Mathematical Programming 52 (1991) 527-553
North-Holland

On the complexity of following the central path of linear programs by linear extrapolation II

G. Sonnevend,* J. Stoer and G. Zhao

Institut für Angewandte Mathemaik und Statistik, Universität Würzburg, Am Hubland, W-8700 Würzburg, Germany

Received 26 January 1990
Revised manuscript received 31 October 1990

A class of algorithms is proposed for solving linear programming problems (with m inequality constraints) by following the central path using linear extrapolation with a special adaptive choice of steplengths. The latter is based on explicit results concerning the convergence behaviour of Newton's method to compute points on the central path $x(r)$, $r>0$, and this allows to estimate the complexity, i.e. the total number $N=N(R,\delta)$ of steps needed to go from an initial point $x(R)$ to a final point $x(\delta)$, $R>\delta>0$, by an integral of the local "weighted curvature" of the (primal–dual) path. Here, the central curve is parametrized with the logarithmic penalty parameter $r\downarrow 0$. It is shown that for large classes of problems the complexity integral, i.e. the number of steps N, is not greater than const $m^{\alpha}\log(R/\delta)$, where $\alpha<\frac{1}{2}$ e.g. $\alpha=\frac{1}{4}$ or $\alpha=\frac{3}{8}$ (note that $\alpha=\frac{1}{2}$ gives the complexity of zero order methods). We also provide a lower bound for the complexity showing that for some problems the above estimation can hold only for $\alpha\geqslant\frac{1}{3}$.

As a byproduct, many analytical and structural properties of the primal–dual central path are obtained: there are, for instance, close relations between the weighted curvature and the logarithmic derivatives of the slack variables; the dependence of these quantities on the parameter r is described. Also, related results hold for a family of weighted trajectories, into which the central path can be embedded.

1. Introduction

In this paper we continue our study (21) of the complexity of suitably constructed linear (i.e. first order) extrapolation algorithms for the solution of linear programs based on following the "central path" associated to these programs. We shall consider linear programs (regarded here as the "primal" ones)

$$(\mathscr{P})\qquad \min\{c^{\mathrm{T}}x\,|\,A^{\mathrm{T}}x\leqslant b,\ x\in\mathbb{R}^n\}=:\lambda^*,$$

where $c\in\mathbb{R}^n$, $b=(b_1,\ldots,b_m)^{\mathrm{T}}\in\mathbb{R}^m$ and $A:=(a_1,\ldots,a_m)$ is an $n\times m$-matrix with columns a_i. Then the dual problem to $(\mathscr{P})$ is

$$(\mathscr{D})\qquad \min\{b^{\mathrm{T}}\mu\,|\,c+A\mu=0,\ \mu\geqslant 0\}=-\lambda^*.$$

By

$$\mathscr{P}:=\{x\,|\,A^{\mathrm{T}}x\leqslant b\}\tag{1.1}$$

we denote the feasible set of $(\mathscr{P})$, by $P^{\circ}:=\{x\,|\,A^{\mathrm{T}}x<b\}$ its interior and by $s=s(x):=b-A^{\mathrm{T}}x$ the vector of slack-variables belonging to x. Further, for vectors u, $v\in\mathbb{R}^m$

[*] On leave from the Institute of Mathematics, Eötvös University Budapest, H-1080 Budapest, Hungary.

we use the notation $u \circ v$ for the vector with components $u_i v_i$, $i = 1, 2, \ldots, m$, and U or also $[u]$ for the diagonal matrix with elements u_i. Similarly, we write $u^2 := u \circ u$, and u^{-1} for the vector with $u^{-1} \circ u = e := (1, 1, \ldots, 1)^{\mathrm{T}}$.

We assume, that $c \neq 0$, rank $A = n$ and P is such that for some finite value of $\lambda > \lambda^*$,

$$P_\lambda := P \cap \{x \mid c^{\mathrm{T}} x \leq \lambda\} \tag{1.2}$$

has a bounded interior $P_\lambda^\circ \neq \emptyset$. Then the mapping $s(x) \leftrightarrow x$ is one-to-one.

By $\mathscr{K}_0$ we denote the class of linear programs $(\mathscr{P})$ satisfying these conditions. We shall consider several specific subclasses $\mathscr{K}$ of $\mathscr{K}_0$ below, showing that the worst case behaviour of the proposed algorithm over these classes can be estimated based on the specific properties of $\mathscr{K}$. We say that a particular constant is a universal constant for a given class $\mathscr{K}$ of problems if it depends only on the class $\mathscr{K}$, but not on the problems in this class. We are mainly interested in the worst case, i.e. guaranteed behaviour and the "asymptotical" case where $m \to \infty$, $n \to \infty$ (while $m \leq \text{const}\, n$).

The central path for the problem $(\mathscr{P})$ is defined as the solution $x = x(r) \in P^\circ$, $r > 0$, of the system of equations

$$\varepsilon(r, x) := \frac{c}{r} + \sum_{i=1}^{m} \frac{a_i}{b_i - a_i^{\mathrm{T}} x} = 0, \quad r > 0. \tag{1.3}$$

For each $r > 0$, $x(r)$ is the unique maximizer in P_λ, where $\lambda := r + c^{\mathrm{T}} x(r)$, of the function

$$\Xi_\lambda(x) := \left((\lambda - c^{\mathrm{T}} x) \prod_{i=1}^{m} (b_i - a_i^{\mathrm{T}} x) \right)^{1/(m+1)}, \tag{1.4}$$

which is concave on P_λ. $x(r)$ is called the "center" of (the "inequality system" specifying) P_λ. See [18, 20, 21] for additional motivations of using these centers in solving more general convex and semiinfinite optimization problems.

We now introduce the dual variables $\mu = \mu(r) > 0$ with $\mu_i(r) = r/(b_i - a_i^{\mathrm{T}} x(r))$, $i = 1, \ldots, m$. The pair $(x(r), \mu(r))$ is the unique positive solution of the *primal-dual* central path equations

$$c + A\mu = 0, \tag{1.5}$$

$$\mu \circ (b - A^{\mathrm{T}} x) = re. \tag{1.6}$$

Here and below we consider — when we speak about the primal-dual central path $(x(r), \mu(r))$ — only those solutions of (1.5)-(1.6) for which both factors in (1.6) are positive.

In the basic algorithm proposed below we use, in an alternating fashion, tangential moves with an adaptively and *explicitly chosen* (tangential) *stepsize* as predictors, and recentering steps to reach (a nearest point of) the central path by means of Newton's method used as a corrector. Based on an explicit knowledge of the convergence behaviour of Newton's method applied to equations (1.3) or (1.5)-(1.6), we are able to choose the tangential stepsize in the predictor phase in a simple constructive way, namely as the largest possible step so as to get back (sufficiently close) to the central path by universally bounded number of Newton steps in the ensuing corrector phase. The precise description of the proposed algorithms is given below.

In order to motivate the choice of algorithms of the above type — for other "interior point" algorithms, see [1, 11, 12, 13, 24] — let us make the following remarks. First of all the proposed algorithms have a sound theoretical basis, their worst and average case behaviour can be analyzed using classical analysis even for cases, where the optimal (primal or dual) solutions are degenerate. As far as we know no such comparable theory is known for the affine scaling type methods in common use, which proceed in a profitable direction almost as far as possible without leaving the interior of the feasible domain and therefore allow the iterates to come very close to the boundary. As path following methods stay away from the boundary they seem to be more stable, performing with perhaps less average speed but with a high reliability on all problems. Our computational experience with an implementation of the algorithm analysed here (but with an option allowing moves to the boundary and switching off recentering) indicates (see [14]) that such algorithms are at least comparable with the standard methods, also as far as the efficiency is concerned. The behaviour of non-path-following methods seems to be less uniform and predictable and, even if they are relatively fast in general, they can be quite slow in some cases. Of course the idea of "occasional" recentering has been around from the beginning after centers and affine scaling have been introduced. We believe that the real advantage of a continual recentering will be even more significant when we turn to higher order extrapolation methods, see below. The motivation to study first order path following methods in more detail was the surprisingly rich structure of the curvature functions of the central path.

Our algorithms also allow "large steps", namely when a natural and explicitly computable quantity, the weighted curvature of the extrapolated path, at the given point is small. A return to the central path has the advantage that, even if we work in either the primal or dual space, on the central path *both* primal and dual variables become known: thus they can be used for computing a primal, a dual or a primal-dual steplength for the next step; there is also the possibility to use "weighted" centers and corresponding trajectories with adaptively chosen weights maintaining dual feasible solutions without exact recentering, see [25] and below.

We measure the complexity of our algorithms by means of the number $N = N(R, \delta)$, $R > \delta > 0$, of extrapolation steps needed by the method to reach from a given $R =: r_0$ the smaller value δ,

$$R = r_0 > r_1 > r_2 > \cdots > \delta \geq r_N.$$

This is justified, as in each step $r_k \to r_{k+1}$ one has to solve only a finite number of sets of linear equations for finding the tangential direction $\dot{x}(r_k)$ and the ensuing Newton steps used for recentering, and this number is bounded by a universal constant independent of the problem $\mathcal{P} \in \mathcal{K}_0$.

We do not go here into the details of implementation, e.g. the problem of the optimal choice of constants regulating the step size in the extrapolation phase, or that of the required accuracy of reaching the central path in the corrector phase, or the way the linear equations in each step are solved (cg-method using sparse Cholesky factorizations as preconditioners), even though all this is quite important for the fine tuning of the algorithms and their numerical efficiency (see [14]).

We would like to stress however that we are not only interested in algorithms with a good theoretical complexity but also with a good practical performance. At present we have a fairly complete theory of analysing the complexity of a primal–dual path following method (Algorithm 1 below). We shall present two further algorithms, in which at each step either a primal or a dual extrapolation step is made: while in a worst case sense for the class $\mathcal{K}_0$ these algorithms are not essentially better than Algorithm 1 we think, that in "most" cases they will provide an acceleration. On the test set of 25 Netlib programs of increasing size including several ones with $m+n\geq 2500$ already a rather preliminary tuning of some parameters yielded an algorithm, which was in the average on large problems comparable in speed with the last version of the wellknown simplex package MINOS 5.3 in the same environment (see [13, 14]). The number of basic extrapolation steps was mostly less than those reported for implementations of other interior point methods (see [1, 12, 13]). The method needed about 2.5 Newton steps per basic iteration and about 5 cg-steps per Newton step, but (so far) also about 1.5 frech refactorizations of a large positive definite matrix per basic iteration, a number, which can be presumably be reduced at the expense of a higher number of cg-steps. We mention these preliminary results in order to correct the often expressed opinion that algorithms which try to follow the central path have only theoretical interest.

Even though the proposed linear extrapolation algorithms perform surprisingly well when compared to higher order polynomial extrapolation (see [9]), we believe, that nonpolynomial extrapolation methods of higher order will provide a further acceleration. A more accurate higher order extrapolation could be constructed, based on the following idea: use as an extrapolating curve at a point $x(r_k)$ the more easily computable central path of a smaller linear program obtained by restricting the original linear program to the subspace generated by the first k derivatives of the path $x(\cdot)$ at r_k. In the paper [20] this idea is shown to work nicely for the problem of minimizing a quadratic function over an ellipsoid.

An important tool for estimating the complexity of the algorithms considered here is an *integral* $I(R, \delta, \mathcal{P})$), *of a* (*weighted*) *curvature* along the section $[\delta, R]$ of the central path, by which we can rigorously bound the number $N(R, \delta, \mathcal{P})$ of iterations needed to reach one endpoint of this section from the other when solving $\mathcal{P}$.

The basic arguments for the derivation of such estimates of the form

$$N(R, \delta, \mathcal{P}) \leq I(R, \delta, \mathcal{P}) \leq \text{const}\ m^{\alpha(\mathcal{K})} \log(R/\delta) \quad \text{for all } \mathcal{P} \in \mathcal{K} \qquad (1.7)$$

for various classes $\mathcal{K}$ of programs, with an $\alpha(\mathcal{K}) < \frac{1}{2}$, have been derived in [21].

In order to explain why the algorithms under study often perform even much better than (1.7) would suggest we mention that often (especially in the asymptotical case $m/n \to \infty$) better estimates can be obtained for many linear semiinfinite problems and for smooth convex nonlinear problems, e.g. for generalised quadratic problems, where all constraints and the objective function are convex quadratic. The estimates developed here essentially remain valid, but sometimes, e.g. with semiinfinite linear or quadratic programs, m can be replaced by n in (1.7)) (see [16, 18]). Note also that a convex quadratic constraint can be replaced (with good accuracy) by a large number of linear constraints (see [18], where the choice of the "proper" weights for the constraints is studied for semiinfinite programs), so that the corresponding central paths are approximately the same.

We derive new relationships and estimates, which add to the understanding of the rich structure of central paths, and which we exploit for the construction of new path-following algorithms. Some of the results were already stated in [21]. The exposition in this paper, however, is self contained.

2. Description of the algorithms

We begin with a short summary of some important global convergence properties of Newton's method for computing a central point close to a strictly (primally and/or dually) feasible point. In order to describe Newton's method, i.e. the corrector steps of the proposed algorithm, we invoke the following observation of Roos and Vial [17] (earlier convergence domains of this type have been proposed and used in a complexity analysis for zero order path following methods, among others in [11]). Consider the least squares problem, where $0 < \mu \in \mathbb{R}^m$ satisfies (1.5) and μ and $r > 0$ are fixed,

$$\inf_z \left\| \frac{\mu \circ (b - A^{\mathrm{T}} z)}{r} - e \right\| =: \delta_{\mathrm{d}}(\mu, r). \tag{2.1}$$

Let its optimal solution be denoted by z^*, and let

$$\mu' := 2\mu - \frac{1}{r} s(z^*) \circ \mu^2. \tag{2.2}$$

Then also μ' satisfies (1.5). If in addition $\delta_{\mathrm{d}}(\mu, r) < 1$, then z^* is a strictly feasible solution of $\mathscr{P}$, and by [17], $\mu' > 0$ is a strictly feasible solution of $(\mathscr{D})$ satisfying

$$\delta_{\mathrm{d}}(\mu', r) \leq \delta_{\mathrm{d}}(\mu, r)^2. \tag{2.3}$$

In fact, μ' is the result of one Newton step (started from μ) for the computation of the dual center $\mu(r) = k - L^{\mathrm{T}}\eta(r)$,

$$\eta(r) := \arg\min_{\eta} \left\{ -\frac{b^{\mathrm{T}} L^{\mathrm{T}} \eta}{r} - \sum_i \log(k_i - l_i^{\mathrm{T}} \eta) \;\middle|\; k - L^{\mathrm{T}}\eta > 0 \right\} \tag{2.4}$$

where $\mu = k - L^{\mathrm{T}}\eta$, $\eta \in \mathbb{R}^{m-n}$ is a parametrization of all μ from the dual feasible set $\{\mu \mid A\mu + c = 0,\ \mu \geqslant 0\}$ by means of an $(m-n) \times m$-matrix $L = (l_1, \ldots, l_m)$ of rank $m-n$ with $AL^{\mathrm{T}} = 0$. Note also, that with $l := -Lb$,

$$\{s = b - A^{\mathrm{T}}x \mid x \in \mathbb{R}^n\} = \{s \mid Ls + l = 0\}.$$

Therefore the roles of $(\mathscr{P})$ and $(\mathscr{D})$ can be interchanged in the above arguments of [17] by just replacing A, b, c by L, k and l, respectively. Thus for any $x \in P^{\circ}$, that is $s = s(x) = b - A^{\mathrm{T}}x > 0$ solves $Ls + l = 0$, the Newton iteration $x \to x'$ for the solution of the system (1.3) can be obtained from the solution μ^* of the least squares problem

$$\delta_{\mathrm{p}}(x, r) := \inf_{\mu} \left\{ \left\| \frac{\mu \circ (b - A^{\mathrm{T}}x)}{r} - e \right\|^2 \middle|\ A\mu + c = 0 \right\} \tag{2.5}$$

by the formula

$$s' = 2s - \frac{s^2 \circ \mu^*}{r}, \tag{2.6}$$

in the following way: As s' again solves $Ls' + l = 0$, it has the form $s' = s(x') = b - A^{\mathrm{T}}x'$ for a vector $x' \in \mathbb{R}^n$ which is the unique solution of the equations

$$b_i - a_i^{\mathrm{T}}x' = 2(b_i - a_i^{\mathrm{T}}x) - (b_i - a_i^{\mathrm{T}}x)^2 \frac{\mu_i^*}{r}, \quad i = 1, 2, \ldots, m.$$

As before, $\delta_{\mathrm{p}}(x, r) < 1$ implies the strict feasibility of x' for $(\mathscr{P})$, $s' = s(x') > 0$, and $\delta_{\mathrm{p}}(x', r) \leqslant \delta_{\mathrm{p}}(x, r)^2$, which again describes a region of global quandratic convergence of Newton's method. Of course we know that

$$x' = x - H_{\mathrm{p}}^{-1}(x)\left(\frac{c}{r} + \sum \frac{a_i}{b_i - a_i^{\mathrm{T}}x}\right), \tag{2.7}$$

where

$$H_{\mathrm{p}}(x) := \sum_{i=1}^{m} \frac{a_i a_i^{\mathrm{T}}}{(b_i - a_i^{\mathrm{T}}x)^2}.$$

Similarly we define

$$H_{\mathrm{d}}(\eta) := \sum_j \frac{l_j l_j^{\mathrm{T}}}{(k_j - l_j^{\mathrm{T}}\eta)^2} \quad \text{and} \quad H(x, \eta) := H_{\mathrm{p}} \oplus H_{\mathrm{d}}. \tag{2.8}$$

Another domain D_γ of superlinear convergence for Newton's method — say for the primal problem — can be obtained as follows:

$$D_\gamma := \{z \mid \|z - x(r)\|_{H_{\mathrm{p}}(r)} \leqslant \gamma\}. \tag{2.9}$$

Here γ is a universal constant, and as usual for any positive definite matrix H, $\|u\|_H$ denotes the norm $\|u\|_H := (u^{\mathrm{T}}Hu)^{1/2}$ (see [9] for a detailed analysis of the superlinear, resp. quadratic convergence of Newton's method in the domain (2.9)).

Algorithm 1. Let $0<\alpha_0<1$ be a fixed constant and suppose that we have already computed the values of $x(r_k)$ and $\mu(r_k)$ together with the derivatives $\dot{x}(r_k)$ and $\dot{\mu}(r_k)$. Define the vector-valued linear extrapolants

$$x_k(r):=x(r_k)+\dot{x}(r_k)(r-r_k)\in\mathbb{R}^n,$$

$$\mu_k(r):=\mu(r_k)+\dot{\mu}(r_k)(r-r_k)\in\mathbb{R}^m,$$

and compute r_{k+1} as the largest solution r (with $r<r_k$) of the equation

$$\left\|\frac{\mu_k(r)\circ(b-A^{\mathrm{T}}x_k(r))}{r}-e\right\|=\alpha_0. \tag{2.10}$$

Perform a number of Newton iterations with the starting vector $\mu_k(r)$, or $x_k(r)$, as described in (2.2) or (2.6), (2.7), till we get $\delta_{\mathrm{d}}(\mu, r_{k+1})$ or $\delta_{\mathrm{p}}(x, r_{k+1})$ small, say to machine accuracy.

Lemma 1. *Condition* (2.10) *is equivalent to the following*:

$$f(r_k)(r_k-r_{k+1})\sqrt{\frac{r_k}{r_{k+1}}}=\sqrt{\alpha_0}, \tag{2.11}$$

where

$$f(r):=\frac{1}{r}\,\|\varphi(r)\|^{1/2},\quad \varphi(r):=\sigma(r)\circ\tau(r)=\sigma(r)-\sigma^2(r),$$

$$\sigma(r):=r\frac{\mathrm{d}}{\mathrm{d}r}\log s(r),\qquad \tau(r):=r\frac{\mathrm{d}}{\mathrm{d}r}\log\mu(r),$$

$$\sigma(r)=M(r)e,\qquad \tau(r)=(I-M(r))e=e-\sigma(r),$$

$$M(r):=\tilde{A}^{\mathrm{T}}(\tilde{A}\tilde{A}^{\mathrm{T}})^{-1}\tilde{A},\quad \tilde{A}:=AS^{-1}(r),$$

and $S(r)$ is the diagonal matrix associated to $s(r):=s(x(r))=b-A^{\mathrm{T}}x(r)$. Moreover, in terms of the vectors and matrices,

$$z(r):=(s(r),\mu(r)),\qquad H(r):=H(x(r),\eta(r)),\qquad \xi(r):=(x(r),\eta(r)),$$

the function $f(r)$ is also given by

$$f(r)=\frac{1}{\sqrt{2}}\,\|\ddot{\xi}(r)\|^{1/2}_{H(r)}=\frac{1}{\sqrt{2}}\,\|z^{-1}(r)\circ\ddot{z}(r)\|_2^{1/2}.$$

Proof. By definition of $x(r)$, $s(r)$ and $\mu(r)$, see (1.5) and (1.6), we have identically in r,

$$c+A\mu(r)=0,$$

$$A^{\mathrm{T}}x(r)+s(r)=b,$$

$$\mu(r)\circ s(r)=re.$$

Differentiation with respect r yields

$$\begin{aligned}
&A\dot{\mu}(r)=0,\\
&A^{\mathrm{T}}\dot{x}(r)+\dot{s}(r)=0, \qquad\qquad (2.12)\\
&\dot{\mu}(r)\circ s(r)+\mu(r)\circ\dot{s}(r)=e.
\end{aligned}$$

In terms of the logarithmic derivatives σ and τ this gives

$$\begin{aligned}
&\tilde{A}\tau(r)=0,\\
&r\tilde{A}^{\mathrm{T}}\dot{x}(r)+\sigma(r)=0,\\
&\tau(r)+\sigma(r)=e,
\end{aligned}$$

which shows $\tau(r)\in\mathcal{N}(\tilde{A})$, $\sigma(r)\in\mathcal{R}(\tilde{A}^{\mathrm{T}})$, $\tau(r)\perp\sigma(r)$, so that

$$\sigma(r)=M(r)e, \qquad \tau(r)=(I-M(r))e,$$

as $M(r)$ is the orthogonal projection on $\mathcal{R}(\tilde{A}^{\mathrm{T}})$. A further differentiation of (2.12) gives

$$\begin{aligned}
&\ddot{\mu}(r)\circ s(r)+\mu(r)\circ\ddot{s}(r)=-2\dot{\mu}(r)\circ\dot{s}(r),\\
&\mu^{-1}\circ\ddot{\mu}(r)\in\mathcal{N}(\tilde{A}), \qquad s^{-1}\circ\ddot{s}(r)\in\mathcal{R}(\tilde{A}^{\mathrm{T}}),\\
&z^{-1}(r)\circ\ddot{z}(r)=-\frac{2}{r^2}\sigma(r)\circ\tau(r),
\end{aligned}$$

showing

$$\frac{1}{r}\|\sigma(r)\circ\tau(r)\|_2^{1/2}=\frac{1}{\sqrt{2}}\|z^{-1}(r)\circ\ddot{z}(r)\|_2^{1/2}=\frac{1}{\sqrt{2}}\|\ddot{\xi}(r)\|_{H(r)}^{1/2}.$$

Finally by (2.12), the definitions of $x_k(r)$, $\mu_k(r)$ and $s_k(r):=s(x_k(r))=b-A^{\mathrm{T}}x_k(r)$ we have

$$\frac{\mu_k(r)\circ s_k(r)}{r}-e=\frac{(r-r_k)^2}{r}\dot{\mu}(r_k)\circ\dot{s}(r_k)=\frac{(r-r_k)^2}{rr_k}\sigma(\rho_k)\circ\tau(r_k).$$

This leads to the formula (2.11) for the solution r ($=: r_{k+1}$) of (2.10), which completes the proof of the lemma. □

The algorithms, quantities, and formulas given above can be generalized to "weighted" central paths $(x(r,\tilde{e}),\mu(r,\tilde{e}))$, obtained when we replace the vector $e=(1,\dots,1)^{\mathrm{T}}$ in (1.5), (1.6) by an arbitrary positive vector $\tilde{e}\in\mathbb{R}^m$, see [17], [25]. In particular, for any strictly primal feasible $\tilde{x}_k$ and any strictly dual feasible $\tilde{\mu}_k$ the weighted central path $(x(r,\tilde{e}_k),\mu(r,\tilde{e}_k))$ with $\tilde{e}_k:=(1/r_k)\tilde{\mu}_k\circ s(\tilde{x}_k)$ passes through $\tilde{x}_k:=x(r_k,\tilde{e}_k)$ and $\tilde{\mu}_k:=\mu(r_k,\tilde{e}_k)$. One may define linear extrapolants $x_k(r):=\tilde{x}_k+(r-r_k)\dot{x}(r_k,\tilde{e}_k)$ and simlarly $\mu_k(r)$ to the weighted trajectories, and steps $r_{k+1}:=r<r_k$ as the solution r of

$$\left\|\frac{\mu_k(r)\circ(b-A^{\mathrm{T}}x_k(r))}{r}-\tilde{e}_k\right\|=\hat{\alpha}_0,$$

where $\hat{\alpha}_0 \min\{(\tilde{e}_k)_i \mid i=1,2,\ldots,m\} = \alpha_0 < 1$. Such a solution r is computable with a straightforward generalization of Algorithm 1, especially of (2.11). Clearly $\mu_k(r_{k+1})$ and $x_k(r_{k+1})$ then are strictly feasible.

The functions $M(r)$ and especially $f(r)$ are most important for the study of the central curves and their first order approximations. Also in connection with our function $f(r)$, which gives a simple expression for the second order (i.e. main error) term for linear extrapolations, a statistical analysis of the "anticipated" behavior of several classes of interior point methods was given in [15]. There, the use of probability distributions is quite nonstandard: it is assumed that at each step the subspace represented by the projector $M(r)$ is randomly, independently and uniformly distributed. Assumptions of this kind seem hard to justify: we outline below a different method which may help to obtain rigorous average complexity results for this class of algorithms. We stress in this context however that it is the main goal of this paper to obtain *robust* algorithms excluding the possiblity of occasional extremely slow or unstable behaviour.

In much the same way as in the proof of Lemma 1 one can derive explicit expressions for the analogous of the distance (2.10), namely of the distances $\delta_{\mathrm{p,d}}$ defined by (2.1), (2.5). For example, δ_{p} is given by

$$\delta_{\mathrm{p}}(x_k(r), r) = \min\left\{ \left\| \frac{s_k(r) \circ \mu}{r} - e \right\|_2 \,\middle|\, A\mu + c = 0 \right\}$$
$$= \min\left\{ \left\| \frac{s_k(r) \circ (\mu_k(r) + \mu)}{r} - e \right\|_2 \,\middle|\, A\mu = 0 \right\}.$$

As in Lemma 1 one finds

$$\delta_{\mathrm{p}}(x_k(r), r) = \min\left\{ \left\| \frac{(r-r_k)^2}{rr_k} \sigma(r_k) \circ \tau(r_k) + \frac{s_k(r) \circ \mu}{r} \right\| \,\middle|\, A\mu = 0 \right\}.$$

Its solution is

$$\delta_{\mathrm{p}}(x_k(r), r) = \frac{(r-r_k)^2}{rr_k} \|M_{s_k(r)} \sigma(r_k) \circ \tau(r_k)\|_2,$$

where M_s is the orthogonal projection

$$M_s := A_s^{\mathrm{T}}(A_s A_s^{\mathrm{T}})^{-1} A_s, \quad A_s := AS^{-1}.$$

By taking limits $r \uparrow r_k$, we find $\lim_{r \uparrow r_k} M_{s_k(r)} = M(r_k)$ and therefore

$$\lim_{r \uparrow r_k} \frac{1}{(r-r_k)^2} \delta_{\mathrm{p}}(x_k(r), r) = \frac{1}{r_k^2} \|M(r_k)\sigma(r_k) \circ \tau(r_k)\|_2$$
$$= \tfrac{1}{2} \|\ddot{x}(r_k)\|_{H_{\mathrm{p}}(r_k)}. \tag{2.13}$$

In a similar way, one also shows

$$\lim_{r \uparrow r_k} \frac{1}{(r-r_k)^2} \delta_{\mathrm{d}}(\mu_k(r), r) = \tfrac{1}{2} \|\ddot{\eta}(r_k)\|_{H_{\mathrm{d}}(r_k)}. \tag{2.14}$$

These formulae suggest the following variant of Algorithm 1 in which either a primal or a dual move is made at each iteration.

Algorithm 2. Let $0<\alpha_0<1$ and assume that $r_k>0$, $x(r_k)$ and $\mu(r_k)$ are given and the linear extrapolants $x_k(r)$, $\mu_k(r)$ are defined as in Algorithm 1. Compute $r_{k+1}:=\min(r^{\mathrm{p}}_{k+1}, r^{\mathrm{d}}_{k+1})$, where $r^{\mathrm{p,d}}_{k+1}$ satisfy

$$\delta_{\mathrm{p}}(x_k(r^{\mathrm{p}}_{k+1}), r^{\mathrm{p}}_{k+1})=\alpha_0, \qquad \delta_{\mathrm{d}}(x_k(r^{\mathrm{d}}_{k+1}), r^{\mathrm{d}}_{k+1})=\alpha_0.$$

Perform a number of Newton corrector steps either with the starting vector $x_k(r)$ or with the vector $\mu_k(r)$, $r:=r_{k+1}$, for finding $x(r)$ and $\mu(r)$, depending on whether $\|\ddot{x}\|_{H_{\mathrm{p}}}$ is smaller than $\|\ddot{\eta}\|_{H_{\mathrm{d}}}$ or not.

We note that the asymptotical properties (2.13), (2.14) for $r=r^{\mathrm{p,d}}_{k+1}$ close to r_k show that we have for small $\alpha_0>0$ approximatively

$$\begin{aligned}&\frac{1}{\sqrt{2}}\|\ddot{x}(r_k)\|^{1/2}_{H_{\mathrm{p}}(r_k)}(r_k-r^{\mathrm{p}}_{k+1})\sqrt{\frac{r_k}{r^{\mathrm{p}}_{k+1}}}\doteq\alpha_0,\\ &\frac{1}{\sqrt{2}}\|\ddot{\eta}(r_k)\|^{1/2}_{H_{\mathrm{d}}(r_k)}(r_k-r^{\mathrm{d}}_{k+1})\sqrt{\frac{r_k}{r^{\mathrm{d}}_{k+1}}}\doteq\alpha_0,\end{aligned} \tag{2.15}$$

formulas, which could also be used to realize Algorithm 2 in practice.

We introduce a further refinement for the conceptual improvement of the above algorithms, by noting that the main goal of each step $r_k\to r_{k+1}$ should be to choose $r_{k+1}>0$ as the smallest number for which one can still find a starting approximation for the computation of $x(r_{k+1})$, $\mu(r_{k+1})$ in the domain of global convergence of Newton's method. This is mainly important for Algorithm 2, as Algorithm 1 already is optimal in the following sense: One could question, whether in Algorithm 1 there is a $\rho<r_{k+1}$ such that

$$\|\mu_k(r_{k+1})\circ(b-A^{\mathrm{T}}x_k(r_{k+1}))-\rho e\|\leqslant\alpha_0\rho$$

because then it is better to choose ρ instead of r_{k+1} in Algorithm 1. Now the minimum of the left hand side with respect to ρ is reached for

$$\rho=\frac{1}{m}\,\mu(r_{k+1})^{\mathrm{T}}s_k(r_{k+1}),$$

which turns out to be equal to $\rho=r_{k+1}$, using that $\dot{\mu}(r_k)^{\mathrm{T}}\dot{s}(r_k)=0$, which follows from Lemma 1.

In the case of primal methods, where $r_k>0$, $x(r_k)$, and $\mu(r_k)$ are given and the linear extrapolants $x_k(r)$, $s_k(r):=b-A^{\mathrm{T}}x_k(r)$, $\mu_k(r)$ are defined as in Algorithm 1, such an optimal r_{k+1} is defined as follows: For $0<r<r_k$ let

$$\psi_p(r):=\min\left\{\left\|\frac{s_k(\rho)\circ\mu}{r}-e\right\|_2 \,\middle|\, 0<\rho<r_k,\ c+A\mu=0\right\} \tag{2.16}$$

with optimal solution $\rho=\rho(r)$. Then take $r_{k+1}=r^{\mathrm{p}}_{k+1}:=r$ as the solution of $\psi_{\mathrm{p}}(r)=\alpha_0$. All in all, such parameter optimizations lead to the following refinement of Algorithm 2:

Algorithm 3. Let $0<\alpha_0<1$, $r_k>0$, $x(r_k)$ and $\mu(r_k)$ be given and define the linear extrapolants $x_k(r)$, $s_k(r)=b-A^{\mathrm{T}}x_k(r)$ and $\mu_k(r)$ as in Algorithm 1. Further let for $r<r_k$,

$$\psi_{\mathrm{p}}(r):=\min\left\{\left\|\frac{s_k(\rho)\circ\mu}{r}-e\right\|_2 \,\middle|\, 0<\rho<r_k,\ c+A\mu=0\right\},$$

$$\psi_{\mathrm{d}}(r):=\min\left\{\left\|\frac{\mu_k(\rho)\circ(b-A^{\mathrm{T}}x)}{r}-e\right\|_2 \,\middle|\, 0<\rho<r_k,\ x\in\mathbb{R}^n\right\}.$$

Take $r_{k+1}:=\min(r^{\mathrm{p}}_{k+1}, r^{\mathrm{d}}_{k+1})$, where $r^{\mathrm{p,d}}_{k+1}$ satisfy

$$\psi_{\mathrm{p}}(r^{\mathrm{p}}_{k+1})=\alpha_0, \qquad \psi_{\mathrm{d}}(r^{\mathrm{d}}_{k+1})=\alpha_0.$$

Compute $x(r_{k+1})$ and $\mu(r_{k+1})$ by a number of Newton corrector steps with the appropriate starting vectors, e.g. in the primal case with $x_k(\rho(r^{\mathrm{p}}_{k+1}))$, where $\rho(r)$ is the optimal argument ρ in the definition of $\psi_{\mathrm{p}}(r)$.

Of course, Algorithm 3 is only a conceptual algorithm, which seems very difficult to realize. To motivate a proposal for an efficient but only approximative realization we use again its asymptotic properties for small $\alpha_0>0$, i.e. for r_{k+1} close to r_k. When discussing Algorithm 2 we already found for $r<r_k$ close to r_k,

$$\delta_{\mathrm{p}}(x(r), r)\doteq\|\tfrac{1}{2}(r-r_k)^2\ddot{x}(r_k)\|_{H_{\mathrm{p}}(r_k)}\doteq\|x_k(r)-x(r)\|_{H_{\mathrm{p}}(r)}.$$

Similarly, we have for $r<r_k$ close to r_k,

$$\begin{aligned}\psi_{\mathrm{p}}(r)&\doteq\|x_k(\rho(r))-x(r)\|_{H_{\mathrm{p}}(r)}\doteq\min_{0<\rho<r}\|x_k(\rho)-x(r)\|_{H_{\mathrm{p}}(r)}\\&\doteq\min_{0<\rho<r}\|x_k(r)-x(\rho)\|_{H_{\mathrm{p}}(\rho)}\doteq\min_{0<\rho<r}\|x_k(r)-x(\rho)\|_{H_{\mathrm{p}}(x_k(r))}.\end{aligned} \tag{2.17}$$

Therefore, by the quadratic convergence of Newton's method

$$x_k(r)-x(\rho)\doteq H_{\mathrm{p}}(x_k(r))^{-1}\varepsilon(\rho, x_k(r)),$$

where $\varepsilon(\cdot,\cdot)$ is defined by (1.3). Hence, the number r^{p}_{k+1} of Algorithm 3 is approximated by any number r satisfying

$$\underline{\alpha}_0\leqslant\min_{\rho}\varepsilon(\rho, x_k(r))^{\mathrm{T}}H_{\mathrm{p}}^{-1}(x_k(r))\varepsilon(\rho, x_k(r))\leqslant\bar{\alpha}_0, \tag{2.18}$$

where $0<\underline{\alpha}_0<\bar{\alpha}_0$ are some threshold values with $\underline{\alpha}_0\leqslant\alpha_0\leqslant\bar{\alpha}_0$. A solution r with (2.18) can be found by common bisection techniques. Note that the minimal ρ in (2.18) is easily computed. We also note that Algorithm 3 of [21] uses this kind of approximation.

In view of the asymptotic relations (2.17) one also sees that the difference between r and $\rho(r)$ for r close to r_k has to do with the tangential acceleration $\ddot{x}(r)^{\mathrm{T}}H_{\mathrm{p}}(r)\dot{x}(r)$, which can be influenced by choosing other parameters $s=s(r)$ than r to parametrize the central curve. In order to find the right parametrization $s=s(r)$ we note that for s close to s_k the point $x_k(s)=x(s_k)+(s-s_k)x'(s_k)$ is closest to $\hat{x}(s):=x(s_k)+(s-s_k)x'(s_k)+\frac{1}{2}(s-s_k)^2x''(s_k)(s_k)\doteq x(s)$ in the $H_{\mathrm{p}}(s_k)$-norm if

$$s=\arg\min_{\alpha}\|(\alpha-s_k)x'(s_k)+\tfrac{1}{2}(s-s_k)^2x''(s_k)\|_{H_{\mathrm{p}}(s_k)},$$

which is the case iff $x''(s)^{\mathrm{T}}H_{\mathrm{p}}(s)x'(s)=0$. We are thus led to consider the conditions

$$x''(u)H_{\mathrm{p}}(u)x'(u)\equiv 0, \quad \text{resp.}\ \ \eta''(v)H_{\mathrm{d}}(v)\eta'(v)\equiv 0, \tag{2.19}$$

where u and v are the new "arclength" parameters along the primal and dual central path, respectively. The corresponding algorithms move with a speed proportional to the inverse of the sequare root of the norm of the curvature with respect to these parameters in view of

$$\|x_k(s)-x(s)\|_{H_{\mathrm{p}}(s)}=\alpha_0^2=\tfrac{1}{2}\|''(s_k)\|_{H_{\mathrm{p}}(s_k)}(s-s_k)^2. \tag{2.20}$$

It turns out that the parametrizations determined by (2.19) are just the following:

$$u(r):=c^{\mathrm{T}}x(r)+\text{const}, \qquad v(r):=b^{\mathrm{T}}\mu(r)+\text{const}. \tag{2.21}$$

The proof can be obtained by direct calculations using the easily verified formulae

$$\begin{aligned}&\dot{u}(r)=c^{\mathrm{T}}\dot{x}(r)=c^{\mathrm{T}}H_{\mathrm{p}}^{-1}c/r^2=e^{\mathrm{T}}\sigma(r)=e^{\mathrm{T}}\sigma^2(r),\\ &x''(u)=\frac{2}{c^{\mathrm{T}}H_{\mathrm{p}}^{-1}c}H_{\mathrm{p}}^{-1}(cc^{\mathrm{T}}H_{\mathrm{p}}^{-1}\tilde{A}^{\mathrm{T}}(r)-c^{\mathrm{T}}H_{\mathrm{p}}^{-1}c\tilde{A}^{\mathrm{T}}(r))\sigma^2(r),\end{aligned} \tag{2.22}$$

where all arguments are taken at r, resp. $u=u(r)$. This analysis also suggest the following for the realization of Algorithm 3: In order to decide whether a primal or a dual step is made, compare

$$\|x''(u_k)\|_{H_{\mathrm{p}}(u_k)}^{1/2}\dot{u}(r_k) \quad \text{and} \quad \|\eta''(v_k)\|_{H_{\mathrm{d}}(v_k)}^{1/2}\dot{v}(r_k)$$

and take a primal step iff the first number is smaller. Use a suitable bisection procedure to refine r_{k+1}^0 to a solution r_{k+1} of (2.18) or its dual analogue.

Note (see example (4.14)) that it is possible to have $\eta''(v)\equiv p$ but $\|x''(u)\|_{H_{\mathrm{p}}}\not\equiv 0$. As a contrast (see Proposition 2 below), the curvature $f(r)$ appearing in Algorithm 1 is positive for all $r>0$ unless $f(r)\equiv 0$.

It is interesting to note, that the weighted curvatures figuring in (2.22) can also be expressed in terms of the ("first order") σ and M variables, but not in terms of the σ-variables alone. A straightforward computation gives

$$\begin{aligned}&(x''(u)^{\mathrm{T}}H_{\mathrm{p}}(r(u))x''(u))^{1/4}\dot{u}(r)\\ &\quad=\frac{\sqrt{2}}{r}\left(\sigma^2(r)^{\mathrm{T}}M(r)\sigma^2(r)-\frac{(T(r))^2}{Q(r)}\right)^{1/4},\end{aligned} \tag{2.23}$$

where

$$\dot{u}(r)^*Q(r)=e^{\mathrm{T}}\sigma^2(r), \quad T(r)=e^{\mathrm{T}}\sigma^3(r).$$

A consequence of the relation (2.21), which expresses the optimality of the arclength parameter u is that

$$\|x''(u)\|^{1/2}_{H_p(u)}u'(r)\leq \|\ddot{x}(r)\|^{1/2}_{H_p(r)} \tag{2.24}$$

(an analogous inequality holds for the dual curvature).

Observe that r is for the primal-dual problem the "arclength" parameter. Indeed it is easy to see that the duality gap along the path, i.e. the primal-dual objective function: $c^{\mathrm{T}}x(r)+b^{\mathrm{T}}\mu(r)=m\,\cdot\,r$. On the other hand there is an essential difference between the behaviour of the primal-dual curvature $f(r)$ and that of the primal or dual curvatures (with respect to the corresponding arclength parameters u, resp. v): the latter ones may vanish at some point without vanishing everywhere while if $f(r)=0$ for some r, then $f(r)\equiv 0$ everywhere, see Proposition 2 below.

Finally, we note that if the primal problem itself is a primal-dual (i.e. self dual) problem, then the dual problem coincides with the primal one and Algorithms 1, 2 (and 3) lead simply to (essentially) identical sequences $\{r_k\}$, thus over $\mathscr{K}_0$ Algorithm 3 has — in the worst case sense — the same (global) complexity as Algorithm 1.

3. Estimation of the total number of iterations by an integral

In this section, we consider estimates for the number $N=N(R,\delta,\mathscr{P})$ of steps needed by an algorithm for solving $\mathscr{P}$ to reach $x(\delta)$ from $x(R)$,

$$R=r_0>r_1>\cdots>\delta\geq r_N>0.$$

Specifically, we would like to obtain estimates of the form

$$N(R,\delta,\mathscr{P})\leq \text{const } m^{\alpha}\log(R/\delta) \tag{3.1}$$

holding for all $\mathscr{P}$ of a class $\mathscr{K}\subset\mathscr{K}_0$ of linear programs, where the number of constraints $m=m(\mathscr{P})$ depends on $(\mathscr{P})$ but the *complexity exponent* $\alpha=\alpha(\mathscr{K})$ and $\text{const}=\text{const}(\mathscr{K})$ depend only on $\mathscr{K}$ but not on $\mathscr{P}\in\mathscr{K}$.

For *zero order* methods (these take the zero order approximation $x(r_k)$ to $x(\cdot)$ as starting point for the Newton-corrector-steps for computing $x(r_{k+1})$) such an estimate holds with $\alpha=\frac{1}{2}$ for the class $\mathscr{K}_0$ of all programs satisfying our assumptions (for first order methods this will be obtained from the estimates below). This follows from the rule

$$r_{k+1}:=r_k\left(1-\frac{\text{const}}{\sqrt{m}}\right) \tag{3.2}$$

adopted by zero order methods. Within the class of zero order path following methods the analysis of Newton's method shows that this is the best one can achieve.

Note that since

$$\frac{r}{m} \leq cx(r) - \lambda^* \leq m \cdot r$$

(see [20]) the value of

$$m^\alpha \log(R/\delta)$$

is "equivalent" to

$$m^\alpha \log((c^\mathrm{T}x(R) - \lambda^*)/(c^\mathrm{T}x(\delta) - \lambda^*))$$

whenever $\alpha > 0$, if $R > \delta > 0$ satisfies

$$\log(R/\delta) \geq \gamma_0 \tag{3.3}$$

for some universal $\gamma_0 > 0$.

The next arguments are borrowed from approximation theory, where they are used to estimate the number of function evaluation steps necessary for a sequential algorithm to achieve an optimal recovery over an interval (δ, R), see [19]. If the condition (2.11) of Algorithm 1 implies that for all k either

$$\int_{r_{k+1}}^{r_k} f(s)\,\mathrm{d}s \geq c \tag{3.4}$$

or

$$\frac{r_k - r_{k+1}}{r_{k+1}} \geq c \tag{3.5}$$

holds for some universal positive constant $c > 0$, then we obtain an estimate

$$N_1(R, \delta, \mathcal{P}) \leq c_1 \int_\delta^R f(r)\,\mathrm{d}r + c_2 \log(R/\delta) \tag{3.6}$$

for the number $N = N_1(R, \delta, \mathcal{P})$ of steps needed by Algorithm 1. Here $c_1 := 1/c$, $c_2 := |\log(1-c)|$ are again universal constants.

Note that the quantity $f(r_k)(r_k - r_{k+1})\sqrt{r_k/r_{k+1}}$ of (2.11) can be considered as an approximation to $\int_{r_{k+1}}^{r_k} f(r)\,\mathrm{d}r$, provided r_k/r_{k+1} is not too large. This makes (3.4), (3.5) and, therefore, (3.6) plausible. In fact, Zhao and Stoer [25] have shown that there are universal constants $\alpha_0 > 0$ and $c > 0$ so that (3.4), (3.5) and, therefore, (3.6) hold for Algorithm 1. The proof is quite complicated though and is based on a proof of the inequality.

$$|f(s) - f(r_k)| \leq \mathrm{const}(\alpha_0)(1 + (r_k f(r_k))^{-1})f(r_k) \tag{3.7}$$

which holds for all $r_{k+1} \leq s \leq r_k$, and on the observation that, for a suitably defined constant $\beta = \beta(c)$, $r_k f(r_k) \leq \beta$ implies (3.5).

By the same techniques used to show the Lipschitz property (3.7) one can also estimate the higher order derivatives of the central path. If we represent the dual variable $\mu = l - L^{\mathrm{T}}\eta$ in terms of $\eta \in \mathbb{R}^{m-n}$ (see (2.4)), set again $\xi(r) := (x(r), \eta(r))$ and use the matrix $H(r) = H(x(r), x(r))$ (see (2.8)) then one can show:

Proposition 1. *There exist universal constants c_1 and c_2, such that for the central path $\xi(r)$, $r > 0$ and arbitrary integers $p \geqslant 2$,*

$$\|\xi^{(p)}(r)\|_{H(r)}^{1/p} \leqslant c_1(\|\ddot{\xi}(r)\|_{H(r)}^{1/2} + c_2/r). \qquad \square \tag{3.8}$$

Now, by Lemma 1 we have the relation

$$f(r) = \frac{1}{\sqrt{2}} \|\ddot{\xi}(r)\|_{H(r)}^{1/2}. \tag{3.9}$$

Thus it is justified to interpret the right hand side of (3.6) as a *curvature integral.* We can now formulate the main conclusions as follows.

Theorem 1. *The number $N_1(R, \delta, \mathcal{P})$ of basic iteration steps needed by Algorithm* 1 *to reach $x(\delta)$ from $x(R)$ when solving the linear program $\mathcal{P}$ can be estimated from above by the formulae* (3.6), (3.9), *i.e. by an integral of weighted curvatures along the central path.* □

Consequences of this estimate, which lead to upper bounds of the type (3.1), will be derived below. Note that it appears harder to derive analogous estimates for potential reduction algorithms.

Using similar arguments one can probably show that the number $N^{(p)}(R, \delta, \mathcal{P})$ of basic iteration steps of a pth order primal–dual path following algorithm constructed exactly as Algorithms 1, 2, but replacing $x_k(r)$ and $\mu_k(r)$ with pth order approximants of $x(r)$ and $\mu(r)$ at r_k as follows:

$$x_k(r) := x(r_k) + (r - r_k)\dot{x}(r_k) + \cdots + \frac{(r - r_k)^p}{p!} x^{(p)}(r_k)$$

can be estimated by the integral

$$N^{(p)}(R, \delta, \mathcal{P}) \leqslant \text{const} \int_{\delta}^{R} \|\xi^{(p+1)}(r)\|_{H(r)}^{1/(p+1)} \, \mathrm{d}r + \text{const} \log(R/\delta).$$

(See [2] and [19] for a more general use of similar ideas and techniques.) The inequality in Proposition 1 shows that these higher order algorithms are *not worse* than the first order algorithm studied here. (See [9] on an implementation and comparison of higher order algorithms using polynomial extrapolation.) Finally we mention, that the number $N_1(R, \delta, \mathcal{P})$ of steps needed by Algorithm 1 is estimated by the integral also from below: there exist positive universal constants k_0, k_1, k_2 such that

$$N_1(R, \delta, \mathcal{P}) \geqslant k_0 \int_{\delta}^{R} f(r) \, \mathrm{d}r - k_1 \log(R/\delta),$$

see [25] for a proof.

Using a similar reasoning we can expect that for Algorithms 2 and 3 the number of iteration steps $N_2(R, \delta, \mathscr{P})$ and $N_3(R, \delta, \mathscr{P})$ can be estimated by

$$N_2(R, \delta, \mathscr{P}) \leqslant \text{const} \int_r^R \min(\|\ddot{x}(r)\|^{1/2}_{H_{\mathrm{p}}(r)}, \|\ddot{\eta}(r)\|^{1/2}_{H_{\mathrm{d}}(r)})\, \mathrm{d}r \tag{3.10}$$

and

$$\begin{aligned} &N_3(R, \delta, \mathscr{P}) \\ &\quad \leqslant \text{const} \int_r^R \min(\|x''(u)\|^{1/2}_{H_{\mathrm{p}}(u)} u'(r), \|\gamma''(v)\|^{1/2}_{H_{\mathrm{d}}(v)} v'(r))\, \mathrm{d}r. \end{aligned} \tag{3.11}$$

Such estimates are suggested by the asymptotic formulas (2.15) and (2.20), but at this moment we have no rigorous proof for them. Concerning Algorithm 2 we refer to the remark below made after Proposition 2 below.

4. Estimation of the curvature integrals

We now turn to the problem of estimating the curvature integral (3.6),

$$I_1(R, \delta) = I_1(R, \delta; \mathscr{P}) := \int_\delta^R f(r)\, \mathrm{d}r,$$

and remember that it is our goal is to consider "long" intervals (see (3.3)), in order to get global estimates of this integral. This is necessary if we wish to get better estimates than

$$I_1 \leqslant \text{const}\, m^{1/2} \log(R/\delta),$$

since for some problems $\mathscr{P}$ and particular values of r it may happen that

$$f(r) \geqslant \text{const}\, \sqrt{m}/r$$

(this occurs in both examples (4.14), (4.16) below). In order to explain why the expected behavior of the integral is given by

$$I_1 \leqslant \text{const}\, m^{1/4} \log(R/\delta) \quad (\text{for } \log(R/\delta) \geqslant \text{const}) \tag{4.1}$$

we could argue that the average value of

$$rf(r) = \|\sigma(r) \circ \tau(r)\|_2^{1/2} = \|(M(r)e) \circ (I - M(r))e\|_2^{1/2}$$

over all possible projection matrices $M(r)$ of rank n in $\mathbb{R}^m$ is smaller than const $m^{1/4}$. Such an analysis was given in [15], but it involves statistical assumptions on the distribution of the matrices $M(r)$ and other quantities which are hard to justify rigorously. In particular, assumptions on the statistical independence of their values found in the different steps of an interior point algorithm are questionable: In fact we will show that $M(r)$ and other values will change with r according to surprisingly simple $\mathscr{K}_0$-universal laws.

We introduce the new parameter $\rho=-\log r$ and note that $r=0, 1, \infty$ corresponds to $\infty, 0, -\infty$ and $\mathrm{d}\rho/\mathrm{d}r=-1/r$. The following computations get easier if we use the relations following from Lemma 1,

$$[Me]=[\sigma]=rS^{-1}\dot{S},$$

where V or $[v]$ denotes the diagonal matrix with entries taken from the vector v.

Lemma 2. *In the variable $\rho=-\log r$, $\mathcal{M}(\rho)=M(e^{-\rho})$ satisfies the following autonomous differential equation*

$$M':=\frac{\mathrm{d}\mathcal{M}}{\mathrm{d}\rho}=\mathcal{M}[\mathcal{M}e]+[\mathcal{M}e]\mathcal{M}-2\mathcal{M}[\mathcal{M}e]\mathcal{M}=:h(\mathcal{M}). \tag{4.2}$$

Proof. With the above notation Lemma 1 gives

$$M=S^{-1}A^{\mathrm{T}}H_{\mathrm{p}}^{-1}AS^{-1}, \quad H_{\mathrm{p}}=AS^{-2}A^{\mathrm{T}}.$$

Now using

$$\frac{\mathrm{d}}{\mathrm{d}r}S^{-1}=-S^{-1}[\sigma]/r, \qquad \dot{H}_{\mathrm{p}}=-2AS^{-3}\dot{S}A^{\mathrm{T}}=AS^{-1}[\sigma]S^{-1}A^{\mathrm{T}},$$

we get $\dot{M}=(-[Me]\cdot M-M\cdot[Me]+2M\cdot[Me]\cdot M)/r$ from which (4.2) follows, since $\mathrm{d}\rho/\mathrm{d}\rho=-1/r$. □

Note that, in the variable ρ, the curvature integral I_1 has the form

$$\int_{\delta}^{R} f(r)\,\mathrm{d}r=-\int_{\rho(\delta)}^{\rho(R)} f_0(\mathcal{M}(\rho))\,\mathrm{d}\rho \quad \text{where}\, f_0(\mathcal{M})=\|\mathcal{M}e\circ(e-\mathcal{M}e)\|_2^{1/2}.$$

Another more symmetric form of this differential equation is

$$\mathcal{M}'(\rho)=\mathcal{M}D-D\mathcal{M}, \quad \text{where}\, D=\mathcal{M}\cdot[\mathcal{M}e]-[\mathcal{M}e]\cdot\mathcal{M},$$

$f_0(\mathcal{M})=\|De\|^{1/2}$, note that $D^*=-D$. It is surprising that the above differential equation is universal, i.e. it does not depend on $\mathcal{P}$ and defines an (autonomous) flow on the Grassmann manifold $G(m, n)$ on n-dimensional subspaces in $\mathbb{R}^m$.

The problem of estimating the worst case value of the curvature integral reduces thus to the standard optimization problem ($\rho_0>\rho_1$)

$$\max_{\mathcal{P}}\left\{\int_{\rho_0}^{\rho_1} f_0(\mathcal{M}(\rho))\,\mathrm{d}\rho \,\middle|\, \mathcal{M}'=h(\mathcal{M}), \mathcal{M}(\rho_0)=\mathcal{M}(\rho_0, \mathcal{P})\right\},$$

which can be reduced (using e.g. the maximum principle) to a two point boundary problem for ODEs. Note, that both functions $h(M)$ and $f_0(M)$ are globally defined over the *compact* manifold $G(m, n)$. The compactness of this manifold implies that there exists an invariant normed "equilibrium" measure for this flow. A basic result of ergodic theory asserts that

$$\int_{G(m,n)} \left(\lim_{T\uparrow\infty} \frac{1}{T} \int_0^T f_0(\mathcal{M}(\rho))\,\mathrm{d}\rho \right) \mathrm{d}\mu(\mathcal{M}(0))$$

$$= \int_{G(m,n)} f_0(\mathcal{M})\,\mathrm{d}\mu(M) = \beta_{m,n}. \tag{4.3}$$

Suppose now that $R=1$, $\rho(R)=0$, $T=\rho(\delta)$ then $T=\log(R/\delta)=-\log\delta$, and we can define a probability distribution on the class $\mathscr{K}_0$ as the inverse image $\mathrm{d}\nu(\mathscr{P}) = F^{-1}(\mathrm{d}\mu)(M_{\mathscr{P}}(1))$ of the measure $\mathrm{d}\mu(\cdot)$ (the direct map associating to each $\mathscr{P}\in\mathscr{K}_0$ the value $F(\mathscr{P}) := M_{\mathscr{P}}(1)$, i.e. for each program $\mathscr{P}$ we look to the value of the projection matrix at $R=1$). Now if the above flow is ergodic, then

$$\lim_{\delta\to 0} \frac{1}{\log(R/\delta)} \int_\delta^R f_{\mathscr{P}}(r)\,\mathrm{d}r = \beta_{m,n}$$

with probability one. Note that the asymptotics of $f_0(\rho)$, $\rho\to\infty$ expressed in Lemma 3 below contradicts ergodicity, but is, of course, favorable for the smallness of $\beta_{m,n}$. In any case, the relation (4.3) says that the average value of the left hand side with respect to $\mathrm{d}\nu(\mathscr{P})$ equals the right hand side.

From the differential equation for the function $M(r)$ we immediately obtain a differential equation connecting the variables $\sigma(r)=M(r)e$ and $\sigma^2(r)-\sigma(r)$,

$$\dot\sigma(r) = (2M(r)-I)\frac{\sigma^2(r)-\sigma(r)}{r}. \tag{4.4}$$

Observe that $U=2M-I$ is an involution $U=U^*=U^{-1}$ so that $\|\dot\sigma(r)\|^{1/2}=f(r)$. An immediate consequence of this relation is the following proposition.

Proposition 2. *If $f(r_0)=0$ for some $r=r_0>0$ then $\dot\sigma(r)\equiv 0$, thus $f(r)\equiv 0$.* □

We even conjecture that the same proposition holds also for the primal or dual curvatures with respect to the r-parameter, more precisely, e.g. in the primal case

$$\mathrm{const}\, f(r) \leqslant \|\ddot x(r)\|_{H_p(r)} \leqslant \mathrm{const}\, f(r), \tag{4.5}$$

so that Algorithm 2 would be not essentially different from Algorithm 1.

We now derive some estimates for the complexity integral I_1 (3.6). A first estimate follows from Lemma 1, which implies

$$\sigma(r) = M(r)e, \qquad \tau(r) = (I-M(r))e,$$

with an orthogonal projection $M(r)$. Hence

$$\|\sigma\|_\infty \leqslant |\sigma\|_2 \leqslant \|e\|_2 = \sqrt{m}, \qquad \|\tau\|_\infty \leqslant \|\tau\|_2 \leqslant \|e\|_2 = \sqrt{m}, \tag{4.6}$$

and thus

$$rf(r) \leqslant \sqrt{m}, \tag{4.7}$$

so that by (3.6) we get the complexity estimate

$$N_1(R, \delta, \mathcal{P}) \leq \sqrt{m} \log \frac{R}{\delta}, \tag{4.8}$$

which is wellknown for zero-order methods, also for the first-order algorithm 1.

The next lemma shows however that (4.7) is asymptotically a rather bad estimate, so that (4.8) will be too pessimistic in most cases.

Lemma 3. *The functions $\sigma_i(r)$, $i = 1, \ldots, m$, have the following limit behaviour at $r = 0$:*

$$\lim_{r \downarrow 0} \sigma_i(r) = \begin{cases} 0 & \textit{if } i \textit{ is inactive at the optimum}, \\ 1 & \textit{if } i \textit{ is active at the optimum}. \end{cases}$$

Thus

$$\lim_{r \downarrow 0} rf(r) = 0. \qquad \square \tag{4.9}$$

The proof of (4.9) follows from the observation, see e.g. [23], that the functions $\dot{s}(r)$ and $\dot{\mu}(r)$ have finite limits for $r \downarrow 0$ for all linear programs $\mathcal{P} \in \mathcal{K}_0$ (even for programs with a degenerate optimum), and from the formulae (see Lemma 1)

$$\begin{aligned} &\sigma(r) = rs^{-1}(r) \circ \dot{s}(r), \quad \tau(r) = r\mu^{-1}(r) \circ \dot{\mu}(r), \quad \sigma(r) + \tau(r) = e, \\ &s(r) \circ \mu(r) = re, \quad \sigma(r) \circ \tau(r) = r\dot{s}(r) \circ \dot{\mu}(r). \end{aligned} \tag{4.10}$$

Using the limiting behaviour (4.9) it is easy to show the following relatively weak asymptotic result:

Corollary. *For each $\mathcal{P} \in \mathcal{K}_0$ and each $R > 0$ there is a $\gamma = \gamma(\mathcal{P}, R) > 0$ so that*

$$N_1(R, \delta, \mathcal{P}) \leq c \cdot m^{1/4} \log \frac{R}{\delta}$$

for all δ with $\log(R/\delta) \geq \gamma$. Here c does not depend on $\mathcal{P}$. □

An interesting consequence of (4.6) is obtained for the following special class of programs: We say that a linear program $\mathcal{P} = \mathcal{P}_1 \oplus \cdots \oplus \mathcal{P}_k$ with the feasible set $\mathcal{P}$ is a direct sum of programs $\mathcal{P}_j$, $j = 1, \ldots, k$, with the feasible sets P_j if $m = m_1 + \cdots + m_k$, $x = x_1 \oplus \cdots \oplus x_k$ and $x \in P$ iff $x_j \in \mathcal{P}_j$ for all j. Suppose that $\mathcal{P}$ is a direct sum such that $m_j \leq m_0$ for $j = 1, \ldots, k$. Then

$$N_1(R, \delta, \mathcal{P}) \leq \text{const } m_0^{1/4} m^{1/4} \log(R/\delta). \tag{4.11}$$

Indeed note that $\sigma(r) = \sigma_1(r) \oplus \cdots \oplus \sigma_k(r)$, $\tau(r) = \tau_1(r) \oplus \cdots \oplus \tau_k(r)$ and according to (4.6) $\|\sigma_j(r)\|_\infty \leq \sqrt{m_0}$, $\|\tau(r)\|_2 \leq \sqrt{m}$. Hence

$$rf(r) = \|\sigma(r) \circ \tau(r)\|_2^{1/2} \leq m_0^{1/4} m^{1/4}. \tag{4.12}$$

Noting that the above curvature integrals are continuous functions of the parameters (R, δ, A, b) for all fixed δ, we get complexity estimates for arbitrarily degenerate Klee–Minty type problems, where $m_0 = 2$.

We next relate the complexity integral I_1 in (3.6) to another integral by showing that there is a constant β $(=\sqrt[4]{2})$ so that

$$rf(r) \leq \beta(rg(r)+m^{1/4}), \tag{4.13}$$

where

$$g(r) := \frac{1}{r}\left(\sum_{i=1}^{m} \sigma_i^4(r)\right)^{1/4} = \frac{1}{r}\,\|\sigma^2(r)\|_2^{1/2}.$$

Proof of (4.13). We have by Lemma 1 and (4.6),

$$\|\sigma \circ \tau\|^2 = \|\sigma^2 - \sigma\|^2 \leq (\|\sigma^2\|^2 + \|\sigma\|)^2 \leq 2(\|\sigma^2\|^2 + \|\sigma\|^2) \leq 2(\|\sigma^2\|^2 + m).$$

Hence

$$rf(r) = \|\sigma \circ \tau\|_2^{1/2} \leq \sqrt[4]{2}\,(\|\sigma^2\|^2 + m)^{1/4} \leq \sqrt[4]{2}\,(\|\sigma^2\|^{1/2} + m^{1/4}). \qquad \square$$

Therefore, and by (4.13), upper bounds for the integral

$$I_2(R, \delta) := \int_\delta^R g(r)\,\mathrm{d}r = \int_\delta^R \frac{1}{r}\,\|\sigma^2(r)\|_2^{1/2}\,\mathrm{d}r$$

also provide upper bounds for the integral $I_1 = \int_\delta^R f(r)\,\mathrm{d}r$ in (3.6). The next observation: The parameter invariance of the integrals

$$I_1 = \int_\delta^R f(r)\,\mathrm{d}r = \int_{t(\delta)}^{t(R)} \left(\sum_i \left(\frac{s_i'(t)}{s_i(t)}\,\frac{\mu_i'(t)}{\mu_i(t)}\right)^2\right)^{1/4} \mathrm{d}t,$$

$$I_2 = \int_\delta^R \frac{(\sum \sigma_i^4(r))^{1/4}}{r}\,\mathrm{d}r = \int_{t(r)}^{t(R)} \left(\sum_i (s_i'(t)/s_i(t))^4\right)^{1/4} \mathrm{d}t \tag{4.14}$$

often allows to estimate them by choosing a particular parameter t depending on the class of problems $\mathscr{K}$. If for example we can find a parameter t such that

$$\left|\frac{s_i'(t)}{s_i(t)}\right| \leq \frac{\text{const}}{t}, \quad i = 1, \ldots, m, \tag{4.15}$$

then, as $I_2(R, \delta) \leq \text{const}\; m^{1/4} \log(t(R)/t(\delta))$, we find the complexity exponent $\alpha = \frac{1}{4}$ provided t and r are equivalent up to factors polynomially bounded in m. Such parameters t can be found for all programs, where $m = n+1$ or for their duals, where $n = 1$. For example, for the program ("simplex problem")

$$\min_{x \in \mathbb{R}^m} \{c^{\mathrm{T}}x \mid \textstyle\sum x_i \leq 1,\, x_i \geq 0,\, i = 1, \ldots, m\} \tag{4.16}$$

it is true that for the parameter $t = t(r) = r(1 - \sum x_i(r))^{-1}$ we have

$$\left|\frac{s_i'(t)}{s_i(t)}\right| \leq \frac{2}{t}, \quad i = 0, \ldots, m,$$

so that

$$N_1(R,\delta)\leqslant \text{const}\; m^{1/4}\log\frac{t(R)}{t(\delta)}\leqslant \text{const}\; m^{1/4}\log\frac{Rm}{\delta}. \tag{4.17}$$

This follows from the easily computable, explicit form of the path (and of the integral (4.14)) when the parameter t is used, noting that $t(R)\leqslant mR$. Note that in this example $\|\sigma(r(t))\|\cong\sqrt{m}$ holds, for instance, for $t=1$, $c_1=1$, $c_2=\cdots=c_m=\sqrt{m}$, but this does not contradict (4.15) since $\mathrm{d}t/\mathrm{d}r\cong\sqrt{m}$.

The duals of the class of "simplex" problems have a one-dimensional feasible set; for general linear programs in $\mathbb{R}^1$,

$$\min\{x\,|\,x\geqslant\alpha_i,\ i=1,\ldots,m_1,\ x\leqslant\beta_i,\ i=1,\ldots,m_2\} \tag{4.18}$$

introduce the parameter

$$t=\frac{x(r)-x^*}{\bar\beta-x(r)},\quad \text{where } x^*:=\max_i\,\alpha_i,\quad \bar\beta:=\min_i\,\beta_i.$$

Then it is easy to show that (4.15) holds with const $=1$. Indeed (4.15) is equivalent to

$$\left|\frac{s_i'(x)}{s_i(x)}\right|\leqslant\frac{t'(x)}{t(x)}\quad\text{for all } i,\,x,$$

the validity of which is easily checked using $|s_i'(x)|\equiv 1$ and

$$\frac{t'(x)}{t(x)}\equiv\frac{1}{x-x^*}+\frac{1}{\bar\beta-x},\ \frac{1}{x_i(x)}\leqslant\frac{1}{x-x^*}+\frac{1}{\bar\beta-x}.$$

For one-dimensional problems (4.18) we have

$$N_1(R,\delta,\mathcal{P})\leqslant\text{const}\; m^{1/4}\log\left(m\frac{c^{\mathrm{T}}x(R)-x^*}{c^{\mathrm{T}}x(\delta)-x^*}\right). \tag{4.19}$$

This follows immediately from (4.14), (4.15) and the definition of $t=t(r)$, since $\bar\beta-x(\delta)\leqslant\bar\beta-x^*$ and $\bar\beta-x(R)\geqslant m^{-1}(\bar\beta-x^*)$ for all R, $\delta>0$.

The presence of the factor m in the argument of the log function in (4.17) and (4.19) makes it difficult to extend the corresponding $m^{1/4}$ estimate to an arbitrary direct sum of the last two kinds of problems unless the number of summands is universally bounded.

Also one-sided bounds like

$$\frac{s_i'(t)}{s_i(t)}\leqslant\frac{\text{const}}{t}\quad\text{and/or}\quad\frac{\mu_i'(t)}{\mu_i(t)}\leqslant\frac{\text{const}}{t},\qquad i=1,\ldots,m, \tag{4.20}$$

are helpful: If e.g. for the parameter $t=r$,

$$\sigma(r)=rs^{-1}(r)\circ\dot{s}(r)\leqslant Ke, \qquad \tau(r)=r\mu^{-1}(r)\dot{\mu}(r)\leqslant Ke,$$

then by $\tau(r)=e-\sigma(r)$ one finds

$$\|\sigma(r)\circ\tau(r)\|_\infty\leqslant\kappa^2$$

so that

$$I_1(R,\delta)=\int_\delta^R\frac{1}{r}\|\sigma(r)\circ\tau(r)\|_2^{1/2}\,\mathrm{d}r\leqslant Km^{1/4}\log\frac{R}{\delta}.$$

Note that inequalities of this type hold for "simplex problems" and therefore in all cases, where P is a direct sume of simplices. Here it is true that

$$\sigma_i(r)\leqslant 1, \quad i=1,\ldots,m, \ r>0. \tag{4.21}$$

The dual condition is $\tau_i(r)=1-\sigma_i(r)\leqslant 1$, that is

$$\sigma_i(r)\geqslant 0, \quad i=1,\ldots,m, \ r>0, \tag{4.22}$$

and this holds for the duals (direct sums of duals) of the elements of the previous class.

Both of these conditions have interesting geometric interpretations: (4.21) requires — see Lemma 1 — that the relative change in the distance to the ith bounding hyperplane $\dot{s}_i(r)/s_i(r)$ be not larger than the relative change $\dot{r}/r=1/r$ of the parameter r, a condition which limits the degeneracy of the problem. This condition is parameter invariant,

$$\frac{\mathrm{d}}{\mathrm{d}t}\log(s_i(t))\leqslant\frac{\mathrm{d}}{\mathrm{d}t}\log(r(t)), \quad i=1,\ldots,m.$$

In [20] it is proved that this condition always holds in the mean

$$\sum_i\frac{s_i'(r)}{s_i(r)}\leqslant m\frac{1}{r}.$$

Condition (4.22) requires that the distances $s_i(r)$, $i=1,\ldots,m$, be monotone functions of r. This seems to hold for most (but not all) problems, which can be put in the form

$$\min_x \max_{i=1,\ldots,M}(\beta_i-\alpha_i^{\mathrm{T}}x).$$

Also this condition always holds in the mean

$$\sum_i\frac{\dot{s}_i}{s_i}=\frac{\mathrm{d}}{\mathrm{d}r}\log(s_1(r)\cdots s_m(r))>0.$$

If only one set inequalities of the type (4.20) holds (e.g. simplex problems or their duals) then one can prove a slightly weaker result:

Proposition 3. *Suppose that* $\sigma_i(r) \leqslant K$ *for* $i = 1, \ldots, m$ *and* $r > 0$, *where* $K \geqslant 1$. *Then for* $\log(R/\delta) \geqslant 1$ *both curvature integrals* I_1 *and* I_2 *satisfy*

$$I_1(R, \delta), I_2(R, \delta) \leqslant c \cdot Km^{3/8} \log \frac{R}{\delta},$$

where c is a $\mathcal{K}_0$*-universal constant.*

Proof. We shall use the following identity:

$$\frac{2}{r} e^{\mathrm{T}} \sigma^3 = \frac{2}{r} \sum_i \sigma_i^3(r) = \frac{2}{r} e^{\mathrm{T}} \sigma^2 + e^{\mathrm{T}} \dot{\sigma}(r), \tag{4.23}$$

which is obtained from the differential equation (4.4),

$$e^{\mathrm{T}} \dot{\sigma}(r) = \frac{1}{r} (2\sigma^{\mathrm{T}} - e^{\mathrm{T}})(\sigma^2 - \sigma)$$

using $Me = \sigma$, $M = M^{\mathrm{T}} = M^2$. Integrating (4.23) between δ and R we get

$$2 \int_\delta^R \frac{\sum_i \sigma_i^3(r)}{r} \mathrm{d}r = 2 \int_\delta^R \frac{\sum_i \sigma_i^2(r)}{r} \mathrm{d}r + \left[\sum_i \sigma_i(r) \right]_\delta^R. \tag{4.24}$$

Because of (4.13) it will be sufficient to estimate I_2. The concavity of the function $t^{1/4}$ gives via Jensen's inequality

$$\int_\delta^R \frac{1}{r} (\sum \sigma_i^4(r))^{1/4} \mathrm{d}r \leqslant \left(\int_\delta^R \frac{\sum \sigma_i^4(r)}{r} \mathrm{d}r \right)^{1/4} \left(\log \frac{R}{\delta} \right)^{3/4}. \tag{4.25}$$

In order to estimate the first factor on the right hand side of (4.25) we fix r for the moment, write briefly σ for $\sigma(r)$ and introduce the index set $\mathcal{J} := \{i \mid |\sigma_i| > K\}$ and its complement $\mathcal{J}^\perp := \{1, \ldots, m\} \backslash \mathcal{J}$. Then $|\sigma_i| \leqslant K$ for $i \in \mathcal{J}^\perp$ and $\sigma_i \geqslant -K$ for $i \in \mathcal{J}$. By $|\sigma_i| \leqslant \sqrt{m}$ for all i (4.6) we get

$$\sum_{i \in \mathcal{J}} \sigma_i^4 \leqslant \sqrt{m} \left| \sum_{\mathcal{J}} \sigma_i^3 \right|,$$

so that

$$\sum_i \sigma_i^4 \leqslant \sqrt{m} \left| \sum_{\mathcal{J}} \sigma_i^3 \right| + mK^4.$$

Also

$$\left| \sum_{\mathcal{J}} \sigma_i^3 \right| \leqslant -\sum_i \sigma_i^3 + \sum_{\mathcal{J}^\perp} |\sigma_i^3| \leqslant -\sum_i \sigma_i^3 + mK^3.$$

Hence

$$\sum_i \sigma_i^4 \leqslant \sqrt{m}\left(-\sum_i \sigma_i^3 + mK^3\right) + mK^4.$$

Therefore, using (4.24) and $0 \leqslant \sigma^{\mathrm{T}}\sigma = e^{\mathrm{T}}\sigma \leqslant m$ (4.6),

$$\begin{aligned}\int_\delta^R \frac{\sum_i \sigma_i^4(r)}{r}\,\mathrm{d}r &\leqslant \sqrt{m}\left(-\tfrac{1}{2}e^{\mathrm{T}}\sigma(r)|_\delta^R - \int_\delta^R \frac{\|\sigma(r)\|^2}{r}\,\mathrm{d}r + mK^3 \log\frac{R}{\delta}\right)\\ &\quad + mK^4 \log\frac{R}{\delta}\\ &\leqslant \sqrt{m}\left(\tfrac{1}{2}m + mK^3\frac{R}{\delta}\right) + mK^4\log\frac{R}{\delta}.\end{aligned}$$

Combining this estimate with (4.25) and using $\log(R/\delta) \geqslant 1$, $K \geqslant 1$ we finally get the desired result,

$$\begin{aligned}I_2(R,\delta) &\leqslant \left[\tfrac{1}{2}m\sqrt{m}\left(1 + K^3\log\frac{R}{\delta}\right) + mK^4\log\frac{R}{\delta}\right]^{1/4}\left(\log\frac{R}{\delta}\right)^{3/4}\\ &\leqslant m^{1/4}[\tfrac{1}{2}\sqrt{m} + \sqrt{m}\,K^3 + K^4]^{1/4}\log\frac{R}{\delta}\\ &< \sqrt[4]{3}\,m^{3/8}\log\frac{R}{\delta}. \qquad \square\end{aligned}$$

Finally we give an example of a linear program, for which the values of the above curvature integrals are not smaller than const $\cdot\, m^{-\varepsilon+1/3}\log(R/\delta)$ for arbitrary $\varepsilon > 0$ and large enough m. This program is a direct sum of k one-dimensional problems, where the components of the central path are given by

$$\frac{c_j}{r} + \frac{1}{1-x_j} - \frac{n}{x_j} = 0, \quad c_j = \rho^j,\ j = 1, \ldots, k.$$

Here

$$\rho := \beta/\alpha, \quad \beta := (n - \sqrt{n})^{-1}, \quad \alpha := (n - \tfrac{1}{4}\sqrt{n})^{-1},$$

and $\delta := c_k\alpha$, $R := 1$, $k := \sqrt{n}$, thus $m = n^{3/2}$.

The idea of this construction was to find first a one-dimensional central path Π on $[\delta, R]$ for which $\|\sigma^2 - \sigma\|^{1/2}$ (or what is the same $\|\sigma\|$) reaches (and keeps) its maximal value $\mathrm{O}(m^{1/2})$ on a relatively long "critical" subinterval $[\alpha, \beta]$ of $[\delta, R]$. By scaling the objective function in the k identical copies of Π (= the direct summands of the example) differently, so that their critical subintervals do not overlap and fill out the full interval $[\delta, R]$, we are able to maintain a relatively large value of the local curvature on the whole interval $[\delta, R]$.

The path Π corresponds to the one-dimensional equation

$$\frac{1}{r}+\frac{1}{1-x}-\frac{n}{x}=0,$$

belonging to the program

$$\min\{x\,|\,x\leqslant 1 \text{ and } x\geqslant 0 \text{ counted } n \text{ times}\}.$$

Denoting $s_0(r):=1-x(r)$, $s^j(r):=1-x_j(r)$, $j=1,\dots,k$, $\varphi_0(r)=\dot{s}_0(r)/s_0(r)$, we observe that

$$I_2(R,\delta)>\int_\delta^R\left(\sum_{j=1}^k\left(\frac{\dot{s}^j(r)}{s^j(r)}\right)^4\right)^{1/4}\mathrm{d}r=\int_\delta^R\left(\sum_j\left(\frac{\varphi_0(r/c_j)}{c_j}\right)^4\right)^{1/4}\mathrm{d}r$$

$$>\sum_j\int_{[c_j\alpha,c_j\beta]}\frac{\varphi_0(r/c_j)}{c_j}\,\mathrm{d}r=k\int_{[\alpha,\beta]}\varphi_0(r)\,\mathrm{d}r\geqslant\tfrac{1}{2}k.$$

Indeed it is easy to see that $\rho\cong 1-3/(4\sqrt{n})$, i.e. $\rho^{\sqrt{n}}\cong r\,\mathrm{e}^{-3/4}$ and $\delta=c_k\alpha\geqslant\rho^k/n$, therefore for large enough n (i.e. $n\geqslant n_0$),

$$\log\frac{R}{\delta}>-\log\delta=-k\log\rho-\log\alpha>\frac{k}{2\sqrt{n}}+\tfrac{1}{2}\log n.$$

On the interval $[\alpha,\beta]$ — whose length is larger than $3\sqrt{n}/4n^2$ — the value of $\varphi_0(r)$ is larger than $n^2/3\sqrt{n}$ and therefore

$$\int_{[\alpha,\beta]}\varphi_0(r)\,\mathrm{d}r\geqslant\tfrac{1}{4}.$$

We omit details, providing only the expression for

$$\varphi_0(r)=x^2(1-x)/r^2(x^2+n(1-x)^2),$$

where $x=x(r)$; on the interval $[\alpha,\beta]$, $x(r)$ varies within $[1-2/\sqrt{n},1-1/\sqrt{n}]$. Thus $I_2\geqslant m^\gamma\log(R/\delta)$ will hold iff

$$k^\gamma n^\gamma\left(\frac{1}{2\sqrt{n}}+\frac{\log n}{2k}\right)\leqslant\tfrac{1}{4}.$$

Setting $k=\sqrt{n}$ the last inequality holds for large enough n iff $\gamma<\frac{1}{3}$. Note also that $\log(R/\delta)\geqslant\frac{1}{2}(1+\log n)$, i.e. the path is not short. For the integral I_1 essentially the same estimations can be derived since on the critical intervals σ^2 and $\sigma(1-\sigma)$ have the same order of magnitude.

Proposition 4. *There exists a class $\mathscr{K}\subset\mathscr{K}_0$ of linear programs with $m=m(\mathscr{P})=n=n(\mathscr{P})^3$ for $\mathscr{P}\in\mathscr{K}$ and the following property: For any constant $c>0$ and any $\varepsilon>0$*

there exist a $\mathcal{P} \in \mathcal{K}$ *and numbers* $R > \delta > 0$ *so that Algorithm* 1 *needs for solving* $\mathcal{P}$ *at least*

$$N_1(R, \delta, \mathcal{P}) \geq cm^{-\varepsilon+1/3} \log(R/\delta)$$

steps to reach $x(\delta)$ *from* $x(R)$. □

Acknowledgement

The authors profited from conversations with Fl. Jarre. They also thank an anonymous referee for suggesting many improvements of the text of this paper.

References

[1] I. Adler, N. Karmarkar, M.G.C. Resende and G. Veiga, "An implementation of Karmarkar's algorithm for linear programming," *Mathematical Programming* 44 (1989) 297-335.

[2] W. Blaschke and K. Reidmeister, *Affine Differentialgeometrie* (Springer, Berlin, 1924).

[3] P.T. Boggs, P.D. Domich, J.R. Donaldson and C. Witzgall, "Algorithmic enhancements to the method of centers for linear programming," *ORSA Journal on Computing* 1 (1989) 159-171.

[4] P.T. Boggs, P.D. Domich, J.R. Donaldson and C. Witzgall, "Optimal 3-dimensional methods for linear programming," Tehcnical Report NISTIR 89-4225, National Institute of Standards and Technology (Gaithersburg, MD, 1989).

[5] J.L. Goffin, A. Haurie and J.P. Vial, "Decomposition and nondifferentiable optimization with the projective algorithm," Preprint G-89-25, Faculty of Manamgement, McGill University (Montreal, Que., 1989).

[6] Cl. Gonzaga, "An algorithm for solving linear programming problems in $O(n^3L)$ operations," in: N. Meggido, ed., *Progress in Mathematical Programming, Interior Point and Related Methods* (Springer, New York, 1988) pp. 1-28.

[7] M. Iri, "Integrability of vector and multivector fields associated with interior point methods for linear programming," *Mathematical Programming (Series B)* 52 (1991) 511-525, this issue.

[8] Fl. Jarre, "On the method of analytic centers for solving smooth, convex programs," in: S. Dolecki, ed., *Optimization, Proceedings of the Fifth French-German Conference. Lecture Notes in Mathematics No. 1405* (Springer, Berlin, 1989) pp. 69-85.

[9] Fl. Jarre, G. Sonnevend and J. Stoer, "An implementation of the method of analytic centers," in: A. Bensoussan and J.L. Lions, eds., *Analysis and Optimization of Systems. Lecture Notes in Control and Information Sciences No. 111* (Springer, Berlin, 1988) pp. 297-307.

[10] N. Karmarkar, "Riemannian geometry underlying interior point methods," Lecture and Preprint presented at the *13th International Symposium on Mathematical Programming, Tokyo* (Tokyo, 1988).

[11] M. Kojima, Sh. Mizuno and A. Yoshise, "An $O(\sqrt{n}\ L)$ iteration potential reduction algorithm for linear complementarity problems," *Mathematical Programming* 50 (1991) 331-342.

[12] I.J. Lustig, R.E. Marsten and D.E. Shanno, "Computational experience with a primal-dual interior point method for linear programming," Technical Report SOR 89-17, School of Engineering and Applied Science, Princeton University (Princeton, NJ, 1989).

[13] A. Marxen, "Primal barrier methods for linear programming," Ph.D. Thesis, Department of Operations Research, Stanford University (Stanford, CA, 1989).

[14] J. Mennicken, "Implementation of a first order central path following algorithm for solving large linear programs," Report No. 202, Institut für Angewandte Mathematik und Statistisk, Universität Würzburg (Würzburg, 1990).

[15] S. Mizuno, M.J. Todd and Y. Ye, "Anticipated behaviour of path-following algorithms for linear programming," Technical Report No. 878-1989, School of Operations Research and Industrial Engineering, Cornell University (Ithaca, NY, 1989).

[16] Ju. Nesterov and A. Nemirovskii, *Self-Concordant Functions and Polynomial-Time Algorithms for Convex Programming* (Central Economical and Mathematical Institute USSR Academy of Science, Moscow, 1989).

[17] C. Roos and J.-Ph. Vial, "A polynomial method of approximate weighted centers for linear programming," to appear in: *Mathematical Programming* (1992).

[18] G. Sonnevend, "Application of analytic centers to feedback design for systems with uncertainties," in: D. Hinrichsen and B. Martensson, eds., *Control of Uncertain Systems* (Birkhäuser, Basel, 1989) pp. 271-289.

[19] G. Sonnevend, "Sequential algorithms of optimal order global error for the uniform recovery of functions with monotone *r*th derivatives," *Analysis Mathematica* 10 (1984) 311-335.

[20] G. Sonnevend and J. Stoer, "Global ellipsoidal approximations and homotopy methods," *Applied Mathematics and Optimization* 21 (1989) 139-166.

[21] G. Sonnevend, J. Stoer and G. Zhao, "On the complexity of following the central path of linear programs by linear extrapolation," *Methods of Operations Research* 62 (1990) 19-31.

[22] M. Todd, "Recent developments and new directions in linear programming," in: M. Iri and K. Tanabe, eds., *Mathematical Programming - Recent Developments and Applications* (Kluwer Academic Publishers, Dordrecht-Boston-London, 1989) pp. 109-157.

[23] C. Witzgall, P.T. Boggs and P.D. Domich, "On the convergence behavior of trajectories for linear programming," to appear in: *Proceedings of the AMS-IME-SIAM Research Conference on "Mathematical Developments Arising from Linear Programming Algorithms," June 26-30, 1988, Bowdoin College* (Brunswick, ME).

[24] Y. Ye, "A class of potential functions for linear programming," Preprint, Department of Management Science, The University of Iowa (Iowa City, IA, 1988/1989).

[25] G. Zhao and J. Stoer, "Estimating the complexity of path following methods for solving linear programs by curvature integrals," Technical Report No. 225, Institut für Angewandte Mathematik und Statistik, Universität Würzburg (Würzburg, 1990).