An Efficient Algorithm for Minimizing a Sum of P-Norms

Guoliang Xue* Yinyu Ye†

September 5, 1997

Abstract

We study the problem of minimizing a sum of p-norms where p is a fixed real number in the interval $[1, \infty]$. Several practical algorithms have been proposed to solve this problem. However, none of them has a known polynomial time complexity. In this paper, we transform the problem into standard conic form. Unlike those in most convex optimization problems, the cone for the p-norm problem is *not* self-dual unless $p = 2$. Nevertheless, we are able to construct two logarithmically homogeneous self-concordant barrier functions for this problem. The barrier parameter of the first barrier function does not depend on p. The barrier parameter of the second barrier function increases with p. Using both barrier functions, we present a primal-dual potential reduction algorithm to compute an ϵ-optimal solution in polynomial time that is independent of p. Computational experiences of a Matlab implementation are also reported.

Key words. Shortest network, Steiner minimum trees, facilities location, polynomial time algorithm, minimizing a sum of norms.

AMS subject classifications. 68Q20, 68Q25, 90C25, 90C35.

1 Introduction

Let $c_1, c_2, \ldots, c_m \in R^d$ be column vectors in the Euclidean d-space and $A_1, A_2, \ldots, A_m \in R^{n\times d}$ be n-by-d matrices each having full column rank. We want to find a point $u \in R^n$ such that the

*Department of Computer Science, The University of Vermont, Burlington, VT 05405-0156. Email: xue@cs.uvm.edu. The research of this author was supported in part by ARO grant DAAH04-96-1-0233 and by NSF grants ASC-9409285 and OSR-9350540.

†Department of Management Sciences, The University of Iowa, Iowa City, IA 52242-1000. Email: yyye@dollar.biz.uiowa.edu. The research of this author was supported in part by NSF Grants DMI-9522507, DMS-9703490, and OSR-9350540.

following sum of p-norms, $p \geq 1$, is minimized.

$$\begin{array}{ll} \min & \sum_{i=1}^{m} ||c_i - A_i^T u||_p \\ \text{s.t.} & u \in R^n \end{array} \tag{1.1}$$

We recall that $|| \bullet ||_p$ is the *Hölder* or *p-norm* [14] defined by

$$||x||_p = (|x_1|^p + \cdots + |x_n|^p)^{1/p}. \tag{1.2}$$

It is clear that $u = 0$ is an optimal solution to (1.1) when all of the c_i are zero. Therefore, we will assume in the rest of this paper that not all of the c_i are zero. Problem (1.1) is a convex programming problem, but its complexity is not explicitly known to this date.

Problem (1.1) has a long history, dating back to the 17th century, when Fermat [20] studied the problem of finding the shortest network interconnecting 3 points on the Euclidean plane where a fourth point may be added to minimize the sum of lengths of the network. The Fermat problem was generalized to the Euclidean facilities location problem and the Euclidean Steiner minimal tree problem. Those problems were in turn further generalized to cases where distances are measured using weighted p-norms.

The facilities location problem is one of locating N new facilities with respect to M existing facilities, the locations of which are known. The problem consists of finding locations of new facilities which will minimize the sum of (non-negatively) weighted distances between the new and existing facilities, and between the new facilities. If there is only one new facility ($N = 1$), the problem is called a single facility location (SFL) problem. If there are more than one new facilities ($N \geq 2$), the problem is called a multi-facility location (MFL) problem.

For the general SFL problem with Euclidean norm, Weiszfeld [33] gave a simple closed form iterative algorithm in 1937. This work started a chain of research on this topic [20, 27, 32, 5, 6, 21]. Miehle [24] was the first to propose an extension of the Weiszfeld algorithm for SFL to solve MFL problems. Again, a number of important results were obtained along this line [27, 29, 34, 10, 30]. For more details on location problems, see the books by Francis, McGinnis and White [12] and by Love, Morris and Wesolowsky [22].

Practical algorithms for solving these problems began with the work of Calamai and Conn [3, 4] and Overton [28], where they proposed projected Newton algorithms with quadratic rate of convergence. They also generalized the location problems to one of minimizing a sum of norms. In recent years, several complexity results and numerically stable algorithms have been obtained for problem (1.1) with $p = 2$ using techniques of interior-point algorithms. In [35], Xue et al. showed that the dual of the Euclidean MFL problem is the minimization of a linear function subject to linear and convex quadratic constraints and therefore can be solved using interior-point techniques in polynomial time. More recently, Andersen [1] used the HAP idea [10] to smooth the objective function by introducing a perturbation $\epsilon > 0$ and applied a Newton barrier method to solving the problem. Andersen and Christiansen [2] and Conn and Overton [7] also proposed a primal-dual method based on the ϵ-perturbation and presented impressive computational results.

For $p \geq 1$, den Hertog et al. [8, 9] (and see references therein) presented a polynomial-time interior-point Newton barrier method for solving a closely related problem, where the objective function is

$$\sum_{i=1}^{m} ||c_i - A_i^T u||_p^p,$$

that is, the sum of the p-powers of p-norms. This objective function is equivalent to ours only if $m = 1$.

In a recent paper [36], we studied the problem of minimizing a sum of Euclidean norms. The problem was formulated in standard conic form and a polynomial time primal-dual potential reduction algorithm was presented. A nice property of the Euclidean case is that the second order cone is self-dual. We have taken advantage of this fact in our work in [36], based on the self-dual theory and primal-dual algorithm developed by Nesterov and Todd [26] (also see Kojima [18]).

In some applications, the problems are better modeled with p-norms where $p \neq 2$. For example, in VLSI design, the 1-norm or Manhattan distance is used. in transportation, several general p-norms are used [22]. Since the p-order cone of the conic form of (1.1) is *not* self-dual unless $p = 2$, a natural question to ask is the following: *Can the techniques used for minimizing a sum of Euclidean norms be generalized to minimize a sum of p-norms in polynomial time?*. The goal of this paper is to answer the above question affirmatively. Specifically, we present a primal-dual potential reduction algorithm which computes an ϵ-optimal solution in at most

$$O(\sqrt{md}(\log(\bar{c}/\epsilon) + \log(md)))$$

iterations, where $\bar{c} = \max_{1 \leq i \leq m} ||c_i||$. Note that this bound is independent of p and is increased only by a factor $\sqrt{d}$ comparing to the bound for $p = 2$.

The rest of this paper is organized as follows. In section 2, the basic problem (1.1) is transformed into a standard convex programming problem in conic form. In section 3, we develop two logarithmically homogeneous self-concordant barrier functions for this problem. In section 4, we present a primal-dual potential reduction algorithm for solving the problem. In section 5, we discuss the computational complexity and simplifications of the potential reduction algorithm. In section 6, we present applications to the SFL problem, the MFL problem, and the SMT problem. In section 7, we present some computational examples on Steiner minimal tree problems. We conclude this paper in section 8.

2 Conic formulation

We will call problem (1.1) the *basic problem* in the rest of our paper. This problem can be formulated as the maximization of a linear function subject to affine and convex cone constraints as follows.

$$\begin{array}{ll} \max & -\sum_{i=1}^m t_i \\ \text{s.t.} & t_1 \geq ||c_1 - A_1^T u||_p, \\ & t_2 \geq ||c_2 - A_2^T u||_p, \\ & \vdots \\ & t_m \geq ||c_m - A_m^T u||_p, \end{array} \tag{2.1}$$

where $t_i \in R, \quad i = 1, 2, \ldots, m.$

In the rest of this paper, when we represent a large matrix with several small matrices, we will use semi-colon ";" for column concatenation and use comma "," for row concatenation. This notation also applies to vectors. We will use 0 to represent a column vector whose elements are all zero, and e the vector of all ones.

In this section, we will transform our basic problem (1.1) into a standard convex programming problem in conic form, where the cone and its associated barrier are *not self-dual* unless $p = 2$.

Consider the p-order cone, where $p \geq 1$,

$$K = \left\{ (t \in R, s \in R^d) : \ t^p \geq \sum_{j=1}^d |s_j|^p \right\}.$$

Then its interior is

$$\text{int}K := \{(t \in R, s \in R^d) : \ t > \|s\|_p\}.$$

The dual of K is

$$K^* = \left\{ (\tau \in R, x \in R^d) : \ \tau^q \geq \sum_{j=1}^d |x_j|^q \right\},$$

where $\frac{1}{p} + \frac{1}{q} = 1$.

Now let

$$\mathcal{B} = \begin{pmatrix} -1 \\ -1 \\ \vdots \\ -1 \\ 0_n \end{pmatrix} \in R^{m+n}, \quad \mathcal{C} = \begin{pmatrix} (0; c_1) \\ (0; c_2) \\ \vdots \\ (0; c_m) \end{pmatrix} \in R^{m+md},$$

and

$$\mathcal{A}^T = \begin{pmatrix} -1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & A_1^T \\ 0 & -1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & A_2^T \\ & & \ddots & & \\ 0 & 0 & \cdots & -1 & 0 \\ 0 & 0 & \cdots & 0 & A_m^T \end{pmatrix} \in R^{(m+md)\times(m+n)}.$$

Then, problem (1.1) or (2.1) can be written in the standard (dual) form

$$\begin{array}{ll} \max & \mathcal{B}^T(t_1; t_2; \cdots; t_m; u) \\ \text{s.t.} & \begin{pmatrix} (t_1; s_1) \\ (t_2; s_2) \\ \vdots \\ (t_m; s_m) \end{pmatrix} = \mathcal{C} - \mathcal{A}^T(t_1; t_2; \cdots; t_m; u) \\ & (t_i; s_i) \in K, \ i = 1, 2, \ldots, m. \end{array} \tag{2.2}$$

Let $(\tau_1; x_1), (\tau_2; x_2), \ldots, (\tau_m; x_m) \in R^{d+1}$. Then its corresponding primal problem is

$$\begin{array}{ll} \min & \mathcal{C}^T((\tau_1; x_1); (\tau_2; x_2); \cdots; (\tau_m; x_m)) \\ \text{s.t.} & \mathcal{A}((\tau_1; x_1); (\tau_2; x_2); \cdots; (\tau_m; x_m)) = \mathcal{B} \\ & (\tau_i; x_i) \in K^*, \ i = 1, 2, \ldots, m. \end{array} \tag{2.3}$$

Thus, using $\mathcal{S} := ((t_1; s_1); (t_2; s_2); \cdots; (t_m; s_m))$, $\mathcal{Y} := (t_1; t_2; \cdots; t_m; u)$, $\mathcal{X} := ((\tau_1; x_1); (\tau_2; x_2); \cdots; (\tau_m; x_m))$, $\mathcal{K} := K^m := K \times K \times \cdots \times K$ and $\mathcal{K}^* := (K^*)^m := K^* \times K^* \times \cdots \times K^*$, we can write the two problems (2.3) and (2.2) as

$$(P) \qquad \begin{array}{ll} \min & \mathcal{C}^T \mathcal{X} \\ \text{s.t.} & \mathcal{A}\mathcal{X} = \mathcal{B} \\ & \mathcal{X} \in \mathcal{K}^* \end{array}$$

and

$$(D) \qquad \begin{array}{ll} \max & \mathcal{B}^T\mathcal{Y} \\ \text{s.t.} & \mathcal{S} = \mathcal{C} - \mathcal{A}^T\mathcal{Y} \\ & \mathcal{S} \in \mathcal{K}. \end{array}$$

This is the pair of problems (P) and (D) in Nesterov and Nemirovskii [25]. Unlike those in most optimization problems, $\mathcal{K} \neq \mathcal{K}^*$ (unless $p = 2$) here. Therefore, the symmetric primal-dual techniques, developed by, e.g., Nesterov and Todd [26], are no longer applicable. However, we can still use interior-point algorithms to compute an ϵ-optimal solution of the problem in polynomial time.

3 Barrier functions for p-order cones

The key to solve problems (P) and (D) is to construct a simple and efficient barrier function for K, when $p \neq 2$. This has been an open question. In this section we present two barrier functions and analyze their barrier parameters.

3.1 Barrier function I

One barrier function is constructed from the following convex set

$$G_p = \{(s \in R^d, z \in R^d) : \ z_j \geq |s_j|^p, \ j = 1, ..., d, \ \sum_{j=1}^d z_j = 1\}.$$

A barrier function for this set is

$$f_p(s, z) = \sum_{j=1}^d (-2\log z_j - \log(z_j^{2/p} - s_j^2)).$$

Its barrier parameter is $4 \cdot d$ (where each term in the summation has parameter 4); see Nesterov and Nemirovskii [25].

Consider the conic hull

$$K(G_p) = \left\{(t, s \in R^d, z \in R^d) : \ t > 0, \ \left(\frac{s}{t}, \frac{z}{t}\right) \in G_p\right\}$$

$$= \left\{(t, s \in R^d, z \in R^d) : \ t > 0, \ t^{p-1}z_j \geq |s_j|^p, \ \sum_{j=1}^d z_j = t\right\},$$

which is equivalent to K for (t, s). In what follows, we prove the following theorem:

Theorem 3.1 *The function*

$$\hat{f}_p(t, s, z) = 25 \cdot \left(f_p\left(\frac{s}{t}, \frac{z}{t}\right) - 8d \log t \right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $200d$.

Proof. It is sufficient to prove

$$\hat{f}_p(t, s \in R, z \in R) = 25 \cdot (-2\log(z/t) - \log((z/t)^{2/p} - (s/t)^2) - 8 \log t)$$

$$= 25 \cdot (-2\log z - \log(z^{2/p} t^{2(p-1)/p} - s^2) - 4 \log t)$$

is a logarithmically homogeneous self-concordant barrier function with parameter 200. Our proof follows from Proposition 5.1.4 of Nesterov and Nemirovskii [25].

Let us fix $o = (t, z, s)$, $t > 0$ and $t^{p-1} z > |s|^p$, and let $w = (-d_t, d_z, d_s) \in R^3$. Let us compute the derivatives up to the order 3 of $\hat{f}_p$ at the point o in the direction w. For $\sigma = d_t/t$ and some $\alpha \in R$, we have

$$\frac{s + \alpha d_s}{t - \alpha d_t} = \frac{s}{t} + \frac{\alpha}{1 - \alpha\sigma}\left(\sigma \frac{s}{t} + \frac{d_s}{t}\right)$$

and

$$\frac{z + \alpha d_z}{t - \alpha d_t} = \frac{z}{t} + \frac{\alpha}{1 - \alpha\sigma}\left(\sigma \frac{z}{t} + \frac{d_z}{t}\right).$$

Let

$$\phi(z) := f_p\left(\frac{s}{t} + z\left(\sigma\frac{s}{t} + \frac{d_s}{t}\right), \frac{z}{t} + z\left(\sigma\frac{z}{t} + \frac{d_z}{t}\right)\right).$$

Then

$$\hat{\phi}(\alpha) := \hat{f}_p(o + \alpha w) = 25 \cdot \left(\phi(\frac{\alpha}{1 - \alpha\sigma}) - 8\log(1 - \alpha\sigma) - 8\log t\right).$$

Therefore, we have

$$\hat{\pi}_1 := \nabla \hat{f}_p(o)[w] = \hat{\phi}'(0) = 25(8\sigma + \pi_1),$$

$$\hat{\pi}_2 := \nabla^2 \hat{f}_p(o)[w, w] = \hat{\phi}''(0) = 25(8\sigma^2 + 2\sigma\pi_1 + \pi_2),$$

$$\hat{\pi}_3 := \nabla^3 \hat{f}_p(o)[w, w, w] = \hat{\phi}'''(0) = 25(16\sigma^3 + 6\sigma^2\pi_1 + 6\sigma\pi_2 + \pi_3),$$

where

$$\pi_1 = \phi'(0), \quad \pi_2 = \phi''(0), \quad \pi_3 = \phi'''(0).$$

Since f_p is a self-concordant barrier with the barrier parameter 4, we have

$$\pi_2 \geq 0, \quad \pi_1^2 \leq 4\pi_2, \quad \text{and} \quad |\pi_3| \leq 2\pi_2^{3/2}.$$

Thus, we see

$$\hat{\pi}_2 \geq 25(8\sigma^2 - 4|\sigma|\sqrt{\pi_2} + \pi_2) \geq 25\max(4\sigma^2, \pi_2/2) \geq 0,$$

which implies that $\hat{f}_p$ is a convex function. Now we need to prove that

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Note that, if $\sigma \leq 0$, we have

$$\begin{aligned}
\hat{\pi}_3/25 &= 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 8\sigma^3 + 3\sigma\pi_2 + \pi_3 \\
&\leq -8\sigma^3 + \pi_3 \\
&\leq 8|\sigma|^3 + 2\pi_2^{3/2} \\
&\leq \frac{8}{1000}\hat{\pi}_2^{3/2} + \frac{4\sqrt{2}}{125}\hat{\pi}_2^{3/2} \\
&\leq \frac{2}{25}\hat{\pi}_2^{3/2},
\end{aligned}$$

and

$$\begin{aligned}
-\hat{\pi}_3/25 &\leq -3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 3\sigma\pi_2 - \pi_3 \\
&\leq \frac{3}{250}\hat{\pi}_2^{3/2} - 3\sigma\pi_2 + 2\pi_2^{3/2}.
\end{aligned}$$

For the latter consider the following maximum problem for any fixed $a \geq 0$

$$\max\ -3xy^2 + 2y^3 :\ 8x^2 + 4xy + y^2 = a,\ x \leq 0,\ y \geq 0.$$

One can verify that the maximum value is below $8a^{3/2}$. Thus,

$$-3\sigma\pi_2 + 2\pi_2^{3/2} \leq \frac{8}{125}\hat{\pi}_2^{3/2},$$

which, together with the above inequality, implies that

$$-\hat{\pi}_3 \leq (\frac{3}{10} + \frac{8}{5})\hat{\pi}_2^{3/2} \leq 2\hat{\pi}_2^{3/2}.$$

Thus, for the case of $\sigma \leq 0$, we have

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Now we consider the case when $\sigma > 0$:

$$\begin{aligned}
\hat{\pi}_3/25 &= 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 8\sigma^3 + 3\sigma\pi_2 + \pi_3 \\
&\leq 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) + 3\sigma\pi_2 + \pi_3 \\
&\leq \frac{3}{25}\sigma\hat{\pi}_2 + 3\sigma\pi_2 + 2\pi_2^{3/2} \\
&\leq \frac{3}{250}\hat{\pi}_2^{3/2} + 3\sigma\pi_2 + 2\pi_2^{3/2}.
\end{aligned}$$

Consider again the following maximum problem for any fixed $a > 0$

$$\max\ 3xy^2 + 2y^3 :\ 8x^2 - 4xy + y^2 = a,\ x \geq 0,\ y \geq 0,$$

where the maximum value is below $8a^{3/2}$. Thus,

$$3\sigma\pi_2 + 2\pi_2^{3/2} \leq \frac{8}{125}\hat{\pi}_2^{3/2},$$

which, together with the above inequality, implies that

$$\hat{\pi}_3 \leq (\frac{3}{10} + \frac{8}{5})\hat{\pi}_2^{3/2} \leq 2\hat{\pi}_2^{3/2}.$$

Also,

$$\begin{aligned}
-\hat{\pi}_3/25 &\leq 8\sigma^3 - \pi_3 \\
&\leq 8|\sigma|^3 + 2\pi_2^{3/2} \\
&\leq \frac{8}{1000}\hat{\pi}_2^{3/2} + \frac{4\sqrt{2}}{125}\hat{\pi}_2^{3/2} \\
&\leq \frac{2}{25}\hat{\pi}_2^{3/2}.
\end{aligned}$$

To summarize, we have proved that

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Finally, it is easy to verify that the barrier parameter is 200 for $\hat{f}_p$. This completes the proof of the theorem. □

3.2 Barrier function II

In this section, we first consider the set

$$\{z \in R^d :\ z_j \geq 0,\ j = 1, ..., d,\ \sum_{j=1}^{d} z_j^p \leq 1\}.$$

A barrier function for this set is

$$f_p(z) = -4\log(1 - \sum_{j=1}^{d} z_j^p) - 4p^2 \sum_{j=1}^{d} \log z_j.$$

It is a convex barrier function. We present its barrier parameter in the following.

Theorem 3.2 *The function $f_p(z)$ is a logarithmically self-concordant function with parameter $4(1+p^2d)$ in the interior of the set $\{z \in R^d :\ \sum_{j=1}^{d} z_j^p \leq 1,\ z \geq 0\}$.*

Proof. Let $z > 0$ and $\delta = \sum_{j=1}^d z_j^p < 1$ be given, and let $w \in R^d$. Let

$$\phi(\alpha) := f_p(z + \alpha w).$$

We compute the derivatives up to the order 3 of f_p at the point z in the direction w:

$$\begin{aligned}\pi_1/4 &:= \nabla f_p(z)[w]/4 = \phi'(0)/4 \\ &= \frac{p}{1-\delta}\sum_{j=1}^d w_j z_j^{p-1} - p^2 \sum_{j=1}^d w_j z_j^{-1},\end{aligned}$$

$$\pi_2/4 := \nabla^2 f_p(z)[w,w]/4 = \phi''(0)/4$$
$$= \frac{p^2}{(1-\delta)^2}(\sum_{j=1}^d w_j z_j^{p-1})^2 + \frac{p(p-1)}{1-\delta}\sum_{j=1}^d w_j^2 z_j^{p-2} + p^2 \sum_{j=1}^d w_j^2 z_j^{-2},$$

and

$$\pi_3/4 := \nabla^3 f_p(z)[w,w,w]/4 = \phi'''(0)/4$$
$$= \frac{2p^3}{(1-\delta)^3}(\sum_{j=1}^d w_j z_j^{p-1})^3 + \frac{2p^2(p-1)}{(1-\delta)^2}(\sum_{j=1}^d w_j z_j^{p-1})(\sum_{j=1}^d w_j^2 z_j^{p-2})$$
$$+\frac{p^2(p-1)}{(1-\delta)^2}(\sum_{j=1}^d w_j z_j^{p-1})(\sum_{j=1}^d w_j^2 z_j^{p-2}) + \frac{p(p-1)(p-2)}{1-\delta}\sum_{j=1}^d w_j^3 z_j^{p-3} + p^2 \sum_{j=1}^d w_j^3 z_j^{-3}.$$

Now let

$$\phi_1 = \frac{p}{1-\delta}\sum_{j=1}^d w_j z_j^{p-1}, \ \phi_2 = p^2 \sum_{j=1}^d w_j z_j^{-1},$$

$$\phi_3 = \sqrt{\frac{p(p-1)}{1-\delta}\sum_{j=1}^d w_j^2 z_j^{p-2}}, \text{and } \phi_4 = \sqrt{p^2 \sum_{j=1}^d w_j^2 z_j^{-2}}.$$

Then,

$$\pi_1/4 = \phi_1 - \phi_2,$$
$$\pi_2/4 = \phi_1^2 + \phi_3^2 + \phi_4^2 > 0,$$

and

$$\begin{aligned}|\pi_3/4| &\le |2\phi_1^3 + 3\phi_1\phi_3^2| + \frac{|p-2|}{p}\phi_3^2\phi_4 + \frac{1}{p}\phi_4^3 \\ &\le 2|\phi_1|(\phi_1^2+\phi_3^2) + |\phi_1|\phi_3^2 + \phi_3(\phi_3^2+\phi_4^2) \\ &\le \pi_2^{3/2}/4 + \pi_2^{3/2}/8 + \pi_2^{3/2}/8 \\ &= \pi_2^{3/2}/2,\end{aligned}$$

i.e.,

$$|\pi_3| \leq 2\pi_2^{3/2}.$$

Furthermore,

$$\begin{aligned}
\pi_1^2 &= 16(\phi_1 - \phi_2)^2 \\
&\leq 16(1+p^2 d)(\phi_1^2 + \phi_4^2) \\
&\leq 4(1+p^2 d)\pi_2.
\end{aligned}$$

□

Similarly, we can prove the following corollary:

Corollary 3.1 *If $p \leq 3$, then*

$$f_p(z) = -4\log(1 - \sum_{j=1}^{d} z_j^p) - 4\sum_{j=1}^{d} \log z_j.$$

is a convex logarithmically self-concordant barrier function with parameter $4(1+d)$ in the interior of the set $\{z \in R^d : \ \sum_{j=1}^{d} z_j^p \leq 1, \ z \geq 0\}$.

□

We now consider the set

$$G_p = \{(s \in R^d, z \in R^d) : \ z_j \geq |s_j|, \ j = 1, ..., d, \ \sum_{j=1}^{d} z_j^p \leq 1\}.$$

A barrier function for this set is

$$f_p(s,z) = -4\log(1 - \sum_{j=1}^{d} z_j^p) - 4p^2\sum_{j=1}^{d} \log z_j - \sum_{j=1}^{d} \log(z_j^2 - s_j^2).$$

The barrier parameter of this function is $4(1+p^2 d) + 2d$, since By the first two summations have parameter $4(1+p^2 d)$ and the last summation has parameter $2d$.

Again consider the conic hull

$$K(G_p) = \left\{(t, s \in R^d, z \in R^d) : \ t > 0, \ \left(\frac{s}{t}, \frac{z}{t}\right) \in G_p\right\}$$

$$= \left\{(t, s \in R^d, z \in R^d) : \ t > 0, \ z_j \geq |s_j|, \ \sum_{j=1}^{d} z_j^p \leq t^p \right\},$$

which is also equivalent to K for (t,s). Again, similar to Theorem 3.1, we can prove the following theorem:

Theorem 3.3 *The function*

$$\hat{f}_p(t,s,z) = \theta^2 \cdot \left(f_p\left(\frac{s}{t}, \frac{z}{t}\right) - (8(1+p^2d)+4d)\log t \right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $\theta^2(8(1+p^2d)+4d)$, where θ is a positive constant; say $\theta = 5$.

We can simplify the above barrier function further. Note that $z_j \geq |s_j|$ implies $z_j \geq 0$, and the Hessian of $\log(z_j^2 - s_j^2)$ is

$$H(z_j, s_j) = \frac{2}{z_j^2 - s_j^2} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{4}{(z_j^2 - s_j^2)^2} \begin{pmatrix} t_j^2 & -t_j s_j \\ -t_j s_j & s_j^2 \end{pmatrix}.$$

For $z_j > |s_j|$, we can prove that

$$H(z_j, s_j) - \begin{pmatrix} z_j^{-2} & 0 \\ 0 & 0 \end{pmatrix}$$

is positive semi-definite. Thus, we have the following corollary, whose proof is similar to that for Theorem 3.2.

Corollary 3.2 *If $p \leq 3$, then*

$$f_p(s,z) = -4\log(1 - \sum_{j=1}^{d} z_j^p) - 4\sum_{j=1}^{d}\log(z_j^2 - s_j^2)$$

is a convex logarithmically self-concordant barrier function with parameter $4(1+2d)$ in the interior of the set $\{(s \in R^d, z \in R^d) : \ \sum_{j=1}^d z_j^p \leq 1, \ z \geq |s_j|\}$. The function

$$\hat{f}_p(t,s,z) = 25 \cdot \left(f_p\left(\frac{s}{t}, \frac{z}{t}\right) - 8(1+2d)\log t \right)$$

$$= 100 \left(-\log(t^p - \sum_{j=1}^{d} z_j^p) - \sum_{j=1}^{d}\log(z_j^2 - s_j^2) - (2-p+2d)\log t \right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $200(1+2d)$.

□

Symmetrically, we may consider the convex set

$$G_q = \{(x \in R^d, v \in R^d) : \ v_j \geq |x_j|^q, \ j = 1, ..., d, \ \sum_{j=1}^{d} v_j = 1\}$$

and the conic hull

$$K(G_q) = \left\{(\tau, x \in R^d, v \in R^d) : \ \tau > 0, \ \left(\frac{x}{\tau}, \frac{v}{\tau}\right) \in G_q\right\}$$

$$= \left\{(\tau, x \in R^d, v \in R^d) : \ \tau > 0, \ v_j \geq |x_j|, \ \sum_{j=1}^{d} v_j^q \leq \tau^q \right\},$$

which is also equivalent to K^* for (τ, x). A direct application of Corollary 3.2 leads to the following

Corollary 3.3 *If $q \leq 3$, then*

$$f_q(x, v) = -4 \log(1 - \sum_{j=1}^{d} v_j^q) - 4 \sum_{j=1}^{d} \log(v_j^2 - x_j^2)$$

is a convex logarithmically self-concordant barrier function with parameter $4(1 + 2d)$ in the interior of the set $\{(x \in R^d, v \in R^d) : \ \sum_{j=1}^{d} v_j^q \leq 1, \ v \geq |x_j|\}$. The function

$$\hat{f}_q(\tau, x, v) = 25 \cdot \left(f_q\left(\frac{x}{\tau}, \frac{v}{\tau}\right) - 8(1 + 2d) \log \tau\right)$$

$$= 100 \left(-\log(\tau^q - \sum_{j=1}^{d} v_j^q) - \sum_{j=1}^{d} \log(v_j^2 - x_j^2) - (2 - q + 2d) \log \tau\right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_q)$, where the barrier parameter is $200(1 + 2d)$.

□

3.3 Legendre transformations

Although it has a higher barrier parameter, barrier function II possesses a structure similar to the barrier function for the second-order cone, which we will use in our following analyses. Since we can solve both (P) and (D) using either the barrier function for the p-order cone in (D) or the barrier function for the q-order cone in (P), we will make the following conventions in this paper:

- When $p > 3$, use the barrier function of the q-order cone in (P).
- When $1 \leq p \leq 3$, use the barrier function of the p-order cone in (D).

It has been proved by Nesterov and Nemirovskii [25] that the Legendre transformation of the logarithmically homogeneous self-concordant barrier function in Corollary 3.2,

$$\hat{f}_p^*(\tau, x \in R^d) = \sup\{-\tau \cdot t - x^T s - \hat{f}_p(t, s, z) : \ (t, s, z) \in K(G_p)\},$$

is a logarithmically homogeneous self-concordant barrier for K^* with the same parameter $200(1+2d)$. It seems hard to find an explicit form of $\hat{f}_p^*(\tau, x)$. Fortunately, for the interior-point algorithm presented in the next section, we do not need such an explicit formula.

We need only the initial barrier function value at a primal initial point in our complexity analysis. We will set the initial point $\tau = 1$ and $x = 0$. Thus, we need to evaluate

$$\hat{f}_p^*(1,0) = \sup\{-t - \hat{f}_p(t,s,z) : \ (t,s,z) \in K(G_p)\}.$$

This is a convex optimization problem with an analytical solution

$$t^* = 200(1+2d), \ s^* = 0, \ z_j^* = \left(\frac{2}{p+2d}\right)^{1/p} t^*, \ j = 1, ..., d.$$

Thus,

$$\hat{f}_p^*(1,0) = 200(1+2d)(\log(200(1+2d)) - 1) + 100 \log \frac{p}{p+2d} + \frac{200d}{p} \log \frac{2}{p+2d}. \tag{3.1}$$

It has been proved by Nesterov and Nemirovskii [25] that the Legendre transformation of the logarithmically homogeneous self-concordant barrier function in Corollary 3.3,

$$\hat{f}_q^*(t, s \in R^d) = \sup\{-\tau \cdot t - x^T s - \hat{f}_q(\tau, x, v) : \ (\tau, x, v) \in K(G_q)\},$$

is a logarithmically homogeneous self-concordant barrier for K with the same parameter $200(1+2d)$. It seems hard to find the exact value of $\hat{f}_q^*(t,s)$ for $s \neq 0$. In the following, we will find an explicit formula for $\hat{f}_q^*(c,0)$ for any $c > 0$ and prove that $\hat{f}_q^*(t,s) \leq \hat{f}_q^*(t - ||s||_p, 0)$ for $(t,s) \in K$.

Note that

$$\hat{f}_q^*(c,0) = \sup\{-\tau - \hat{f}_q(\tau,x,v) : \ (\tau,x,v) \in K(G_q)\}.$$

This is a convex optimization problem with an analytical solution

$$\tau^* = 200(1+2d)/c, \ x^* = 0, \ v_j^* = \left(\frac{2}{q+2d}\right)^{1/q} \tau^*, \ j = 1, ..., d.$$

Thus,

$$\hat{f}_q^*(c,0) = 200(1+2d)(\log(200(1+2d)/c) - 1) + 100 \log \frac{q}{q+2d} + \frac{200d}{q} \log \frac{2}{q+2d}. \tag{3.2}$$

For any $(t,s) \in K$ and $(\tau, x) \in K^*$, we have

$$t \geq ||s||_p \quad \tau \geq ||x||_q.$$

Since

$$|s^T x| \leq ||s||_p ||x||_q,$$

we have

$$\begin{aligned} -t\tau - s^T x &\leq -t\tau + ||s||_p ||x||_q \\ &\leq -t\tau + ||s||_p \tau \\ &= -\tau(t - ||s||_p). \end{aligned}$$

Therefore

$$\hat{f}_q^*(t, s) \leq \hat{f}_q^*(t - ||s||_p, 0). \tag{3.3}$$

4 A primal-dual potential reduction algorithm

In this section, we will present a primal-dual potential reduction algorithm for computing an ϵ-optimal solutions for (P) and (D) in polynomial time. We will use either dual scaling or primal scaling depending on the value of p.

4.1 Use dual scaling when $p \in [1, 3]$

When $p \in [1, 3]$, we may solve (P) and (D) using the barrier function for the p-order cone. Let

$$F_p^*(\mathcal{X}) = \sum_{i=1}^{m} \hat{f}_p^*(\tau_i, x_i) \quad \text{and} \quad F_p(\mathcal{S}, \mathcal{Z}) = \sum_{i=1}^{m} \hat{f}_p(t_i, s_i, z_i \in R^d), \tag{4.1}$$

where

$$\mathcal{Z} := (z_1; z_2; \cdots; z_m).$$

They are logarithmically homogeneous self-concordant barriers for (P) and (D), where the barrier parameter is $\theta := 200(1 + 2d)m$. A primal-dual potential function for the pair (P) and (D) is

$$\phi_\rho(\mathcal{X}, \mathcal{S}, \mathcal{Z}) := \rho \log\langle \mathcal{X}, \mathcal{S} \rangle + F_p^*(\mathcal{X}) + F_p(\mathcal{S}, \mathcal{Z}) + \theta, \tag{4.2}$$

where $\rho = \theta + \gamma\sqrt{\theta}$, $\gamma \geq 1$. Note that

$$\langle \mathcal{X}, \mathcal{S} \rangle = \mathcal{X}^T \mathcal{S} = \mathcal{C}^T \mathcal{X} - \mathcal{B}^T \mathcal{Y}$$

and from Nesterov and Nemirovskii [25]

$$\phi_\theta(\mathcal{X}, \mathcal{S}, \mathcal{Z}) \geq \theta \log \theta. \tag{4.3}$$

The main iteration of a potential reduction algorithm starts with a strictly feasible primal-dual pair $\mathcal{X}$ and $(\mathcal{Y}, \mathcal{S})$, i.e.,

$$\begin{array}{ll} \mathcal{A}\mathcal{X} = \mathcal{B}, & \mathcal{S} = \mathcal{C} - \mathcal{A}^T\mathcal{Y}, \\ \mathcal{X} \in \text{int}\mathcal{K}^*, \quad \text{and} & \mathcal{S} \in \text{int}\mathcal{K}. \end{array}$$

It computes a search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$ via solving a system of linear equations. After obtaining $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$, a new strictly feasible primal-dual pair $\mathcal{X}^+$ and $(\mathcal{Y}^+, \mathcal{S}^+)$ is generated from

$$\mathcal{X}^+ = \mathcal{X} + \alpha d_\mathcal{X}, \ \mathcal{Y}^+ = \mathcal{Y} + \beta d_\mathcal{Y}, \ \mathcal{S}^+ = \mathcal{S} + \beta d_\mathcal{S}, \ \mathcal{Z}^+ = \mathcal{Z} + \beta d_\mathcal{Z},$$

for some step-sizes α and β, and

$$\phi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{Z}^+) \le \phi_\rho(\mathcal{X}, \mathcal{S}, \mathcal{Z}) - \Omega(1).$$

The search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$ is determined by the following equations.

$$\mathcal{A}d_\mathcal{X} = 0, \quad d_\mathcal{S} = -\mathcal{A}^T d_\mathcal{Y} \quad \text{(feasibility)} \tag{4.4}$$

and

$$\begin{pmatrix} d_\mathcal{X} \\ 0 \end{pmatrix} + F_p''(\mathcal{S}, \mathcal{Z}) \begin{pmatrix} d_\mathcal{S} \\ d_\mathcal{Z} \end{pmatrix} = -\frac{\rho}{\mathcal{X}^T\mathcal{S}} \begin{pmatrix} \mathcal{X} \\ 0 \end{pmatrix} - F_p'(\mathcal{S}, \mathcal{Z}). \tag{4.5}$$

The algorithm generates an ϵ-optimal solution $(\mathcal{X}, \mathcal{Y}, \mathcal{S})$, i.e.,

$$\langle \mathcal{X}, \mathcal{S} \rangle \le \epsilon$$

in a guaranteed $O(\gamma\sqrt{\theta}\log(\langle \mathcal{X}^0, \mathcal{S}^0 \rangle/\epsilon) + \phi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{Z}^0) - \theta\log\theta)$ iterations.

In practice, one usually finds the largest step-sizes $\bar{\alpha}$ and $\bar{\beta}$ such that

$$\mathcal{X} + \bar{\alpha} d_\mathcal{X} \in \mathcal{K}^*, \quad \text{and} \quad \mathcal{S} + \bar{\beta} d_\mathcal{S} \in \mathcal{K} \tag{4.6}$$

then takes $\alpha \in [0, \bar{\alpha}]$ and $\beta \in [0, \bar{\beta}]$, via a line-search, to minimize $\phi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{Z}^+)$, or simply chooses

$$\alpha = (0.5 \sim 0.99)\bar{\alpha} \quad \text{and} \quad \beta = (0.5 \sim 0.99)\bar{\beta} \tag{4.7}$$

as long as ϕ_ρ is reduced.

4.2 Use primal scaling when $p \in [3, \infty]$

The dual scaling algorithm is good when p is small $(1 \le p \le 3)$. For larger values of p, the barrier parameter for the p-order cone becomes larger. In this case, it is better to use the barrier function of the q-order cone in (P), where $\frac{1}{p} + \frac{1}{q} = 1$. It is clear that $q \in [1, 2]$ whenever $p \in [2, \infty]$.

Let

$$F_q(\mathcal{X}, \mathcal{V}) = \sum_{i=1}^{m} \hat{f}_q(\tau_i, x_i, v_i \in R^d) \quad \text{and} \quad F_q^*(\mathcal{S}) = \sum_{i=1}^{m} \hat{f}_q^*(t_i, s_i), \tag{4.8}$$

where

$$\mathcal{V} := (v_1; v_2; \cdots; v_m).$$

They are logarithmically homogeneous self-concordant barriers for (P) and (D), where the barrier parameter is $\theta := 200(1 + 2d)m$. A primal-dual potential function for the pair (P) and (D) is

$$\phi_\rho(\mathcal{X}, \mathcal{V}, \mathcal{S}) := \rho \log\langle \mathcal{X}, \mathcal{S}\rangle + F_q(\mathcal{X}, \mathcal{V}) + F_q^*(\mathcal{S}) + \theta, \tag{4.9}$$

where $\rho = \theta + \gamma\sqrt{\theta}$, $\gamma \ge 1$. Note that

$$\langle \mathcal{X}, \mathcal{S}\rangle = \mathcal{X}^T \mathcal{S} = \mathcal{C}^T \mathcal{X} - \mathcal{B}^T \mathcal{Y}$$

and from Nesterov and Nemirovskii [25]

$$\phi_\theta(\mathcal{X}, \mathcal{V}, \mathcal{S}) \ge \theta \log \theta. \tag{4.10}$$

The main iteration of a potential reduction algorithm starts with a strictly feasible primal-dual pair $(\mathcal{X}, \mathcal{V})$ and $(\mathcal{Y}, \mathcal{S})$, i.e.,

$$\begin{aligned} \mathcal{A}\mathcal{X} = \mathcal{B}, &\qquad \mathcal{S} = \mathcal{C} - \mathcal{A}^T \mathcal{Y}, \\ \mathcal{X} \in \text{int}\mathcal{K}, &\quad \text{and} \quad \mathcal{S} \in \text{int}\mathcal{K}^*. \end{aligned}$$

It computes a search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{V})$ via solving a system of linear equations. After obtaining $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{V})$, a new strictly feasible primal-dual pair $\mathcal{X}^+$ and $(\mathcal{Y}^+, \mathcal{S}^+)$ is generated from

$$\mathcal{X}^+ = \mathcal{X} + \alpha d_\mathcal{X}, \;\; \mathcal{Y}^+ = \mathcal{Y} + \beta d_\mathcal{Y}, \;\; \mathcal{S}^+ = \mathcal{S} + \beta d_\mathcal{S}, \;\; \mathcal{V}^+ = \mathcal{V} + \alpha d_\mathcal{V},$$

for some step-sizes α and β, and

$$\phi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{V}^+) \le \phi_\rho(\mathcal{X}, \mathcal{S}, \mathcal{V}) - \Omega(1).$$

The search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{V})$ is determined by (4.4) and the following equations.

$$\begin{pmatrix} d_\mathcal{S} \\ 0 \end{pmatrix} + F_q''(\mathcal{X}, \mathcal{V}) \begin{pmatrix} d_\mathcal{X} \\ d_\mathcal{V} \end{pmatrix} = -\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} \mathcal{S} \\ 0 \end{pmatrix} - F_q'(\mathcal{X}, \mathcal{V}). \tag{4.11}$$

The algorithm generates an ϵ-optimal solution $(\mathcal{X}, \mathcal{Y}, \mathcal{S})$, i.e.,

$$\langle \mathcal{X}, \mathcal{S} \rangle \leq \epsilon$$

in a guaranteed $O(\gamma\sqrt{\theta}\log(\langle \mathcal{X}^0, \mathcal{S}^0\rangle/\epsilon) + \phi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{V}^0) - \theta\log\theta)$ iterations.

In practice, one usually finds the largest step-sizes $\bar{\alpha}$ and $\bar{\beta}$ such that

$$\mathcal{X} + \bar{\alpha} d_{\mathcal{X}} \in \mathcal{K}, \quad \text{and} \quad \mathcal{S} + \bar{\beta} d_{\mathcal{S}} \in \mathcal{K}^* \tag{4.12}$$

then takes $\alpha \in [0, \bar{\alpha}]$ and $\beta \in [0, \bar{\beta}]$, via a line-search, to minimize $\phi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{V}^+)$, or simply chooses

$$\alpha = (0.5 \sim 0.99)\bar{\alpha} \quad \text{and} \quad \beta = (0.5 \sim 0.99)\bar{\beta} \tag{4.13}$$

as long as ϕ_ρ is reduced.

5 Complexity and implementation

As we have seen, the number of iterations required to compute an ϵ-optimal solution to problem (2.1) depends on the initial point $(\mathcal{X}^0, \mathcal{S}^0, \mathcal{Z}^0)$. In this section, we discuss initial point selection and other computational issues for solving problem (2.1) using the algorithms presented in section 4.

5.1 Initial point for dual scaling

The algorithms discussed in the previous section all require a pair of strictly primal-dual interior feasible solutions. In the following, we give one such pair.

Let

$$\bar{c} = \max_{1 \leq i \leq m} \|c_i\|,$$

and, for $i = 1, 2, \cdots, m$, let

$$u^0 = 0,\ s_i^0 = c_i,\ t_i^0 = \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2}\left(\frac{p+2d}{2}\right)^{\frac{1}{p}},\ z_i^0 = \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2}\ e \in R^d,$$

and

$$\tau_i^0 = 1, \quad x_i^0 = 0 \in R^d.$$

Then, one can verify that $\mathcal{X}^0$ is an interior feasible solution to (P) and $\mathcal{S}^0$ and $\mathcal{Y}^0$ form an interior feasible solution to (D). One can also verify that

$$\langle \mathcal{X}^0, \mathcal{S}^0 \rangle = (\mathcal{X}^0)^T \mathcal{S}^0 = \sum_{i=1}^m t_i^0 \tau_i^0 = \left(\frac{p+2d}{2}\right)^{\frac{1}{p}} \sum_{i=1}^m \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2}$$

$$\leq \left(\frac{p+2d}{2}\right)^{\frac{1}{p}} \bar{c}m\sqrt{1+m(1+2d)}$$

and the initial value

$$\begin{aligned}
\hat{f}_p(t_i^0, s_i^0, z_i^0) &\leq 100\left(-(2+2d)\log t_i^0 - \log\frac{p}{p+2d} - d\log m(1+2d)\bar{c}^2\right) \\
&\leq -100(1+2d)\log m(1+2d)\bar{c}^2 - \frac{200(1+d)}{p}\log\frac{p+2d}{2} - 100\log\frac{p}{p+2d}.
\end{aligned}$$

From this inequality and (3.1), we have

$$\begin{aligned}
&F_p^*(\mathcal{X}^0) + F_p(\mathcal{S}^0, \mathcal{Z}^0) \\
&= \sum_{i=1}^m \hat{f}_p^*(1,0) + \sum_{i=1}^m \hat{f}_p(t_i^0, s_i^0, z_i^0) \\
&\leq \theta(\log(200(1+2d)) - 1) - 100(1+2d)m\log m(1+2d)\bar{c}^2 + \frac{\theta}{p}\log\frac{2}{p+2d}
\end{aligned}$$

Thus, from these inequalities

$$\begin{aligned}
&\phi_\theta(\mathcal{X}^0, \mathcal{S}^0) - \theta\log\theta \\
&= \theta\log\langle\mathcal{X}^0, \mathcal{S}^0\rangle + F_p^*(\mathcal{X}^0) + F_p(\mathcal{S}^0, \mathcal{Z}^0) + \theta - \theta\log\theta \\
&\leq \theta\log(\bar{c}m\sqrt{1+m(1+2d)}) - 100(1+2d)m\log m(1+2d)\bar{c}^2 \\
&\quad +\theta\log(200(1+2d)) - \theta\log\theta \\
&= 100(1+2d)m\log(1+m(1+2d)) - 100(1+2d)m\log m(1+2d) \\
&= 100(1+2d)m\log(1+\frac{1}{m(1+2d)}) \\
&\leq 100.
\end{aligned}$$

With this initial point, we have the following corollary.

Corollary 5.1 *Let the initial feasible primal solution $\mathcal{X}^0$ and dual feasible solution $(\mathcal{Y}^0, \mathcal{S}^0, \mathcal{Z}^0)$ be given as above and $1 \leq p \leq 3$. Then, an ϵ-optimal solution to problem (2.1) can be obtained by the (dual) potential reduction algorithm in at most*

$$O\left(\gamma\sqrt{200(1+2d)m}\,(\log(\bar{c}/\epsilon) + \log(md))\right)$$

iterations, where

$$\bar{c} = \max_{1\leq i\leq m} \|c_i\|.$$

□

5.2 Initial point for primal scaling

In this section, we give a pair of strictly primal-dual interior feasible solutions for the primal scaling algorithm. It is assumed that $p \geq 2$.

Let
$$\bar{c} = \max_{1\leq i\leq m} \|c_i\|,$$
and, for $i = 1, 2, \cdots, m$, let
$$u^0 = 0, \;\; s_i^0 = c_i, \;\; t_i^0 = ||c_i||_p + m(1+2d)\bar{c}.$$
and
$$\tau_i^0 = 1, \quad x_i^0 = 0 \in R^d, v_i^0 = (\frac{2}{q+2d})^{\frac{1}{q}} e.$$
Then, one can verify that $\mathcal{X}^0$ is an interior feasible solution to (P) and $\mathcal{S}^0$ and $\mathcal{Y}^0$ form an interior feasible solution to (D). One can also verify that
$$\langle \mathcal{X}^0, \mathcal{S}^0 \rangle = (\mathcal{X}^0)^T \mathcal{S}^0 = \sum_{i=1}^m t_i^0 \tau_i^0 \leq m\bar{c} + m^2(1+2d)\bar{c},$$
and the initial value
$$\begin{aligned}
\hat{f}_q(\tau_i^0, x_i^0, v_i^0) &= 100\left(-\log((\tau_i^0)^q - \sum_{j=1}^d (v_j^0)^q) - \sum_{j=1}^d \log(v_j^0)^2 - (2-q+2d)\log\tau_i^0\right) \\
&= 100\left(-\log(1 - \frac{2d}{q+2d}) - \frac{2d}{q}\log\frac{2}{q+2d}\right) \\
&= 100\left(-\log(\frac{q}{q+2d}) - \frac{2d}{q}\log\frac{2}{q+2d}\right). \\
\hat{f}_i^*(t_i, s_i) &\leq 200(1+2d)\left(\log\frac{200(1+2d)}{m(1+2d)\bar{c}} - 1\right) \\
&+ 100\left(\log\frac{q}{q+2d} + \frac{2d}{q}\log\frac{2}{q+2d}\right).
\end{aligned}$$
Therefore
$$\hat{f}_q(\tau_i, x_i, v_i) + \hat{f}_i^*(t_i, s_i) \leq 200(1+2d)\left(\log\frac{200(1+2d)}{m(1+2d)\bar{c}} - 1\right).$$
From these inequalities, we have
$$\begin{aligned}
&F_q^*(\mathcal{S}^0) + F_q(\mathcal{X}^0, \mathcal{V}^0) \\
&= \sum_{i=1}^m \hat{f}_q^*(t_i^0, s_i^0) + \sum_{i=1}^m \hat{f}_q(\tau_i^0, x_i^0, v_i^0) \\
&\leq \theta(\log\frac{200(1+2d)}{m(1+2d)\bar{c}} - 1).
\end{aligned}$$

Thus,

$$
\begin{aligned}
&\phi_\theta(\mathcal{X}^0,\mathcal{S}^0)-\theta\log\theta\\
&= \theta\log\langle\mathcal{X}^0,\mathcal{S}^0\rangle+F_q^*(\mathcal{S}^0)+F_q(\mathcal{X}^0,\mathcal{V}^0)+\theta-\theta\log\theta\\
&\le \theta\log(m\bar{c}+m^2(1+2d)\bar{c})+\theta\left(\log\frac{200(1+2d)}{m(1+2d)\bar{c}}-1\right)+\theta-\theta\log\theta\\
&= \theta\log\left(1+\frac{1}{m(1+2d)}\right)\\
&\le 200.
\end{aligned}
$$

With this initial point, we have the following corollary.

Corollary 5.2 *Let the initial feasible primal solution* $(\mathcal{X}^0,\mathcal{V}^0)$ *and dual feasible solution* $(\mathcal{Y}^0,\mathcal{S}^0)$ *be given as above and* $2\le q\le 3$*. Then, an* ϵ*-optimal solution to problem (2.1) can be obtained by the (primal) potential reduction algorithm in at most*

$$O\left(\gamma\sqrt{200(1+2d)m}\,(\log(\bar{c}/\epsilon)+\log(md))\right)$$

iterations, where

$$\bar{c}=\max_{1\le i\le m}\|c_i\|.$$

□

5.3 Search direction

At each step of the potential reduction algorithm, we need to compute the search direction $d_\mathcal{X}$, $d_\mathcal{S}$, $d_\mathcal{Y}$ and $d_\mathcal{Z}$ by solving a system of linear equations. In what follows, we will show that this can be further simplified, taking advantage of the special structure of the problem.

Consider the search direction defined by dual scaling (4.5). For $i=1,\cdots,m$, it can be decomposed as

$$
\begin{pmatrix} d_{\tau_i}\\ 0_d\\ d_{x_i}\end{pmatrix}+
\begin{pmatrix}
\frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial t_i\partial t_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial t_i\partial z_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial t_i\partial s_i}\\
\frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial z_i\partial t_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial z_i\partial z_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial z_i\partial s_i}\\
\frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial s_i\partial t_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial s_i\partial z_i} & \frac{\partial^2\hat{f}_p(t_i,s_i,z_i)}{\partial s_i\partial s_i}
\end{pmatrix}
\begin{pmatrix} d_{t_i}\\ d_{z_i}\\ d_{s_i}\end{pmatrix}
=-\frac{\rho}{\mathcal{X}^T\mathcal{S}}\begin{pmatrix}\tau_i\\ 0\\ x_i\end{pmatrix}-
\begin{pmatrix}
\frac{\partial\hat{f}_p(t_i,s_i,z_i)}{\partial t_i}\\
\frac{\partial\hat{f}_p(t_i,s_i,z_i)}{\partial z_i}\\
\frac{\partial\hat{f}_p(t_i,s_i,z_i)}{\partial s_i}
\end{pmatrix}
$$

Note that $s_i = c_i - A_i^T u$, $d_{s_i} = -A_i^T d_u$, $\tau_i = 1$, and $d_{\tau_i} = 0$ for $i = 1, \cdots, m$. The system can be written as

$$\begin{pmatrix} 0 \\ 0 \\ d_{x_i} \end{pmatrix} + \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial s_i} \end{pmatrix} \begin{pmatrix} d_{t_i} \\ d_{z_i} \\ -A_i^T d_u \end{pmatrix} = -\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \\ x_i \end{pmatrix} - \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial s_i} \end{pmatrix}$$

Let

$$J_i = \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial z_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial z_i} \end{pmatrix}.$$

One can easily verify that J_i is positive definite. Therefore we can compute J_i^{-1} in at most $O(d^3)$ time. From the first two equations, we get

$$\begin{pmatrix} d_{t_i} \\ d_{z_i} \end{pmatrix} = J_i^{-1} \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial s_i} \end{pmatrix} A_i^T d_u - \frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right]. \tag{5.1}$$

Substituting this into the third equation, we get

$$\begin{aligned} d_{x_i} \quad &+ \quad \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix} - \frac{\partial^2 \hat{f}_p(t_i, s_i, z_i)}{\partial s_i \partial s_i} \right] A_i^T d_u \\ &= \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \left[\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right] - \frac{\rho}{\mathcal{X}^T \mathcal{S}} x_i - \frac{\partial \hat{f}_p(t_i, s_i, z_i)}{\partial s_i} \end{aligned} \tag{5.2}$$

Moreover, since

$$\sum_{i=1}^m A_i x_i = 0, \ \sum_{i=1}^m A_i d_{x_i} = 0,$$

we have

$$\begin{aligned} &\sum_{i=1}^m A_i \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix} - \frac{\partial^2 \hat{f}_p(t_i, s_i, z_i)}{\partial s_i \partial s_i} \right] A_i^T d_u = \\ &\sum_{i=1}^m A_i \left(\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \left[\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right] - \frac{\rho}{\mathcal{X}^T \mathcal{S}} x_i - \frac{\partial \hat{f}_p(t_i, s_i, z_i)}{\partial s_i} \right). \end{aligned} \tag{5.3}$$

Note that the system for computing d_u may not have full rank. If that is the case, any feasible solution is acceptable.

It requires $O(m(d^3+nd^2+n^2d)$ operations to set up the system (5.3) for computing d_u. Solving the system requires $O(n^3)$ operations. Once d_u is computed, $O(m(d^3+nd^2+n^2d)$ operations are required to compute $d_{\mathcal{X}}$, $d_{\mathcal{S}}$ and $d_{\mathcal{Z}}$. Therefore, the number of arithmetic operations in each iteration is bounded by $O(n^3+md^3+md^2n+mdn^2)$. The following theorem follows from Corollary 5.1 and the above analysis.

Theorem 5.1 *Let the initial feasible primal solution $\mathcal{X}^0$ and dual feasible solution $(\mathcal{Y}^0;\mathcal{S}^0)$ be given as above. Then, an ϵ-optimal solution to problem (2.1) can be obtained by the potential reduction algorithms in at most*

$$O\left(\gamma\sqrt{200(1+2d)m}\,(\log(\bar{c}/\epsilon)+\log m+\log d)\right)$$

iterations, where

$$\bar{c}=\max_{1\le i\le m}\|c_i\|,$$

and each iteration requires $O(n^3+md^3+md^2n+mdn^2)$ arithmetic operations. □

Note that if both d and γ are constants and the problem is normalized such that $\bar{c}=1$, i.e., all of c_i is within the unit ball in R^d, then the iteration bound is $O(\sqrt{m}(\log(1/\epsilon)+\log m))$. We will further discuss this issue in following applications.

6 Applications

In this section, we will apply the algorithms presented in the previous sections to solve the p-norm multi-facility location (PMFL) problem and the p-norm Steiner minimal tree (MST) problem under a given topology. We will also take advantage of the special structures of these special problems and obtain improved computational complexity results wherever possible.

6.1 The p-norm multi-facility location problem

Let $a_1,a_2,\ldots,a_M$ be M points in R^d, the d-dimensional Euclidean space. Let w_{ji}, $j=1,2,\ldots,N$, $i=1,2,\ldots,M$, and $v_{jk}, 1\le j<k\le N$ be given nonnegative numbers. Find a point $x=(x_1;x_2;\cdots;x_N)\in R^{dN}$ that will minimize

$$f_p(x)=\sum_{j=1}^{N}\sum_{i=1}^{M}w_{ji}\|x_j-a_i\|_p+\sum_{1\le j<k\le N}v_{jk}\|x_j-x_k\|_p,\ p\ge 1. \tag{6.1}$$

This is the so-called p-norm multi-facility location problem (PMFL). For ease of notation, we assume that $v_{jj}=0$ for $j=1,2,\ldots,N$ and that $v_{jk}=v_{kj}$ for $1\le k<j\le N$.

In the PMFL problem, $a_1, a_2, \ldots, a_M$ represent the locations of M existing facilities; $x_1, x_2, \ldots, x_N$ represent the locations of N new facilities; the objective function $f_p(x)$ is the sum of weighted Euclidean distances from each new facility to each existing facility and those between each pair of new facilities; our goal is to find optimal locations for the new facilities, i.e., to minimize $f_p(x)$.

In problem (6.1), some of the weights w_{ji} and v_{jk} may be zero. Let m be the number of nonzero weights in (6.1). Then the PMFL problem (6.1) is the minimization of m p-norms. Without loss of generality, we assume that for each $j \in \{1, 2, \ldots, N\}$, there exists a nonzero w_{ji} for some $i \in \{1, 2, \ldots, M\}$ or a nonzero v_{jk} for some $k \in \{1, 2, \ldots, N\}$.

To transform the PMFL problem (6.1) into an instance of problem (1.1), we simply do the following. Let $u = (x_1; x_2; \cdots; x_N)$. It is clear that $u \in R^n$ where $n = dN$. For each nonzero w_{ji}, there is a corresponding term of Euclidean norm $||c(w_{ji}) - A(w_{ji})^T u||$ where $c(w_{ji}) = w_{ji} a_i$ and $A(w_{ji})^T$ is a row of N blocks of d-by-d matrices whose jth block is $w_{ji} I_d$ and whose other blocks are all zero. For each nonzero v_{jk}, there is a corresponding term of Euclidean norm $||c(v_{jk}) - A(v_{jk})^T u||$ where $c(v_{jk}) = 0$ and $A(v_{jk})^T$ is a row of N blocks of d-by-d matrices whose jth and kth blocks are $-v_{jk} I_d$ and $v_{jk} I_d$, respectively, and whose other blocks are all zero.

Now it is clear that we have transformed the PMFL problem (6.1) into an instance of (2.1) where $n = dN$, and m is the number of non-zero weights w_{ji} and v_{jk}. Note that the system (5.3) can be set up with $O(md^2)$ operations. Therefore it follows from Theorem 5.1 that we have

Theorem 6.1 *An ϵ-optimal solution to the PMFL problem (6.1) can be computed using any of our algorithms in at most*

$$O\left(\gamma\sqrt{200(1+2d)MN}\,(\log(\bar{c}/\epsilon) + \log(MN) + \log d)\right)$$

iterations where $\bar{c} = \max_{1 \le j \le n \quad 1 \le i \le m} ||w_{ji} a_i||$, and each iteration requires $O(d^3N^3 + MNd^2)$ arithmetic operations. □

6.2 The p-norm Steiner minimal tree problem

The *Euclidean Steiner minimal tree problem* is given by a set of points $P = \{p_1, p_2, \ldots, p_N\}$ in the Euclidean plane, and asks for the shortest planar straight-line graph spanning P. The solution takes the form of a tree, called the *Steiner minimal tree*, that includes all the given points, called *regular points*, along with some extra vertices, called *Steiner points*. It is known that there are at most $N-2$ Steiner points and the degree of each Steiner point is at most 3. See [13, 23] for details.

Definition 6.1 ([13, 16, 17]) *A* full Steiner topology *of point set P is a tree graph whose vertex set contains P and $N-2$ Steiner points and that the degree of each vertex in P is exactly 1 and that the degree of each Steiner vertex is exactly 3.*

Computing a Steiner minimal tree for a given set of N points in the Euclidean plane is NP-hard. However, the problem of computing the shortest network under a given full Steiner topology can be solved efficiently. Recently, there have been increased interests in this latter problem and several algorithms have been proposed [16, 17, 31]. We will formulate this problem as a special case of problem (1.1).

Let $m = 2N-3$, $d = 2$, $n = 2N-4$. Let $u \in R^{2N-4}$ represent the locations of the $N-2$ Steiner points. Without loss of generality, we may order the edges in the given full Steiner topology in such a way that the first N edges each connects a regular point to a Steiner point. For $i = 1, 2, \ldots, N$, c_i is p_{i_1} where i_1 is the index of the regular point on the ith edge; $A_i^T \in R^{2\times n}$ is a row of $N-2$ 2-by-2 block matrices where only the i_2th block is I_2 and the rest are all zero, where i_2 is the index of the Steiner point on the ith edge. For $i = N+1, N+2, \ldots, m$, $c_i = 0$ and $A_i^T \in R^{2\times n}$ is a row of $N-2$ 2-by-2 block matrices where the i_1st block is $-I_2$, the i_2nd block is I_2, and the rest blocks are all zero, where i_1 and i_2 are the indices of the two Steiner points on the ith edge. It is clear that we have transformed the problem of computing a shortest network under a full Steiner topology into an instance of (2.1) where $d = 2$, $n = 2N-4$ and $m = 2N-3$. Therefore it can be solved efficiently using our interior-point algorithm.

Note that we can move the point set P so that its gravitational center is the origin. Therefore the Euclidean norms of the regular points are bounded by the largest pair wise distance among the points in P which corresponds to the constant $\bar{c}$ in previous theorems. Furthermore, as illustrated in [36], the search direction can be computed in $O(N)$ arithmetic operations using a technique known as *Gaussian elimination on leaves of a tree* [31]. Therefore we have

Theorem 6.2 *An ϵ-optimal solution to the shortest network under a given full Steiner topology of N regular points in the l_p plane can be computed using our potential reduction algorithms in at most $O(\sqrt{N}(\log(\bar{c}/\epsilon)+\log N))$ iterations where $\bar{c}$ is the largest pair wise distance among the regular points and each iteration requires $O(N)$ arithmetic operations. Therefore, the computation of an ϵ-optimal solution requires $O(N\sqrt{N}(\log(\bar{c}/\epsilon)+\log N))$ arithmetic operations.* □

The problem of computing the shortest network under a full Steiner topology was first studied by Hwang [16], Hwang and Weng [17], and Smith [31]. Hwang [16] presented a linear time exact algorithm that can output the shortest network under a given full Steiner topology if there exists a non-degenerate SMT corresponding to that given topology and quits other wise. Hwang and Weng [17] presented an $O(N^2)$ time graphical algorithm that can output the shortest network under a given full Steiner topology if the shortest network under the given topology is a tree with maximum vertex degree 3 and quits other wise. Xue and Ye [36] proposed a primal-dual potential reduction algorithm which can always output an ϵ-optimal network under the given topology in $O(N\sqrt{N}(\log(\bar{c}/\epsilon) + \log N))$ operations where $\bar{c}$ is the largest pair wise distance among the given points.

7 Computational Examples

We have implemented all three versions of our algorithm using Matlab. Our intention here is to justify the theory developed in this paper. Therefore, our primary interest is in the number of iterations required by the algorithms. Our implementation here is very preliminary. Extensive computational study of the algorithms will be reported in a separate paper. For test problems, we have taken the 10-regular points Steiner minimum tree problem from [36]. The coordinates of the regular points are given in Table 1. The tree topology is given in Table 2 where for each edge, indices of its two vertices are shown next to the index of the edge.

index	x-coordinate	y-coordinate	index	x-coordinate	y-coordinate
9	2.30946900	9.20821100	14	7.59815200	0.61583600
10	0.57736700	6.48093800	15	8.56812900	3.07917900
11	0.80831400	3.51906200	16	4.75750600	3.75366600
12	1.68591200	1.23167200	17	3.92609700	7.00879800
13	4.11085500	0.82111400	18	7.43649000	7.68328400

Table 1: The coordinates of the 10 regular points.

In our implementation, we used $\gamma = 2m$ to take *long steps* instead of using the conservative theoretical parameter $\gamma = 1$. Also, we used 0.75 times the largest feasible step-size as the actual step-size rather than using the theoretical step-size or a line search. The algorithm stops whenever the absolute duality gap is smaller than 10^{-5}.

To test the flexibility of our algorithm, we have used the values of $p = 1.01, 1.5, 2.0, 3.0$ and 101 on the 10-point case. Although with a larger parameter, the algorithm based on barrier function 2 works better in our examples. All 5 cases were solved within 30 to 40 iterations. The output of the algorithm were presented in Tables 3-7. Note that for the last case, we have used the primal

edge-index	ea-index	eb-index	edge-index	ea-index	eb-index
1	9	7	10	18	8
2	10	1	11	5	6
3	11	2	12	6	4
4	12	3	13	4	3
5	13	4	14	3	2
6	14	5	15	2	1
7	15	5	16	1	7
8	16	6	17	7	8
9	17	8			

Table 2: The tree topology.

scaling algorithm where $q = 1.01$. The second column in the tables shows the cost of the current network (i.e., the sum of p-norms in the current network). The third column shows the duality gap which is an upper bound of the error in the cost of the current network to the cost of the optimal (shortest) network. The last two columns show the largest dual and primal feasible step-sizes.

iteration	network-cost	duality-gap	dstep-max	pstep-max
1	1.9810e+03	1.9863e+03	3.1250e-01	9.7656e-03
2	9.3134e+02	9.3948e+02	3.1250e-01	9.7656e-03
3	4.6604e+02	4.6352e+02	3.1250e-01	7.8125e-02
4	2.6185e+02	2.5141e+02	3.1250e-01	1.9531e-02
5	1.6673e+02	1.5389e+02	3.1250e-01	4.8828e-03
6	1.1164e+02	1.0050e+02	3.1250e-01	1.9531e-02
7	8.2320e+01	7.0048e+01	3.1250e-01	4.8828e-03
8	6.3980e+01	4.8173e+01	3.1250e-01	4.8828e-03
9	5.1500e+01	3.2779e+01	3.1250e-01	4.8828e-03
10	4.3166e+01	2.2726e+01	3.1250e-01	2.4414e-03
11	3.7916e+01	1.5499e+01	3.1250e-01	1.2207e-03
12	3.4452e+01	1.0726e+01	3.1250e-01	6.1035e-04
13	3.2373e+01	6.5718e+00	3.1250e-01	6.1035e-04
14	3.0782e+01	3.5208e+00	3.1250e-01	6.1035e-04
15	2.9691e+01	2.0491e+00	3.1250e-01	3.0518e-04
16	2.9255e+01	1.3093e+00	3.1250e-01	7.6294e-05
17	2.9023e+01	6.9142e-01	3.1250e-01	7.6294e-05
18	2.8824e+01	3.9670e-01	3.1250e-01	3.8147e-05
19	2.8737e+01	2.3084e-01	3.1250e-01	1.9073e-05
20	2.8682e+01	1.4074e-01	3.1250e-01	9.5367e-06
21	2.8649e+01	9.0313e-02	3.1250e-01	4.7684e-06
22	2.8630e+01	5.2046e-02	3.1250e-01	4.7684e-06
23	2.8615e+01	2.7875e-02	3.1250e-01	4.7684e-06
24	2.8605e+01	1.7067e-02	3.1250e-01	2.3842e-06
25	2.8601e+01	1.1235e-02	3.1250e-01	5.9605e-07
26	2.8599e+01	6.6474e-03	3.1250e-01	5.9605e-07
27	2.8597e+01	3.5436e-03	3.1250e-01	5.9605e-07
28	2.8596e+01	2.0801e-03	3.1250e-01	5.9605e-07
29	2.8595e+01	1.3643e-03	3.1250e-01	7.4506e-08
30	2.8594e+01	5.2467e-04	6.2500e-01	7.4506e-08
31	2.8594e+01	2.8420e-04	6.2500e-01	1.8626e-08
32	2.8594e+01	1.5954e-04	6.2500e-01	9.3132e-09
33	2.8594e+01	9.4942e-05	6.2500e-01	4.6566e-09
34	2.8594e+01	4.2928e-05	6.2500e-01	4.6566e-09
35	2.8594e+01	1.6635e-05	6.2500e-01	2.3283e-09
36	2.8594e+01	9.0572e-06	6.2500e-01	5.8208e-10

Table 3: Output for $p = 1.01$ using dual scaling.

iteration	network-cost	duality-gap	dstep-max	pstep-max
1	6.3022e+02	6.3501e+02	3.1250e-01	9.7656e-03
2	3.0949e+02	3.0829e+02	3.1250e-01	7.8125e-02
3	2.4246e+02	2.3339e+02	1.5625e-01	9.7656e-03
4	1.4290e+02	1.2897e+02	3.1250e-01	9.7656e-03
5	9.0251e+01	7.7152e+01	3.1250e-01	1.9531e-02
6	6.3431e+01	5.2464e+01	3.1250e-01	4.8828e-03
7	4.9956e+01	3.5671e+01	3.1250e-01	2.4414e-03
8	4.1268e+01	2.2964e+01	3.1250e-01	2.4414e-03
9	3.5487e+01	1.3659e+01	3.1250e-01	2.4414e-03
10	3.1635e+01	9.2619e+00	3.1250e-01	6.1035e-04
11	2.9957e+01	5.0433e+00	3.1250e-01	6.1035e-04
12	2.8370e+01	2.8946e+00	3.1250e-01	6.1035e-04
13	2.7627e+01	1.7436e+00	3.1250e-01	1.5259e-04
14	2.7198e+01	1.0872e+00	3.1250e-01	7.6294e-05
15	2.6951e+01	5.7794e-01	3.1250e-01	7.6294e-05
16	2.6757e+01	3.1499e-01	3.1250e-01	7.6294e-05
17	2.6666e+01	1.8555e-01	3.1250e-01	1.9073e-05
18	2.6621e+01	1.1079e-01	3.1250e-01	9.5367e-06
19	2.6595e+01	6.8918e-02	3.1250e-01	4.7684e-06
20	2.6581e+01	3.5549e-02	3.1250e-01	4.7684e-06
21	2.6569e+01	1.8558e-02	3.1250e-01	4.7684e-06
22	2.6563e+01	1.0725e-02	3.1250e-01	1.1921e-06
23	2.6561e+01	6.2128e-03	3.1250e-01	5.9605e-07
24	2.6559e+01	3.7579e-03	3.1250e-01	2.9802e-07
25	2.6558e+01	2.3707e-03	3.1250e-01	1.4901e-07
26	2.6557e+01	7.8395e-04	6.2500e-01	1.4901e-07
27	2.6557e+01	3.5326e-04	6.2500e-01	3.7253e-08
28	2.6557e+01	1.4104e-04	6.2500e-01	1.8626e-08
29	2.6557e+01	7.4709e-05	6.2500e-01	4.6566e-09
30	2.6557e+01	4.4346e-05	6.2500e-01	2.3283e-09
31	2.6557e+01	1.8446e-05	6.2500e-01	2.3283e-09
32	2.6557e+01	9.9644e-06	3.1250e-01	1.1642e-09

Table 4: Output for $p = 1.50$ using dual scaling.

iteration	network-cost	duality-gap	dstep-max	pstep-max
1	3.5099e+02	3.5513e+02	3.1250e-01	9.7656e-03
2	2.6731e+02	2.6493e+02	1.5625e-01	9.7656e-03
3	2.0753e+02	1.9761e+02	1.5625e-01	9.7656e-03
4	1.1948e+02	1.0726e+02	3.1250e-01	4.8828e-03
5	7.5355e+01	6.1541e+01	3.1250e-01	4.8828e-03
6	5.2555e+01	4.2327e+01	3.1250e-01	4.8828e-03
7	4.2703e+01	2.6665e+01	3.1250e-01	2.4414e-03
8	3.5369e+01	1.6715e+01	3.1250e-01	2.4414e-03
9	3.1385e+01	1.1311e+01	3.1250e-01	6.1035e-04
10	2.9399e+01	6.6423e+00	3.1250e-01	6.1035e-04
11	2.7736e+01	3.4730e+00	3.1250e-01	6.1035e-04
12	2.6586e+01	2.0449e+00	3.1250e-01	3.0518e-04
13	2.6111e+01	1.3087e+00	3.1250e-01	7.6294e-05
14	2.5840e+01	7.2836e-01	3.1250e-01	7.6294e-05
15	2.5625e+01	3.6757e-01	3.1250e-01	7.6294e-05
16	2.5489e+01	2.1680e-01	3.1250e-01	3.8147e-05
17	2.5439e+01	1.3341e-01	3.1250e-01	9.5367e-06
18	2.5408e+01	8.5911e-02	3.1250e-01	4.7684e-06
19	2.5390e+01	5.8265e-02	3.1250e-01	2.3842e-06
20	2.5379e+01	3.6359e-02	3.1250e-01	2.3842e-06
21	2.5370e+01	2.3864e-02	3.1250e-01	1.1921e-06
22	2.5365e+01	1.7729e-02	3.1250e-01	2.9802e-07
23	2.5363e+01	1.1680e-02	3.1250e-01	5.9605e-07
24	2.5361e+01	8.1149e-03	3.1250e-01	2.9802e-07
25	2.5359e+01	5.2303e-03	3.1250e-01	2.9802e-07
26	2.5358e+01	2.9632e-03	3.1250e-01	2.9802e-07
27	2.5357e+01	1.9414e-03	3.1250e-01	7.4506e-08
28	2.5357e+01	1.2411e-03	3.1250e-01	7.4506e-08
29	2.5357e+01	7.0021e-04	3.1250e-01	7.4506e-08
30	2.5356e+01	3.7630e-04	3.1250e-01	7.4506e-08
31	2.5356e+01	2.2183e-04	3.1250e-01	3.7253e-08
32	2.5356e+01	1.4140e-04	3.1250e-01	9.3132e-09
33	2.5356e+01	7.8081e-05	3.1250e-01	9.3132e-09
34	2.5356e+01	4.8133e-05	3.1250e-01	4.6566e-09
35	2.5356e+01	2.9645e-05	3.1250e-01	2.3283e-09
36	2.5356e+01	1.5629e-05	3.1250e-01	2.3283e-09
37	2.5356e+01	8.3688e-06	3.1250e-01	4.6566e-09

Table 5: Output for $p = 2.00$ using dual scaling.

iteration	network-cost	duality-gap	dstep-max	pstep-max
1	3.0811e+02	3.1301e+02	1.5625e-01	1.9531e-02
2	2.3295e+02	2.3135e+02	1.5625e-01	9.7656e-03
3	1.7991e+02	1.7375e+02	1.5625e-01	4.8828e-03
4	1.4208e+02	1.3316e+02	1.5625e-01	2.4414e-03
5	8.4541e+01	7.0951e+01	3.1250e-01	4.8828e-03
6	6.9270e+01	5.8281e+01	1.5625e-01	4.8828e-03
7	4.8602e+01	3.6038e+01	3.1250e-01	4.8828e-03
8	3.8005e+01	2.4417e+01	3.1250e-01	1.2207e-03
9	3.2760e+01	1.5201e+01	3.1250e-01	1.2207e-03
10	2.9182e+01	9.9133e+00	3.1250e-01	6.1035e-04
11	2.7335e+01	6.7457e+00	3.1250e-01	3.0518e-04
12	2.6230e+01	4.0957e+00	3.1250e-01	3.0518e-04
13	2.5345e+01	2.6476e+00	3.1250e-01	1.5259e-04
14	2.4854e+01	1.4456e+00	3.1250e-01	1.5259e-04
15	2.4433e+01	6.9807e-01	3.1250e-01	1.5259e-04
16	2.4164e+01	4.1758e-01	3.1250e-01	3.8147e-05
17	2.4079e+01	2.4954e-01	3.1250e-01	1.9073e-05
18	2.4018e+01	1.5750e-01	3.1250e-01	9.5367e-06
19	2.3985e+01	8.6084e-02	3.1250e-01	9.5367e-06
20	2.3958e+01	4.3858e-02	3.1250e-01	9.5367e-06
21	2.3942e+01	2.5412e-02	3.1250e-01	4.7684e-06
22	2.3936e+01	1.5580e-02	3.1250e-01	1.1921e-06
23	2.3932e+01	9.9903e-03	3.1250e-01	5.9605e-07
24	2.3930e+01	5.6315e-03	3.1250e-01	5.9605e-07
25	2.3928e+01	2.9602e-03	3.1250e-01	5.9605e-07
26	2.3927e+01	1.6010e-03	3.1250e-01	5.9605e-07
27	2.3927e+01	1.0595e-03	3.1250e-01	3.7253e-08
28	2.3926e+01	4.9997e-04	6.2500e-01	3.7253e-08
29	2.3926e+01	2.0940e-04	1.2500e+00	1.8626e-08
30	2.3926e+01	6.3721e-05	2.5000e+00	9.3132e-09
31	2.3926e+01	2.7372e-05	2.5000e+00	2.3283e-09
32	2.3926e+01	1.2168e-05	1.2500e+00	1.1642e-09
33	2.3926e+01	5.4989e-06	6.2500e-01	5.8208e-10

Table 6: Output for $p = 3.00$ using dual scaling.

iteration	network-cost	duality-gap	dstep-max	pstep-max
1	9.8742e+01	9.4987e+01	4.8828e-03	6.2500e-01
2	6.9168e+01	5.7033e+01	2.4414e-03	1.2500e+00
3	3.9986e+01	2.4935e+01	2.4414e-03	1.2500e+00
4	2.7451e+01	1.0022e+01	1.2207e-03	6.2500e-01
5	2.5071e+01	6.3698e+00	3.0518e-04	3.1250e-01
6	2.3931e+01	4.3982e+00	1.5259e-04	3.1250e-01
7	2.2760e+01	2.2091e+00	1.5259e-04	6.2500e-01
8	2.2027e+01	1.2413e+00	7.6294e-05	6.2500e-01
9	2.1698e+01	7.2307e-01	3.8147e-05	6.2500e-01
10	2.1342e+01	2.8390e-01	3.8147e-05	6.2500e-01
11	2.1243e+01	1.2907e-01	1.9073e-05	3.1250e-01
12	2.1218e+01	8.2945e-02	9.5367e-06	1.5625e-01
13	2.1203e+01	5.1531e-02	9.5367e-06	1.5625e-01
14	2.1201e+01	3.7818e-02	2.3842e-06	1.5625e-01
15	2.1193e+01	1.8277e-02	2.3842e-06	3.1250e-01
16	2.1190e+01	1.1403e-02	5.9605e-07	3.1250e-01
17	2.1185e+01	5.6478e-03	5.9605e-07	3.1250e-01
18	2.1184e+01	2.8930e-03	2.9802e-07	3.1250e-01
19	2.1183e+01	1.4873e-03	1.4901e-07	3.1250e-01
20	2.1183e+01	7.7856e-04	7.4506e-08	3.1250e-01
21	2.1183e+01	4.1633e-04	3.7253e-08	3.1250e-01
22	2.1183e+01	2.2987e-04	1.8626e-08	3.1250e-01
23	2.1182e+01	1.3274e-04	9.3132e-09	3.1250e-01
24	2.1182e+01	5.1656e-05	9.3132e-09	3.1250e-01
25	2.1182e+01	2.6681e-05	9.3132e-09	1.5625e-01
26	2.1183e+01	2.7794e-04	1.4901e-07	1.5625e-01
27	2.1183e+01	1.6209e-04	9.3132e-09	5.0000e+00
28	2.1182e+01	9.0529e-05	4.6566e-09	1.2500e+00
29	2.1182e+01	5.8100e-05	2.3283e-09	2.5000e+00
30	2.1182e+01	2.7823e-05	2.3283e-09	1.2500e+00
31	2.1182e+01	1.3559e-05	1.1642e-09	6.2500e-01
32	2.1182e+01	6.2516e-06	5.8208e-10	6.2500e-01

Table 7: Output for $p = 101$ $(q = 1.01)$ using primal scaling.

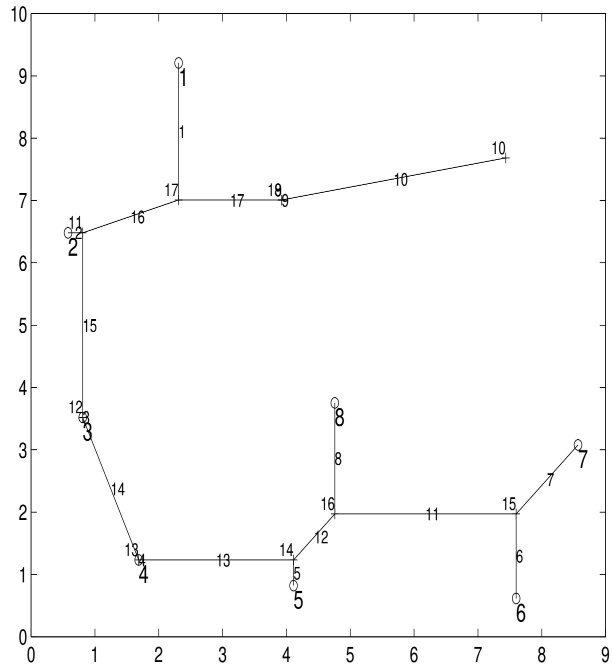

Figure 1: The shortest network for $p = 1.01$.

The shortest networks for the five cases are plotted in Figures 1-5, where regular points are labeled by + and Steiner points are labeled by o. It is interesting to see the slight change in the network when p is changed from 1.01 to 1.5 and eventually to 101.

8 Conclusions

In this paper, we have transformed the problem of minimizing a sum of p-norms into a standard convex programming problem in conic forms. Unlike those in most convex optimization problems, the cone for this problem is not self-dual. We have constructed two barrier functions and studied its associated parameters. Using these barrier functions, we have presented a polynomial time primal-dual potential reduction algorithm for solving this problem. In particular, the number of iterations required to produce an ϵ-optimal solution is at most $O(\sqrt{md}(\log(\bar{c}/\epsilon) + \log(md)))$. As applications, we have shown that computing an ϵ-optimal solution of the shortest p-norm network under a tree topology interconnecting N regular points on the Euclidean plane requires only $O(N^{1.5}(\log(\bar{c}/\epsilon) + \log N))$ arithmetic operations, where $\bar{c}$ is the largest pair wise Euclidean distance among the given point set. Our implementation is only preliminary. Computational issues of our algorithm are under investigation and will be reported in another paper.

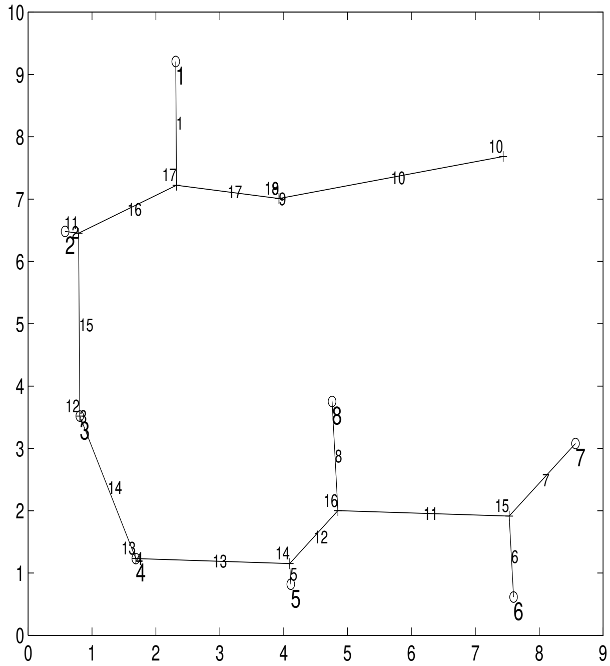

Figure 2: The shortest network for $p = 1.50$.

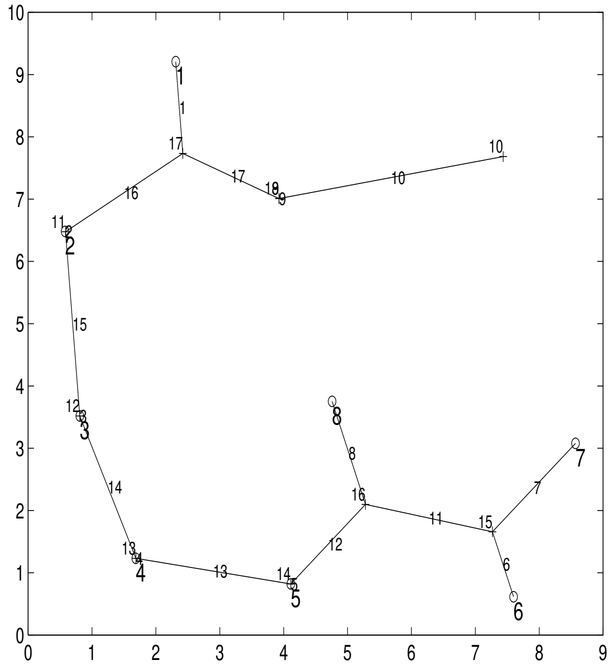

Figure 3: The shortest network for $p = 2.00$.

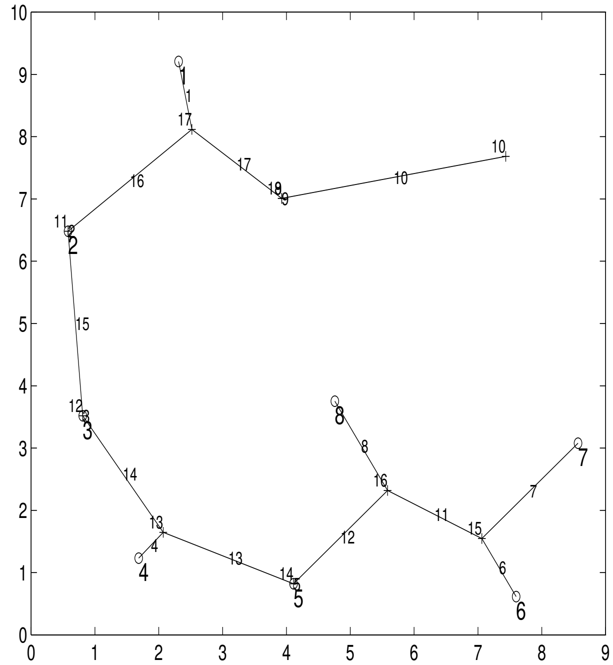

Figure 4: The shortest network for $p = 3.00$.

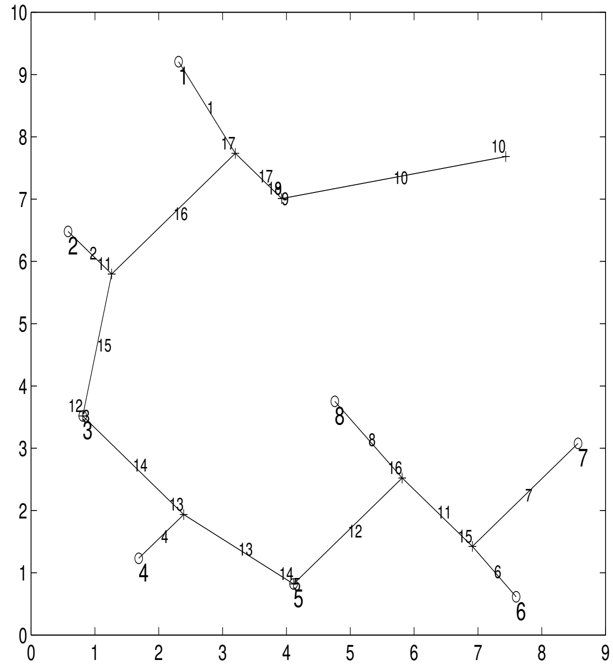

Figure 5: The shortest network for $p = 101$.

References

[1] K.D. Andersen, An efficient Newton barrier method for minimizing a sum of Euclidean norms, *SIAM Journal on Optimization*, Vol. 6(1996), pp. 74-95.

[2] K.D. Andersen and E. Christiansen, A symmetric primal-dual Newton method for minimizing a sum of norms, manuscript, Odense University, 5230 Odense M, Denmark, 1995.

[3] P.H. Calamai and A.R. Conn, A second-order method for solving the continuous multifacility location problem, in Watson, G.A. ed., *Numerical Analysis: Proceedings of the Ninth Biennial Conference, Dundee, Scotland*, Lecture Notes in Mathematics 912, Springer-Verlag(1982), pp. 1-25.

[4] P.H. Calamai and A.R. Conn, A projected Newton method for l_p norm location problems, *Mathematical Programming*, Vol. 38(1987), pp. 75-109.

[5] R. Chandrasekaran and A. Tamir, Open questions concerning Weiszfeld's algorithm for the Fermat-Weber location problem, *Mathematical Programming*, Vol. 44(1989), pp. 293-295.

[6] R. Chandrasekaran and A. Tamir, Algebraic optimization: the Fermat-Weber location problem, *Mathematical Programming,* Vol. 46(1990), pp. 219-224.

[7] A.R. Conn and M.L. Overton, A primal-dual interior point method for minimizing a sum of Euclidean distances, in preparation, 1995.

[8] D. den Hertog, *Interior Point Approach to Linear, Quadratic and Convex Programming,* Kluwer Academic Publishers, 1994.

[9] D. den Hertog, F. Jarre, C. Roos, and T. Terlaky, A sufficient condition for self-concordance, with application to some classes of structured convex programming problems, *Mathematical Programming,* Vol. 69(1995), pp. 75-88.

[10] J.W. Eyster, J.A. White and W.W. Wierwille, On solving multifacility location problems using a hyperboloid approximation procedure, *AIIE Transactions*, Vol. 5(1973), pp. 1-6.

[11] R.L. Francis and A.V. Cabot, Properties of a multifacility location problem involving Euclidean distances, *Naval Research Logistics Quarterly,* Vol. 19(1972), pp. 335-353.

[12] R.L. Francis, Leon F. McGinnis, Jr. and John A. White, *Facility Layout and Location: An Analytical Approach*, Prentice Hall, 1991.

[13] E.N. Gilbert and H.O. Pollak, Steiner minimal trees, *SIAM Journal on Applied Mathematics*, Vol. 16(1968), pp. 1-29.

[14] G.H. Golub and C.F. van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, Maryland, 1983.

[15] O. Güler, On the self-concordance of the universal barrier function, *SIAM Journal on Optimization*, Vol. 7(1997), pp. 295-303.

[16] F.K. Hwang, A linear time algorithm for full Steiner trees, *Operations Research Letters*, Vol. 4(1986), pp. 235-237.

[17] F.K. Hwang and J.F. Weng, The shortest network under a given topology, *Journal of Algorithms*, Vol. 13(1992), pp. 468-488.

[18] M. Kojima, S. Shindoh, and S. Hara Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. Research Reports on Information Sciences, No. B-282, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan, 1994, Revised April 1995.

[19] H.W. Kuhn, On a pair of dual nonlinear programs, in J. Abadie eds, *Nonlinear Programming,* pp. 39-54, North-Holland, 1967.

[20] H.W. Kuhn, A note on Fermat's problem, *Mathematical Programming*, Vol. 4(1973), pp. 98-107.

[21] Y. Li, A Newton acceleration of the Weiszfeld algorithm for minimizing the sum of Euclidean distances, *TR95-1552, Computer Science Department, Cornell University*, November, 1995.

[22] R.F. Love, J.G. Morris and G.O. Wesolowsky, *Facilities Location: Models & Methods*, North-Holland, 1988.

[23] Z.A. Melzak, On the problem of Steiner, *Canadian Mathematical Bulletin*, Vol. 16(1961), pp. 143-148.

[24] W. Miehle, Link length minimization in networks, *Operations Research*, Vol. 6(1958), pp. 232-243.

[25] Yu. E. Nesterov and A. Nemirovskii, *Interior Polynomial Algorithms in Convex Programming,* SIAM, Philadelphia, 1994.

[26] Yu. E. Nesterov and M. J. Todd, Self-scaled barriers and interior-point methods for convex programming, *SIAM Journal on Optimization*, accepted for publication.

[27] L.M. Ostresh, The multifacility location problem: applications and descent theorems, *Journal of Regional Science,* Vol. 17(1977), pp. 409-419.

[28] M.L. Overton, A quadratically convergent method for minimizing a sum of Euclidean norms, *Mathematical Programming*, Vol. 27(1983), pp. 34-63.

[29] J.B. Rosen and G.L. Xue, On the convergence of Miehle's algorithm for the Euclidean multifacility location problem, *Operations Research*, Vol. 40(1992), pp. 188-191.

[30] J.B. Rosen and G.L. Xue, On the convergence of a hyperboloid approximation procedure for solving the perturbed Euclidean multifacility location problem, *Operations Research*, Vol. 41(1993), pp. 1164-1171.

[31] W.D. Smith, How to find Steiner minimal trees in Euclidean d-space, *Algorithmica*, Vol. 7(1992), pp. 137-177.

[32] C.Y. Wang et al., On the convergence and rate of convergence of an iterative algorithm for the plant location problem (in Chinese), *Qufu Shiyun Xuebao,* Vol. 2(1975), pp. 14-25.

[33] E. Weiszfeld, Sur le point par lequel le somme des distances de n points donnes est minimum, *Tohoku Mathematical Journal,* Vol. 43(1937), pp. 355-386.

[34] G.L. Xue, *Algorithms for Computing Extreme Points of Convex Hulls and the Euclidean Facilities Location Problem*, Ph.D Thesis, Computer Science Department, University of Minnesota, Minneapolis, MN 55455, 1991.

[35] G.L. Xue, J.B. Rosen and P.M. Pardalos, A polynomial time dual algorithm for the Euclidean multifacility location problem, in Proceedings of *Second Conference on Integer Programming and Combinatorial Optimization*, Pittsburgh, 1992, pp. 227-236.

[36] G.L. Xue and Y.Y. Ye, An efficient algorithm for minimizing a sum of Euclidean norms with applications, *SIAM Journal on Optimization*, accepted for publication.