Nonlinear (local) Optimization
The State of the Art

P. Spellucci
TU Darmstadt, Dept. of Mathematics

December 13, 2000

Abstract

Nonlinear Optimization came into live about fifty years ago with the seminal paper of Kuhn and Tucker [79], very shortly only after the birth of LP praxis through Dantzig's pioneering work on the simplex algorithm. Since then it has grown to a mathematical discipline in its own right, with deep interconnections with nonlinear analysis and numerical analysis. The appearance of more and more powerful computer equipment and more and more involved numerical solution techniques made the solution of nonlinear models a routine job, which forty years ago were beyond any possible consideration. This is true at least for the solution of problems involving some hundred variables and some thousand constraints, solution meant in a local and weak sense, that means in the sense of identifying points which satisfy certain necessary optimality conditions. The research in this area came to some maturity about ten years ago and a large amount of software is now available even in the public domain [87]. During the past ten years many working groups throughout the world aimed at solution techniques also capable of coping with true large-scale nonlinear optimization problems. Several overview papers about this subject appeared in recent years, e.g. [25] and [64]. In this overview article we give a short introduction into *NLP* theory first and then review some of the most promising solution techniques. Whereas convex problems can be dealt with also in very high dimension successfully already, the treatment of nonconvex cases offers resistance to a satisfactory solution approach, since obviously methods which worked well for medium large problems cannot be transfered to very high dimensions.

1 Some theory

The general nonlinear optimization problem is

$$\textit{NLP:} \qquad \text{Minimize } f(x) \text{ with respect to } x \in \mathcal{S}\ ,$$

where the so called feasible set $\mathcal{S}$ is defined through

$$\mathcal{S} \;=\; \{x \in \mathbb{R}^n : h(x) \;=\; 0,\; g(x) \;\geq\; 0\}\ .$$

We assume here that h and g are defined everywhere on $\mathbb{R}^n$ with values in some $\mathbb{R}^E$ and $\mathbb{R}^I$ and are of class C^2 at least, although some results are valid for the class C^1 too. The nonsmooth case requires quite special treatment, which is beyond the scope of this paper. For f we assume also differentiability of class C^2 on an open superset of $\mathcal{S}$. It is important to note that the same set $\mathcal{S}$ can be represented quite differently by functions h and g. It is the representation by a given triple (f, g, h) which defines a nonlinear programming problem in the sequel.

From a practical point of view, other formulations of *NLP* are more suitable, e.g. one standard form is

$$\begin{aligned} \mathcal{S} \;=\; \{x \;\in\; \mathbb{R}^n : \qquad & x_u \;\leq\; x \;\leq\; x_o \\ & b_u \;\leq\; Ax \;\leq\; b_o \\ & c_u \;\leq\; c(x) \leq\; c_o\ \}\ , \end{aligned}$$

where x_u, x_o, b_u, b_o, c_u, c_o are given constants, possibly $\in \{-\infty, \infty\}$ or possibly of equal size (defining an equation), and A a given matrix. This formulation reflects the different numerical treatment which may be applied to different parts of the constraints. In order to obtain necessary or sufficient optimality conditions, we need further assumptions on h and g, <u>not</u> on $\mathcal{S}$. These assumptions are known as constraint qualifications. Much research took place for obtaining minimal assumptions of this kind, and indeed there is known such a, in some sense minimal one, the *Guignard* constraint qualification [69], [65]. However, finally from additional considerations it turned out that the following one, which much more lends itself to a geometric interpretation, is "the" qualification one should have, first given by Mangasarian and Fromowitz in [85]:

$$\textit{MFCQ:}\quad \nabla h(x) \quad \text{is of full rank}$$
$$\text{and}$$
$$\nabla h(x)^T z \;=\; 0,\; \nabla g_{\mathcal{A}}(x)^T z \;>\; 0 \qquad \text{is solvable}$$

where

$$\mathcal{A} \;=\; \mathcal{A}(x) \;=\; \{i \in I : g_i(x) \;\le\; 0\}$$

is the so called "active set". (The full rank condition means that the gradients of the equality constraints are linearly independent. From this it follows via the implicit function theorem that the solution set of $\{z : h(x) = h(z)\}$ is locally (around x) a smooth nonlinear manifold in $\mathbb{R}^n$.) Geometrically *MFCQ* means that at x there exists a direction z which is tangential to the nonlinear manifold defined by $\{z : h(z) = h(x)\}$ whereas all active inequality constraints can be strongly improved towards feasibility along z. From the rank condition on ∇h it follows that it is also possible to find some $\tilde{z}$ which improves feasibility with respect to h simultaneously if $h(x) \neq 0$, [113].

Observe that we have defined *MFCQ* here also for infeasible points, whereas the usual definition in the literature only takes feasible points into account. A much stronger condition is the so called regularity assumption

$$\textit{LICQ:}\quad (\nabla h(x), \nabla g_{\mathcal{A}}(x)) \text{ is of full rank .}$$

Obviously *LICQ* implies *MFCQ*. There are many problems of practical relevance, there the former is violated. *MFCQ* has some far reaching consequences:

Theorem 1.1 *Let $x^* \in \mathcal{S}$ be a local minimizer of problem NLP and MFCQ be satisfied there. Then there exists a <u>bounded set</u> $\mathcal{L}$ of multipliers $\lambda^* \in \mathbb{R}^I_+$ and for every such λ^* an unique multiplier $\mu^* \in \mathbb{R}^E$ such that there holds*

$$\begin{aligned} \textit{KTC:}\quad \nabla f(x^*) - \nabla g(x^*)\lambda^* - \nabla h(x^*)\mu^* &= 0\;, \\ g_i(x^*)\lambda_i^* &= 0 \\ \textit{with } \lambda^* \in \mathcal{L} \textit{ and } \mu^* = \mu^*(\lambda^*) & \qquad . \end{aligned}$$

Reverse, if KTC holds with a bounded set of multipliers (λ^*, μ^*), then MFCQ is satisfied. Finally, if f is convex, h affinely linear and g_i concave for all $i \in I$ (a convex optimization problem), then KTC is also sufficient for global optimality.

The proof of this can be found in [55] respectively in standard textbooks of nonlinear optimization, e.g. [96]. *MFCQ* has even more implications for a solution of *NLP*, e.g. it is the weakest known condition which guarantess exact penalizability of *NLP* based on properties of h and g alone, see e.g. [17] and also [68]. By exact penalization is meant a transformation of *NLP* into an unconstrained optimization problem, whose (local and global) solutions coincide with those of *NLP*. Such a penalization can be obtained via penalty functions of the form

$$f(x) + \gamma(||h(x)|| + ||g(x)^-||)$$

where γ is a suitably large constant and the norm is an absolute one, i.e. $||y|| \;=\; ||\,|y|\,||\;\forall\, y$. Here and in the following we use the following convention: given a vector x by x^- respectively x^+ we understand a vector with components $\min\{0, x_i\}$ resp. $\max\{0, x_i\}$.

The Mangasarian-Fromowitz condition plays also a fundamental role in questions of stability of *NLP*. E.g.

Theorem 1.2 *Consider the perturbed problem NLP(g_0, h_0) where $\mathcal{S}$ is replaced by*

$$\mathcal{S}(g_0, h_0) \;=\; \{x \;\in\; \mathbb{R}^n : h(x) \;=\; h_0,\; g(x) \;\geq\; g_0\}\;.$$

For $x \in \mathcal{S}(0,0)$ there exist some $\tilde{x} \in \mathcal{S}(h_0, g_0)$ with $\|x - \tilde{x}\| \leq C \times (\|h_0\| + \|g_0^+\|)$ for some constant $C = C(x)$ and (h_0, g_0) in some neighborhood of $(0,0)$ if and only if MFCQ holds at x.

For the proof, see [104]. A similar result applies to perturbations of the problem functions h and g in a nonparametric (topological) sense [66]. Hence one may consider a problem instance of *NLP* without *MFCQ* as incorrectly stated. This is surely true as long as feasible points are considered only. But observe that in practice one also needs that condition for infeasible points. Unfortunately it is far from being true that *MFCQ* is naturally satisfied there. Consider the simple and extremely well behaved example with

$$n \;=\; 2,\; f(x) \;=\; -x_1 - x_2,\; h(x) \;=\; x_1^2 + x_2^2 - 2 \quad \text{and} \quad g(x) \;=\; x\;. \tag{1.1}$$

The feasible set is the quarter circle of radius $\sqrt{2}$ in the positive quadrant and even *LICQ* is satisfied here in a small neighborhood of the feasible set. However *MFCQ* is violated for $x_1 < 0,\; x_2 < 0$ and, surprisingly enough, some highly renowned codes fail on this example if provided with an initial guess in this region, if not g is given special treatment. (Most codes do not accept initial guesses which violate the given bound constraints. However, it is possible to construct similar examples with a more complicated, nonlinear g, hence this trouble remains.)

There exist also sufficient local optimality conditions for nonconvex problems. These involve the Hessian matix of the so called Lagrangian of the Problem *NLP*, namely

$$L(x, \lambda, \mu) \;=\; f(x) - h(x)^T \mu - g(x)^T \lambda\;,$$

e.g.

Theorem 1.3 *Let be given a point $x^* \in \mathcal{S}$ which satisfies MFCQ and KTC with some pair (λ^*, μ^*). If in addition*

$$z^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) z > 0 \quad \forall\; z \;\neq\; 0$$

with

$$\begin{aligned}
\nabla h(x^*)^T z \;&=\; 0, \\
\nabla g_i(x^*)^T z \;&\geq\; 0 \text{ if } \lambda_i^* \;=\; 0 \\
\nabla g_i(x^*)^T z \;&=\; 0 \text{ if } \lambda_i^* \;>\; 0 \quad \forall\, i \in\; I \text{ and some } \lambda^* \;\in \mathcal{L}
\end{aligned}$$

then x^* is a strong local minimizer of NLP.

For a proof, again see [96]. Assumptions as given in the previous theorems are also typical for the convergence analysis of optimization algorithms for solving *NLP*.

2 Unconstrained Minimization

The original problem

$$f(x) \;\stackrel{!}{=}\; \min_{x \in \mathbb{R}}$$

is typically reduced to the identification of a single stationary point, that means one solution of the (in general nonlinear) equation

$$\nabla f(x) \;\stackrel{!}{=}\; 0\;.$$

There are known also methods which aim in identifying points which in addition satisfy the necessary second order condition "$\nabla^2 f(x^*)$ positive semidefinite". The classical approach here is the damped

Newton method, either in form of employing a line search along the Newton direction or in form of a trust-region approach. Newtons method uses second derivatives and before the advent of automatic differentiation techniques this has been considered as a severe disadvantage. Much effort therefore has gone into the development of methods which avoid usage of second derivatives, which lead to the class of quasi-Newton methods. Astonishingly enough the use of e.g. the well analyzed methods from the Broyden class, especially the *BFGS*-update formula, often results in algorithms of higher efficiency than the usage of exact second derivatives, maybe because these methods in some sense remember the global behaviour of f whereas a true Newton step employs pure local information only. It can safely be stated that unconstrained problems of small and medium large dimension, say up to some hundred variables, can nowadays routinely and safely be solved using available quasi-Newton based codes, even if the Hessian is rather illconditioned. Unfortunately, no method of practical relevance evolved in this class which were capable of dealing with sparsity in the Hessian (updates from the Broyden class always give full matrices even for a diagonal Hessian). However much progress has been made towards the solution of large-scale problems by exploiting special structure in the Hessian or by using methods adapted from large-scale linear solvers, see e.g. work by Griewank and Toint [67], Nash [92], Buckley and Lenir [18], Byrd, Nocedal and Schnabel [24] and Liu and Nocedal [83]. Griewank and Toint consider problems which show a special additive structure, the class of so called partial separable functions, that are functions f with

$$f(x) \;=\; \sum_{j=1}^{m} f_j(x_j) \text{ with } x_j \text{ a lowdimensional subvector of } x \;.$$

The central idea is to devise updating formulae for the Hessians of the f_j by dense matrices A_j separately and combining these to a sparse approximation of the Hessian of f. There is public domain software which implements this. The method works well in the convex case and if dimensionality is small enough to allow direct solution techniques for the linear system with the matrix $\sum_j A_j$. All other methods mentioned can be characterized as modifications of Newton's method (so called inexact Newton methods) or of the method of conjugate gradients, known for the linear case since 1951. Using these methods problems with some 10000 variables can be solved without much trouble, at least if illconditioning of the Hessian is not too hard.

In high dimensional unconstrained minimization the main effort comes from the determination of the so called direction of descent d^k with $(\nabla f(x^k))^T d^k < 0$. With this in mind the majority of methods which have been proposed up to now follow the linesearch model, i.e.

$$x^{k+1} \;=\; x^k + \sigma_k d^k \;,$$

where σ_k is chosen such that $\{f(x^k)\}$ decreases strongly monotonically satisfying the principle of sufficient decrease

$$f(x^k) - f(x^{k+1}) \to 0 \Rightarrow \nabla f(x^k) \to 0 \;,$$

with d^k satisfying the condition

$$\frac{|(\nabla f(x^k))^T d^k|}{||d^k||} \;\geq\; \phi(||\nabla f(x^k)||).$$

Here ϕ is a strongly monotonically increasing function with $\phi(0) = 0$, not necessarily explicitly known. The stepsize σ_k can be computed by a variety of algorithms, the simplest one being backtracking:

$$\sigma_k \;=\; \text{argmax}_j\{\beta^j\sigma_{0,k} : j \;\in\; \mathbb{N}_0 \text{ and } f(x^k) - f(x^k + \beta^j\sigma_{0,k}d^k) \;\geq\; \delta\beta^j\sigma_{0,k}(-\nabla f(x^k))^T d^k\}$$

where $0 \;<\; \beta \;<\; 1$ and $0 \;<\; \delta \;<\; \frac{1}{2}$ are two constants and $\sigma_{0,k}$ an initial guess, maybe simply 1 or some value obtained from a preliminary interpolation, e.g. using the values $f(x^k)$, $f(x^k + d^k)$ and $\nabla f(x^k)^T d^k$.
The so called trust-region methods also require a descent of this form, using formally $\sigma_k = 1$. Here a local model for f is used which has the form

$$\tilde{f}(d) \;=\; f(x_k) + (\nabla f(x^k))^T d + \frac{1}{2} d^T A_k d, \; ||d|| \;\leq\; \rho_k \;.$$

The (global) minimizer of this problem yields d^k and the step $x^{k+1} = x^k + d^k$ is accepted if

$$0 \;<\; \eta \;<\; \frac{f(x^k) - f(x^{k+1})}{\tilde{f}(0) - \tilde{f}(d^k)}$$

for some small constant $\eta > 0$, e.g. $\eta = 10^{-3}$. There is then an adaptive choice of ρ_k. If the test fails, then ρ_k is decreased and d^k must be computed anew, that means a further (large) linear system must be solved, which increases the algebraic complexity considerably. But the function information need not be computed anew and for very expensive evaluations the additional algebraic effort may be worthwhile. If the step is accepted, then ρ_{k+1} is chosen equal to or larger than ρ_k. Trust-region methods have advantages over linesearch methods because of weaker conditions on the models matrix A_k and stronger theoretical results (typically these methods converge to second order necessary points without using negative curvature information if $A_k = \nabla^2 f(x^k)$ for all k.) A trust-region Newton's method is realized in the code `TRON` of Lin and Moré [82].

In the (theoretically) ideal case the direction d^k is given by Newton's method. The linear system

$$\nabla^2 f(x^k) d^k \;=\; -\nabla f(x^k)$$

can be solved by direct methods for large n if the Hessian is very sparse only. Fortunately, it is not necessary to solve this system very precisely, at least far from the solution, without loosing the good convergence properties. This observation leads to the so called inexact Newton methods, where the Newton direction is approximated by some steps of the preconditioned conjugate gradient method or some solver based on the Lanzcos algorithm. This latter approach requires a routine which computes the product of the Hessian with a given arbitrary vector. In Nash's Code `TN/TNB` this is replaced by taking a forward difference of the gradient $(\nabla f(x^k + \tau v) - \nabla f(x^k))/\tau$, which provokes additional problems. If the solution precision is controlled appropriately (one needs a final residual less than $\alpha_k ||\nabla f(x^k)||$, $\alpha_k \to 0$), then superlinear convergence speed is maintained. The computation of d^k requires several matrix-vector products respectively several evaluations of ∇f.

Alternatively one may use the so called quasi-Newton methods with limited-memory or the nonlinear preconditioned conjugate gradient method, applied directly to f. These methods generate directions of descent also for nonconvex f and need as essential information only one gradient of f every step. The nonlinear conjugate gradient method, originally proposed by Fletcher and Reeves and now known in a lot of variants is the simpler of these two approaches. Here d^k is computed recursively

$$\begin{aligned}
g^k &= \nabla f(x^k) \\
\hat{g}^k &= B g^k, (B \text{ is the preconditioner, positive definite }) \\
\gamma_k &= (g^k)^T \hat{g}^k \\
d^k &= \begin{cases} \hat{g}^k \text{ if } k = 0(\bmod N) \\ (1 - \frac{(g^k)^T d^{k-1}}{\gamma_{k-1}})\hat{g}^k + \frac{\gamma_k}{\gamma_{k-1}} d^{k-1} \text{ otherwise.} \end{cases}
\end{aligned}$$

N is the so called restartindex. In theory one has $N \geq n$, and, as shown by AlBaali [2], N may be even infinite. With the exception of the leading factor of $\hat{g}^k$ these are the relations of the classical (linear) cg-algorithm. The modification forces $(g^k)^T d^k = \gamma_k > 0$, independently of the properties of f and σ_k. In practical computations it makes little sense to let N grow very much because the quality of the directions is deteriorated very strongly by roundoff and finally the determination of the stepsize σ comes into trouble .

Remark.: There is a point which often is neglected in theoretical discussions of this method. In the linear case, with a fixed positive definite Hessian, one knows for sure that the "optimal" stepsize produces descent and never checks that, whereas here monotonic descent of f must be verified, for fixing the stepsize. However, under the influence of roundoff, the test for descent may fail even if the direction d^k is a good one. □

If one chooses $N \geq n$ then N-step quadratic convergence of this method can be shown. However, for very large n this is practically irrelevant.

The usual quasi-Newton methods replace the Hessian G_k by a positive definite "approximation" A_k and compute d^k from

$$A_k d^k = -\nabla f(x^k)$$

resp. with $H_k = A_k^{-1}$

$$d^k = -H_k \nabla f(x^k) .$$

None of the quasi-Newton methods designed for the maintenance of a given sparsity pattern of the Hessian has shown practical success, whereas the successful ones all produce full matrices even for diagonal Hessian . That caused the research in limited memory methods. One of the best practical methods is based on the so called *BFGS*-formula

$$A_{k+1} = A_k - \frac{A_k s^k (s^k)^T A_k}{(s^k)^T A_k s^k} + \frac{y^k (y^k)^T}{(y^k)^T s^k} ,$$

which for $(y^k)^T s^k > 0$ and positive definite A_k produces a positive definite A_{k+1} . (It is even possible to show that for uniformly convex f the eigenvalues of A_k and A_k^{-1} remain uniformly bounded under very weak conditions on the step $x^k \rightarrow x^{k+1}$, see e.g. [23]. The condition $(y^k)^T s^k > 0$ is satisfied automatically for f uniformly convex and is enforced otherwise through the stepsize selection using the so called Powell-Wolfe conditions). In this formula

$$s^k = x^{k+1} - x^k$$

is the current step and

$$y^k = \nabla f(x^{k+1}) - \nabla f(x^k) .$$

For A_k resp. $H_k = A_k^{-1}$ there exists for the case $A_0 = \frac{1}{\gamma} I$ another closed formula [24]

$$H_{k+1} = \gamma I + C_k \begin{pmatrix} D_{11,k} & D_{12,k} \\ D_{21,k} & O \end{pmatrix} (C_k)^T$$

where

$$\begin{aligned} C_k &= \begin{pmatrix} S_k & \gamma Y_k \end{pmatrix} , \\ D_{21,k} &= -R_k^{-1} , \\ D_{12,k} &= (D_{21,k})^T , \\ D_{11,k} &= D_{12,k}(\Delta_k + \gamma (Y_k)^T Y_k) D_{21,k} \end{aligned}$$

with

$$\begin{aligned} (R_k)_{ij} &= \begin{cases} (s^i)^T y^j \text{ for } 0 \le i \le j \le k \\ 0 \text{ otherwise} \end{cases} \\ \Delta_k &= \mathrm{diag}_{0 \le i \le k}((s^i)^T y^i) , \\ S_k &= (s^0, \ldots, s^k) , \\ Y_k &= (y^0, \ldots, y^k) . \end{aligned}$$

The idea of quasi-Newton methods with limited memory simply consists in defining in this formula Y_k, S_k, Δ_k and R_k using the last m instances $y^k, \ldots, y^{k-m+1}, s^k, \ldots, s^{k-m+1}$ and a variable $\gamma = \gamma_k$ and fixing a formula for H_k this way. Of course this applies for $k > m$ only. Recommended values for m are in the range $\{7, \ldots, 20\}$. This is implemented in the code `LBFGS(B)` of Byrd , Nocedal and coworkers [22]. With $m \ge n$ this method should, from theory, give the same results as the original method, if applied to uniformly convex quadratic f. Unfortunately this is far from being true. The numerical evaluation of H_k using this "direct" formula seems to be subject to numerical instability and further research in this direction is required. However, practioners like the method because of their simplicity. (no Hessian required). But from numerical experiments it became clear that for strongly nonquadratic f the incomplete Newton methods are to be preferred. Numerical experience is reported in the papers [57], [83] , [84], [119].

3 Bound constrained problems

For reasons of simple exposition we restrict the discussion to the problem

$$f(x) \stackrel{!}{=} \min_{x \in \mathbb{R}^n : x \geq 0} \; .$$

The case of general lower and upper bounds and of only partially constrained components of x can be derived with ease from this special one. The necessary first order optimality conditions now read

$$\nabla f(x) - \lambda = 0, \quad \lambda_i x_i = 0, \quad \lambda \geq 0 \; .$$

This can be rewritten as

$$\nabla f(x)_i \;=\; \begin{cases} 0 \text{ if } x_i > 0 \\ \geq 0 \text{ otherwise} \; . \end{cases}$$

The first methods for this type of problem were of the type of active set methods. Initially f is minimized using an unconstrained technique. However, by proper selection of the stepsize, the constraints remain satisfied. If a variable meets its lower bound, it is fixed there and minimization of f continues on a boundary linear manifold of the feasible set. This process continues until either a (resp. the) constrained minimizer is located or otherwise the components of the gradient corresponding to free variables became small whereas one of its components corresponding to a fixed variable stays relatively large and of negative sign, indicating a false constraining manifold. In the latter case the corresponding variable is freed and moved into the relative interior of the feasible set, with minimization of f continuing. Using some simple criteria for fixing what is meant by "small" and "relatively large" here convergence of this process to a constrained minimizer can be shown. This kind of method is simple to implement and robust, but necessarily very slow if many changes of the active set are required. Nash's code `TNB` is of this kind. All of the newer methods for this kind of problem however make use of a special intermediate step as soon as a variable meets its bound. This special step consists in (at least one) step of the gradient projection method. This is given by exact (for linear of quadratic f) or approximate minimization of f along the projection of the ray $x^k - \sigma \nabla f(x^k)$ onto the feasible set, which in the case of bound constraints is trivial , namely

$$\mathcal{P}_{\mathcal{S}}(x^k - \sigma \nabla f(x^k)) \;=\; \max\{0, x^k - \sigma \nabla f(x^k)\} \; .$$

with max for a vector understood componentwise. If the active set $\mathcal{A}$ does not change, then unconstrained minimization of f on a constraining manifold (using a method better than the mere gradient descent) is resumed. This leads to fast changes in the active set and rather efficient overall performance, e.g. Byrd et. al. [22] (code `LBFGSB`), Conn, Gould und Toint [26] (code `SBMIN` gone into `LANCELOT`), [27], Moré and Lin [82] , code `TRON`, Moré and Toraldo [88], [89] (code `GPCG` by Felkel, see [87]) and Felkels code `PL2`, again see [87] and Felkels PhD dissertation [39], see also Moré's `MINPACK 2` project [3].

Using this technique it is possible to solve bound constrained quadratic problems with almost the same effort as unconstrained ones. However, this author does not completely participate in the enthusiasm of many colleagues for gradient projection. It is true that the sole role of this method is the fast identification of the correct active set. Nevertheless, for a general nonlinear f it is necessary to check the descent of f along the projected ray if the function is nonconvex or if the computation of the two directional derivatives of f at the breakpoints of the ray cannot be computed cheaply. Because the optimal stepsize along the projection might be quite small for mildly illconditioned Hessian of f, this may lead to premature termination. The optimal stepsize along a ray $x - \sigma \nabla f(x)$ is of the form

$$\sigma^* \;\in [\frac{1}{\mathrm{eig}_{\max}(G)}, \frac{1}{\mathrm{eig}_{\min}(G)}](1 + \mathcal{O}(||\nabla f||)) \; ,$$

where $\mathrm{eig}(G)$ are the eigenvalues of the Hessian G of f. The correction

$$\sigma^* \nabla f(x) \stackrel{def}{=} s$$

can be as small as

$$\frac{3\sqrt{3}}{2}\frac{1}{cond(G)}||x - x_{\text{opt}}||(1 + \mathcal{O}(\frac{1}{cond(G)}))$$

whereas the difference $f(x) - f(x+s)$ may be of the order of

$$\frac{9}{4}\frac{1}{cond(G)}\text{eig}_{\min}(G)||x - x_{\text{opt}}||^2$$

(which is much smaller than $f(x) - f(x_{\text{opt}})$.) If this value reaches the roundoff level in the evaluation of f the method terminates of course, at a relatively large value $||x - x_{\text{opt}}||$, even for $\nabla^2 f$ only mildly illconditioned. Numerical tests verified this behaviour [100]. Therefore it might finally not be the best solution to move along constraining manifolds at all. There exist newer approaches which make use of a primal-dual approach, including movement in the dual variables $(\nabla f(x))_i$ for $x_i = 0$, e.g. [33]. Others aim in avoiding movement along constraining manifolds using moves that at least in a number of variables are always interior to the feasible set, e.g. [13]. But thorough testing of all these solution approaches on an identical testbed are necessary to assess their relative merits.

4 General linearly constrained problems: active set methods

Linearly restricted problems are mucher easier to handle than those with general constraints. The feasible set is convex. There exist methods for the elimination of redundant equalities and inequalities from systems of linear equality and inequality constraints which work (at least theoretically). Numerically there might be trouble due to illconditioning. (Indeed there are lots of examples in the `NETLIB LP`-library where infeasibility is hard to discern.) With this reduction done *MFCQ* is satisfied globally for these systems. There exist finite algorithms to decide feasibility of such constraints (e.g. phase I of the simplex algorithm). No one of these properties translates to general constraints.

Minimization methods of the active set type (minimization on submanifolds) can easily be devised for general linear constraints, hence it is possible to maintain feasibility in linearly constrained optimization quite easily, once a feasible point is known. Some of the *SQP*-methods discussed later in this paper use this fact. Contrary to this the gradient projection method cannot be implemented efficiently for this type of constraints, since the determination of the projection of an (infeasible) point onto a polyhedron involves the solution of a convex quadratic programming problem whose solution might be as costly as the solution of the original problem.

For the convex quadratic linearly constrained optimization problem, the so called convex *QP* problem, i.e.

$$f(x) \;=\; \frac{1}{2}(x)^T Ax - a^T x \quad \text{where } A \text{ is positive semidefinite} \quad .$$

with the constraints

$$g(x) \;=\; G^T x + g^0 \;\geq\; 0 \;, \quad h(x) \;=\; H^T x + h^0 \;=\; 0 \;,$$

including the case $A = O$, the *LP* problem, there exist specialized highly efficient methods, e.g. the simplex method , the dual simplex method and its generalizations to the quadratic case, e.g. methods given by Fletcher [42], Goldfarb [60] and Goldfarb and Idnani [61]. These methods are part of many optimization libraries and the core of most of the *SQP* methods for medium large problems, which will be dealt with later in this paper. Using sparse matrix methods these algorithms can also be extended to higher dimensional problems, for the quadratic case see e.g. [62]. For the *LP* problem there exists a bunch of implementations, commercial as well as in the public domain, capable of solving problems up to 10^6 variables, given sufficient sparsity in the constraints.

There is newer work [49] which deals with the possibility to combine classical active set technology with activation and inactivation of an arbitrary number of constraints per step, which makes these methods even more efficient, see also sections 3.4.6 and 3.4.7 in [113]. Nevertheless the theoretical worst case complexity of such an approach is always exponential in dimension (and input size). Algorithmically, these methods can be quite involved.

Besides direct treatment of the active linear constraints via elimination techniques (using a choice of numerical linear algebra techniques) there exists also the possibility to transform the problem to a bound constrained one and using the powerful techniques described in the previous section for the latter. For simplicity of exposition let us assume that the constraints are already in the form used by the primal simplex method, namely

$$B^T x - b \;=\; 0 \;, x \;\geq\; 0 \;.$$

Indeed any linearly constrained problem can be transformed such that the constraints take this form. Let us also assume that B is of full rank (In reality this is often not the case, and the numerical treatment of rank deficiency is extremely delicate.) The first order necessary optimality conditions then read

$$\begin{aligned} \nabla f(x) - B\mu - \lambda &= 0 \;, \\ \lambda &\geq 0 \;, \\ \lambda_i x_i &= 0 \;,\; i = 1, \ldots, n \;. \end{aligned}$$

If f is convex then these conditions are also sufficient for optimality of a feasible point. If there exists a so called Slater point $\bar{x}$ with $B^T \bar{x} - b \;=\; 0 \;, \bar{x} \;>\; 0$, then the set of possible Lagrangian multipliers is bounded. Friedlander, Martínez and Santos [54] have shown, that in the case f convex and of class C^2 the solution can be obtained via the following bound constrained minimization problem involving the primal and the dual variables x, μ and λ simultaneously:

$$M(x, \mu, \lambda) \;=\; \frac{1}{2}\big(||\nabla f(x) - B\mu - \lambda||^2 + ||B^T x - b||^2 + (x^T \lambda)^2\big) \;\stackrel{!}{=}\; \min_{x \geq 0,\; \lambda \geq 0} \;.$$

An obvious disadvantage of this approach is the increase in dimensionality. Kanzow [76], using the so called Fischer-function

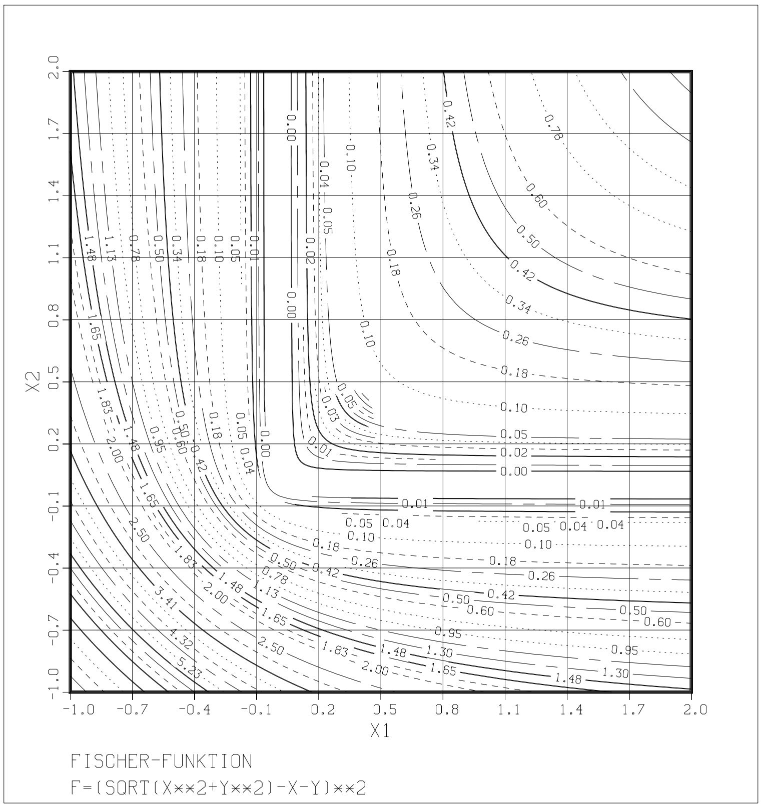
$$\phi(\alpha, \beta) \;=\; (\sqrt{\alpha^2 + \beta^2} - \alpha - \beta)^2$$

succeeded in reformulating the problem as a completely unconstrained one:

$$K(x, \mu, \lambda) \;=\; \frac{1}{2}\big(||\nabla f(x) - B\mu - \lambda||^2 + ||B^T x - b||^2 + \sum_{i=1}^{n} \phi(x_i, \lambda_i)\big) \;\stackrel{!}{=}\; \min \;.$$

Fischer's function ϕ has the interesting property to be stationary exactly for $\alpha, \beta \geq 0, \alpha\beta = 0$ and penalizing simultaneously the primal and dual bound constraints as well as the so called complementarity condition $x_i \lambda_i = 0$. There are many other functions of this type which are heavily used in solution methods for the so called complementarity problem

$$F(x)^T y \;=\; 0, \quad F(x) \;\geq\; 0, \quad y \;\geq\; 0 \;.$$



Numerical tests with high dimensional convex QP problems have shown superiority of the Friedlander et. al. approach compared with Kanzow's. Both methods transform a convex quadratic problem to a nonconvex nonquadratic one, which may be considered as a disadvantage. Indeed any convex QP problem can be rewritten as a convex bound constrained QP problem, as shown by the present author [110]. With

$$L(x,\mu,\lambda) = f(x) - \mu^T(B^Tx - b) - \lambda^T x$$

and

$$\Phi(x,\mu,\lambda;\delta,\eta) = -L(x,\mu,\lambda) + \tfrac{\delta}{2}\,\|B^Tx - b\|^2 + \tfrac{\eta}{2}\,\|Ax - a - B\mu - \lambda\|^2$$

there holds the following: If A is positive semidefinite and in addition positive definite on the kernel of B^T, then the minimization of Φ with respect to $x,\ \mu,\ \lambda$ under the bound constraints

$$x \ \geq\ 0\ ,\ \lambda\ \geq\ 0$$

is equivalent to solving the original QP provided $\delta > 0$ and $\eta > \rho((Z^TAZ)^{-1})$, with Z an orthonormal basis of B^T's kernel. $\rho(.)$ denotes the spectral radius of a matrix. Here the trouble comes through the determination of η. Since the (nearly exact) determination of the theoretical lower bound for this parameter is out of discussion for a truly large-scale problem, much remains to be done here. All these approaches introduce an additional problem (also present for all known primal dual differentiable exact penalty functions for the general NLP problem): the condition number of the Jacobian of the corresponding KTC-system is in the order of the square of the condition number of the KTC-system of the original problem and there seems to be no simple way to alleviate this effect. It is an open question whether there exist exact primal-dual smooth penalty functions (a function for which a single unconstrained minimization gives the solution of NLP together with the multipliers) without that nasty effect. Numerical tests which such methods are given in [40], [114]. In [114] the QP problems are constructed artificially with known solution and known condition number and minimized with the code `LBFGSB` of [22]. This solver was used with identical parameters and identical termination criterion, requiring maximum attainable accuracy in the objective function . The tables which follow show the precision obtained. `DXR`, `DMR` and `DLR` denote the norm of the error divided by the norm of the true solution, measured in the euclidean norm, for the variables x, μ and λ respectively. The condition number of the Hessian and the reduced Hessian was chosen in $\{10, 100, 1000, 10000\}$, just the reciprocal of the value $\sigma_{\min}$. n varied from 1000 to 10000 in steps of 1000.

Method of Friedlander et. al.

$\sigma_{\min}$	$DXR \in$	$DMR \in$	$DLR \in$
10^{-1}	$[5 \cdot 10^{-6}, 4 \cdot 10^{-5}]$	$[6 \cdot 10^{-5}, 1 \cdot 10^{-3}]$	$[7 \cdot 10^{-7}, 3 \cdot 10^{-5}]$
10^{-2}	$[3 \cdot 10^{-4}, 3 \cdot 10^{-3}]$	$[5 \cdot 10^{-2}, 0.1]$	$[1 \cdot 10^{-5}, 3 \cdot 10^{-3}]$
10^{-3}	$[1 \cdot 10^{-3}, 1 \cdot 10^{-2}]$	$[8 \cdot 10^{-2}, 0.1]$	$[2 \cdot 10^{-5}, 0.03]$
10^{-4}	$[3 \cdot 10^{-3}, 3 \cdot 10^{-2}]$	$[0.09, 0.2]$	$[3 \cdot 10^{-5}, 0.06]$

Method of Kanzow

$\sigma_{\min}$	$DXR \in$	$DMR \in$	$DLR \in$
10^{-1}	$[2 \cdot 10^{-5}, 1 \cdot 10^{-2}]$	$[4 \cdot 10^{-4}, 1 \cdot 10^{-1}]$	$[9 \cdot 10^{-8}, 3 \cdot 10^{-2}]$
10^{-2}	$[2 \cdot 10^{-3}, 2 \cdot 10^{-2}]$	$[0.2, 0.3]$	$[6 \cdot 10^{-3}, 8 \cdot 10^{-2}]$
10^{-3}	$[7 \cdot 10^{-3}, 4 \cdot 10^{-2}]$	$[0.1, 0.2]$	$[8 \cdot 10^{-3}, 0.1]$
10^{-4}	$[1 \cdot 10^{-2}, 5 \cdot 10^{-2}]$	$[0.1, 0.3]$	$[6 \cdot 10^{-3}, 0.1]$

Method of Spellucci

$\sigma_{\min}$	$DXR \in$	$DMR \in$	$DLR \in$
10^{-1}	$[1 \cdot 10^{-6}, 2 \cdot 10^{-5}]$	$[2 \cdot 10^{-5}, 3 \cdot 10^{-4}]$	$[2 \cdot 10^{-11}, 3 \cdot 10^{-6}]$
10^{-2}	$[7 \cdot 10^{-5}, 3 \cdot 10^{-4}]$	$[5 \cdot 10^{-3}, 4 \cdot 10^{-2}]$	$[4 \cdot 10^{-12}, 1 \cdot 10^{-6}]$
10^{-3}	$[2 \cdot 10^{-4}, 0.1]$	$[0.05, 0.09]$	$[6 \cdot 10^{-12}, 0.02]$
10^{-4}	$[3 \cdot 10^{-3}, 0.3]$	$[0.04, 0.09]$	$[2 \cdot 10^{-11}, 0.04]$

5 Linearly constrained problems: Interior-point methods

In the following we again restrict the discussion to the special problem

$$f(x) \stackrel{!}{=} \min$$

with the constraint

$$B^T x - b \;=\; 0\,, \quad x \;\geq\; 0\,.$$

Again we require full rank of B. At first we only consider a convex quadratic f, including the LP case. Since Karmarkar's seminal paper [77] on the construction of methods of polynomial complexity for the LP problem there has been an enormous research effort on what is now known as "interior-point methods". Meanwhile the online archive [74] contains papers which appeared since 1994, and also a pointer to an older collection from Eberhard Kranich, making up some thousand contributions. Today the so called primal-dual interior-point methods are considered as the most powerful approach and we discuss here a typical candidate from this class. The development starts with considering the KTC conditions which in the convex case, given a Slater-point, are necessary and sufficient characterizations of the solution:

$$\begin{aligned} Ax - a - B\mu - \lambda &= 0\,, \\ \lambda &\geq 0\,, \\ x &\geq 0\,, \\ \lambda_i x_i &= 0\,,\ i = 1, \ldots, n \\ B^T x - b &= 0\,. \end{aligned}$$

We consider this as a nonlinear system in x, μ and λ. We parametrize it changing the complementarity condition to

$$\begin{aligned} Ax - a - B\mu - \lambda &= 0\,, \\ \lambda_i x_i &= \epsilon\,,\ i = 1, \ldots, n\,, \\ B^T x - b &= 0\,. \end{aligned}$$

On the set of "interior" points

$$x > 0 \,, \lambda > 0$$

the Jacobian of this system is always regular. For $\epsilon > 0$ the unique solution describes a smooth curve with parameter ϵ, (known as the "central trajectory") which tends for $\epsilon \to 0$ to one of the solutions of the initial problem.
Under the additional condition of strict complementarity of one solution this limit point can be precisely characterized (it is the "analytic" center of the solution set).
Remark.: Observe that in the *LP*- and also in the not strictly convex *QP*-case the solution will not necessarily be unique. □
This solution curve is now computed pointwise and approximately only, using the damped Newton method for ϵ fixed, and varying ϵ slowly. During this computation the constraints $x > 0$, $\lambda > 0$ must be maintained. If this is done naively then one runs into serious trouble because of the singularities at the boundary and the true art in devising an efficient algorithm consists in a senseful change of ϵ and a senseful selection of the stepsize parameter for damping the Newton step. It turns out that ϵ_k should depend on the duality gap

$$(x^k)^T\lambda^k/n.$$

A typical algorithm is the one of Kojima, Mizuno and Yoshise [78]:

$$\begin{aligned}
\mathcal{J}_F(z^k)\Delta z_N^k &= -F(z^k) \text{ to be solved : Newton step} \\
\mathcal{J}_F(z^k)\Delta z_C^k &= \rho_k\hat{e} \text{ to be solved : centering step} \\
\rho_k &= \sigma_k((x^k)^T\lambda^k)/n \ , \\
\Delta z^k &= \Delta z_N^k + \Delta z_C^k \stackrel{def}{=} (\Delta x^k,\ \Delta\mu^k,\ \Delta\lambda^k)^T \\
\hat{\alpha}_k &= \min\{\min\{x_i^k/(-\Delta x_i^k) : \Delta x_i^k < 0\}, \min\{\lambda_i^k/(-\Delta\lambda_i^k) : \Delta\lambda_i^k < 0\}\} \ , \\
\alpha_k &= \min\{1, \tau_k\hat{\alpha}_k\} \ , \\
z^{k+1} &= z^k + \alpha_k\Delta z^k \ .
\end{aligned}$$

Here $z = (x, \mu, \lambda)$, $\tau_k \in]0, 1[$, $\sigma_k \in [0, 1[$ and $\hat{e} = (0, \dots, 0, e)^T$ where $e = (1, \dots, 1)^T \in \mathbb{R}^n$.
The function F is given by

$$F(x, \mu, \lambda) = \begin{pmatrix} Ax - a - B\mu - \lambda \\ -B^Tx + b \\ X\lambda \end{pmatrix}$$

where $X = diag(x_1, \dots, x_n)$. In the *LP* and convex *QP* case one can show total polynomial complexity of the algorithm for reaching the exact solution (in the case of exact rational input data) with a step number of $\mathcal{O}(\sqrt{n}L)$, where L is the so called input length. In the case of integer coefficients and a *LP* this is defined as

$$L = \sum_{i,j}\log_2(|b_{ij}| + 1) + \sum_j \log_2(|b_j| + 1) + \sum_i \log_2(|a_i| + 1) \ .$$

After a finite number of steps the correct constraining manifold can be identified exactly, which allows subsequently to abandon the path following and to move to the boundary in $\mathcal{O}(n)$ steps. Of course, this is a nice theoretical result (concerning the so called "small step" methods) but without any practical relevance. A rule of thumb states that "thirty steps suffice" (for the so called "large step" methods) to come sufficiently near to the desired solution. It remains to discuss how to solve the linear systems in this method in the true large-scale situation. The Jacobian of F reads

$$\mathcal{J}_F(z) = \begin{pmatrix} A & -B & -I \\ -B^T & O & O \\ \Lambda & O & X \end{pmatrix} \ , \ \Lambda = \text{diag}(\lambda_i) \ .$$

A is the Hessian of f and hence positive semidefinite for convex f. If $A = O$, i.e. the *LP* case, then the solution of the linear systems with this matrix is relatively easy, since it can be reduced to a linear system with the (by assumption) positive definite and much smaller matrix $B^T\Lambda^{-1}XB$. In case of a general semidefinite A this is no longer possible and a first resort would be methods for large indefinite symmetric systems, e.g. the techniques developed by Duff, Reid and coworkers [34], [35]. Fortunately these matrices are not general indefinite, such that more efficient solution methods are available see e.g. [59], [116]. There exist other specialized techniques, e.g. iterative preconditioned techniques like those considered by Freund and Jarre [53]. Primal-dual methods for the LP case are discussed in depth by St. Wright in [115]. Some typical results obtained for some instances of LP problems from the `NETLIB` collection are listed below. It can be seen that in many instances interior-point methods can by far outperform the simplex method, but there are problems concerning the reliability of such codes, which however continue to diminish in time to date. `BPMPD` (by Meszaros) and `HOPDM` (by Gondzio) roughly correpond to the method sketched above, whereas the codes `LOQO` (by Vanderbei), `LIPSOL` (by Zhang) and `PCx` (by St. Wright) use modified approaches. `SOPLEX` is a well written simplex-code (in C) by Wunderlich, formerly with ZIB-Berlin, which is maintained by the ZIB. All information is from [86] and [87]. The table only gives computing time for a HP9000/735-99 with 192MB main memory. "*" designates a failure. It is possible to derive interior-point methods from the minimization of the so called logarithmic barrier function (a method known for a long time [41] and disregarded subsequently because of the inherent illconditioning. In the case of a problem satisfying the linear independence and second order sufficiency condition $E + |\mathcal{A}|$ of the eigenvalues of the Hessian tend to infinity whereas the rest remains bounded. If path following is used and the exact Hessian is available, this has only little effect. However, if one uses a general method of descent then even slight illconditioning has a detrimental effect due to the loss of centrality. Restricted to the central path the conditioning of the problem is much better). The logaritmic barrier method penalizes a step towards the boundary by a term which grows to infinity there, multiplied by a parameter (the ϵ above) which is driven to zero, such that the minimizer is approached from the interior of the feasible set. Mostly one uses the so called logarithmic barrier function

$$f(x) - \epsilon \sum_{i=1}^{n} \ln(x_i) \stackrel{!}{=} \min_{x}, \quad B^T x - b = 0$$

with $\epsilon \to 0$. The direct application of this idea requires a point $x > 0$, which satisfies the equality constraints. This defines a "feasible interior-point" approach, but there exist also "infeasible interior-point" methods, where the equality constraints are satisfied in the limit only.

Remark: If the QP problem is nonconvex, then there is no known algorithm of polynomial complexity to solve it. Even the decision whether a stationary point found is a local minimizer, is NP hard, [107], [118]. The methods described above cannot be used for nonconvex cases. □

Remark: In addition to the results mentioned above it is at least of theoretical interest that it is also possible to obtain superlinear convergence using these methods by proper choice of the parameters, see e.g. [102] □

If f is a general smooth convex function, then the same techniques can in principle be applied, simply replacing the matrix A there by $\nabla^2 f(x^k)$. The step size selection then has in addition to check the descent property of f as the sole additional complication. An algorithm around this ideas has been developed from `HOPDM`, see [37].

codes/versions:

BPMPD-2.14	HOPDM-2.13	LIPSOL-0.31	LOQO-3.04	PCx-1.0	SOPLEX-1.0

SOPLEX has been used in two versions, C/R, C is the column oriented and R the row oriented mode.

problem	BPMPD	HOPDM	LIPSOL	LOQO	PCx	SOPLEX_C/R
8 baxter	115	14454	10822	128	4979	168/139
7 dano3mip_lp	1802	3676	*	2175	1161	1698/4082
8 dbir1	994	*	*	6874	5835	959/568
9 fxm3_16	223	267	1047	549	*	3219/6268
11 gen4	917	*	1265	696	*	982/2208
8 nsct1	1808	*	*	*	14868	422/128
8 nsir2	50	219	761	261	262	122/41
8 nug12	1825	4808	5211	8179	8081	41299/29923
12 qap12	2747	2331*	2711*	8005	5062*	132405/22691
8 rat1	14	16	697	1447	27	1402/7278
8 rat7a	427	682	2038	5192	459	6697/26752
8 rlfddd	240	355	871	572	157	47/2134
8 rlfprim	251	*	*	2152	*	137/229
7 seymour_lp	24	10142	8991	28	4547	54/153
4 world	588	1020	*	1419	*	17419/15014

additional information concerning the problems

problem	rows	equal.	columns(=n)	B-nonzeros	longest col.
baxter	27441	11836	15128	109823	46
dano3mip_lp	3202	1224	13873	79656	625
dbir1	18804	384	27355	1067815	223
fxm3_16	41340	19927	64162	372452	36
gen4	1537	1536	4297	110174	96
nsct1	22901	421	14981	656259	617
nsir2	4453	113	5717	152746	235
nug12	3192	3192	8856	44244	24
qap12	3192	3192	8856	44244	24
rat1	3136	3136	9408	94539	27
rat7a	3136	3136	9408	275180	88
rlfddd	4050	0	57471	310044	48
rlfprim	58866	4204	8052	265975	200
seymour_lp	4944	0	1272	34921	273
world	35510	93	32734	220748	17

6 Nonlinearly constrained problems

Concerning nonlinearly constrained problems were is presently not yet such a large progress as with linearly constrained ones, although several successful solution approaches exist. The solution of small to medium-scale general nonlinear problems is nowadays routine. There exist commercial solvers with dense linear algebra, e.g. in the libraries NAG and IMSL and some more powerful ones which already use sparse matrix technology, also capable to deal with some thousand variables at least, see e.g. the optimization software guide by Moré and Wright [90] and the nonlinear programming FAQ [51]. There are even codes in the public domain which perform quite well [86], [87].

From the older approaches like penalty and the classical augmented Lagrangian methods there survived only the generalized reduced gradient method, e.g. the code `GRG2` of Lasdon (see e.g. [51]). This is an active set method and can be considered as an adaptation of the simplex method to nonlinear constraints. Typically the problem is formulated as

$$f(x) \stackrel{!}{=} \min\ ,\ h(x) \ = \ 0,\ x_{\mathcal{J}} \ \geq \ 0\ .$$

by introducing positive slacks for the original inequality constraints. Given a feasible point x a direction of descent d is computed which satisfies $(\nabla h(x))^T d \ = \ 0$ and $d_i \ = \ 0$ if $x_i \ = \ 0$ for $i \in \mathcal{J}$. For a prospective move

$$x \ \mapsto \ x + \sigma d$$

a restoration value $r(x, d, \sigma)$ is computed such that

$$h(x + \sigma d + r(x, d, \sigma)) \ \equiv \ 0\ , x + \sigma d + r(x, d, \sigma) \ \geq \ 0\ .$$

Of course $r(x, d, \sigma) \ = \ \mathcal{O}(\sigma^2)$. The computation of r requires the solution of a nonlinear system of equations for every instance of σ. σ itself is computed using e.g. backtracking for f along that arc. Freedom exists in defining d, it may be obtained by solving

$$\begin{pmatrix} A & B \\ B^T & O \end{pmatrix} \begin{pmatrix} d \\ -\begin{pmatrix} \mu \\ \lambda_{\mathcal{A}} \end{pmatrix} \end{pmatrix} = \begin{pmatrix} -\nabla f(x) \\ 0 \end{pmatrix} \tag{6.2}$$

with the sole restriction of A being positive definite on the kernel of B^T. Here A stands for the Hessian of the Lagrangian or a replacement thereof and B is $(\nabla h(x), (e^i)_{i \in \mathcal{A}})$. In addition one needs some inactivation scheme for eliminating indices from $\mathcal{A}$ based on the signs of the current multiplier estimates. This type of method is reliable but too expensive due to the maintenance of feasibility and the necessity of providing a feasible point first, which requires some phase I like method. Therefore other approaches are currently in the center of the interest. Especially *SQP* methods, which can be interpreted as modified Newton methods for the system *KTC* combined with some globalization scheme (mostly via a so called merit function) have shown enormous success for medium-scale problems.

However, for large-scale problems (with more than some thousand variables and even larger ones) the solution approaches have not yet matured. One might discern five different approaches to tackle such large-scale problems:

i Modification of well established methods simply by introducing sparse matrix techniques into the linear algebra part.

ii Transformation into bound constrained problems with subsequent use of the specialized solution techniques for the latter.

iii Methods of linearization *SLP*.

iv New variants of the *SQP* method.

v Direct adaptations of interior-point methods.

In the first group we find the further development of the code MINOS by Saunders [106], work of Gill, Murray and Saunders on SNOPT [58] (a special instance of a *SQP*-method which evolved from NPSOL) and for example the *SQP* methods of Nickel and Tolle [95], Betts and Frank [6], Bartholomew-Biggs and Hernandez [5], Ni [94] and Biegler and Nocedal [7].

MINOS is very efficient if there are only few and only mildly nonlinear constraints. Here a sequence of nonlinear but linearly constrained optimization problems is solved, following Robinson's method [103]. Globalization is obtained by the classical exterior quadratic penalty function. The linear constraints are dealt with using the elimination technique of the simplex method (indeed, MINOS is the long standing reference for an implementation of the simplex method.) If nonlinearity is more pronounced and there are many nonlinear constraints, this method is inferior to other approaches.

SNOPT takes advantage of eventual sparsity of the gradients of the constraints using sparse matrix methods for a factorization, which is used in the steps of a primal *QP* solver within a *SQP* method. Using this factorization a reduced Hessian is defined which is needed to solve the *KTC* like systems. It is normally a full matrix, hence either one restricts oneself to problems with a rather modest number of free variables (a case for which SNOPT was originally designed, a discretized control problem) or one resorts to the use of limited-memory quasi-Newton matrices.

Similarly Bartholomew-Biggs's approach is based on exploiting sparsity of constraint gradients and limited-memory approximations of the Hessian of the Lagrangian (this time the full matrix). However, as other users of the limited-memory approach too, the authors report about poor performance of this technique in the case of nonconvex or illconditioned testcases. This corresponds to the numerical experience of the present author.

The papers [4], [5] describe how sparse-matrix techniques can be used directly in the kernel of an otherwise unchanged *SQP*-code, the *QP* solver. Since one cannot assume that the Hessian of the Lagrangian is positive definite the arising systems are not quasi-definite as in interior-point methods for convex *QP*'s. The key role here play techniques for solving sparse symmetric indefinite systems, as developed by Duff, Reid and coworkers and gone into the codes MA27 and MA47 of the Harwell library. This technique also plays a key role in many other approaches. There is a disadvantage: the user must specify the nonzero patterns of the gradients and of the Hessian of the Lagrangian. If one uses a modeling system, this will be provided automatically, but otherwise it may be a prohibitively painstaking job. These systems have the form

$$\begin{pmatrix} A & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} d \\ w \end{pmatrix} = - \begin{pmatrix} g \\ c \end{pmatrix} .$$

A being sparse requires exact second derivatives or difference approximations thereof, since quasi-Newton updates will be full. Finite differences might be very costly, if possible at all. For problems of very high dimension the Bunch-Parlett decomposition is of little use because of the increased fill in involved with it and also because of numerical problems occuring if pivoting is limited. Iterative solvers for such systems are known "in principle", e.g. the classic SYMMLQ [97], see also [9] . However, due to problems with detecting rank deficiency and indefiniteness of the reduced Hessian they are not reliable. But there is an intense research in this area, see e.g. [53], [63]. The last cited work applies the conjugate gradient method to the system with the reduced Hessian without setting this up explicitly.

There is an even more problematic point. Indefinite *QP* problems which occur quite often in the *SQP* method if applied to nonconvex problems using the exact Hessian are regularized by a spectral shift, using a norm of the matrix or a similar crude eigenvalue bound. This is done for example by Franke [52] and by Conn, Gould and Toint in [29], with obviously bad outcome.

Ni's approach is a further development of the ideas of Nickel and Tolle. Here the inverse of the Hessian of the Lagrangian resp. of a positive definite replacement thereof is approximated by a limited-memory quasi-Newton update. The *QP* subproblem is solved via its dual. Problems with an incompatible primal *QP* are circumvented by regularizing the dual such that it always has a bounded solution.

Biegler, Nocedal und Schmid [7] consider the equality constrained case only. In principle they use the standard *SQP* approach, however decomposing the correction explicitly into a "vertical" and a "horizontal" part. (The "vertical" part here and at other places concerns always a correction of infeasibility. This correction is typically in the range space of the gradients of the (active) constraints). The essential contribution of this paper is the demonstration how to compute this decomposition efficiently in the large-scale case. Numerical results are presented in the followup paper [8].

6.1 Transformation into an only bound constrained problem

Transformation into an only bound constrained problem is the basic idea of the `LANCELOT` project of Conn, Gould and Toint [28]. One introduces slack variables into the general inequality constraints and obtains a problem with only equality constraints as general constraints and simple bound constraints as the only inequality constraints. For reasons of simple presentation we consider the case that all variables are bounded from below by zero.

$$(NLP:)\left\{\begin{array}{ccc} f(x) & = & \min \\ x & \geq & 0 \\ c(x) & = & 0 \end{array}\right. .$$

Then the `LANCELOT` (_large _and _nonlinearly _constrained _extended _Lagrangian _optimization _technique) approach is based on the following

Theorem 6.1 *Let x^* be a local solution of NLP with multipliers μ^* and λ^* for the equality and inequality constraints. Assume $\nabla c(x^*)$ being of full rank and $\nabla^2_{xx}L(x^*,\mu^*,\lambda^*)$ positive definite on the kernel of $\nabla c(x^*)^T$. Then there exists some $\gamma_0 > 0$ such that for $\gamma \geq \gamma_0$*

$$x(\mu) \stackrel{def}{=} argmin_x\{f(x) - (\mu)^T c(x) + \frac{\gamma}{2}||c(x)||^2 \ : \ x \geq 0\}$$

is well defined on $\mathcal{U}(\mu^*)$ as a function of μ and

$$\Psi(\mu) \stackrel{def}{=} f(x(\mu)) - (\mu)^T c(x(\mu)) + \frac{\gamma}{2}||c(x(\mu))||^2$$

posesses in μ^* an unconstrained strict local maximizer.

□

For given μ $x(\mu)$ will be computed by a bound constrained minimization and $\Psi(\mu)$ is simply the minimal value in this process. The function $\Psi(\mu)$ itself is maximized with respect to μ. It is of great help that $\nabla\Psi(\mu) = -c(x(\mu))$ and $\nabla^2\Psi(\mu) = -\frac{1}{\gamma}I + \mathcal{O}(\frac{1}{\gamma^2})$. This has the consequence that for sufficiently large γ even the simplest gradient method will converge quickly. All these relations hold for a general c indeed locally only, whereas for an originally convex problem the validity of the approach is global. There exist theoretical bounds for γ_0, but these cannot be evaluated in practice. They involve the singular values of ∇c and the eigenvalues of the Hessian and the reduced Hessian of the original Lagrangian function. This facts and the necessity to choose γ appropriately (large enough, but not too large, since otherwise the minimization will become troublesome) make the approach problematic. But for a long time this had been the only access to really large-scale problems. Meanwhile alternatives evolved, see below. If the required strong regularity assumptions are not satisfied, this will cause additional trouble, although a rank deficient ∇c doesn't hurt too much. The inner/outer iteration is expensive although the outer maximization converges fast usually. It is possible to increase the efficiency by using the problems structure introduced by the slacks. This is discussed for example in the paper [30]. The weak points of the extended Lagrangian approach are discussed in the book [113], section 3.5.2 and 3.5.3, and also in [68].

6.2 Linearization methods

The pure linearization methods (*SLP*-methods) use the fact that there exists a large choice of highly sophisticated software for *LP* problems which is capable of solving problems with up to millions of variables and constraints (sparsity assumed). One computes a direction of descent d^k for a given exact penalty function at a given guess x^k as a solution of the *LP*-problem

$$\begin{array}{rcl} \nabla f(x^k)^T d + \gamma w &=& \min\limits_{d,w} \ , \\ -g(x^k) - \nabla g(x^k)^T d &\le& w \ , \\ -h(x^k) - \nabla h(x^k)^T d &\le& w \ , \\ h(x^k) + \nabla h(x^k)^T d &\le& w \ , \\ w &\ge& 0 \ , \end{array}$$
$$-\rho_k \ \le d_i^k \ \le \rho_k \ , \ i = 1, \ldots, n \ .$$

Here w is a slack variable, which serves the purpose to make the linear constraints of this subproblem always compatible, such that the subproblem is solvable, due to the artificial compactness conditions on d^k. One aims in choosing γ sufficiently large, such that $w = 0$ is achieved whenever this is possible. ρ_k is the so called "trust-region radius", which is computed adaptively, depending on the descent obtained for the penalty function (in this case the l_∞ exact penaltyfunction.) This method cannot converge fast, it corresponds to the ordinary gradient descent in unconstrained minimization. But there exists a lot of enhancements, see e.g. [47], [75].

6.3 Modifications of the *SQP* method

The articles [64] and [91] describe existing possibilities and the second one also gives some review of existing software. We restrict ourselves here to a short review of the most promising approaches for which there is some numerical experience.

Franke [52] uses a classical *SQP* method, essentially Schittkowski's method [109]. The emerging *QP* subproblem

$$\frac{1}{2}x^T A x - a \ \stackrel{!}{=} \ \min \ , \ B^T x - b \ = \ 0, \ C^T x - c \ \ge \ 0$$

is then solved by an interior-point method, considering the parametrized *KTC* system

$$\begin{array}{rcl} Ax - a - B\mu - C\lambda &=& 0 \ , \\ B^T x - b &=& 0 \ , \\ w = C^T x - c &>& 0 \ , \\ \lambda &>& 0 \ , \\ \Lambda w - \epsilon e &=& 0 \end{array}$$

with parameter ϵ, $\epsilon \to 0$. This nonlinear system is solved approximately via Newton's method. The resulting linear systems are solved by a modification of the sparse Bunch-Parlett code from the `MESCHACH`-library of Steward and Leyk (to be found as a tar-file in `netlib/cephes`). Here A represents the Hessian of the Lagrangian. If this matrix turns out to be indefinite, then Franke regularizes it as done in [4], [29] using estimates of the eigenvalues provided by Gerschgorin discs. As is to be expected the numerical results are disappointing, if this occurs. As a further severe restriction Franke's approach misses a systematic treatment of inconsistent *QP*-Problems, which occur rather often in nonconvex cases. The approach has gone into a code `HQP` (see [87]) which solved a bunch of highdimensional problems from the `CUTE` collection successfully.

Plantenga [99] first transforms a general *NLP* into an equality constrained one with additional bound constraints, using slack variables for the inequality constraints. For reasons of simple presentation we again assume that all variables are subject to the positivity constraints:

$$\begin{array}{rcl} f(x) &\stackrel{!}{=}& \min \ , \\ c(x) &=& 0 \ , \\ x &\ge& 0 \ . \end{array}$$

The solution of this problem is then approximated by the minimization of a barrier function subject to the equality constraints:

$$\begin{aligned} f(x) - \epsilon \sum_{i=1}^{n} \ln(x_i) &\stackrel{!}{=} \min \ , \\ c(x) &= 0 \ , \\ x &> 0 \ , \end{aligned}$$

that means the barrier technique here is solved at the outer stage as opposed to Franke's approach, which uses this at the inner stage. The resulting equality constrained problem in turn is solved using the method of Byrd und Omojokun. This latter is a specialized method for equality constrained problems, a variant of Burke's method [16]. This is a trust-region method.

In every step two QP problems with an additional trust-region constraints have to be solved.

$$||\nabla c(x^k)^T v + c(x^k)||_2^2 \stackrel{!}{=} \min \ , \tag{6.3}$$
$$||v||_2^2 \leq \zeta \Delta_k^2 \ ,$$
$$\nabla f(x^k)^T d + \frac{1}{2} d^T G_k d \stackrel{!}{=} \min \ , \tag{6.4}$$
$$\nabla c(x^k)^T d = \nabla c(x^k)^T v^k \ , \tag{6.5}$$
$$||d||_2^2 \leq \Delta_k^2 \ .$$

Here v^k is the solution of (6.3). This is an example of the so called "vertical" correction (hence the name). This vertical correction serves the purpose of diminishing the infeasibility. The parameter ζ is chosen arbitrarily, but fixed in $]0,1[$. In (6.4), (6.5) d is decomposed into

$$d = v^k + Z_k u$$

where Z_k is a basis of the kernel of $\nabla c(x^k)^T$. Then the condition (6.5) is satisfied automatically. In order to compute such a basis for the problem (6.4) one uses sparse matrix techniques, e.g. the code `MA28` of the Harwell library. The trust-region radius Δ_k is computed adaptively as usual, checking descent for the exact penalty function

$$f(x) - \epsilon \sum_{i=1}^{n} \ln(x_i) + \gamma ||c(x)||_2$$

and ϵ is diminished in the outer iteration. The first version of Plantenga showed performance comparable with `LANCELOT`. A detailed exposition is to be found in the paper [80]. This method has been further developed by Hribar [73].

A completely different approach is taken by Boggs and coworkers, [10], [11], [12]. The algorithm can be used for problems with mixed constraints, but like the authors in their papers we restrict ourselves here to the inequality constrained case, changing also the formulation into our standard form with constraints of the form $g(x) \geq 0$. Therefore the original problem now reads

$$f(x^I) \stackrel{!}{=} \min \ , g(x^I) \geq 0 \ .$$

After introducing slacks $x^{II} \geq 0$ the problem becomes formally one with equality constraints, the difference lying in the fact that the slacks are kept strictly positive.

$$x = (x^I, x^{II}) \ , c(x) = g(x^I) - x^{II} = 0 \ .$$

The backbone of the method is the classical SQP method with Fletchers differentiable penalty function as a merit function. This is the function

$$F(x) = f(x) - \mu(x)^T c(x) + \frac{\gamma}{2} ||A(x)^{1/2} c(x)||^2$$

where

$$\mu(x) \;=\; (\nabla c(x)^T \nabla c(x))^{-1} \nabla c(x)^T \nabla f(x) \ .$$

Since the QP solver used is an interior-point method which maintains strict positivity of the slacks (by means of the stepsize selection) , these can be eliminated from the computation. That means that no increase of dimension occurs here. Infeasible initial values are treated by a "big M" method, the QP problems being of the type

$$\begin{aligned} c^T d + \tfrac{1}{2} d^T A d + M\theta &\stackrel{!}{=} \min \ , \\ B^T d + b + \theta e &\geq 0 \ , \\ \theta &\geq 0 \ . \end{aligned}$$

If the algorithm is to be used with an equality constrained problem, then this "big M" phase persists throughout (since a strictly feasible point for the original problem is never found). During the stepsize algorithm the estimate $\mu(x)$ for the multipliers and the weight matrix $A(x)$ for the quadratic penalty term are frozen in order to reduce the evaluation effort for the Fletcher-function. Nevertheless every function value requires the solution of a linear system with the matrix

$$(\nabla g^T(x^k) \nabla g(x^k) + Z_k),$$

where g is the vector of inequality constraints and Z_k the diagonal matrix built from the values of the slack variables $x^{k,II}$. This matrix is always positive definite, but may be illconditioned. This introduces some trouble if the system must be solved iteratively. Furthermore the parameter γ in Fletchers function is not so easy to obtain, which produces even more trouble for nonconvex problems.

The QP solver itself is an inexact one and works with small dimensional subproblems of dimension up to three, which in turn are solved exactly. The basis of these subspaces is spanned by solutions of systems of the form

$$(\nabla g(x^k) Z_k^{-2} \nabla g^T(x^k) + Q_k/\beta_k) p_{i,k} \;=\; t_{i,k}, \ i = 1, 2, 3$$

where Q_k is an approximation for the Hessian of the Lagrangian, and β_k is chosen sufficiently large. These are systems of exactly the same structure as they occur when applying the classical barrier method for inequality constrained problems in connection with Newtons method. Should Q_k turn out to be not positive definite then also directions of negative curvature for the Lagrangian are used. The paper [12] contains an overview over numerical results obtained so far.

In the approach of Felkel and Spellucci the starting point again is a classical SQP method, this time using the nondifferentiable exact penalty function

$$\Phi(x; \gamma) \;=\; f(x) + \gamma ||h(x), g(x)^-||_\infty$$

for a problem with mixed constraints, with γ chosen adaptively. This function has the advantage of requiring minimal global conditions on the problem. Its clear disadvantage is its scale dependency, which requires adaptive and dynamic scaling of the problem. For this task a satisfactory practical solution was given already in [71].

A direction of descent for this function is obtained from the QP problem

$$\begin{aligned} \frac{1}{2} d^T B_k d + \nabla f(x^k)^T d + \gamma w + \frac{1}{2}\alpha w^2 &= \min \ , \\ -g(x^k) - \nabla g(x^k)^T d &\leq we \ , \\ -h(x^k) - \nabla h(x^k)^T d &\leq we \ , \\ h(x^k) + \nabla h(x^k)^T d &\leq we \ , \\ w &\geq 0 \ , \end{aligned}$$

where w is a single slack variable which makes the problem compatible. As usual, $e = (1, \ldots, 1)^T$. But of course one wishes to obtain $w = 0$ which in turn guarantees that d^k is also a direction of descent for the penalty term alone. This is obtained by increasing γ cautiously. The *QP* subproblem is not solved exactly as in [71], but approximately only minimizing a shifted logarithmic barrier function for this problem (observe that there are only inequality constraints due to the transformation of an equality constraint into two inequalites with slack), maintaining the constraint $w \geq 0$. For a restriction $r_i(d)$ of the *QP* problem the barrier term is

$$b(r_i, s, \eta) = \begin{cases} \ln(r_i + s) & \text{if } r_i(x) > -s\eta \\ \alpha_0 + \alpha_1 r_i + \alpha_2 r_i^2 + \alpha_3 r_i^3 & \text{otherwise} \end{cases} ,$$

where the parameters are chosen such that C^3 continuity is satisfied. Here $s > 0$ is the shift, which must be chosen sufficiently small but must not tend to zero and η is a parameter. This avoids the singular behaviour at the solution, [101].

This minimization is accomplished by an inexact Newton method based on the Lanczos algorithm. This *QP* solver has first been tested as a stand alone solver for *QP* problems with great success, see [38]. In the current version of the *NLP* solver there are to be observed several inefficiencies. These occur if the matrices B_k are not sufficiently positive definite or if there is a large number of strongly nonlinear equality constraints. In the last case there occur steeply grooved and narrow valleys (due to the choice of a possible large γ and a small slack w), which make the method slow even if combined with second order corrections. Also the descent property of d^k is sometimes lost if the solution precision in the *QP* solver is relaxed. Improvements are a theme of current research.

The filter*SQP* method of Fletcher and Leyffer [43], [44], [45], [46] avoids the usage of a merit function at all and thereby the complications involved in chosing the penalty parameters appropriately. Rather it considers *NLP* as a special instance of vector optimization

$$\begin{pmatrix} f(x^*) \\ \theta(x^*) \end{pmatrix} \overset{!}{\leq} \begin{pmatrix} f(x) \\ \theta(x) \end{pmatrix} ,$$

where

$$\theta(x) \overset{def}{=} \max\{||h(x)||_\infty, ||(g(x))^-||_\infty\} .$$

Two types of moves are considered. If the trust-region *QP* problem

$$\begin{aligned} m_k(d) &\overset{def}{=} f(x^k) + \nabla f(x^k)^T d + \tfrac{1}{2} d^T A_k d \overset{!}{=} \min_d \\ & g(x^k) + \nabla g(x^k)^T d \geq 0 , \\ & h(x^k) + \nabla h(x^k)^T d = 0 , \\ & ||d|| \leq \Delta_k , \end{aligned}$$

with the trust-region radius Δ_k is solvable, then its solution is taken as a prospective move and its acceptability is checked, otherwise a restoration move is computed, aiming in diminishing θ. This might be e.g. a solution of

$$\begin{aligned} w &\overset{!}{=} \min \\ g(x^k) + \nabla g(x^k)^T d + we &\geq 0 , \\ h(x^k) + \nabla h(x^k)^T d + we &\geq 0 , \\ -h(x^k) - \nabla h(x^k)^T d + we &\geq 0 , \\ ||d|| &\leq \Delta_k . \end{aligned}$$

The acceptance of a move and the trust-region radius are controlled by a so called "filter". This is a set of values $(f(x^s), \theta(x^s))$ none of which is dominated by the others in the sense of the half order of $\mathbb{R}^2$ and with $\theta(x^s) \neq 0$. If the *QP* problem is infeasible, the corresponding values are always added to the filter and a move is made by a restoration step, aiming in computing some pair (x^{k+1}, Δ_{k+1}) such that the corresponding new *QP* problem becomes feasible. Otherwise the values at x^k are added to the filter if

$$m_k(0) - m_k(d^k) \leq \kappa\theta_k^2$$

for some constant $\kappa > 0$. The point $x^{k+1} = x^k + d^k$ is accepted if

$$\theta(x^{k+1}) \;\le\; (1-\gamma_\theta)\theta_j \quad \text{or} \quad f(x^{k+1}) \;\le\; f_j - \gamma_\theta \theta_j$$

for all values in the filter and if in addition

$$\frac{f(x^k) - f(x^{k+1})}{m_k(0) - m_k(d^k)} \;\ge\; \eta_1 \;>\; 0 \;.$$

In this case Δ_{k+1} can be increased, in all other cases Δ_{k+1} is decreased. γ_θ is chosen arbitrarily but fixed in $]0,1[$. After adding a pair (θ_k, f_k) to the filter this is purged by deleting all pairs with $\theta_j \;\ge\; \gamma_\theta \theta_k$ <u>and</u> $f_j \;\ge\; f_k - \gamma_\theta \theta_k$. This method is quite flexible, in principle it uses a merit function whose weights may change every step (via scalarization of the vector optimization problem). The code `filter` can be used via `NEOS`.

6.4 Adaptation of interior-point methods

The modified *SQP* methods described above result in multistage algorithms of inner/outer iteration type. Since the *QP* problems inherit the combinatorial nature of the original problem, their solution itself may be quite costly, especially in the large-scale case. Therefore there was much effort in avoiding this by using some more direct approach.

Conn, Gould and Toint [29] describe a new access for minimizing general nonlinear functions subject to linear equality and bound constraints. Nonlinear constraints could be incorporated via an augented Lagrangian. The problem

$$\begin{aligned} f(x) &\stackrel{!}{=} \min{}_x \;, \\ B^T x &= b \;, \\ x &\ge 0 \end{aligned}$$

is embedded into a higher dimensional one

$$\begin{aligned} f(x) + \frac{1}{2}\rho(\xi+1)^2 &\stackrel{!}{=} \min_{x,\xi} \;, \\ B^T x - b &= \xi(B^T x^0 - b) \;, \\ x &\ge 0 \;, \\ \xi &\ge 0 \;, \end{aligned}$$

where the penalty parameter ρ is fixed in principle but has to be determined adaptively. The Kuhn-Tucker conditions for this extended problem become parametrized and the parametric systems are solved by Newton's method, with the Bunch-Parlett-decomposition as linear solver:

$$\begin{aligned} \nabla f(x) - B\mu - \lambda &= 0 \;, \\ -(B^T x^0 - b)^T \mu + \rho(\xi+1) &= 0 \;, \\ B^T x - b - \xi r^0 &= 0 \;, \\ X\Lambda e - \epsilon e &= 0 \;, \\ X, \Lambda &> 0 \;. \end{aligned}$$

The logarithmic barrier function

$$f(x) + \frac{1}{2}\rho(\xi+1)^2 - \epsilon \sum_{i=1}^{n} \ln(x_i)$$

serves as a merit function for stepsize control. In the case $\nabla^2 f(x)$ not positive definite it is regularized by the additive term $||\nabla^2 f(x)||I$. Numerical results are reported for large *QP* problems. In the convex case the results are very good, compared e.g. with Goulds direct method , code `VE09`, but for nonconvex cases they are completely disappointing, as is to be expected from this crude regularization. Divising a good adaptive scheme for the penalty parameter ρ is difficult. It depends besides others on the reciprocal value of the smallest <u>positive</u> eigenvalue of the indefinite matrix

$$\begin{pmatrix} \nabla^2 f(x) & B \\ B^T & O \end{pmatrix} .$$

There one also tried to make use of the classical barrier and penalty functions more directly using some modifications for large dimensional problems in the hope to take advantage of the progress made for highdimensional unconstrained minimization, e.g. Nash and Sofer [93] or Shanno and Breitfeld [15]. The results obtained this way are not overly encouraging. The main reason for this lies in the fact that one loses too much of the "centering" when the penalty parameter is changed. Along the central trajectory the behaviour of these functions is much more amenable than a bit outside, especially if one uses quasi Newton or conjugate gradient type methods.

Forsgren and Gill [50] use the minimization of a mixed penalty-barrier function which was already described by Fiacco and McCormick. One of the problems in connection with this function is the fact that the theoretical multiplier estimates provided by these functions, e.g.

$$\frac{\rho}{g_i(x(\rho))} \approx \lambda_i$$

are quite unreliable in practice. The authors try to alleviate this effect by adding penalty terms for these deviations in a primal-dual framework:

$$\begin{aligned} \Psi(x,\mu,\lambda;\epsilon,\gamma) &= f(x) + \frac{1}{2\epsilon}\sum_{i=1}^{p}(c_i(x)^2 + \gamma(c_i(x)+\epsilon\mu_i)^2) \\ &\quad -\epsilon\sum_{i=1}^{n}(\ln(x_i) + \gamma(\ln(\frac{x_i\lambda_i}{\epsilon}) + 1 - \frac{x_i\lambda_i}{\epsilon})) . \end{aligned}$$

This function is minimized simultaneously with respect to x, μ and λ. ϵ is the path parameter and we have $\epsilon \to 0$ but $\gamma > 0$ fixed. For ϵ fixed minimization is performed by the damped Newton method, making use also of directions of negative curvature, which are available through the Bunch-Parlett decomposition of the Hessian. A special pivoting rule for this decomposition is at the heart of their algorithm. Thereby convergence to second order stationary points can be guaranteed. Numerical results are described in [81]. The outcome of Laux's experiments is a severe scale dependency of the chosen penalty function. Furthermore a good choice of the parameters involved is problematic and last not least, a point satisfying the inequality constraints must be known as an initial guess.

In the work [19] a method is investigated which treats only the slack variables by an interior-point approach. A general *NLP* problem

$$f(x) \stackrel{!}{=} \min\ ,\ c_E(x) = 0\ ,\ c_I(x) \geq 0$$

is transformed into

$$f(x) \stackrel{!}{=} \min\ ,\ c_E(x) = 0\ ,\ c_I(x) - s = 0\ ,\ s \geq 0\ ,$$

and this in turn is embedded in a family of equality constrained problems

$$\begin{aligned} f(x) + \epsilon\sum_{i=1}^{m}\ln(s_i) &\stackrel{!}{=} \min\ , \\ c_E(x) &= 0\ , \\ c_I(x) - s &= 0\ . \end{aligned}$$

For ϵ fixed this is an equality constrained nonlinear optimization problem which is solved by a *SQP* method. A trust-region approach is taken. It turns out that in constructing the trust-region the slack variables need a special treatment, namely using a norm like

$$|| \begin{pmatrix} d_x \\ \max\{\beta, \Delta\} S^{-1} d_s \end{pmatrix} || \leq \Delta$$

where d_x designates the change in x and d_s the change in s and, as usual, $S = \mathrm{diag}(s_1, \ldots, s_I)$. β is a chosen constant. A quite general convergence theory is developed. These investigations are continued in the paper by Byrd, Hribar and Nocedal [20]. There comes up an essentially new aspect, namely the formulation of an adequate quadratic model for the Lagrangian of the barrier-problem with respect to the slack variable s. It turns out that the obvious choice

$$\mu S^{-2}$$

for the quadratic part corresponding to s is not favourable, since it gives directions which tend to violate feasibility of s and hence enforce a reduction of the step size. Much better seems to be the choice

$$S^{-1}\Lambda \ ,$$

where Λ designates the diagonal matrix with the multiplier estimates for the equality constraints $c_i(x) + s = 0$. These estimates are computed separately using the least squares estimate from the *KTC* conditions for x and s fixed. The paper contains detailed discussion how to solve the subproblems efficiently and approximately only. Promising numerical results are presented for the implementation `NITRO`. `NITRO` can be tested via the `NEOS` submission tool.

Gay, Overton and M. Wright [56] consider a general *NLP* problem with mixed equality and inequality constraints. Different from their competitors they treat the inequality constraints directly by barrier terms. Hence

$$f(x) \stackrel{!}{=} \min \ , \ h(x) \ = \ 0 \ , \ g(x) \ \geq \ 0$$

is solved via

$$f(x) - \epsilon \sum_i \ln(g_i(x)) \ \stackrel{!}{=} \ \min \ , \ h(x) \ = \ 0$$

with $\epsilon \to 0$. The parameter dependent *KTC* conditions

$$\begin{aligned} \nabla h(x) y + \nabla g(x) z - \nabla f(x) &= 0 \ , \\ h(x) &= 0 \ , \\ G(x) z - \epsilon e &= 0 \ , \ G = \mathrm{diag}(g_i(x)) \end{aligned}$$

are solved by Newtons method, after symmetrization and partial elimination:

$$\begin{pmatrix} -K & A \\ A^T & O \end{pmatrix} \begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{pmatrix} \nabla f - Ay - \epsilon B^T G^{-1} e \\ -h \end{pmatrix} \text{ with } K \ = \ H + B^T G^{-1} Z B \ .$$

An indefinite K becomes regularized to definiteness, but such that its eigenvalues tending to infinity are not touched. Two merit functions are used, namely the norm of the right hand side of the Newton equation and periodically also some type of extended Lagrangian

$$f(x) - \epsilon \sum_i \ln g_i(x) - y^T h(x) + \frac{c}{2} ||h(x)||^2 \ ,$$

(with c finite and in principle fixed). This device serves the purpose to prevent convergence of Newtons method to saddle points or maxima. Numerical results for low dimensional problems are reported. No complete convergence analysis seems to exist.

Vanberbei and Shanno [117] make use of the powerful *QP* solver `LOQO` of Vanderbei. It serves at the heart of a *SQP* code. Here too inequality constraints $g(x) \geq 0$ are transformed into equations using positive slacks which in turn are treated by the logarithmic barrier term. The method computes directions of descent for the inexact penalty barrier function

$$f(x) - \epsilon \sum_{i=1}^{m} \ln(s_i) + \frac{\beta}{2} ||g(x) - s||^2$$

from the reduced system

$$\begin{pmatrix} H & O & -A \\ O & -S^{-1}\Lambda & I \\ -A^T & I & O \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta s \\ \Delta \lambda \end{pmatrix} = \begin{pmatrix} -\nabla f + A\lambda \\ \epsilon S^{-1} e - \lambda \\ g(x) - s \end{pmatrix} , \tag{6.6}$$

where $A = \nabla g(x)$ and $H = \nabla^2_{xx} L$. This descent property is obtained for β sufficiently large. Indefinite H is regularized by addition of a suitable multiple of the unit matrix. ϵ is chosen dependent on the duality gap:

$$\epsilon = \gamma \min\{(1-r)\frac{1-\xi}{\xi}, 2\}^3 \frac{s^T\lambda}{I}$$

with

$$0 < r < 1 , \quad \xi = \frac{\min_i s_i \lambda_i}{s^T \lambda / I} .$$

(Remember that I is the number of inequality constraints). In principle we must have $\beta \to \infty$, but since in practice one contents oneself with an approximate solution anyway the authors consider this fact as not so critical. The paper also describes a special treatment of bound- and interval constraints such that only general inequalities must be transformed by slacks. Equality constraints are dealt with directly by a quadratic penalty term. The paper also reports numerical experience. `LOQO` can also be used through `NEOS`.

Akrotirianakis and Rustem [1] describe an approach similar to the one given by Vanderbei und Shanno. Starting point is a *NLP* problem

$$f(x) \overset{!}{=} \min, \; g(x) = 0 , \; x \geq 0 .$$

As a merit function serves the mixed penalty-barrier function

$$\Phi(x; c; \epsilon) = f(x) - \epsilon \sum_{i=1}^{n} \ln(x_i) + \frac{c}{2} ||g(x)||^2$$

subject to the equality constraints

$$g(x) = 0 ,$$

that means the equality constraints are treated in dublicate: they enter the merit function and are posed explicitly. The barrier parameter ϵ is controlled in an outer and the penalty parameter c in an inner iteration. For ϵ and c fixed the directions of descent come from a Newton step for a zero of the function

$$F(x, y, z; c, \epsilon) = \begin{pmatrix} \nabla f(x) - z + c(\nabla g(x)) g(x) - \nabla g(x) y \\ g(x) \\ XZe - \epsilon e \end{pmatrix}$$

Here y represents the vector of Lagrange multipliers corresponding to the equality constraints and z is obtained as a new free variable from the relation

$$z = \epsilon X^{-1} e .$$

As usual $e = (1, \ldots, 1)^T$. z corresponds to the Lagrange multipliers of the bounds. For sufficiently large c the descent property of this direction can be guaranteed and the adaptive computation of c is based on that fact. The Jacobian of F has as left upper block

$$H = \nabla^2 f(x) - y \times \nabla^2 g(x) + c \nabla g(x) (\nabla g(x))^T + c g(x) \times \nabla^2 g(x)$$

$H + X^{-1}Z$ must be invertible in order to have a well defined direction d. This is of course a restrictive condition, which also occurs in other work. Even stronger, uniform boundedness and uniform positive definiteness is assumed for this matrix in the convergence analysis. Different stepsizes are used for the primal and the dual variables. For a special version of the algorithm there is a proof of convergence. No numerical results are presented. The proof of the uniform boundedness of c needed for that seems to be incomplete.

6.5 Homotopy methods for the KTC-system

Thereas the work of the previous section results in some inner/outer iteration scheme with the possibility to choose subalgorithms independently it is also possible to use the primal-dual interior-point formulation known from convex *LP* and *QP* directly for a general *NLP*. This is shown in the interesting paper of El Bakry et. al. in [32]. These investigations have been continued by Durazzi [36]. The procedure is best interpreted as a transfer of the Kojima, Mizuno and Yoshise method from the *LP* to the the *NLP* case. As is known that one results in the *LP* case to the most efficient solvers presently known. Starting point is a general problem with mixed constraints $h(x) = 0$ and $g(x) \geq 0$. The trajectory defined by the system

$$
\begin{aligned}
\nabla f(x) - \nabla h(x)\mu - \nabla g(x)\lambda &= 0\,, \\
\lambda - z &= 0\,, \\
h(x) &= 0\,, \\
g(x) - s &= 0\,, \\
ZSe - \epsilon e &= 0\,,
\end{aligned}
$$

formally written as $(x, \mu, \lambda, z, s)(\epsilon)$ with $\lambda, z, s > 0$ is traced with $\epsilon \rightarrow 0$, the nonlinear system in turn being approximately solved by Newton's method. The Jacobian of this system can be symmetrized and once again one has to solve a large symmetric indefinite linear system each step. Some numerical evidence with small-scale problems is provided. There are some critical points. On point is the large number of artificial unknowns (a total of $n + 3m + p$, if m is the dimension of g and p that of h). The structure of the linear system is much less amenable than in the *LP* case. The convergence conditions are quite strong and seem to restrict the treatment to convex problems. Among other things one needs regularity of the matrix

$$\nabla^2_{xx} L(x, \mu, \lambda) + \nabla g(x) S^{-1} Z \nabla g(x)^T$$

for all values of x, μ, λ, z, s in a neighborhood of the trajectory.

A quite similar approach is followed by Epelly, Gondzio and Vial [37] who restrict themselves to the convex case from the very beginning.

7 Some numerical results

In the following we give an excerpt of the results obtained by the `COPS` project [31]. The test comprises four solvers, `LANCELOT, LOQO, MINOS` and `SNOPT` in their newest version (2000). Default options were used for all codes. The problems are formulated in `AMPL`. Most of them are discretized control problems. There are also three large *QP* problems in the testsuite. The evaluation was done on a SUN ULTRA SPARC 2 under Solaris 7. We give here a part of the results only, for the lowest and the highest dimension tested. The table gives computing time in seconds and the optimal value of the objective function. "F" denotes a failure (either premature termination or time out). The numbers `x-y-z` give the number of variables, of the contraints and of the bound constraints.

problem/solver	`LANCELOT`	`LOQO`	`MINOS`	`SNOPT`
largest small polygon	12.83	3.49	2.11	1.14
50-49-50	7.9715e-01	7.64714e-01	7.64383e-01	7.79740e-01
largest small polygon	F	F	344.81	256.6
200-199-200			7.72803e-01	7.85040e-01
electron on sphere	3.98	0.84	6.22	9.65
75-25-0	2.43812e+02	2.43812e+02	2.43812e+02	2.43812e+02
electron on sphere	371.6	2437.78	F	1600.48
600-200-0	1.84389e+04	1.84389e+04		1.84390e+04
hanging chain	13.91	17.24	1.22	5.72
100-51-0	5.07230e+00	5.07226e+00	5.07226e+00	5.07226e+00
hanging chain	577.24	1028.35	73.9	F
800-401-0	5.06788e+00	5.06862e+00	5.06862e+00	
shape of cam	42.74	0.68	0.87	0.6
100-202-100	4.30178e+00(F)	4.28414e+00	4.28414e+00	4.28414e+00
shape of cam	1887.03	12.47	21.72	25.17
800-1602-800	4.85693e+00(F)	4.27427e+00	4.27426e+00	4.23739e+00
alpha-pinene	1426.01	28.85	1.98	3.74
505-500-5	1.96766e+01	1.98715e+01	1.98715e+01	1.98715e+01
alpha-pinene	F	16.87	F	235.44
4005-4000-5		1.98721e+01		1.98721e+01
marine population	623.75	2.07	6.58	85.37
615-592-15	1.97522e+07	1.97522e+07	1.97522e+07	1.97522e+07
marine population	F	38.4	F	1502.26
4815-4792-15		1.97465e+07		1.97465e+07
flow in channel	F	1.55	1.09	2.14
400-400-0		1.00000	1.00000	1.00000
flow in channel	F	22.54	32.65	98.5
3200-3200-0		1.00000	1.00000	1.00000
robot arm	F	1.03	2.82	10.22
460-300-306		9.14687	9.14687	9.14687
robot arm	F	F	161.18	2671.63
3610-2400-2406			9.14108	9.14101
particle steering	30.83	923.3	1.58	3.25
256-200-51	5.54672e-01	5.54668e-01	5.54668e-01	5.54668e-01
particle steering	2997.88	F	143.09	147.37
2006-1600-401	5.54552e-01		5.54572e-01	5.54573e-01
goddar rocket	F	3.34	1.69	3.04
205-150-153		1.01281	1.01280	1.01282
goddar rocket	F	12.42	F	64.48
1605-1200-1203		1.01283	F	1.01283
hang glider	F	F	F	11.14
256-200-153				1.28239e+03
hang glider	F	F	F	1268.67
2006-1601-1203				1.24797e+03
cracking oil	918.28	1.37	5.04	5.25
503-500-3	5.23633e-03	5.23664e-03	5.23664e-03	5.23664e-03
cracking oil	F	49.62	161.99	179.71
4003-4000-3		5.23659e-03	5.23659e-03	5.23659e-03
methanol to hydrocarbon	196.62	2.13	5.05	12.92
605-600-5	9.02300e-03	9.02229e-03	9.02228e-03	9.02228e-03
methanol to hydrocarbon	F	45.2	263.67	512.16
4805-4800-5		9.02229e-03	9.02228e-03	9.02228e-03

problem/solver	LANCELOT	LOQO	MINOS	SNOPT
catalyst mixing	7.71	0.66	2.39	3.99
303-200-101	-4.77480e-02	-4.80694e-02	-4.80605e-02	-4.80579e-02
catalyst mixing	424.61	8.25	17.88	181.24
2403-1600-801	-4.71856e-02	-4.80559e-02	-4.74787e-02	-4.80451e-02
elastic plastic torsion	3.01	2.99	108.31	125.58
1250-0-1250	-4.17510e-01	-4.17510e-01	-4.17510e-01	-4.17510e-01
elastic plastic torsion	17.19	15.86	F	F
5000-0-5000	-4.18239e-01	-4.18239e-01		
journal bearing	3.02	3.36	173.65	722.68
1250-0-1250	-1.54015e-01	-1.54015e-01	-1.54015e-01	-1.54015e-01
journal bearing	17.89	13.33	F	F
5000-0-5000	-1.55042e-01	-1.55042e-01		
obstacle	2.77	2.98	103.76	137.68
1250-0-1250	2.51948	2.51948	2.51948	2.51948
obstacle	16.33	F	F	F
5000-0-5000	2.50694e+00			

Besides these tests there are also many interesting benchmarks provided by H.D.Mittelmann [86], showing the same effect: there is no clear cut winner and reliability of the codes is far beyond the level meanwhile reached for small- and medium-scale problems. Hence much remains to be done.

8 Conclusion

The great impact of interior-point methods on present optimization technology can clearly be seen from the discussion above. Summarizing it can be safely said that convex problems, be it *QP* or more general *NLP* problems, can be solved by a variety of methods with good success, even in the large-scale case. As T. Rockafellar states: "The great watershed in optimization isn't between linearity and nonlinearity but convexity and nonconvexity" [105]. First of all we must be prepared to accept a local solution, possibly much weaker than the desired global one. Which of the possibly many local solutions is identified is subject to details of the algorithm's implementation and the initial guess and it is almost impossible to take influence on that. For a nonconvex problem often some kind of restoration steps must be used. This typically requires the linear independence of the gradients of the violated constraints or a least the selection of a subset of constraints which satisfy this condition. In the small-scale area the *QR* or the *SVD* decomposition of the matrix of these gradients helps to do that. This technique cannot be used for large-scale applications because the enormous fill in produced by orthogonal transformations. Other linear algebra techniques are much less reliable. The detection of infeasibility cannot be based on the unboundedness of the dual variables. The detection of nonconvexity via detection of indefiniteness of the Hessian of the Lagrangian is an unreliable process, especially for large scale applications. How to regularize a nonconvex problem, especially in connection with interior-point methods, is an open problem. The crude technique of addition of a multiple of the unit matrix is clearly inadequate. Hence much remains to be done in order to increase the reliability and efficiency of existing methods.

References

[1] I. Akrotirianakis, B. Rustem: *A globally convergent interior-point algorithm for general nonlinear programming problems.* Technical Report 97/14. Department of Computing. Imperial College, London.

[2] M. Al-Baali: *Descent property and global convergence of the Fletcher-Reeves method with inexact line searches.* J.I.M.A. Num. Anal. 5, (1985), 121–124.

[3] B. M. Averick, J. J. Moré: *Evaluation of large-scale optimization problems on vector and parallel architectures* Preprint MCS-P379-0893, Argonne National Laboratory, Argonne, Illinois, 1993. Software in info.mcs.anl.gov: /pub/MINPACK-2.

[4] M. C. Bartholomew-Biggs, M. de F. G. Hernandez: *Using the KKT-matrix in the augmented Lagrangian SQP method for sparse constrained optimization.* J.O.T.A. 85, (1995), 201–220 .

[5] M. C. Bartholomew-Biggs, M. de F. G. Hernandez: *Modifications to the subroutine OPALQP for dealing with large problems.* J. Econ. Dyn. Control 18, (1994), 185–203 .

[6] J. T. Betts, P. D. Frank: *A sparse nonlinear optimization algorithm.* J.O.T.A. 82, (1994), 519–541.

[7] L.T. Biegler, J. Nocedal, C. Schmid: *A reduced Hessian method for large-scale constrained optimization*, SIOPT5, (1995), 314–347

[8] L.T. Biegler, J. Nocedal, C. Schmid, D. Ternet: *Numerical experience with a reduced Hessian method for large-scale constrained optimization.* Report 97/05. Optimization Technology Center, Northwestern University, Evanston.

[9] Å. Bjoerck: *Numerical stability of methods for solving augmented systems.* American Mathematical Society. Contemp. Math. 204, 51–60 (1997).

[10] P. T. Boggs, P. T. Domich , J. E. Rogers: *An interior-point method for large-scale quadratic programming problems.* Ann. Oper. Res. 62, (1996), 419–437 .

[11] P. T. Boggs, A. J. Kearsley, J. W. Tolle: *A truncated SQP algorithm for large-scale nonlinear programming problems.* pp 69–77 in "Advances in Optimization and Numerical Analysis", S. Gomez and J.P. Hennart (eds.), Kluwer 1994.

[12] P. T. Boggs, A. J. Kearsley, J. W. Tolle: *A practical algorithm for general large-scale nonlinear optmization problems.* SIOPT 9, (1999), 755–778.

[13] M. A. Branch, Th. F. Coleman, Y. Li: *A subspace, interior and conjugate gradient method for large-scale bound-constrained minimization problems.* SIAM J. Sci Comp. 21, (1999), 1–23.

[14] I. Bongartz, A. R. Conn, N. I. M. Gould, Ph. L. Toint: *CUTE: Constrained and Unconstrained Testing Environment.* Research Report RC 18860, IBM T. J. Watson Research Center, Yorktown Heights 1993.

[15] M. G. Breitfeld, D. Shanno: *Computational experience with modified log-barrier methods for nonlinear programming.* Research report RRR 17-93, Rutgers Center for Operations Research, New Brunswick 1993.

[16] J. V. Burke: *A robust trust-region method for constrained nonlinear programming problems.* SIOPT 2, (1992), 325–347.

[17] J. V. Burke: An exact penalization viewpoint of constrained optimization. SIAM J. Control and Optimization 29(1991), 968–998.

[18] A. Buckley, L. Lenir: *QN-like variable storage conjugate gradients.* Math. Prog. 27, (1983), 155–175.

[19] R.H. Byrd, J.Ch. Gilbert, J. Nocedal: *A trust-region method based on interior-point techniques for nonlinear programming* Report 96/05. Optimization Technology Center, Northwestern University, Evanston.

[20] R.H. Byrd, M.E.Hribar, J. Nocedal: *An interior-point algorithm for large-scale nonlinear programming.* SIOPT 9, (1999), 877–900.

[21] R.H.Byrd, G. Liu, J.Nocedal: *On the local behaviour of an interior-point method for nonlinear programming.* Report 98/02. Optimization Technology Center, Northwestern University, Evanston.

[22] R. H. Byrd, P. Lu, J. Nocedal, C. Zhu: *A limited memory algorithm for bound constrained optimization.* SIAM J. Sci. Comp. 16, (1995), 1190–1208 .

[23] R. H. Byrd, J. Nocedal: *A tool for the analysis of quasi-Newton methods with applications to unconstrained minimization.* J. SIAM Numer. Anal. 26, (1989), 727–739.

[24] R. H. Byrd, J. Nocedal, R. B. Schnabel: *Representations of quasi-Newton matrices and their use in limited memory methods.* Math. Prog. 63, (1994), 129–156.

[25] A. R. Conn, N. I. M. Gould, Ph. L. Toint: *Large-scale nonlinear constrained optimization: a current survey.* CERFACS report TR/PA/94/03.

[26] A. R. Conn, N. I. M. Gould, Ph. L. Toint: *Global convergence of a class of trust-region algorithms for optimization with simple bounds.* SINUM 25, (1988), 433–460, SINUM 26, (1989), 764–767.

[27] A. R. Conn, N. I. M. Gould, Ph. L. Toint: *Testing a class of methods for solving minimization problems with simple bounds on the variables.* Math. Comp. 50, (1988), 399–430.

[28] A. R. Conn, N. I. M. Gould, Ph. L. Toint: *A globally convergent Lagrangian algorithm for optimization with general constraints and simple bounds.* SINUM 28, (1991), 545–572.

[29] A. R. Conn, N. Gould, Ph. L. Toint: *A primal-dual algorithm for minimizing a nonconvex function subject to bound and linear equality constraints.* RAL Report 96/9, (1996) .

[30] A. R. Conn, N.I.M. Gould, Ph. L. Toint: *A globally convergent Lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds.* Math. Comp. 66, (1997), 261–288.

[31] E.D. Dolan, J.J. Moré : *Benchmarking Optimization Software with COPS.* Argonne National Laboratory Mathematics and Computer Science Division Technical Report ANL/MCS-246, Nov. 2000.

[32] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, Y. Zhang: *On the formulation and theory of the Newton interior-point method for nonlinear programming.* J.O.T.A. 89, (1996), 507–541 .

[33] G. Di Pillo, S. Lucidi, l. Palagi: *A superlinearly convergent primal-dual algorithm model for constrained optimization problems with bounded variables*, OMS 14, (2000) , 49–73.

[34] I. S. Duff, J. K. Reid: *The multifrontal solution of indefinite sparse symmetric linear systems.* TOMS 9, (1983) , 302–323.

[35] I. S. Duff , J. K. Reid: *MA47: A set of FORTRAN subroutines for solving sparse symmetric sets of linear equations.* Research Report. Rutherford Appleton Laboratory, Chilton, 1993.

[36] C. Durazzi: *On the Newton interior-point method for nonlinear programming problems.* J.O.T.A. 104, (2000), 73–90.

[37] O. Epelly, J. Gondzio and J.-P. Vial: *An Interior-Point Solver for Smooth Convex Optimization with an Application to Environmental-Energy-Economic Models.* Logilab Technical Report 2000.8, Department of Management Studies, University of Geneva, Switzerland, July 2000. TR 2000.8.

[38] R. Felkel: *An interior-point method for large-scale QP problems.* THD FB Math. Report 1850(1996). Available through
http://www.mathematik.th-darmstadt.de/ags/ag8/felkel/felkel-d.html

[39] R. Felkel: *On solving large-scale nonlinear programming problems using iterative methods.* Aachen: Shaker ISBN 3-8265-6797-8 , (1999).

[40] R. Felkel, P. Spellucci: *Numerical experiments with an exact penalty function for convex inequality constrained QP-problems.* THD Math. Dept. Rep. 1815, (1996) .

[41] A. V. Fiacco, G. P. McCormick: *Nonlinear Programming. Sequential unconstrained minimization techniques.* New York: Wiley 1968. Reprinted by SIAM publications 1990.

[42] R. Fletcher: *A general quadratic programming algorithm.* J.I.M.A. 7, (1971), 76–91.

[43] R. Fletcher, S. Leyffer: *Nonlinear programming without a penalty function.* Num. Anal. Rep. NA/171, Dept. of Math., University of Dundee, 1997.

[44] R. Fletcher, S. Leyffer: *User manual for filterSQP.* Num. Anal. Rep. NA/181, Dept. of Math., University of Dundee, 1998.

[45] : *On the global convergence of an SLP-filter algorithm.* Tech. Rep. 98/13, Dept. of Math., FUNDP, Namur, Belgium, 1998.

[46] R. Fletcher, N.I.M. Gould, S. Leyffer, Ph. L. Toint: *Trust-region SQP-filter algorithms for general nonlinear programming.* Tech. Rep. 99/03. Dept. of Math., FUNDP, Namur, Belgium, 1999.

[47] R. Fletcher, E. Sainz de la Maza: *Nonlinear programming and nonsmooth optimization by successive linear programming.* Math. Prog. 43, (1989) , 235–256.

[48] R. Fontecilla: *Inexact secant methods for nonlinear constrained optimization.* SINUM 27, (1990), 154–165.

[49] A. Forsgren, W. Murray: *Newton Methods for large-scale linear inequality constrained minimization.* SIOPT7, (1997), 162–176.

[50] A. Forsgren, Ph. E. Gill: *Primal-dual interior methods for nonconvex nonlinear programming.* SIAM J. Optim. 8,(1998), 1132–1152 (1998).

[51] R. Fourer: *linear/nonlinear programming FAQ's.* World Wide Web http://www.mcs.anl.gov/otc/Guide/faq .

[52] R. Franke: *HQP - a solver for sparse nonlinear optimization.* Internal report, TU Ilmenau dept. of autom. and systems theory. (1995).

[53] R. Freund, F. Jarre: *Iterative solution of indefinite systems arising in large-scale optimization by the symmetric QMR algorithm* Presented at XVISMP Ann Arbor 1994.

[54] A. Friedlander, J. M. Martínez, A. S. Santos: *On the resolution of linearly constrained convex minimization problems.* SIOPT 4, (1994), 331–339.

[55] J. Gauvin: *A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming.* Math. Prog. 12, (1977), 136–138.

[56] D. Gay, M. Overton, M.H. Wright: *A primal dual interior-point method for nonconvex nonlinear programming.* Techn. Rep. 97-4-08 Computing Sciences Research Center, Bell Laboratories, Murray Hill, New Jersey

[57] J. Ch. Gilbert, C. Lemarechal: *Some numerical experiments with variable-storage quasi-Newton methods.* Math. Prog. 45, (1989), 407–435.

[58] Ph. E. Gill, W. Murray, M. Saunders: *User's guide for SNOPT (Version 5.3): A FORTRAN package for large-scale nonlinear programming.* Technical report NA 97-4, Department of mathematics, university of California at San Diego, 1997.

[59] Ph. E. Gill, M. A. Saunders, J. R. Shinnerl: *On the stability of Cholesky factorization for symmetric quasidefinite systems.* SIAM J. Matrix Anal. Appl. 17, (1996), 35–46.

[60] D. Goldfarb: *Extensions of Newton's method and simplex methods for solving quadratic programs.* In Lootsma: *Numerical methods for nonlinear optimization.* New York: Acad. Press 1972.

[61] D. Goldfard, A. Idnani: *A numerically stable method for solving strictly convex quadratic programs.* Math. Prog. 27, (1983), 1–33.

[62] N. I. M. Gould: *An algorithm for large-scale quadratic programming.* J.I.M.A. Num. Anal. 11, (1991), 299–324.

[63] N.I.M. Gould, M.E.Hribar, J. Nocedal: *On the solution of equality constrained quadratic programming problems arising in optimization.* Report 09/24/98. Optimization Technology Center, Northwestern University, Evanston.

[64] N. I. M. Gould, Ph. L. Toint: *SQP Methods for Large-Scale Nonlinear Programming*, Report 99/05 FUNDP , available trough anonymous ftp at "pub/reports" on thales.math.fundp.ac.be .

[65] F. J. Gould, J. W. Tolle: *A necessary and sufficient qualification for constrained optimization.* SIAM J. Appl. Math. 20, (1971), 164–172.

[66] J. Guddat, H.Th. Jongen, J. Rueckmann: *On stability and stationary points in nonlinear optimization.* J. Austr. Math. Soc. Ser. B 28, (1986), 36–56.

[67] A. Griewank, Ph. L. Toint: *Partitioned variable metric updates for large structured optimization problems.* Num. Math. 39, (1982), 429–448.

[68] J. Guddat, F. Guerra, D. Nowack: *On the role of the Mangasarian-Fromowitz constraint qualification for penalty-, exact penalty-, and Lagrange multiplier methods.* In: Fiacco, Anthony (ed.) *Mathematical programming with data perturbations.* 17th symposium, George Washington Univ. New York: Marcel Dekker. Lect. Notes Pure Appl. Math. 195, 159–183 (1997).

[69] Guignard, M.: *Generalized Kuhn-Tucker Conditions for Mathematical Programming in a Banach Space.* SIAM J. Control 7, (1969), 232–241.

[70] S. P. Han, O. L. Mangasarian: *A dual differentiable exact penalty function.* Math. Prog. 25, (1983), 293–306.

[71] J. Heinz, P. Spellucci: *A successful implementation of the Pantoja-Mayne sqp method.* Optimization Methods and Software 4, (1994), 1–28.

[72] W. Hock, K. Schittkowski: *Test examples for nonlinear programming codes.* Lecture Notes in Econ. and. Math. Syst. 187, Springer 1981.

[73] M. E. Hribar: *Large-scale constrained optimization.* Northwestern University PhD thesis 1996. Evanston, Illinois.

[74] Interior Points Online: WWW http://www.mcs.anl.gov/otc/InteriorPoint/

[75] K. Jonasson, K. Madsen: *Corrected sequential linear programming for sparse minimax optimization.* BIT 34, (1994) , 372–387.

[76] Ch. Kanzow: *An unconstrained optimization technique for large-scale linearly constrained convex minimization problems.* Computing 53, (1994) , 101–117.

[77] N. Karmarkar: *A new polynomial time algorithm for linear programming.* Combinatorica 4, (1984), 373–395.

[78] M. Kojima, S. Mizuno, A. Yoshise: *A primal-dual interior-point algorithm for linear programming.* In: N. Megiddo, ed.: *Progress in mathematical programming: interior-point and related methods.* New York: Springer 1989, 29–47.

[79] H. W. Kuhn, A. W. Tucker : *Nonlinear Programming.* Proc. 2. Berkeley Symp. Math. Stat. Prob. 1951, Univ. Cal. Press, 481–492.

[80] M. Lalee, J. Nocedal, T. Plantenga: *On the implementation of an algorithm for large-scale equality constrained optimization.* SIOPT8 , (1998), 682–706.

[81] Th. Laux: *Numerische Untersuchungen zur primal-dualen Methode von A. Forsgren und P. E. Gil.* Diploma thesis, TU Darmstadt, Dept. of mathematics, 1999.

[82] Chih-Jen Lin, Chih-Jen, J.J. Moré: *Newton's method for large bound-constrained optimization problems.* SIAM J. Optim. 9, No.4, (1999), 1100–1127 .

[83] D. C. Liu, J. Nocedal: *On the limited memory BFGS method for large-scale optimization.* Math. Prog. B, 45, (1989), 503–528.

[84] St. Lucidi, M. Roma: *Numerical experiences with new truncated Newton methods in large-scale unconstrained optimization.* Comp. Opt. Appl. 7, (1997), 71–87.

[85] Mangasarina, O.L., Fromowitz, S.: *The Fritz John necessary optimality conditions in the presence of equality and inequality constraints.* J. Math. Anal. Appl. 17, (1967), 37–47.

[86] H. D. Mittelmann: *benchmarks for optimization software*, world wibe web at http://plato.la.asu.edu/bench.html (2000)

[87] H. D. Mittelmann, P. Spellucci: *Decision Tree for Optimization Software*, world wide web at http://plato.la.asu.edu/guide.html (2000)

[88] J. J. Moré, G. Toraldo: *Algorithms for bound constrained quadratic programming problems.* Num. Math. 55, (1989), 377–400.

[89] J. J. Moré, G. Toraldo: *On the solution of large quadratic programming problems with bound constraints.* SIOPT 1, (1991), 93–113.

[90] J. J. Moré, St. J. Wright: *Optimization software guide.* Philadelphia: SIAM 1993.

[91] W. Murray: *Sequential quadratic programming methods for large-scale problems.* COAP7, (1997), 127–142.

[92] St. G. Nash: *Newton type minimization via the Lanczos method.* SINUM 21, (1984), 770–788.

[93] St. G. Nash, A. Sofer: *A barrier method for large-scale constrained optimization.* ORSA J. on Computing 5, (1993), 40–53

[94] Q. Ni: *General large-scale nonlinear programming using sequential quadratic programming methods.* Bayreuther Mathematische Schriften 45, (1993), 133–236.

[95] R. H. Nickel, J. W. Tolle: *A sparse sequential quadratic programming algorithm.* J.O.T.A. 60, (1989), 453–473.

[96] J. Nocedal, St. J. Wright: *Numerical Optimization.* New York: Springer 1999.

[97] C. C. Paige, M. A. Saunders: *Solution of sparse indefinite systems of linear equations.* SIAM J. Numer. Anal. 12, (1975), 617–629.

[98] J. F. A. de O. Pantoja, D. Q. Mayne: *Exact penalty function algorithm with simple updating of the penalty parameter.* J.O.T.A. 69, (1991), 441–467.

[99] T. D. Plantenga: *Large-scale constrained optimization using trust-regions.* PhD dissertation. Northwestern University, Evanston, Illinois, 1994.

[100] W. Pinz: *Ein neues Verfahren zum Lösen quadratischer Optimierungsprobleme.* Diplomarbeit THD FB Mathematik, 1996.

[101] R. Polyak: *Modified barrier functions (theory and methods).* Math. Prog. 54, (1992), 177–222 .

[102] F. Potra, R. Tapia and Y.Zhang: *On the superlinear convergence of interior-point algorithms for a general class of problems.* SIAM J. Optim. 3, (1993), 413–422.

[103] St. Robinson: *A quadratically convergent algorithm for general nonlinear programming problems.* Math. Prog. 1, (1972), 145–156.

[104] St. Robinson: *Stability theory for systems of inequalities. Part II. Differentiable nonlinear systems.* SIAM J. Num. Anal. 13, (1976), 497–512.

[105] R. T. Rockafellar: *Lagrange multipliers and optimality.* SIAM Rev. 35, (1993), 183–238.

[106] M.A. Saunders: *Recent developments to MINOS (MINOS5.4).* Presented at the XVISMP at Ann Arbor, 1994

[107] S. Sahni: *Computationally related problems.* SIAM J. Comp. 3, (1974), 262–279.

[108] K. Schittkowski: *More testexamples for nonlinear programming codes.* Lect. Notes in Econ. and Math. Syst. 282, Springer 1987

[109] K. Schittkowski: *On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function.* Math. Oper. and Stat. Series Opt. 14, (1983), 197–216.

[110] P. Spellucci: *Solving general convex QP problems via an exact quadratic Lagrangian with bound constraints.* THD FB4 preprint 1563 (1993). Revised version 7/94.

[111] P. Spellucci: *An SQP method for general linear programs using only equality constrained subproblems.* Math. Prog. 82, (1998), 413–448.

[112] P. Spellucci: *A new technique for inconsistent QP problems in the SQP method.* Math. Meth. OR 47, (1998), 355–400.

[113] P. Spellucci: *Numerische Verfahren der nichtlinearen Optimierung.* Birkhäuser 1993.

[114] P. Spellucci: *Numerical experiments with modern methods for large-scale QP problems.* In "Recent advances in optimization" (Gritzmann, Horst, Sachs, Tichatschke eds.) Lect. Notes in Econ. and Math. Syst. 452. Springer (1997) . 313–335.

[115] St. J. Wright: *Primal-dual interior-point methods.* Philadelphia: SIAM publications 1997.

[116] R. J. Vanderbei: *Symmetric quasidefinite matrices.* SIAM J. Optim. 5, (1995) 100–113 .

[117] R.J. Vanderbei, D.F.Shanno: *An interior-point algorithm for nonconvex nonlinear programming.* SOR-97-21. Statistics and Operations Research, Princeton University.

[118] S. Vavasis: *Quadratic programming is in NP.* Info. Proc. Lett. 36 (1990), 73–77.

[119] X. Zou, I.M. Navon, M. Berger, K.H. Phua, T. Schlick, F.X. Le Dimet: *Numerical experience with limited-memory quasi-Newton and truncated Newton methods.* SIOPT 3, (1993), 582–608 .