_\phi/t}$*.*

Proof. From Lemma 3.2 we know that $d(t, \cdot)$ is $\mathcal{C}^3$ with positive definite Hessian. By virtue of the barrier ϕ for the set X the optimal solution $x(t, \lambda)$ of (10) satisfies $x(t, \lambda) \in \mathrm{int}(X)$ and so the first-order optimality conditions for optimization problem (10) are: there exists $\nu(t, \lambda) \in \mathbb{R}^p$ such that

$$\nabla f(x(t, \lambda)) + t\nabla\phi(x(t, \lambda)) + B^T\lambda + A^T\nu(t, \lambda) = 0 \text{ and } Ax(t, \lambda) = a. \tag{11}$$

First we determine the formula for the Hessian. It follows immediately from (11) that

$$\nabla d(t,\lambda) = -Bx(t,\lambda) \quad \text{and} \quad \nabla^2 d(t,\lambda) = -B\nabla x(t,\lambda).$$

Let us introduce the following notation:

$$H(t,\lambda) := \nabla^2 f(x(t,\lambda)) + t\nabla^2\phi(x(t,\lambda)).$$

For simplicity, we drop the dependence of all the functions on $x(t,\lambda)$ and (t,λ). Differentiating (11) with respect to λ we arrive at the following system in ∇x and $\nabla\nu$:

$$\begin{bmatrix} \nabla^2 f + t\nabla^2\phi & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \nabla x \\ \nabla\nu \end{bmatrix} = \begin{bmatrix} -B^T \\ 0 \end{bmatrix}.$$

Since H is positive definite and according to our assumption A is full row rank, it follows that the system matrix is invertible. Using the well-known formula for inversion of partitioned matrices we find that:

$$\nabla^2 d = B[H^{-1} - H^{-1}A^T(AH^{-1}A^T)^{-1}AH^{-1}]B^T. \tag{12}$$

Differentiating the first part of (11) with respect to t and using the same procedure as before we arrive at a similar system as above in the unknowns x' and ν'. We find that

$$x' = -[H^{-1} - H^{-1}A^T(AH^{-1}A^T)^{-1}AH^{-1}]\nabla\phi \quad \text{and} \quad \nabla d' = -Bx'.$$

We also introduce the following notation: $F := H^{-1}A^T(AH^{-1}A^T)^{-1}AH^{-1}$ and $G := H^{-1} - F$, which are positive semidefinite. Using a similar reasoning as in [4] and Cauchy-Schwarz inequality we obtain:

$$|\langle \nabla d', h\rangle| = |h^T BG\nabla\phi| \leq \sqrt{h^T BGB^T h}\sqrt{\nabla\phi^T G\nabla\phi} = \sqrt{h^T(\nabla^2 d)h}\sqrt{\nabla\phi^T G\nabla\phi}.$$

Since $G = H^{-1} - F \preceq H^{-1} = (\nabla^2 f + t\nabla^2\phi)^{-1} \ \preceq 1/t(\nabla^2\phi)^{-1}$ and using (8) it follows that $|\langle \nabla d', h\rangle| \leq \sqrt{h^T(\nabla^2 d)h}\sqrt{N_\phi/t}$. $\square$

Lemma 3.4. *The inequality* $|\langle \nabla^2 d'(t,\lambda)h, h\rangle| \leq 2\eta_t \ h^T \nabla^2 d(t,\lambda)h$ *holds true for each* $t > 0$ *and* $\lambda, h \in \mathbb{R}^m$*, where* $\eta_t = (M_t/2)\sqrt{N_\phi/t} + (1/2t)$.

Proof. We recall that $H(t,\lambda) = \nabla^2 f(x(t,\lambda)) + t\nabla^2\phi(x(t,\lambda))$. Therefore

$$h^T H'(t,\lambda)h = (\nabla^3 f(x(t,\lambda)) + t\nabla^3\phi(x(t,\lambda)))[x'(t,\lambda), h, h] + h^T\nabla^2\phi(x(t,\lambda))h,$$
$$h^T(H^{-1}(t,\lambda))'h = -h^T H^{-1}(t,\lambda)H'(t,\lambda)H^{-1}(t,\lambda)h.$$

We again drop the dependence on (t,λ) and after some straightforward algebra computations we arrive at the following expression:

$$\langle \nabla^2 d'h, h\rangle = h^T B^T(H^{-1} - F)H'(H^{-1} - F)Bh.$$

Let us denote with $u := (H^{-1} - F)Bh$. Taking into account the expression of H' derived above we obtain: $|\langle \nabla^2 d'h, h\rangle| = |u^T H'u| = |(\nabla^3 f + t\nabla^3\phi)[x', u, u] + u^T\nabla^2\phi u|$. Using the self-concordance property (7) for $f + t\phi$ we obtain that:

$$|(\nabla^3 f + t\nabla^3\phi)[x', u, u]| \leq M_t u^T(\nabla^2 f + t\nabla^2\phi)u\sqrt{(x')^T(\nabla^2 f + t\nabla^2\phi)x'}$$
$$= M_t u^T H u\sqrt{(x')^T H x'}.$$

Moreover, since f is convex, $\nabla^2 f$ is positive semidefinite and thus: $u^T\nabla^2\phi u \leq (1/t)\, u^T H u$. Combining the last two inequalities we obtain:

$$|\langle \nabla^2 d'h, h\rangle| \leq M_t u^T H u\sqrt{(x')^T H x'} + (1/t)u^T H u. \tag{13}$$

With some algebra we can check that the following identity holds: $FH(H^{-1} - F) = 0$. Based on this identity we can compute $u^T H u$ and $(x')^T H x'$. Indeed,

$$u^T H u = h^T B^T(H^{-1} - F)H(H^{-1} - F)Bh =$$
$$h^T B^T(H^{-1} - F)Bh - h^T B^T F H(H^{-1} - F)Bh = h^T B^T(H^{-1} - F)Bh = h^T\nabla^2 d\ h.$$

Similarly,

$$(x')^T H x' = \nabla\phi^T (H^{-1} - F)\nabla\phi \leq \nabla\phi^T H^{-1}\nabla\phi \leq (1/t)\nabla\phi^T(\nabla^2\phi)^{-1}\nabla\phi \leq N_\phi/t.$$

The inequality from lemma follows then by replacing the last two relations in (13). □

The main result of this section is summarized in the next theorem.

Theorem 3.1. *Under the assumptions mentioned above, $\{d(t,\lambda)\}_{t>0}$ is a strongly self-concordant family in the sense of Definition*[5] *3.1.1 in [14] with parameters $\alpha_t = M_t, \xi_t = (M_t/2)\sqrt{N_\phi/t}$ and $\eta_t = (M_t/2)\sqrt{N_\phi/t} + (1/2t)$, where M_t is defined in Lemma 3.2.*

Proof. Basically, from Definition 3.1.1 in [14] we must check three properties: self-concordance of $d(t,\lambda)$ (Lemma 3.2) and that the first and second order derivative of $d(t,\cdot)$ vary with t at a rate proportional to the derivative itself (Lemmas 3.3 and 3.4). In conclusion, the Lemmas 3.2–3.4 prove our theorem. □

It is known [14] that self-concordant families of functions can be minimized by path-following methods in polynomial time. Therefore, this type of family of augmented dual functions $\{d(t,\cdot)\}_{t>0}$ plays an important role in the algorithm of the next section.

4 Parallel implementation of an interior-point based decomposition method

In this section we develop an interior-point Lagrangian decomposition method for the separable convex problem given by (1)–(2). Theorem 3.1 is the major contribution of our paper since it allows us to effectively utilize the Newton method for tracing the trajectory of optimizers of the self-concordant family of augmented dual functions (4).

[5] Note that according to Definition 3.1.1 in [14] $\gamma_t = 1$ and $\mu_t = 1$ for our case.

4.1 An interior-point Lagrangian algorithm

The following assumptions for optimization problem (1)–(2) will be valid in this section:

Assumption 4.1. *(i) The sets X_i are compact convex sets with nonempty interior and ϕ_{X_i} are N_i-self-concordant barriers for X_i.*
(ii) Each function f_i is either linear or convex quadratic or M_{f_i}-self-concordant or X_i is a box and f_i satisfies condition (9)*.*
(iii) The matrices A_i for all $i = 1 \cdots N$ and B have full row rank and the set $\big\{\{x \in \mathbb{R}^n : A_i x_i = a_i,\ Bx = a\} \cap int(X)\big\} \neq \emptyset$.

Note that boundedness of the set X_i can be relaxed to X_i does not contain straight lines and the set of optimal solutions to problem (1)–(2) is bounded. The constraint qualification condition from Assumption 4.1 (iii) guarantees that strong duality holds for problem (1)–(2) and thus there exists a primal-dual optimal solution (x^*, λ^*).

Let us introduce the dual function:

$$\begin{aligned} d(t,\lambda) &= \max_x \{-L_t^{\text{sc}}(x,\lambda) : x_i \in \text{int}(X_i),\ A_i x_i = a_i\ \forall i = 1\cdots N\} \\ &= \langle \lambda, a\rangle + \sum_{i=1}^N \max_{x_i}\{-f_i(x_i) - t\phi_{X_i}(x_i) - \langle \lambda, B_i x_i\rangle : x_i \in \text{int}(X_i),\ A_i x_i = a_i\} \\ &= \langle \lambda, a\rangle + \sum_{i=1}^N d_i(t,\lambda). \end{aligned}$$

Note that the function $d(t,\cdot)$ can be computed in *parallel* by decomposing the original large optimization problem (1)–(2) into N independent small convex subproblems.

Lemma 4.1. *(i) The family $\{d_i(t,\cdot)\}_{t>0}$ is strongly self-concordant with the parameters $\alpha_i(t) = M_i(t), \xi_i(t) = (M_i(t)/2)\sqrt{N_i/t}$ and $\eta_i(t) = (M_i(t)/2)\sqrt{N_i/t} + (1/2t)$, where $M_i(t)$ is either $2/\sqrt{t}$ or $\max\{M_{f_i}, 2/\sqrt{t}\}$ or $2(1+\beta)/\sqrt{t}$ for all $i = 1\cdots N$.*

(ii) The family $\{d(t,\cdot)\}_{t>0}$ is strongly self-concordant with parameters $\alpha(t) = \alpha/\sqrt{t}$, $\xi(t) = \xi/t$ and $\eta(t) = \eta/t$, for some fixed positive constants α, ξ and η depending on (N_i, M_{f_i}, β).

Proof. (i) is a straightforward consequence of Assumption 4.1 and Theorem 3.1.

(ii) Note that $d(t,\lambda) = \langle\lambda, a\rangle + \sum_{i=1}^{N} d_i(t,\lambda)$. From Proposition 3.1.1 in [14] we have that the sum of strongly self-concordant family of functions is also strongly self-concordant family with the parameters: $\alpha(t) \geq \max_i\{\alpha_i(t)\}$ is a positive continuously differentiable function on $\mathbb{R}_+$, $\xi(t) = \alpha(t)\max_i\{2\xi_i(t)/\alpha_i(t)\}$ and $\eta(t) = \max_i\{\eta_i(t)\}$. Since $\alpha_i(t)$ is either $2/\sqrt{t}$ or $\max\{M_{f_i}, 2/\sqrt{t}\}$ or $2(1+\beta)/\sqrt{t}$ for all $i = 1\cdots N$, it follows that we can always choose $\alpha(t) = \alpha/\sqrt{t}$, where $\alpha = 2$ or $\alpha = 2(1+\beta)$. Similarly, we can show that there exists positive constants ξ and η depending on N_i, M_{f_i} and β such that $\xi(t) = \xi/t$ and $\eta(t) = \eta/t$. □

From Assumption 4.1 and the discussion from previous section, it follows that the optimizer of each maximization is unique and denoted by

$$x_i(t,\lambda) := \arg\max_{x_i}\{-f_i(x_i) - t\phi_{X_i}(x_i) - \langle\lambda, B_i x_i\rangle : x_i \in \mathrm{int}(X_i),\ A_i x_i = a_i\} \qquad (14)$$

and $x(t,\lambda) := [x_1(t,\lambda)^T \cdots x_N(t,\lambda)^T]^T$. It is clear that the augmented dual function $d^{\mathrm{sc}}(t,\lambda) = -d(t,\lambda)$ and let $\lambda(t) := \arg\max_{\lambda\in\mathbb{R}^m} d^{\mathrm{sc}}(t,\lambda)$, or equivalently

$$\lambda(t) = \arg\min_{\lambda\in\mathbb{R}^m} d(t,\lambda).$$

From Assumption 4.1 and the proof of Lemma 3.2 it follows that the Hessian $\nabla^2 d(t,\lambda)$ is positive definite for all $t > 0$ and $\lambda \in \mathbb{R}^m$. Hence, the dual function $d(t,\cdot)$ is strictly convex and thus $\lambda(t)$ is unique. Therefore, we can consistently define the set $\{(x(t,\lambda(t)),\lambda(t)) : t > 0\}$, called the *central path*. Let us introduce the N_ϕ-self-concordant barrier function $\phi_X(x) := \sum_{i=1}^{N}\phi_{X_i}(x_i)$ for the set X, where $N_\phi = \sum_{i=1}^{N} N_i$.

Lemma 4.2. *The central path* $\{(x(t,\lambda(t)),\lambda(t)) : t > 0\}$ *converges to the optimal solution* (x^*,λ^*) *as* $t \to 0$ *and* $\{x(t,\lambda(t)) : t > 0\}$ *is feasible for the problem* (1)–(2).

Proof. Let $x(t) := \arg\min_x\{f(x)+t\phi_X(x) : Bx = a, x_i \in \mathrm{int}(X_i), A_i x_i = a_i \ \forall i\}$, then it is known that $x(t) \to x^*$ as $t \to 0$. It is easy to see that the Hessian of $f+t\phi_X$ is positive definite and thus $f+t\phi_X$ is strictly convex and $x(t)$ is unique. From Assumption 4.1 it also follows that strong duality holds for this barrier function problem and therefore

$$\min_x\{f(x)+t\phi_X(x) : Bx = a, x_i \in \mathrm{int}(X_i), A_i x_i = a_i \ \forall i\} =$$

$$\max_\lambda \min_x\{f(x)+t\phi_X(x) + \langle \lambda, Bx-a\rangle : x_i \in \mathrm{int}(X_i), A_i x_i = a_i \ \forall i\} =$$

$$\min_x\{f(x)+t\phi_X(x) + \langle \lambda(t), Bx-a\rangle : x_i \in \mathrm{int}(X_i), A_i x_i = a_i \ \forall i\}.$$

In conclusion, $x(t) = x(t,\lambda(t))$ and thus $x(t,\lambda(t)) \to x^*$ as $t \to 0$. As a consequence it follows that $x(t,\lambda(t))$ is feasible for the original problem, i.e. $Bx(t,\lambda(t)) = a$, $A_i x_i(t,\lambda(t)) = a_i$ and $x_i(t,\lambda(t)) \in \mathrm{int}(X_i)$. It is also clear that $\lambda(t) \to \lambda^*$ as $t \to 0$. □

The next theorem describes the behavior of the central path:

Theorem 4.1. *For* $x(t) = x(t,\lambda(t))$ *the following bound holds for the central path: given any* $0 < \tau < t$ *then,*

$$f(x(t)) - f(x(\tau)) \leq N_\phi(t-\tau).$$

Proof. For any $s > 0$, $x(s) = [x_1(s)^T \cdots x_N(s)^T]^T$ satisfies the following optimality conditions (see (11) and Lemma 4.2): there exists $\nu(s) \in \mathbb{R}^{\sum_{i=1}^N m_i}$ such that

$$\nabla f(x(s)) + s\nabla\phi_X(x(s)) + B^T\lambda(t) + D_A^T\nu(s) = 0, \quad Bx(s) = a \ \text{ and } \ A_i x_i(s) = a_i.$$

It follows immediately that $\langle \nabla f(x(s)), x'(s)\rangle = -s\langle \nabla\phi_X(x(s)), x'(s)\rangle$. Since $0 < \tau < t$, then there exists $\tau \leq s \leq t$ such that

$$f(x(t)) - f(x(\tau)) = (t-\tau)\langle \nabla f(x(s)), x'(s)\rangle = -s(t-\tau)\langle \nabla \phi_X(x(s)), x'(s)\rangle.$$

From (6) we have that

$$-\langle \nabla \phi_X(x(s)), x'(s)\rangle \le \big(N_\phi \; x'(s)^T \nabla^2 \phi_X(x(s)) x'(s)\big)^{1/2}.$$

Using a similar reasoning as in Lemma 3.3 we have:

$$x'(s) = -[H^{-1}(s) - H^{-1}(s) D_A^T (D_A H^{-1}(s) D_A^T)^{-1} D_A H^{-1}(s)] \nabla \phi_X(x(s)),$$

where we denote with $H(s) = \nabla^2 f(x(s)) + s\nabla^2 \phi_X(x(s))$. Using this expression for $x'(s)$ and since $0 \prec \nabla^2 \phi_X(x(s)) \preceq 1/sH(s)$ and $H^{-1}(s) \preceq 1/s\big(\nabla^2\phi_X(x(s))\big)^{-1}$ we obtain:

$$\begin{aligned} x'(s)^T \nabla^2 \phi_X(x(s)) x'(s) &\le (1/s) \nabla \phi_X(x(s))^T H^{-1}(s) \nabla \phi_X(x(s)) \\ &\le (1/s^2) \nabla \phi_X(x(s))^T \big(\nabla^2 \phi_X(x(s))\big)^{-1} \nabla \phi_X(x(s)) \le N_\phi / s^2. \end{aligned}$$

It follows immediately that $f(x(t)) - f(x(\tau)) \le N_\phi(t-\tau)$. □

A simple consequence of Theorem 4.1 is that the following bounds on the approximation of the optimal value function f^* hold:

$$0 \le f(x(t)) - f^* \le tN_\phi.$$

Indeed, from Lemma 4.2 we know that $\{x(t,\lambda(t)) : t > 0\}$ is feasible for the original problem (1)–(2). Since $x(t) = x(t,\lambda(t))$, it follows that $f(x(t)) \ge f^*$. It remains to show the upper bound. However, taking the limit as $\tau \to 0$ in Theorem 4.1 and using Lemma 4.2 we obtain also the upper bound. This upper bound gives us a stopping criterion in the algorithm that we derive below: if ϵ is the required accuracy for the approximation of f^*, then for any $t_f \le \epsilon/N_\phi$ we have that $x(t_f)$ is an ϵ-approximation of the optimum, i.e. $x(t_f)$ is feasible for problem (1)–(2) and $f(x(t_f)) - f(x^*) \le \epsilon$. Although $\lambda(t)$ is the minimizer of the dual function $d(t,\cdot)$ over $\mathbb{R}^m$, so various unconstrained minimization techniques (e.g. Newton, quasi-Newton and conjugate gradient methods) can be used to approximate $\lambda(t)$, our goal is to trace the central path $\{(x(t, \lambda(t)), \lambda(t)) : t > 0\}$ utilizing Newton method for the self-concordant family $\{d(t, \cdot)\}_{t>0}$.

It is easy to see that the gradient of the self-concordant function $d(t, \cdot)$ is given by

$$\nabla d(t, \lambda) = a + \sum_{i=1}^{N} \nabla d_i(t, \lambda) = a - \sum_{i=1}^{N} B_i x_i(t, \lambda) = a - Bx(t, \lambda).$$

For every (t, λ) let us define the positive definite matrix

$$H_i(t, \lambda) := \nabla^2 f_i(x_i(t, \lambda)) + t\nabla^2 \phi_{X_i}(x_i(t, \lambda)).$$

The Hessian of function $d_i(t, \cdot)$ is positive definite and from (12) it has the form

$$\nabla^2 d_i(t, \lambda) = B_i[H_i(t, \lambda)^{-1} - H_i(t, \lambda)^{-1} A_i^T \left(A_i H_i(t, \lambda)^{-1} A_i^T\right)^{-1} A_i H_i(t, \lambda)^{-1}] B_i^T.$$

In conclusion, the Hessian of dual function $d(t, \cdot)$ is also positive definite and given by:

$$\nabla^2 d(t, \lambda) = \sum_{i=1}^{N} \nabla^2 d_i(t, \lambda).$$

Denote the Newton direction associated to self-concordant function $d(t, \cdot)$ at λ with

$$\Delta\lambda(t, \lambda) := -\left(\nabla^2 d(t, \lambda)\right)^{-1} \nabla d(t, \lambda).$$

For every $t > 0$, we define the Newton decrement of the function $d(t, \cdot)$ at λ as:

$$\delta(t, \lambda) := \alpha(t)/2\sqrt{\nabla d(t, \lambda)^T \left(\nabla^2 d(t, \lambda)\right)^{-1} \nabla d(t, \lambda)}.$$

Note that $\delta(t, \hat{\lambda}) = 0$ if and only if $\hat{\lambda} = \lambda(t)$ (recall that $\lambda(t) = \arg\min_{\lambda \in \mathbb{R}^m} d(t, \lambda)$).

Algorithm 4.1. *(initialization of a path-following algorithm)*

```
0. input $t_0 > 0$, $\lambda_0 \in \mathbb{R}^m$, $\epsilon_V > 0$ and $r = 0$
1. compute $x_i^r = x_i(t_0, \lambda_r)$ $\forall i$ and $\delta_r = \delta(t_0, \lambda_r)$; if $\delta_r \leq \epsilon_V$, then $r_f = r$ and go to step 3
```

```
2. determine a step size $\sigma$ and compute Newton iterate: $\lambda_{r+1} = \lambda_r + \sigma \Delta\lambda(t_0, \lambda_r)$;
   replace $r$ by $r + 1$ and go to step 1
3. **output** $(t^0, \lambda^0) = (t_0, \lambda_{r_f})$.
```

Note that Algorithm 4.1 approximates the optimal Lagrange multiplier $\lambda(t_0)$ of the dual function $d(t_0, \cdot)$, i.e. the sequence (t_0, λ_r) moves into the neighborhood $V(t, \epsilon_V) = \{(t, \lambda) : \delta(t, \lambda) \leq \epsilon_V\}$ of the trajectory $\{(t, \lambda(t)) : t > 0\}$.

Algorithm 4.2. *(a path-following algorithm)*

```
0. input: $(t^0, \lambda^0)$ satisfying $\delta(t^0, \lambda^0) \leq \epsilon_V$ , $k = 0$, $0 < \tau < 1$ and $\epsilon > 0$
1. if $t^k N_\phi \leq \epsilon$, then $k_f = k$ and go to step 5
2. (outer iteration) let $t^{k+1} = \tau t^k$ and go to inner iteration (step 3)
3. (inner iteration) initialize $\lambda = \lambda^k$, $t = t^{k+1}$ and $\delta = \delta(t^{k+1}, \lambda^k)$
while $\delta > \epsilon_V$ do
    3.1 compute $x_i = x_i(t, \lambda)$ $\forall i$, determine a step size $\sigma$ and compute $\lambda^+ = \lambda + \sigma \Delta\lambda(t, \lambda)$
    3.2 compute $\delta^+ = \delta(t, \lambda^+)$ and update $\lambda = \lambda^+$ and $\delta = \delta^+$
4. $\lambda^{k+1} = \lambda$ and $x_i^{k+1} = x_i$; replace $k$ by $k + 1$ and go to step 1
5. output: $(x_1^{k_f}, \cdots, x_N^{k_f}, \lambda^{k_f})$.
```

In Algorithm 4.2 we trace numerically the trajectory $\{(t, \lambda(t)) : t > 0\}$ from a given initial point (t^0, λ^0) close to this trajectory. The sequence $\{(x_1^k, \cdots, x_N^k, \lambda^k)\}_{k>0}$ lies in a neighborhood of the central path and each limit point of this sequence is primal-dual optimal. Indeed, since $t^{k+1} = \tau t^k$ with $\tau < 1$, it follows that $\lim_{k\to\infty} t^k = 0$ and using Theorem 4.1 the convergence of the sequence $x^k = [(x_1^k)^T \cdots (x_N^k)^T]^T$ to x^* is obvious.

The step size σ in the previous algorithms is defined by some line search rule. There are many strategies for choosing τ. Usually, τ can be chosen independent of the problem (long step methods), e.g. $\tau = 0.5$, or depends on the problem (short step methods). The choice for τ is crucial for the performance of the algorithm. An example is that in practice long step interior-point algorithms are more efficient than short step interior-point algorithms. However, short step type algorithms have better worst-case complexity iteration bounds than long step algorithms. In the sequel we derive a theoretical strategy to update the barrier parameter τ which follows from the theory described in [14] and consequently complexity bounds for short step updates. Complexity iteration bounds for long step updates can also be derived using the same theory (see Section 3.2.6 in [14]). The next lemma estimates the reduction of the dual function at each iteration.

Lemma 4.3. *For any $t > 0$ and $\lambda \in \mathbb{R}^m$, let $\Delta\lambda = \Delta\lambda(t, \lambda)$ be the Newton direction as defined above. Let also $\delta = \delta(t, \lambda)$ be the Newton decrement and $\delta_* = 2 - \sqrt{3}$.*

(i) If $\delta > \delta_*$, then defining the step length $\sigma = 1/(1 + \delta)$ and the Newton iterate $\lambda^+ = \lambda + \sigma\Delta\lambda$ we have the following decrease in the objective function $d(t, \cdot)$

$$d(t, \lambda^+) - d(t, \lambda) \leq -(4t/\alpha^2)(\delta - \ln(1 + \delta)).$$

(ii) If $\delta \leq \delta_*$, then defining the Newton iterate $\lambda^+ = \lambda + \Delta\lambda$ we have

$$\delta(t, \lambda^+) \leq \delta^2/(1 - \delta)^2 \leq \delta/2, \quad d(t, \lambda) - d(t, \lambda(t)) \leq (16t/\alpha^2)\delta.$$

(iii) If $\delta \leq \delta_*/2$, then defining $t^+ = \frac{2c}{2c+1}t$, where $c = 1/4 + 2\xi/\delta_* + \eta$, we have

$$\delta(t^+, \lambda) \leq \delta_*.$$

Proof. (i) and (ii) follow from Theorem 2.2.3 in [14] and Lemma 4.1 from above.

(iii) is based on the result of Theorem 3.1.1 in [14]. In order to apply this theorem, we first write the metric defined by (3.1.4) in [14] for our problem: given $0 < t^+ < t$ and using Lemma 4.1 we obtain

$$\rho_{\delta_*/2}(t, t^+) = (1/4 + 2\xi/\delta_* + \eta)\ln(t/t^+).$$

Since $\delta \leq \delta_*/2 < \delta_*$ and since for $t^+ = \frac{2c}{2c+1}t$, where c is defined above, one can verify that $\rho_{\delta_*/2}(t, t^+) = c\ln(1 + 1/2c) \leq 1/2 \leq 1 - \delta/\delta_*$, i.e. t^+ satisfies the condition (3.1.5) of Theorem 3.1.1 in [14], it follows that $\delta(t^+, \lambda) \leq \delta_*$. □

Define the following step size: $\sigma(\delta) = 1/(1+\delta)$ if $\delta > \delta_*$ and $\sigma(\delta) = 1$ if $\delta \leq \delta_*$. With Algorithm 4.1 for a given t^0 and $\epsilon_V = \delta_*/2$, we can find (t^0, λ^0) satisfying $\delta(t^0, \lambda^0) \leq \delta_*/2$ using the step size $\sigma(\delta)$ (see previous lemma). Based on the analysis given in Lemma 4.3 it follows that taking in Algorithm 4.2 $\epsilon_V = \delta_*/2$ and $\tau = 2c/(2c + 1)$, then the inner iteration stage (step 3) reduces to only one iteration:

3. compute $\lambda^{k+1} = \lambda^k + \Delta\lambda(t^{k+1}, \lambda^k)$.

However, the number of outer iterations is larger than in the case of long step algorithms.

4.2 Practical implementation

In this section we discuss the practical implementation of our algorithm and we give some estimates of the complexity for it. Among the assumptions considered until now in the paper the most stringent one seems to be the one requiring to solve exactly the maximization problems (14), i.e. the exact computation of the maximizers $x_i(t, \lambda)$. Note that the gradient and the Hessian of $d(t, \cdot)$ at λ depends on $x_i(t, \lambda)$'s. When $x_i(t, \lambda)$'s are computed approximatively, the expressions for the gradient and Hessian derived in the previous section for $d(t, \cdot)$ at λ are not the true gradient and Hessian of $d(t, \cdot)$ at this point. In simulations we considered the following criterion: find $\tilde{x}_i(t, \lambda) \in \text{int}(X_i)$

and $\tilde{\nu}_i(t,\lambda) \in \mathbb{R}^{m_i}$ such that $A_i \tilde{x}_i(t,\lambda) = a_i$ and the following condition holds

$$\|\nabla f_i(\tilde{x}_i(t,\lambda)) + t\nabla\phi_{X_i}(\tilde{x}_i(t,\lambda)) + B_i^T\lambda + A_i^T\tilde{\nu}_i(t,\lambda)\| \leq t\epsilon_x,$$

for some $\epsilon_x > 0$. Note however that even when such approximations are considered, the vector $\Delta\lambda$ still defines a search direction in the λ-space. Moreover, the cost of computing an extremely accurate maximizer of (14) as compared to the cost of computing a good maximizer of (14) is only marginally more, i.e. a few Newton steps at most (due to quadratic convergence of the Newton method close to the solution). Therefore, it is not unreasonable to assume even exact computations in the proposed algorithms.

In the rest of this section we discuss the complexity of our method and parallel implementations for solving efficiently the Newton direction $\Delta\lambda$. At each iteration of the algorithms we need to solve basically a linear system of the following form:

$$\big(\sum_{i=1}^{n} G_i\big)\Delta\lambda = g, \tag{15}$$

where $G_i = B_i[H_i^{-1} - H_i^{-1}A_i^T\left(A_iH_i^{-1}A_i^T\right)^{-1}A_iH_i^{-1}]B_i^T$, the positive definite matrix H_i denotes the Hessian of $f_i + t\phi_{X_i}$ and some appropriate vector g. In order to obtain the matrices H_i we can solve in parallel N small convex optimization problems of the form (14) by Newton method, each one of dimension n_i and with self-concordant objective function. The cost to solve each subproblem (14) by Newton method is $\mathcal{O}(n_i^3(n_\lambda + \ln\ln 1/t\epsilon_x))$, where n_λ denotes the number of Newton iterations before the iterates x_i reaches the quadratic convergence region (it depends on the update λ) and $t\epsilon_x$ is the required accuracy for the approximation of (14). Note that using the Newton method for solving (14) we automatically obtain also the expression for H_i^{-1} and $A_iH_i^{-1}A_i^T$. Assuming that a Cholesky factorization for $A_iH_i^{-1}A_i^T$ is used to solve the Newton system corresponding to the optimization subproblem (14), then this factorization can also be used to compute in parallel the matrix of the linear system (15). Finally, we can use a Cholesky factorization of this matrix and then forward and backward substitution to obtain the Newton direction $\Delta\lambda$. In conclusion, we can compute the Newton direction $\Delta\lambda$ in $\mathcal{O}(\sum_{i=1}^{N} n_i^3)$ arithmetic operations.

Note however that in many applications the matrices H_i, A_i and B_i are very sparse and have special structures. For example in network optimization (see Section 5.2 for more details) the H_i's are diagonal matrices, B_i's are the identity matrices and the matrices A_i's are the same for all i (see (17)), i.e. $A_i = A$. In this case the Cholesky factorization of $AH_i^{-1}A^T$ can be done very efficiently since the sparsity pattern of those matrices is the same in all iterations and coincides with the sparsity pattern of AA^T, so the *analyse* phase has to be done only once, before optimization.

For large problem instances we can also solve the linear system (15) approximately using a preconditioned conjugate gradient algorithm. There are different techniques to construct a good preconditioner and they are spread across optimization literature. Detailed simulations for the method proposed in this paper and comparison of different techniques to solve the Newton system (15) will be given elsewhere.

Let us also note that the number of Newton iterations performed in Algorithm 4.1 can be determined via Lemma 4.3 (i). Moreover, if in Algorithm 4.2 we choose $\epsilon_V = \delta_*/2$ and $\tau = 2c/(2c+1)$ we need only one Newton iteration at the inner stage. It follows that for this particular choice for ϵ_V and τ the total number of Newton iterations of the algorithm is given by the number of outer iterations, i.e. the algorithm terminates in polynomial-time, within $\mathcal{O}\big(\frac{1}{\ln(\tau^{-1})} \ln(t^0/\epsilon)\big)$ iterations. This choice is made only for a worst-case complexity analysis. In a practical implementation one may choose larger values using heuristic considerations.

5 Applications with separable structure

In this section we briefly discuss some of the applications to which our method can be applied: distributed model predictive control and network optimization.

5.1 Distributed model predictive control

A first application that we will discuss here is the control of large-scale systems with interacting subsystem dynamics. A distributed model predictive control (MPC) framework is appealing in this context since this framework allows us to design local subsystem-base controllers that take care of the interactions between different subsystems and physical constraints. We assume that the overall system model can be decomposed into N appropriate subsystem models:

$$x^i(k+1) = \sum_{j \in \mathcal{N}(i)} A_{ij} x^j(k) + B_{ij} u^j(k) \ \ \forall i = 1 \cdots N,$$

where $\mathcal{N}(i)$ denotes the set of subsystems that interact with the ith subsystem, including itself. The control and state sequence must satisfy local constraints: $x^i(k) \in \Omega_i$ and $u^i(k) \in U_i$ for all i and $k \geq 0$, where the sets Ω_i and U_i are usually convex compact sets with the origin in their interior (in general box constraints). Performance is expressed via a stage cost, which we assume to have the following form [3]: $\sum_{i=1}^{N} \ell_i(x^i, u^i)$, where usually ℓ_i is a convex quadratic function, but not strictly convex in (x^i, u^i). Let N_p denote the prediction horizon. In MPC we must solve at each step k, given $x^i(k) = x^i$, an optimal control problem of the following form [15]:

$$\min_{x_l^i, u_l^i} \Big\{ \sum_{l=0}^{N_p-1} \sum_{i=1}^{N} \ell_i(x_l^i, u_l^i) : x_0^i = x^i, \ x_{l+1}^i = \sum_{j \in \mathcal{N}(i)} A_{ij} x_l^j + B_{ij} u_l^j, \ x_l^i \in \Omega_i, \ u_l^i \in U_i \ \forall l, i \Big\}. \quad (16)$$

A similar formulation of distributed MPC for coupled linear subsystems with decoupled costs was given in [3], but without state constraints. In [3], the authors proposed to solve the optimization problem (16) in a decentralized fashion, using the Jacobi algorithm [5]. But, there is no theoretical guarantee of the Jacobi algorithm about how good the approximation to the optimum is after a number of iterations and moreover one needs strictly convex functions f_i to prove asymptotic convergence to the optimum.

Let us introduce $\mathbf{x}_i = [x_0^i \cdots x_N^i \; u_0^i \cdots u_{N-1}^i], X_i = \Omega_i^{N+1} \times U_i^N$ and the self-concordant functions $f_i(\mathbf{x}^i) = \sum_{l=0}^{N_p-1} \ell_i(x_l^i, u_l^i)$, since ℓ_i are assumed to be convex quadratic. The control problem (16) can be recast then as a separable convex program (1)–(2), where the matrices A_i's and B_i's are defined appropriately, depending on the structure of the matrices A_{ij} and B_{ij}. In conclusion, Assumption 4.1 holds for this control problem so that our decomposition method can be applied.

5.2 Network optimization

Network optimization furnishes another area in which our algorithm leads to a new method of solution. The optimization problem for routing in telecommunication data networks has the following form [1, 6]:

$$\min_{x_i \in [0,\ \bar{x}_i], y_j \in [0,\ d_j]} \Big\{ \sum_{j=1}^{n} f_j(y_j) + \sum_{i=1}^{N} \langle c_i, x_i \rangle : A x_i = a_i,\ \sum_{i=1}^{N} x_i = y \Big\}, \tag{17}$$

where we consider a multicommodity flow model with N commodities and n links. The matrix $A \in \mathbb{R}^{m \times n}$ is the node-link incidence matrix representing the network topology. One of the most common cost functions used in the communication network literature is the total delay function [1, 6]: $f_j(y_j) = \frac{y_j}{d_j - y_j}$.

Corollary 5.1. *Each function $f_j \in \mathcal{C}^3\big([0,\ d_j)\big)$ is convex and f_j is 3-compatible with the self-concordant barrier $\phi_j(y_j) = -\ln(y_j(d_j - y_j))$ on the interval $(0,\ d_j)$.*

			DIP		ADI	
m_1	n_1	N	CPU	fct. eval.	CPU	fct. eval.
20	50	10	7.85	58	61.51	283
25	50	20	16.88	82	145.11	507
50	150	50	209.91	185	4621.42	1451
80	250	100	1679.81	255	16548.23	1748
170	500	100	10269.12	367	*	*
20	50	30	19.02	95	182.27	542
40	100	40	143.7	152	3043.67	1321
60	150	50	229.32	217	10125.42	2546
90	250	100	2046.09	325	32940.67	3816
100	300	120	4970.52	418	*	*

Proof. Note that the inequality (9) holds for all $y_j \in (0,\ d_j)$ and $h \in \mathbb{R}$. Indeed,

$$|\nabla^3 f_j(y_j)| = 3\nabla^2 f_j(y_j)\sqrt{1/(d_j - y_j)^2} \leq 3\nabla^2 f_j(y_j)\sqrt{1/(d_j - y_j)^2 + 1/y_j^2}.$$

Therefore, we can solve this network optimization problem with our method. Note that the standard dual function d_0 is not differentiable since it is the sum of a differentiable function (corresponding to the variable y) and a polyhedral function (corresponding to the variable x). In [6] a bundle-type algorithm is developed for maximizing the nonsmooth function d_0, in [1] the dual subgradient method is applied for maximizing d_0, while in [5, 9] alternating direction methods were proposed.

5.3 Numerical results

We illustrate the efficiency of our method on a random set of problems of the form (17) and (16), i.e. with total delay (first half table) and quadratic (second half) objective function, respectively. Here, the sets X_i are assumed to have the form $[0,\ u_i] \in \mathbb{R}^{n_1}$, i.e. $n_i = n_1$ and also $m_i = m_1$ for all i. In the table we display the CPU time (seconds) and the number of calls of the dual function for our dual interior-point algorithm (DIP) and an algorithm based on alternating direction method [9] (ADI) for different values of m, n, N and fixed accuracy $\epsilon = 10^{-4}$. The code is implemented in Matlab on a Linux operating system. The computational time can be considerably reduced, e.g. by treating sparsity using more efficient techniques as explained in Section 4.2. For two problems the ADI algorithm did not produce the result after running one day.

6 Conclusions

A new decomposition method in convex programming is developed in this paper using dual decomposition and interior-point framework. Our method combines the fast local convergence rates of the Newton method with the efficiency of structural optimization for solving separable convex programs. Although our algorithm resembles augmented Lagrangian methods, it differs both in the computational steps and in the choice of the parameters. Contrary to most augmented Lagrangian methods that use gradient based directions to update the Lagrange multipliers, our method uses Newton directions and thus the convergence rate of the proposed method is faster. The reason for this lies in the fact that by adding self-concordant barrier terms to the standard Lagrangian we proved that under appropriate conditions the corresponding family of augmented dual functions is also self-concordant. Another appealing theoretical advantage of our interior-point Lagrangian decomposition method is that it is fully automatic, i.e. the parameters of the scheme are chosen as in the path-following methods, which are crucial for justifying its global convergence and polynomial-time complexity.

References

[1] Xiao, L., Johansson, M., and Boyd, S., *Simultaneous routing and resource allocation via dual decomposition*, IEEE Transactions on Communications, Vol. 52, No. 7, pp. 1136–1144, 2004.

[2] Gondzio, J., and Sarkissian, R., *Parallel interior point solver for structured linear programs*, Mathematical Programming, Vol. 96, pp. 561–584, 2003.

[3] Venkat, A., Hiskens, I., Rawlings, J., and Wright, S., *Distributed MPC strategies with application to power system automatic generation control*, IEEE Transactions on Control Systems Technology, to appear, 2007.

[4] Zhao, G., *A lagrangian dual method with self-concordant barriers for multi-stage stochastic convex programming*, Mathematical Programming, Vol. 102, pp. 1–24, 2005.

[5] Bertsekas, D. P., and Tsitsiklis, J. N., *Parallel and distributed computation: Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[6] Lemarechal, C., Ouorou, A., and Petrou, G., *A bundle-type algorithm for routing in telecommunication data networks*, Computational Optimization and Applications, to appear, 2006.

[7] Miele, A., Moseley, P. E., Levy, A. V., and Coggins, G. M, *On the method of multipliers for mathematical programming problems*, Journal of Optimization Theory and Applications, Vol. 10, pp. 1–33, 1972.

[8] Tseng, P., *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM Journal of Control and Optimization, Vol. 29, No. 1, pp. 119–138, 1991.

[9] Kontogiorgis, S., De Leone, R., and Meyer, R., *Alternating direction splittings for block angular parallel optimization*, Journal of Optimization Theory and Applications, Vol. 90, No. 1, pp. 1–29, 1996.

[10] Chen, G. and Teboulle, M., *A proximal-based decomposition method for convex minimization problems*, Mathematical Programming, Vol 64, pp. 81–101, 1994.

[11] Mahey, P., Oualibouch, S., and Tao, P., *Proximal decomposition on the graph of a maximal monotone operator*, SIAM Journal of Optimization, Vol. 5, pp. 454–466, 1995.

[12] Necoara, I. and Suykens, J. A. K., Application of a smoothing technique to decomposition in convex optimization, Dept. of Electrical Engineering Report No. 06–08 (submitted for publication), Katholieke Universiteit Leuven, 2008.

[13] Kojima, M., Megiddo, N., Mizuno, S., and Shindoh, S., *Horizontal and vertical decomposition in interior point methods for linear programs* Dept. of Mathematical and Computing Sciences Technical report, Tokyo Institute of Technology, 1993.

[14] Nesterov, Y., and Nemirovskii, A., *Interior Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics (SIAM Studies in Applied Mathematics), Philadelphia, 1994.

[15] Mayne D. Q., Rawlings J. B., Rao C. V., and Scokaert P. O. M., *Constrained Model Predictive Control: Stability and Optimality*, Automatica, Vol. 36, No. 7, pp. 789–814, 2000.