An Efficient Interior-Point Method for Convex Multicriteria Optimization Problems

Jörg Fliege

February 25, 2004

Abstract

In multicriteria optimization, several objective functions, conflicting with each other, have to be minimized simultaneously. We propose a new efficient method for approximating the solution set of a multiobjective programming problem, where the objective functions involved are arbitary convex functions and the set of feasible points is convex. The method is based on generating warm-start points for an efficient interior-point algorithm, while the approximation computed consists of a finite set of discrete points. Complexity results for the method proposed are derived. It turns out that the number of operations per point *decreases* when the number of points generated for the approximation *increases*.

1 Introduction

In multicriteria optimization, several conflicting objective functions have to be minimized simultaneously. Applications for this problem type are abound, see, e. g., [14, 17, 9, 21] and the references contained therein. In general, no single point will minimize all of the several objective functions at once. For this reason, a feasible point is defined as optimal if there does not exist a different feasible point with the same or smaller objective function values such that there is a strict decrease in at least one objective function value. As a consequence, different optimal points of a multicriteria problem will be incomparable to each other in terms of the order defined above. Due to this, no single point can represent the whole solution set, which will usually be quite large. Therefore, it is important to gain as much information as possible about the solution set of a given problem, preferably by constructing a well-defined approximation to it. This is the subject of this paper.

The necessity for a strategy like the one outlined above was also observed by various other authors, see, e. g., Benson and Sayın [1], Das and Dennis [6, 5], Hillermeier [17],

Fliege and Heseler [9], or by Fliege [8] and the references contained therein. However, only in [9] the method under consideration is subject to a worst-case analysis with respect to its computational complexity. Berkelaar et al. [2, 3] propose a basis identification algorithm based on interior-point methodology for convex quadratic bicriteria optimization problems. These results where further generalized by Hadigheh et al. [13]. However, since these algorithms compute combinatorial information of all optimal bases of the solutions, it is not clear if they are computationally or theoretically efficient.

2 Convex Multiobjective Optimization

2.1 The Problem

Let there be given $n > 1$ convex objective functions

$$f_i : G \longrightarrow \mathbb{R}, \qquad i = 1, \ldots, n, \tag{1}$$

as well as a feasible set $G \subseteq \mathbb{R}^m$. We are interested in minimizing simultaneously the functions f_i on the set G in a sense to be specified as follows. We have to define the notion of a solution to the given multicriteria problem. We do this by defining minimal elements of an arbitrary set in the image space.

Definition 2.1 *Let $M \subseteq \mathbb{R}^n$ be a nonempty set. The element $y^* \in M$ is called* efficient *(or* Pareto-optimal*), if and only if there is no other $y \in M$ with*

$$y_i \leq y_i^* \quad \forall\, i \in \{1, 2, \ldots, n\}$$

and

$$y_k < y_k^* \quad \textit{for at least one } k \in \{1, 2, \ldots, n\}.$$

The set of all efficient points of the set M is called the efficient set $E(M)$.

Define the function $f : G \longrightarrow \mathbb{R}^n$ by $f = (f_1, \ldots, f_n)^\top$. With the definition above, it becomes clear that in multicriteria optimization we are in search for the whole set $E(f(G))$ and, obviously, for the corresponding set of optimal decision variables $f^{-1}(E(f(G)))$. Now note that two efficient points $f(x^{(1)}), f(x^{(2)}) \in E(f(G))$ ($x^{(1)}, x^{(2)} \in G$) with $f(x^{(1)}) \neq f(x^{(2)})$ are incomparable to each other. By their very nature of being efficient, there exist two indices $i, j \in \{1, \ldots, n\}$ such that $f_i(x^{(1)}) < f_i(x^{(2)})$ and $f_j(x^{(2)}) < f_j(x^{(1)})$. Therefore, just one efficient point can hardly capture the possible optimal alternatives we face when solving a multicriteria optimization problem. This shows that human decision makers need information about the whole set $E(f(G))$.

In what follows, we assume that G is compact, convex, and $G \neq \emptyset$.

2.2 Scalarization

It is well-known that we can find a point close to $E(f(G))$ of the multicriteria problem specified above by solving the single-objective optimization problem

$$\begin{aligned} \min \quad & z^T f(x) \\ \text{subject to} \quad & x \in G, \end{aligned} \tag{2}$$

with $z \in Z$ an arbitrary weight vector from the set

$$Z := \left\{ z \in \mathbb{R}^n \;\middle|\; \sum_{i=1}^{n} z_i = 1,\ z_i > 0 \quad \forall\, i \in \{1, 2, \ldots n\} \right\}. \tag{3}$$

This approach is often called *scalarization*. (For a discussion of this and other scalarization techniques see e. g. [12, 20, 17, 8].) Indeed, defining the set of *properly efficient points* P by

$$P(f(G)) := \left\{ f(x^*) \;\middle|\; z \in Z,\ x^* \in G,\ f(x^*) = \min_{x \in G} \langle\, z, f(x) \,\rangle \right\}, \tag{4}$$

it can be shown [10, 26] that

$$P(f(G)) \subseteq E(f(G)) \subseteq \mathrm{cl}(P(f(G))) \tag{5}$$

holds. Here, $\mathrm{cl}(\cdot)$ is the closure operator. In fact, this result holds for arbitrary functions $f : G \longrightarrow \mathbb{R}^n$ as long as the set $f(G) + \mathbb{R}^n_+$ is closed and convex. Since we can not distinguish numerically between a set and its closure, we can therefore replace E by P in all applications involving convex functions. However, with the scalarization above, the optimal value function

$$\varphi : z \longmapsto \arg\min\{\langle\, z, f(x) \,\rangle \mid x \in G\} \tag{6}$$

is set-valued and in general not continuous in any sense. It has recently been shown that this effect does not occur when one does not use a linear objective function as above (linear scalarization) but a quadratic one (quadratic scalarization) in the sense of

$$Q(f(G)) := \left\{ f(x^*) \;\middle|\; z \in Z,\ x^* \in G,\ f(x^*) = \min_{x \in G} \langle\, \mathrm{diag}(z) f(x), f(x) \,\rangle \right\},$$

see [8]. As usual, $\mathrm{diag}(z)$ is the diagonal matrix in $\mathbb{R}^{n \times n}$ with the vector $z \in \mathbb{R}^n$ on its main diagonal. Indeed, if $f(G) \subset \mathrm{int}(\mathbb{R}^n_+)$ (a very weak assumption), we have that

$$P(f(G)) \subseteq Q(f(G)) \subseteq E(f(G))$$

holds and the optimal value function

$$\psi : z \longmapsto \arg\min\{\langle\, \mathrm{diag}(z) f(x), f(x) \,\rangle \mid x \in G\} \tag{7}$$

is locally Lipshitz-continuous, see [8]. For more details about different scalarization techniques, the reader is referred to [8, 17, 4, 7].

Introducing artificial variables, we see that we have to consider an infinite family of scalar problems of the form

$$\begin{array}{ll} \text{minimize} & \langle\, z, t \,\rangle \\ \text{subject to} & f_i(x) \leq t_i \qquad (i = 1, \ldots, n), \\ & x \in G \end{array} \tag{8}$$

resp.

$$\begin{array}{ll} \text{minimize} & \langle\, \mathrm{diag}(z)t, t \,\rangle \\ \text{subject to} & f_i(x) \leq t_i \qquad (i = 1, \ldots, n), \\ & x \in G. \end{array} \tag{9}$$

Accordingly, we can assume that the objective function $f : G \longrightarrow \mathbb{R}$ of the scalarized problem is linear or strictly convex quadratic and the set of feasible points is fixed, i. e. independent of the parameterization. This is an important simplification in order to employ interior-point methods, since quadratic objective functions are 0-compatible with self-concordant barrier functions [22] and constant sets of feasible points reduce the amount of work needed to find strictly feasible starting points.

Of course, in applications we have to content ourselves with a finite subset $\tilde{Z} \subset Z$ from which we have to choose our weight vectors. This will lead to a discrete approximation of the set of efficient points. However, it has been observed [9] that an a priori discretization of the set of weights Z will usually result in a waste of a large amount of computational effort, since it is not a priori clear in which part of the weight space a finer discretization (i. e. a finer grid) is necessary. Basically, we want to use more parameters in those regions of the parameter space where weight vectors which are close together result in efficient points whose images in the image space $\mathbb{R}^n$ are far apart from each other. Furthermore, to save work when computing some additional efficient points (i. e. when solving some additional, new optimization problems), we propose to use a warm-start strategy. With such a strategy, points from the iteration history of scalar problems already solved are used as starting points for the optimization problems currently under consideration.

The rest of this paper is as follows: in Section 3, the standard primal path-following interior-point algorithm as presented in [22] is rehearsed in short. It turns out that the first stage of the algorithm (the construction of a point close to the ”center” of the set of feasible points) can be used without any modification in a multicriteria framework. In Section 4, an algorithm is presented in which two scalar programs with linear objective functions are solved simultaneously by way of an adapted interior-point technique. Moreover, an efficiency estimate will elucidate the dependence of the number of steps of the algorithm on the perturbation size. The corresponding algorithmic technique will be generalized to scalar

programs with quadratic objective functions in Section 5 and to arbitrary smooth convex scalar programs in Section 6. In Section 7, bicriteria problems are considered, and a general algorithm based on a refinement strategy in the parameter space is proposed. It turns out that the theoretical complexity of this algorithm is lower than the naïve approach of using a prespecified number of parameter values to achieve a certain density of approximation in the image space $\mathbb{R}^n$ of the multicriteria optimization problem considered. Finally, the results from Section 7 are extended to the case of more than two criteria in Section 8. There, a general algorithm for approximating the set of efficient points of a smooth convex multicriteria optimization problem is discussed.

3 The Standard Primal Path-Following Interior-Point Algorithm

We recall in short the standard primal path-following algorithm presented by Nesterov and Nemirovskii [22, Chapter 3]. This is done not for the sake of completeness, but because several crucial details of this algorithm will be analyzed and adapted further on. Moreover, the notation used in the rest of this paper is introduced, together with some slight generalizations. For most of the proofs of the results presented here, the reader is referred to [22].

3.1 The First Stage

Let $G \subseteq \mathbb{R}^m$ be a compact convex set with nonempty interior and $F : \text{int}(G) \longrightarrow \mathbb{R}$ a self-concordant barrier for G with self-concordancy parameter $\vartheta \geq 1$. Moreover, let $f : G \longrightarrow$ be a β-compatible objective function. In a scalarized multicriteria setting, the discussion of the last section has shown that it makes sense to assume that f is linear or at most quadratic. We therefore assume $\beta = 0$.

It is known that for compact sets G the Hessian $\nabla^2 F(x)$ at an arbitrary point $x \in \text{int}(G)$ is positive definite. This allows us to define for $x \in \text{int}(G)$ the *Newton decrement of F at x* by

$$\lambda(F, x) := \Big(\Big\langle \nabla F(x)), (\nabla^2 F(x))^{-1} \nabla F(x) \Big\rangle\Big)^{1/2},$$

cmp. [22, p. 15–16]. Define the auxiliary function $\omega :]0, 1/3[\longrightarrow \mathbb{R}$ by

$$\omega(\lambda) := 1 - (1 - 3\lambda)^{1/3}.$$

In the standard path-following scheme, there is needed a 5-tuple

$$(\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3') \in \mathbb{R}^5$$

basically controlling the neighbourhood of the central path (to be defined below) used, such that the following system of inequalities holds:

$$0 < \left(\frac{\lambda_i}{1-\lambda_i}\right)^2 \leq \lambda_i' < \lambda_i < 2-\sqrt{3} \qquad (i = 1, 3), \tag{10}$$

$$\lambda_1' < \lambda_2 < \lambda_3, \tag{11}$$

$$\frac{\omega(\lambda_3')}{1-\omega(\lambda_3')} < 1, \tag{12}$$

$$\frac{\omega(\lambda_2)}{1-\omega(\lambda_2)} < \frac{1}{3}, \tag{13}$$

$$\frac{\omega^2(\lambda_1')(1+\omega(\lambda_1'))}{1-\omega(\lambda_1')} \leq \frac{1}{9}, \tag{14}$$

see [22, p. 70]. These inequalities will be put to heavy use in Section 4, and it is therefore important to take a closer look at them. Indeed, (12) is equivalent to $\lambda_3' \leq 7/24$, while (13) is equivalent to $\lambda_2 < 37/192$. Moreover, some simple calculations show that (14) is equivalent to

$$\lambda_1' \leq \frac{1}{3} - \frac{1}{3}\left(\frac{4}{3} - \frac{1}{9}\sqrt[3]{27+3\sqrt{57}} - \frac{2}{3}\frac{1}{\sqrt[3]{27+3\sqrt{57}}}\right)^3 =: \varrho.$$

Due to $\varrho > 37/192$, (11) tells us that we can drop (14). As a consequence, the system (10)–(14) can be simplified to

$$0 < \lambda_i' < \lambda_i < 2-\sqrt{3} \qquad (i = 1, 3), \tag{15}$$

$$\left(\frac{\lambda_1}{1-\lambda_1}\right)^2 \leq \lambda_1', \tag{16}$$

$$\left(\frac{\lambda_3}{1-\lambda_3}\right)^2 \leq \lambda_3' < \frac{7}{24}, \tag{17}$$

$$\lambda_1' < \lambda_2 < \lambda_3, \tag{18}$$

$$\lambda_2 < \frac{37}{192}. \tag{19}$$

The first stage of the interior-point method starts at an arbitrary point $y^{(0)} \in \operatorname{int}(G)$ and calculates an approximation $y^{(j)}$ to the unique minimizer of F such that the Newton decrement at $y^{(j)}$ is bounded by λ_2, i. e.

$$\lambda(F, y^{(j)}) \leq \lambda_2. \tag{20}$$

In these iterations, only standard Newton steps for the function F are performed. The number of iterations j needed to achieve this is bounded from above by

$$1 + \frac{\lambda_1 + \sqrt{\vartheta}}{\lambda_1 - \lambda_1'}\left(\ln\left(\frac{\vartheta}{a(y^{(0)}, G)}\right) + \ln\left(\frac{2}{\lambda_2 - \lambda_1'}\right)\right),$$

see [22, Proposition 3.2.3], and only the parameters λ_1, λ_1', and λ_2 are used within the algorithm. In the run-time estimate above, $a(y^{(0)}, G)$ is the asymmetry coefficient of $y^{(0)}$ with respect to G. For lower bounds on this number, the reader is referred to [4]. Obviously, for large ϑ the leading term $\sqrt{\vartheta}\ln\vartheta$ in the estimate above has the coefficient $1/(\lambda_1 - \lambda_1')$, so it makes sense to maximize $\lambda_1 - \lambda_1'$. Within the bounds given above, the maximum is taken at

$$\lambda_1 = -1/3\,\sqrt[3]{27 + 3\,\sqrt{105}} + 2\,\frac{1}{\sqrt[3]{27 + 3\,\sqrt{105}}} + 1 = 0.2290\ldots$$

and $\lambda_1' = \lambda_1^2/(1-\lambda_1)^2 = 0.0882\ldots$. The leading coefficient then becomes $1/(\lambda_1 - \lambda_1') = 7.103\ldots$.

In a multicriteria setting, the computation of an approximation of the analytic center of G can be executed without any modification at all. If we do not have any further information on the parameters of the scalarization, no better starting point for minimizing the scalarized problems is at hand.

3.2 The Second Stage

In the second stage, we start at the just obtained $x^{(0)} := y^{(j)}$ and follow the central path to a minimum of f by computing discrete approximations $x^{(i)} \in G$ ($i \in \mathbb{N}$) to this central path. More precisely, we consider the function

$$F_t(x) := tf(x) + F(x)$$

as well as the *central path*

$$\{x(t) \in G \mid \lambda(F_t, x(t)) = 0;\, t > 0\} = \{x(t) \in G \mid x(t) \text{ minimum of } F_t;\, t > 0\}$$

and compute parameter values $t(i) > 0$ with $\lim_{i\to\infty} t(i) = \infty$ as well as

$$x^{(i+1)} := x^{(i)} - \left(\nabla_x^2 F_{t(i+1)}(x^{(i)})\right)^{-1} \nabla_x F_{t(i+1)}(x^{(i)}), \quad i = 0, 1, 2, \ldots. \tag{21}$$

The parameter t plays the role of the barrier parameter and is increased in each step. The generated sequence $(x^{(i)})_{i\in\mathbb{N}}$ is a discrete approximation to the central path of the problem considered.

The sequence of barrier parameters can be defined by

$$t(i) := t(1)\exp\left((i-1)\frac{\lambda_3 - \lambda_3'}{\lambda_3 + \sqrt{\vartheta}}\right), \qquad i = 1, 2, \ldots. \tag{22}$$

We will take a look at the derivation of this rule in Lemma 3.4. Of course, an initialization $t(1)$ of the sequence is needed. A proper way for initializing this starting value is discussed now.

For a given $x \in \text{int}(G)$, define the norm $\|\cdot\|_{x,F}$ by

$$\|y\|_{x,F} := \left(\left\langle y, \nabla^2 F(x) y \right\rangle\right)^{1/2}.$$

(Recall that $\nabla^2 F(x)$ is positive definite.) With this norm, the *Dikin ellipsoid of F centered at $x \in \text{int}(G)$ with radius r* is defined by

$$W(x,r) := \{y \in \mathbb{R}^m \mid \|y - x\|_{x,F} \leq r\}.$$

One of the most useful relations with respect to this ellipsoid is

$$W(x,1) \subseteq \text{cl}(G) \tag{23}$$

for all $x \in \text{int}(G)$, see [22, Proposition 2.3.2]. In fact, it can be shown [24] that self-concordancy of the function F is equivalent to this inclusion and just one further inequality which we will use to proof Lemma 3.2 below. As usual, the dual norm to $\|\cdot\|_{x,F}$ will be denoted by $\|\cdot\|^{\circ}_{x,F}$, i. e.

$$\|p\|^{\circ}_{x,F} := \sup\{\langle p, y \rangle \mid \|y\|_{x,F} \leq 1\}.$$

The following lemmas contain some simple generalizations of known results. Since these results are crucial in the following derivations, the most important proofs are given for the sake of completeness.

Lemma 3.1 *Let there be given $x \in \text{int}(G)$, $\mu > \lambda(F,x)$, and $t \geq 0$ with*

$$t \leq \frac{\mu - \lambda(F,x)}{\|\nabla f(x)\|^{\circ}_{x,F}}.$$

Then

$$\lambda(F_t, x) \leq \mu.$$

Proof: This is a slight generalization of what is shown in the proof of [22, Proposition 3.2.4]. □

As a consequence, with $\mu = \lambda_3 > \lambda_2 \geq \lambda(F, x^{(0)})$, a sufficiently small $t(1)$ can always be used to start with the path-following method.

The denominator of the estimate for t in Lemma 3.1 can be estimated itself. This is the purpose of the next two lemmas.

Lemma 3.2 *Let there be given $x \in \text{int}(G)$ with $\lambda(F,x) \leq \delta < 1/3$ and $r := \omega(\delta)/(1 - \omega(\delta)) < 1/3$. Denote the minimum of F by y. Then*

$$W(x,\tau) \subseteq W(y, r + \tau/(1-r))$$

for all $\tau \geq 0$.

Proof: According to [22, Theorem 2.2.2], we have

$$\|x-y\|_{y,F} \leq \frac{\omega(\lambda(F,x))}{1-\omega(\lambda(F,x))} \leq \frac{\omega(\delta)}{1-\omega(\delta)}$$

which means $x \in W(y,r)$. Due to Theorem 2.1.1 from [22], it follows that

$$\frac{1-2\omega(\delta)}{1-\omega(\delta)}\|h\|_{y,F} = (1-r)\|h\|_{y,F} \leq (1-\|x-y\|_{y,F})\|h\|_{y,F} \leq \|h\|_{x,F}$$

for all $h \in \mathbb{R}^m$. Now let $z \in \mathbb{R}^m$. Then

$$\|z-y\|_{y,F} \leq \|z-x\|_{y,F} + \|x-y\|_{y,F} \leq \|z-x\|_{y,F} + r \leq \frac{1}{1-r}\|z-x\|_{x,F} + r,$$

from which the result follows. □

Lemma 3.3 *Let there be given $x \in \mathrm{int}(G)$ with $\lambda(F,x) \leq \delta < 1/3$ and $r := \omega(\delta)/(1-\omega(\delta)) < 1/3$. Denote the minimum of F by y and let $p \in \mathbb{R}^m$ be a subgradient of a convex function $f : G \longrightarrow \mathbb{R}$ at x. Then*

$$\|p\|_{x,F}^{\circ} \leq \frac{1}{\tau}\left(\sup_{z \in W(y,r+\tau/(1-r))} f(z) - f(x)\right) \tag{24}$$

for all $\tau \in]\,0,(1-r)^2[$.

Proof: Due to the subgradient inequality and Lemma 3.2, we have

$$\begin{aligned}
\|p\|_{x,F}^{\circ} &= \frac{1}{\tau}\sup_{z \in W(x,\tau)} \langle\, p, z-x \,\rangle \\
&\leq \frac{1}{\tau}\sup_{z \in W(x,\tau)} (f(z)-f(x)) \\
&\leq \frac{1}{\tau}\left(\sup_{z \in W(y,r+\tau/(1-r))} f(z)-f(x)\right).
\end{aligned}$$

□

One can easily get rid of $f(x)$ in (24) by using

$$f(x) \geq \inf_{z \in W(y,r)} f(z), \tag{25}$$

see again the proof of Lemma 3.2.

Numerous simple bounds for $\|p\|_{x,F}^{\circ}$ can be derived from this lemma. For example, using $\tau := (1-r)^2/2$ and $r < 1/3$ leads to

$$\begin{aligned}
\|p\|_{x,F}^{\circ} &\leq \frac{2}{(1-r)^2}\left(\sup_{z \in W(y,(1+r)/2)} f(z)-f(x)\right) \\
&< \frac{9}{2}\left(\sup_{z \in W(y,2/3)} f(z)-f(x)\right),
\end{aligned} \tag{26}$$

which will be used in the next section. Likewise, $\tau := (1-r)(1/2-r)$ leads to

$$\begin{aligned} \|p\|_{x,F}^{\circ} &\leq \frac{1}{(1-r)(1/2-r)} \left(\sup_{z \in W(y,1/2)} f(z) - f(x) \right) \\ &< 9 \left(\sup_{z \in W(y,1/2)} f(z) - f(x) \right), \end{aligned}$$

which, together with (25), leads to the bound for $\|\nabla f(x)\|_{x,F}^{\circ}$ used in [22].

We now turn our attention to the efficiency of Newton's method applied to the function F_t.

Lemma 3.4 *Let there be given a 5-tuple $(\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3') \in \mathbb{R}^5$ feasible for the system (15)–(19), let $x \in \text{int}(G)$ be given such that*

$$\lambda(F_t, x) \leq \lambda_3 \tag{27}$$

holds, and define y by

$$y := x - \left(\nabla_x^2 F_t(x) \right)^{-1} \nabla_x F_t(x).$$

Then $y \in \text{int}(G)$ and

$$\lambda(F_t, y) \leq \lambda_3'. \tag{28}$$

Proof: Due to (27), $\lambda_3 < 2 - \sqrt{3}$, and [22, Theorem 2.2.2], we have $y \in \text{int}(G)$ as well as

$$\lambda(F_t, y) \leq \left(\frac{\lambda(F_t, x)}{1 - \lambda(F_t, x)} \right)^2.$$

Using (27) again, it follows that $\lambda(F_t, y) \leq \lambda_3^2/(1-\lambda_3)^2$. The result follows with (17). □

As a consequence of this lemma, defining the sequence $(x^{(i)})_{i \in \mathbb{N}}$ by the Newton-process (21) and the update rule (22) will lead to $\lambda(F_{t(i)}, x^{(i-1)}) \leq \lambda_3$ as well as $\lambda(F_{t(i)}, x^{(i)}) \leq \lambda_3'$ for all $i \geq 1$, provided that the starting point generated during the first stage is used. More precisely, we need $\lambda(F_{t(1)}, x^{(0)}) < \lambda_3$ and a starting value $t(1)$ for the barrier parameter bounded by the right-hand side of the estimate in Lemma 3.1 with $\mu = \lambda_3$.

It can be shown [22] that the number of iterations i needed to achieve an absolute accuracy of

$$f(x^{(i)}) - \min_{x \in G} f(x) \leq \varepsilon, \tag{29}$$

$0 < \varepsilon < 1/\|\nabla f(x^{(0)})\|_{x^{(0)},F}^{\circ}$, is bounded above by

$$\begin{aligned} 1 + \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \quad & \left(\ln \left(2\vartheta + \frac{\zeta(\lambda_3')}{2} \right) + \ln(1/\varepsilon) + \ln(\|\nabla f(x^{(0)})\|_{x^{(0)},F}^{\circ}) \right. \\ & \left. + \ln \left(\frac{9}{\lambda_3 - \lambda_2} \right) \right) \end{aligned} \tag{30}$$

where the auxiliary function ζ is defined with the help of the function ω by

$$\zeta(\lambda) := (\omega(\lambda))^2 \frac{1+\omega(\lambda)}{1-\omega(\lambda)}.$$

(Note that (15) implies $\lambda_3 \neq \lambda_3'$.) We will call an $x \in G$ satisfying (29) an ε-*solution* to the optimization problem at hand. The term $\|\nabla f(x^{(0)})\|^{\circ}_{x^{(0)},F}$ in this estimate can be bounded using one of the various estimates given after Lemma 3.3. All of these upper bounds have the form $\sup_{z\in W} f(z) - f(x^{(0)})$ with a suitable set $W \subseteq G$ containing the minimum of F. Note also that $f(x^{(0)})$ can be replaced by using (25).

For large ϑ the leading term $\sqrt{\vartheta}\ln\vartheta$ has the coefficient $1/(\lambda_3 - \lambda_3')$, and it makes sense to choose $\lambda_3 = \lambda_1$ and $\lambda_3' = \lambda_1'$, with λ_1, λ_1' as above. It makes then sense to choose $\lambda_2 = (\lambda_1 + \lambda_1')/2 = 0.1586\ldots$.

4 Solving Two Single-Criteria Programs with Linear Objective Functions: a Warm-Start Approach

Let $c, \hat{c} \in \mathbb{R}^m$ be given with $c \neq 0 \neq \hat{c}$ and define $\delta := \|c - \hat{c}\|_2 > 0$. Again, let G be a compact convex set with nonempty interior. Moreover, suppose that $\|x\|_2 \leq R$ holds for a number $R > 0$ and all $x \in G$. (This number will be used only in a subsequent theoretical analysis, but not in any algorithm to be presented below.) We are now interested in solving the two linear optimization problems

$$\begin{array}{ll} \text{minimize} & \langle\, c, x \,\rangle \\ \text{subject to} & x \in G \end{array} \tag{31}$$

and

$$\begin{array}{ll} \text{minimize} & \langle\, \hat{c}, x \,\rangle \\ \text{subject to} & x \in G. \end{array} \tag{32}$$

Note that we do *not yet* want to consider the bicriteria program consisting of the two objective functions $\langle\, c, \cdot \,\rangle$ and $\langle\, \hat{c}, \cdot \,\rangle$. This is the subject of Section 7. Right now, we consider two different single-criteria problems and we want to solve both of them. Again, note that the discussion in Subsection 2.2 has shown that we have to solve many single-criteria linear programs, not just two of them. We discuss here the case of exactly two for the sake of simplicity.

Since in general we can not solve an optimization problem exactly, we have to be content with an ε-solution to each of our problems in the sense of (29), where $\varepsilon > 0$ is our

prespecified accuracy measure. Now let $x \in G$ be an ε-solution to (31) and suppose that $\|c\|_2 = \|\hat{c}\|_2$ holds. Then a simple calculation shows that x is an $(\varepsilon + 2\delta R)$-solution to (32). As a consequence, for sufficiently small perturbations δ one may accept the approximative solution to the first problem as an approximative solution to the second one. The following discussion will lead to an efficient algorithm for arbitrary δ.

The two objective functions given by $f(x) = \langle\, c, x \,\rangle$ as well as $\hat{f}(x) = \langle\, \hat{c}, x \,\rangle$ are linear and therefore β-compatible with $\beta = 0$ for any self-concordant barrier. Now let F be such a self-concordant barrier for G with self-concordancy parameter ϑ. Obviously, we can compute an approximation $x^{(0)}$ to the analytic center of G in the sense of (20) first, before proceeding to approximate a solution to (31) or (32). To solve both problems, we might then start from $x^{(0)}$ to execute the second stage for the first problem, and then restart from $x^{(0)}$ to solve the second problem, again executing the second stage. This, however, means that we are "recycling" or "reusing" only the starting point to solve two different problems which are usually in close connection to each other, i. e. $\|c - \hat{c}\|_2 = \delta \ll 1$. Obviously, the O-constants in (30) get multiplied by a factor of 2, independently of the size of δ.

It will now be shown that we can recycle additional steps of the main stage, and how the number of reusable steps depends on δ. To this end, one of the feasible points close to the central path of the first problem that was generated while solving the first problem will be used as a warm-start point for the second problem.

Let there be given a 5-tuple $(\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3')$ feasible for the system (15)–(19). Choose a sixth parameter $\hat{\lambda}_3$ such that $\lambda_3 < \hat{\lambda}_3$ holds and the 5-tuple $(\lambda_1, \lambda_1', \lambda_2, \hat{\lambda}_3, \lambda_3')$ is feasible for (15)–(19), too. As an example, choose $\lambda_1 = 0.193$, $\lambda_1' = (\lambda_1/(1-\lambda_1))^2 = 0.057\ldots$, $\lambda_2 = 0.150$, $\lambda_3 = 4/25 = 0.16$, $\lambda_3' = 1/9$, and $\hat{\lambda}_3 = 1/4$. Define the parameterized functions $F_t, \hat{F}_t$ in the usual way, i. e.

$$\begin{aligned} F_t(x) &:= tf(x) + F(x), \\ \hat{F}_t(x) &:= t\hat{f}(x) + F(x). \end{aligned}$$

When solving (32), we can simply replace f by $\hat{f}$, i. e. F_t by $\hat{F}_t$. Suppose that we try to solve (31) *and* (32) with the parameter tuple $(\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3')$, using $x^{(0)}$ as a starting point. According to Lemma 3.4, inequality (27) is sufficient for (28) if updates according to Newton's scheme are made, and we might therefore concentrate on (27). The idea is now to use the points $x^{(i)}$, computed as approximations to the central path of problem (31), as approximations to the central path of the problem (32). More precisely, the point $x^{(i-1)}$ will be used as an approximation to a point on the central path of problem (32) as long as

$$\lambda(\hat{F}_{t(i)}, x^{(i-1)}) < \hat{\lambda}_3 \tag{33}$$

holds.

To start with this scheme, we need a starting point $w := x^{(0)} \in G$ and a barrier parameter value $t(1)$ with $\lambda(F_{t(1)}, w) \leq \lambda_3$ as well as $\lambda(\hat{F}_{t(1)}, w) \leq \lambda_3$. According to Lemma 3.1, the value

$$t(1) := \frac{\lambda_3 - \lambda(F, w)}{\max\{\|\nabla f(w)\|^\circ_{w,F}, \|\nabla \hat{f}(w)\|^\circ_{w,F}\}} = \frac{\lambda_3 - \lambda(F, w)}{\max\{\|c\|^\circ_{w,F}, \|\hat{c}\|^\circ_{w,F}\}} \tag{34}$$

can be used as a starting value for *both* optimization problems. Then, (33) holds for $i = 1$. Assuming appropriate scaling of the objective functions, by (26) we clearly have

$$t(1) > \frac{\lambda_3 - \lambda_2}{9R \max\{\|c\|_2, \|\hat{c}\|_2\}} = O\left(\frac{1}{R}\right). \tag{35}$$

Suppose now that (33) does not hold for an $i > 1$, but that we still have that $x^{(i-2)}$ is close to the central path of the second problem, i. e. $\lambda(\hat{F}_{t(i-1)}, x^{(i-2)}) < \hat{\lambda}_3$. How is it now possible to obtain a point $\hat{x}^{(i-2)}$ for which

$$\lambda(\hat{F}_{t(i-1)}, \hat{x}^{(i-2)}) \leq \lambda_3 \tag{36}$$

holds without too much work? (Especially, without running through the whole Stage 2 of the standard path-following algorithm again?) Inequality (36) means that (27) holds with respect to $\hat{x}^{(i-2)}$. With such a point, we can go on with Stage 2 for problem (32) in the following way. Define the points $y^{(j)}_{i-2} \in \mathbb{R}^m$ by making Newton steps for the function $\hat{F}_{t(i-1)}$, starting at the point $x^{(i-2)}$: set $y^{(0)}_{i-2} := x^{(i-2)}$ and

$$y^{(j+1)}_{i-2} := y^{(j)}_{i-2} - (\nabla^2 \hat{F}_{t(i-1)}(y^{(j)}_{i-2}))^{-1} \nabla \hat{F}_{t(i-1)}(y^{(j)}_{i-2}), \tag{37}$$

$j = 0, 1, \ldots$. According to Lemma 3.4 (cmp. also Theorem 2.2.3 and especially the implication (2.2.39) from [22]), one has due to $\hat{\lambda}_3 < 2 - \sqrt{3}$ that

$$\lambda(\hat{F}_{t(i-1)}, y^{(j+1)}_{i-2}) \leq \left(\frac{\lambda(\hat{F}_{t(i-1)}, y^{(j)}_{i-2})}{1 - \lambda(\hat{F}_{t(i-2)}, y^{(j)}_{i-2})}\right)^2 \leq \frac{\lambda(\hat{F}_{t(i-1)}, y^{(j)}_{i-2})}{2} < \frac{\hat{\lambda}_3}{2}$$

($j = 0, 1, \ldots$) holds. Obviously, $\lambda(\hat{F}_{t(i-1)}, y^{(j)}_{i-2}) < \hat{\lambda}_3 / 2^j$ for $j = 0, 1, \ldots$ follows.

Therefore, if we make at least

$$k := \left\lceil \frac{\ln \hat{\lambda}_3 - \ln \lambda_3}{\ln 2} \right\rceil \tag{38}$$

Newton steps in the scheme to compute the $y^{(j)}_{i-2}$, we arrive at (36) by defining $\hat{x}^{(i-2)} := y^{(k)}_{i-2}$. Obviously, this is a worst-case estimate, it may be the case that one needs fewer iterations.

We might envision these additional Newton steps as *warm-start* or *corrector steps*. (In the example with the numerical values of the λ-parameters given as above, we obtain $k > 0.64\ldots$ as a sufficient condition for k, so we need just one corrector step and immediately

use $\hat{x}^{(i-2)} := y^{(1)}_{i-2}$.) After the corresponding corrector steps, Stage 2 of the interior-point algorithm can be used on problem (32) as usual. In this case, we have saved the computations needed for the first $i-1$ iterations (i. e. we have saved i Newton steps), and we need an additional amount of work of at most k iterations. The number of additional operations k, however, is independent of the size of the perturbation δ, since the points already generated are still relatively close to the central path of the second problem in the sense that $\lambda(\hat{F}_{t(i-1)}, x^{(i-2)}) < \hat{\lambda}_3$ holds.

But how long does (33) hold, i. e. how long are we allowed to skip iterations to solve (32) while iterating to solve (31)? According to (23), we have

$$\{z \in \mathbb{R}^m \mid \langle\, (z-x), \nabla^2 F_t(x)(z-x)\,\rangle \leq 1\} \subseteq G$$

for all $t > 0$ and $x \in \mathrm{int}(G)$, because F_t is self-concordant for all $t > 0$ (compare Proposition 3.2.2, Definition 3.1.1, and Definition 2.1.1 of [22]). Since G is bounded, $\nabla^2 F_t(x)$ is positive definite. Denote by $\mu_1(x,t), \ldots, \mu_n(x,t)$ the eigenvalues of $\nabla^2 F_t(x)$. Then, the inequality

$$\mu_i(x,t) \geq \frac{1}{4R^2} \tag{39}$$

($i = 1, \ldots, n$) holds, where $R > 0$ is, as above, a number with $\|x\|_2 \leq R$ for all $x \in G$. Obviously, this bound is independent of x and t. All this holds also if one replaces F_t by $\hat{F}_t$. But $\nabla^2 F_t(x) = \nabla^2 \hat{F}_t(x) = \nabla^2 F(x)$ and $\nabla F_t(x) = tc + \nabla F(x)$ as well as $\nabla \hat{F}_t(x) = t\hat{c} + \nabla F(x)$. As a consequence, we have

$$\begin{aligned}
\lambda(\hat{F}_t, x))^2 &= \left\langle \nabla \hat{F}_t(x), (\nabla^2 \hat{F}_t(x))^{-1} \nabla \hat{F}_t(x) \right\rangle \\
&= \left\langle t(\hat{c}-c) + tc + \nabla F(x), (\nabla^2 F(x))^{-1} \left(t(\hat{c}-c) + tc + \nabla F(x)\right) \right\rangle \\
&= t^2 \left\langle \hat{c}-c, (\nabla^2 F(x))^{-1}(\hat{c}-c) \right\rangle + 2t \left\langle \hat{c}-c, (\nabla^2 F(x))^{-1} \nabla F_t(x) \right\rangle \\
&\quad + \left\langle \nabla F_t(x), (\nabla^2 F_t(x))^{-1} \nabla F_t(x) \right\rangle \\
&= t^2 \left\langle \hat{c}-c, (\nabla^2 F(x))^{-1}(\hat{c}-c) \right\rangle - 2t \left\langle \hat{c}-c, \Delta x \right\rangle + (\lambda(F_t, x))^2,
\end{aligned}$$

where Δx is the Newton step defined by $\nabla^2 F(x) \Delta x = -\nabla F_t(x)$. Consider now the case $x = x^{(i-1)}$ and $t = t(i)$. Since $x^{(i)}, x^{(i-1)} \in G$ (cmp. [22, Proposition 3.2.4 (i)]), the inequality $\|\Delta x\|_2 = \|x^{(i)} - x^{(i-1)}\|_2 \leq 2R$ immediately follows. (This is a global, rather conservative estimate. What is important in what follows is that this estimate is independent of i.) Moreover, the largest eigenvalue of $(\nabla^2 F^{(2)}_t(x))^{-1}$ is bounded above by $4R^2$ uniformly in x and t (see (39)), and $\lambda(F_{t(i)}, x^{(i-1)}) \leq \lambda_3$ holds for all i (see Lemma 3.4). Therefore, the chain of equations above leads to

$$\lambda(\hat{F}_{t(i)}, x^{(i-1)}) \leq \left(4\delta^2 R^2 (t(i))^2 + 4t(i)\delta R + \lambda_3^2\right)^{1/2}$$

(recall $\delta = \|c - \hat{c}\|_2$), which means that $\lambda(\hat{F}_{t(i)}, x^{(i-1)}) < \hat{\lambda}_3$ holds as long as

$$4\delta^2 R^2 (t(i))^2 + 4t(i)\delta R + \lambda_3^2 < \hat{\lambda}_3^2$$

holds. Therefore, we arrive at

$$t(i) < \frac{(\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1}{2\delta R} = O\left(\frac{1}{\delta R}\right) = O\left(\frac{t(1)}{\delta}\right). \tag{40}$$

The update rule for $t(i)$ is (22), while $t(1)$ is initialized as in (34). Solving for i yields

$$\begin{aligned} i \;<\; 1 + \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \Big(&\ln(\max\{\|c\|_{w,F}^{\circ}, \|\hat{c}\|_{w,F}^{\circ}\}) - \ln R - \ln \delta \\ &+ \ln((\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1) - \ln(\lambda_3 - \lambda_2) - \ln 2\Big). \end{aligned}$$

Note that in this estimate, the term $\ln \vartheta$ does not appear. Define

$$K := \max\{\|c\|_{w,F}^{\circ}, \|\hat{c}\|_{w,F}^{\circ}\}$$

and abbreviate the constant terms by

$$\tau = \tau(\lambda_2, \lambda_3, \hat{\lambda}_3) := \ln((\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1) - \ln(\lambda_3 - \lambda_2) - \ln 2. \tag{41}$$

It is clear that we does not need to take a look at problem (32) for the first

$$1 + \left\lceil \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} (\ln K - \ln R - \ln \delta + \tau) \right\rceil \tag{42}$$

iterations of the primal interior-point method applied to solve (31). Only after that we need to *branch*, i. e. take a look at two separate problems. (If the number above is less than zero, it is understood that branching is immediately necessary.) Note that this is a worst-case estimate, i. e. it may be the case that one is allowed to make more iterations before switching into a "parallel" process by branching. Moreover, if δ is replaced by, e. g., $\delta/2$, the number of iterations we are allowed to execute before branching increases by $O(\sqrt{\vartheta})$. We will return to this observation in Section 7. (Using a larger value for K when defining $t(1)$, like the one given in estimate (35), will *increase* the number of iterations in which (33) holds.)

Recall that $\varepsilon > 0$ is the prespecified solution accuracy that we want to obtain and suppose that

$$\delta \geq \varepsilon^2 \frac{(\lambda_3 - \lambda_2)^2 e^{\tau}}{81(2\vartheta + \zeta(\lambda_3')/2)^2 KR} = O\left(\frac{\varepsilon^2}{\vartheta^2 KR}\right) \tag{43}$$

holds. (This just means that ε is "small" compared to the perturbation δ.) Taking the result obtained above together and considering (30), we see that

$$2 + 2\frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \left(\ln\left(2\vartheta + \frac{\zeta(\lambda_3')}{2}\right) + \ln(1/\varepsilon) + \ln K + \ln\left(\frac{9}{\lambda_3 - \lambda_2}\right)\right)$$

$$
\begin{aligned}
& -\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}(\ln K-\ln R-\ln\delta+\tau)-1+k \\
= \;& \frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(2\ln\left(2\vartheta+\frac{\zeta(\lambda_3')}{2}\right)+2\ln(1/\varepsilon)+\ln K+\ln R+\ln\delta\right. \qquad (44)\\
& \left.+2\ln\left(\frac{9}{\lambda_3-\lambda_2}\right)-\tau\right)+k+1
\end{aligned}
$$

Newton-steps are needed instead of

$$
2+2\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(\ln\left(2\vartheta+\frac{\zeta(\lambda_3')}{2}\right)+\ln(1/\varepsilon)+\ln K+\ln\left(\frac{9}{\lambda_3-\lambda_2}\right)\right)
$$

Newton-steps in Stage 2 to achieve an accuracy of ε for both problems. The lower bound (43) just ensures that we have not obtained ε-accuracy before branching and there is still some work to do, and that therefore the estimate above on the number of steps is a nonnegative number. Instead of introducing (43), one might replace (44) by an estimate of the form $\max\{0,\ldots\}$, where the dots stand for (44).

If $\delta<1$, there are savings in computation time proportional to

$$
-\sqrt{\vartheta}(\ln R+\ln\delta)=\sqrt{\vartheta}\ln(1/R\delta)
$$

iterations. Obviously, substantial savings will take place if $\delta<1/R$. Again, note that this estimate is rather conservative, mainly due to the fact that the inequality $\|x^{(i)}-x^{(i-1)}\|_2\le 2R$ has been used to derive (40). Note also that Stage I is still needed, though.

The following algorithm summarizes the main points of the scheme, and the next theorem restates the results found by the analysis above.

Algorithm 4.1

1. Choose $(\lambda_1,\lambda_1',\lambda_2,\lambda_3,\lambda_3')\in\mathbb{R}^5$ feasible for the system (15)–(19).

   Choose $\hat\lambda_3\in\mathbb{R}$ such that $\lambda_3<\hat\lambda_3$ holds and such that $(\lambda_1,\lambda_1',\lambda_2,\hat\lambda_3,\lambda_3')$ is feasible for (15)–(19), too.

2. Compute a point $x^{(0)}\in\operatorname{int}(G)$ such that (20) holds, i. e. $\lambda(F,x^{(0)})<\lambda_2$. (This is Stage I and can be done along the lines of [22, Section 3.2.3].)

3. (Initialization of barrier parameter, compare (34).)

$$
t(1):=\frac{\lambda_3-\lambda(F,x^{(0)})}{\max\{\|c\|^\circ_{x^{(0)},F},\|\hat c\|^\circ_{x^{(0)},F}\}}
$$

4. $i:=1$

5. (Solve the first problem.) `WHILE NOT` stopping criterion for problem (31) fulfilled `DO`

   (a) (Do a Newton-step, compare (21).)

   $$x^{(i)} := x^{(i-1)} - \left(\nabla_x^2 F_{t(i)}(x^{(i-1)})\right)^{-1} \nabla_x F_{t(i)}(x^{(i-1)})$$

   (b) (Update barrier parameter compare (22).)

   $$t(i+1) := t(i) \exp\left(\frac{\lambda_3 - \lambda_3'}{\lambda_3 + \sqrt{\vartheta}}\right)$$

   (c) $i := i + 1$

6. (Compute index of branching step.)

   $$i := \max\{i \geq 0 \mid \lambda(\hat{F}_{t(i)}, x^{(i-1)}) < \hat{\lambda}_3\}$$

7. $y_{i-1}^{(0)} := x^{(i-1)}$, $j := 1$

8. (Branching) `WHILE` $\lambda(\hat{F}_{t(i)}, y_{i-1}^{(j-1)}) \geq \lambda_3$ `DO`

   (a) (Do a corrector Newton-step, compare (37).)

   $$y_{i-1}^{(j)} := y_{i-1}^{(j-1)} - \left(\nabla_x^2 \hat{F}_{t(i)}(y_{i-1}^{(j-1)})\right)^{-1} \nabla_x \hat{F}_{t(i)}(y_{i-1}^{(j-1)})$$

   (b) $j := j + 1$

9. $\hat{x}^{(i-1)} := y_{i-1}^{(j-1)}$

10. (Solve the second problem.) `WHILE NOT` stopping criterion for problem (32) fulfilled `DO`

    (a) (Do a Newton-step, compare (21).)

    $$\hat{x}^{(i)} := \hat{x}^{(i-1)} - \left(\nabla_x^2 \hat{F}_{t(i)}(\hat{x}^{(i-1)})\right)^{-1} \nabla_x \hat{F}_{t(i)}(\hat{x}^{(i-1)})$$

    (b) (Update barrier parameter, compare (22).)

    $$t(i+1) := t(i) \exp\left(\frac{\lambda_3 - \lambda_3'}{\lambda_3 + \sqrt{\vartheta}}\right)$$

    (c) $i := i + 1$

Theorem 4.2 *Let $G \subseteq \mathbb{R}^m$ be convex and compact with nonempty interior and let F be a self-concordant barrier for G with self-concordancy parameter ϑ. Moreover, let $\|x\|_2 \leq R$ hold for all $x \in G$ for a number $R > 0$. Let $c, \hat{c} \in \mathbb{R}^m$ with $c \neq 0 \neq \hat{c}$ and let there be given an accuracy measure $\varepsilon > 0$. With the λ-parameters chosen in Step 1 of Algorithm 4.1 and $x^{(0)}$ computed as in Step 2, define*

$$\tau = \tau(\lambda_2, \lambda_3, \hat{\lambda}_3) := \ln((\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1) - \ln(\lambda_3 - \lambda_2) - \ln 2$$

as well as

$$K := \max\{\|c\|^\circ_{x^{(0)},F}, \|\hat{c}\|^\circ_{x^{(0)},F}\}.$$

Suppose that

$$\|c - \hat{c}\|_2 \geq \varepsilon^2 \frac{(\lambda_3 - \lambda_2)((\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1)}{162(2\vartheta + \zeta(\lambda_3')/2)^2 K R}$$

as well as $0 < \varepsilon K < 1$ holds. Then, the total number of Newton-steps in Steps 5–10 of Algorithm 4.1 needed to achieve ε-accuracy for both optimization problems is bounded above by

$$\frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \left(2 \ln \left(2\vartheta + \frac{\zeta(\lambda_3')}{2} \right) + 2 \ln(1/\varepsilon) + \ln K + \ln R + \ln \|c - \hat{c}\|_2 \right.$$
$$\left. + 2 \ln \left(\frac{9}{\lambda_3 - \lambda_2} \right) - \tau \right) + \left\lceil \frac{\ln \hat{\lambda}_3 - \ln \lambda_3}{\ln 2} \right\rceil + 1.$$

Note that the logarithm of the perturbation size, $\ln \|c - \hat{c}\|_2$, is *added* to inside the bracketed term! Accordingly, if $\|c - \hat{c}\|_2 < 1$, this logarithm is negative and a certain amount of work is *saved* due to the warm-start technique employed here.

Note also that the lower bound on the perturbation size $\delta = \|c - \hat{c}\|_2$ in the theorem above was introduced just for convenience. For smaller δ it might be the case that the `WHILE`-loop in Step 10 is never executed.

We would like to remark here that the exact computation of the O-constants of the run-time behaviour of Algorithm 4.1, as done above, is, unfortunately, not "computational gymnastics" [25], although we wished that this would be the case. It turns out that we need precise knowledge of these constants to equilibrate the amount of work necessary for solving a large number of optimization problems (and not just two of them) between different parts of a corresponding algorithm. This is the subject of Section 7.

Let us also stress that the number R enters the run-time estimate of the theorem, while it is not used in algorithm. No knowledge on G is needed other than boundedness and the fact we know a self-concordant barrier F for G.

The approach presented above uses only primal information and makes only few assumptions about the feasible set G. (Most importantly, that G is compact and that we know

a self-concordant barrier for it.) If primal-dual algorithms are considered, care has to be taken with respect to the (strict) feasibility of the iterates considered. Indeed, if a primal objective function is perturbed, the perturbation shows up in the dual constraints, and it is not *a priori* clear how dual feasibility might be guaranteed. Up to now, only primal-dual schemes for problems with special structure, most notably with linear constraints and linear objective functions [28, 11] or convex-quadratic objective functions [9] have been considered.

5 Solving Two Single-Criteria Programs with Quadratic Objective Functions

Suppose now that we have given $c, \hat{c} \in \mathbb{R}^m$ as well as $Q, \hat{Q} \in \mathbb{R}^{m \times m}$ positive semidefinite. Define $\delta_c := \|c - \hat{c}\|_2$ and $\delta_Q := \|Q - \hat{Q}\|$. Again, let G be a compact convex set with nonempty interior such that there is given a number $R > 0$ with $\|x\|_2 \leq R$ for all $x \in G$. We are interested in solving the two quadratic optimization problems

$$\begin{array}{ll} \text{minimize} & \langle\, c, x \,\rangle + \dfrac{1}{2}\langle\, x, Qx \,\rangle \\ \text{subject to} & x \in G \end{array} \tag{45}$$

and

$$\begin{array}{ll} \text{minimize} & \langle\, \hat{c}, x \,\rangle + \dfrac{1}{2}\langle\, x, \hat{Q}x \,\rangle \\ \text{subject to} & x \in G \end{array} \tag{46}$$

Both objective functions $f(x) := \langle\, c, x \,\rangle + (1/2)\langle\, x, Qx \,\rangle$ and $\hat{f}(x) := \langle\, \hat{c}, x \,\rangle + (1/2)\langle\, x, \hat{Q}x \,\rangle$ are β-compatible with $\beta = 0$ for any self-concordant barrier. Let F be such a self-concordant barrier for G with self-concordancy parameter ϑ.

Using the same strategy and notation as in the last section, we see that we need an estimate for t such that $\lambda(\hat{F}_t, x) < \hat{\lambda}_3$ holds for all $x \in \text{int}(G)$ with $\lambda(F_t, x) < \lambda_3$. By definition, we have

$$\begin{aligned} (\lambda(\hat{F}_t, x))^2 &= \left\langle \nabla\hat{F}_t(x), (\nabla^2\hat{F}_t(x))^{-1}\nabla\hat{F}_t(x) \right\rangle \\ &= \left\langle t\nabla\hat{f}(x) + \nabla F(x), (\nabla^2\hat{F}_t(x))^{-1}(t\nabla\hat{f}(x) + \nabla F(x)) \right\rangle, \end{aligned}$$

and due to $t\nabla\hat{f}(x) + \nabla F(x) = t(\nabla\hat{f}(x) - \nabla f(x)) + \nabla F_t(x))$ we arrive at

$$\begin{aligned} &(\lambda(\hat{F}_t, x))^2 \\ &= t^2 \left\langle \nabla\hat{f}(x) - \nabla f(x), (\nabla^2\hat{F}_t(x))^{-1}(\nabla\hat{f}(x) - \nabla f(x)) \right\rangle \\ &\quad + 2t \left\langle \nabla\hat{f}(x) - \nabla f(x), (\nabla^2\hat{F}_t(x))^{-1}\nabla F_t(x) \right\rangle \\ &\quad + \left\langle \nabla F_t(x), (\nabla^2\hat{F}_t(x))^{-1}\nabla F_t(x) \right\rangle. \end{aligned}$$

Since $\hat{F}_t$ is self-concordant, the eigenvalues of $\nabla^2 \hat{F}_t(x)$ are bounded below by $1/4R^2$. The estimate

$$\begin{aligned} & \left\langle \nabla \hat{f}(x) - \nabla f(x), (\nabla^2 \hat{F}_t(x))^{-1} (\nabla \hat{f}(x) - \nabla f(x)) \right\rangle \\ \leq \; & 4R^2 \|\nabla \hat{f}(x) - \nabla f(x)\|_2^2 \end{aligned} \tag{47}$$

immediately follows. Suppose that the sequence $(x^{(i)})_{i \in \mathbb{N}}$ ($x^{(i)} \in G$) is generated by the standard primal path-following algorithm applied to (45) with the barrier F. Write $\nabla^2 \hat{F}_t(x) = t\nabla^2 \hat{f}(x) + \nabla^2 F(x) = t(\nabla^2 \hat{f}(x) - \nabla^2 f(x)) + \nabla^2 F_t(x)$. Together with $x^{(i)} - x^{(i+1)} = (\nabla^2 F_t(x))^{-1} \nabla F_t(x)$, this leads to

$$\begin{aligned} & (\nabla^2 \hat{F}_t(x))^{-1} \nabla F_t(x) \\ = \; & (I - t(\nabla^2 \hat{F}_t(x))^{-1} (\nabla^2 \hat{f}(x) - \nabla^2 f(x)))(x^{(i)} - x^{(i+1)}), \end{aligned} \tag{48}$$

and therefore

$$\begin{aligned} & \left\langle \nabla \hat{f}(x) - \nabla f(x), (\nabla^2 \hat{F}_t(x))^{-1} \nabla F_t(x) \right\rangle \\ \leq \; & \|\nabla \hat{f}(x) - \nabla f(x)\|_2 \|x^{(i)} - x^{(i+1)}\|_2 (1 + 4R^2 t \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\|) \\ \leq \; & 2R \|\nabla \hat{f}(x) - \nabla f(x)\|_2 (1 + 4R^2 t \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\|) \end{aligned} \tag{49}$$

holds. Moreover, using (48) twice again leads to

$$\begin{aligned} & \left\langle \nabla F_t(x), (\nabla^2 \hat{F}_t(x))^{-1} \nabla F_t(x) \right\rangle \\ = \; & (\lambda(F_t, x))^2 - t \left\langle \nabla F_t(x), (\nabla^2 \hat{F}_t(x))^{-1} (\nabla^2 \hat{f}(x) - \nabla^2 f(x)) (\nabla^2 F_t(x))^{-1} \nabla F_t(x) \right\rangle \\ < \; & \lambda_3^2 + t \|x^{(i)} - x^{(i+1)}\|_2 \|(\nabla^2 \hat{F}_t(x))^{-1} \nabla F_t(x)\|_2 \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\| \\ \leq \; & t \|x^{(i)} - x^{(i+1)}\|_2^2 \|I - t(\nabla^2 \hat{F}_t(x))^{-1} (\nabla^2 \hat{f}(x) - \nabla^2 f(x))\| \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\| \\ & + \lambda_3^2 \end{aligned}$$

which results in

$$\begin{aligned} & \left\langle \nabla F_t(x), (\nabla^2 \hat{F}_t(x))^{-1} \nabla F_t(x) \right\rangle \\ < \; & \lambda_3^2 + 4R^2 t (1 + 4R^2 t \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\|) \|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\|. \end{aligned} \tag{50}$$

By noting $\|\nabla \hat{f}(x) - \nabla f(x)\|_2 \leq \delta_c + \delta_Q R$, $\|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\| = \delta_Q$ as well as (47), (49), and (50), we arrive at

$$\begin{aligned} (\lambda(\hat{F}_t, x))^2 \; < \; & 4R^2 (\delta_c + \delta_Q R)^2 t^2 + 4Rt(\delta_c + \delta_Q R)(1 + 4R^2 \delta_Q t) \\ & + \lambda_3^2 + 4R^2 \delta_Q t (1 + 4R^2 \delta_Q t) \\ = \; & 4R^2 (\delta_c^2 + 6R\delta_c \delta_Q + 9R^2 \delta_Q^2) t^2 + 4R(2R\delta_Q + \delta_c) t + \lambda_3^2 \\ = \; & 4R^2 (\delta_c + 3R\delta_Q)^2 t^2 + 4R(\delta_c + 2R\delta_Q) t + \lambda_3^2. \end{aligned}$$

Accordingly, $\lambda(\hat{F}_t, x) < \hat{\lambda}_3$ holds as long as we have

$$4R^2(\delta_c + 3R\delta_Q)^2 t^2 + 4R(\delta_c + 2R\delta_Q)t + \lambda_3^2 < \hat{\lambda}_3^2,$$

and solving for t yields

$$t < \frac{\delta_c + 2R\delta_Q}{2R(\delta_c + 3R\delta_Q)^2} \left(\left(1 + \hat{\lambda}_3^2 - \lambda_3^2\right)^{1/2} - 1 \right). \tag{51}$$

It is easy to see that $(\delta_c + 2R\delta_Q)/(\delta_c + 3R\delta_Q)^2 \geq 1/(\delta_c + (9/2)R\delta_Q)$ holds. Therefore,

$$t < \frac{\left(1 + \hat{\lambda}_3^2 - \lambda_3^2\right)^{1/2} - 1}{2R(\delta_c + (9/2)R\delta_Q)} =: s$$

is sufficient for $\lambda(\hat{F}_t, x) < \hat{\lambda}_3$. Clearly, this estimate is a generalization of (40). In virtually the same way as in the last section, we conclude that

$$1 + \left\lceil \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} (\ln K - \ln R - \ln(\delta_c + (9/2)R\delta_Q) + \tau) \right\rceil \tag{52}$$

iterations of the primal interior-point method applied to solve problem (45) can be made before considering problem (46). Here,

$$K := \max\{\|c + Qx^{(0)}\|_{x^{(0)},F}^{\circ}, \|\hat{c} + \hat{Q}x^{(0)}\|_{x^{(0)},F}^{\circ}\}.$$

Likewise,

$$\begin{aligned} \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \Bigg(& 2\ln\left(2\vartheta + \frac{\zeta(\lambda_3')}{2}\right) + 2\ln(1/\varepsilon) + \ln K + \ln R \\ & + \ln\left(\delta_c + \frac{9}{2}R\delta_Q\right) + 2\ln\left(\frac{9}{\lambda_3 - \lambda_2}\right) - \tau \Bigg) + k + 1 \end{aligned}$$

Newton-steps are needed to solve both problems to ε-optimality. Note that with $\delta := \max\{\delta_c, \delta_Q\}$ we have that

$$\ln R + \ln(\delta_c + (9/2)R\delta_Q) = O\left(\ln \delta + \ln R\right),$$

so, from the point of view of complexity theory, the situation is not worse than in the linear case. On the other hand, it has to be noted that the computation of the quantity $\lambda(\hat{F}_t, x)$ needs as many operations as a Newton step for the function $\hat{F}_t$. This is in stark contrast to the linear case, where $\nabla_x^2 \hat{F}_t(x) = \nabla_x^2 F_t(x)$ holds and it is therefore possible to reuse the factorization of the corresponding matrices. As such, as long as we do not want to fall back on heuristics to estimate the number of steps we are allowed to recycle, we have that at the present moment only the estimate (51) is available for saving unnecessary work during the course of solving two quadratic programs at once. For some heuristics useful in a primal-dual framework as well as computational experience with them, see [16, 9].

6 Solving Two Single-Criteria Programs with General Objective Functions

It is easy to see that the ideas from the last section carry over to general case of two smooth convex objective functions $f, \hat{f} \in C^3(G)$. Indeed, if uniform bounds δ_1, δ_2 of the form $\|\nabla \hat{f}(x) - \nabla f(x)\|_2 \leq \delta_1$, $\|\nabla^2 \hat{f}(x) - \nabla^2 f(x)\| \leq \delta_2$ for all $x \in G$ are known, the estimates (47), (49), and (50) immediately lead to

$$\begin{aligned}(\lambda(\hat{F}_t, x))^2 &\leq 4R^2\delta_1^2 t^2 + 4R\delta_1 t(1 + 4R^2\delta_2 t) + \lambda_3^2 + 4R^2\delta_2 t(1 + 4R^2\delta_2 t) \\ &= 4R^2(\delta_1 + 2R\delta_2)^2 t^2 + 4R(\delta_1 + R\delta_2)t + \lambda_3^2\end{aligned}$$

and an analysis similar to the one in the last two sections will lead to similar results, provided that the parameter needed in defining β-compatibility is used in the interior-point method in the standard way (cmp. (22) and [22, Chapter 3]). This case, however, is of minor interest when discussing the use of interior-point methods for multicriteria problems, since the discussion in Subsection 2.2 has shown that parameterized linear or parameterized quadratic problems are sufficient for approximating the set of efficient points.

7 Solving Many Convex Programs "Simultaneously"

Suppose now that two cost vectors $c_1, c_2 \in \mathbb{R}^m$ are given and that we want to solve "all" the problems with the set of feasible points G and with cost vectors $c(\mu) := \mu c_1 + (1 - \mu)c_2$ ($\mu \in]\,0, 1[$). More precisely, we are in search for an approximation of the set

$$\bigcup_{\mu \in]\,0,1[} \arg\min\{\langle\, c(\mu), x \,\rangle \mid x \in G\} \tag{53}$$

by a discrete set of points. This is exactly the situation in standard bicriteria optimization.

In a standard approach, our task amounts to discretizing the parameter interval $[\,0, 1\,]$ by a (prespecified) number of parameters μ_i such that $\|c(\mu_i) - c(\mu_{i+1})\|_2 \leq \delta$ holds with a prespecified accuracy measure $\delta \in]\,0, 1[$. Under the assumption $\|c_1\|_2, \|c_2\|_2 \leq 1$ (made without loss of generality) we might use any discretization μ_i of $[\,0, 1\,]$ for which $|\mu_i - \mu_{i+1}| \leq \delta/2$ holds. With this, we basically have to solve $2/\delta$ different standard optimization problems with the same set of feasible points G.

In what follows, a numerical scheme is described that uses the ideas from the last section to approximate the set (53). First, a short run-down of the general idea is presented, before the actual algorithm is described and its complexity is analyzed.

7.1 The General Idea

Using the results from Section 4, we start by defining $\delta_0 := 1/2$ and $\mu_0 := 1/2$. In this way, instead of solving two problems "simultaneously", we execute Newton iterations for all problems with cost vectors $c(\mu)$ for which $|\mu - \mu_0| \leq \delta_0$ holds (i. e. all problems we have to consider) up to the time when we have to "branch". (Not counting the ubiquitous Stage 1, this takes

$$\frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} (\ln K - \ln R + \ln 2 + \tau)$$

Newton iterations, where

$$K := \max\{\|c(\mu)\|_{x^{(0)},F} \mid \mu \in [0,1]\} \leq 2R.)$$

After that, a "branch" is needed to proceed. Define $\delta_1 :== \delta_0/2 = 1/4$, $\mu_1^{(1)} := 1/4$, and $\mu_1^{(2)} := 3/4$. Now, the two problems corresponding to the parameter values $\mu_1^{(1)}, \mu_1^{(2)}$ are considered. In order to reach the neighborhoods of their central paths, at most $2k$ corrector steps, where k is given as in (38), are needed. After these corrector steps, Newton iterations are executed for all problems with parameters μ for which $|\mu - \mu_1^{(1)}| \leq \delta_1$ holds until we have to branch and after that, Newton iterations are executed for all problems with parameters μ for which $|\mu - \mu_1^{(2)}| \leq \delta_1$ until branching is needed. (Again, this means that all possible problems are considered. Moreover, by the estimate (42) on the number of steps saved, it is clear that

$$\frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \ln 2$$

Newton-steps are made in each branch before the process stops and we have to branch again.)

Then, define $\delta_2 := 1/8$, $\mu_2^{(1)} = 1/8$, $\mu_2^{(2)} = 3/8$, $\mu_2^{(3)} = 5/8$, $\mu_2^{(4)} = 7/8$, make $4k$ corrector steps, and repeat the scheme.

This scheme of branching and refining will be used until we arrive at a stage N with $\delta_N = 1/2^{N+1} \leq \delta$. Then, $1/\delta_N \leq 1/\delta$ different problems are considered and we can now solve them to ε-optimality in the standard way. The different parameters μ_i, $i = 1, \ldots, 2^{N+1}$, considered after the final branching step define a δ-*covering* of the unit interval $[0,1]$, i. e. $\mu_i \in [0,1]$ for all i and for each $\mu \in [0,1]$ there exists an i with $|\mu - \mu_i| < \delta$. (In fact, if equality is allowed in the last inequality, the points μ_i ($i = 1, \ldots, 2^{N+1}$) define a δ_N-covering.)

7.2 The Algorithm

The scheme presented in the last section can be described more formally by the following algorithm. Each problem is identified with its parameter $\mu \in [0,1]$, and the central path of each of these problems is followed by the standard path-following method. With respect

to bookkeeping, for each problem only one vector of variables $x_\mu \in G$ and one barrier parameter $t_\mu > 0$ is needed. In what follows, the abbreviations

$$\begin{aligned} c(\mu) &:= \mu c_1 + (1-\mu)c_2, \\ f_\mu(x) &:= \langle\, c(\mu), x \,\rangle, \\ F_t^\mu(x) &:= t f_\mu(x) + F(x) \end{aligned}$$

will be used. Within the algorithm, the discrete sets $S(j) \subset [0,1]$ represent the set of parameters which are considered as problem parameters in the path-following scheme.

Algorithm 7.1 (Approximating the Efficient Set of a Bicriteria Program)

1. Choose $(\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3') \in \mathbb{R}^5$ feasible for the system (15)–(19).

   Choose $\hat{\lambda}_3 \in \mathbb{R}$ such that $\lambda_3 < \hat{\lambda}_3$ holds and such that $(\lambda_1, \lambda_1', \lambda_2, \hat{\lambda}_3, \lambda_3')$ is feasible for (15)–(19), too.

2. Choose a discretization size $0 < \delta < 1$.

3. Compute a point $x \in \text{int}(G)$ such that (20) holds, i. e. $\lambda(F, x) < \lambda_2$. (This is Stage I and can be done along the lines of [22, Section 3.2.3].)

4. (Initialization of the barrier parameter, compare (34) and Lemma 3.1.)

$$t := \frac{\lambda_3 - \lambda(F,x)}{\max\{\|c_1\|^\circ_{x,F}, \|c_2\|^\circ_{x,F}\}}$$

5. $\delta(0) := 1/2$, $\mu := 1/2$, $S(0) := \{\mu\}$, $j := 0$, $x_\mu := x$, $t_\mu := t$

6. `WHILE` $\delta(j) > \delta$ `DO`

   (a) `FORALL` $\mu \in S(j)$ `DO`

      i. `WHILE` $\lambda(F^\mu_{t_\mu}, x_\mu) \geq \lambda_3$ `DO`
      (corrector steps for $F^\mu_{t_\mu}$)

$$x_\mu := x_\mu - \left(\nabla_x^2 F^\mu_{t_\mu}(x_\mu)\right)^{-1} \nabla_x F^\mu_{t_\mu}(x_\mu)$$

      ii. `WHILE` $\lambda(F^{\mu+\delta(j)}_{t_\mu}, x_\mu) < \hat{\lambda}_3$ `AND` $\lambda(F^{\mu-\delta(j)}_{t_\mu}, x_\mu) < \hat{\lambda}_3$ `AND NOT` stopping criterion for problem with parameter μ fulfilled `DO`
      (Path following for the problem with parameter μ until branching is necessary.)

A. (Newton step)

$$x_\mu := x_\mu - \left(\nabla_x^2 F_{t_\mu}^\mu(x_\mu)\right)^{-1} \nabla_x F_{t_\mu}^\mu(x_\mu)$$

B. (Update barrier parameter.)

$$t_\mu := t_\mu \exp\left(\frac{\lambda_3 - \lambda_3'}{\lambda_3 + \sqrt{\vartheta}}\right)$$

(b) (Now x_μ represents the point along the central path for the problem with parameter μ "as far away" from the analytic center x as possible.)

$j := j + 1$, $\delta(j) := \delta(j-1)/2$

(c) `FORALL` $\mu \in S(j-1)$ `DO` (branch)

$x_{\mu-\delta(j)} := x_\mu$, $x_{\mu+\delta(j)} := x_\mu$, $t_{\mu-\delta(j)} := t_\mu$, $t_{\mu+\delta(j)} := t_\mu$

(d) (refinement) $S(j) := \{\mu + \delta(j), \mu - \delta(j) \mid \mu \in S(j-1)\}$

7. (Solve all problems in $S(j)$ to the accuracy specified.)

`FORALL` $\mu \in S(j)$ `DO`

(a) (Corrector steps for the last refinement.) `WHILE` $\lambda(F_{t_\mu}^\mu, x_\mu) \geq \lambda_3$ `DO`

$$x_\mu := x_\mu - \left(\nabla_x^2 F_{t_\mu}^\mu(x_\mu)\right)^{-1} \nabla_x F_{t_\mu}^\mu(x_\mu)$$

(b) `WHILE NOT` stopping criterion for problem with parameter μ fulfilled `DO`

i. (Newton step)

$$x_\mu := x_\mu - \left(\nabla_x^2 F_{t_\mu}^\mu(x_\mu)\right)^{-1} \nabla_x F_{t_\mu}^\mu(x_\mu)$$

ii. (Update barrier parameter.)

$$t_\mu := t_\mu \exp\left(\frac{\lambda_3 - \lambda_3'}{\lambda_3 + \sqrt{\vartheta}}\right)$$

7.3 Analysis

As usual, define

$$K := \max\{\|c(\mu)\|_{x,F}^\circ \mid \mu \in [0,1]\} = \max\{\|c_1\|_{x,F}^\circ, \|c_2\|_{x,F}^\circ\},$$

where $x \in \text{int}(G)$ is the starting point computed in Step 3 of Algorithm 7.1. (Proper scaling of c_1 and c_2 ensures $K \leq 9R$.) At the jth step of the `WHILE`-loop of Step 6 of the algorithm ($j = 0, 1, \ldots$), the set $S(j)$ is a $\delta(j)$-covering of the parameter set $[0,1]$, and we have

$\delta(j) = 1/2^{j+1}$ as well as $|S(j)| = 2^j$ Moreover, for $j > 0$ we have $|S(j)| = 1/\delta(j-1)$. The number of times the `WHILE`-loop of Step 6 is executed is given by

$$N := \left\lceil \frac{\ln(1/\delta)}{\ln 2} \right\rceil - 1.$$

The total number of all elements in all sets $S(j)$ considered, i. e. the number of branches made is $\sum_{j=0}^{N-1} |S(j)| = \sum_{j=0}^{N-1} 2^j = 2^N - 1 < 1/\delta - 1$. As a consequence, the total number of corrector steps in Step 6 (a) (i) is at most $k(1/\delta - 1)$, where k is the constant given by (38). An additional number of at most $kS(N) \leq k/\delta$ corrector steps is executed in Step 7 (a). Therefore, the total number of corrector steps executed by the algorithm is bounded above by $2k/\delta - k$.

Assume now in what follows that with respect to the amount of work done the worst case occurs, i. e. the stopping criterion of the `WHILE`-loop of Step 6 (a) ii is fulfilled as soon as possible. Then, the smallest possible number of Newton steps has been executed since the last branch. According to the analysis in Section 4 (see especially (42)), there will be

$$I := 2 + \left\lceil \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \left(\ln K - \ln R + \ln 2 + \tau \right) \right\rceil$$

Newton steps made before the first branch occurs. Now let $j \in \{1, \ldots, N-1\}$. Since $\delta(j)$ was replaced by $\delta(j-1)/2$ in previous execution of the `WHILE`-loop of Step 6, it is clear by the estimate (42) on the number of steps saved that at most

$$1 + \left\lceil \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \ln 2 \right\rceil$$

Newton-steps are executed for each problem considered before branching occurs again. (With the asymptotically optimal (for large ϑ) values for λ_3 and λ_3' given on p. 11, the bracketed number is approximatively $4.922(0.229 + \sqrt{\vartheta}) \geq 6.050$.) This number has to be multiplied with the number of problems considered during this run through the `WHILE`-loop of Step 6 of Algorithm 7.1, which is 2^j. Using the same estimate as above, the total number of Newton steps made in the `WHILE`-loop of Step 6 after the first branch is bounded above by

$$\left(\frac{1}{\delta} - 1\right) \left(\left\lceil \ln 2 \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \right\rceil + 1 \right).$$

Therefore, in the worst case the total number of Newton steps executed in Step 6 and 7 (a) is at most

$$\begin{aligned} & I + \left(\frac{1}{\delta} - 1\right) \left(\left\lceil \ln 2 \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \right\rceil + 1 \right) + \frac{2k}{\delta} - k \\ \leq \; & 3 + \frac{\lambda_3 + \sqrt{\vartheta}}{\lambda_3 - \lambda_3'} \left(\ln K - \ln R + \ln 2 + \tau \right) \end{aligned}$$

$$
\begin{aligned}
&+\left(\frac{1}{\delta}-1\right)\left(\ln 2\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}+2\right)+\frac{2k}{\delta}-k \\
\leq\ & 1-k+\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(\frac{\ln 2}{\delta}+\ln K-\ln R+\tau\right)+\frac{2k+2}{\delta}. \qquad (54)
\end{aligned}
$$

But after the execution of Step 7 (a), we have —for each of the $S(N)=2^N<1/\delta$ different single-criterion problems now considered— found a point close to its central path as if we had made at least

$$
\begin{aligned}
& I-1+N\left\lfloor \ln 2\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\right\rfloor \\
\geq\ & I-1+\left(\frac{\ln(1/\delta)}{\ln 2}-1\right)\left(\ln 2\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}-1\right) \\
\geq\ & 1+\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(\ln K-\ln R+\ln 2+\tau\right)+\left(\frac{\ln(1/\delta)}{\ln 2}-1\right)\left(\ln 2\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}-1\right) \\
=\ & 2+\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(\ln K-\ln R+\tau+\ln(1/\delta)\right)-\frac{\ln(1/\delta)}{\ln 2}
\end{aligned}
$$

Newton steps in Stage 2 of the standard primal path-following algorithm. Now suppose that an accuracy measure $\varepsilon>0$ is specified for all problems considered. By assuming

$$
\delta\geq\varepsilon\frac{(\lambda_3-\lambda_2)e^{\tau}}{9R(2\vartheta+\zeta(\lambda_3')/2)}
$$

and using the estimate above as well as (30), we see that we need at most

$$
\begin{aligned}
&\frac{\ln(1/\delta)}{\ln 2} \\
&\quad+\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}\left(\ln\left(2\vartheta+\frac{\zeta(\lambda_3')}{2}\right)+\ln(1/\varepsilon)+\ln\left(\frac{9}{\lambda_3-\lambda_2}\right)+\ln R-\tau+\ln\delta\right) \qquad (55)
\end{aligned}
$$

further Newton steps for each of the problems considered to solve it to ε-optimality, which is exactly what is done in Step 7 (b) of Algorithm 7.1. The lower bound on δ was introduced for the same reasons as in Section 4: it is convenient to assume that none of the problems considered is already solved to ε-accuracy in Step 6 of our algorithm. If this assumption is not satisfied, the algorithm will be even faster. Multiplying (55) by $1/\delta$ and adding the result to the actual number of Newton steps already made (see (54)), we see that we need in total at most

$$
\frac{1}{\delta}\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}(\ln\vartheta+\ln(1/\epsilon)+\ln R+\ln\delta+C_1)+\ \frac{2k}{\delta}+\frac{\ln(1/\delta)}{\delta\ln 2}+\frac{\lambda_3+\sqrt{\vartheta}}{\lambda_3-\lambda_3'}(\ln K-\ln R+C_2)
$$

(C_1, C_2 suitable constants depending only on $\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3', \hat{\lambda}_3$) Newton iterations to solve *all* problems considered to ε-optimality. Note that, due to $\vartheta\geq 1$, (15), and (17),

$(\lambda_3 + \sqrt{\vartheta})/(\lambda_3 - \lambda_3') > 1/\ln 2$ holds. Moreover, if the asymptotically optimal (for large ϑ) values for the λ-parameters are used, the $(\ln \delta)/\delta$-term from above has approximatively the form

$$(7.102(\sqrt{\vartheta} + 0.229) - 1.442)\frac{\ln \delta}{\delta}$$

and is therefore negative as long as $\delta < 1$. As a consequence, we can drop the term $\ln(1/\delta)/\delta \ln 2$ in the estimate above completely by re-adjusting the constant C_1.

We summarize our findings in the next theorem.

Theorem 7.2 (Complexity Estimate) *Let $G \subseteq \mathbb{R}^m$ be convex and compact with nonempty interior and let F be a self-concordant barrier for G with self-concordancy parameter ϑ. Moreover, let $\|x\|_2 \leq R$ hold for all $x \in G$ and a number $R > 0$. Let $c_1, c_2 \in \mathbb{R}^m$ with $c \neq 0 \neq c_2$ and let there be given an accuracy measure $\varepsilon > 0$. With x computed in Step 3 of Algorithm 7.1, define*

$$K = \max\{\|c_1\|_{x,F}^\circ, \|c_2\|_{x,F}^\circ\}$$

and with the λ-parameters chosen as in Step 1 suppose that

$$\delta \geq \varepsilon \frac{(\hat{\lambda}_3^2 - \lambda_3^2 + 1)^{1/2} - 1}{18R(2\vartheta + \zeta(\lambda_3')/2)}$$

as well as $0 < \varepsilon K < 1$ holds. Then, the total number of Newton-steps in Steps 6–7 of Algorithm 4.1 needed to compute a δ-covering $(\mu_i)_i$ of $[0,1]$, $\mu_i \in [0,1]$, and points x_{μ_i}, each of them an ε-solution to the problem

$$\begin{array}{ll} \textit{minimize} & \langle\, \mu_i c_1 + (1-\mu_i)c_2, x \,\rangle \\ \textit{subject to} & x \in G, \end{array}$$

is

$$O(1)\left(\frac{\sqrt{\vartheta}}{\delta}(\ln \vartheta + \ln(1/\varepsilon) + \ln R + \ln \delta) + \sqrt{\vartheta}(\ln K - \ln R)\right),$$

where the O-constant depends on $\lambda_1, \lambda_1', \lambda_2, \lambda_3, \lambda_3', \hat{\lambda}_3$.

Note that the term "$+\ln \delta$" in the complexity estimate above is *not* a typo! Indeed, the sum of logarithms in this estimate *decreases* if δ decreases. This reflects the nature of Algorithm 7.1, in which information from one scalar problem can be more easily put to use while solving other problems when the perturbation size δ decreases. Of course, the overall factor $1/\delta$ increases with decreasing δ, and this is as it should be: the output size of Algorithm 7.1 is $O(1/\delta)$ points.

In contrast to this, the standard approach consists of solving each problem by starting from the same starting point (the approximation to the analytic center) without reusing any

information. This strategy needs

$$O(1)\frac{1}{\delta}\sqrt{\vartheta}(\ln\vartheta + \ln(1/\epsilon) + \ln K)$$

Newton steps. Clearly, for δ small enough, the proposed method has a higher efficiency.

Note that in each step of the algorithm, each of the points x_μ is an $\tilde{\varepsilon}$-solution to the problem parameterized by μ, where $\tilde{\varepsilon} > 0$ is an accuracy measure implicitly defined by the number of Newton steps already made. More precisely, if i is the number of Newton steps made, then $\tilde{\varepsilon} = O(\vartheta/\exp(i/\sqrt{\vartheta}))$ is the accuracy achieved.

We have chosen here a branching into two different branches. Due to this, the δ-covering of the parameter set $[\,0,1\,]$ is refined in each step to a $\delta/2$-covering, multiplying the number of parameters to be considered by a factor of 2. Clearly, other branching schemes can also be used, in which a δ-covering is refined to a δ/ℓ-covering, $\ell > 2$.

8 More than Two Criteria

Consider the case in which $n > 2$ criteria are given. It is rather easy to adapt the algorithm described in Section 7.2 to this general case. However, we need to discretize the $(n-1)$-dimensional set Z (see (3)), which is computationally a nontrivial task. Moreover, it has already been mentioned in the introduction that we will not get rid of the factor $O(1/\delta^{n-1})$ in the computational complexity of any algorithm considered. Therefore, instead of pursuing further theoretical run-time estimates, some discussion about proper implementation strategies for the scheme presented are in order. The main question that we have to answer is: how do we discretize the set Z?

8.1 Voronoi-Tesselations

Let μ denote the coordinates of a parameterization of Z. Usually, $\mu \in \mathbb{R}^{n-1}$. Let $z(\mu) \in Z$ be the element of Z parameterized by μ and define

$$\begin{aligned} f_\mu(x) &:= \langle\, z(\mu), x \,\rangle, \\ F_t^\mu(x) &:= t f_\mu(x) + F(x). \end{aligned}$$

We can start with one parameter μ and define $\delta(0) := \sup_{z\in Z} \|z - z(\mu)\|_2$. Again, the set $S = S(j)$ denotes the set of parameters μ which are considered as problem parameters in the path-following scheme. This set induces a Voronoi-tesselation $(V_\mu)_{\mu\in S}$ of Z. Now, the variable $x_\mu \in G$ ($\mu \in S$) is supposed to represent the solution to all problems in the Voronoi cell V_μ. Therefore, the stopping criterion of the `WHILE`-loop in Step 6 (a) ii has to be adapted to

6 (a) ii. WHILE $\lambda(F^{\nu}_{t_\mu}, x_\mu) < \hat{\lambda}_3$ for all $\nu \in V_\mu$ AND NOT stopping criterion for problem with parameter μ fulfilled DO

Moreover, the set S has to be refined in Step 6 (d) according to some strategy, i. e. points have to be added or at least one point has to be replaced by two or more points. If a standard Cartesian grid is used, we arrive at similar complexity estimates as their counterparts which have been derived in Section 7.3.

However, this algorithmic concept is rather problematic, since it is in no way clear how the abovementioned stopping criterion can be implemented in an efficient way, given that we have to compute (or rather update) Voronoi-tesselations in $\mathbb{R}^{n-1}$. Therefore, different strategies, not necessarily based on Voronoi diagrams might be in order.

8.2 Decomposition-based Strategies

A more feasible strategy is to decouple the point $z(\mu)$ from the subset of the parameter set which it represents. To this end, let the set $S = S(j)$ consist of pairs $(z(\mu), S_\mu)$ with $z(\mu) \in S_\mu \subseteq Z$ such that the finite family $(S_\mu)_\mu$ defines a decomposition of Z. Clearly, Voronoi tesselations are a special case, but the additional degrees of freedom imposed by the possibility of choosing $z(\mu) \in S_\mu$ arbitrarily as well as choosing the geometry of the decomposition has some distinct advantages. For example, we might choose an arbitrary refinement of the sets S_μ when replacing $S(j)$ by $S(j+1)$. The most simple example would be to start with $S(0) = \{((1/n)e, Z)\}$ and to use simplicial refinements only. Techniques for refining simplicial decompositions are well developed, see, e. g. [19, 18, 27]. Whenever a set S_μ which is part of a decomposition of Z is polyhedral, Step 6 (a) ii can be replaced by

6 (a) ii. WHILE $\lambda(F^{\nu}_{t_\mu}, x_\mu) < \hat{\lambda}_3$ for all vertices ν of S_μ AND NOT stopping criterion for problem with parameter μ fulfilled DO ,

a formulation which has obvious advantages for simplices S_μ.

8.3 Adaptive Grids

The strategies discussed in Section 8.1 and 8.2 both have their merits when we try to discretize the parameter set Z up to a certain discretization size. (Which can be measured, e. g., by $\max_{S_\mu} \mathrm{diam}(S_\mu)$, the maximal diameter of the sets considered.) However, we have to keep in mind that the ultimate goal is not a discretization of Z, but of $E(f(G))$. The only interconnection between Z and $E(f(G))$ is the function φ resp. ψ, see (6) resp. (7). Therefore, an estimate of the quality of a discrete approximation of $E(f(G))$ should not be based on the quality of a discretization of Z (which is a trivial problem). Instead, a bad

discretization of $P(f(G))$ (see (4) and (5)) should be refined locally wherever the quality is worst, thereby inducing a further refinement of the given discretization of Z.

One possibility to achieve this is to note that usually $E(f(G)) \subseteq \mathrm{bd}(f(G))$ holds [12, Satz 2.9], i. e. the set of efficient points has at most the same dimension as Z, and that therefore a simplicial decomposition of $E(f(G))$ is in order. For example, let $(\hat{S}_i)_i$ be a triangulation of a surface in $\mathbb{R}^n$ such that all images of vertices $f(\varphi(z_j))$ of this triangulation lie in $E(f(G))$. Moreover, let there be given a topologically equivalent triangulation of the preimages z_j in Z. This triangulation in Z induces a one-to-one relation between the set of $n-1$-dimensional simplices $(\hat{S}_i)_i$ and a corresponding set of $n-1$-dimensional simplices $(S_i)_i$ in Z. With this, one example of the refinement step is

> 6. (d) Let $\hat{S}_i$ be the $n-1$-dimensional simplex with largest volume and S_i the corresponding simplex in Z. Divide S_i into subsimplices $S_i^{(j)}$ ($j = 1, \ldots, k$). Denote by $z^{(\ell)}$ ($\ell = 1, \ldots, L$) the new vertices. Replace S_i by the $S_i^{(j)}$, add the $z^{(\ell)}$ to the set of all z_j and keep track of the refined triangulation.

Denoting by vol_{n-1} the $n-1$-dimensional Lebesgue measure of a set, we see that $\max_i \mathrm{vol}_{n-1}(\hat{S}_i)$ is a global accuracy measure for the approximation to $E(f(G)$ just constructed, and this measure is monotonically decreasing as the algorithm proceeds.

A simplicial approximation of $E(f(G))$ has advantages with respect to a visualization of the approximation. Two possibilities are immediately at hand. First, each discrete point computed lying in $E(f(G))$ can be represented by a point in the plane. These planar points can then be connected according to the edge structure of the decomposition. The second method computes the dual representation of the graph computed by the first method. There, every simplex of the approximation of $E(f(G))$ is represented by a point in the plane, and these points are then connected according to the neighborhood structure of the simplices. Both graph representations are minimal in the sense that each vertex of the computed graph is connected to exactly n other vertices, and both representations map a neighborhood structure in the set $E(f(G))$ onto edges of a graph.

Computational experiences with such a scheme will be presented in a forthcoming paper [15].

References

[1] Harold P. Benson and Serpil Sayın. Towards Finding Global Representations of the Efficient Set in Multiple Objective Mathematical Programming. *Naval Research Logistics*, 44:47–67, 1997.

[2] A. B. Berkelaar. Complementarity and Interior Point Methodology. Master's thesis, Department of Statistics, Probability and Operations Research, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, October 31 1995.

[3] Arjan B. Berkelaar, Benjamin Jansen, Kees Roos, and Tamás Terlaky. An Interior-Point Approach to Parametric Convex Quadratic Programming. Working paper, Erasmus University Rotterdam, Erasmus University Rotterdam, Rotterdam, The Netherlands, February 29 1996. Revised version from December 23, 1997.

[4] Emilio Carrizosa and Jörg Fliege. Generalized goal programming: Polynomial methods and applications. *Mathematical Programming*, 93:281–303, 2002.

[5] Indraneel Das and J. E. Dennis. Normal-Boundary Intersection: A New Method for Generating Pareto Optimal Points in Nonlinear Multicriteria Optimization Problems. *SIAM Journal on Optimization*, 8(3):631–657, 1998.

[6] Indraneel Das and John Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14(1):63–69, 1997.

[7] Jörg Fliege. Approximation Techniques for the Set of Efficient Points. Professorial dissertation, Fachbereich Mathematik, Universität Dortmund, 44221 Dortmund, Germany, January 31 2001.

[8] Jörg Fliege. Gap-Free Computation of Pareto-Points by Quadratic Scalarizations. *Mathematical Methods of Operations Research*, 59, 2004.

[9] Jörg Fliege and Andree Heseler. Constructing approximations to the efficient set of convex quadratic multiobjective problems. Ergebnisberichte Angewandte Mathematik 211, Universität Dortmund, Fachbereich Mathematik, Universität Dortmund, 44221 Dortmund, Germany, January 2002. Submitted.

[10] Arthur M. Geoffrion. Proper Efficiency and the Theory of Vector Maximization. *Journal of Optimization Theory and Applications*, 22:618–630, 1968.

[11] Jacek Gondzio and Andreas Grothey. Reoptimization with the primal-dual interior point method. *SIAM Journal on Optimization*, 13:561–584, 2003.

[12] A[lfred] Göpfert and R[einhard] Nehse. *Vektoroptimierung — Theorie, Verfahren und Anwendungen*, volume 74 of *Mathematisch-Naturwissenschaftliche Bibliothek*. BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1990.

[13] Alireza Ghaffari Hadigheh, Oleksandr Romanko, and Tamás Terlaky. Sensitivity Analyis in Convex Quadratic Optimization: Simultaneous Perturbation of the Objective and Right-Hand-Side Vectors. AdvOl-Report 2003/6, McMaster University, Advanced Optimization Laboratory, Hamilton, Ontario, Canada, December 2003.

[14] Y. Y. Haimes and R[alph] E. Steuer, editors. *Research and Practice in Multiple Criteria Decision Making*, volume 487, Berlin, Heidelberg, 2000. Springer Verlag. Lecture Notes in Economics and Mathematical Systems.

[15] Christoph Heermann, Jörg Fliege, and Bernd Weyers. Computational experience with a new adaptive algorithm for multicriteria optimization. Working paper, Universität Dortmund, Fachbereich Mathematik, 2004. Forthcoming.

[16] Andree Heseler. Bestimmung effizienter Portefeuilles durch Warmstart-Innere-Punkt-Methoden. Diplomarbeit, Lehrstuhl für Wissenschaftliches Rechnen, Fachbereich Mathematik, Universität Dortmund, 44221 Dortmund, Germany, 2000.

[17] Claus Hillermeier. *Nonlinear Multiobjective Optimization: a Generalized Homotopy Approach*, volume 25 of *International Series of Numerical Mathematics*. Birkäuser Verlag, Basel — Boston — Berlin, 2001.

[18] R. Horst and P. Pardalos, editors. *Handbook of Global Optimization*. Kluwer Academic Publishers, Dordrecht, 1995.

[19] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. Springer, Berlin, second edition, 1993.

[20] Johannes Jahn. *Mathematical Vector Optimization in Partially Ordered Linear Spaces*. Verlag Peter Lang, Frankfurt, 1986.

[21] M. Köksalan and S. Zionts, editors. *Multiple Criteria Decision Making in the New Millennium*, Berlin, Heidelberg, 2001. Springer Verlag. Proceedings of the Fifteenth International Conference on Multiple Criteria Decision Making (MCDM), Ankara, Turkey, 2000.

[22] Yurii Nesterov and Arkadii Nemirowskii. *Interior-Point Polynomial Algorithms in Convex Programming*, volume 13 of *SIAM Studies in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, 1994.

[23] James Renegar. Linear programming, complexity theory and elementary functional analysis. *Mathematical Programming*, 70A:279–351, 1995.

[24] James Renegar. *A Mathematical View of Interior-Point Methods for Convex Optimizations*. MPS-SIAM Series on Optimization, 2001.

[25] James Renegar. Linear Programming, Complexity Theory and Elementary Functional Analysis. Published as [23], June 1994.

[26] Yoshikazu Sawaragi, Hirotaka Nakayama, and Tetsuzo Tanino. *Theory of Multiobjective Optimization*, volume 176 of *Mathematics in Science and Engineering*. Academic Press, Inc., Orlando, San Diego, New York, London, 1985.

[27] Wohlmuth, Barbara I. and Hoppe, Ronald H. W. A comparison of a posteriori error estimators for mixed finite element discretizations by Raviart-Thomas elements. *Mathematical Computation*, 68:1347–1378, 1999.

[28] E. Alper Yıldırım and S. J. Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12:782–810, 2002.