Global Convergence of
Trust-Region SQP-Filter Algorithms
for General Nonlinear Programming

by R. Fletcher[1], N.I.M. Gould[2],
S. Leyffer[1] and Ph. L. Toint[3]

Report 99/03 November 12, 1999

[1] Department of Mathematics, University of Dundee,
Dundee, Scotland
Email: fletcher@mcs.dundee.ac.uk, sleyffer@mcs.dundee.ac.uk

[2] Rutherford Appleton Laboratory,
Computational Science and Engineering Departement
Chilton, Oxfordshire, England
Email: gould@rl.ac.uk
Current reports available from
http://www.numerical.rl.ac.uk/reports/reports.html

[3] Department of Mathematics, University of Namur,
61, rue de Bruxelles, B-5000 Namur, Belgium,
Email: pht@math.fundp.ac.be
Current reports available by anonymous ftp from the directory
"pub/reports" on thales.math.fundp.ac.be

Global Convergence of Trust-Region SQP-Filter Algorithms for General Nonlinear Programming

Roger Fletcher, Nick Gould , Sven Leyffer and Philippe L. Toint

4 November 1999

Abstract

Global convergence to first-order critical points is proved for two trust-region SQP-filter algorithms of the type introduced by Fletcher and Leyffer (1997). The algorithms allow for an approximate solution of the quadratic subproblem and incorporate the safeguarding tests described in Fletcher, Leyffer and Toint (1998). The first algorithm decomposes the step into its normal and tangential components, while the second replaces this decomposition by a stronger condition on the associated model decrease.

1 Introduction

We analyze an algorithm for solving optimization problems where a smooth objective function is to be minimized subject to smooth nonlinear constraints. No convexity assumption is made. More formally, we consider the problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) \geq 0, \end{array} \tag{1.1}$$

where f is a twice continuously differentiable real valued function of the variables $x \in \mathbb{R}^n$ and $c_{\mathcal{E}}(x)$ and $c_{\mathcal{I}}(x)$ are a twice continuously differentiable functions from $\mathbb{R}^n$ into $\mathbb{R}^m$ and from $\mathbb{R}^n$ into $\mathbb{R}^p$, respectively. Let $c(x)^T = (c_{\mathcal{E}}(x)^T \;\; c_{\mathcal{I}}(x)^T)$.

The class of algorithms that we discuss belongs to the class of trust-region methods and, more specifically, to that of *filter methods* introduced by Fletcher and Leyffer (1997), in which the use of a penalty function, a common feature of the large majority of the algorithms for constrained optimization, is replaced by the introduction of a so-called "filter".

A global convergence theory for an algorithm of this class is proposed in Fletcher et al. (1998), in which the objective function is locally approximated by a linear function, leading, at each iteration, to the (exact) solution of a linear program. This algorithm therefore mixes the use of the filter with sequential linear programming (SLP). In this paper, we consider approximating the objective function by a quadratic model, which results in a sequential quadratic programming (SQP) technique. This in turn leads us to modifications of the algorithm discussed in Fletcher et al. (1998). Besides the very important fact that the objective function's model is quadratic instead of linear, the method discussed here has a different mechanism for deciding on the compatibility of the associated (QP) subproblem. It also allows for an approximate solution of this subproblem at each iteration.

2 A Class of Trust-Region SQP-Filter Algorithms

2.1 An approximate SQP framework

Sequential quadratic programming methods are iterative. At a given iterate x_k, they implicitly apply Newton's method to solve (a local version of) the first-order necessary optimality conditions by solving the quadratic programming subproblem $\mathrm{QP}(x_k)$ given by

$$\begin{array}{ll} \text{minimize} & f_k + \langle g_k, s\rangle + \frac{1}{2}\langle s, H_k s\rangle \\ \text{subject to} & c_{\mathcal{E}}(x_k) + A_{\mathcal{E}}(x_k)s = 0 \\ & c_{\mathcal{I}}(x_k) + A_{\mathcal{I}}(x_k)s \geq 0, \end{array} \tag{2.1}$$

where $f_k = f(x_k)$, $g_k = g(x_k) = \nabla_x f(x_k)$, where $A_{\mathcal{E}}(x_k)$ and $A_{\mathcal{I}}(x_k)$ are the Jacobians of the constraint functions $c_{\mathcal{E}}$ and $c_{\mathcal{I}}$ at x_k and where H_k is a symmetric matrix. We will not immediately be concerned about how H_k is obtained, but we will return to this point in Section 3. Assuming a suitable value of H_k can be found, the solution of $\mathrm{QP}(x_k)$ then yields a step s_k.

Unfortunately, due to the locally convergent nature of Newton's iteration, the step s_k may not always be very good. One possible way to cope with this difficulty is to perform a linesearch with respect to an appropriate merit function along the step s_k, which is where penalty functions typically play a role. However, as one of our objectives is to avoid penalty functions (and the need to update the associated penalty parameter), we instead consider a trust-region approach that will not use any penalty function. In such an approach, the objective function of $\mathrm{QP}(x_k)$ is only intended to be of local interest, that is we restrict the step s_k in norm to ensure that $x_k + s_k$ remains in a *trust-region* centred at x_k. In other words, we replace $\mathrm{QP}(x_k)$ by the subproblem $\mathrm{TRQP}(x_k, \Delta_k)$ given by

$$\begin{array}{ll} \text{minimize} & m_k(s) \\ \text{subject to} & c_{\mathcal{E}}(x_k) + A_{\mathcal{E}}(x_k)s = 0, \\ & c_{\mathcal{I}}(x_k) + A_{\mathcal{I}}(x_k)s \geq 0, \\ \text{and} & \|s\| \leq \Delta_k, \end{array} \tag{2.2}$$

for some (positive) value of the *trust-region radius* Δ_k, where we have defined

$$m_k(s) = f_k + \langle g_k, s\rangle + \tfrac{1}{2}\langle s, H_k s\rangle, \tag{2.3}$$

and where $\|\cdot\|$ denotes the Euclidean norm.

Remarkably, most of the existing SQP algorithms assume that an exact local solution of $\mathrm{QP}(x_k)$ or $\mathrm{TRQP}(x_k, \Delta_k)$ is found, although attempts have been made by Dembo and Tulowitzki (1983) and Murray and Prieto (1995) to design conditions under which an approximate solution of the subproblem is acceptable. We revisit this issue in what follows, and start by noting that the step s_k may be viewed as the sum of two distinct components, a *normal step* n_k, such that $x_k + n_k$ satisfies the constraints of $\mathrm{TRQP}(x_k, \Delta_k)$, and a *tangential step* t_k, whose purpose is to obtain reduction of the objective function's model while continuing to satisfy those constraints. This framework is therefore similar in spirit to the Byrd-Omojokun technique proposed by Omojokun (1989) and later developed by several authors, including Biegler, Nocedal and Schmid (1995), El-Alem (1995, 1996), Byrd, Gilbert and Nocedal (1996), Byrd, Hribar and Nocedal (1997), Bielschowsky and Gomes (1998), Liu and Yuan (1998) and Lalee, Nocedal and Plantenga (1998). More formally, we write

$$s_k = n_k + t_k \tag{2.4}$$

and assume that

$$c_{\mathcal{E}}(x_k) + A_{\mathcal{E}}(x_k) n_k = 0, \quad c_{\mathcal{I}}(x_k) + A_{\mathcal{I}}(x_k) n_k \geq 0, \quad \|n_k\| \leq \Delta_k, \tag{2.5}$$

$$\|s_k\| \leq \Delta_k, \tag{2.6}$$

and

$$c_{\mathcal{E}}(x_k) + A_{\mathcal{E}}(x_k) s_k = 0, \quad c_{\mathcal{I}}(x_k) + A_{\mathcal{I}}(x_k) s_k \geq 0. \tag{2.7}$$

Of course, this is a strong assumption, since in particular (2.5) may not have a feasible point. We shall return to this possibility shortly. Given our assumption, there are many ways to compute n_k and t_k. For instance, we could compute n_k from

$$n_k = P_k[x_k] - x_k, \tag{2.8}$$

where P_k is the orthogonal projector onto the feasible set of $\mathrm{QP}(x_k)$. In what follows, we do not make any specific choice for n_k, but we shall make the assumptions that n_k exists when the maximum violation of the nonlinear constraints at the k-th iterate $\theta_k \stackrel{\text{def}}{=} \theta(x_k)$, where

$$\theta(x) = \max\left[0, \max_{i \in \mathcal{E}} |c_i(x)|, \max_{i \in \mathcal{I}} -c_i(x)\right], \tag{2.9}$$

is sufficiently small, and that n_k is reasonably scaled with respect to the values of the constraints in that

$$\|n_k\| \leq \kappa_{\text{usc}} \theta_k \tag{2.10}$$

for some constant $\kappa_{\text{usc}} > 0$, whenever θ_k is sufficiently small. We can interpret this assumption in terms of the constraint functions themselves and the geometry of the boundary of the feasible set. For instance, if we define

$$\mathcal{F}(x) \stackrel{\text{def}}{=} \{v \in \mathbb{R}^n \mid c_{\mathcal{E}}(x) + A_{\mathcal{E}}(x)(v - x) = 0, \;\; c_{\mathcal{I}}(x) + A_{\mathcal{I}}(x)(v - x) \geq 0\}$$

and assume that, at every limit point x_* of the sequence of iterates, the relative interior of the linearized constraints $\mathrm{ri}\{\mathcal{F}(x_*)\}$ is non-empty, we know, by applying a continuity argument, that the same must be true for any iterate x_k that is close enough to x_*. Hence the feasible set of $\mathrm{QP}(x_k)$ is non-empty for such an x_k, which implies that P_k is well defined and that a normal step n_k of the form (2.8) exists. Furthermore, if the singular values of the Jacobian of constraints active at x_*, $A_{\mathcal{A}(x_*)}(x_*)$, are bounded away from zero, the same must be true by continuity for $A_{\mathcal{A}(x_*)}(x_k)$, and the projection operator

$$P_k = A^T_{\mathcal{A}(x_*)}(x_k)\left[A_{\mathcal{A}(x_*)}(x_k)A^T_{\mathcal{A}(x_*)}(x_k)\right]^{-1}A_{\mathcal{A}(x_*)}(x_k)$$

must be bounded in norm for all x_k sufficiently close to x_*. Since only the constraints active at x_* can be active in a sufficiently small neighbourhood of this limit point, the boundedness of the projection operator in turn guarantees that (2.10) holds for the normal step

$$-A^T_{\mathcal{A}(x_*)}(x_k)\left[A_{\mathcal{A}(x_*)}(x_k)A^T_{\mathcal{A}(x_*)}(x_k)\right]^{-1}c_{\mathcal{A}(x_*)}(x_k),$$

for all k sufficiently large, provided the sequence of iterates remains bounded, because this latter assumption ensures that x_k must be arbitrarily close to a least one limit point of the sequence $\{x_k\}$ for such k. Thus we see that (2.10) does not impose conditions on the constraints or the normal step itself that are unduly restrictive.

Having defined the normal step, we write

$$x_k^{\mathrm{N}} = x_k + n_k, \tag{2.11}$$

and observe that n_k satisfies the constraints of $\mathrm{TRQP}(x_k,\Delta_k)$ and thus also of $\mathrm{QP}(x_k)$. It is crucial to note, at this stage, that n_k may fail to exist because the constraints of $\mathrm{QP}(x_k)$ may be incompatible, in which case P_k is undefined.

Let us continue to consider the case where this problem does not arise, and a normal step n_k has been found. We then have to find a tangential step t_k, starting from x_k^{N} and satisfying (2.6) and (2.7), with the aim of decreasing the value of the objective function. As always in trust-region methods, this is achieved by computing a step that produces a sufficient decrease in m_k, which is to say that we wish $m_k(x_k^{\mathrm{N}}) - m_k(x_k+s_k)$ to be "sufficiently large". Of course, this is only possible if the maximum size of t_k is not too small, which is to say that x_k^{N} is not too close to the trust-region boundary. We formalize this condition by requiring that

$$\|n_k\| \le \kappa_\Delta \Delta_k \min[1, \kappa_\mu \Delta_k^\mu], \tag{2.12}$$

for some $\kappa_\Delta \in (0,1)$, some $\kappa_\mu > 0$ and some $\mu \in (0,1)$. If condition (2.12) does not hold, we assume that the computation of t_k is unlikely to produce a satisfactory decrease in m_k, and proceed just as if the feasible set of $\mathrm{TRQP}(x_k,\Delta_k)$ were empty. If n_k can be computed and (2.12) holds, we shall say that $\mathrm{TRQP}(x_k,\Delta_k)$ is *compatible*. In this case at least a sufficient decrease seems possible. In order to formalize what we mean, we recall that the feasible set of $\mathrm{TRQP}(x_k,\Delta_k)$ is convex, and we can therefore introduce the first-order criticality measure

$$\chi_k = \chi(x_k) = |\min_{\substack{c_{\mathcal{E}}(x_k)+A_{\mathcal{E}}(x_k)d=0\\ c_{\mathcal{I}}(x_k)+A_{\mathcal{I}}(x_k)d\geq 0\\ \|d\|\leq 1}} \langle g(x_k^N), d\rangle| \tag{2.13}$$

(see Conn, Gould, Sartenaer and Toint, 1993). Note that this function is continuous in its argument because both the gradient of the objective function and the Jacobian of the constraints are continuous. We also observe that $\chi_k = 0$ when $x_k^{\rm N}$ is a first-order critical point of TRQP(x_k, Δ_k), and if $x_k = x_k^{\rm N}$ and $\chi_k = 0$, x_k is itself a first-order critical point for the original problem.

Having already considered the conditions we require for the normal step, we now formulate our requirement that the tangential step yields a sufficient model decrease in the form of a familiar Cauchy-point condition and assume that there exists a constant $\kappa_{\rm tmd} > 0$ such that

$$m_k(x_k^{\rm N}) - m_k(x_k^{\rm N} + t_k) \geq \kappa_{\rm tmd}\chi_k \min\left[\frac{\chi_k}{\beta_k}, \Delta_k\right], \tag{2.14}$$

where $\beta_k = 1 + \|H_k\|$. We know from Toint (1988) and Conn et al. (1993) that this condition holds if the model reduction exceeds that which would be obtained at the generalized Cauchy point, that is the point resulting from a backtracking curvilinear search along the projected gradient path from $x_k^{\rm N}$, that is

$$x_k(\alpha) = P_k[x_k^{\rm N} - \alpha\nabla_x m_k(x_k^{\rm N})].$$

Let us now return to the case where TRQP(x_k, Δ_k) is not compatible, that is when the feasible set determined by the constraints of QP(x_k) is empty, or the freedom left to reduce m_k within the trust region is too small in the sense that (2.12) fails. In this situation, solving TRQP(x_k, Δ_k) is most likely pointless, and we must consider an alternative. We base this on the intuitive observation that, if $c(x_k)$ is sufficiently small and the true nonlinear constraints are locally compatible, the linearized constraints should also be compatible, since they approximate the nonlinear constraints (locally) correctly. Furthermore, the feasible region for the linearized constraints should then be close enough to x_k for there to be some room to reduce m_k, at least if Δ_k is large enough. If the nonlinear constraints are locally incompatible, we have to find a neighbourhood where this is not the case, since the problem (1.1) does not make sense in the current one. We thus rely on a *restoration procedure*, whose aim is to produce a new point $x_k + r_k$ for which TRQP$(x_k + r_k, \Delta_{k+1})$ is compatible for some $\Delta_{k+1} > 0$—we will actually need another condition which we will discuss shortly.

The idea of the restoration procedure is to (approximately) solve

$$\min_{x\in\mathbb{R}^n} \theta(x) \tag{2.15}$$

starting from x_k, the current iterate. This is a non-smooth problem, but we know that there exist methods, possibly of trust-region type (such as that suggested by Yuan, 1994), which can be successfully applied to solve it. Thus we will not describe the restoration procedure in detail. Note that we have chosen here to reduce the infinity norm of the constraint violation, but we could equally well consider other norms, such as ℓ_1 or ℓ_2, in which case the methods of Fletcher and Leyffer (1998) or of El-Hallabi and Tapia (1995) and Dennis, El-Alem and Williamson (1999) can respectively be considered). Of course, this technique only guarantees convergence to a first-order critical point of the chosen measure of constraint violation, which means that, in fact, the restoration procedure may fail as this critical point may not be feasible for the constraints of (1.1). However, even in this case, the result of the procedure is of interest because it typically produces a local minimizer of $\theta(x)$, or of whatever other measure of constraint violation we choose for the restoration, yielding a point of locally-least infeasibility.

There is no easy way to circumvent this drawback, as it is known that finding a feasible point or proving that no such point exists is a global optimization problem and can be just as difficult as the optimization problem (1.1) itself[(1)]. We therefore accept two possible outcomes of the restoration procedure: either the procedure fails in that it does not produce a sequence of iterates converging to feasibility, or a point $x_k + r_k$ is produced such that $\theta(x_k + r_k)$ is as small as we wish. We will shortly see that this is all we need.

2.2 The Notion of a Filter

Having computed a step $s_k = n_k + t_k$ (or r_k), we still need to decide whether the trial point $x_k + s_k$ (or $x_k + r_k$) is any better than x_k as an approximate solution to our original problem (1.1). We shall use a concept borrowed from multi-criteria optimization. We say that a point x_1 *dominates* a point x_2 whenever

$$\theta(x_1) \leq \theta(x_2) \text{ and } f(x_1) \leq f(x_2).$$

Thus, if iterates x_k dominates iterate x_j, the latter is of no real interest to us since x_k is at least as good as x_j on account of both feasibility and optimality. All we need to do now is to remember iterates that are not dominated by any other iterates using a structure called a filter. A *filter* is a list $\mathcal{F}$ of pairs of the form (θ_i, f_i) such that

$$\theta_i < \theta_j \text{ or } f_i < f_j$$

for $i \neq j$. We thus aim to accept a new iterate x_i only if it is not dominated by any other iterate in the filter. In the vocabulary of multi-criteria optimization, this amounts to building elements of the efficient frontier associated with the bi-criteria problem of reducing infeasibility and the objective function value. Figure 2.1 illustrates the concept of a filter by showing the pairs (θ_k, f_k) as black dots in the (θ, f) space. Each such pair is called the (θ, f)-pair associated with x_k. The lines radiating from each

[(1)] In practice, this is rarely the case since the solution set for the former is almost always far bigger than that for the latter.

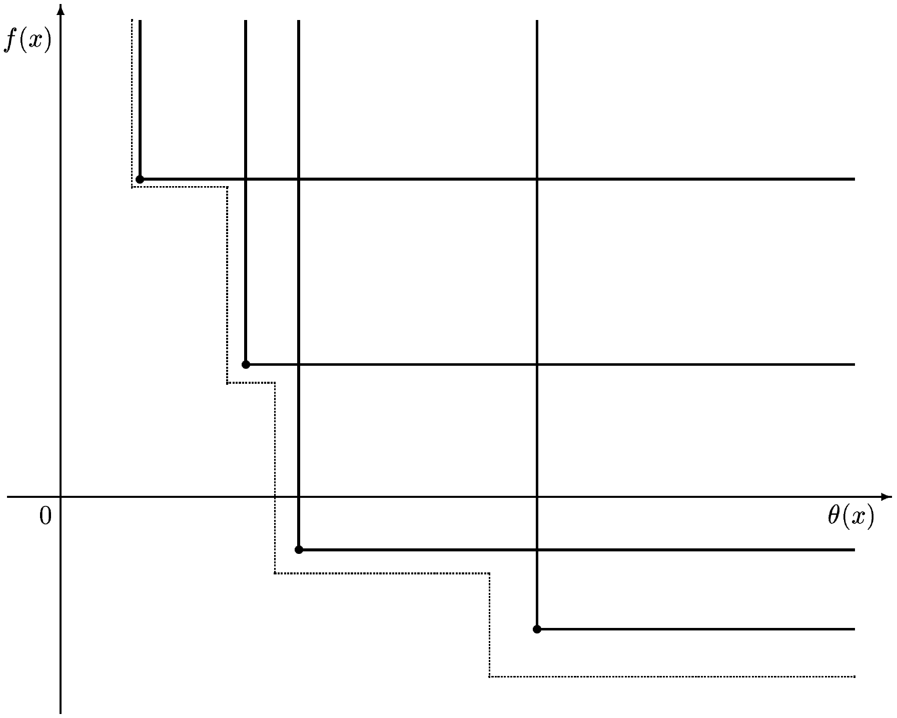

Figure 2.1: A filter with four pairs.

(θ, f)-pair indicate that any iterate whose associated (θ, f)-pair occurs above and to the right of that of a given filter point is dominated by this (θ, f)-pair.

While the idea of not accepting dominated trial points is simple and elegant, it needs to be refined a little in order to provide an efficient algorithmic tool. In particular, we do not wish to accept $x_k + s_k$ if its (θ, f)-pair is arbitrarily close to that of x_k or that of a point already in the filter. Thus we set a small "margin" around the border of the dominated part of the (θ, f)-space in which we shall also reject trial points. Formally, we say that a point x is *acceptable for the filter* if and only if

$$\theta(x) < (1-\gamma_\theta)\theta_j \quad \text{or} \quad f(x) < f_j - \gamma_\theta \theta_j \quad \text{for all} \quad (\theta_j, f_j) \in \mathcal{F}, \tag{2.16}$$

for some $\gamma_\theta \in (0,1)$. In Figure 2.1, the set of acceptable points corresponds to the set of (θ, f)-pairs below the thin line. We also say that x is "acceptable for the filter and x_k" if (2.16) holds with $\mathcal{F}$ replaced by $\mathcal{F} \cup (\theta_k, f_k)$. We thus move from x_k to $x_k + s_k$ only if $x_k + s_k$ is acceptable for the filter and x_k.

As the algorithm progresses, we may want to *add a (θ, f)-pair to the filter*. If an iterate x_k is acceptable for $\mathcal{F}$, we do this by adding the pair (θ_k, f_k) to the filter and by removing from it every other pair (θ_j, f_j) such that $\theta_j \geq \gamma_\theta \theta_k$ and $f(x) \geq f_k - \gamma_\theta \theta_k$. We also refer to this operation as "adding x_k to the filter" although, strictly speaking, it is the (θ, f)-pair which is added.

2.3 An SQP-Filter Algorithm

We have now discussed the main ingredients of the class of algorithms we wish to consider, and we are now ready to define it formally.

A flow-chart of the algorithm is given in Appendix. Reasonable values for the constants might be

$$\gamma_0 = 0.1, \quad \gamma_1 = 0.5, \quad \gamma_2 = 2, \quad \eta_1 = 0.01, \quad \eta_2 = 0.9,$$
$$\gamma_\theta = 10^{-4}, \quad \kappa_\Delta = 0.7, \quad \kappa_\mu = 100, \quad \mu = 0.01, \quad \kappa_\theta = 10^{-4}, \text{ and } \kappa_{\rm tmd} = 0.01.$$

but it is too early to know if these are even close to the best possible choices.

Observe first that, by construction, every iterate x_k must be acceptable for the filter at the beginning of iteration k, irrespective of the possibility that it is added to the filter later. Also note that the restoration step r_k cannot be zero, that is restoration cannot simply entail enlarging the trust-region radius to ensure (2.12), even if n_k exists. This is because x_k is added to the filter before r_k is computed, and $x_k + r_k$ must be acceptable for the filter which now contains x_k.

For the restoration procedure in Step 1 to succeed, we have to evaluate whether $\text{TRQP}(x_k + r_k, \Delta_{k+1})$ is compatible for a suitable value of Δ_{k+1}. This requires that a suitable normal step be computed which successfully passes the test (2.12). Of course, once this is achieved, this normal step may be reused at iteration $k+1$. Thus we shall require the normal step calculated to verify compatibility of $\text{TRQP}(x_k + r_k, \Delta_{k+1})$ should actually be used as n_{k+1}.

As it stands, the algorithm is not specific about how to choose Δ_{k+1} during a restoration iteration. On one hand, there is an advantage to choosing a large Δ_{k+1}, since this allows a large step and one hopes good progress. On the other, it may be unwise to choose it to be too large, as this may possibly result in a large number of unsuccessful iterations, during which the radius is reduced, before the algorithm can make any progress. A possible choice might be to restart from the radius obtained during the restoration iteration itself, if it uses a trust-region method. Reasonable alternatives would be to use the average radius observed during past successful iterations, or to apply the internal doubling strategy of Byrd, Schnabel and Shultz (1987) to increase the new radius, or even to consider the technique described by Sartenaer (1997). However, we recognize that numerical experience with the algorithm is too limited at this stage to make definite recommendations.

If the iterate x_k is feasible, then $x_k = x_k^{\rm N}$ and we obtain that

$$0 = \kappa_\theta \theta_k^2 \le m_k(x_k^{\rm N}) - m_k(x_k + s_k) = m_k(x_k) - m_k(x_k + s_k).$$

As a consequence *no feasible iterate is ever included in the filter*, which is crucial in allowing finite termination of the restoration procedure. Indeed, if the restoration procedure is required at iteration k of the filter algorithm and produces a sequence of points $\{x_{k,j}\}$ converging to feasibility, there must be an iterate $x_{k,j}$ for which

$$\theta_{k,j} \stackrel{\rm def}{=} \theta(x_{k,j}) \le \min\left[(1-\gamma_\theta)\theta_k^{\rm min}, \frac{\kappa_{\rm usc}}{\kappa_\Delta}\Delta_{k+1}\min[1, \kappa_\mu \Delta_{k+1}^\mu]\right],$$

Algorithm 2.1: SQP-Filter Algorithm

Step 0: Initialization. Let an initial point x_0, an initial trust-region radius $\Delta_0 > 0$ and an initial symmetric matrix H_0 be given, as well as constants $\gamma_0 < \gamma_1 \leq 1 \leq \gamma_2$, $0 < \eta_1 \leq \eta_2 < 1$, $\gamma_\theta \in (0,1)$, $\kappa_\theta \in (0,1)$, $\kappa_\Delta \in (0,1]$, $\kappa_\mu > 0$, $\mu \in (0,1)$, and $\kappa_{\text{tmd}} \in (0,1]$. Compute $f(x_0)$ and $c(x_0)$ and set $\mathcal{F} = \emptyset$ and $k = 0$.

Step 1: Ensure compatibility. Attempt to compute a step n_k. If TRQP (x_k, Δ_k) is compatible, go to Step 2. Otherwise, include x_k in the filter and compute a restoration step r_k for which TRQP$(x_k + r_k, \Delta_{k+1})$ is compatible for some $\Delta_{k+1} > 0$, and $x_k + r_k$ is acceptable for the filter. If this proves impossible, stop. Otherwise, define $x_{k+1} = x_k + r_k$ and go to Step 6.

Step 2: Determine a trial step. Compute a step t_k for which (2.14) holds, and set $s_k = n_k + t_k$.

Step 3: Tests to accept the trial step.

- Evaluate $c(x_k + s_k)$ and $f(x_k + s_k)$.
- If $x_k + s_k$ is not acceptable for the filter and x_k, set $x_{k+1} = x_k$, choose $\Delta_{k+1} \in [\gamma_0 \Delta_k, \gamma_1 \Delta_k]$, increment k by one and go to Step 1.
- If

$$m_k(x_k) - m_k(x_k + s_k) \geq \kappa_\theta \theta_k^2, \tag{2.17}$$

and

$$\rho_k \overset{\text{def}}{=} \frac{f(x_k) - f(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)} < \eta_1, \tag{2.18}$$

again set $x_{k+1} = x_k$, choose $\Delta_{k+1} \in [\gamma_0 \Delta_k, \gamma_1 \Delta_k]$, increment k by one and go to Step 1.

Step 4: Test to include the current iterate in the filter. If (2.17) fails, include x_k in the filter $\mathcal{F}$.

Step 5: Move to the new iterate. Set $x_{k+1} = x_k + s_k$ and choose

$$\Delta_{k+1} \in \begin{cases} [\gamma_1 \Delta_k, \Delta_k] & \text{if } \rho_k \in [\eta_1, \eta_2), \\ [\Delta_k, \gamma_2 \Delta_k] & \text{if } \rho_k \geq \eta_2. \end{cases}$$

Step 6: Update the Hessian approximation. Determine H_{k+1}. Increment k by one and go to Step 1.

for any given $\Delta_{k+1} > 0$, where

$$\theta_k^{\text{min}} = \min_{i \in \mathcal{Z},\, i \leq k} \theta_i > 0$$

and

$$\mathcal{Z} = \{k \mid x_k \text{ is added to the filter}\}.$$

Moreover, $\theta_{k,j}$ must eventually be small enough to ensure, using our assumption on the normal step, the existence of a normal step $n_{k,j}$ from $x_{k,j}$. In other words, the restoration iteration must eventually find an iterate $x_{k,j}$ which is acceptable for the filter and for which the normal step exists and satisfies (2.12), i.e. an iterate x_j which is both acceptable and compatible. As a consequence, the restoration procedure will terminate in a finite number of steps, and the filter algorithm may then proceed. Note that the restoration step may not terminate in a finite number of iterations if we do not assume the existence of the normal step when the constraint violation is small enough, even if this violation converges to zero (see Fletcher et al., 1998, for an example).

Notice also that (2.17) ensures that the denominator of ρ_k in (2.18) will be strictly positive whenever θ_k is. If $\theta_k = 0$, then $x_k = x_k^{\rm N}$, and the denominator of (2.18) will be strictly positive unless x_k is a first-order critical point because of (2.14).

Finally, we recognise that (2.14) may be difficult to verify in practice, since it may be expensive to compute $x_k^{\rm N}$ and P_k. We shall consider a possibly cheaper alternative in Section 4.

3 Convergence to First-Order Critical Points

We now prove that our algorithm generates a globally convergent sequence of iterates, at least if the restoration iteration always succeeds. For the purpose of our analysis, we shall consider

$$\mathcal{S} = \{k \mid x_{k+1} = x_k + s_k\},$$

the set of (indices of) successful iterations, and

$$\mathcal{R} = \{k \mid n_k \text{ does not exist or } \|n_k\| > \kappa_\Delta \Delta_k \min[1, \kappa_\mu \Delta_k^\mu]\},$$

the set of *restoration* iterations; we shall refer to those iterations whose indices do not lie in $\mathcal{R}$ as *normal* iterations. In order to obtain our global convergence result, we will use the assumptions

AS1: f and the constraint functions $c_{\mathcal{E}}$ and $c_{\mathcal{I}}$ are twice continuously differentiable;

AS2: there exists $\kappa_{\rm umh} > 1$ such that

$$\|H_k\| \leq \kappa_{\rm umh} - 1 < \kappa_{\rm umh} \text{ for all } k,$$

AS3: the iterates $\{x_k\}$ remain in a bounded domain $X \subset \mathbb{R}^n$.

If, for example, H_k is chosen as an approximation of the Hessian of the Lagrangian function

$$\ell(x,y) = f(x) + \langle y_{\mathcal{E}}, c_{\mathcal{E}}(x)\rangle + \langle y_{\mathcal{I}}, c_{\mathcal{I}}(x)\rangle$$

at x_k, in that

$$H_k = \nabla_{xx} f(x_k) + \sum_{i \in \mathcal{E} \cup \mathcal{I}} [y_k]_i \nabla_{xx} c_i(x_k), \tag{3.1}$$

where $[y_k]_i$ denotes the i-th component of the vector of Lagrange multipliers $y_k^T = (y_{\mathcal{E},k}^T \;\; y_{\mathcal{I},k}^T)$, then we see from AS1 and AS3 that AS2 is satisfied when these multipliers remain bounded. The same is true if the Hessian matrices in (3.1) are replaced by bounded approximations.

A first immediate consequence of AS1 is that there exists a constant $\kappa_{\rm ubh} > 1$ such that, for all k,

$$|f(x_k + s_k) - m_k(x_k + s_k)| \le \kappa_{\rm ubh} \Delta_k^2. \tag{3.2}$$

A proof of this property, based on Taylor expansion, may be found, for instance, in Toint (1988). A second important consequence of our assumptions is that AS1 and AS3 together directly ensure that, for all k,

$$f^{\rm min} \le f(x_k) \le f^{\rm max} \;\; \text{and} \;\; 0 \le \theta_k \le \theta^{\rm max} \tag{3.3}$$

for some constants $f^{\rm min}$, $f^{\rm max}$ and $\theta^{\rm max} > 0$. Thus the part of the (θ, f)-space in which the (θ, f)-pairs associated with the filter iterates lie is restricted to the rectangle

$$\mathcal{A}_0 = [0, \theta^{\rm max}] \times [f^{\rm min}, f^{\rm max}],$$

whose area, surf$(\mathcal{A}_0)$, is clearly finite. If there are (θ, f)-pairs in the filter $\mathcal{F}$ at iteration k, we let $\mathcal{A}_k$ be the part of $\mathcal{A}_0$ in which the (θ, f)-pairs associated with a new iterate must fall for this iterate to be acceptable, that is

$$\mathcal{A}_k = \{(\theta(x), f(x)) \in \mathcal{A}_0 \mid (2.16) \;\; \text{holds} \;\; \}.$$

We also note the following simple consequence of (2.10) and AS3.

Lemma 3.1 Suppose that (2.10) and AS3 hold, and that $\{x_{k_i}\}$ is a subsequence of iterates for which

$$\lim_{i \to \infty} \theta_{k_i} = 0.$$

Then there exists a constant $\kappa_{\rm lsc} > 0$ such that

$$\kappa_{\rm lsc} \theta_{k_i} \le \|n_{k_i}\| \tag{3.4}$$

for i sufficiently large.

Proof. Consider an iterate x_{k_i} for which $\theta_{k_i} > 0$ and for which n_{k_i} exists (as a consequence of (2.10)), and define

$$\mathcal{V}_{k_i} \stackrel{\rm def}{=} \{j \in \mathcal{E} \mid \theta_{k_i} = |c_j(x_{k_i})|\} \bigcup \{j \in \mathcal{I} \mid \theta_{k_i} = -c_j(x_{k_i})\},$$

that is the subset of most-violated constraints. From the definitions of θ_{k_i} in (2.9) and of the normal step in (2.5) we obtain, using the Cauchy-Schwartz inequality, that

$$\theta_{k_i} \leq |\langle \nabla_x c_j(x_{k_i}), n_{k_i} \rangle| \leq \|\nabla_x c_j(x_{k_i})\| \, \|n_{k_i}\| \tag{3.5}$$

for all $j \in \mathcal{V}_{k_i}$. But AS3 ensures that there exists a constant $\kappa_{\rm lsc} > 0$ such that

$$\max_{j \in \mathcal{E} \cup \mathcal{I}} \max_{x \in X} \|\nabla_x c_j(x)\| \stackrel{\rm def}{=} \frac{1}{\kappa_{\rm lsc}}.$$

We then obtain the desired conclusion by substituting this bound in (3.5). □

We start our analysis by examining what happens when an infinite number of iterates (that is, their (θ, f)-pairs) are added to the filter.

Lemma 3.2 Suppose that AS1 and AS3 hold and that $\{k_j\}$, the subsequence at which the current iterate x_{k_j} is added to the filter, is infinite. Then

$$\liminf_{j \to \infty} \theta_{k_j} = 0.$$

Proof. Suppose, for the purpose of obtaining a contradiction, that

$$\theta_{k_j} \geq \epsilon \tag{3.6}$$

for all j and for some $\epsilon > 0$. At each iteration k_j, the (θ, f)-pair associated with x_{k_j}, that is (θ_{k_j}, f_{k_j}), is added to the filter. This means that no other (θ, f)-pair can be added to the filter at a later stage within the square

$$[\theta_{k_j} - \gamma_\theta \epsilon, \theta_{k_j}] \times [f_{k_j} - \gamma_\theta \epsilon, f_{k_j}],$$

or with the intersection of this square with $\mathcal{A}_0$. But the area of each of these squares is $\gamma_\theta^2 \epsilon^2$. Thus the set $\mathcal{A}_0$ is completely covered by at most $\lceil \text{surf}(\mathcal{A}_0)/\gamma_\theta^2 \epsilon^2 \rceil$ such squares, where $\lceil \alpha \rceil$ is the smallest integer larger or equal to α, that is α rounded up. This puts a finite upper bound on the number of iterations in $\{k_j\}$. Hence (3.6) is impossible, and the conclusion follows. □

We next examine the size of the constraint violation before and after a normal iteration.

Lemma 3.3 Suppose that AS1 and AS3 hold, that $k \notin \mathcal{R}$ and that n_k satisfies (3.4). Then

$$\theta_k \leq \kappa_{\rm ubt} \Delta_k^{1+\mu} \tag{3.7}$$

and

$$\theta(x_k + s_k) \leq \kappa_{\rm ubt} \Delta_k^2. \tag{3.8}$$

for some constant $\kappa_{\rm ubt} \geq 0$.

Proof. If $k \notin \mathcal{R}$, we have, from (3.4) and (2.12) that

$$\kappa_{\rm lsc}\theta_k \leq \|n_k\| \leq \kappa_\Delta \kappa_\mu \Delta_k^{1+\mu}. \tag{3.9}$$

Now, the i-th constraint function at $x_k + s_k$ can be expressed as

$$c_i(x_k+s_k) = c_i(x_k) + \langle e_i, A_k s_k\rangle + \tfrac{1}{2}\langle s_k, \nabla_{xx}c_i(\xi_k)s_k\rangle,$$

for $i \in \mathcal{E}\cup\mathcal{I}$, where we have used AS1, the mean-value theorem, and where ξ_k belongs to the segment $[x_k, x_k+s_k]$. Using AS3, we may bound the Hessian of the constraint functions and we obtain from (2.7), the Cauchy-Schwartz inequality, and (2.6) we have that

$$|c_i(x_k+s_k)| \leq \tfrac{1}{2}\max_{x\in X}\|\nabla_{xx}c_i(x)\|\,\|s_k\|^2 \leq \kappa_1\Delta_k^2,$$

if $i \in \mathcal{E}$, or

$$-c_i(x_k+s_k) \leq \tfrac{1}{2}\max_{x\in X}\|\nabla_{xx}c_i(x)\|\,\|s_k\|^2 \leq \kappa_1\Delta_k^2,$$

if $i \in \mathcal{I}$, where we have defined

$$\kappa_1 \stackrel{\rm def}{=} \tfrac{1}{2}\max_{i\in\mathcal{E}\cup\mathcal{I}}\max_{x\in X}\|\nabla_{xx}c_i(x)\|.$$

This gives the desired bound with

$$\kappa_{\rm ubt} = \max[\kappa_1, \kappa_\Delta\kappa_\mu/\kappa_{\rm lsc}].$$

□

We next assess the model decrease when the trust-region radius is sufficiently small.

Lemma 3.4 Suppose that AS1, (2.14), AS3 and AS2 hold, that $k \notin \mathcal{R}$, that

$$\chi_k \geq \epsilon, \tag{3.10}$$

and that

$$\Delta_k \leq \min\left[\frac{\epsilon}{\kappa_{\rm umh}}, \left(2\frac{\kappa_{\rm ubg}}{\kappa_{\rm umh}\kappa_\Delta\kappa_\mu}\right)^{\frac{1}{1+\mu}}, \left(\frac{\kappa_{\rm tmd}\epsilon}{4\kappa_{\rm ubg}\kappa_\Delta\kappa_\mu}\right)^{\frac{1}{\mu}}\right] \stackrel{\rm def}{=} \delta_m, \tag{3.11}$$

where $\kappa_{\rm ubg} \stackrel{\rm def}{=} \max_{x\in X}\|\nabla_x f(x)\|$. Then

$$m_k(x_k) - m_k(x_k+s_k) \geq \tfrac{1}{2}\kappa_{\rm tmd}\epsilon\Delta_k.$$

Proof. We first note that, by (2.14), AS2, (3.10) and (3.11),

$$m_k(x_k^{\rm N}) - m_k(x_k+s_k) \geq \kappa_{\rm tmd}\chi_k \min\left[\frac{\chi_k}{\kappa_{\rm umh}}, \Delta_k\right] \geq \kappa_{\rm tmd}\epsilon\Delta_k. \tag{3.12}$$

Now

$$m_k(x_k^{\rm N}) = m_k(x_k) + \langle g_k, n_k\rangle + \tfrac{1}{2}\langle n_k, H_k n_k\rangle$$

and therefore, using the Cauchy-Schwartz inequality, (2.12) and (3.11) that

$$\begin{array}{rcl}
|m_k(x_k) - m_k(x_k^{\rm N})| & \leq & \|n_k\|\,\|g_k\| + \tfrac{1}{2}\|H_k\|\,\|n_k\|^2 \\
& \leq & \kappa_{\rm ubg}\|n_k\| + \tfrac{1}{2}\kappa_{\rm umh}\|n_k\|^2 \\
& \leq & \kappa_{\rm ubg}\kappa_\Delta\kappa_\mu\Delta_k^{1+\mu} + \tfrac{1}{2}\kappa_{\rm umh}\kappa_\Delta^2\kappa_\mu^2\Delta_k^{2(1+\mu)} \\
& \leq & 2\kappa_{\rm ubg}\kappa_\Delta\kappa_\mu\Delta_k^{1+\mu} \\
& \leq & \tfrac{1}{2}\kappa_{\rm tmd}\epsilon\Delta_k.
\end{array}$$

We thus conclude from this last inequality and (3.12) that the desired conclusion holds. □

We continue our analysis by showing, as the reader has grown to expect, that iterations have to be very successful when the trust-region radius is sufficiently small.

Lemma 3.5 Suppose that AS1, (2.14), AS3, AS2 and (3.10) hold, that $k \not\in \mathcal{R}$, and that

$$\Delta_k \leq \min\left[\delta_m, \frac{(1-\eta_2)\kappa_{\rm tmd}\epsilon}{2\kappa_{\rm ubh}}\right] \stackrel{\rm def}{=} \delta_\rho. \tag{3.13}$$

Then

$$\rho_k \geq \eta_2.$$

Proof. Using (2.18), (3.2), Lemma 3.4 and (3.13), we find that

$$|\rho_k - 1| \leq \frac{|f(x_k+s_k) - m_k(x_k+s_k)|}{|m_k(x_k) - m_k(x_k+s_k)|} \leq \frac{\kappa_{\rm ubh}\Delta_k^2}{\frac{1}{2}\kappa_{\rm tmd}\epsilon\Delta_k} \leq 1-\eta_2,$$

from which the conclusion immediately follows. □

Now, we also show that the test (2.17) will always be satisfied in the above circumstances.

Lemma 3.6 Suppose that AS1, (2.14), AS3, AS2 and (3.10) hold, that $k \not\in \mathcal{R}$, that n_k satisfies (3.4), and that

$$\Delta_k \leq \min\left[\delta_m, \left(\frac{\kappa_{\rm tmd}\epsilon}{2\kappa_\theta\kappa_{\rm ubt}^2}\right)^{\frac{1}{1+2\mu}}\right] \stackrel{\rm def}{=} \delta_f. \tag{3.14}$$

Then

$$m_k(x_k) - m_k(x_k+s_k) \geq \kappa_\theta\theta_k^2.$$

Proof. This directly results from the inequalities

$$\kappa_\theta \theta_k^2 \leq \kappa_\theta \kappa_{\rm ubt}^2 \Delta_k^{2(1+\mu)} \leq \tfrac{1}{2}\kappa_{\rm tmd}\epsilon\Delta_k \leq m_k(x_k) - m_k(x_k+s_k),$$

where we successively used Lemma 3.3, (3.14) and Lemma 3.4. □

We may also guarantee a decrease in the objective function, large enough to ensure that the trial point is acceptable with respect to the (θ, f)-pair associated with x_k, so long as the constraint violation is itself sufficiently small.

Lemma 3.7 Suppose that AS1, (2.14), AS3, AS2, (3.10) and (3.13) hold, that $k \notin \mathcal{R}$, that n_k satisfies (3.4), and that

$$\theta_k \leq \kappa_{\rm ubt}^{-\frac{1}{\mu}} \left(\frac{\eta_2 \kappa_{\rm tmd}\epsilon}{2\gamma_\theta}\right)^{\frac{1+\mu}{\mu}} \stackrel{\rm def}{=} \delta_\theta. \tag{3.15}$$

Then

$$f(x_k+s_k) \leq f(x_k) - \gamma_\theta\theta_k.$$

Proof. Applying Lemmas 3.3–3.5—which is possible because of (3.10), (3.13), $k \notin \mathcal{R}$ and n_k satisfies (3.4)—and (3.15), we obtain that

$$\begin{array}{rcl} f(x_k) - f(x_k+s_k) & \geq & \eta_2[m_k(x_k) - m_k(x_k+s_k)] \\ & \geq & \frac{1}{2}\eta_2\kappa_{\rm tmd}\epsilon\Delta_k \\ & \geq & \frac{1}{2}\eta_2\kappa_{\rm tmd}\epsilon\left(\frac{\theta_k}{\kappa_{\rm ubt}}\right)^{\frac{1}{1+\mu}} \\ & \geq & \gamma_\theta\theta_k \end{array}$$

and the desired inequality follows. □

We now establish that if the trust-region radius and the constraint violation are both small at a non-critical iterate x_k, TRQP(x_k, Δ_k) must be compatible.

Lemma 3.8 Suppose that AS1, (2.10), (2.14), AS3, AS2, and (3.10) hold, and that

$$\Delta_k \leq \min\left[\gamma_0\delta_\rho, \left(\frac{1}{\kappa_\mu}\right)^{\frac{1}{\mu}}, \left(\frac{\gamma_0^2(1-\gamma_\theta)\kappa_\Delta\kappa_\mu}{\kappa_{\rm usc}\kappa_{\rm ubt}}\right)^{\frac{1}{1-\mu}}\right]. \tag{3.16}$$

Suppose furthermore that θ_k is arbitrarily small. Then $k \notin \mathcal{R}$.

Proof. Because θ_k is arbitrarily small, we know from (2.10) and Lemma 3.1 that n_k exists and satisfies (2.10) and (3.4), and also that (3.15) holds. Assume, for the purpose of deriving a contradiction, that $k \in \mathcal{R}$, that is

$$\|n_k\| > \kappa_\Delta\kappa_\mu\Delta_k^{1+\mu}, \tag{3.17}$$

where we have used (2.12) and the fact that $\Delta_k \leq 1$. In this case, the mechanism of the algorithm then ensures that $k-1 \notin \mathcal{R}$. Now assume that iteration $k-1$ is unsuccessful. Because of Lemmas 3.5 and 3.7, which hold at iteration $k-1 \notin \mathcal{R}$ because of (3.16), the fact that $\theta_k = \theta_{k-1}$, (2.10), and (3.15), we obtain that

$$\rho_{k-1} \geq \eta_2 \text{ and } f(x_{k-1}+s_{k-1}) \leq f(x_{k-1}) - \gamma_\theta \theta_{k-1}.$$

Hence, given that x_{k-1} is acceptable for the filter at the beginning of iteration $k-1$, if this iteration is unsuccessful, it must be because

$$\theta(x_{k-1}+s_{k-1}) > (1-\gamma_\theta)\theta_{k-1} = (1-\gamma_\theta)\theta_k.$$

But Lemma 3.3 and the mechanism of the algorithm then imply that

$$(1-\gamma_\theta)\theta_k \leq \kappa_{\text{ubt}}\Delta_{k-1}^2 \leq \frac{\kappa_{\text{ubt}}}{\gamma_0^2}\Delta_k^2.$$

Combining this last bound with (3.17) and (2.10), we deduce that

$$\kappa_\Delta \kappa_\mu \Delta_k^{1+\mu} < \|n_k\| \leq \kappa_{\text{usc}}\theta_k \leq \frac{\kappa_{\text{usc}}\kappa_{\text{ubt}}}{\gamma_0^2(1-\gamma_\theta)}\Delta_k^2$$

and hence that

$$\Delta_k^{1-\mu} > \frac{\gamma_0^2(1-\gamma_\theta)\kappa_\Delta\kappa_\mu}{\kappa_{\text{usc}}\kappa_{\text{ubt}}}.$$

Since this last inequality contradicts (3.16), our assumption that iteration $k-1$ is unsuccessful must be false. Thus iteration $k-1$ is successful and $\theta_k = \theta(x_{k-1}+s_{k-1})$. We then obtain from (3.17), (2.10) and (3.8) that

$$\kappa_\Delta \kappa_\mu \Delta_k^{1+\mu} < \|n_k\| \leq \kappa_{\text{usc}}\theta_k \leq \kappa_{\text{usc}}\kappa_{\text{ubt}}\Delta_{k-1}^2 \leq \frac{\kappa_{\text{usc}}\kappa_{\text{ubt}}}{\gamma_0^2}\Delta_k^2,$$

which is again impossible because of (3.16) and because $(1-\gamma_\theta) < 1$. Hence our initial assumption (3.17) must be false, which yields the desired conclusion. □

We now distinguish two mutually exclusive cases. For the first, we consider what happens if there happens to be an infinite subsequence of iterates belonging to the filter.

Lemma 3.9 Suppose that AS1, (2.10), (2.14), AS3 and AS2 hold. Suppose furthermore that there exists an infinite subsequence $\{k_j\} \in \mathcal{Z}$. Then we have that, either the restoration procedure terminates unsuccessfully, or there exists a subsequence $\{k_i\} \subseteq \{k_j\}$ such that

$$\lim_{i\to\infty} \theta_{k_i} = 0 \tag{3.18}$$

and

$$\lim_{i\to\infty} \chi_{k_i} = 0. \tag{3.19}$$

Proof. Suppose that the restoration procedure always terminates successfully. Let $\{k_i\}$ be any infinite subsequence of $\{k_j\}$ such that (3.18) holds, which is possible because of Lemma 3.2. Combining this with (2.10) ensures that n_{k_i} exists for $i \geq i_0$, say, and also that

$$\lim_{i\to\infty} \|n_{k_i}\| = 0. \tag{3.20}$$

As we noted in the proof of Lemma 3.4,

$$|m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i}^{\rm N})| \leq \kappa_{\rm ubg}\|n_{k_i}\| + \tfrac{1}{2}\kappa_{\rm umh}\|n_{k_i}\|^2,$$

which in turn, with (3.20), yields that

$$\lim_{i\to\infty} [m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i}^{\rm N})] = 0. \tag{3.21}$$

Suppose now that

$$\chi_{k_i} \geq \epsilon_2 > 0 \tag{3.22}$$

for all i and some $\epsilon_2 > 0$. Suppose furthermore that there exists $\epsilon_3 > 0$ such that, for all $i \geq i_0$,

$$\Delta_{k_i} \geq \epsilon_3. \tag{3.23}$$

Then we deduce from (2.14) and AS2 that

$$m_{k_i}(x_{k_i}^{\rm N}) - m_{k_i}(x_{k_i} + s_{k_i}) \geq \kappa_{\rm tmd}\epsilon_2 \min\left[\frac{\epsilon_2}{\kappa_{\rm umh}}, \epsilon_3\right] \stackrel{\rm def}{=} \delta > 0. \tag{3.24}$$

We now decompose the model decrease in its normal and tangential components, that is

$$m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i} + s_{k_i}) = m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i}^{\rm N}) + m_{k_i}(x_{k_i}^{\rm N}) - m_{k_i}(x_{k_i} + s_{k_i}).$$

Substituting (3.21) and (3.24) into this decomposition, we find that

$$\lim_{i\to\infty} [m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i} + s_{k_i})] \geq \delta > 0. \tag{3.25}$$

We now observe that, because x_{k_i} is added to the filter at iteration k_i, we know from the mechanism of the algorithm and from Lemma 3.8 that either iteration $k_i \in \mathcal{R}$ or (2.17) must fail. If $k_i \in \mathcal{R}$ for $i \geq i_0$, we obtain from (2.12) that,

$$\|n_{k_i}\| > \kappa_\Delta\kappa_\mu\Delta_{k_i}^{1+\mu},$$

and (3.20) then implies that Δ_{k_i} is arbitrarily small for i sufficiently large. This contradicts (3.23) and, hence, (2.17) must fail for i sufficiently large, that is

$$m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i} + s_{k_i}) < \kappa_\theta\theta_{k_i}^2. \tag{3.26}$$

Combining this bound with (3.25), we find that θ_{k_i} is bounded away from zero for i sufficiently large, which is impossible in view of (3.18). We therefore deduce that (3.23) cannot hold and obtain that there is a subsequence $\{k_\ell\} \subseteq \{k_i\}$ for which

$$\lim_{\ell\to\infty} \Delta_{k_\ell} = 0.$$

We now restrict our attention to the tail of this subsequence, that is to the set of indices k_ℓ that are large enough to ensure that (3.16) holds (with ϵ replaced by ϵ_2) and that (3.15) also holds, which is possible because of (3.18). For these indices, we may therefore apply Lemma 3.8, and deduce that iteration $k_\ell \not\in \mathcal{R}$ for ℓ sufficiently large. Hence, as above, (3.26) must hold for ℓ sufficiently large. However, we may also apply Lemma 3.6, which contradicts (3.26), and therefore (3.22) cannot hold, yielding that

$$\liminf_{i\to\infty} \chi_{k_i} = 0.$$

The required result then follows from the fact that we have chosen the subsequence $\{k_i\}$ such that (3.18) holds. □

Thus, if an infinite subsequence of iterates is added to the filter, it contains a subsequence that converges to a first-order critical point. Our remaining analysis then naturally concentrates on the possibility that there may be no such infinite subsequence. In this case, no further iterates are added to the filter for k sufficiently large. In particular, this means that the number of restoration iterations, $|\mathcal{R}|$, must be finite. In what follows, we assume that $k_0 \geq 0$ is the last iteration for which x_{k_0-1} is added to the filter.

Lemma 3.10 Suppose that AS1, (2.10), (2.14), AS3 and AS2 hold. Suppose furthermore that (2.17) holds for all $k \geq k_0$. Then we have that

$$\lim_{k\to\infty} \theta_k = 0. \tag{3.27}$$

Furthermore, n_k exists and satisfies (2.10) and (3.4) for all k sufficiently large.

Proof. Consider any successful iterate with $k \geq k_0$. Then we have that

$$f(x_k) - f(x_{k+1}) \geq \eta_1[m_k(x_k) - m_k(x_k + s_k)] \geq \eta_1 \kappa_\theta \theta_k^2 \geq 0. \tag{3.28}$$

Thus the objective function does not increase for all successful iterations with $k \geq k_0$. But AS1 and AS3 imply (3.3) and therefore we must have, from the first part of this statement, that

$$\lim_{k\to\infty} f(x_k) - f(x_{k+1}) = 0. \tag{3.29}$$

(3.27) then immediately follows from (3.28) and the fact that $\theta_j = \theta_k$ for all unsuccessful iterations j that immediately follow the successful iteration k, if any. The last conclusion then results from (2.10). □

We now show that the trust-region radius cannot become arbitrarily small if the (asymptotically feasible) iterates stay away from first-order critical points.

Lemma 3.11 Suppose that AS1, (2.14), AS3 and AS2 hold. Suppose furthermore that (2.17) and (3.10) hold for all $k \geq k_0$. Then there exists a $\Delta_{\min} > 0$ such that

$$\Delta_k \geq \Delta_{\min}$$

for all k.

Proof. Suppose that $k_1 \geq k_0$ is chosen sufficiently large to ensure that (3.15) holds and that n_k exists and satisfies (2.10) for all $k \geq k_1$, which is possible because of Lemma 3.10. Suppose also, for the purpose of obtaining a contradiction, that iteration j is the first iteration following iteration k_1 for which

$$\Delta_j \leq \gamma_0 \min\left[\delta_\rho, \sqrt{\frac{(1-\gamma_\theta)\theta^{\mathrm{F}}}{\kappa_{\mathrm{ubt}}}}, \Delta_{k_1}\right] \stackrel{\mathrm{def}}{=} \gamma_0 \delta_s, \tag{3.30}$$

where

$$\theta^{\mathrm{F}} \stackrel{\mathrm{def}}{=} \min_{i \in \mathcal{Z}} \theta_i$$

is the smallest constraint violation appearing in the filter. Note that the inequality $\Delta_j \leq \gamma_0 \Delta_{k_1}$, which is implied by (3.30), ensures that $j \geq k_1 + 1$ and hence that $j - 1 \geq k_1$. Then the mechanism of the algorithm implies that

$$\Delta_{j-1} \leq \delta_s \tag{3.31}$$

and Lemma 3.5, which is applicable because (3.30) and (3.31) together imply (3.13) with k replaced by $j-1$, then ensures that

$$\rho_{j-1} \geq \eta_2. \tag{3.32}$$

Furthermore, Lemma 3.3, (3.30) and (3.31) give that

$$\theta(x_{j-1} + s_{j-1}) \leq \kappa_{\mathrm{ubt}} \Delta_{j-1}^2 \leq (1-\gamma_\theta)\theta^{\mathrm{F}}. \tag{3.33}$$

We may also apply Lemma 3.7 because (3.30) and (3.31) ensure that (3.13) holds and because (3.15) also holds for $j \geq k_1$. Hence we deduce that

$$f(x_{j-1} + s_{j-1}) \leq f(x_{j-1}) - \gamma_\theta \theta_{j-1}.$$

This last relation and (3.33) ensure that $x_{j-1} + s_{j-1}$ is acceptable for the filter. Combining this conclusion with (3.32), the fact that (2.17) holds for iteration $j-1$ and the mechanism of the algorithm, we obtain that $\Delta_j \geq \Delta_{j-1}$. As a consequence, and since (2.17) also holds at iteration $j-1$, iteration j cannot be the first iteration following k_1 for which (3.30) holds. This contradiction shows that $\Delta_k \geq \gamma_0 \delta_s$ for all $k > k_1$, and the desired result follows if we define

$$\Delta_{\min} = \min[\Delta_0, \ldots, \Delta_{k_1}, \gamma_0 \delta_s].$$

□

We may now analyze the convergence of χ_k itself.

Lemma 3.12 Suppose that AS1, (2.14), AS3 and AS2 hold. Suppose furthermore that (2.17) holds for all $k \geq k_0$. Then

$$\liminf_{k\to\infty} \chi_k = 0. \tag{3.34}$$

Proof. We start by observing that Lemma 3.10 implies that (2.10) holds for k sufficiently large. Moreover, as in Lemma 3.10, we obtain (3.28) and therefore (3.29) for each $k \in \mathcal{S}$, $k \geq k_0$. Suppose now, for the purpose of obtaining a contradiction, that (3.10) holds and, as for the case where a subsequence of iterates is included in the filter, notice that

$$m_k(x_k) - m_k(x_k + s_k) = m_k(x_k) - m_k(x_k^{\rm N}) + m_k(x_k^{\rm N}) - m_k(x_k + s_k). \tag{3.35}$$

Moreover, note, as in Lemma 3.4, that

$$|m_k(x_k) - m_k(x_k^{\rm N})| \leq \kappa_{\rm ubg}\|n_k\| + \kappa_{\rm umh}\|n_k\|^2,$$

which in turn yields that

$$\lim_{k\to\infty} [m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i}^{\rm N})] = 0$$

because of Lemma 3.10 and (2.10). This limit, together with (3.28), (3.29) and (3.35), then gives that

$$\lim_{\substack{k\to\infty\\ k\in\mathcal{S}}} [m_k(x_k^{\rm N}) - m_k(x_k + s_k)] = 0. \tag{3.36}$$

But (2.14), (3.10), AS2 and Lemma 3.11 together imply that, for all $k \geq k_0$

$$m_k(x_k^{\rm N}) - m_k(x_k + s_k) \geq \kappa_{\rm tmd}\chi_k \min\left[\frac{\chi_k}{\beta_k}, \Delta_k\right] \geq \kappa_{\rm tmd}\epsilon \min\left[\frac{\epsilon}{\kappa_{\rm umh}}, \Delta_{\min}\right], \tag{3.37}$$

immediately giving a contradiction with (3.36). Hence (3.10) cannot hold and the desired result follows. □

We may summarize all of the above in our main global convergence result.

Theorem 3.13 Suppose that AS1, (2.10), (2.14), AS3 and AS2 hold. Let $\{x_k\}$ a sequence of iterates produced by Algorithm 2.1. Then either the restoration procedure terminates unsuccessfully by converging to an infeasible first-order critical point of problem (2.15), or there is a subsequence $\{k_j\}$ for which

$$\lim_{j\to\infty} x_{k_j} = x_*$$

and x_* is a first-order critical point for problem (1.1).

Proof. Suppose that the restoration iteration always terminates successfully. From Lemmas 3.9, 3.10 and 3.12, we obtain that, for some subsequence $\{k_j\}$,

$$\lim_{j\to\infty} \theta_{k_j} = \lim_{j\to\infty} \chi_{k_j} = 0.$$

The convergence of θ_{k_j} to zero and (2.10) then give that

$$\lim_{j\to\infty} \|n_{k_j}\| = 0,$$

and therefore that

$$\lim_{j\to\infty} \|x^{\rm N}_{k_j} - x_{k_j}\| = 0.$$

The conclusion then follows from the continuity of both θ and χ. □

Can we dispense with AS3 to obtain this result? Firstly, this assumption ensures that the objective and constraint functions remain bounded above and below (see (3.3)). This is crucial for the rest of the analysis because the convergence of the iterates to feasibility depends on the fact that the area of the filter is finite. Thus, if AS3 does not hold, we have to verify that (3.3) holds for other reasons. The second part of this statement may be ensured quite simply by initializing the filter to $(\theta^{\rm max}, -\infty)$, for some $\theta^{\rm max} > \theta_0$, in Step 0 of the algorithm. This has the effect of putting an upper bound on the infeasibility of all iterates, which may be useful in practice. However, this does not prevent the objective function from being unbounded below in

$$\mathcal{C}(\theta^{\rm max}) = \{x \in \mathbb{R}^n \mid \theta(x) \le \theta^{\rm max}\}$$

and we cannot exclude the possibility that a sequence of infeasible iterates might both continue to improve the value of the objective function and satisfy (2.17). If $\mathcal{C}(\theta^{\rm max})$ is bounded, AS3 is most certainly satisfied. If this is not the case, we could assume that

$$f^{\rm min} \le f(x) \le f^{\rm max} \;\text{ and }\; 0 \le \theta(x) \le \theta^{\rm max} \;\text{ for }\; x \in \mathcal{C}(\theta^{\rm max}) \tag{3.38}$$

for some values of $f^{\rm min}$ and $f^{\rm max}$ and simply monitor that the values $f(x_k)$ are reasonable—in view of the problem being solved—as the algorithm proceeds. To summarize, we may replace AS1 and AS3 by the following assumption.

AS4: The functions f and c are twice continuously differentiable on an open set containing $\mathcal{C}(\theta^{\rm max})$ and (3.38) holds.

The reader should note that the comments following the statement of (2.10) no longer apply if limit points at infinity are allowed.

4 An Alternative Step Strategy

It is also interesting to return to the question of whether it is possible to find a cheaper alternative to computing a normal step, finding a generalized Cauchy point and explicitly checking (2.14). Suppose, for now, that it is possible to compute a point $x_k + s_k{}'$

directly to satisfy the constraints of TRQP(x_k, Δ_k) and for which

$$m_k(x_k) - m_k(x_k + {s_k}') \geq \epsilon_1 \min[\pi_k, \Delta_k] \tag{4.1}$$

for a small positive constant ϵ_1 and $\pi_k = \pi(x_k)$, where π is a continuous function of its argument. Furthermore, consider the following algorithm.

Algorithm 4.1: Single-step SQP-Filter Algorithm

As Algorithm 2.1 (with s_k replaced by ${s_k}'$), except that Step 1 and 2 are replaced by the following.

Step 1: Ensure compatibility. If the feasible set of TRQP(x_k, Δ_k) is empty, include x_k in the filter and compute a restoration step r_k such that TRQP$(x_k + r_k, \Delta_{k+1})$ is compatible for some $\Delta_{k+1} > 0$, and $x_k + r_k$ is acceptable for the filter. If this proves impossible, stop. Otherwise, define x_{k+1} to be $x_k + r_k$ and go to Step 6.

Step 2: Determine a trial step. Compute a step ${s_k}'$ that is feasible for this subproblem and such that (4.1) holds.

Interestingly, most of the properties of Algorithm 2.1 remain true for the modified Algorithm 4.1, as we shall now see by reconsidering the convergence theory of the previous section. Lemmas 3.1 and 3.3 are unmodified. Replacing n_k by ${s_k}'$ in Lemma 3.2 yields (3.7) with $\mu = 0$— if ${s_k}'$ can be computed, this implies that we do not have to worry about the existence of n_k. As the proof of (3.8) is also unmodified, we conclude that Lemma 3.3 remains true with $\mu = 0$. We now suppose, instead of (3.10), that

$$\pi_k \geq \epsilon \tag{4.2}$$

and obtain the conclusion of Lemma 3.4 immediately from (4.1). Lemma 3.5 is again unmodified, while Lemma 3.6 remains true if one uses the modified version of Lemma 3.3 (with $\mu = 0$) to deduce its conclusion. The same applies to Lemma 3.7. The proof of Lemma 3.8 remains true but needs a slightly less trivial modification. In particular, the bound in (3.16) needs to be understood with the particular values

$$\mu = 0 \text{ and } \kappa_\Delta = \kappa_\mu = 1. \tag{4.3}$$

The existence of n_k at the beginning of the proof is a direct result of the existence of ${s_k}'$. The rest of the proof follows immediately using (4.3). We may dispense with its first part of the proof of Lemma 3.9, and immediately assume that

$$\pi_{k_i} \geq \epsilon_2 \tag{4.4}$$

(instead of (3.22)). We may then continue the proof of this lemma as stated using (4.3), except that we do not have to consider the decomposition of the step into tangential and normal components to obtain (3.25) from (3.24), and that we should substitute ${s_{k_i}}'$ for n_{k_i} when necessary. Lemma 3.10 and 3.11 are unmodified. The proof of Lemma 3.12 simplifies because we may deduce (3.37) directly from (4.4) (instead of (3.22)) and (4.1). Gathering those results, we may therefore deduce the following result.

Theorem 4.1 Suppose that AS1, AS2 and AS3 hold. Let $\{x_k\}$ be a sequence of iterates produced by Algorithm 4.1. Then either the restoration procedure terminates unsuccessfully by converging to an infeasible first-order critical point of problem (2.15), or there is a subsequence $\{k_j\}$ for which

$$\lim_{j\to\infty} \theta_{k_j} = 0$$

and

$$\lim_{j\to\infty} \pi_{k_j} = 0.$$

Thus we have established convergence of a subsequence to a first-order critical point provided we can exhibit a continuous function $\pi(x)$ such that

$$\pi(x_*) = \theta(x_*) = 0$$

is equivalent to the first-order criticality of x_* and such that a step satisfying (4.1) can always be found. For instance, we might consider

$$\pi(x) = \|g(x) - A(x)^T y^{\rm LS}(x)\|$$

where $y^{\rm LS}(x)$ is the vector of least-squares multipliers at x. We may even replace this vector at iteration k by any y_k provided we can guarantee that $\|y_k - y_k^{\rm LS}\|$ converges to zero when θ_k converges to zero. Of course, the main difficulty is still to find a step ${s_k}'$ satisfying (4.1). If we have difficulty in finding such a step for a particular iteration, we can always return to the composite-step technique we considered in the previous section. This then results in a somewhat hybrid but potentially more efficient algorithm.

5 Conclusion and Perspectives

We have defined two variants of a trust-region SQP-filter algorithm for general nonlinear programming, and have shown these algorithms to be globally convergent to first-order critical points. Since Fletcher and Leyffer (1997) indicates that such algorithms may be very efficient in practice, the theory developed in this paper provides the reassurance that they also have reasonable convergence properties, which then makes these methods very attractive.

We are however aware that the convergence study is not complete, as we have not discussed local convergence properties. It is very likely that such a study will have to introduce second-order corrections (see Fletcher, 1987, Section 14.4) to ensure that the Maratos effect does not take place and that a fast (quadratic) rate of convergence can be achieved. Moreover, convergence to a second-order critical points also remains, for now, an open question. These questions are the subject of ongoing work, and will hopefully be reported on in the near future.

References

L. T. Biegler, J. Nocedal, and C. Schmid. A reduced Hessian method for large-scale constrained optimization. *SIAM Journal on Optimization*, **5**(2), 314–347, 1995.

R. H. Bielschowsky and F. A. M. Gomes. Dynamical control of infeasibility in nonlinearly constrained optimization. Presentation at the Optimization 98 Conference, Coimbra, 1998.

R. H. Byrd, J. Ch. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. Technical Report 2896, INRIA, Rocquencourt, France, 1996.

R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. Technical Report 97/05, Optimization Technology Center, Argonnne National Laboratory, Argonne, Illinois, USA, 1997.

R. H. Byrd, R. B. Schnabel, and G. A. Shultz. A trust region algorithm for nonlinearly constrained optimization. *SIAM Journal on Numerical Analysis*, **24**, 1152–1170, 1987.

A. R. Conn, N. I. M. Gould, A. Sartenaer, and Ph. L. Toint. Global convergence of a class of trust region algorithms for optimization using inexact projections on convex constraints. *SIAM Journal on Optimization*, **3**(1), 164–221, 1993.

R. S. Dembo and U. Tulowitzki. On the minimization of quadratic functions subject to box constraints. School of Organization and Management Working paper Series B no. 71, Yale University, Yale, USA, 1983.

J. E. Dennis, M. El-Alem, and K. A. Williamson. A trust-region approach to nonlinear systems of equalities and inequalities. *SIAM Journal on Optimization*, **9**(2), 291–315, 1999.

M. El-Alem. Global convergence without the assumption of linear independence for a trust-region algorithm for constrained optimization. *Journal of Optimization Theory and Applications*, **87**(3), 563–577, 1995.

M. El-Alem. A global convergence theory for a general class of trust-region-based algorithms for constrained optimization without assuming regularity. Technical Report TR96-08, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, USA, 1996.

M. El-Hallabi and R. A. Tapia. An inexact trust-region feasible-point algorithm for nonlinear systems of equalities and inequalities. Technical Report TR95-09, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, USA, 1995.

R. Fletcher. *Practical Methods of Optimization*. J. Wiley and Sons, Chichester, England, second edn, 1987.

R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. Numerical Analysis Report NA/171, Department of Mathematics, University of Dundee, Dundee, Scotland, 1997.

R. Fletcher and S. Leyffer. User manual for filterSQP. Numerical Analysis Report NA/181, Department of Mathematics, University of Dundee, Dundee, Scotland, 1998.

R. Fletcher, S. Leyffer, and Ph. L. Toint. On the global convergence of an SLP-filter algorithm. Technical Report 98/13, Department of Mathematics, FUNDP, Namur, Belgium, 1998.

M. Lalee, J. Nocedal, and T. D. Plantenga. On the implementation of an algorithm for large-scale equality constrained optimization. *SIAM Journal on Optimization*, **8**(3), 682–706, 1998.

X. Liu and Y. Yuan. A robust trust-region algorithm for solving general nonlinear programming problems. Presentation at the International Conference on Nonlinear Programming and Variational Inequalities, Hong Kong, 1998.

W. Murray and F. J. Prieto. A sequential quadratic programming algorithm using an incomplete solution of the subproblem. *SIAM Journal on Optimization*, **5**(3), 590–640, 1995.

E. O. Omojokun. *Trust region algorithms for optimization with nonlinear equality and inequality constraints*. PhD thesis, University of Colorado, Boulder, Colorado, USA, 1989.

A. Sartenaer. Automatic determination of an initial trust region in nonlinear programming. *SIAM Journal on Scientific Computing*, **18**(6), 1788–1803, 1997.

Ph. L. Toint. Global convergence of a class of trust region methods for nonconvex minimization in Hilbert space. *IMA Journal of Numerical Analysis*, **8**(2), 231–252, 1988.

Y. Yuan. Trust region algorithms for nonlinear equations. Technical Report 049, Dept of Mathematics, Hong Kong University, Hong Kong, 1994.

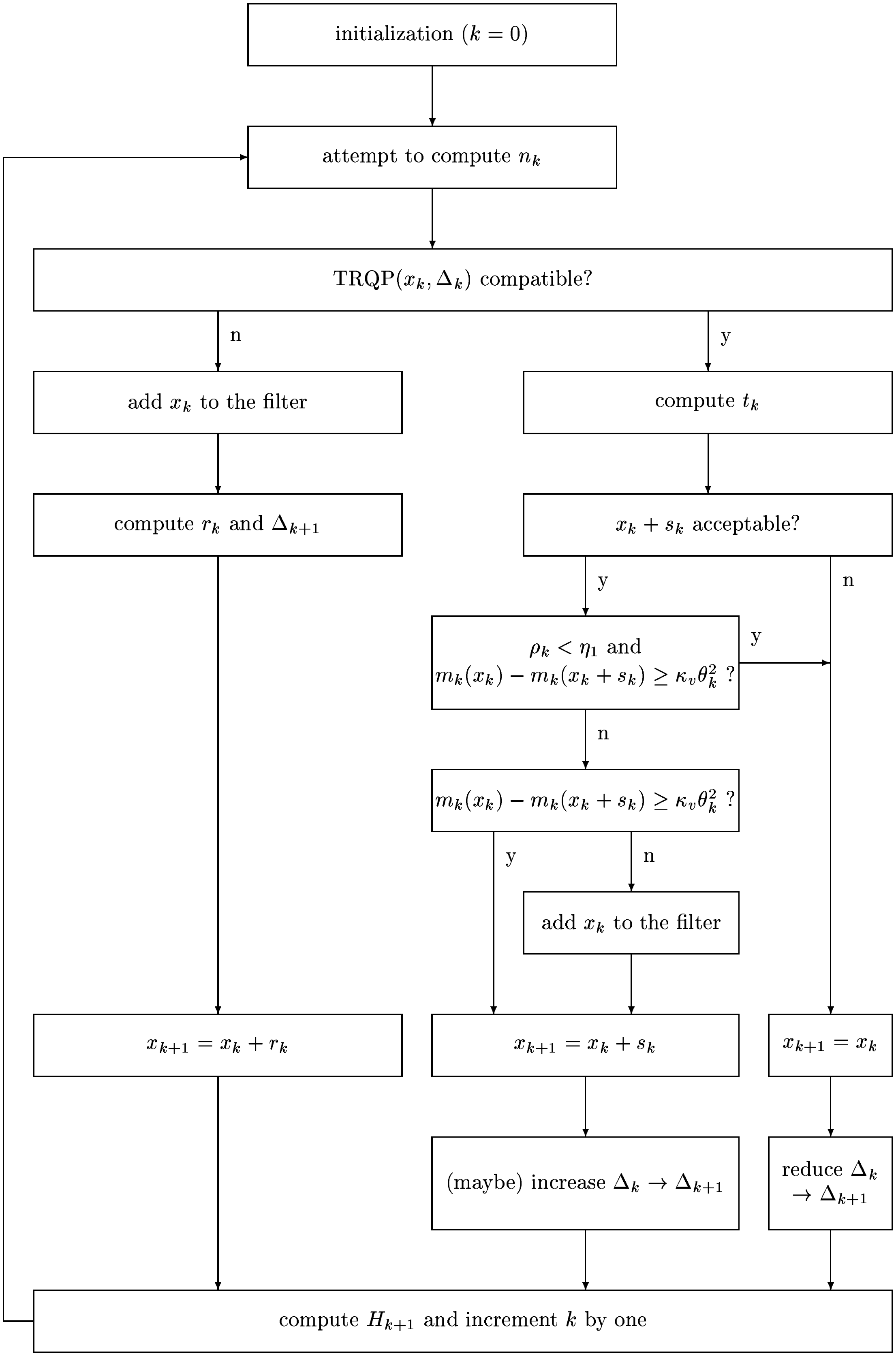
initialization ($k = 0$)
attempt to compute n_k
TRQP(x_k, Δ_k) compatible?
n
y
add x_k to the filter
compute t_k
compute r_k and Δ_{k+1}
$x_k + s_k$ acceptable?
y
n
$\rho_k < \eta_1$ and $m_k(x_k) - m_k(x_k + s_k) \geq \kappa_v \theta_k^2$?
y
n
$m_k(x_k) - m_k(x_k + s_k) \geq \kappa_v \theta_k^2$?
y
n
add x_k to the filter
$x_{k+1} = x_k + r_k$
$x_{k+1} = x_k + s_k$
$x_{k+1} = x_k$
(maybe) increase $\Delta_k \to \Delta_{k+1}$
reduce Δ_k $\to \Delta_{k+1}$
compute H_{k+1} and increment k by one

Figure 6.1: Flowchart of the algorithm.