An Interior-Point Method for Multi-Fractional Programs with Convex Constraints

Roland W. Freund
AT&T Bell Laboratories
600 Mountain Avenue, Room 2C–420
Murray Hill, New Jersey 07974–0636, U.S.A.

Florian Jarre
Institut für Angewandte Mathematik und Statistik
Universität Würzburg
Am Hubland
D–8700 Würzburg, Federal Republic of Germany

Abstract. We present an interior-point method for a family of multi-fractional programs with convex constraints. The programs under consideration consist of minimizing the maximum of a finite number of linear fractions over some convex set. First, we present a simple "short-step" algorithm for solving such multi-fractional programs. Then, we describe a practical implementation of the proposed method, and we report results of numerical experiments with this algorithm. These results show that the proposed method is a viable alternative to the standard Dinkelbach-type algorithms for solving multi-fractional programs. We also present some theoretical results for the short-step variant of our interior-point method. In particular, it is shown that, under additional assumptions, the short-step algorithm converges in polynomial time. Here the convergence rate depends very weakly on the data of the problem.

Key Words. Multi-fractional programming, convex set, interior-point method, short-step algorithm, predictor-corrector step, self-concordant barrier function, polynomial-time convergence.

AMS(MOS) subject classifications. 90C32, 90C60.

1. Introduction

Karmarkar's 1984 landmark paper [20] about a practical polynomial-time interior-point algorithm for linear programs triggered extensive research efforts into interior-point techniques for nonlinear optimization problems, and since 1984, numerous new algorithms have been proposed. However, in contrast to linear programs, interior-point methods for nonlinear programs converge in polynomial time only in special cases, such as classes of convex programming problems; we refer the reader to [1, 10, 17, 18, 22, 23, 28] and the references given there.

In [15], we developed a polynomial-time interior-point method for a class of fractional programs, namely the minimization of a single fraction of linear functions subject to convex constraints. We stress that such programs are not convex, but only pseudo-convex. In this paper, we extend the approach of [15], and we present a polynomial-time interior-point method for a class of *multi-fractional programs*, namely the minimization of the maximum of l linear fractions subject to convex constraints. We remark that the multi-fractional case $l > 1$ is especially important in practice, since many applications give rise to such programs; examples include von Neumann's model of an expanding economy [30], goal programming and multi-criteria optimization [25, 26, 27], and discrete rational Chebyshev approximation [2]. We implemented a primal predictor-corrector version of the proposed interior-point method, and we tested this algorithm on a variety of problems. The standard methods for solving multi-fractional programs are Dinkelbach's algorithm [11] and its variants; see [3, 4, 8, 9, 12, 26] and the references therein. The results of our numerical tests show that the proposed interior-point approach is a viable alternative to the usual Dinkelbach-type algorithms.

We also present some theoretical results for the short-step version of our interior-point method. In particular, we show that, under some additional assumptions, the algorithm converges in polynomial time. Here the rate of convergence is measured in terms of the area of suitably constructed two-dimensional convex sets $\mathcal{C}_{j,k}$. More precisely, at each iteration step k, the product of the areas of the l sets $\mathcal{C}_{j,k}$, $j = 1, 2, \ldots, l$, is reduced by a constant factor $c < 1$. Furthermore, this product is zero if, and only if, the iterate generated at the kth step is an optimal solution of the multi-fractional program. These theoretical results extend our earlier convergence proof [15] for the single-fraction case $l = 1$ to the multi-fractional case $l > 1$. As in [15], our analysis is based on the concept of self-concordant barrier functions introduced by Nesterov and Nemirovsky [23]. In particular, the reduction factor c in our convergence result depends only on the self-concordance parameter of the logarithmic barrier function associated with the feasible set of the multi-fractional program. The second key property that is used in our analysis is a Lipschitz continuity of the gradient and the Hessian of the barrier function.

We remark that interior-point methods for solving a class of fractional programming problems with convex constraints are also presented in [24] and [5]; however, neither paper gives a rigorous complexity analysis. The present paper differs substantially from [5, 24] in several respects: the proposed method is different; we give a complete complexity analysis; and we present a numerical comparison of our approach with standard algorithms for multi-fractional programming.

1.1. The Problem in Standard Form

We consider multi-fractional problems of the following standard form:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \max_{j=1,2,\ldots,l} \frac{x_{2j}}{x_{2j-1}} \qquad \text{subject to} \quad f_i(x) \leq 0, \quad i = 1, 2, \ldots, s. \tag{1.1}$$

Here, it is assumed that $l \geq 1$ and $n \geq 2l$, and x_{2j-1} and x_{2j} denote the $(2j-1)$st and $(2j)$th component of $x \in \mathbb{R}^n$. Furthermore, $f_i : \mathbb{R}^n \mapsto \mathbb{R} \cup \infty$, $i = 1, 2, \ldots, s$, are constraint functions such that the feasible domain

$$\mathcal{S} := \{\, x \in \mathbb{R}^n \mid f_i(x) \leq 0,\ i = 1, 2, \ldots, s \,\} \tag{1.2}$$

of (1.1) is convex. We always assume that $\mathcal{S}$ has a nonempty interior. Note that the objective function

$$f_0(x) = \max_{j=1,2,\ldots,l} \frac{x_{2j}}{x_{2j-1}}$$

of (1.1) is not convex, and thus (1.1) is not a convex optimization problem. Furthermore, if the program (1.1) is "degenerate", i.e., the feasible set $\mathcal{S}$ contains a point $\hat{x}$ with $\hat{x}_i = 0$ for all $i = 1, 2, \ldots, 2l$, then the minimum of problem (1.1) does not exist—only its infimum exists. We stress that we do not exclude this degenerate situation, and that our analysis covers this case as well.

1.2. Applications

Multi-fractional programs of the standard form (1.1) arise in important applications. First, consider optimization problems of the type

$$\underset{x\in\mathbb{R}^n}{\text{maximize}} \quad \min_{j=1,2,\ldots,l} \frac{g_j(x)}{h_j(x)} \qquad \text{subject to} \quad x \in \mathcal{S}. \tag{1.3}$$

Here, $\mathcal{S}$ is a set of the from (1.2), and the denominator functions h_j, $j = 1, 2, \ldots, l$, are assumed to satisfy $h_j(x) > 0$ for all feasible points $x \in \mathcal{S}$. The problem (1.3) is called a *concave generalized fractional program* [26] if the numerator functions g_j are all concave, and all denominator functions h_j as well as all constraint functions f_i in (1.2) are convex. Concave generalized fractional programs arise in a variety of applications, for example, in economics; see [4, 12, 25, 26, 27, 30] and the references given there. It is straightforward to convert problem (1.3) to a problem of the standard form (1.1) by introducing additional slack variables $x_1, x_2, \ldots, x_{2l}$ and constraints

$$x_{2j-1} - g_j(x) \le 0 \quad \text{and} \quad h_j(x) - x_{2j} \le 0 \quad \text{for all} \quad j = 1, 2, \ldots, l;$$

see [15].

A second class of problems that lead to programs of the type (1.1) is discrete rational Chebyshev approximation [2, 6]. Here, one wishes to solve programs of the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \max_{j=1,2,\ldots,l} \left| \frac{f_j(x)}{g_j(x)} - r_j \right| \qquad \text{subject to} \quad x \in \mathcal{S}, \tag{1.4}$$

where f_j and g_j are linear functions of x, and $r_j \in \mathbb{R}$ are given target values. Furthermore, it is assumed that $g_j(x) > 0$ for all $j = 1, 2, \ldots, l$ and all feasible points $x \in \mathcal{S}$. A special case of (1.4) is the problem of approximating a given function $h(z)$ by a rational function $p(z)/q(z)$ on a discrete point set:

$$\underset{p\in\Pi_N,\ q\in\Pi_M}{\text{minimize}} \quad \max_{j=1,2,\ldots,l} \left| \frac{p(z_j)}{q(z_j)} - h(z_j) \right|. \tag{1.5}$$

Here, Π_N and Π_M denote the spaces of real polynomials of degree at most N and M, respectively, or some other linear spaces of dimension $N+1$ and $M+1$, respectively, and $z_1, z_2, \ldots, z_l \in \mathbb{R}$ are given points. The problem (1.5) is a special case of (1.4) with $n = N + M + 2$ and $r_j = h(z_j)$, $j = 1, 2, \ldots, l$. Clearly, any discrete rational Chebyshev approximation problem (1.4) can be rewritten as the following program:

$$\begin{aligned}
&\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \max_{j=1,2,\ldots,l} \frac{\mu_j}{\lambda_j} \\
&\text{subject to} \quad x \in \mathcal{S}, \\
&\qquad\qquad -\mu_j \le f_j(x) - r_j g_j(x) \le \mu_j \quad \text{for all} \quad j = 1, 2, \ldots, l, \\
&\qquad\qquad 0 \le \lambda_j \le g_j(x) \quad \text{for all} \quad j = 1, 2, \ldots, l,
\end{aligned}$$

which is just a multi-fractional program of the type (1.1).

1.3. Outline

The remainder of this paper is organized as follows. In Section 1.4, we introduce some notation, and in Section 1.5, we formulate our basic assumptions on (1.1). In Section 2, we present a short-step variant of our interior-point method for multi-fractional programs (1.1). In Section 3, we sketch a practical implementation, and in Section 4, results of numerical experiments with this algorithm are reported. In Section 5, we first recall some known results about self-concordant functions, and then we formulate additional assumptions that are needed for our theoretical investigations. In Section 6, we derive some basic properties of the short-step algorithm. In Section 7, we show that this algorithm converges in polynomial time. Finally, in Section 8, we make some concluding remarks.

1.4. Notation

Throughout this paper, all vectors and matrices are assumed to be real. We will use the following notation. We denote by $M^T = [\, m_{kj}\,]$ the transpose of the matrix $M = [\, m_{jk}\,]$. The vector norm $\|x\| := \sqrt{x^T x}$ is always the Euclidean norm, and $\|M\| := \max_{\|x\|=1} \|Mx\|$ is the associated matrix norm. For symmetric positive definite matrices M, we denote by $\|x\|_M := \sqrt{x^T M x}$ the vector norm induced by M. For symmetric matrices M and N, the notation $M > N$, respectively $M \geq N$, is used to indicate that $M - N$ is positive definite, respectively positive semidefinite. For vectors $x \in I\!R^n$, $n \geq 2$, we always denote by x_i the ith component of x, and

$$x_{[j]} := \begin{bmatrix} x_{2j-1} \\ x_{2j} \end{bmatrix} \in I\!R^2, \quad 1 \leq j \leq n/2,$$

is the truncated vector with components x_{2j-1} and x_{2j}. The symbol $0_{(p)}$ is used for the zero vector in $I\!R^p$. Finally, for sets $\mathcal{C} \in I\!R^m$, we denote by $\mathcal{C}^\circ$ the interior (with respect to the Euclidean topology of $I\!R^m$) of $\mathcal{C}$.

1.5. Basic Assumptions

We now state our assumptions on the multi-fractional program (1.1) and its feasible domain $\mathcal{S}$ defined in (1.2).

Throughout this paper, $\mathcal{S}$ is assumed to be a bounded[1] convex set in $I\!R^n$ with nonempty interior

$$\mathcal{S}^\circ \neq \emptyset. \tag{1.6}$$

Moreover, we assume that the denominators in (1.1) are nonnegative for all feasible points:

$$x_{2j-1} \;\geq\; 0 \quad \text{for all} \quad j = 1, 2, \ldots, l \quad \text{and} \quad x \in \mathcal{S}. \tag{1.7}$$

We associate with $\mathcal{S}$ the *logarithmic barrier function* ϕ defined by

$$\phi(x) := -\sum_{i=1}^{s} \log(-f_i(x)) \quad \text{for all} \quad x \in \mathcal{S}^\circ. \tag{1.8}$$

We assume that the function ϕ is strictly convex on $\mathcal{S}^\circ$. The constraint functions f_i, $i = 1, 2, \ldots, s$, in (1.2) are required to be twice continuously differentiable on $\mathcal{S}^\circ$. This guarantees that the gradient $g = \nabla\phi$ and the Hessian $H = \nabla^2\phi$ of the barrier function (1.8) exist. Note that, in view of the strict convexity of ϕ, the matrix H is positive definite for all $x \in \mathcal{S}^\circ$.

[1]For the practical implementation described in Section 3, the boundedness of $\mathcal{S}$ is not a crucial assumption, and it is only made in order to keep the exposition simple. In fact, $\mathcal{S}$ is unbounded for the examples in Section 4.4.

2. A Method of Centers

In this section, we derive a short-step interior-point method for multi-fractional programs. We stress that this algorithm is mainly of theoretical interest, since it allows a simple proof of polynomial convergence. A practical "long-step" implementation of the method will be sketched in Section 3.

2.1. An Equivalent Formulation

Our interior-point method is based on the following reformulation of the multi-fractional program (1.1). For each $j = 1, 2, \ldots, l$, we set

$$v_j(t) := \begin{bmatrix} 0_{(2j-2)} \\ -\sin t \\ \cos t \\ 0_{(n-2j)} \end{bmatrix} \in \mathbb{R}^n \quad \text{and} \quad u_j(x,t) := v_j(t)^T x \quad \text{for all} \quad x \in \mathbb{R}^n, \ t \in \mathbb{R}. \tag{2.1}$$

In view of (1.7), we have, for all $x \in \mathcal{S}^\circ$ and $t \in (-\pi/2, \pi/2]$,

$$u_j(x,t) \ \leq \ 0 \quad \Longleftrightarrow \quad \frac{x_{2j}}{x_{2j-1}} \ \leq \ \tan t. \tag{2.2}$$

Thus the multi-fractional program (1.1) is equivalent to the following problem:

$$\begin{aligned} &\underset{t \in (-\pi/2, \pi/2],\, x \in \mathbb{R}^n}{\text{minimize}} \quad t \\ &\text{subject to} \quad x \in \mathcal{S} \quad \text{and} \quad u_j(x,t) \ \leq \ 0 \quad \text{for all} \quad j = 1, 2, \ldots, l. \end{aligned} \tag{2.3}$$

In the remainder of this paper, we consider the solution of problem (2.3).

Next, we introduce some further notation. For each fixed t, we denote by

$$\mathcal{S}(t) := \{\, x \in \mathcal{S} \mid u_j(x,t) \leq 0 \text{ for all } j = 1, 2, \ldots, l \,\}$$

the set of points x that are feasible for (2.3), and by

$$\mathcal{S}^\circ(t) := \{\, x \in \mathcal{S}^\circ \mid u_j(x,t) < 0 \text{ for all } j = 1, 2, \ldots, l \,\} \tag{2.4}$$

the interior of $\mathcal{S}(t)$. Note that $u_j(x, \pi/2) = -x_{2j-1}$, and by (1.7) and (1.6), we have

$$\mathcal{S}(\pi/2) = \mathcal{S} \quad \text{and} \quad \mathcal{S}^\circ(\pi/2) \neq \emptyset. \tag{2.5}$$

Furthermore, we always denote by $t_{\text{opt}} \in [-\pi/2, \pi/2]$ the optimal value of (2.3). We stress that there are two cases. If the minimum in (2.3) exists, then $t_{\text{opt}} > -\pi/2$. However, it is not excluded that only the infimum in (2.3) exists, and then $t_{\text{opt}} = -\pi/2$. Obviously, in both cases, we have

$$\mathcal{S}(t_{\text{opt}}) = \Big\{\, x \in \mathcal{S} \,\Big|\, \max_{j=1,2,\ldots,l} u_j(x, t_{\text{opt}}) = 0 \Big\}, \quad \mathcal{S}^\circ(t_{\text{opt}}) = \emptyset.$$

Moreover, if $t_{\text{opt}} > -\pi/2$, then $\mathcal{S}(t_{\text{opt}})$ is just the set of optimal solutions of the multi-fractional program (1.1). If $t_{\text{opt}} = -\pi/2$, then the "optimal value" of (1.1) is $-\infty$, and formally this value is attained for all points in the set

$$\{\, x \in \mathcal{S} \mid x_{2j-1} = 0 \text{ and } x_{2j} \leq 0 \text{ for all } j = 1, 2, \ldots, l \,\}.$$

Finally, we remark that, in both cases,

$$\mathcal{S}^\circ(t) \neq \emptyset \quad \text{if} \quad t_{\text{opt}} < t \leq \pi/2, \quad \text{and} \quad \mathcal{S}(t) = \emptyset \quad \text{if} \quad -\pi/2 \leq t < t_{\text{opt}}. \tag{2.6}$$

In the following, let t with $t_{\text{opt}} < t \leq \pi/2$ be arbitrary, but fixed. Clearly, $\mathcal{S}(t)$ is a convex set whose interior, by (2.6), is nonempty. We define a logarithmic barrier function $\varphi(\cdot, t)$ for $\mathcal{S}^\circ(t)$ by setting

$$\varphi(x,t) := \phi(x) - m \sum_{j=1}^{l} \log(-u_j(x,t)) \quad \text{for all} \quad x \in \mathcal{S}^\circ(t). \tag{2.7}$$

Here, ϕ is the logarithmic barrier function for $\mathcal{S}$ given by (1.8), and the factor m in front of the second term in (2.7) is a suitably chosen positive constant. For our numerical experiments we chose $m = s$, the number of constraints in (1.1). The significance of the choice of m for our theoretical investigations will become apparent in Sections 5–7; in particular, in order to optimize the reduction factor in our convergence result, we will use a specific value of m in Section 7. In the following, we denote by $g(\cdot, t)$ and $H(\cdot, t)$ the gradient and the Hessian of $\varphi(\cdot, t)$, respectively. In view of (2.7) and (2.1), g and H are given by

$$g(x,t) := \nabla_x \varphi(x,t) = \nabla \phi(x) - m \sum_{j=1}^{l} \frac{v_j(t)}{u_j(x,t)}, \tag{2.8}$$

$$H(x,t) := \nabla_x^2 \varphi(x,t) = \nabla^2 \phi(x) + m \sum_{j=1}^{l} \frac{v_j(t) v_j(t)^T}{u_j^2(x,t)}. \tag{2.9}$$

Recall that the first term ϕ in (2.7) is assumed to be strictly convex. Since the second term is always convex, it follows that $\varphi(\cdot, t)$ is strictly convex on $\mathcal{S}^\circ(t)$. Note that $\varphi(x,t) \to \infty$ as x approaches the boundary of $\mathcal{S}(t)$. Therefore, it is guaranteed that the unconstrained optimization problem

$$\underset{x \in \mathcal{S}(t)}{\text{minimize}} \;\; \varphi(x,t) \tag{2.10}$$

always has a unique optimal solution, which we denote by $x(t)$. The point $x(t)$ is called the *analytic center* of the set $\mathcal{S}(t)$; see [28]. It can be shown that $x(t)$ converges to a point in $\mathcal{S}(t_{\text{opt}})$ as $t \to t_{\text{opt}} + 0$, and hence

$$x^{\text{opt}} := \lim_{t \to t_{\text{opt}} + 0} x(t) \tag{2.11}$$

is an optimal solution of the original multi-fractional program (1.1).

2.2. A Short-Step Method

The property (2.11) of the analytic centers $x(t)$ suggests the following procedure for computing an optimal solution of the multi-fractional program (1.1). We choose a sequence

$$\{t_k\}_{k=0}^{\infty}, \quad \text{where} \quad t_{\text{opt}} < t_{k+1} < t_k \leq \pi/2 \quad \text{for all} \quad k,$$

and we determine for each k an approximation—denoted by $x^{(k)}$ in the sequel[2]—to the analytic center $x(t_k)$ of $\mathcal{S}(t_k)$. At the start of our method, we choose a t_0 such that $-\pi/2 < t_0 \leq \pi/2$ and

[2]Quantities corresponding to the kth iteration of this procedure are always marked by subscripts k, with the exception of the iterates $x^{(k)}$ for which superscripts (k) are used. This is done to avoid confusion with the notation x_i for the ith component of a vector x.

$\mathcal{S}^\circ(t_0) \neq \emptyset$. For example, by (2.5), one can set $t_0 = \pi/2$. Then, we determine a point $x^{(0)} \in \mathcal{S}^\circ(t_0)$ that is "close" to the analytic center of $\mathcal{S}(t_0)$ in the following sense:

$$\|g(x^{(0)}, t_0)\|_{(H(x^{(0)},t_0))^{-1}} \;\leq\; \frac{1}{16}. \tag{2.12}$$

A point $x^{(0)}$ satisfying (2.12) can easily be obtained in polynomial time by using any of the algorithms for convex programming in [18, 23]. At each kth iteration, our algorithm reduces the current value t_k by setting $t_{k+1} = t_k - \Delta t_k$, where the step size $\Delta t_k > 0$ is chosen based on the kth iterate $x^{(k)}$. Then we perform one step of Newton's method (without line search) for minimizing $\varphi(\cdot, t_{k+1})$ over $\mathcal{S}(t_{k+1})$ with the current iterate $x^{(k)}$ as starting point. The resulting Newton iterate is $x^{(k+1)}$. As we will see in Section 6, the iterates $x^{(k)}$ generated by this procedure remain close to the analytic center $x(t_k)$ in the sense that

$$\|g(x^{(k)}, t_k)\|_{(H(x^{(k)},t_k))^{-1}} \;\leq\; \frac{1}{16} \quad \text{for all} \quad k = 0, 1, \dots; \tag{2.13}$$

see Theorem 6.5. Note that, in (2.13), the kth gradient is measured in the $(H(x^{(k)}, t_k))^{-1}$-norm. Moreover, we remark that, in view of (2.10), $g(x(t_k), t_k) = 0$, and hence the left-hand side in (2.13) would be zero if $x^{(k)}$ were chosen as the analytic center $x(t_k)$ of $\mathcal{S}(t_k)$.

In the sequel, it will be convenient to use the following notation. We set

$$r_j(x,t) := \frac{|u_j(x,t)|}{\|x_{[j]}\|} \quad \text{and} \quad \mathbf{r}(x,t) := \min_{j=1,2,\dots,l} r_j(x,t) \tag{2.14}$$

for $x \in \mathcal{S}^\circ$ and $t \in (-\pi/2, \pi/2]$. Here, $u_j(x,t)$, $j = 1, 2, \dots, l$, are the functions defined in (2.1). Furthermore, we set

$$\begin{aligned}
g_k &:= g(x^{(k)}, t_k), & \bar{g}_k &:= g(x^{(k)}, t_{k+1}), & (2.15)\\
H_k &:= H(x^{(k)}, t_k), & \bar{H}_k &:= H(x^{(k)}, t_{k+1}), & (2.16)\\
B_k &:= H_k^{-1}, & \bar{B}_k &:= \bar{H}_k^{-1}, & (2.17)\\
\mathbf{r}_k &:= \mathbf{r}(x^{(k)}, t_k), & \bar{\mathbf{r}}_k &:= \mathbf{r}(x^{(k)}, t_{k+1}).
\end{aligned}$$

Our short-step algorithm can then be formulated as follows. Recall that $l \geq 1$ is the number of fractions in (1.1), and $m > 0$ is the constant that appears in the definition (2.7) of the barrier function $\varphi(\cdot, t)$.

Algorithm 2.1. (Short-Step Interior-Point Method for Multi-Fractional Programs.)

0) Choose $t_0 \in (-\pi/2, \pi/2]$ such that $\mathcal{S}^\circ(t_0) \neq \emptyset$.
   Determine a point $x^{(0)} \in \mathcal{S}^\circ(t_0)$ such that

$$\|g_0\|_{B_0} \;\leq\; \frac{1}{16}.$$

For $n = 0, 1, \dots,$ do:

1) Choose $\sigma_k > 0$ and set

$$\Delta t_k = \frac{\sigma_k\, \mathbf{r}_k}{l\, m}. \tag{2.18}$$

2) Set

$$t_{k+1} = t_k - \Delta t_k. \tag{2.19}$$

3) Set

$$x^{(k+1)} = x^{(k)} - \bar{B}_k \, \bar{g}_k. \tag{2.20}$$

We remark that the quantities generated in Algorithm 2.1 are only defined if we can guarantee that $t_{\rm opt} < t_k$ and $x^{(k)} \in \mathcal{S}^\circ(t_k)$ for all k. In Theorem 6.5 below, we will show that, under certain additional assumptions, t_k and $x^{(k)}$ indeed satisfy these properties if $\sigma_k \le 1/18$ for all k. Obviously, it is desirable to always choose the maximal value $\sigma_k = 1/18$ in (2.18) so that the sequence $\{t_k\}_{k=0}^{\infty}$ decreases as fast as possible. As our main theoretical result, namely Theorem 7.3 below, we will prove that Algorithm 2.1 converges in polynomial time if $\sigma_k = 1/18$ for all k.

3. A Practical Implementation

Despite its nice feature of polynomial convergence, Algorithm 2.1 needs to be "accelerated" in order to be viable in practice. This is done by including a predictor step in an actual implementation of our interior-point method. First, we briefly motivate this predictor step, and then we present an outline of the resulting implementation.

3.1. The Predictor Step

By the implicit function theorem, the curve $x(t)$, $t > t_{\rm opt}$, is smooth. Furthermore, in view of (2.11), the point $x(t)$ approaches an optimal solution as $t \to t_{\rm opt}$. Fiacco and McCormick [13] showed that these properties also hold for the curves $\tilde{x}(t)$ where $\tilde{x}(t) = \arg\min_x(\varphi(x,t) + w^T x)$ with some constant linear perturbation $w^T x$.

Instead of computing $x^{(k+1)}$ by Newton's method starting from $x^{(k)}$, this suggests to use

$$\tilde{x}^{(k+1)} := x^{(k)} - \tilde{x}'(t_k)\,\Delta t_k \tag{3.1}$$

as an initial guess for Newton's iteration. Here, the linear perturbation w defining $\tilde{x}$ is chosen as $w = g(x^{(k)}, t_k)$ and depends on k. (It is chosen such that $\tilde{x}(t_k) = x^{(k)}$!) Recall from (2.8) that $g(x,t) = \nabla_x \varphi(x,t)$ denotes the gradient of $\varphi(\cdot, t)$. In order to be able to use (3.1), we need an approximation to the tangent vector $\tilde{x}'(t_k)$ of the curve $\tilde{x}(t)$ at $t = t_k$. Since $\tilde{x}(t)$ minimizes $\varphi(x,t) + w^T x$, and by (2.8), we have

$$-\,w = g(\tilde{x}(t), t) = \nabla\phi(\tilde{x}(t)) - m \sum_{j=1}^{l} \frac{v_j(t)}{u_j(\tilde{x}(t), t)} \quad \text{for all} \quad t. \tag{3.2}$$

Differentiating (3.2) with respect to t, it follows that

$$0 = \left(\nabla^2\phi(\tilde{x}(t))\right)\tilde{x}'(t) - m \sum_{j=1}^{l} \left(\frac{v_j'(t)}{u_j(\tilde{x}(t), t)} - \frac{v_j(t)\left((v_j'(t))^T \tilde{x}(t) + (v_j(t))^T \tilde{x}'(t)\right)}{(u_j(\tilde{x}(t), t))^2} \right) \tag{3.3}$$

for all t. For $t = t_k$, the relation (3.3) can be rewritten in the form

$$\begin{aligned} &\left(\nabla^2\phi(x^{(k)}) + m \sum_{j=1}^{l} \frac{v_j(t_k) v_j(t_k)^T}{(u_j(x^{(k)}, t_k))^2}\right) \tilde{x}'(t_k) \\ &\qquad = m \sum_{j=1}^{l} \left(\frac{v_j'(t_k)}{u_j(x^{(k)}, t_k)} - \frac{v_j(t_k)\,(v_j'(t_k))^T x^{(k)}}{(u_j(x^{(k)}, t_k))^2} \right). \end{aligned} \tag{3.4}$$

Hence we can compute $\tilde{x}'(t_k)$ by solving the nonsingular system of linear equations (3.4).

3.2. Implementable Algorithm

The implementation that is used for the numerical experiments in the next section can be outlined as follows:

- Let an initial point $x^{(0)} \in \mathcal{S}^o$ be given.
- Set $t_0 = \max\{\, 0.1, \max_{j=1,2,\ldots,l} \arctan(2x_{2j}^{(0)}/x_{2j-1}^{(0)})\,\}$.
- Use Newton's method to minimize $\varphi(x, t_0)$, and let the result be $x^{(1)}$. Set $k = 1$ and $t_1 = t_0$.
- (Outer iteration.) As long as the stopping criterion is not satisfied repeat the following steps:
  - Compute $\tilde{x}'(t_k)$ by solving the linear system (3.4).
  - Find $\Delta t_{\max} = \max\{\, \Delta t \in I\!R \mid x^{(k)} - \Delta t\, \tilde{x}'(t_k) \in \mathcal{S}(t_k - \Delta t)\,\}$.
  - Choose a reduction factor $\tau_k \in (0, 1)$ and set $\Delta t_k = \tau_k\, \Delta t_{\max}$.
  - (Extrapolation step.) Set $\tilde{x}^{(k+1)} = x^{(k)} - \Delta t_k\, \tilde{x}'(t_k)$ and $t_{k+1} = t_k - \Delta t_k$.
  - (Inner iteration.) Use Newton's method with line search and starting point $\tilde{x}^{(k+1)}$ to minimize $\varphi(x, t_{k+1})$. Let the result be $x^{(k+1)}$.
  - Set $k = k + 1$.

The extrapolation step is also called "affine scaling step" in the literature (for linear and convex programming), and Newton's method bringing the extrapolated point closer to the point $x(t_{k+1})$ on the central curve is referred to as "corrector step" or "centering step".

Next, we briefly discuss some technical details of the above algorithm.

- The following stopping criterion is used for the outer iteration of the algorithm:

  $$|\mathbf{r}(x^{(k)}, t_k)| \;\le\; \mathtt{tol}, \tag{3.5}$$

  where `tol` is some convergence tolerance. We note that the objective function of the program (1.1) is strictly quasi-convex, and this guarantees that the *Karush-Kuhn-Tucker conditions* are necessary and sufficient for optimality of program (1.1); see, e.g., Mangasarian [21]. As shown in [15], the test (3.5) represents a "naturally" scaled version of the Karush-Kuhn-Tucker conditions.

- As stopping criterion for Newton's method (inner iteration) we chose

  $$\|g(x,t)\|_{(H(x,t))^{-1}} \;\le\; \epsilon_1 := \frac{1}{2}. \tag{3.6}$$

  Here, the choice of $\epsilon_1 < 1$ is arbitrary; the value $\frac{1}{2}$ was found practical in our test runs.

- The step length $\tau_k < 1$ for the extrapolation is initially set to 0.9, and is changed during the algorithm according to the following rule. If Newton's method in the previous iteration took more than 3 steps, then τ is reduced by setting $\tau_{k+1} = \max\{0.5,\ 2\tau_k - 1\}$, and if Newton's method converged in less than 2 iterations, then we set $\tau_{k+1} = (\tau_k + 1)/2$. This adaptive procedure is intended to keep the step length τ_k for the extrapolation as large as possible so that Newton's method still converges rapidly at each iteration.

- As pointed out in the introduction, a concave generalized fractional program (1.3) can be converted to a problem of the standard form (1.1). For our implementation, it turned out to be more efficient to solve the problem in its original form (1.3) directly by our interior-point method, instead of rewriting it in standard form (1.1). In this case, the "objective constraints" in (2.3), namely

$$u_j(x,t) = x_{2j} \cos t - x_{2j-1} \sin t \ \leq \ 0 \quad \text{for all} \quad j = 1, 2, \ldots, l,$$

  are replaced by

$$g_j(x) \cos t - h_j(x) \sin t \ \leq \ 0 \quad \text{for all} \quad j = 1, 2, \ldots, l,$$

  and the barrier function φ is defined accordingly. The necessary changes for this reformulation are straightforward. Obviously, the number of variables of the standard form (1.1) is $n + 2l$, so that the dimension of the Hessian matrix H grows linearly with l, while the dimension of H for problem (1.3) is independent of l.

4. Numerical Experiments

The algorithm sketched in Section 3.2 has been implemented in MATLAB™. As test problems, we used examples that appeared in the vast literature on multi-fractional programming. This allows us to compare our method with the standard Dinkelbach-type algorithms that have been used to solve these test problems. In particular, we tested our algorithm on numerical examples by Jagannathan [16], Barrodale, Powell, and Roberts [2], Ferland and Potvin [12], and on an example taken from the classical paper [30] by von Neumann. When interpreting the results below, it is helpful to note that—for interior-point methods—the factorizations of the Hessian matrix dominate the overall computational costs (at least for large problems). It is the number of these factorizations that we focus on in our numerical results. For linear and convex problems, the numerical experience is that the number of factorizations of the Hessian grows very slowly with the number of inequality constraints, and it is almost independent of the dimension of the problem. The results of our numerical experiments suggest that the proposed interior-point algorithm for multi-fractional programs exhibits the very same behavior.

4.1. Examples by Jagannathan

In [16], Jagannathan presented two examples of generalized fractional programs, both depending on two variables. The first problem [16, Example 1] involves two fractions with a nonlinear numerator function and linear denominator, and, in addition, four linear constraints. The second problem [16, Example 2] consists of four fractions of linear functions and again four linear constraints. Jagannathan solved these problems with a variant of Dinkelbach's method. Each iteration of his algorithm involves the solution of a convex program, and in case of the second problem, the convex program is a linear program. In Table 1, we list the number of iterations for Jagannathan's algorithm (`Jag`) and for our interior-point method (`IP`), as well as the number of factorizations of the Hessian matrix and the overall number of line-search steps (during Newton's method) that were performed by our interior-point algorithm. Both algorithms were stopped as soon as the convergence check (3.5) (with tolerance $\texttt{tol} = 10^{-4}$) was satisfied.

We would like to stress that the number of factorizations needed by our interior-point method can still be reduced if a more sophisticated implementation is used. In particular, for the first

	#it `Jag`	#it `IP`	#fact	#line
1st problem	5	13	33	50
2nd problem	5	6	14	22

Table 1: Examples by Jagannathan

problem, we have tested the strategy of using the factorization of the Hessian for the computation of three "Newton" steps (one true Newton step and two inexact "Newton" steps with the old Hessian), instead of recomputing the Hessian for each Newton step. This resulted in a run with 15 iterations, 18 factorizations and 74 line-search steps. For larger problems, the work for the line-search steps is negligible, so that the second run corresponds to a reduction of almost 50 percent of the running time of the algorithm.

Of course, interior-point and Dinkelbach-type methods cannot be compared with each other based on these two academic examples alone. However, we remark that the evaluation of 18 or 14 Hessian matrices during our algorithm roughly corresponds to the solution of one or two linear (or convex) programs by an interior-point method, while Dinkelbach-type algorithms typically solve 5 to 10 linear (or convex) subproblems to find the optimal solution of the fractional program.

4.2. Examples by Barrodale, Powell, and Roberts

A classical algorithm for solving discrete rational Chebyshev approximation problems (1.4) is the *differential-correction algorithm* of Cheney and Loeb [7]; see also [2, 6]. Actually, as noted in [14], Dinkelbach's algorithm [11] for multi-fractional programs can be viewed as a special case of the differential-correction algorithm. However, it seems that this close relationship of the two algorithms has largely been overlooked in the literature on multi-fractional programming.

In [2], Barrodale, Powell, and Roberts present numerical results for the ordinary differential-correction (`ODC`) algorithm for solving discrete rational Chebyshev approximation problems of the form (1.5). We have tested our interior-point method on the examples from [2], and we now present some typical results. The test problem is always of the form (1.5), with $l = 21$ equally spaced points $z_j = -1 + (j-1)\cdot 0.1$, $j = 1, 2, \ldots, 21$, and $h(z) = e^z$. This function h is approximated by rational functions p/q of type (N, M), where N and M denote the maximal degree of p and q, respectively. In Table 2, for different values of (N, M), we give the number of iterations needed by the `ODC` algorithm and the statistics for our interior-point method (`IP`). As starting point we chose the zero approximation, $p \equiv 0$, $q \equiv 1$, and the convergence tolerance in (3.5) was $\texttt{tol} = 10^{-6}$. We remark that the solution obtained by our algorithm always corresponded to 7 digits of (absolute) accuracy to the solution given in [2].

4.3. Examples by Ferland and Potvin

Ferland and Potvin [12] present numerical results for 240 random examples of linear multi-fractional programs that were solved by various Dinkelbach-type algorithms. We chose six of these examples, all of the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \max_{j=1,2,\ldots,l} \frac{f_j(x)}{g_j(x)} \qquad \text{subject to} \quad 0 \le x_i \le 10, \quad i = 1, 2, \ldots, n. \tag{4.1}$$

(N/M)	#it `ODC`	#it `IP`	#fact	#line
(1/1)	6	13	35	52
(2/2)	10	12	34	47
(1/3)	8	13	36	54
(4/2)	10	11	29	38

Table 2: Examples by Barrodale, Powell, and Roberts

(n/l)	#it `IP`	#fact	#line
(10/10)	15	40	67
(10/20)	8	20	29
(50/10)	18	64	166
(50/20)	11	34	66
(100/10)	16	60	184
(100/20)	18	66	164

Table 3: Examples by Ferland and Potvin

Here, $f_j(x) = a_j^T x + \alpha_j$ and $g_j(x) = b_j^T x + \beta_j$, with uniformly distributed random numbers $(a_j)_i \in [0,1]$, $\alpha_j \in [0,1]$, $(b_j)_i \in [0,50]$, and $\beta_j \in [0.01, 50]$. In Table 3, we list the number of iterations, factorizations of the Hessian, and line-search steps of our method for $n = 10,\ 50,\ 100$, $l = 10,\ 20$, and convergence tolerance `tol` $= 10^{-4}$. The best algorithm in [12] is NEWMODM, which is also a variant of Dinkelbach's method. The NEWMODM algorithm requires the solution of one linear program at each iteration. Ferland and Potvin report for the NEWMODM algorithm an overall average of 6.92 iterations for the particular set of test problems (4.1).

4.4. Examples by von Neumann

It seems that von Neumann [30][3] was the first to present an application that leads to multi-fractional programs. He also proved a surprisingly simple duality theory for linear multi-fractional programs based on a model of general economic equilibrium. We have used one of the problems from [30] as an test example for our interior-point method. The problem involves $n+m$ variables $y_1, y_2, \ldots, y_n \geq 0$ and $x_1, x_2, \ldots, x_m \geq 0$, and constraints of the type

$$\sum_{j=1}^{n} a_{ij} y_j \ \geq\ \sum_{j=1}^{n} b_{ij} y_j \quad \text{and} \quad \sum_{i=1}^{m} a_{ij} x_i \ \leq\ \sum_{i=1}^{m} b_{ij} x_i. \tag{4.2}$$

We chose uniformly distributed random numbers $a_{ij} \in [0,1]$ and $b_{ij} \in [0,1]$ for the coefficients in (4.2), and we solved this problem for different values of n and m. The feasible domain for these examples is unbounded, and thus not all the assumptions stated in Section 1.5 are satisfied. However, our implementation does not make explicit use of the boundedness of the feasible set, and in fact, it solved these problems well. In Table 4, for various values of (n/m), we list the number

[3] The original German version of this paper appeared already in 1938.

(n/m)	#it IP	#fact	#line
(10/10)	7	25	71
(10/20)	7	27	74
(50/10)	8	31	66
(50/20)	8	31	87
(100/10)	7	24	64
(100/20)	8	26	83

Table 4: Examples by von Neumann

of iterations, factorizations of the Hessian, and line-search steps for our interior-point method. For all runs, we used the stopping criterion (3.5) with tolerance `tol` $= 10^{-4}$.

We now turn to the theoretical investigation of our interior-point method. In particular, we show that the short-step algorithm is globally convergent, and that the rate of convergence is polynomial. This analysis extends our earlier results [15] for the single-fraction case $l = 1$ to the multi-fractional case $l > 1$. In particular, some of the auxiliary results in Sections 5–7 are reproduced from [15] in only slightly modified form. This was done in order to keep the present paper self-contained.

5. Self-Concordant Functions and Further Assumptions

Our theoretical results are based on the notion of self-concordance, which is defined and examined in great detail by Nesterov and Nemirovsky in [22, 23]. First, we briefly summarize some known results about self-concordant functions. Then, we state the additional assumptions on problem (1.1) that are needed to establish polynomial-time convergence of Algorithm 2.1.

5.1. Results About Self-Concordant Functions

Let $\mathcal{S} \subset I\!R^n$ be a given convex set with nonempty interior $\mathcal{S}^\circ$. Let $\phi : \mathcal{S}^\circ \mapsto I\!R$ be a given function that is assumed to be three times continuously differentiable on $\mathcal{S}^\circ$. For each pair of points $x \in \mathcal{S}^\circ$ and $h \in I\!R^n$, we define a function $\psi = \psi_{x,h}$ in an open interval about $\beta = 0$ by setting

$$\psi(\beta) = \psi_{x,h}(\beta) := \phi(x + \beta h) \quad \text{for all} \quad \beta \in I\!R \quad \text{with} \quad x + \beta h \in \mathcal{S}^\circ. \tag{5.1}$$

5.1.1. Self-Concordance

In slight variation of the definition in [23], we call the function ϕ *self-concordant* if ϕ is convex and the functions $\psi = \psi_{x,h}$ satisfy

$$|\psi'''(0)| \le 2\psi''(0)^{3/2} \quad \text{for all} \quad x \in \mathcal{S}^\circ \quad \text{and} \quad h \in I\!R^n. \tag{5.2}$$

This condition is indeed a very natural condition, it just states that the *relative* change of $\nabla^2\phi(x)$ is bounded in a small neighborhood of x (since ψ''', which describes the absolute change of ψ'', is bounded by a suitable power of ψ''). Thus, (5.2) is equivalent to a relative Lipschitz condition of the Hessian $\nabla^2\phi$, see [17, 18].

A self-concordant function ϕ is said to be *strongly self-concordant* if $\lim_{i\to\infty} \phi(x^{(i)}) = \infty$ for every sequence $\{x^{(i)}\}_{i=o}^{\infty} \subset \mathcal{S}^\circ$ that converges to a boundary point of $\mathcal{S}$. In the following, we always assume that the function ϕ is strongly self-concordant. Next, we list a number of properties that follow from the strong self-concordance of ϕ; we refer the reader to [18, 23] for proofs of these results.

5.1.2. Inner Ellipsoid

First, we note that ϕ is strictly convex on $\mathcal{S}^\circ$. Therefore, for all $x \in \mathcal{S}^\circ$, the Hessian $H_x := \nabla^2\phi(x)$ of ϕ is positive definite, and we can associate the canonical norm $\|\cdot\|_{H_x}$ with x. Furthermore, for each point $x \in \mathcal{S}^\circ$, the unit ball with respect to the H_x-norm describes an *inner ellipsoid* [28, 29] for $\mathcal{S}$. More precisely, we have

$$x + h \in \mathcal{S} \quad \text{for all} \quad x \in \mathcal{S}^\circ \quad \text{and} \quad h \in \mathbb{R}^n \quad \text{with} \quad \|h\|_{H_x} \le 1. \tag{5.3}$$

5.1.3. Newton's Method

Self-concordance of ϕ implies that Newton's method without line search converges "well" (in a sense to be specified below) when applied to the unconstrained minimization problem

$$\underset{x\in\mathcal{S}}{\text{minimize}} \;\; \phi(x). \tag{5.4}$$

We now assume that $\mathcal{S}$ is a bounded set in $\mathbb{R}^n$. Since ϕ is strictly convex and strongly self-concordant, it follows that the problem (5.4) has a unique optimal solution $x_\star$ and $x_\star \in \mathcal{S}^\circ$; the point $x_\star$ is called the *analytic center* [28] of $\mathcal{S}$. Let $x \in \mathcal{S}^\circ$, and let

$$\Delta x := -H_x^{-1}\nabla\phi(x) \tag{5.5}$$

denote the Newton step for obtaining the next Newton iterate $\tilde{x} := x + \Delta x$. If $\|\Delta x\|_{H_x} \le 1/4$, then

$$\tilde{x} \in \mathcal{S}^\circ \tag{5.6}$$

and Newton's method is guaranteed to be quadratically convergent. More precisely, with $\Delta\tilde{x}$ denoting the next Newton step starting at $\tilde{x}$, we have

$$\|\Delta\tilde{x}\|_{H_{\tilde{x}}} \;\le\; \frac{\|\Delta x\|_{H_x}^2}{(1-\|\Delta x\|_{H_x})^2}. \tag{5.7}$$

Furthermore, if $\|\Delta x\|_{H_x} < 1/3$, then the distance of x to $x_\star$ is bounded as follows:

$$\|x - x_\star\|_{H_x} \;\le\; 1-(1-3\|\Delta x\|_{H_x})^{1/3}. \tag{5.8}$$

The importance of these results on Newton's method is that they do not depend at all on the data of the problem.

We remark that, by (5.5), the H_x-norm $\|\Delta x\|_{H_x}$ of the Newton step is the same as the H_x^{-1}-norm of the gradient:

$$\|\Delta x\|_{H_x} = \|\nabla\phi(x)\|_{H_x^{-1}}. \tag{5.9}$$

The equivalence (5.9) will frequently be used later on. It also explains the choice of the particular norm in (2.12), (2.13), and (3.6). We note that the restrictions $\|\Delta x\|_{H_x} \le 1/4$, respectively $\le 1/3$, for the above results can be relaxed somewhat; however, if $\epsilon_1 \ge 1$ in (3.6), then none of the statements in this section can be maintained.

5.1.4. θ-Self-Concordance

In the following, we assume that the functions $\psi = \psi_{x,h}$ defined in (5.1), in addition to (5.2), satisfy a second condition, which was also introduced by Nesterov and Nemirovsky [23]. We call the self-concordant function ϕ *θ-self-concordant* if there is a constant $\theta \geq 1$ such that

$$|\psi'(0)| \leq \sqrt{\theta}\sqrt{\psi''(0)} \quad \text{for all} \quad x \in \mathcal{S}^\circ \quad \text{and} \quad h \in I\!R^n.$$

5.1.5. Outer Ellipsoid

For bounded sets $\mathcal{S}$, the θ-self-concordance of the function ϕ implies that we have the following *outer ellipsoid* for $\mathcal{S}$ centered at the minimizer $x_\star$ of (5.4):

$$\mathcal{S} \subseteq x_\star + \left\{ h \in I\!R^n \;\middle|\; \|h\|_{H_{x_\star}} \leq \theta + 2\sqrt{\theta} \right\}. \tag{5.10}$$

For a proof of (5.10), see [19].

5.1.6. Sums of Self-Concordant Functions

Finally, we will make use of the following result from [23, Proposition 3.1 (ii)].

Lemma 5.1. *If ϕ_j, $j = 1, 2$, are θ_j-self-concordant functions on $\mathcal{S}^\circ$, then the function $\phi := \phi_1 + \phi_2$ is $(\theta_1 + \theta_2)$-self-concordant on $\mathcal{S}^\circ$.*

5.2. Further Assumptions

For the remainder of this paper, we now make the following assumptions on the multi-fractional program (1.1), in addition to the conditions stated in Section 1.5. We stress that these additional assumptions are only needed for our convergence analysis. They are not crucial for the actual implementation described in Section 3. In fact, some of the examples that were solved by our program in Section 4 do not satisfy all of these additional assumptions.

Instead of (1.7), we now require that the feasible domain $\mathcal{S}$ of (1.1) fulfills the following stronger condition:

$$x_{2j-1} \geq 0 \quad \text{and} \quad x_{2j} \geq 0 \quad \text{for all} \quad j = 1, 2, \ldots, l \quad \text{and} \quad x \in \mathcal{S}. \tag{5.11}$$

In view of (2.2) and (5.11), we only need to consider $t \in [0, \pi/2]$ in (2.3), and in particular, the optimal value t_{opt} of (2.3) now satisfies $t_{\text{opt}} \geq 0$.

From now on, we assume that the constraint functions f_i, $i = 1, 2, \ldots, s$, are three times continuously differentiable on $\mathcal{S}^\circ$. This implies that the logarithmic barrier function ϕ defined in (1.8) is also three times continuously differentiable on $\mathcal{S}^\circ$. Furthermore, we now assume that ϕ is θ-self-concordant for some constant number $\theta \geq 1$. We refer to θ as the *self-concordance parameter* of ϕ. As shown in [10, 23], the logarithmic barrier function fulfills this condition for important classes of problems. For instance, ϕ is θ-self-concordant with self-concordance parameter $\theta = s$ if all constraint functions f_i in (1.1) are linear or convex quadratic. We remark that any θ-self-concordant logarithmic barrier function (1.8) is always strongly θ-self-concordant since, by construction, $\phi(x)$ tends to infinity as x approaches the boundary of $\mathcal{S}$. Finally, we assume that the parameter m in the definition of the function $\varphi(\cdot, t)$ in (2.7) is an arbitrary constant $m \geq 1$. The results of Section 6 hold true for any such choice of m, while the results in Section 7 will require a specific choice of m. Note that the functions $-m \log(-u_j(\cdot, t))$, $j = 1, 2, \ldots, l$, are all m-self-concordant. Therefore, in view of Lemma 5.1, the function $\varphi(\cdot, t)$ is strongly $(\theta + l\, m)$-self-concordant. Consequently, $\varphi(\cdot, t)$ satisfies all the properties listed in Section 5 (with $\mathcal{S}$ replaced by $\mathcal{S}(t)$ and m replaced by $\theta + l\, m$).

6. Analysis of the Short-Step Algorithm

In this section, we establish some basic properties of Algorithm 2.1. We show that the algorithm is well defined, and that the iterates remain in a prescribed neighborhood of the central points $x(t_k)$. We begin by stating three auxiliary results.

Proposition 6.1. *Let $0 < t \leq \pi/2$ and $x \in \mathcal{S}^\circ(t)$. Let $\mathbf{r}(x,t)$ be given by (2.14), and set*

$$\alpha(x) := \max_{j=1,2,\ldots,l} \ \arctan\left(\frac{x_{2j}}{x_{2j-1}}\right). \tag{6.1}$$

Then:

$$\mathbf{r}(x,t) < t - \alpha(x). \tag{6.2}$$

Proof. First, we remark that, by (2.14) and (2.1), we have

$$r_j(x,t) = \frac{-u_j(x,t)}{\sqrt{x_{2j-1}^2 + x_{2j}^2}} = \frac{\sin t - (x_{2j}/x_{2j-1})\cos t}{\sqrt{1 + (x_{2j}/x_{2j-1})^2}} \quad \text{for all} \quad j = 1,2,\ldots,l. \tag{6.3}$$

Without loss of generality, let $j = 1$ be the index that maximizes $\arctan(x_{2j}/x_{2j-1})$ in the definition (6.1) of $\alpha(x)$. It follows from (6.3) that $j = 1$ is also the index for which the minimum is attained in the definition (2.14) of $\mathbf{r}(x,t)$. Thus, to prove (6.2), it suffices to show that

$$\alpha(x) = \arctan\left(\frac{x_2}{x_1}\right) \ < \ t - \mathbf{r}(x,t) = t - \frac{x_1}{\sqrt{x_1^2 + x_2^2}}\sin t + \frac{x_2}{\sqrt{x_1^2 + x_2^2}}\cos t. \tag{6.4}$$

Setting $\alpha := \alpha(x)$ and using standard trigonometric relations, we see that (6.4) is equivalent to the inequality

$$t - \alpha \ > \ \cos\alpha \sin t - \sin\alpha\cos t = \sin(t-\alpha). \tag{6.5}$$

Since $x \in \mathcal{S}^\circ(t)$ and by (2.4), we have $t - \alpha > 0$. Thus (6.5) follows from the inequality $\sin\beta < \beta$ for $\beta > 0$. □

In the following proposition, we state some relations between the Euclidean norm $\|\cdot\|$, and the norm $\|\cdot\|_{H_x}$. For a proof of this result, see [15, Proposition 3.2].

Proposition 6.2. *Let $x \in \mathcal{S}^\circ$, and set $H_x = \nabla^2\phi(x)$. Then:*

$$\|h_{[j]}\| \ \leq \ \|x_{[j]}\| \quad \textit{for all} \quad h \in I\!R^n \quad \textit{with} \quad \|h\|_{H_x} \ \leq \ 1, \tag{6.6}$$

$$\|h\|_{H_x^{-1}} \ \leq \ \|h\| \cdot \|x_{[j]}\| \quad \textit{for all} \quad h = \begin{bmatrix} 0_{(2j-2)} \\ h_{[j]} \\ 0_{(n-2j)} \end{bmatrix} \in I\!R^n, \quad h_{[j]} \in I\!R^2. \tag{6.7}$$

Finally, we will need the following lemma about rank-l perturbations of positive definite matrices. This lemma is a generalization of the result in [15, Lemma 3.3] about rank-one perturbations.

Lemma 6.3. *Let W, $\Delta W \in I\!R^{n\times l}$ and $\Gamma = \Gamma^T \in I\!R^{l\times l}$ be given, and assume that Γ is nonsingular. Let M and $\bar{M}$ be the matrices defined by*

$$M := I + WW^T \quad \textit{and} \quad \bar{M} = I + \bar{W}\bar{W}^T, \quad \textit{where} \quad \bar{W} := (W + \Delta W)\Gamma^{-1}, \tag{6.8}$$

and I is the $n \times n$ identity matrix. Then:

$$M \ \leq \ c\,\bar{M}, \quad \textit{where} \quad c := \max\{1, \|\Gamma\|^2\} + \|\Delta W\|\left(\|\Gamma\| + \|\Delta W\|\right). \tag{6.9}$$

Proof. First, we remark that $M \leq c\bar{M}$ if, and only if,

$$\frac{y^T M y}{y^T \bar{M} y} \;\leq\; c \quad \text{for all} \quad y \in I\!R^n \quad \text{with} \quad \|y\| = 1. \tag{6.10}$$

Let $y \in I\!R^n$ with $\|y\| = 1$ be arbitrary, and set $\varrho := \|\bar{W}^T y\|$. With (6.8), it follows that

$$\begin{aligned} \frac{y^T M y}{y^T \bar{M} y} &= \frac{1 + \|W^T y\|^2}{1 + \|\bar{W}^T y\|^2} = \frac{1 + \|\Gamma \bar{W}^T y - (\Delta W)^T y\|^2}{1 + \varrho^2} \\ &= \frac{1 + \|\Gamma \bar{W}^T y\|^2}{1 + \varrho^2} + \frac{\|(\Delta W)^T y\|^2 - 2(\bar{W}^T y)^T \Gamma (\Delta W)^T y}{1 + \varrho^2}. \end{aligned} \tag{6.11}$$

Next, we note that, for all $\varrho \in I\!R$, one has

$$\frac{1 + \|\Gamma\|^2 \varrho^2}{1 + \varrho^2} \;\leq\; \max\{1, \|\Gamma\|^2\}, \quad \frac{1}{1 + \varrho^2} \;\leq\; 1, \quad \text{and} \quad \frac{2\varrho}{1 + \varrho^2} \;\leq\; 1. \tag{6.12}$$

Using (6.12) and the estimate

$$|(\bar{W}^T y)^T \Gamma (\Delta W)^T y| \;\leq\; \varrho \,\|\Gamma\| \cdot \|\Delta W\|,$$

we obtain from (6.11) that

$$\frac{y^T M y}{y^T \bar{M} y} \;\leq\; \max\{1, \|\Gamma\|^2\} + \|\Delta W\| \left(\|\Gamma\| + \|\Delta W\|\right). \tag{6.13}$$

In view of (6.9), the upper bound in (6.13) is just c, and thus (6.13) is the desired inequality (6.10). $\Box$

In the following theorem, we establish some basic properties of the kth iterate $x^{(k)}$ of Algorithm 2.1.

Theorem 6.4. *Let $0 < t_k \leq \pi/2$, $x^{(k)} \in \mathcal{S}^\circ(t_k)$, and assume that*

$$\|g_k\|_{B_k} \;\leq\; \frac{1}{16}. \tag{6.14}$$

Let $\sigma_k > 0$, and let t_{k+1} be defined by (2.19) and (2.18) in Algorithm 2.1. Then:

a)

$$\left| \frac{\bar{u}_j}{u_j} - 1 \right| \;\leq\; \frac{\sigma_k}{m\, l} \quad \textit{for all} \quad j = 1, 2, \ldots, l. \tag{6.15}$$

b) *If $0 < \sigma_k \leq m\, l$, then:*

$$t_{\mathrm{opt}} < t_{k+1} < t_k \quad \textit{and} \quad x^{(k)} \in \mathcal{S}^\circ(t_{k+1}). \tag{6.16}$$

c) *If $\sigma_k \leq 1/18$, then:*

$$\|\bar{g}_k\|_{\bar{B}_k} \;\leq\; \frac{1}{5}. \tag{6.17}$$

Proof. For simplicity we omit indices k by setting $x = x^{(k)}$, $t = t_k$, $\Delta t = \Delta t_k$, $\sigma = \sigma_k$, $g = g_k$, $\bar{g} = \bar{g}_k$, and $\bar{B} = \bar{B}_k$. Furthermore, we set

$$v_j := v_j(t), \quad \bar{v}_j = v_j(t - \Delta t), \quad u_j := u_j(x,t) \quad \text{and} \quad \bar{u}_j = u_j(x, t - \Delta t) \tag{6.18}$$

and we denote by $H_x = \nabla^2 \phi(x)$ the Hessian of ϕ at x.

First, we prove the inequality (6.15) in part a). Let $j \in \{1, 2, \ldots, l\}$ be arbitrary. Remark that, by (2.9), we have $0 < H_x \le \bar{H}_k$, and thus

$$\bar{B} = \bar{H}_k^{-1} \;\le\; H_x^{-1}. \tag{6.19}$$

Using (2.1), (6.18), and standard trigonometric identities, it follows that

$$v_j - \bar{v}_j = 2\sin(\Delta t/2)\dot{v}_j, \quad \text{where} \quad \dot{v}_j := \begin{bmatrix} 0_{(2j-2)} \\ -\cos(t - \Delta t/2) \\ -\sin(t - \Delta t/2) \\ 0_{(n-2j)} \end{bmatrix} \in I\!R^n. \tag{6.20}$$

Note that $\|\dot{v}_j\| = 1$, and by (6.7) and (6.19), we have

$$\|\dot{v}_j\|_{\bar{B}} \;\le\; \|\dot{v}_j\|_{H_x^{-1}} \;\le\; \|x_{[j]}\|. \tag{6.21}$$

By multiplying (6.20) from the left by x^T and by using the estimate

$$2\sin(\Delta t/2) \;\le\; \Delta t, \tag{6.22}$$

we obtain

$$|u_j - \bar{u}_j| = 2\sin(\Delta t/2)|x^T \dot{v}_j| \;\le\; \Delta t \|x_{[j]}\|. \tag{6.23}$$

Combining (6.23) and (2.18) gives

$$|u_j - \bar{u}_j| \;=\; \frac{\sigma}{m\,l}\|x_{[j]}\| \min_{p=1,2,\ldots,l} \frac{|u_p|}{\|x_{[p]}\|} \;\le\; \frac{\sigma\,|u_j|}{m\,l}. \tag{6.24}$$

Note that, by assumption, $x \in \mathcal{S}^\circ(t)$, and thus, by (2.4), we have $|u_j| = -u_j > 0$. In particular, $|u_j| \ne 0$, and dividing (6.24) by $|u_j|$ gives the desired estimate (6.15). This concludes the proof of part a).

Part b) is a direct consequence of (2.6) and Proposition 6.1, since $t_{\text{opt}} < \alpha(x)$ for strictly feasible points x.

Finally, we turn to part c), and we assume that $0 < \sigma \le 1/18$. We will prove the following two inequalities:

$$\|g\|_{\bar{B}} \;\le\; \frac{1}{14}, \tag{6.25}$$

$$\|\bar{g} - g\|_{\bar{B}} \;\le\; \frac{9}{70} = \frac{1}{5} - \frac{1}{14}. \tag{6.26}$$

By the triangle inequality, (6.25) and (6.26) imply the desired estimate (6.17).

To show (6.25), we will derive the following more general result:

$$\bar{B} \;\le\; \frac{64}{49}\, B. \tag{6.27}$$

Clearly, from (6.27) and (6.14) it follows that

$$\|g\|_{\bar{B}} \;\leq\; \sqrt{\frac{64}{49}}\,\|g\|_B \;\leq\; \frac{8}{7}\cdot\frac{1}{16} = \frac{1}{14},$$

which is just (6.25). Since $\bar{B} > 0$, the claim (6.27) is equivalent to

$$B^{-1} \;\leq\; \frac{64}{49}\,\bar{B}^{-1}. \tag{6.28}$$

We note in passing that property (6.28) can be viewed as a local Lipschitz continuity of the Hessian with respect to t, while property (6.26) implies a local Lipschitz continuity of the gradient—measured in the canonical norm. These two properties essentially guarantee that Algorithm 2.1 is well defined. Recall that the matrix H_x is symmetric positive definite, and hence it has a symmetric positive definite square root $H_x^{-1/2}$. By multiplying (6.28) from the left and the right by $H_x^{-1/2}$, we obtain the relation

$$M \;\leq\; \frac{64}{49}\,\bar{M}, \quad \text{where} \quad M := H_x^{-1/2}B^{-1}H_x^{-1/2}, \quad \bar{M} := H_x^{-1/2}\bar{B}^{-1}H_x^{-1/2}. \tag{6.29}$$

Obviously, (6.29) and (6.28) are equivalent; we will prove (6.29). Using (2.9), (2.16), (2.17), and (6.18), one easily verifies that the matrices M and $\bar{M}$ in (6.29) can be represented as follows:

$$\begin{aligned} M &= I + \sum_{j=1}^{l} w_j w_j^T, \quad \text{where} \quad w_j := \frac{\sqrt{m}}{u_j} H_x^{-1/2} v_j, \\ \bar{M} &= I + \sum_{j=1}^{l} \bar{w}_j \bar{w}_j^T, \quad \text{where} \quad \bar{w}_j := \frac{\sqrt{m}}{\bar{u}_j} H_x^{-1/2} \bar{v}_j. \end{aligned} \tag{6.30}$$

By (6.20), the vectors w_j and $\bar{w}_j$ satisfy

$$\bar{w}_j = \frac{1}{\gamma_j}(w_j + \Delta w_j), \quad \text{where} \quad \gamma_j := \frac{\bar{u}_j}{u_j}, \quad \Delta w_j := -\frac{\sqrt{m}}{u_j} 2\sin(\Delta t/2) H_x^{-1/2}\dot{v}_j.$$

Clearly,

$$\gamma_j \;\leq\; 1, \tag{6.31}$$

and using (6.22), (2.18), (6.21), and (2.14), we obtain that

$$\|\Delta w_j\| \;\leq\; \frac{\sqrt{m}}{|u_j|}\,\Delta t\,\|\dot{v}_j\|_{H_x^{-1}} \;\leq\; \sqrt{m}\,\frac{\sigma\,\mathbf{r}}{l\,m}\,\frac{\|x_{[j]}\|}{|u_j|} \;=\; \frac{\sigma\,\mathbf{r}}{r_j\,l\,\sqrt{m}} \;\leq\; \frac{\sigma}{l\,\sqrt{m}} \;\leq\; \frac{1}{18\,l}. \tag{6.32}$$

We now apply Lemma 6.3 to the matrices M and $\bar{M}$ in (6.30). From (6.32) and (6.31), we have

$$\|\Delta W\| = \|\,[\,\Delta w_1 \;\;\cdots\;\; \Delta w_l\,]\,\| \;\leq\; \frac{1}{18} \quad \text{and} \quad \|\Gamma\| = \|\,\mathrm{diag}(\gamma_1,\ldots,\gamma_l)\| \;\leq\; 1,$$

and, by (6.9), it follows that

$$M \;\leq\; c\,\bar{M}, \quad c \;\leq\; 1 + \frac{18+1}{18^2} \;<\; \frac{64}{49},$$

which is just the claim (6.29).

It remains to prove the estimate (6.26). Note that, by (2.8) and (2.15), we have

$$\bar{g} - g = m \sum_{j=1}^{l} \left(\frac{v_j}{u_j} - \frac{\bar{v}_j}{\bar{u}_j} \right) = \sum_{j=1}^{l} \left(1 - \frac{u_j}{\bar{u}_j} \right) \frac{m}{u_j} v_j + \frac{m}{\bar{u}_j} (v_j - \bar{v}_j),$$

and thus

$$\|\bar{g} - g\|_{\bar{B}} \;\leq\; \sum_{j=1}^{l} \left| 1 - \frac{u_j}{\bar{u}_j} \right| \frac{m}{|u_j|} \|v_j\|_{\bar{B}} + \sum_{j=1}^{l} \frac{m}{|\bar{u}_j|} \|v_j - \bar{v}_j\|_{\bar{B}}. \tag{6.33}$$

From definition (2.9) of H and by the rank-one-update formula for inverse matrices, it follows that

$$\bar{B} = (M + zz^T)^{-1} = M^{-1} - \frac{M^{-1} z z^T M^{-1}}{1 + z^T M^{-1} z}, \quad \text{where} \quad z = \frac{\sqrt{m}\, v_j}{u_j}, \quad M = H_x + m \sum_{\substack{\nu=1 \\ \nu \neq j}}^{l} \frac{v_\nu v_\nu^T}{u_\nu^2}.$$

This implies

$$\sqrt{m}\, \frac{\|v_j\|_{\bar{B}}}{|u_j|} = \|z\|_{\bar{B}} = \frac{\|z\|_{M^{-1}}}{\sqrt{1 + \|z\|_{M^{-1}}^2}} \;\leq\; 1 \quad \text{for all} \quad j = 1, 2, \ldots, l,$$

and thus

$$\sum_{j=1}^{l} \frac{m}{|u_j|} \|v_j\|_{\bar{B}} \;\leq\; \sqrt{m}\, l \;\leq\; m\, l.$$

Moreover, from (6.15), we obtain that

$$\left| 1 - \frac{u_j}{\bar{u}_j} \right| \;\leq\; \frac{\sigma/(m\, l)}{1 - \sigma/(m\, l)},$$

so that

$$\sum_{j=1}^{l} \left| 1 - \frac{u_j}{\bar{u}_j} \right| \frac{m}{|u_j|} \|v_j\|_{\bar{B}} \;\leq\; \frac{\sigma}{1 - \sigma/(m\, l)}. \tag{6.34}$$

From (6.20)–(6.22) and (2.18), we get the estimate

$$\sum_{j=1}^{l} \frac{m}{|u_j|} \|v_j - \bar{v}_j\|_{\bar{B}} \;\leq\; m\, \Delta t \sum_{j=1}^{l} \frac{\|x_{[j]}\|}{|u_j|} = \sum_{j=1}^{l} \frac{\sigma\, \mathbf{r}}{l\, r_j} \;\leq\; \sigma. \tag{6.35}$$

Finally, by combining (6.33)–(6.35), and by using $m\, l \geq 1$ and $\sigma \leq 1/18$, we obtain

$$\|\bar{g} - g\|_{\bar{B}} \;\leq\; \sigma + \frac{\sigma}{1 - \sigma/(m\, l)} \;<\; \frac{9}{70},$$

which is the desired estimate (6.26). This concludes the proof of Theorem 6.4. ∎

We now show that, if $\sigma_k \leq 1/18$, then $t_{\text{opt}} < t_k$ and $x^{(k)} \in \mathcal{S}^{\circ}(t_k)$ for all k, and consequently, all quantities involved in Algorithm 2.1 are always well defined.

Theorem 6.5. *Let t_k and $x^{(k)}$ be the iterates generated by Algorithm* 2.1, *and assume that* $0 < \sigma_k \leq 1/18$ *for all k. Then, for all $k = 0, 1, \ldots,$ the following properties are satisfied:*

a)

$$t_{\text{opt}} < t_k \ \le \ \pi/2 \quad and \quad x^{(k)} \in \mathcal{S}^\circ(t_k), \tag{6.36}$$

$$\|g(x^{(k)}, t_k)\|_{(H(x^{(k)},t_k))^{-1}} \ \le \ \frac{1}{16}. \tag{6.37}$$

b)

$$\|x^{(k)} - x(t_k)\|_{H(x^{(k)},t_k)} \ \le \ \frac{1}{14}, \tag{6.38}$$

$$\|x^{(k)} - x(t_k)\|_{H(x(t_k),t_k)} \ \le \ \frac{1}{13}. \tag{6.39}$$

Proof. We use induction on k to show part a). For $k = 0$, both properties (6.36) and (6.37) are satisfied. Indeed, the initial point $x^{(0)}$ of Algorithm 2.1 is chosen such that (6.37) holds. Furthermore, we have $0 < t_0 \le \pi/2$, and thus, by part b) of Theorem 6.4 (with $k = 0$), $t_{\text{opt}} < t_0$. Now assume that (6.36) and (6.37) hold for some $k \ge 0$. We need to show that

$$t_{\text{opt}} < t_{k+1} \ \le \ \pi/2, \tag{6.40}$$

$$x^{(k+1)} \in \mathcal{S}^\circ(t_{k+1}), \tag{6.41}$$

$$\|g_{k+1}\|_{B_{k+1}} \ \le \ \frac{1}{16}. \tag{6.42}$$

Note that, by (6.36)–(6.37), the assumptions of Theorem 6.4 are fulfilled, and from (6.16), we immediately get (6.40). We now set $x := x^{(k)}$ and $\tilde{x} := x^{(k+1)}$. By the second part of (6.16), we have $x \in \mathcal{S}^\circ(t_{k+1})$. Recall that $\tilde{x} = x + \Delta x$ is obtained by performing one step—starting at x—of Newton's method (without line search) for minimizing $\varphi(\cdot, t_{k+1})$ over $\mathcal{S}(t_{k+1})$. Moreover, by (2.20) and since $\bar{B}_k = \bar{H}_k^{-1}$, we have

$$\Delta x = -\bar{B}_k \, \bar{g}_k = -\bar{H}_k^{-1} \, \bar{g}_k.$$

Together with part c) of Theorem 6.4, it follows that

$$\|\Delta x\|_{\bar{H}_k} = \|\bar{g}_k\|_{\bar{B}_k} \ \le \ \frac{1}{5} \ < \ \frac{1}{4}. \tag{6.43}$$

Thus, we can apply the results (5.6) and (5.7) on Newton's method in Section 5.1.3 (with $\mathcal{S}$ replaced by $\mathcal{S}(t_{k+1})$ and $\phi(\cdot)$ replaced by $\varphi(\cdot, t_{k+1})$). From (5.6), we get $\tilde{x} \in \mathcal{S}^\circ(t_{k+1})$, which is just (6.41). Using (5.7) and (6.43), we obtain

$$\|\Delta\tilde{x}\|_{H_{\tilde{x}}} \ \le \ \frac{\|\Delta x\|^2_{\bar{H}_k}}{(1 - \|\Delta x\|_{\bar{H}_k})^2} \ \le \ \frac{1/25}{(1 - 1/5)^2} = \frac{1}{16}. \tag{6.44}$$

Note that, in view of (5.9) and since $\tilde{x} = x^{(k+1)}$, we have

$$\|\Delta\tilde{x}\|_{H_{\tilde{x}}} = \|g_{k+1}\|_{B_{k+1}},$$

and thus (6.44) is equivalent to (6.42). This completes the proof of part a).

We now turn to part b). Let $k \ge 0$ be fixed, and set $x = x^{(k)}$, $t = t_k$, and $H = H(x^{(k)}, t_k)$. To prove (6.38), we make use of the property (5.8) of Newton's method in Section 5.1.3 (with $\mathcal{S}$

replaced by $\mathcal{S}(t)$ and $\phi(\cdot)$ replaced by $\varphi(\cdot, t)$). By (5.5) and since $B_k = H^{-1}$, the Newton step at x is given by

$$\Delta x = -H^{-1}\nabla_x \varphi(x,t) = -B_k\, g_k.$$

Note that Algorithm 2.1 does not compute this step but first updates $t_{k+1} = t_k - \Delta t_k$ and then computes the Newton step $-\bar{B}_k \bar{g}_k$ for the updated function $\varphi(\,\cdot\,, t_{k+1})$. Here, we only analyze the "virtual step" Δx to obtain a bound on $x - x(t)$. By (6.37), we have $\|\Delta x\|_H \leq 1/16 < 1/3$, and by applying (5.8) (with $x_\star = x(t)$), we get

$$\|x - x(t)\|_H \;\leq\; 1 - \left(1 - \frac{3}{16}\right)^{1/3} \;<\; \frac{1}{14}.$$

This is just the desired estimate (6.38).

Finally, the inequality (6.39) follows immediately from (6.38) by means of a general result on the norms induced by strongly self-concordant functions. Indeed, by applying [18, Lemma 2], we obtain

$$\|x - x(t)\|_{H(x(t),t)} \;\leq\; \frac{\|x - x(t)\|_H}{1 - \|x - x(t)\|_H} \;\leq\; \frac{1/14}{1 - 1/14} = \frac{1}{13}.$$

This completes the proof of Theorem 6.5. $\square$

7. Convergence Analysis

In this section, we present our main theoretical result, namely the polynomial-time convergence of Algorithm 2.1. To keep the exposition simple, we assume that $\theta \geq l$. This assumption can be made without loss of generality, by setting $\theta := l$ if $\theta < l$; note that l is given, but θ is not uniquely determined. However, we stress that this assumption will result in a weaker complexity estimate if the original θ actually satisfied $\theta < l$. From now on, we set $m = \theta/l$ in the definition (2.7) of the barrier function $\varphi(\cdot, t)$. Since $\theta \geq l$, we have $m \geq 1$, and thus all assumptions and results of Section 6 hold true for our choice of m. Note that, since $\theta + m\,l = 2\theta$, the function $\varphi(\cdot, t)$ is now 2θ-self-concordant; this property will be used frequently in the following.

We continue to use the notation introduced in Section 2. In addition, for each fixed $j \in \{1, 2, \ldots, l\}$, we consider a relaxed version of (2.3), namely the problem

$$\operatorname*{minimize}_{x \in I\!R^n} \; \arctan\left(\frac{x_{2j}}{x_{2j-1}}\right) \qquad \text{subject to} \quad x \in \mathcal{S}. \tag{7.1}$$

We denote by t^j_{opt} the optimal value of (7.1). Clearly, we have $t^j_{\text{opt}} \leq t_{\text{opt}}$. For $t^j_{\text{opt}} \leq t \leq t_0$, we denote by

$$\mathcal{Q}_j(t) := \{\, z \in I\!R^2 \mid z = x_{[j]} \text{ for some } x \in \mathcal{S}(t)\,\} \tag{7.2}$$

the set in $I\!R^2$, which consists of all truncations $x_{[j]}$ of the vectors $x \in \mathcal{S}(t)$. We denote by $\mathcal{Q}^\circ_j(t)$ the interior of $\mathcal{Q}_j(t)$. By construction, it follows that

$$\{\, x_{[j]} \in I\!R^2 \mid x \text{ is an optimal solution of (1.1)}\,\} \;\subset\; \mathcal{Q}_j(t)$$

for $t \geq t_{\text{opt}}$, i.e., $\mathcal{Q}_j(t)$ contains the truncations $x_{[j]}$ of all optimal solutions of problem (1.1). For all $k = 0, 1, \ldots$, we define sets

$$\mathcal{T}_{j,k} := \{\, z \in I\!R^2 \mid 0 \geq z^T v_{[j]}(t_k) \geq -d_{j,k}\,\}, \quad \text{where} \quad d_{j,k} := \frac{13\,l}{6}\,\left|u_j(x^{(k)}, t_k)\right|. \tag{7.3}$$

Here, in view of (2.1), the vector $v_{[j]}(t)$ is given by

$$v_{[j]}(t) = \begin{bmatrix} -\sin t \\ \cos t \end{bmatrix}. \tag{7.4}$$

We remark that $\mathcal{T}_{j,k}$ is a stripe in $I\!R^2$ of width $d_{j,k}$. Finally, we set

$$\mathcal{D}_j := \left\{ z \in I\!R^2 \;\middle|\; z_1,\, z_2 \geq 0 \text{ and } \|z\| \leq \left(1+\sqrt{2\theta}\right)^2 \|x_{[j]}(t_0)\| \right\}, \tag{7.5}$$

where $x(t_0)$ is the analytic center of $\mathcal{S}(t_0)$. Note that $\mathcal{D}_j$ is the quarter of a disk about the origin.

Both sets $\mathcal{Q}_j(t_k)$ and $\mathcal{T}_{j,k}$ contain the truncation $x^{(k)}_{[j]}$ of the kth iterate of Algorithm 2.1, as well as the truncation of any optimal solution x^{opt} of the convex fractional program (7.1). In the next theorem, we summarize these results.

Theorem 7.1. *Let t_k and $x^{(k)}$ be the iterates generated by Algorithm* 2.1*, and assume that* $0 < \sigma_k \leq 1/18$ *for all k. Then, for all $k = 0, 1, \ldots,$ and all $j = 1, 2, \ldots, l$, the following properties are satisfied:*

$$x^{(k)}_{[j]} \in \mathcal{Q}^\circ_j(t_k), \tag{7.6}$$

$$x^{\text{opt}}_{[j]} \in \mathcal{Q}_j(t_{k+1}) \subseteq \mathcal{Q}_j(t_k) \subseteq \mathcal{D}_j \quad \text{for all optimal solutions } x^{\text{opt}} \text{ of } (7.1), \tag{7.7}$$

$$\mathcal{Q}_j(t_k) \subset \mathcal{T}_{j,k}. \tag{7.8}$$

For the proof of Theorem 7.1, we will need yet another property of the m-self-concordant logarithmic barrier function ϕ. For a proof of this result see [15, Lemma 4.2].

Lemma 7.2. *Let $\bar{\mathcal{S}} \in I\!R^n$ be a bounded convex set and $c : \bar{\mathcal{S}} \mapsto I\!R$ be a continuously differentiable convex function such that the set*

$$\bar{\mathcal{S}}_c := \{\, x \in \bar{\mathcal{S}} \mid c(x) \leq 0 \,\}$$

has a nonempty interior $\bar{\mathcal{S}}^\circ_c$. Let $\bar{\phi}$ be an $\bar{\theta}$-self-concordant barrier function associated with $\bar{\mathcal{S}}$, let $\bar{\kappa} \geq 1$ be a constant, and let $\hat{x} \in \bar{\mathcal{S}}_c$ be the unique optimal solution of

$$\underset{x \in \bar{\mathcal{S}}_c}{\text{minimize}} \;\; \left(\bar{\phi}(x) - \bar{\kappa}\log(-c(x))\right). \tag{7.9}$$

Then:

$$\left(1 + \frac{\bar{\theta}}{\bar{\kappa}}\right) c(\hat{x}) \;\leq\; c(x) \;\leq\; 0 \quad \text{for all} \quad x \in \bar{\mathcal{S}}_c. \tag{7.10}$$

Proof of Theorem 7.1. From the second relation in (6.36), we immediately get (7.6). We now turn to (7.7). First, we show that

$$\mathcal{Q}_j(t_0) \subseteq \mathcal{D}_j. \tag{7.11}$$

Let $z \in \mathcal{Q}_j(t_0)$ be arbitrary, but fixed. Since $x_{2j-1},\; x_{2j} \geq 0$ for all $x \in \mathcal{S}$, we have $z_1, z_2 \geq 0$. It remains to show the second condition in (7.5). We set $x := x(t_0)$ and $H_x := \nabla^2\phi(x)$. Note that, in view of (2.9), we have $H_x \le H(x,t_0)$. Using (5.10) (applied to the (2θ)-self-concordant function $\varphi(\cdot,t_0)$), we conclude that

$$z = x_{[j]} + \left(2\theta + 2\sqrt{2\theta}\right) h_{[j]} \quad \text{for some} \quad h \in I\!R^n \quad \text{with} \quad \|h\|_{H_x} \le 1. \tag{7.12}$$

From (7.12) and (6.6), it follows that

$$\|z\| \le \left(1 + (2\theta + 2\sqrt{2\theta})\right) \|x_{[j]}\| = \left(1 + \sqrt{2\theta}\right)^2 \|x_{[j]}\|,$$

and thus $z \in \mathcal{D}_j$.

Next, we note that $v_{[j]}(t)$ given by (7.4) clearly satisfies

$$z^T v_{[j]}(t') \ge z^T v_{[j]}(t) \quad \text{for all} \quad 0 \le t' < t \le \frac{\pi}{2} \quad \text{and} \quad z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \quad z_1, z_2 \ge 0.$$

Consequently, we have $\mathcal{Q}_j(t') \subseteq \mathcal{Q}_j(t)$ for all $0 \le t' < t \le \pi/2$. Since t_k is monotonously decreasing, this—along with (7.11)—implies (7.7).

We now turn to (7.8). By the definitions of $\mathcal{Q}_j(t)$ and $\mathcal{T}_{j,k}$ in (7.2) and (7.3), and since $|u_j(x^{(k)},t_k)| = -(x^{(k)})^T v_j(t_k)$, the inclusion (7.8) is equivalent to the following estimates:

$$0 \ge x^T v_j(t_k) \ge \frac{13\,l}{6} (x^{(k)})^T v_j(t_k) \quad \text{for all} \quad x \in \mathcal{S}(t_k). \tag{7.13}$$

In order to prove (7.13), we apply Lemma 7.2 with $c(x) := x^T v_j(t_k)$, $\bar{\mathcal{S}}_c := \mathcal{S}(t_k)$, $\bar{\phi}(x) := \phi(x) - \sum_{\nu=1,\ \nu\ne j}^{l} \log(-v_\nu^T x)$, $\bar{\theta} := m(2l-1)$, and $\bar{\kappa} := m$. By (2.10) and (2.7), the unique optimal solution $\hat{x}$ of (7.9) is just the analytic center $x(t_k)$ of $\mathcal{S}(t_k)$, and from (7.10), we obtain that

$$0 \ge x^T v_j(t_k) \ge 2\,l\,(x(t_k))^T v_j(t_k) \quad \text{for all} \quad x \in \mathcal{S}(t_k). \tag{7.14}$$

It only remains to show that

$$(x(t_k))^T v_j(t_k) \ge \frac{13}{12} (x^{(k)})^T v_j(t_k), \tag{7.15}$$

since (7.14) and (7.15) imply the desired estimate (7.13). To this end, note that, by inequality (6.39) from Theorem 6.5, the vector

$$h := 13\left(x^{(k)} - x(t_k)\right) \tag{7.16}$$

satisfies $\|h\|_{H(x(t_k),t_k)} \le 1$. Thus, by the inner ellipsoidal approximation in (5.3) (with $\mathcal{S}$ replaced by $\mathcal{S}(t_k)$), we have

$$x(t_k) + h \in \mathcal{S}(t_k).$$

Together with (7.16), it follows that

$$0 \ge (x(t_k) + h)^T v_j(t_k) = 13\,(x^{(k)})^T v_j(t_k) - 12\,(x(t_k))^T v_j(t_k).$$

This inequality is equivalent to (7.15). □

After these preparations, we are now in a position to prove our convergence result for Algorithm 2.1. We set

$$\mathcal{C}_{j,k} := \mathcal{D}_j \cap \mathcal{T}_{j,0} \cap \mathcal{T}_{j,1} \cap \cdots \cap \mathcal{T}_{j,k},$$

where $\mathcal{D}_j$ and $\mathcal{T}_{j,\nu}$, $\nu = 0,1,\ldots,k$, are the sets defined in (7.5) and (7.3). Furthermore, for compact sets $\mathcal{C} \subset I\!R^2$, we denote by $\mu(\mathcal{C}) := \text{area}(\mathcal{C})$ the area of $\mathcal{C}$. Our convergence result can then be formulated as follows.

Theorem 7.3. *Let $x^{(k)}$ be the kth iterate generated by Algorithm 2.1 with $\sigma_k = 1/18$ for all k. Let m be the self-concordance parameter of the logarithmic barrier function (1.8) associated with the feasible set $\mathcal{S}$ of the multi-fractional program (1.1). Then, for all $k = 0, 1, \ldots,$ the following properties are satisfied:*

$$x_{[j]}^{(k)} \in \mathcal{C}_{j,k}^{\circ} \quad \textit{for all} \quad j = 1, 2, \ldots, l, \tag{7.17}$$

$$x_{[j]} \in \mathcal{C}_{j,k} \quad \textit{for all} \quad j = 1, 2, \ldots, l \quad \textit{and all optimal solutions } x \textit{ of } (1.1), \tag{7.18}$$

$$\prod_{j=1}^{l} \mu(\mathcal{C}_{j,k+1}) \ \leq \ \left(1 - \frac{1}{80\,\theta\, l}\right) \prod_{j=1}^{l} \mu(\mathcal{C}_{j,k}), \tag{7.19}$$

$$\prod_{j=1}^{l} \mu(\mathcal{C}_{j,k}) \ \to \ 0 \quad \textit{if, and only if,} \quad t_k \ \to \ t_{\mathrm{opt}}. \tag{7.20}$$

Proof. The properties (7.17) and (7.18) follow immediately from the relations (7.6)–(7.8) in Theorem 7.1.

We now turn to (7.19) and show, that for at least one of the sets $\mathcal{C}_{j,k}$ the area is reduced by a factor $c \leq 1 - 1/(80\,\theta\, l)$. For this part of the proof, we assume without loss of generality, that at iteration step k the quantity $\mathbf{r}$ is determined by the index $j = 1$, i.e., $r_1 = \min_j r(x,t)$. Thus, we only need to consider the case $j = 1$, and for brevity, we will sometimes omit the index 1. We examine the area of $\mathcal{C}_{1,k}$ and $\mathcal{C}_{1,k+1}$. Note that, in view of (7.3), each stripe $\mathcal{T}_{1,\nu}$ is bounded by two parallel lines with slope $\tan t_\nu$. Moreover, the left line passes through the origin 0 in $\mathbb{R}^2$. Recall that $0 < t_{\nu+1} < t_\nu \leq \pi/2$ for all ν. It follows that the upper left boundary of the set $\mathcal{C}_{1,k}$ consists of a segment of the form $[0, p_k]$ of the left borderline of $\mathcal{T}_{1,k}$. Here, p_k is given by

$$p_k := \rho_k \begin{bmatrix} \cos t_k \\ \sin t_k \end{bmatrix}, \quad \text{where} \quad \rho_k := \max \left\{ \rho \in \mathbb{R} \;\middle|\; \rho \begin{bmatrix} \cos t_k \\ \sin t_k \end{bmatrix} \in \mathcal{C}_{1,k} \right\}.$$

Let p_{k+1} be the corresponding boundary point of $\mathcal{C}_{1,k+1}$, and let $\mathcal{F}_k$ denote the triangle with vertices 0, p_k, and p_{k+1}. Note that

$$\mu(\mathcal{F}_k) = \frac{1}{2}\rho_k \rho_{k+1} \sin \Delta t_k, \tag{7.21}$$

where, by (2.19), (2.18) and the choice of $j = 1$ (and $\sigma_k = 1/18$),

$$t_k - t_{k+1} = \Delta t_k = \frac{|u_1(x^{(k)}, t_k)|}{18\ \theta\, \|x_{[1]}^{(k)}\|}. \tag{7.22}$$

By construction of the triangle $\mathcal{F}_k$, we have

$$\mathcal{F}_k \subset \mathcal{C}_{1,k} \quad \text{and} \quad \mathcal{C}_{1,k+1} \subset \mathcal{C}_{1,k} \setminus \mathcal{F}_k^{\circ},$$

and thus

$$\mu(\mathcal{C}_{1,k+1}) \ \leq \ \mu(\mathcal{C}_{1,k}) - \mu(\mathcal{F}_k) = \left(1 - \frac{\mu(\mathcal{F}_k)}{\mu(\mathcal{C}_{1,k})}\right) \mu(\mathcal{C}_{1,k}). \tag{7.23}$$

Next, recall the definition of $d_k := d_{1,k}$ in (7.3), and set

$$q_k := d_k \begin{bmatrix} \sin t_k \\ -\cos t_k \end{bmatrix}, \quad \text{where} \quad d_k = \frac{13\, l}{6}\, |u_1(x^{(k)}, t_k)|. \tag{7.24}$$

Observe that, by construction of $\mathcal{C}_{1,k}$ as the intersection of the disk $\mathcal{D}_1$ and the stripes $\mathcal{T}_{1,\nu}$, $\nu = 0, 1, \ldots, k$, the angle of the set $\mathcal{C}_{1,k}$ at the vertex p_k is at most $\pi/2$, and hence

$$\mathcal{C}_{1,k} \subset \left\{ z \in I\!\!R^2 \;\middle|\; z^T \begin{bmatrix} \cos t_k \\ \sin t_k \end{bmatrix} \leq \rho_k \right\}. \tag{7.25}$$

Furthermore, the radius ρ_k of the vertex p_k is nonincreasing, i.e., $\rho_\nu \geq \rho_{\nu+1}$ for all $\nu = 0, 1, \ldots$, implying that

$$\mathcal{C}_{1,k} \subset \left\{ z \in I\!\!R^2 \;\middle|\; \|z\| \leq \rho_k \right\}. \tag{7.26}$$

In view of (7.25) and since $\mathcal{C}_{1,k} \subset \mathcal{T}_{1,k}$, the set $\mathcal{C}_{1,k}$ is contained in the rectangle $\mathcal{R}_k$ whose vertices are the points 0, p_k, $p_k + q_k$, and q_k. Moreover, one easily verifies that p_{k+1} is a boundary point of $\mathcal{C}_{1,k}$, and that

$$\mathcal{C}_{1,k} \subset \mathcal{R}_k \setminus (\Delta R_k)^\circ \quad \text{and} \quad \Delta R_k \subset \mathcal{R}_k. \tag{7.27}$$

Here, ΔR_k is the rectangle with vertices p_{k+1}, $p_k + s_k q_k$, $p_k + q_k$, and $c_k p_k + q_k$, and

$$s_k = \rho_{k+1} \sin \Delta t_k \quad \text{and} \quad c_k = \rho_{k+1} \cos \Delta t_k. \tag{7.28}$$

From (7.27) and (7.28), it follows that

$$\begin{aligned} \mu(\mathcal{C}_{1,k}) \;&\leq\; \mu(\mathcal{R}_k) - \mu(\Delta R_k) \\ &=\; \rho_k d_k - (\rho_k - \rho_{k+1} \cos \Delta t_k)(d_k - \rho_{k+1} \sin \Delta t_k) \\ &\leq\; \rho_k \rho_{k+1} \sin \Delta t_k + d_k \rho_{k+1} \cos \Delta t_k. \end{aligned} \tag{7.29}$$

Note that, by (7.17) and (7.26), we have $\|x_{[1]}^{(k)}\| \leq \rho_k$. Together with (7.22) and (7.24), it follows that

$$\tan \Delta t_k \;\geq\; \Delta t_k = \frac{6\, d_k}{13\, l \cdot 18\, \theta\, \|x_{[1]}^{(k)}\|} \;\geq\; \frac{d_k}{39\, \theta\, l\, \rho_k}. \tag{7.30}$$

From (7.29), (7.21), and (7.30), we obtain

$$\frac{\mu(\mathcal{C}_{1,k})}{\mu(\mathcal{F}_k)} \;\leq\; 2 \left(1 + \frac{d_k}{\rho_k \tan \Delta t_k} \right) \;\leq\; 2 + 78\, \theta\, l \;\leq\; 80\, \theta\, l.$$

In combination with (7.23), this implies the desired estimate (7.19).

Finally, we turn to (7.20). Note that, by construction, $0 \in \mathcal{C}_{j,k}$ for all j and all k. Furthermore, the truncations $x_{[j]}^{\text{opt}}$ of all optimal solutions of problem (1.1) are contained in $\mathcal{C}_{j,k}$. If one of the areas of the sets $\mathcal{C}_{j,k}$ tends to zero, then, in view of the convexity of $\mathcal{C}_{j,k}$, this implies that $\mathcal{C}_{j,k}$ converges to a line segment containing the origin and the truncations $x_{[j]}^{\text{opt}}$ of all optimal solutions of (1.1). Thus, by construction of the sets $\mathcal{C}_{j,k}$, it follows $t_k \to \arctan(x_{2j}^{\text{opt}} / x_{2j-1}^{\text{opt}}) \leq t_{\text{opt}}$. Together with $t_k \geq t_{\text{opt}}$, this implies (7.20). □

Note that $\mathcal{C}_{j,0} \subseteq \mathcal{D}_j$, and thus, by (7.5), we have

$$\mu(\mathcal{C}_{j,0}) \;\leq\; \rho_{j,0}^2 \, \frac{\pi}{4}, \quad \text{where} \quad \rho_{j,0} = \left(1 + \sqrt{2\theta} \right)^2 \|x_{[j]}(t_0)\|.$$

Together with (7.19), it follows that

$$\prod_{j=1}^{l} \mu(\mathcal{C}_{j,k}) \;\leq\; \left(1 - \frac{1}{80\, \theta\, l} \right)^k \prod_{j=1}^{l} \left(\rho_{j,0}^2 \, \frac{\pi}{4} \right) \quad \text{for all} \quad k = 0, 1, \ldots .$$

Therefore, the product of the areas $\mu(\mathcal{C}_{j,k})$, which measure the convergence of Algorithm 2.1, converge to zero at a polynomial rate.

8. Concluding Remarks

We presented an interior-point method for minimizing the maximum of l linear fractions subject to convex constraints. We discussed a practical implementation of the proposed method, and we presented results of various numerical experiments. These results suggest that the proposed method is a viable alternative to the standard techniques for solving multi-fractional programs using Dinkelbach's algorithm or variants thereof. We also showed that, under some additional assumptions, a short-step version of our interior-point method converges in polynomial time. Here, convergence is measured in terms of suitably constructed two-dimensional sets.

After we had completed our analysis, we were informed by Professor Nemirovsky[4] that he also found a convergence proof for an interior-point method for solving multi-fractional programs. However, a written version of his results was not available yet at the time we completed the present paper. The class of problems covered by Nemirovsky's analysis is slightly more general than the family of multi-fractional programs studied here. Nevertheless, there are definite differences between his and our analysis. For instance, convergence is measured differently, and while the theoretical complexity of his algorithm seems to be better than our estimates, it also appears that his convergence proof requires somewhat stronger assumptions.

Acknowledgment

We would like to thank Professor Nemirovsky for stimulating "discussions" via electronic mail.

References

[1] Alizadeh, F., *Optimization Over the Positive Semi-Definite Cone: Interior-Point Methods and Combinatorial Applications*, Advances in Optimization and Parallel Computing, Edited by P. Pardalos, North-Holland, Amsterdam, pp. 1–25, 1992.

[2] Barrodale, I., Powell, M. J. D., and Roberts, F. D. K., *The Differential Correction Algorithm for Rational l_∞-Approximation*, SIAM Journal on Numerical Analysis, Vol. 9, pp. 493–504, 1972.

[3] Bector, C. R., and Cambini, A., *Fractional Programming—Some Recent Results*, Generalized Convexity and Fractional Programming with Economic Applications (Pisa, 1988), Lecture Notes in Economics and Mathematical Systems, Vol. 345, Springer-Verlag, Berlin, pp. 86–98, 1989.

[4] Benadada, Y., Crouzeix, J. P., and Ferland, J. A., *An Interval-Type Algorithm for Generalized Fractional Programming*, Generalized Convexity and Fractional Programming with Economic Applications (Pisa, 1988), Lecture Notes in Economics and Mathematical Systems, Vol. 345, Springer-Verlag, Berlin, pp. 106–120, 1989.

[5] Boyd, S., El Ghaoui, L., *Method of Centers for Minimizing Generalized Eigenvalues*, Preprint, Electrical Engineering Department, Stanford University, Stanford, California, 1992.

[6] Braess, D., *Nonlinear Approximation Theory*, Springer-Verlag, Berlin, 1986.

[7] Cheney, E. W., and Loeb, H. L., *Two New Algorithms for Rational Approximation*, Numerische Mathematik, Vol. 3, pp. 72–75, 1961.

[8] Crouzeix, J. P., and Ferland, J. A., *Algorithms for Generalized Fractional Programming*, Mathematical Programming, Vol. 52, pp. 191–207, 1991.

[9] Crouzeix, J. P., Ferland, J. A., and Schaible, S., *An Algorithm for Generalized Fractional Programs*, Journal of Optimization Theory and Applications, Vol. 47, pp. 35–49, 1985.

[10] den Hertog, D., Jarre, F., Roos, C., and Terlaky, T., *A Unifying Investigation of Interior-Point Methods for Convex Programming*, Report No. 92–89, Faculty of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands, 1992.

[11] Dinkelbach, W., *On Nonlinear Fractional Programming*, Management Science, Vol. 13, pp. 492–498, 1967.

[4] Private communication, May 1993

[12] Ferland, J. A., and Potvin, J. Y., *Generalized Fractional Programming: Algorithms and Numerical Experimentation*, European Journal of Operational Research, Vol. 20, pp. 92–101, 1985.

[13] Fiacco A. V., and McCormick, G. P., *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, New York, 1968.

[14] Flachs, J., *Generalized Cheney-Loeb-Dinkelbach-Type Algorithms*, Mathematics of Operations Research, Vol. 10, pp. 674–687, 1985.

[15] Freund, R. W., and Jarre, F., *An Interior-Point Method for Convex Fractional Programming*, AT&T Numerical Analysis Manuscript No. 93–03, Bell Laboratories, Murray Hill, New Jersey, 1993.

[16] Jagannathan. R., *An Algorithm for a Class of Nonconvex Programming Problems with Nonlinear Fractional Objectives*, Management Science, Vol. 31, pp. 847–851, 1985.

[17] Jarre, F., *On the Method of Analytic Centers for Solving Smooth Convex Programs*, Optimization (Varetz, 1988), Lecture Notes in Mathematics, Vol. 1405, Springer-Verlag, Berlin, pp. 69–85, 1989.

[18] Jarre, F., *Interior-Point Methods for Convex Programming*, Applied Mathematics and Optimization, Vol. 26, pp. 287–311, 1992.

[19] Jarre, F., *Optimal Ellipsoidal Approximations Around the Analytic Center*, Applied Mathematics and Optimization, to appear.

[20] Karmarkar, N., *A New Polynomial-Time Algorithm for Linear Programming*, Combinatorica, Vol. 4, pp. 373–395, 1984.

[21] Mangasarian, O. L., *Nonlinear Programming,* McGraw-Hill, New York, New York, 1969.

[22] Nesterov, J. E., and Nemirovsky, A. S., *A General Approach to Polynomial-Time Algorithms Design for Convex Programming*, Report, Central Economical and Mathematical Institute, USSR Acad. Sci., Moscow, Russia, 1988.

[23] Nesterov, J. E., and Nemirovsky, A. S., *Self-Concordant Functions and Polynomial-Time Methods in Convex Programming*, Report, Central Economical and Mathematical Institute, USSR Acad. Sci., Moscow, Russia, 1989.

[24] Nesterov, J. E., and Nemirovsky, A. S., *An Interior Point Method for Generalized Linear-Fractional Programming*, Report, Central Economical and Mathematical Institute, USSR Acad. Sci., Moscow, Russia, 1991.

[25] Schaible, S., *Bibliography in Fractional Programming*, Zeitschrift für Operations Research, Vol. 26, pp. 211–241, 1982.

[26] Schaible, S., *Fractional Programming*, Zeitschrift für Operations Research, Vol. 27, pp. 39–54, 1983.

[27] Schaible, S., *Multi-Ratio Fractional Programming—A Survey*, Optimization, Parallel Processing and Applications, Lecture Notes in Economics and Mathematical Systems, Vol. 304, Springer-Verlag, Berlin, pp. 57–66, 1988.

[28] Sonnevend, G., *An 'Analytical Centre' for Polyhedrons and New Classes of Global Algorithms for Linear (Smooth, Convex) Programming*, System Modelling and Optimization (Budapest, 1985), Lecture Notes in Control and Information Sciences, Vol. 84, Springer-Verlag, Berlin, pp. 866–875, 1986.

[29] Sonnevend, G., and Stoer, J., *Global Ellipsoidal Approximations and Homotopy Methods for Solving Convex Analytic Programs*, Applied Mathematics and Optimization, Vol. 21, pp. 139–165, 1990.

[30] von Neumann, J., *A Model of General Economic Equilibrium*, Review of Economic Studies 13, pp. 1–9, 1945.