11 Self-concordant barriers

A *barrier function* on a set $X \subset \mathbb{R}^n$ is a function $F : X^o \to \mathbb{R}$ such that $\lim_{x \in \partial X} F(x) = +\infty$. For convenience we may define $F(x) = +\infty$ for $x \notin \operatorname{int} K$. Adding a barrier to the objective function of a minimization problem over X prevents the iterates of an iterative algorithm to step out of the interior of X. In contrast to barriers, a *penalty function* on X is a function $F : \mathbb{R}^n \to \mathbb{R}$ which is zero on X, positive outside of X and increases with the distance from X. It hence penalizes iterates which move too far away from X. From one iterate to the next the weight of a barrier function is decreased, while the weight of a penalty function is increased. Both the weighted barrier function and the weighted penalty function thus converge point-wise to the function

$$F^*(x) = \begin{cases} 0, & x \in X, \\ +\infty, & x \notin X \end{cases}$$

outside of ∂X.

In the case of conic programs

$$\min_{x \in K} \langle c, x \rangle : \ Ax = b$$

the difficulty is hidden in the conic constraint $x \in K$. Here $K \subset \mathbb{R}^n$ is a regular convex cone.

The basic idea of interior-point methods for solving conic programs is to eliminate the conic constraint by the addition of a barrier function $F : K^o \to \mathbb{R}$ to the linear objective. The barrier function should satisfy the following requirements:

- $F(x)$ is convex (the problem should remain convex),
- $F(x)$ is sufficiently smooth (to be able to use second-order methods),
- $\lim_{x \to \partial K} F(x) = +\infty$ (acts as a barrier for K),
- F behaves well with the Newton method.

The last requirement needs a formalization, which is achieved by the property of *self-concordance*.

11.1 Newton method

We consider the problem of minimizing a convex C^2 function $f : D \to \mathbb{R}$ defined on a convex domain $D \subset A$, where A is some affine space. For simplicity we assume that the Hessian f'' of the function is positive definite everywhere on D. Given an iterate $x_k \in D$, the (damped) *Newton algorithm* computes the next iterate as

$$x_{k+1} = x_k - \gamma_k (f''(x_k))^{-1} f'(x_k).$$

Here $\gamma_k \in (0, 1]$ is the damping coefficient. If $\gamma_k = 1$ the algorithm makes a full, for $\gamma_k < 1$ a damped Newton step.

For $\gamma_k = 1$ the point x_{k+1} can be interpreted as the minimizer of the strictly convex second order Taylor polynomial of f at x_k,

$$q_k(x) = f(x_k) + \langle f'(x_k), x - x_k \rangle + \frac{1}{2}(x - x_k)^T f''(x_k)(x - x_k).$$

The above formulae need an explanation. The gradient $f'(x_k)$ and the Hessian $f''(x_k)$ are a linear form and a quadratic form, respectively, on the tangent space to A at x_k, which can be identified with the vector space V underlying the affine space. Hence the gradient is an element in the dual vector space V^*, and the Hessian can also be considered as a self-adjoint linear map from V to V^*. The difference $x - x_k$ is a vector, and the gradient and Hessian can be applied to it, leading to a number and an element of V^*, respectively. The expression $(f''(x_k))^{-1} f'(x_k)$ is also a vector and can be subtracted from a point in affine space. The formulae can hence be seen as identities in the affine space A or in $\mathbb{R}$, respectively, and yield the same result independently of the chosen coordinate system. Such a description is called *coordinate-free*.

The Newton algorithm is hence *affinely invariant*, i.e., its output does not change when computed in another coordinate system on the affine space. Therefore there exists no other natural norm on V than the Euclidean

norm $||\cdot||_{x_k}$ defined by the Hessian $f''(x_k)$ at the current iterate. In this norm the level subsets $\{x \in A \,|\, q_k(x) \leq c\}$ are norm balls around the minimizer x_{k+1} of the second order approximation $q_k(x)$. This norm induces a dual norm on V^*, whose matrix is given by the inverse Hessian $(f''(x_k))^{-1}$. We shall denote this norm also by $||\cdot||_{x_k}$. There is no danger of confusion, as the matrix depends on the object the norm is applied to.

For a full Newton step the current iterate lies at a distance

$$\begin{aligned}\rho &= ||x_{k+1} - x_k||_{x_k} = \sqrt{(x_{k+1} - x_k)^T f''(x_k)(x_{k+1} - x_k)} \\ &= ||f'(x_k)||_{x_k} = \sqrt{f'(x_k)^T (f''(x_k))^{-1} f'(x_k)}\end{aligned}$$

from the minimizer of q_k, which at the same time equals the norm of the gradient at the current iterate. This quantity can also be expressed through the difference between the current function value and the minimum value of $q_k(x)$,

$$f(x_k) - q_k(x_{k+1}) = -\langle f'(x_k), x_{k+1} - x_k\rangle - \frac{1}{2}(x_{k+1} - x_k)^T f''(x_k)(x_{k+1} - x_k) = \frac{1}{2} f'(x_k)^T (f''(x_k))^{-1} f'(x_k) = \frac{\rho^2}{2}.$$

Definition 11.1. The value $\rho = ||f'(x)||_x$ is called the *Newton decrement* of the function f at the point x.

By affine invariance the Newton decrement carries full information about the position of the current iterate x_k with respect to the minimizer of the quadratic approximation q_k of f. We may hence limit the analysis of the Newton method to the change of the Newton decrement after the Newton step. Note that the decrement is zero if and only if the current iterate is the minimizer of f.

Geometric interpretation: We shall now give a geometric interpretation of the Newton method. The minimizer x^* of the function f is characterized by the condition $f'(x^*) = 0$. We may then ignore the information on the values of f and consider only its gradient.

If V is the vector space underlying the affine space A containing the domain of interest, then the gradient is an element of the dual vector space V^*. The graph of the map $\nabla f : x \mapsto f'(x)$ is then an n-dimensional *submanifold* M of the $2n$-dimensional product space $A \times V^*$. The minimizer x^* is given by the point $(x^*, 0)$ of intersection of M with the horizontal subspace $H_0 = \{(x, 0) \,|\, x \in A\}$ of $A \times V^*$.

Let us interpret the Newton algorithm in these terms. At the point x_k, we construct a quadratic approximation $q_k(x)$ of f. In the same way as the gradient map ∇f the gradient map ∇q_k has a graph $M_k \subset A \times V^*$. However, since $q_k(x)$ is a quadratic function, its gradient will be *linear*, and hence M_k is actually even a *subspace* of $A \times V^*$. Since the gradient and the Hessian of $q_k(x)$ coincide with those of $f(x)$ at x_k, this subspace is the *tangent space* to M at the point $(x_k, f'(x_k))$. The progress made by the Newton step hence depends on how well the tangent subspace M_k approximates the non-linear submanifold M (see Fig. 1).

11.2 Self-concordant functions

In this section we introduce an affinely invariant class of convex functions which is well suited for minimization by the Newton method. The proofs of most of the statements can be found in [7].

In order to be able to make assertions about the behaviour of the Newton algorithm, we must ensure that the norm defined by the Hessian f'' does not change too much when passing from the current iterate to the next. This corresponds to a Lipschitz condition on f''. On the other hand, in an affinely invariant framework any such changes can be compared only against the step length measured in the norm defined by the current iterate.

In other words, the change of the Hessian in the neighbourhood of some point z must be measured in and compared against the local norm at z. A natural condition is hence

$$\limsup_{x,y \to z} \frac{||f''(x) - f''(y)||_\infty}{||x - y||_z} \leq 2 \quad \forall\ z \in D, \tag{1}$$

where the spectral norm is also defined with respect to $||\cdot||_z$,

$$||A||_\infty = \max_{||u||_z = 1} u^T A u.$$

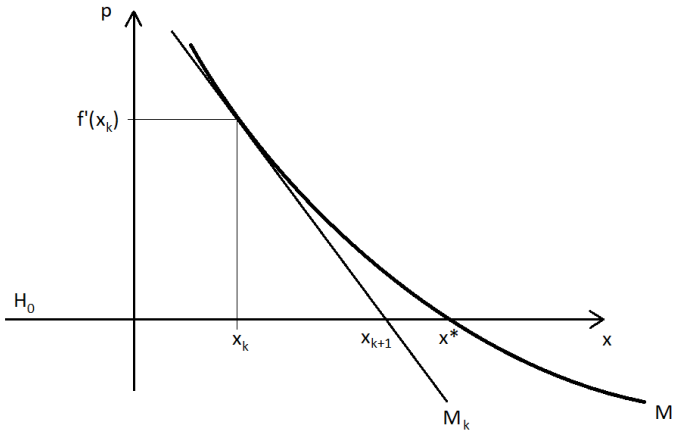

Figure 1: Geometric interpretation of the Newton step

The constant 2 is introduced for computational convenience. Changing this constant is equivalent to considering an appropriate multiple of f.

For C^3 functions this leads to the following definition.

Definition 11.2. A convex C^3 function $f : D \to \mathbb{R}$ on a domain D in some affine space is called *self-concordant* if it satisfies the inequality

$$|f'''(x)[h,h,h]| \leq 2(f''(x)[h,h])^{3/2}$$

for all $x \in D$ and all tangent vectors h.

It is called *strongly* self-concordant if in addition $\lim_{x\to\partial D} f(x) = +\infty$.

Here the derivatives of f are as above treated as multi-linear maps on the space of tangent vectors, i.e.,

$$f''(x)[h,h] = \sum_{i,j=1}^{n} \frac{\partial^2 f(x)}{\partial x_i \partial x_j} h_i h_j, \qquad f'''(x)[h,h,h] = \sum_{i,j,k=1}^{n} \frac{\partial^3 f(x)}{\partial x_i \partial x_j \partial x_k} h_i h_j h_k,$$

where n is the dimension of the space. The exponent $\frac{3}{2}$ on the right-hand side guarantees the same degree of homogeneity in h on both sides of the inequality.

Remark 11.3. Just as the set of C^1 functions with derivative bounded by L is not closed in the C^0 norm, the set of self-concordant functions is not closed in the C^2 norm. While the closure of the functions with bounded derivative are the L-Lipschitz continuous functions, the closure of the set of self-concordant functions are those satisfying condition (1). This condition is hence "almost equivalent" to self-concordance. The theory of interior-point methods actually does not rely on the barriers being C^3 and is valid also for the slightly larger class of C^2 functions satisfying (1).

The limit condition means that f tends to $+\infty$ if evaluated at any sequence of points in D which tend to a boundary point of D. This condition ensures that the level sets $\{x \in D \,|\, f(x) \leq c\}$ are closed for all constants $c \in \mathbb{R}$.

Note that if $f''(x_0)[h,h] = 0$ for some point $x_0 \in D$ and some vector h, then the self-concordance condition implies $f'''(x_0)[\cdot,h,h] = 0$. It follows that $f''(x)[h,h] \equiv 0$ for all $x \in D$, and f is actually linear in the direction h. But then strong self-concordance implies that D is the direct product of a line in the direction h and a transversal factor, while f is the sum of a convex function on this factor and a linear function along the direction h. We hence assume without loss of generality that $f'' \succ 0$ everywhere on D.

Self-concordance is left invariant by a number of operations.

- $f(x)$ is self-concordant $\Rightarrow$ $g(x) = f(Ax + b)$ is self-concordant
- f is self-concordant $\Rightarrow$ the sum $f + l$ is self-concordant for l linear

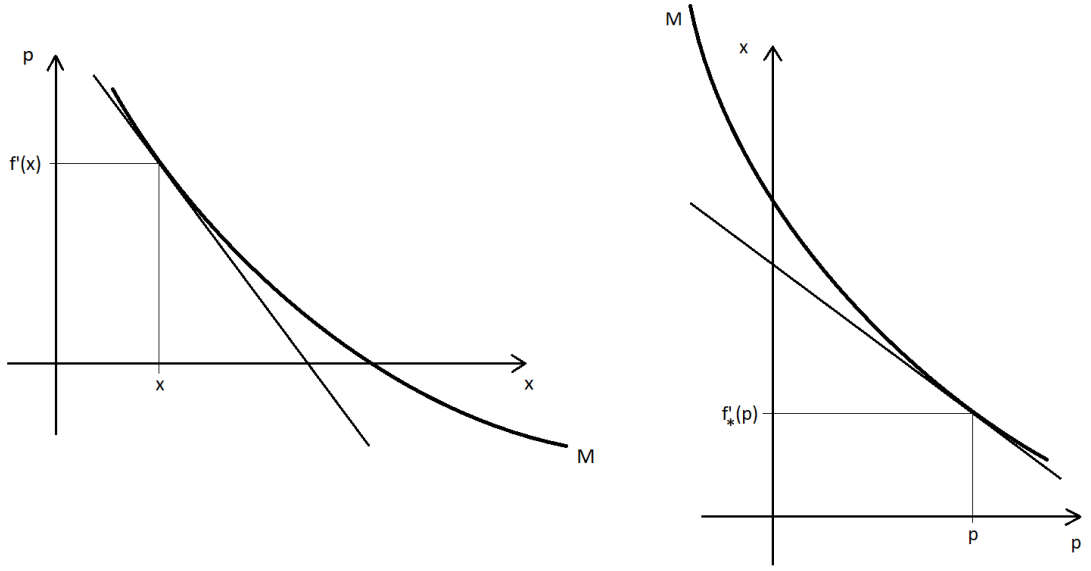

Figure 2: The gradient graphs of a function and its Legendre dual are essentially the same manifold.

- f is self-concordant, L an affine subspace $\Rightarrow$ the restriction $f|_L$ is self-concordant
- f is self-concordant $\Rightarrow$ the multiple αf is self-concordant for $\alpha \geq 1$
- f, g are self-concordant $\Rightarrow$ the sum $f + g$ is self-concordant
- f on D_1, g on D_2 are self-concordant $\Rightarrow$ $h(x, y) = f(x) + g(y)$ is self-concordant on $D = D_1 \times D_2$
- f is self-concordant $\Rightarrow$ the Legendre dual f_* is self-concordant

Here the Legendre dual of the function f is given by

$$f_*(p) = \sup_{x \in D} \langle p, x \rangle - f(x).$$

The graphs of the gradients f' and f'_* are closely related. They are essentially obtained from each other by exchange of argument and value (see Fig. 2).

Let us return to the geometric interpretation of the Newton method given above. The approximation of the function f by the quadratic Taylor polynomial q_k corresponds to the approximation of the non-linear graph M of the gradient f' by the tangent plane M_k. The quality of approximation of a sub-manifold in the neighbourhood of some point by its tangent subspace at that point is given by the *curvature* of the sub-manifold. The self-concordance condition then states that the curvature of M is uniformly bounded by the constant 1.

We shall make the notion of curvature more precise. The ambient space $A \times V^*$ where M resides possesses an indefinite non-degenerate scalar product. Namely, two tangent vectors $(u_x, u_p), (v_x, v_p) \in V \times V^*$ can be multiplied by the formula

$$\langle (u_x, u_p), (v_x, v_p) \rangle = \frac{1}{2}(\langle u_x, v_p \rangle + \langle v_x, u_p \rangle).$$

Here on the right-hand side the brackets denote the usual dual pairing between vectors and co-vectors.

Let now $\sigma_0 = (x_0, f'(x_0))$ be a point and $\sigma(t) = (x(t), f'(x(t)))$ a C^2 curve on M such that $\sigma(0) = \sigma_0$. Then the first derivative $u = \dot{\sigma}(0)$ is tangent to M at σ_0, while the second derivative $\ddot{\sigma}(0)$ can be decomposed into an orthogonal sum of a component $\ddot{\sigma}_t$ tangent to M and a component $\ddot{\sigma}_n$ normal to M. The *curvature* of M in the direction u is then defined as the quotient $\frac{||\ddot{\sigma}_n||}{||\dot{\sigma}||^2}$.

11.3 Newton method on self-concordant functions

We shall now analyze how the Newton method behaves on a strongly self-concordant function. The following result guarantees that the Newton algorithm can safely make steps of finite length, where "safely" means that the next iterate stays in the domain D and features a lower decrement than the previous one.

Lemma 11.4. *Let $f : D \to \mathbb{R}$ be a strongly self-concordant function. Then for every $x_0 \in D$ the open* Dikin ellipsoid

$$E_{x_0} = \{x \,|\, ||x - x_0||_{x_0} < 1\}$$

around x_0 is contained in the domain D.

It follows that if the damping coefficient γ_k at each step is smaller than the Newton decrement ρ_k, then the sequence of iterates will never leave the domain D. In particular, for ρ_k the full Newton step is guaranteed to stay in D.

The lemma follows immediately from the following estimate on the Hessian $f''(x)$ in terms of $f''(x_0)$.

Lemma 11.5. *Let $f : D \to \mathbb{R}$ be a strongly self-concordant function. Then for every $x_0 \in D$ and $x \in E_{x_0}$ we have*

$$(1 - ||x - x_k||_{x_k})^2 f''(x_k) \preceq f''(x) \preceq (1 - ||x - x_k||_{x_k})^{-2} f''(x_k).$$

In order for the Newton method to make progress we have also to assure that the Newton decrement decreases at each step. Here we have different guarantees on the decrease depending on the damping coefficient [7, 6], see Fig. 3.

Lemma 11.6. *Let f be a strongly self-concordant function, x_k the current iterate and $\rho_k < 1$ the Newton decrement at x_k. Let x_{k+1} be the next iterate after a Newton step with damping coefficient γ_k. Then the Newton decrement ρ_{k+1} at the next iterate is upper bounded by*

$$\rho_{k+1} \leq \left(\frac{\rho_k}{1 - \rho_k}\right)^2$$

for $\gamma_k = 1$;

$$\rho_{k+1} \leq \frac{\rho_k^2(2 + \rho_k)}{1 + \rho_k}$$

for $\gamma_k = \frac{1}{1+\rho_k}$;

$$\rho_{k+1} \leq \rho_k^2 \left(1 + \rho_k + \frac{\rho_k}{1 + \rho_k + \rho_k^2}\right)$$

for $\gamma_k = \frac{1+\rho_k}{1+\rho_k+\rho_k^2}$.

The bound on the Hessian in Lemma 11.5 does not distinguish between the direction of the gradient and orthogonal direction. A more thorough analysis allows to obtain an optimal bound on ρ_{k+1} for given γ_k. Minimizing this bound with respect to γ_k yields the optimal damping coefficient as a function of ρ_k. The optimal bounds cannot be expressed analytically, but can be computed numerically (see Fig. 4).

11.4 Self-concordant barriers

In order for the algorithms to have a linear convergence rate the barrier functions must satisfy an additional property.

Definition 11.7. A strongly self-concordant function $f : D \to \mathbb{R}$ is called a *self-concordant barrier* with parameter ν if it satisfies the condition $||f'(x)||_x^2 \leq \nu$ for all $x \in D$.

We have the following properties:

- $f(x)$ has parameter $\nu \Rightarrow g(x) = f(Ax + b)$ has parameter ν
- f has parameter ν, L an affine subspace $\Rightarrow f|_L$ has parameter ν

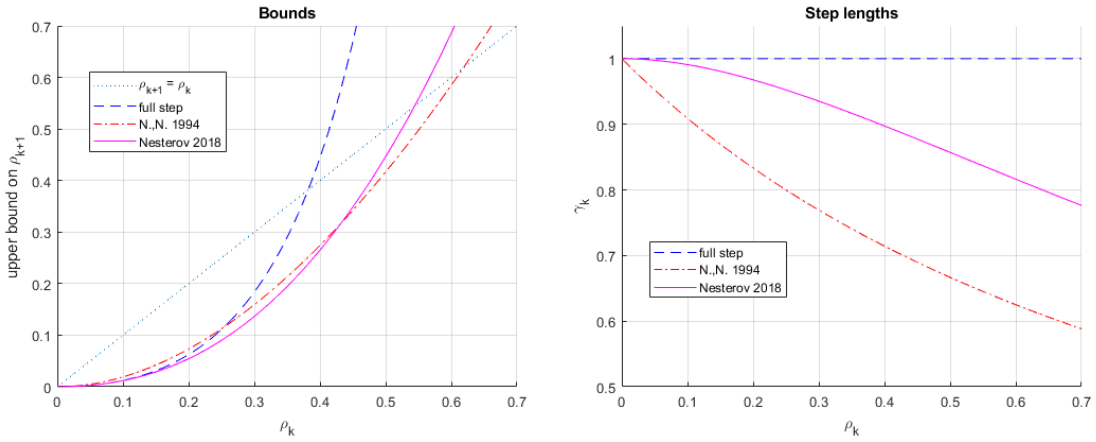

Figure 3: Upper bounds on the decrement at the next iterate (left) and damping coefficients (right).

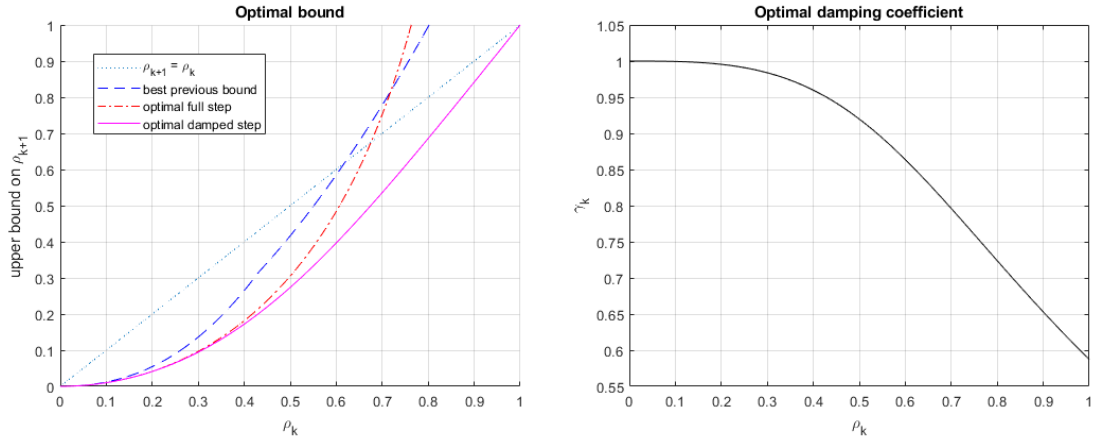

Figure 4: Optimal upper bounds on the decrement (left) and optimal damping coefficient (right).

- f has parameter $\nu \Rightarrow \alpha f$ has parameter $\alpha\nu$ for $\alpha \geq 1$
- f, g have parameter $\nu_1, \nu_2 \Rightarrow f + g$ has parameter $\nu_1 + \nu_2$
- f on D_1 has parameter ν_1, g on D_2 has parameter $\nu_2 \Rightarrow h(x, y) = f(x) + g(y)$ on $D = D_1 \times D_2$ has parameter $\nu_1 + \nu_2$

These properties easily follow from the following characterization. The parameter of a self-concordant barrier is not exceeding ν if and only if

$$\begin{pmatrix} \nu & (f')^T \\ f' & f'' \end{pmatrix} \succeq 0$$

everywhere on D.

In conic optimization the domain is a cone, which has an additional structure, namely invariance with respect to homotheties. It is natural to demand this invariance for the barrier function F too. More precisely, we shall demand that the function F is modified by an additive constant under homotheties. This leads to the following definition.

Definition 11.8. Let $K \subset \mathbb{R}^n$ be a regular convex cone. A strongly self-concordant function $F : K^o \to \mathbb{R}$ is called a *logarithmically homogeneous self-concordant barrier* with *barrier parameter* ν if it is in addition logarithmically homogeneous of degree $-\nu$, i.e.,

$$F(\alpha x) = -\nu \log \alpha + F(x) \tag{2}$$

for all $x \in K^o$ and $\alpha > 0$.

It turns out that logarithmic homogeneity uniformly bounds the Newton decrement, and this definition is compatible with Definition 11.7 above. Namely, the decrement of a logarithmically homogeneous barrier with parameter ν is equal to the constant $\sqrt{\nu}$. Indeed, differentiating (2) with respect to x we obtain $\alpha F'(\alpha x) = F'(x)$. Differentiating this and (2) with respect to α at $\alpha = 1$, we obtain

$$F'(x) + F''(x) \cdot x = 0, \qquad \langle F'(x), x \rangle = -\nu.$$

It follows that $(F''(x))^{-1} F'(x) = -x$ and hence $(F'(x))^T (F''(x))^{-1} F'(x) = -\langle F'(x), x \rangle = \nu$.

It is desirable to have barriers with parameter as low as possible at our disposal, because a lower value of the parameter corresponds to faster convergence. Actually, the tractability of a conic program is essentially determined by the availability of a computable self-concordant barrier with sufficiently low parameter on the underlying cone. We now give concrete examples of such barriers.

Symmetric cones: For symmetric cones a barrier is available via the Jordan algebra structure, namely the logarithm of the inverse of the determinant.

symmetric cone	$\mathcal{S}^n_+$, $\mathcal{H}^n_+$	$\mathbb{R}^n_+$	L_n	$K = \prod_j K_j$
barrier	$-\log\det A$	$-\sum_{j=1}^n \log x_j$	$-\log(x_0^2 - x_1^2 - \cdots - x_{n-1}^2)$	$\sum_j F_j$
parameter ν	n	n	2	$\sum_j \nu_j$

The barrier parameter of these barriers is optimal. These barriers possess an additional advantageous property, they are *self-scaled*. This property allows for especially efficient interior-point methods for solving symmetric cone programs [3],[4].

***p*-norm cone:** Consider the cone

$$K_p = \{(x_0, x) \in \mathbb{R} \times \mathbb{R}^n \,|\, x_0 \geq ||x||_p\}.$$

This cone allows to formulate constraints on the p-norm of linear combinations of the decision variables of the conic program. For non-rational p this is a transcendental cone. We shall represent K_p as a linear projection of another cone

$$\hat{K}_p = \{(x_0, x, y) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \,|\, y \geq 0,\ \sum_i y_i = x_0,\ y_i^{1/p} x_0^{1-1/p} \geq |x_i|\}.$$

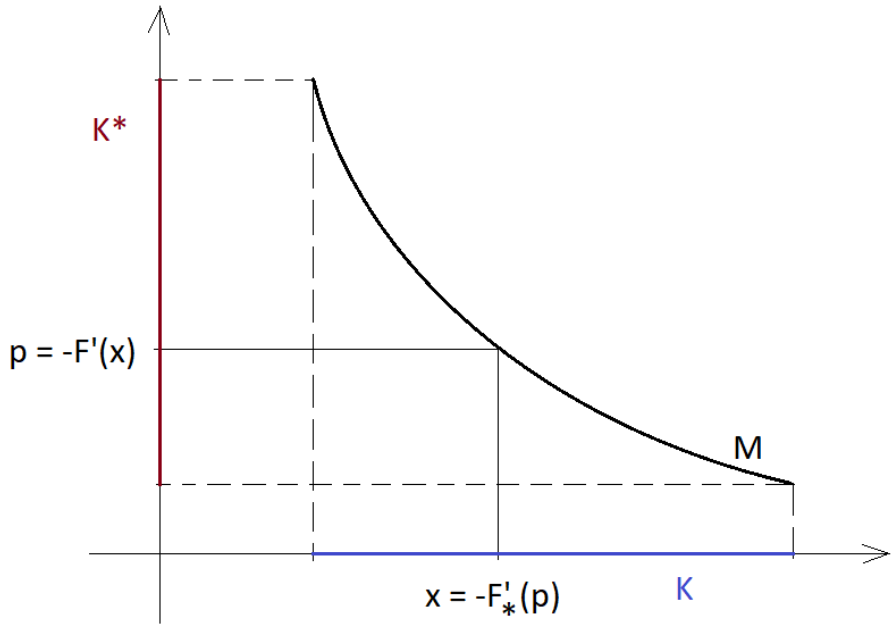

Figure 5: The non-linear submanifold M is the graph of $-F'$ or $-F'_*$, depending on the interpretation.

Indeed, $(x_0, x) \in K_p$ if and only if there exists y such that $(x_0, x, y) \in \hat{K}_p$. The function

$$\Phi(x_0, x, y) = -\sum_i \Big(\log(y_i^{2/p} x_0^{2-2/p} - x_i^2) + \log y_i + \log x_0\Big)$$

is a self-concordant barrier on $\hat{K}_p$ with parameter $\nu = 4n$. Details can be looked up in [5].

Exponential cone: Consider the cone

$$K_{\exp} = \{(x, y, 0) \,|\, x \le 0,\ y \ge 0\} \cup \{(x, y, z) \,|\, z > 0,\ y \ge z e^{x/z}\},$$

encountered in geometric programming. On this cone we have the barrier

$$F(x, y, z) = -\log(z \log \frac{y}{z} - x) - \log y - \log z$$

with parameter $\nu = 3$. This parameter value is optimal.

11.5 Legendre duality

Every barrier $F : \operatorname{int} K \to \mathbb{R}$ on a regular convex cone K with parameter ν generates a barrier $F_* : \operatorname{int} K^* \to \mathbb{R}$ on the dual cone with the same parameter. This *dual barrier* is given by the *Legendre transform*

$$F_*(p) = \sup_{x \in K} (-\langle p, x\rangle - F(x)).$$

The supremum is achieved at the point $x \in K^o$ satisfying $F'(x) = -p$, which always exists if $p \in \operatorname{int} K^*$. The map $\mathcal{L} : x \mapsto -F'(x)$ is a bijection between the interiors of K and K^*. Actually, the graph of $-F'$ can be considered also as the graph of $-F'_*$ if the roles of x and p are switched (see Fig. 5).

If we consider these interiors as Riemannian manifolds equipped with the Hessians $F''(x)$ and $F''_*(p)$ as metrics, respectively, then $\mathcal{L}$ is an isometry.

If $K = K^*$ is a symmetric cone and $F(x) = -\log\det x$, then $F_* = F$ and the map $\mathcal{L}$ is the inversion $x \mapsto x^{-1}$ in the Jordan algebra underlying the cone.

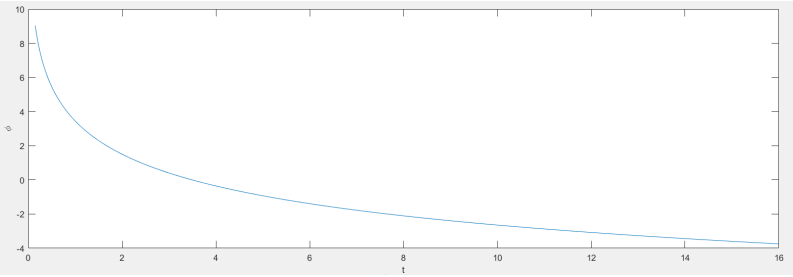

Figure 6: Graph of $\phi(t)$ for the canonical barrier of $K_{\exp}$

11.6 Universal barriers

The availability of a computationally tractable logarithmically homogeneous self-concordant barrier on a convex cone enables us to solve conic programs over this cone. Naturally the question arises whether barriers exist on arbitrary cones at all. The answer to this question is positive, and we shall learn to know two constructions of such barriers.

Universal barrier. For a regular convex cone $K \subset \mathbb{R}^n$, consider its *characteristic function*

$$\varphi(x) = \int_{K^*} e^{-\langle x,y\rangle}\, dy$$

for x in the interior of K. Then the function $F(x) = \log \varphi(x)$ is a logarithmically homogeneous self-concordant barrier on K with barrier parameter $\nu = n$, the *universal barrier*. It has been introduced in [7] with a bound $O(n)$ on the parameter, the actual value of the parameter has been established in [1],[2].

The universal barrier is given by a multi-dimensional integral over the dual cone and is difficult to compute even for relatively simple non-homogeneous cones. It also suffers from the draw-back that its Legendre dual is in general not the universal barrier for K^*.

Entropic barrier. This is the dual barrier to the universal barrier. Its parameter also equals the dimension of the cone. For details see [1].

Canonical barrier. The construction of this barrier relies on a deep result in the theory of partial differential equations.

Theorem 11.9. *Let $D \subset \mathbb{R}^n$ be a convex domain containing no line. Then there exists a unique smooth solution $F : D \to \mathbb{R}$ with positive definite Hessian of the PDE $\log\det F'' = 2F$ with boundary condition $\lim_{x\to\partial D} F(x) = +\infty$.*

If the domain D is a regular convex cone, then this solution can be proven to be a logarithmically homogeneous self-concordant barrier with barrier parameter $\nu = n$, the *canonical barrier*. Its Legendre dual can be proven to coincide with the canonical barrier on K^*. It is also difficult to compute, but for a few non-homogeneous cones explicit expressions are available.

Example: For the 3-dimensional cone $K_{\exp}$ the canonical barrier is given by

$$F_{can}(x,y,z) = -\log y - 2\log z + \phi(\log\frac{y}{z} - \frac{x}{z}),$$

where the scalar function $\phi : \mathbb{R}_{++} \to \mathbb{R}$ is given implicitly by the curve

$$\left\{ \begin{pmatrix} t \\ \phi \end{pmatrix} = \frac{1}{2} \begin{pmatrix} \log(1+\kappa) + 2\kappa \\ \log(1+\kappa) - 3\log\kappa \end{pmatrix} \;\middle|\; \kappa \in \mathbb{R}_{++} \right\}.$$

For homogeneous cones the three barriers coincide, and for symmetric cones they are proportional to the self-scaled barrier seen above.

12 Interior-point methods

We shall now consider a class of efficient methods for solving conic programs. Interior-point methods are iterative methods which generate sequences of point in the *interior* of the underlying convex cone. This contrasts, e.g., with the simplex method, which generates a sequence of extremal points of the feasible polyhedron.

The applicability of interior-point methods is conditioned on the availability of an efficiently computable self-concordant barrier on the underlying cone. Such a barrier allows to apply *short-step path-following methods* with polynomial complexity. In some cases, in particular if the cone is symmetric, there exist barriers with additional structure which allow to speed up the convergence considerably in practice by passing to *long-step mentods*. Note that the theoretical complexity of many long-step methods is worse than that of short-step methods, although they show far superior performance on generic problem instances.

Interior-point methods can also be classified in primal or primal-dual ones, according to whether they generate sequences of points in the primal cone only or sequences of primal-dual pairs of points. In most methods the iterates exactly satisfy the linear equality constraints of the problem. This necessitates, however, a preliminary phase which looks for a feasible point or primal-dual pair. In *infeasible* methods the generated sequence does not satisfy the constraints, rather the slack is reduced from one iterate to the next simultaneously with an advance in the cost function value. Such methods do not need a preliminary phase and may start from an arbitrary interior point of the cone.

12.1 Primal short-step path-following method

We now consider the theoretically simplest version of an interior-point method, which follows the central path by making short steps. Here a step is called *short* if it its length is of order $O(1)$ in the local metric. Steps within the Dikin ellipsoid around the current iterate are hence short.

Consider the conic program

$$\min_{x \in K} \langle c, x \rangle : \quad Ax = b.$$

Here $K \subset V$ is a regular convex cone, and V a real vector space. The cost function is then defined by the dual vector $c \in V^*$. We assume that the problem is strictly feasible, i.e., the affine subspace $\mathcal{A}$ given by the linear constraints intersects the interior K^o of the cone, and that a solution $x^* \in \partial K$ exists. Suppose further that we have a barrier F on K with parameter ν at our disposal.

Let $f = F|_{\mathcal{A}}$ be the restriction of the barrier F to the affine subspace $\mathcal{A}$. Then f is a barrier on the domain $D = \mathcal{A} \cap K^o$ with parameter ν. Let also $\mathcal{L} = \{x \mid Ax = 0\}$ be the linear subspace underlying the affine subspace $\mathcal{A}$. Then the gradient of f is an element of the dual subspace $\mathcal{L}^*$. Let further $u \in \mathcal{L}^*$ be the gradient of the restriction of the cost function $\langle c, x \rangle$ to $\mathcal{A}$. With some abuse of notation we shall denote this restriction by $\langle u, x \rangle$.

The barrier F on the cone K serves to get rid of the non-linear conic constraint by replacing the original linear cost function by a composite one, namely a weighted sum $\tau \cdot \langle c, x \rangle + F(x)$ of the original cost and the barrier. We hence consider the auxiliary problem

$$\min_{x \in \mathcal{A}} \tau \cdot \langle u, x \rangle + f(x), \tag{3}$$

parameterized by a real parameter $\tau \geq 0$. Note that the composite objective function $f_\tau(x) = \tau \cdot \langle u, x \rangle + f(x)$ is still strongly self-concordant.

In the absence of degeneracy the auxiliary problem has a unique solution $x^*(\tau)$ for any large enough τ. Moreover, the solution $x^*(\tau)$ is differentiable with respect to τ and the locus of these solutions is a C^1 curve, the so-called *central path*. The minimum $x^*(\tau)$ of f_τ is characterized by the condition $f'_\tau = 0$, or equivalently $f' = -\tau u$.

For $\tau \to +\infty$ the central path tends to the solution $x^* = x^*(+\infty)$ of the original problem. If the solution of the original problem is not unique, then the limit of the central path lies in the relative interior of the set of solutions. The point $x^*(0)$ is called the *analytic center* of the feasible set of the original problem. It exists if and only if the feasible set $K \cap \mathcal{A}$ is bounded.

Let us describe the short-step path-following method. It generates a sequence of iterates $x_k \in \mathcal{A}$ which lie in the vicinity of the central path and at the same time progress along this path to higher values of the parameter τ. Let us denote the Newton decrement of the function f_τ at x_k by $\rho_k(\tau)$. To each x_k is associated a value τ_k of

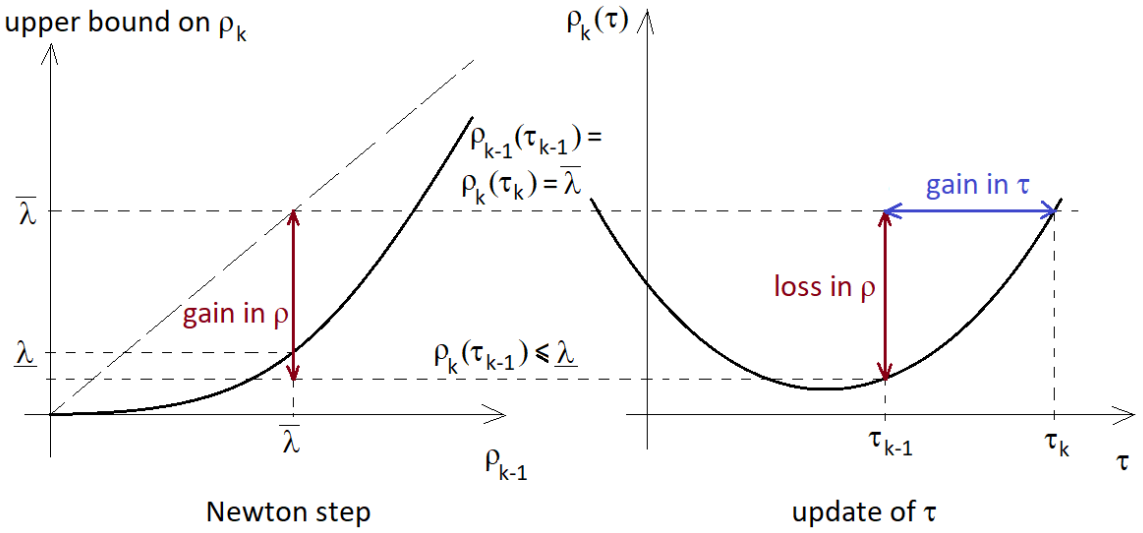

Figure 7: Choice of the next parameter value τ_k.

the parameter τ, corresponding to a *target point* $x^*(\tau_k)$ on the central path. The next iterate x_{k+1} is generated by a Newton step towards the target point, i.e.,

$$x_{k+1} = x_k - \gamma_k (f''_{\tau_k}(x_k))^{-1} f'_{\tau_k}(x_k).$$

Recall that if the Newton decrement $\rho_k(\tau_k)$ is not exceeding a certain small enough value $\overline{\lambda}$, then after the Newton step the decrement $\rho_{k+1}(\tau_k)$ is upper bounded by some value $\underline{\lambda} < \overline{\lambda}$, depending on $\overline{\lambda}$ and the damping coefficient γ_k. Based, e.g., on the estimate $\rho_{k+1} \leq \left(\frac{\rho_k}{1-\rho_k}\right)^2$ for $\gamma_k = 1$, we may choose $\overline{\lambda} = \frac{1}{4}$, $\underline{\lambda} = \frac{1}{9}$ and make full Newton steps.

The points x_k and values τ_k are chosen such that for all $k > 0$ the condition $\rho_k(\tau_{k-1}) \leq \underline{\lambda}$ holds. In order to assure this condition at the next iterate, i.e., $\rho_{k+1}(\tau_k) \leq \underline{\lambda}$, we have to ensure that $\rho_k(\tau_k) \leq \overline{\lambda}$. This gives us the update rule for the parameter τ.

Consider the dependence of the decrement $\rho_k(\tau) = \sqrt{(f'_\tau(x_k))^T (f''_\tau(x_k))^{-1} f'_\tau(x_k)}$ on the parameter τ. We have $f''_\tau = f''$, $f'_\tau = f' + \tau u$, and the function under the square root is a quadratic function in τ with leading coefficient $u^T (f''(x_k))^{-1} u$. Since $\rho_k(\tau_{k-1}) \leq \underline{\lambda}$, we have that the equation $\rho_k(\tau) = \overline{\lambda}$ has two real roots. The next parameter value τ_k is then set to the larger one of these two roots (see Fig. 7).

Let us estimate the gain in τ made during one iteration. We have $|\frac{d\rho_k(\tau)}{d\tau}| \leq \sqrt{u^T (f''(x_k))^{-1} u} = ||u||_{x_k}$. Therefore

$$\tau_k - \tau_{k-1} \geq \frac{\rho_k(\tau_k) - \rho_k(\tau_{k-1})}{||u||_{x_k}} \geq \frac{\overline{\lambda} - \underline{\lambda}}{||u||_{x_k}}.$$

Let us estimate $||u||_{x_k}$. Denote $\delta_k = f'(x_k) + \tau_k u$ and note that $||\delta_k||_{x_k} = \rho_k(\tau_k) = \overline{\lambda}$. We obtain

$$||u||_{x_k} = \tau_k^{-1} || - f'(x_k) + \delta_k||_{x_k} \leq \tau_k^{-1} (||f'(x_k)||_{x_k} + \overline{\lambda}).$$

Finally this yields

$$\log \tau_k - \log \tau_{k-1} \geq \frac{\tau_k - \tau_{k-1}}{\tau_k} \geq \frac{\overline{\lambda} - \underline{\lambda}}{||f'(x_k)||_{x_k} + \overline{\lambda}} \geq \frac{\overline{\lambda} - \underline{\lambda}}{\sqrt{\nu} + \overline{\lambda}}.$$

Thus at each iteration the quantity $\log \tau$ increases by an amount of order $O(\nu^{-1/2})$, and the number of iterations to reach a given threshold for τ is proportional to $\sqrt{\nu}$.

The larger the parameter ν, the more iterations we need to reach a given precision and the slower the algorithm will converge. Therefore it is of interest to have barriers with a parameter as low as possible.

In short-step methods the parameter τ of the target point is multiplied by a factor of $1 + O(\nu^{-1/2})$ at each iteration. The iterates stay in a narrow neighbourhood of the central path, such that the Dikin ellipsoid of

radius $\underline{\lambda}$ around each iterate has a non-empty intersection with the central path. The sequence of iterates hence closely follows the central path and reproduces all its turns.

12.2 Long-step methods

Long-step methods update the parameter τ more aggressively, multiplying it by a quantity $\theta > 1$. In this case the first Newton step towards the updated target point carries the iterate out of a narrow neighbourhood of the central path. Hence one has either to choose a large neighbourhood, or to perform several Newton steps to return in the vicinity of the central path. In the first case additional structure is necessary in order to measure the distance to the central path by a means other than the decrement. We shall consider such methods for symmetric cone programming below. In the second case one can show that in order to return to the central path an number of $O(\nu)$ Newton steps is necessary. In this case the theoretical complexity increases to $O(\nu)$ iterates to reach a given precision.

12.3 Primal-dual methods on symmetric cones

In practice the most successful algorithms for solving conic programs are specialised algorithms for the solution of *symmetric* cone programs, including LP, SOCP, and SDP. The reason for this is the rich structure of the symmetric cones, which allows for the existence of barriers with additional properties, so-called *self-scaled* barriers.

Consider a primal-dual pair of conic programs. In the primal program

$$\min_{x \in K} \langle c, x \rangle : \quad Ax = b$$

the linear equality constraints determine the affine subspace $\mathcal{P} = \{x \mid Ax = b\}$, while in the dual program

$$\max_{p \in K^*, s} \langle b, s \rangle : \quad A^T s + p = c$$

they determine the affine subspace $\mathcal{D} = Im\, A^T + c$. The linear subspaces $L = \ker A$, $L^\perp = Im\, A^T$ underlying the affine subspaces $\mathcal{P}$, $\mathcal{D}$ are complementary.

At the point $x^*(\tau)$ of the primal central path we have $F' + \tau c \in L^\perp$. Hence

$$-F' \in \tau c + L^\perp = \tau \cdot \mathcal{D}$$

and

$$(x^*(\tau), -F'(x^*(\tau))) \in \mathcal{P} \times (\tau \cdot \mathcal{D})).$$

It follows that the curve given by the intersection

$$M \cap (\mathcal{P} \times (\mathbb{R} \cdot \mathcal{D}))$$

projects to the central path by its primal component. Here $M = \{(x, -F'(x)) \mid x \in \operatorname{int} K\}$ is the graph of $-F'$.

Likewise, the dual central path is the projection of the curve given by the intersection

$$M \cap ((\mathbb{R} \cdot \mathcal{P}) \times \mathcal{D})$$

on the dual component.

If $(x, p) \in M$, then $(\alpha x, \alpha^{-1} p) \in M$ for all $\alpha \in \mathbb{R}_{++}$. Let $x = x^*(\tau)$ be a point on the primal central path corresponding to some value τ of the parameter. Then $(x, \tau p) \in M$, where $p = -\tau^{-1} F'(x)$. But then also $(\tau x, p) \in M$. This means that p is the point of the dual central path corresponding to the same value τ. The *primal-dual central path*, which consists of the pairs $(x^*(\tau), p^*(\tau))$ and projects to both the primal and dual central path, is then given by the set of points $(x, p) \in \mathcal{P} \times \mathcal{D}$ such that $(\sqrt{\tau} x, \sqrt{\tau} p) \in M$ for some τ, or by the intersection

$$(\mathbb{R} \cdot M) \cap (\mathcal{P} \times \mathcal{D}).$$

The equations of the primal-dual central path are non-linear and can be written in different ways, e.g., $p = -\mu F'(x)$ or $x = -\mu F'_*(p)$, where $\mu = \tau^{-1}$. Hence there are different linearizations and corresponding *search*

directions which bring the iterate closer to a target point on the central path. Equivalently, there are different ways to approximate the non-linear graph M of $-F'$ by an affine subspace.

A barrier F on a cone $K \subset V$ with parameter ν is called self-scaled if for every pair $(x,p) \in \operatorname{int} K \times \operatorname{int} K^*$ there exists a unique so-called *scaling point* $w \in \operatorname{int} K$, which is defined as the minimizer of the function $\Phi(z) = \langle F'(z), x\rangle - \langle z, p\rangle$. In addition, it satisfies the relation

$$F_*(p) = F(x) - 2F(w) - \nu.$$

To the scaling point one may put in correspondence a dual point $w_* = -F'(w)$, which minimizes the function $\Xi(t) = \langle F_*'(t), p\rangle - \langle x, t\rangle$.

The pair $(w, w_*) \in M$ has a simple geometric interpretation. As we have seen above, on the primal-dual product space $V \times V^*$ there exists a pseudo-scalar product defined by

$$\langle (u,v), (u',v')\rangle := \frac{\langle u, v'\rangle + \langle u', v\rangle}{2}.$$

This scalar product generates an indefinite metric with squared distance

$$d^2((u,v),(u',v')) = \langle (u-u', v-v'), (u-u', v-v')\rangle = \langle u-u', v-v'\rangle.$$

Note that this quantity may be positive or negative.

Let $\alpha > 0$ be given. Then the point $\alpha(w, w_*) \in \alpha M$ is the closest point on αM to (x,p) in the above metric. Indeed, the squared distance from (x,p) to $\alpha(z, -F'(z)) \in \alpha M$ is given by

$$\langle x - \alpha z, p + \alpha F'(z)\rangle.$$

The expression $\langle F'(z), z\rangle = -\nu$ is constant on M by logarithmic homogeneity of F, and the quantity $\langle x, p\rangle$ is also independent of z. Therefore the squared distance equals $\Phi(z)$ up to an additive constant, and its extremum is given by the scaling point. Note that the scaling point does not depend on the value of α.

In LP the scaling point corresponding to a primal-dual pair (x,p) of positive vectors is given by the vector $w = \sqrt{\frac{x}{p}}$, where the ratio and the square root are taken element-wise. In SDP, let (X,P) be a primal-dual pair of positive definite matrices, and let U be the orthogonal matrix which simulatenously diagonalizes X and P, i.e., $X = U\Lambda_X U^T$, $P = U\Lambda_P U^T$ with diagonal matrices Λ_X, Λ_P. Then the scaling point is given by $W = U\Lambda_X^{1/2}\Lambda_P^{-1/2}U^T$. There exists also the more explicit expression $W = X^{1/2}(X^{1/2}PX^{1/2})^{-1/2}X^{1/2}$. The most costly operation in primal-dual long-step methods for solving SDPs is the computation of the scaling point. It can be computed in $O(n^4)$ arithmetic operations, where n is the matrix size [8].

The scaling point is used to construct a linear approximation of the non-linear gradient graph M in the vicinity of the current iterate (x,p). In general this iterate does not lie on M, and it is natural to employ the tangent plane T_w to M at the closest point to (x,p), which is the primal-dual scaling point pair (w, w_*). However, in the case of symmetric cones this tangent plane is not used directly. One rather uses the affine plane M_k which is parallel to T_w and passes through the points $(x, -F'(x)), (-F_*(p), p) \in M$ (see Fig. 8). Such a plane exists due to the additional properties of the scaling point for symmetric cones.

The plane M_k is used to construct an approximation of the primal-dual central path by the intersection $(\mathbb{R} \cdot M_k) \cap (\mathcal{P} \times \mathcal{D})$. This approximation as a straight line in the product space $V \times V^*$. The direction of this line is called the (Nesterov-Todd or NT) *affine scaling direction*. The orthogonal projection of (x,p) on the line defines the *centering direction*. Their linear combinations are the *Nesterov-Todd search directions*. An advance along the affine scaling direction corresponds to a movement parallel to the central path. It improves the cost function values of the primal and the dual problem. An advance along the centering direction brings the current primal-dual pair closer to the central path without attempting to improve the function values.

Short-step methods move along the search directions towards a concrete target point on the primal-dual central path. In contrast, long-step methods make steps along the affine scaling direction until the boundary of a "large" neighbourhood of the central path is hit. In so-called *predictor-corrector* methods, after a long "predictor" step along the affine scaling direction one or several "corrector" steps along the centering direction are made until the iterates are again close enough to the central path. There exist also higher-order methods which approximate the central path by a polynomial instead of a line. The cost to compute the higher-order corrections is smaller because factorizations from the computation of the affine scaling direction can be re-used.

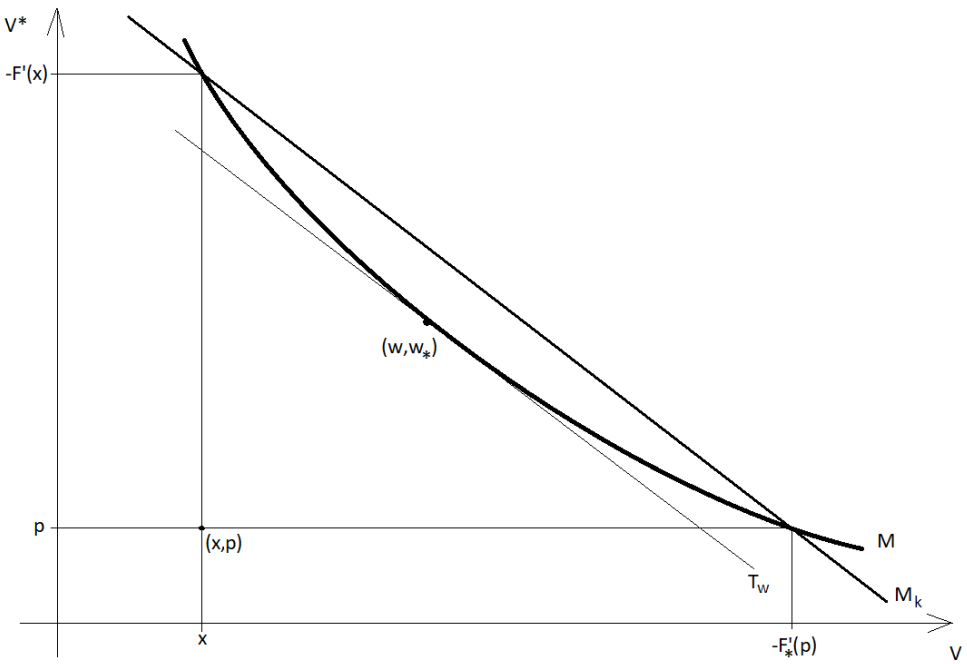

Figure 8: Affine approximation M_k of the gradient graph M.

In LP, the "large" neighbourhood of the central path is defined as follows. For a primal-dual pair (x, p) of points, form the component-wise product $y = x \cdot p$. The central path consists of those pairs for which $y = \mu \cdot \mathbf{1}$, with $\mathbf{1}$ the all-ones vector. Actually, every primal-dual strictly feasible pair of points corresponds to exactly one value of $y \in \mathbb{R}^n_{++}$. The "large" neighbourhood of the central path is then defined by a bound on the condition number of y, $N_\gamma = \{(x, p) \in \mathcal{P} \times \mathcal{D} \in \frac{\max_j y_j}{\min_j y_j} \leq \gamma\}$. Typically one chooses $\gamma \sim 10^3$.

In SDP the role of the vector y is played by the spectrum of the matrix product (X, P). Hence the "large" neighbourhood is defined by $N(\gamma) = \left\{(X, P) \in \mathcal{P} \times \mathcal{D} \mid \frac{\lambda_{\max}(XP)}{\lambda_{\min}(XP)} \leq \gamma\right\}$.

The number of iterations which is necessary in practice to achieve a given accuracy weakly depends on the dimension of the problem and is given by several dozens even for large problem dimensions.

12.4 Potential-reduction methods

Potential-reduction methods generate sequences of primal-dual pairs (x_k, p_k) of strictly feasible points which are not linked to any target points on the central path. Instead, the methods decrease the potential function

$$\Phi(x, p) = (\nu + \sqrt{\nu}) \log\langle x, p\rangle + F(x) + F_*(p),$$

which is defined on the product of the interiors of the primal and dual cones. Note that on the product of the primal and dual feasible sets the quadratic function $\langle x, p\rangle$ is actually *linear*. This follows from the fact that the vector spaces underlying the affine hulls of the primal and dual feasible sets are mutually orthogonal. For any two primal-dual feasible pairs (x, p) and (x_0, p_0) we then get $\langle x - x_0, p - p_0\rangle = 0$ and hence

$$\langle x, p\rangle = \langle x, p_0\rangle + \langle x_0, p\rangle - \langle x_0, p_0\rangle,$$

which is indeed linear in (x, p).

Therefore the potential is "almost convex", in the sense that it is convex on any hyperplane of the form $H_\mu = \{(x, p) \mid \langle x, p\rangle = \mu\}$. The decrease in potential is bounded from below by a constant $O(1)$. However, every sequence on which the potential tends to $-\infty$ tends to a pair of primal and dual solutions of the primal and dual conic programs.

References

[1] Sébastien Bubeck and Ronen Eldan. The entropic barrier: a simple and optimal universal self-concordant barrier. In *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, 2015.

[2] Yin Tat Lee and Man-Chung Yue. Universal barrier is n-self-concordant. Optimization Online 2018/09/6810, 2018.

[3] Yuri E. Nesterov and Michael J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Math. Oper. Res.*, 22:1–42, 1997.

[4] Yuri E. Nesterov and Michael J. Todd. Primal-dual interior-point methods for self-scaled cones. *SIAM J. Optimiz.*, 8(2):324–364, 1998.

[5] Yurii Nesterov. Towards non-symmetric conic optimization. *Optim. Method Softw.*, 27(5):893–917, 2012.

[6] Yurii Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, 2018.

[7] Yurii Nesterov and Arkadii Nemirovskii. *Interior-point Polynomial Algorithms in Convex Programming*, volume 13 of *SIAM Stud. Appl. Math.* SIAM, Philadelphia, 1994.

[8] Michael J. Todd, Kim-Chuan Toh, and Reha H. Tütüncü. On the Nesterov-Todd direction in semidefinite programming. *SIAM J. Optimiz.*, 8(3):769–796, 1998.