A SEQUENTIAL QUADRATIC PROGRAMMING ALGORITHM WITH A PIECEWISE LINEAR MERIT FUNCTION

FRANCISCO A. M. GOMES*

Abstract. A sequential quadratic programming algorithm for solving nonlinear programming problems is presented. The new feature of the algorithm is related to the definition of the merit function. Instead of using one penalty parameter per iteration and increasing it as the algorithm progresses, we suggest that a new point is to be accepted if it stays sufficiently below the piecewise linear function defined by some previous iterates on the $(f, \|C\|_2^2)$ space. Therefore, the penalty parameter is allowed to decrease between successive iterations. Besides, one need not to decide how to update the penalty parameter. This approach resembles the filter method introduced by Fletcher and Leyffer [7], but it is less tolerant since a merit function is still used.

Key words. nonlinear programming, sequential quadratic programming, merit functions.

AMS subject classifications. 65K05, 90C55, 90C30, 90C26

1. Introduction. In this paper we are concerned with the problem

$$\begin{aligned} &\textit{minimize} && f(x) \\ &\textit{subject to} && C(x) = 0 \\ & && l \le x \le u \end{aligned} \tag{1.1}$$

where $f : I\!R^n \to I\!R$ is a C^2 nonlinear function, $C : I\!R^n \to I\!R^m$ represents a set of C^2 nonlinear constraints and we suppose that $-\infty \le l_i \le u_i \le \infty$, for $i = 1, \ldots, n$. Naturally, some of the components of x in (1.1) may be slack variables generated when converting inequality constrains to this form.

Algorithms based on the sequential quadratic programming (SQP) approach are one of the most effective methods for solving (1.1). Some interesting algorithms of this class are given, for example, in [2, 3, 9, 15]. A complete coverage of such methods can be found in [5, 14].

Since SQP algorithms do not require the iterates to be feasible, they have to concern with two conflicting objectives at each iteration: the reduction of the infeasibility and the reduction of function f. Both objectives must be taken in acount when deciding if the new iterate is to be accepted or rejected. To make this choice, most algorithms rely on a merit function.

If the problem contains no inequality constraints or bounds on the variables, or if the algorithm assures that the bounds are never violated by the iterates, the augmented Lagrangian, written here in an unusual way as

$$\mathcal{L}(x, \lambda, \theta) = \theta[f(x) + C(x)^T\lambda] + \frac{(1-\theta)}{2}\|C(x)\|_2^2, \tag{1.2}$$

is a good choice for the merit function.

In (1.2), θ is a "penalty parameter" used as a weight to balance the Lagrangian function for the equality constrained problem, defined as

$$\ell(x, \lambda) = f(x) + C(x)^T\lambda,$$

*University of Campinas – IMECC – Department of Applied Mathematics. CP 6065, 13081-970, Campinas, São Paulo, Brazil (chico@ime.unicamp.br).

with a measure of the infeasibility, given by

$$\varphi(x) = \frac{1}{2}\|C(x)\|_2^2.$$

When the bounds on the variables are not supposed to be satisfied by the iterates or when inequality constraints are explicitly handled by the SQP algorithm, a good merit fuction is the L_1 exact penalty function. For the problem (1.1), this function can be defined as

$$\Psi(x,\theta) = f(x) + \theta\|C(x)\|_1 + \theta\sum_{i=1}^{n}(u_i - x_i)^- + \theta\sum_{i=1}^{n}(x_i - l_i)^-,$$

where $w^- = \max\{0, -w\}$. Again, the penalty parameter θ is used to establish an equilibrium between optimality and feasibility.

At iteration k, a new point $x_+ = x_k + s$ is accepted if the ratio between the actual and the predicted reduction of the merit function (when moving from x_k to x_+) is greater than a positive constant.

When the augmented Lagrangian is used, the actual reduction of the merit function at the candidate point x_+ is defined as

$$A_{red}(x_k, s, \theta) = \mathcal{L}(x_k, \theta) - \mathcal{L}(x_k + s, \theta).$$

The predicted reduction of the merit function depends on the strategy used to approximately solve (1.1). One common choice is to approximate (1.1) by the quadratic programming problem

$$\begin{aligned} &\textit{minimize} && \overline{Q}(H,x,\lambda,s) = \frac{1}{2}s^T Hs + \nabla\ell(x,\lambda)^T s + \ell(x,\lambda) \\ &\textit{subject to} && A(x)s + C(x) = 0 \\ &&& l \le x + s \le u \end{aligned}$$

where H is a symmetric $n \times n$ matrix and $A(x) = (\nabla C_1(x), \ldots, \nabla C_m(x))^T$ is the Jacobian of the constraints.

In this case, denoting

$$M(x,s) = \frac{1}{2}\|A(x)s + C(x)\|_2^2,$$

as the approximation of $\varphi(x)$, the predicted reduction of the augmented Lagrangian merit function is given by

$$P_{red}(H,x,s,\theta) = \theta P_{red}^{opt}(H,x,s) + (1-\theta)P_{red}^{fsb}(x,s), \tag{1.3}$$

where

$$P_{red}^{fsb}(x,s) = M(x,0) - M(x,s) \tag{1.4}$$

is the predicted reduction of the infeasibility and

$$P_{red}^{opt}(H,x,s) = \overline{Q}(H,x,0) - \overline{Q}(H,x,s) \tag{1.5}$$

is the predicted reduction of the Lagrangian.

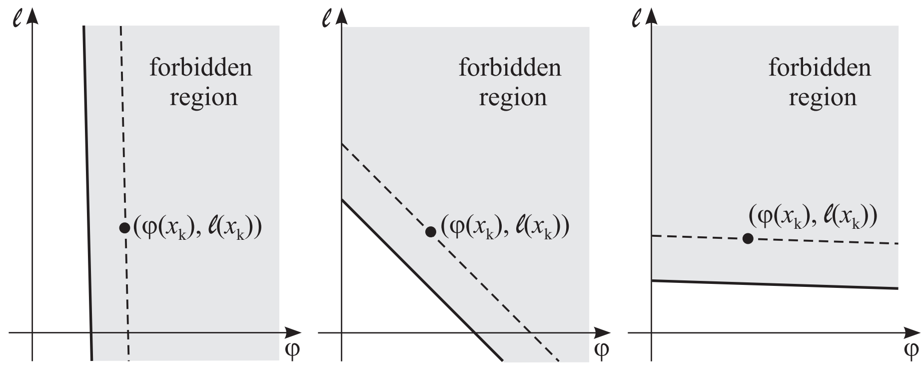

FIG. 1.1. *Three merit functions on the (φ, ℓ) plane, showing the influence of the penalty parameter θ. On the left, $\theta = 1/50$. In the middle, $\theta = 1/2$. On the right, $\theta = 49/50$.*

Now, let us analyse the role of the penalty parameter θ. Supposing, for example, (1.2) is used as the merit function, $(\theta - 1)/\theta$ can be viewed as the slope of the line that defines the *forbidden region* in the (φ, ℓ)-plane, that is, the semi-space that contains all the points that are not acceptable at the current iteration. This is illustrated in Figure 1.1, where the forbidden region defined by the augmented Lagrangian is highlighted for different values of θ.

In general, an algorithm starts with $\theta \approx 1$ and decreases this penalty parameter at some iterations, so feasibility is eventually attained.

Merit functions have been criticized for many reasons. First, it is not so easy to choose an initial value for θ, since $\ell(x, \lambda)$ and $\varphi(x)$ usually have very different meanings and units. Besides, it is necessary to decrease θ as the algorithm progresses to force it to find a feasible solution. If the initial penalty parameter used is near to 1 and θ is decreased slowly, the algorithm may take too many iterations to reach a feasible point. On the other hand, starting from a small θ or decreasing this factor too quickly may force iterates to stay almost feasible, shortening the steps even when we are far from the optimal solution.

As shown in [9], the adoption of a nonmonotone strategy for the reduction of θ is very effective to avoid this premature step shortening, but it also allows the algorithm to cicle between small and large penalty parameters, inducing some zigzaging in many cases.

To overcome these difficulties, Fletcher and Leyffer [7] introduced the idea of using a filter. This approach was promptly followed by may authors, mainly in conjunction with SLP (sequential linear programming), SQP and interior-point type methods (see, for instance, [1, 4, 5, 6, 8, 10, 11, 12, 13, 16, 17, 18]). In the SQP-filter method presented in [5], a point is accepted whenever it satisfies

$$\phi(x) < \gamma\phi_j \quad \text{or} \quad f(x) < f_j - \gamma\phi(x) \quad \text{for all} \quad (\phi_j, f_j) \in \mathcal{F}, \tag{1.6}$$

where

$$\phi(x) = \max\{0, \max_{i=1,\cdots,m} |C_i(x)|, \max_{i=1,\cdots,n} [x_i - u_i], \max_{i=1,\cdots,n} [l_i - x_i]\},$$

$\mathcal{F}$ is a set of previously generated points in the (ϕ, f)-space and $\gamma \in (0, 1)$ is a constant.

However, the efficiency of the SQP-filter method is also questionable, since it is too tolerant. In fact, requiring only the infeasibility *or* the optimality to be improved may allow the acceptance of points that are only marginally less infeasible but are much less optimal (in the sense that $f(x)$ is greater) than the current iterate, or vice-versa. Besides, one of the objectives of the filter is to avoid adding terms with different measures, but the last inequality of the acceptance criteria (1.6) has a merit function flavor.

Anyway, the SQP-filter method can give us some good hints on how to improve the algorithms based on merit functions.

The first hint is that the same merit function that is reliable for points in the (φ, f)-plane that are near to $(\varphi(x_k), f(x_k))$ may be not so useful when the step is large, so the trial point is far from the current iterate. As illustrated in Fig.1.1, for values of θ near to 1, the acceptance criteria based on a merit function cut off a significative portion of the feasible region, including, in many cases, the optimal solution of the problem.

The second good idea behind the SQP-filter method is that a restoration should be used sometimes. The objective of a restoration is to obtain a point that is less infeasible than the current one and is also acceptable for the filter. In [5, sec. 15.5], a restoration step is computed when the trust region quadratic subproblem is incompatible (i.e. has an empty feasible set), while the algorithm of Gonzaga et al. [10] computes a restoration at every step. We believe that this strategy can be used by an algorithm with a merit function always that staying away from feasibility seems not to be worth. Thus, if the decrease in f is small and the current point is very infeasible, it is better to move off and find a more feasible point.

The last lesson we can take from the SQP-filter method is that feasible points could never be refused by any merit function. This assures that the optimal solution will always be accepted by the algorithm and a restoration will always succeed.

Our objective here is to present an algorithm that takes advantages from both the merit function and the filter ideas.

This paper is organized as follows. In the next section, we present the piecewise linear function we use to accept or reject points. Section 3 introduces the proposed algorithm. In section 4, we prove that the algorithm is well defined. Sections 5 and 6 contain the main convergence results. Finally, in section 7 some conclusion are pressented, along with lines for future work.

Through the paper, we will omit some (or even all) of the arguments of a function, if this does not lead to confusion. Therefore, sometimes $Q(H, x, s)$ will be expressed as $Q(s)$, for example, if there is no ambiguity on H and x.

2. A piecewise linear merit function. As we have seen, a merit function deals with two different concepts: the infeasibility and the optimality of the current point.

In this paper, we will introduce a new merit function that compares points generated at different iterations. For this reason, this function cannot be based on the augmented Lagrangian, as in [9], since it depends on the Lagrange multiplier estimates used and, obviously, these estimates change from one iteration to another. Therefore, we decided to adopt the so called *smooth ℓ_2 merit function*, defined as:

$$\psi(x, \theta) = \theta f(x) + (1 - \theta)\varphi(x). \tag{2.1}$$

Unfortunately, it is well known that this function suffers from the Maratos effect and that an efficient implementation of the algorithm should include some safeguard for this undesired behaviour, such as a second order correction. However, as the circunvention of the Maratos effect is not required for the convergence analysis of our algorithm, we will not deal with this drawback of the merit function in this paper.

The actual reduction of the ℓ_2 merit function is given by

$$A_{red}(x,s,\theta) = \theta A_{red}^{opt}(x,s) + (1-\theta)A_{red}^{fsb}(x,s),$$

where

$$A_{red}^{opt}(x,s) = f(x) - f(x+s) \quad \text{and} \quad A_{red}^{fsb}(x,s) = \varphi(x) - \varphi(x+s).$$

Similarly, the predicted reduction of the merit function can be defined as in (1.3), replacing (1.5) by

$$P_{red}^{opt}(H,x,s) = Q(H,x,0) - Q(H,x,s),$$

where

$$Q(H,x,s) = \frac{1}{2}s^T Hs + \nabla f(x)^T s + f(x). \tag{2.2}$$

Generally, for a trial point to be accepted, it is necessary that the actual reduction of the merit function satisfies

$$A_{red}(x,s,\theta) \geq \eta P_{red}(H,x,s,\theta),$$

where $\eta \in (0,1)$ is a given parameter.

However, this scheme based on a linear merit function usually is unreliable for trial points that are far from the current iterate. Therefore, we suggest the use of a piecewise linear function to accept or reject new points.

In order to define this new merit function, let F be a set of p points (φ_i, f_i) in the (φ, f) plane. Suppose that these pairs are ordered so that $\varphi_1 < \varphi_2 < \cdots < \varphi_p$. Suppose also that each point (φ_i, f_i) in F is below the line segment joining (φ_{i-1}, f_{i-1}) and (φ_{i+1}, f_{i+1}), for $i = 2, \cdots, p-1$. Thus the piecewise linear function that passes through all of the points in F is convex.

For each point (φ_i, f_i) in F, define another point $(\overline{\varphi}_i, \overline{f}_i)$ by moving a little towards the southwest. Let $\overline{F}$ be the set of points $(\overline{\varphi}_i, \overline{f}_i)$. The convex piecewise linear function that connects the points in $\overline{F}$ is defined by

$$\mathcal{P}(\overline{F},\varphi) = \begin{cases} \infty, & \text{if } \varphi < \overline{\varphi}_1; \\ \frac{(\overline{f}_i - \overline{f}_{i-1})}{(\overline{\varphi}_i - \overline{\varphi}_{i-1})}\varphi + \frac{(\overline{f}_{i-1}\overline{\varphi}_i - \overline{f}_i\overline{\varphi}_{i-1})}{(\overline{\varphi}_i - \overline{\varphi}_{i-1})}, & \text{if } \overline{\varphi}_{i-1} \leq \varphi < \overline{\varphi}_i; \\ \overline{f}_p - \gamma(\varphi - \overline{\varphi}_p), & \text{if } \varphi \geq \overline{\varphi}_p. \end{cases}$$

where γ is a small positive constant, such as 10^{-4}.

This new function, illustrated in Fig. 2.1, is formed by $p+1$ line segments that can be viewed as merit functions in the form (2.1). The i-th of these functions is defined by the penalty parameter

$$\overline{\theta}_i = \begin{cases} 0, & \text{if } i = 0; \\ \frac{\overline{\varphi}_{i+1} - \overline{\varphi}_i}{\overline{f}_i - \overline{f}_{i+1} + \overline{\varphi}_{i+1} - \overline{\varphi}_i}, & \text{if } i < p; \\ 1/(1+\gamma), & \text{if } i = p. \end{cases} \tag{2.3}$$

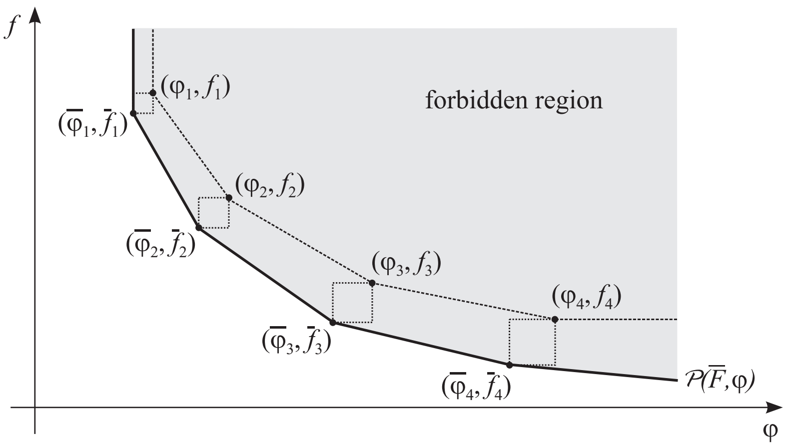

FIG. 2.1. *The set F and the piecewise linear function $\mathcal{P}(\overline{F}, \varphi)$.*

and a particular choice of η that will be defined below.

At each iteration k, $\overline{F}_k$ is generated defining, for each point $(\varphi_i, f_i) \in F_k$, another point $(\overline{\varphi}_i, \overline{f}_i)$ such that,

$$\overline{\varphi}_i = \min\{\varphi_i - \gamma_c P_{red}^{fsb}(x_k, s_c), (1 - \gamma_f)\varphi_i\}, \tag{2.4}$$

and

$$\overline{f}_i = min\{f_i - \gamma_f P_{red}^{opt}(H_k, x_k, s_c), f_i - (\varphi_i - \overline{\varphi}_i)\}, \tag{2.5}$$

for some $0 < \gamma_f < \gamma_c < 1$. Reasonable values for these constants are $\gamma_f = 10^{-4}$ and $\gamma_c = 10^{-3}$.

Our algorithm starts with $F_0 = \emptyset$. At the beginning of an iteration, say k, we define the temporary set $\mathcal{F}_k$ as

$$\mathcal{F}_k = F_k \bigcup \{(f(x_k), \varphi(x_k))\}.$$

A new iterate $x_+ = x_k + s_c$ is rejected if $f(x_k + s_c)$ is above the piecewise-linear function $\mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c))$ or if we predict a good reduction for the merit function, but the real reduction is deceiving. In the SQP jargon, x_k is not accepted if

$$A_{red}(x_k, s_c, \theta_k) \leq \overline{\eta} P_{red}(x_k, s_c, \theta_k), \tag{2.6}$$

or

$$P_{red}^{opt}(x_k, s_c) \geq \kappa\varphi(x_k) \quad \text{and} \quad A_{red}(x_k, s_c, \theta_k^{sup}) < \gamma_g P_{red}(x_k, s_c, \theta_k^{sup})$$

where $\gamma_g \in (0, 1)$,

$$\theta_k = \begin{cases} \overline{\theta}_0, & \text{if } \varphi(x_+) < \overline{\varphi}_1; \\ \overline{\theta}_i, & \text{if } \overline{\varphi}_i \leq \varphi(x_+) < \overline{\varphi}_{i+1}; \\ \overline{\theta}_p, & \text{if } \varphi(x_+) \geq \overline{\varphi}_p \end{cases} \tag{2.7}$$

and

$$\theta_k^{sup} = sup\{\theta \in [0, 1] \mid P_{red}(x_k, s_c, \theta) \geq 0.5[M(x_k, 0) - M(x_k, s_c)]\}. \tag{2.8}$$

When using this new scheme, the parameter $\overline{\eta}$ that defines the required ratio between A_{red} and P_{red} cannot be set by hand. In fact, it has a very complicated formula. In the case $P_{red}^{fsb} \geq \gamma_f \varphi_i/\gamma_c$ and $P_{red}^{opt} \geq (\varphi_i - \overline{\varphi}_i)/\gamma_f$, for example, this formula reduces to

$$\overline{\eta} = \frac{(\overline{f}_{i-1} - \overline{f}_i)\gamma_c P_{red}^{fsb} + (\overline{\varphi}_i - \overline{\varphi}_{i-1})\gamma_f P_{red}^{opt}}{(\overline{f}_{i-1} - \overline{f}_i) P_{red}^{fsb} + (\overline{\varphi}_i - \overline{\varphi}_{i-1}) P_{red}^{opt}}, \tag{2.9}$$

where i is defined in such a manner that $\overline{\varphi}_{i-1} \leq \varphi(x_+) \leq \overline{\varphi}_i$.

As it will become clear in the next section, depending on the behavior of the algorithm, the pair $(f(x_k), \varphi(x_k))$ may be permanently added to F_{k+1} at the end of the iteration. Thus, the cardinality of the set F_k is a nondecreasing function of k.

3. An SQP algorithm. In the general framework of a trust region sequential quadratic programming algorithm, a step s_c is obtained approximating problem (1.1), in a neighbourhood of an iterate x_k, by a quadratic programming (QP) problem.

In our case, this QP problem has the form

$$\textit{minimize} \quad Q(H_k, x_k, s) \tag{3.1a}$$
$$\textit{subject to} \quad A(x_k)s + C(x_k) = 0 \tag{3.1b}$$
$$l \leq x_k + s \leq u \tag{3.1c}$$
$$\|s\|_\infty \leq \Delta, \tag{3.1d}$$

where $Q(H, x, s)$ is defined by (2.2), x_k is supposed to belong to

$$\Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}$$

and H_k is an approximation of the Hessian of the Lagrangian at x_k. The infinity norm was chosen here so the constraints (3.1c) and (3.1d) can be grouped into one simple set of box constraints.

We will use the term *φ-stationary* to say that a point $\hat{x}$ satisfies the first order optimality conditions of

$$\begin{aligned} &\textit{minimize} \quad \varphi(x) \\ &\textit{subject to} \quad x \in \Omega. \end{aligned}$$

Unfortunately, if x_k is not φ-stationary, the constraints of (3.1) may be inconsistent, so this problem may not have a solution. A common practice to overcome this difficulty is to divide the step s_c into two components. The first of these components, called *normal step*, or simply s_n, is obtained as the solution of the feasibility problem

$$\begin{aligned} &\textit{reduce} \quad M(x_k, s) \\ &\textit{subject to} \quad l \leq x_k + s \leq u \\ &\qquad \|s\|_\infty \leq 0.8\Delta. \end{aligned} \tag{3.2}$$

If $M(x_k, s_n) = 0$, then x_k can be substituted by $x_k + s_n$ in (3.1) to make this problem feasible, so it can be solved by any QP algorithm. Otherwise, the second component of s_c, called the *tangential step*, or s_t, is computed so Q is reduced but the predicted reduction of the infeasibility obtained so far is retained. In other words, s_c

is the solution of the (now consistent) problem

$$
\begin{aligned}
(3.3) \qquad & \textit{reduce} && Q(H_k, x_k, s) \\
& \textit{subject to} && A(x_k)s = A(x_k)s_n \\
& && l \le x_k + s \le u \\
& && \|s\|_\infty \le \Delta.
\end{aligned}
$$

One should notice that the trust region radius was increased from (3.2) to (3.3). This is done to enlarge the feasible region when s_n is in the border of the trust region, so s_n is not the only solution of (3.3).

To assure a sufficient decrease of M, a Cauchy point, s_n^{dec}, is computed. This Cauchy point is based on a decent direction for $\varphi(x)$ given by $P_\omega(x_k - \nabla\varphi(x_k))$, the orthogonal projection of $x_k - \nabla\varphi(x_k)$ on Ω. The solution of (3.2) is required to keep at least ninety percent of the reduction obtained by s_n^{dec}.

A similar procedure is adopted for (3.3). In this case, s_t^{dec}, the Cauchy point, is obtained from a descent direction for $f(x)$ on the tangent space, given by $P_x(-\nabla Q(s_n))$, the orthogonal projection of $-\nabla Q(s_n)$ on the set

$$\mathcal{T} = \{y \in \mathcal{N}(A(x_k)) \mid (x_k + s_n + y) \in \Omega\}.$$

Again, the decrease on Q obtained by the solution of (3.3) must not be less than ninety percent of the reduction supplied by the Cauchy point.

The main steps of the algorithm are given below, supposing that an initial point $x_0 \in \Omega$, an initial trust-region radius $\Delta_0 \ge \Delta_{min}$ and an initial symmetric matrix H_0 are given.

We start from $k = 0$ and take $F_0 = \emptyset$ as the initial set of points used to define the piecewise linear function $\mathcal{P}(F)$.

ALGORITHM 3.1. *A new SQP algorithm*

```
1.       WHILE the stopping criteria are not satisfied
1.1.       F_k ← F_k ⋃{(f(x_k), φ(x_k))};
1.2.       IF ||C(x_k)|| = 0  (x_k is feasible),
1.2.1.       s_n ← 0;
1.3.       ELSE
1.3.1.       Compute d_n (a descent direction for φ(x)):
               d_n ← P_ω(x_k − γ_n ∇φ(x_k)) − x_k;
1.3.2.       Determine s_n^dec (the decrease step for φ(x)), the solution of
               minimize    M(x_k, s)
               subject to  l ≤ x_k + s ≤ u
                           ||s||_∞ ≤ 0.8Δ_k
                           s = t d_n,  t ≥ 0;
1.3.3.       Compute s_n (the normal step) such that
               l ≤ x_k + s_n ≤ u,
               ||s_n||_∞ ≤ 0.8Δ_k, and
               M(x_k, 0) − M(x_k, s_n) ≥ 0.9[M(x_k, 0) − M(x_k, s_n^dec)];
1.4.       Compute d_t (a descent direction for f(x) on the tangent space):
             d_t ← P_x(−γ_t ∇Q(s_n));
```

1.5. Determine s_t^{dec} (the decrease step for $f(x)$), the solution of

$$\begin{array}{ll} \textit{minimize} & Q(s) \\ \textit{subject to} & l \le x_k + s \le u \\ & \|s\|_\infty \le \Delta_k \\ & s = s_n + td_t,\ t \ge 0; \end{array}$$

1.6. Compute a trial step s_c such that
$A(x_k)s_c = A(x_k)s_n$,
$l \le x_k + s_c \le u$,
$\|s_c\|_\infty \le \Delta_k$, and
$Q(s_n) - Q(s_c) \ge 0.9[Q(s_n) - Q(s_t^{dec})]$;

1.7. IF $(f(x_k + s_c) \ge \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c)))$ OR
$(P_{red}^{opt} \ge \kappa\varphi(x_k)$ AND $A_{red}(x_k, s_c, \theta_k^{sup}) < \gamma_g P_{red}(H_k, x_k, s_c, \theta_k^{sup}))$,

1.7.1. $\Delta_k \leftarrow \alpha_R \min\{\Delta_k, \|s_c\|_\infty\}$; (reduce Δ)

1.8. ELSE

1.8.1. $\rho_k \leftarrow A_{red}^{opt}(x_k, s_c)/P_{red}^{opt}(H_k, x_k, s_c)$;

1.8.2. IF $P_{red}^{opt}(H_k, x_k, s_c) < \kappa\varphi(x_k)$ OR $\rho_k < \gamma_f$,

1.8.2.1. $F_{k+1} \leftarrow \mathcal{F}_k$; (include $(f(x_k), \varphi(x_k))$ in F)

1.8.3. ELSE $F_{k+1} \leftarrow F_k$;

1.8.4. Accept the trial point:
$x_{k+1} \leftarrow x_k + s_c$;

$$\Delta_{k+1} \leftarrow \begin{cases} \max\{\alpha_R \min\{\Delta_k, \|s_c\|_\infty\}, \Delta_{min}\}, & \text{if } \rho_k < \gamma_g, \\ \max\{\alpha_A \Delta_k, \Delta_{min}\}, & \text{if } \rho_k \ge \eta; \end{cases}$$

Determine H_{k+1};
$k \leftarrow k + 1$;

1.9. IF $\Delta_k < \Delta_{rest}$ AND $\varphi(x_k) > \epsilon_h \Delta_k^2$,

1.9.1. Compute a restoration step s_r so that
$(\varphi(x_k + s_r) < \epsilon_h \Delta_k^2$ AND $f(x_k + s_r) < \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_r)))$ OR
$x_k + s_r$ is φ-stationary but infeasible;

1.9.2. $F_{k+1} \leftarrow \mathcal{F}_k$; (include $(f(x_k), \varphi(x_k))$ in F)

1.9.3. Accept the new point:
$x_{k+1} \leftarrow x_k + s_r$;
$\Delta_{k+1} \leftarrow \max\{\beta\Delta_{rest}, \Delta_{min}\}$;
Determine H_{k+1};
$k \leftarrow k + 1$;

The constants used here must satisfy $\kappa > 0$, $0 < \gamma_f < \gamma_g < \eta < 1$, $\gamma_n > 0$, $\gamma_t > 0$, $\Delta_{min} > 0$, $0 < \alpha_R < 1$, $\alpha_A \ge 1$, $\epsilon_h > 0$ and $\beta > 0$. Parameters γ_n, γ_t, Δ_{min}, ϵ_h and β are problem dependent and may be chosen according with some measure of problem data. Reasonable values for the remaining parameters might be $\gamma_f = 0.01$, $\gamma_g = 0.1$, $\eta = 0.5$, $\alpha_R = 0.5$ and $\alpha_A = 2.0$. The constant η should not be confused with the parameter $\overline{\eta}$ defined in (2.9).

If x_k is feasible, then the condition $P_{red}^{opt} < \kappa\varphi(x_k)$ is never satisfied, since P_{red}^{opt} is always greater or equal to zero. Besides, the condition $A_{red}^{opt} < \gamma_f P_{red}^{opt}$ is also never satisfied when x_k is feasible and $f(x_k + s_c) < \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c))$. Therefore, all of the points in F_k are infeasible, although $\mathcal{F}_k$ may contain a feasible point. This result is very important for two reasons. First, it prevents the optimal solution of problem 1.1 to be refused by the algorithm. Moreover, it assures the algorithm is well defined, as stated in the next section.

4. The algorithm is well defined. An iteration of algorithm 3.1 ends only when a new point $x_k + s$ is below the piecewise linear function $\mathcal{P}(\mathcal{F}_k, \varphi(x_k + s))$, besides satisfying some other conditions stated at steps 1.7 or 1.9.1. While such a point is not found, the trust region radius is reduced and the iteration is repeated. It is not obvious that an acceptable point will be obtained, as we may generate a sequence of points that are always rejected by the algorithm. In this section, we prove that the algorithm is well defined, i.e. a new iterate x_{k+1} can always be obtained unless the algorithm stops by finding a φ-stationary but infeasible point or a feasible but not regular point.

In the following lemma, we consider the case where x_k is infeasible.

LEMMA 4.1. *If x_k is not φ-stationary, then after a finite number of repetitions of steps 1.1 to 1.9, a new iterate x_{k+1} is obtained by the algorithm.*

Proof. At each iteration k, if $f(x_k + s_c) < \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c))$ and one of the conditions $P_{red}^{opt} \leq \kappa\varphi(x_k)$ or $A_{red}(x_k, s_c, \theta_k^{sup}) \geq \gamma_g P_{red}(H_k, x_k, s_c, \theta_k^{sup})$ is satisfied, then $x_k + s_c$ is accepted and we move to iteration $k + 1$. Otherwise, Δ_k is reduced and after some unfruitful steps, $\Delta_k < \Delta_{rest}$ and $\varphi(x_k) > \epsilon_h \Delta_k^2$, so a restoration is called.

Suppose that a φ-stationary but infeasible point is never reached (otherwise the algorithm fails). As the restoration generates a sequence of steps $\{s_j\}$ converging to feasibility, and since $\mathcal{F}_k$ does not include feasible points (because x_k is infeasible and no feasible point is included in F_k), there must exist an iterate $x_k + s_r$ that satisfies $\varphi(x_k + s_r) < \min\{\overline{\varphi}_1, \epsilon_h \Delta_k^2\}$, so we can proceed to the next iteration. $\square$

Now, in order to prove that the algorithm is also well defined when x_k is feasible, we need to make the following assumptions.

A1. $f(x)$ and $C_i(x)$ are twice-continuously differentiable functions of x.

A2. The sequence of Hessian approximations $\{H_k\}$ is bounded.

As a consequence of A1 and A2, the difference between the actual and the predicted reduction of the merit function is proportional to Δ^2, so the step is accepted for a sufficiently small trust region radius, as stated in the following lemma.

LEMMA 4.2. *Suppose that A1 and A2 hold and that x_k is feasible and regular for problem 1.1 but the KKT conditions do not hold. Then, after a finite number of trust region reductions, the algorithm finds a new point $x_k + s_c$ that satisfies $f(x_k + s_c) < \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c))$ and $A_{red}(x_k, s_c, \theta_k^{sup}) \geq \gamma_g P_{red}(H_k, x_k, s_c, \theta_k^{sup})$.*

Proof. Since x_k is feasible, $s_n = 0$. Supposing that x_k is regular and non-stationary, there must exist a vector $d_t \neq 0$ satisfying

$$l \leq x_k + d_t \leq u, \quad A(x_k)d_t = 0, \quad \textit{and} \quad d_t^T \nabla f(x_k) < 0.$$

Let us define, for all $\Delta > 0$,

$$p(\Delta) = t(\Delta)d_t,$$

where

$$t(\Delta) = max\{t > 0 \mid [x_k, x_k + td_t] \subset \Omega, \quad \text{and} \quad \|td_t\|_\infty \leq \Delta\}.$$

Clearly, $x + d_t \in \Omega$, so we have that $\|t(\Delta)d_t\|_\infty = \Delta$ whenever $\Delta \leq \|d_t\|_\infty$. Define, in this case,

$$c = -\frac{1}{2}d_t^T \nabla f(x_k)/\|d_t\|_\infty = -\frac{1}{2}d_t^T \nabla Q(0)/\|d_t\|_\infty > 0.$$

Since $Q(s_t^{dec}) \le Q(p(\Delta))$, by elementary properties of one-dimensional quadratics, there exists $\Delta_1 \in (0, \|d_t\|_\infty]$ such that, for all $\Delta \in (0, \Delta_1)$,

$$Q(0) - Q(s_t^{dec}) \ge -\frac{1}{2} d_t^T \nabla Q(0) t(\Delta) = -\frac{1}{2} \frac{d_t^T \nabla Q(0)}{\|d_t\|_\infty} \Delta = c\Delta.$$

Moreover, since x_k is feasible and $As_n = 0$, we have that $M(x_k, 0) = M(x_k, s_c) = 0$, so

$$P_{red}^{fsb}(x_k, s_c) = 0, \quad \text{and}$$

$$P_{red}^{opt}(H_k, x_k, s_c(\Delta)) = Q(0) - Q(s_c(\Delta)) \ge 0.9[Q(0) - Q(s_t^{dec})] \ge 0.9c\Delta.$$

Once x_k is feasible, $(\varphi(x_k), f(x_k))$ is the first pair in $\mathcal{F}_k$. Thus, there exists $\overline{\Delta}_2 \in (0, \Delta_1]$ such that, for $\Delta < \overline{\Delta}_2$, we need to consider only the portion of $\mathcal{P}(\mathcal{F}_k, \varphi)$ defined on the interval $[0, \varphi_2]$. This linear function may be rewritten so the condition

$$f(x_k + s_c) < \mathcal{P}(\mathcal{F}_k, \varphi(x_k + s_c))$$

is equivalent to

$$A_{red}(x_k, s_c(\Delta), \overline{\theta}_1) \ge \eta_1 P_{red}(H_k, x_k, s_c(\Delta), \overline{\theta}_1), \tag{4.1}$$

where,

$$P_{red}(H_k, x_k, s_c(\Delta), \overline{\theta}_1) = \overline{\theta}_1 P_{red}^{opt}(H_k, x_k, s_c(\Delta)) \ge 0.9c\Delta\overline{\theta}_1, \tag{4.2}$$

$$A_{red}(x_k, s_c(\Delta), \overline{\theta}_1) = \overline{\theta}_1[f(x_k) - f(x_k + s_c(\Delta))] + (1 - \overline{\theta}_1)\varphi(x_k + s_c(\Delta))$$

and $\overline{\theta}_1 > 0$ is given by (2.3).

Now, by A1, A2 and the definition of P_{red}, we have

$$A_{red}(x_k, s_c, \theta) = P_{red}(H_k, s_k, s_c, \theta) + \overline{c}_1 \|s_c\|^2. \tag{4.3}$$

So, using (4.2) and (4.3) we deduce that

$$\left| \frac{A_{red}(\Delta)}{P_{red}(\Delta)} - 1 \right| \le \frac{\overline{c}_1 \Delta}{0.9c\overline{\theta}_1}. \tag{4.4}$$

Thus, for $\Delta < \min\{(1 - \eta_1)0.9c\overline{\theta}_1/\overline{c}_1, \overline{\Delta}_2\} = \overline{\Delta}_3$, the inequality (4.1) necessarily takes place.

Now, using the fact that $\theta_k^{sup} = 1$ for x_k feasible and replacing $\overline{\theta}_1$ by 1 in (4.4), we can conclude that, for

$$\Delta < \min\{(1 - \gamma_g)0.9c/\overline{c}_1, \overline{\Delta}_3\} = \overline{\Delta}_4, \tag{4.5}$$

the condition $A_{red}(x_k, s_c(\Delta), \theta_k^{sup}) \ge \gamma_g P_{red}(H_k, x_k, s_c(\Delta), \theta_k^{sup})$ is also satisfied and the step is accepted. $\square$

5. The algorithm converges to a feasible point. As mentioned in the last section, our algorithm can stop if a φ-stationary but infeasible point is found. Moreover, the restoration procedure can also fail to obtain a more feasible point. Naturally, this unexpected behavior of the algorithm makes somewhat pretentious the title of this section.

Formally, what we will prove in this section is that, supposing that a φ-stationary but infeasible point is never reached and that the restoration always succeeds, an infinite sequence of iterates converges to feasibility.

In the proofs of the lemmas presented here, we will suppose that A1 and the following assumption are satisfied.

A3. The sequence of iterates $\{x_k\}$ lies within a closed and bounded domain Ω_0.

As mentioned in [5, p.730], assumptions A1 and A3 together ensure that, for all k,

$$f^{min} \le f(x_k) \le f^{max} \quad \text{and} \quad 0 \le \varphi(x_k) \le \varphi^{max}$$

for some constants f^{min}, f^{max} and $\varphi^{max} > 0$. Our analysis will be based on the fact that the rectangle $[0, \varphi^{max}] \times [f^{min}, f^{max}]$ is covered by a finite number of rectangles with area greater than a small constant. Therefore, each time we expand the forbidden region (see fig (2.1)) by adding to it a small rectangle, we drive the iterates towards feasibility.

Let us start investigating what happens to $\varphi(x)$ when an infinite sequence of iterates is added to F.

LEMMA 5.1. *Suppose that A1 and A3 hold and that $\{k_i\}$ is any infinite subsequence at which the iterate x_{k_i} is added to F. Then*

$$\lim_{i\to\infty} \varphi(x_{k_i}) = 0.$$

Proof. Let us suppose, for the purpose of obtaining a contradiction, that there exists an infinite subsequence $\{k_j\} \subseteq \{k_i\}$ for which

$$\varphi(x_{k_j}) \ge \epsilon, \tag{5.1}$$

where $\epsilon > 0$.

At iteration k_j, the (φ, f)-pair associate with x_{k_j} is included in F at position m, which means that $\varphi_{m-1} \le \varphi_{k_j} (\equiv \varphi_m) \le \varphi_{m+1}$ and $f_{m-1} \ge f_{k_j} (\equiv f_m) \ge f_{m+1}$. Thus, as long as the pair (φ_{k_j}, f_{k_j}) remains in F, no other (φ, f)-pair is accepted within the rectangle

$$r_m = \{(\varphi, f) \mid \overline{\varphi}_m \le \varphi \le \varphi_m, \;\; \overline{f}_m \le f \le f_m\}.$$

Notice that, by (2.4) and (2.5), the area of this rectangle is

$$(\varphi_m - \overline{\varphi}_m)(f_m - \overline{f}_m) \ge (\varphi_m - \overline{\varphi}_m)^2 \ge [(1-\gamma_f)\varphi_{k_j}]^2 \ge (1-\gamma_f)^2 \epsilon^2.$$

Assume now that (φ_{k_j}, f_{k_j}) is excluded from F by another pair (φ_{k_l}, f_{k_l}), included in F at an iteration $k_l > k_j$. This case is illustrated in Fig. 5.1. Notice that (φ_{k_l}, f_{k_l}) cannot fall in regions I and V since, in this case, (φ_{k_j}, f_{k_j}) will not be excluded from F. It can be easily verified that the worst case occurs when (φ_{k_l}, f_{k_l}) lies on $\ell_1(\varphi)$ or $\ell_2(\varphi)$.

Suppose (φ_{k_l}, f_{k_l}) lies on $\ell_2(\varphi)$, as depicted in Fig. 5.1. In this case, the rectangle r_m will be entirely above $\overline{\ell}_2$, the line that connects $(\overline{\varphi}_{k_l}, \overline{f}_{k_l})$ to $(\overline{\varphi}_{m+1}, \overline{f}_{m+1})$. Since $\overline{\ell}_2$ will be included in the new piecewise linear function $\mathcal{P}(\mathcal{F})$, no point within r_m can ever be reached by a new iterate.

The same idea can be applied in the case (φ_{k_l}, f_{k_l}) lies on $\ell_1(\varphi)$. Therefore, once (φ_{k_j}, f_{k_j}) is included in F, r_m will always be above $\mathcal{P}(\mathcal{F})$. Since the area of this rectangle is at least $(1-\gamma_f)^2\epsilon^2$ and the set A_0 is completely covered by at most $\text{Surf}(A_0)/[(1-\gamma_f)^2\epsilon^2]$ of such rectangles, it is impossible for an infinite subsequence of $\{k_i\}$ to satisfy (5.1), and the conclusion follows. ◻

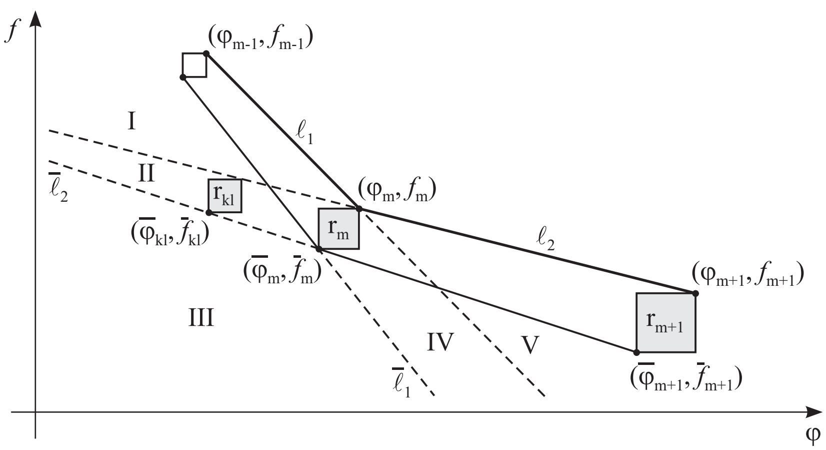

FIG. 5.1. *Adding a new iterate that excludes (φ_{k_j}, f_{k_j}) from F.*

Finally, we are going to consider the case where no point is added to F_k for k sufficiently large.

LEMMA 5.2. *Suppose that assumptions A1 and A3 hold. Suppose also that, for all $k > k_0$, x_k is never included in F_k. Then,*

$$\lim_{k\to\infty} \varphi(x_k) = 0. \tag{5.2}$$

Proof. Since x_k is not included in F_k, no restorations are made and both conditions stated at step 1.8.2 of algorithm 3.1 are never satisfied for $k > k_0$. Therefore, we have

$$f(x_k) - f(x_{k+1}) \ge \gamma_f P_{red}^{opt} \ge \gamma_f \kappa \varphi(x_k) \ge 0, \tag{5.3}$$

for all $k > k_0$, which means that the objective function always decrease between infeasible iterations. Since A1 and A3 imply $f^{min} \le f(x_k) \le f^{max}$, we must have

$$\lim_{k\to\infty} f(x_k) - f(x_{k+1}) = 0. \tag{5.4}$$

Then, (5.2) follows from (5.3) and (5.4). ◻

6. The algorithm finds a critical point. Finally, we are able to prove the convergence of the algorithm to a stationary point for (1.1). In order to do that, we will need to make one aditional assumption on the choice of the normal step s_n.

A4. The choice of s_n at step 1.3.3 of algorithm 3.1 is such that

$$\|s_n(x_k, \Delta_k)\| \le \kappa_n \|C(x_k)\|_2.$$

In the following lemma, derived from lemma 6.1 of [9], we show that in the neighborhood of a feasible, regular and non-stationary point, the directional derivative of the quadratic model (2.2) along d_t is bounded away from zero.

LEMMA 6.1. *Suppose that A2 and A4 hold and that $\{x_{k_i}\}$ is an infinite subsequence that converges to the feasible and regular point $x^* \in \Omega$, which is not stationary for (1.1). Then, there exists $k_1, c_1 > 0$ such that*

$$-\nabla Q(s_n(x,\Delta))^T d_t(H,x,\Delta) \ge c_1 \tag{6.1}$$

for all $x \in \{x_{k_i} \mid k \ge k_1\}$. Moreover, $\|d_t(H,x,\Delta)\|$ is bounded and bounded away from 0 for all $x \in \{x_{k_i} \mid k \ge k_1\}$.

Proof. For all $x \in \{x_{k_i}\}$, we have that

$$d_t(H,x,\Delta) = P_x(-\gamma_t \nabla Q(s_n(x,\Delta))) = P_x(-\gamma_t[Hs_n(x,\Delta) + \nabla f(x)]).$$

By the contractive property of the orthogonal projections,

$$\|P_x(-\gamma_t[Hs_n(x,\Delta) + \nabla f(x)]) - P_x(-\gamma_t \nabla f(x))\|_2 \le \gamma_t \|H\|_2 \|s_n(x,\Delta)\|_2.$$

So, by A2 and A4, we have that

$$\|d_t(H,x,\Delta) - P_x(-\gamma_t \nabla f(x))\|_2 \le \gamma_1 \|C(x)\| \tag{6.2}$$

and, by the continuity of $\nabla f(x)$ and the fact that $\{x_{k_i}\}$ converges, we deduce that

$$\|\nabla f(x_{k_i})^T P_x(-\gamma_t \nabla f(x_{k_i})) - \nabla f(x_{k_i})^T d_t(H_{k_i}, x_{k_i}, \Delta_{k_i})\|_2 \le \gamma_2 \|C(x_{k_i})\|. \tag{6.3}$$

Notice that $P_x(-\gamma_t \nabla f(x_{k_i}))$ is the solution of

$$\begin{array}{ll} \textit{minimize} & \|-\gamma_t \nabla f(x_{k_i}) - z\|_2^2 \\ \textit{subject to} & A(x_{k_i})z = 0 \\ & l \le x_{k_i} + s_n + z \le u. \end{array}$$

Now, define $P_{x^*}(-\gamma_t \nabla f(x^*))$ as the solution of

$$\begin{array}{ll} \textit{minimize} & \|-\gamma_t \nabla f(x^*) - z\|_2^2 \\ \textit{subject to} & A(x^*)z = 0 \\ & l \le x^* + z \le u. \end{array} \tag{6.4}$$

Since x^* is regular but is not a stationary point for (1.1), it follows that $z = 0$ is not a solution for (6.4). So, $P_{x^*}(-\gamma_t \nabla f(x^*)) \ne 0$. Moreover, since $z = 0$ is feasible for (6.4), we have that

$$\|-\gamma_t \nabla f(x^*) - P_{x^*}(-\gamma_t \nabla f(x^*))\|_2^2 < \|-\gamma_t \nabla f(x^*)\|_2^2,$$

which implies that $\nabla f(x^*)^T P_{x^*}(-\gamma_t \nabla f(x^*)) < 0$.

Using the fact that $P_x(-\gamma_t \nabla f(x))$ is a continuous function of x and s_n for all regular x (see [9]), we can define $\overline{c}_2, \overline{c}_3, \overline{c}_4 > 0$ and $\overline{k}_2 \in \mathbb{N}$ such that, for all $x \in \{x_{k_i} \mid k \ge \overline{k}_2\}$, we have

$$\overline{c}_2 \le \|P_x(-\gamma_t \nabla f(x))\| \le \overline{c}_3 \quad \text{and} \quad \nabla f(x)^T P_x(-\gamma_t \nabla f(x)) \le -\overline{c}_4. \tag{6.5}$$

Now, from (6.2), (6.3) and (6.5), the continuity of $C(x)$ and the feasibility of x^*, there exists $\overline{k}_3 \geq \overline{k}_2$ such that, whenever $x \in \{x_{k_i} \mid k \geq \overline{k}_3\}$,

$$\frac{\overline{c}_2}{2} \leq \|d_t(H, x, \Delta)\| \leq 2\overline{c}_3 \quad \text{and} \quad \nabla f(x)^T d_t(H, x, \Delta) \leq \frac{-\overline{c}_4}{2}.$$

Therefore, $\|d_t(H, x, \Delta)\|$ is bounded and bounded away from zero for all $x \in \{x_{k_i} \mid k \geq \overline{k}_3\}$.

Finally, since $d_t \in \mathcal{N}(A(x))$, assumptions A2 and A4 hold, and $\|d_t\|$ is bounded, we have that, for all $x \in \{x_{k_i} \mid k \geq \overline{k}_3\}$,

$$\nabla Q(s_n)^T d_t = \nabla f(x)^T d_t + d_t^T H s_n \leq -\frac{\overline{c}_4}{2} + \gamma_3 \|C(x)\|,$$

where $\gamma_3 > 0$. Then, (6.1) follows defining $c_1 = \overline{c}_4/4$ and choosing $k_1 > \overline{k}_3$ such that $\|C(x)\| \leq \overline{c}_4/(4\gamma_3)$. ☐

Using Lemma 6.1, we prove in the next lemma that, in the neighborhood of a feasible, regular and non-stationary point, the decrease of the quadratic model (2.2) is proportional to the trust region radius Δ.

LEMMA 6.2. *Suppose that A2 and A4 hold and that $\{x_{k_i}\}$ is an infinite subsequence that converges to the feasible and regular point $x^* \in \Omega$, which is not stationary for (1.1). Then, there exists $c_2, k_2 > 0$ and $\Delta_0 \in (0, \Delta_{min})$ such that*

$$Q(x, s_n(x, \Delta)) - Q(x, s_c)) \geq c_2 \min\{\Delta, \Delta_0\}$$

for all $x \in \{x_{k_i} \mid k \geq k_2\}$.

Proof. See Lemma 6.2 of [9]. ☐

Now, we are able to present a crucial lemma, derived from Lemma 6.3 of [9], that relates P_{red}^{opt} to the trust region radius in the neighborhood of a feasible point. Besides, we also show that, in this case, P_{red}^{opt} is sufficiently large so $\theta^{sup} = 1$.

LEMMA 6.3. *Suppose that A1, A2 and A4 hold and that $\{x_{k_i}\}$ is an infinite subsequence that converges to the feasible and regular point $x^* \in \Omega$, which is not stationary for (1.1). Then, there exists $\epsilon, c_3, k_3 > 0$ and $\Delta_1 \in (0, \Delta_{min})$ such that, for $k_i > k_3$, if*

$$\varphi(x_{k_i}) \leq \epsilon \Delta^2, \tag{6.6}$$

we have that

$$P_{red}^{opt}(x_{k_i}, s_c) = Q(x_{k_i}, 0) - Q(x_{k_i}, s_c) \geq c_3 \min\{\Delta, \Delta_1\} \tag{6.7}$$

and

$$\theta_{k_i}^{sup} = 1. \tag{6.8}$$

Proof. By Lemma 6.2, assumptions A1 and A4 and the convergence of $\{x_{k_i}\}$, we have that

$$Q(0) - Q(s_c) \geq Q(s_n) - Q(s_c) - |Q(0) - Q(s_n)| \geq c_2 \min\{\Delta, \Delta_1\} - \gamma_4 \|C(x)\|$$

for all $x \in \{x_{k_i} \mid k \geq k_2\}$, where c_2, k_2 and Δ_1 are defined as in Lemma 6.2 and $\gamma_4 > 0$. Therefore, (6.7) follows if we choose $c_3 < c_2$ and $k_3 \geq k_2$ such that $\epsilon \leq \alpha^2(c_2 - c_3)^2/(2\gamma_4^2)$, where $\alpha = \min\{1, \Delta_1/\Delta\}$.

Now, from assumption A4, we have that

$$M(0) - M(s_c) = M(0) - M(s_n) \leq \gamma_5 \|C(x_{k_i})\|,$$

so

$$P_{red}(x_{k_i}, s_c, 1) - 0.5[M(x_{k_i}, 0) - M(x_{k_i}, s_c)] \geq c_3 \min\{\Delta, \Delta_1\} - \gamma_5 \|C(x_{k_i})\|.$$

Again, (6.8) follows if we choose $\epsilon \leq \alpha^2 c_3^2/(2\gamma_5^2)$. ☐

We next examine what happens if Δ is bounded away from zero and an infinite subsequence of points is added to F.

LEMMA 6.4. *Suppose that A1, A2, A3 and A4 hold and that $\{x_{k_j}\}$ is an infinite subsequence at which x_{k_j} is added to F. Suppose furthermore that the restoration always terminates successfully and that $\Delta_{k_i} \geq \Delta_2$, where Δ_2 is a positive scalar. Then there exists a limit point of this sequence that is a stationary point for (1.1).*

Proof. From assumption A3, we know that there exists a convergent subsequence $\{x_{k_i}\}$. Let us suppose that the limit point of this subsequence is not stationary for (1.1).

From Lemma 5.1 we know that there exists $\overline{k}_5 \in I\!N$ such that, for $k_i > \overline{k}_5$,

$$\varphi(x_{k_i}) < \epsilon_h \Delta_2^2.$$

Thus, a restoration is never called for $k_i > \overline{k}_5$. So, the hypotesis that x_{k_i} is added to F_{k_i} implies that one of the inequalities stated at step 1.8.2 of the algorithm must be satisfied at iteration k_i.

Suppose, for the purpose of obtaining a contradiction, that $\{x_{k_i}\}$ converges to a point that is not stationary for (1.1). So, from Lemma 5.1 and (6.7), there exists $\overline{k}_6 \geq \overline{k}_5$ such that $\varphi(x_{k_i}) < \epsilon \Delta_{k_i}^2$ and

$$P_{red}^{opt}(x_{k_i}, s_c) \geq c_3 \min\{\Delta_1, \Delta_2\},$$

for all $k_i > \overline{k}_6$.

Using Lemma 5.1 again, we can deduce that there exists $\overline{k}_7 \geq \overline{k}_6$ such that $\varphi(x_{k_i}) < (c_3/\kappa) \min\{\Delta_1, \Delta_2\}$ and the condition $P_{red}^{opt} < \kappa\varphi(x_k)$ is never satisfied for $k_i > \overline{k}_7$.

Therefore, $f(x_{k_i}) - f(x_{k_i} + s_c) < \gamma_f P_{red}^{opt}$ must hold. To show that this is not possible, let us write the inequality $A_{red}(x_{k_i}, s_c, \theta_{k_i}^{sup}) \geq \gamma_f P_{red}(x_{k_i}, s_c, \theta_{k_i}^{sup})$ as

$$\begin{gathered}\theta_{k_i}^{sup}(f(x_{k_i}) - f(x_{k_i} + s_c)) + (1 - \theta_{k_i}^{sup})(\varphi(x_{k_i}) - \varphi(x_{k_i} + s_c)) \geq \\ \gamma_g \theta_{k_i}^{sup} P_{red}^{opt}(x_{k_i}, s_c) + \gamma_g (1 - \theta_{k_i}^{sup}) P_{red}^{fsb}(x_{k_i}, s_c).\end{gathered}$$

Using the hypothesis that $f(x_{k_i}) - f(x_{k_i} + s_c) < \gamma_f P_{red}^{opt}(x_{k_i}, s_c)$ and the fact that $P_{red}^{fsb}(x_{k_i}, s_c) \geq 0$, we have

$$\theta_{k_i}^{sup} \gamma_f P_{red}^{opt}(x_{k_i}, s_c) + (1 - \theta_{k_i}^{sup})(\varphi(x_k) - \varphi(x_k + s_c)) \geq \gamma_g \theta_{k_i}^{sup} P_{red}^{opt}(x_{k_i}, s_c).$$

Then, taking $k_4 > k_3$ (defined in Lemma 6.3), we deduce from (6.7) that, for $k_i > k_4$,

$$(1 - \theta_{k_i}^{sup})(\varphi(x_{k_i}) - \varphi(x_{k_i} + s_c)) \geq (\gamma_g - \gamma_f) \theta_{k_i}^{sup} c_3 \min\{\Delta_1, \Delta_2\}.$$

But, since, $\gamma_g > \gamma_f$ and $\lim_{i \to \infty} \varphi_{k_i} = 0$, we must have

$$\lim_{i \to \infty} \theta_{k_i}^{sup} = 0,$$

which contradicts (6.8). Therefore, $\{x_{k_i}\}$ must converge to a stationary point for (1.1). ∎

Supposing again that Δ is bounded away from zero, we will now complete our analysis investigating what happens when no iterates are added to F for k sufficiently large.

LEMMA 6.5. *Suppose that A1, A2, A3 and A4 hold, that x_k is always accepted but F_k remains unchanged for $k > k_5$ and that $\Delta_k \geq \Delta_3$, for some positive Δ_3. Suppose also that the limit points of the infinite sequence $\{x_k\}$ are feasible and regular. Then there exists a limit point of $\{x_k\}$ that is a stationary point of (1.1).*

Proof. Assumption A3 implies that there exists a convergent subsequence $\{x_{k_i}\}$. If the limit point of this subsequence is not stationary for (1.1), then from Lemma 6.3, we have

$$P_{red}^{opt} \geq c_3 \min\{\Delta_3, \Delta_1\}$$

for all $k_i > \max\{k_5, k_3\}$. Moreover, since x_{k_i} is always accepted and F_k is not changed, we deduce that

$$f(x_{k_i}) - f(x_{k_i} + s_c) \geq \gamma_f P_{red}^{opt}.$$

Therefore, $f(x_{k_i}) - f(x_{k_i} + s_c) \geq \gamma_f c_3 \min\{\Delta_1, \Delta_3\}$ for all k_i sufficiently large, which contradicts the compactness assumption A3. ∎

In the last part of this section, we will discuss the behavior of the algorithm when $\Delta \to 0$. We will start showing that the predicted reduction of the quadratic model is sufficiently large when Δ is small.

LEMMA 6.6. *Suppose that A2 and A4 hold and that $\{x_{k_i}\}$ is an infinite subsequence that converges to the feasible and regular point $x^* \in \Omega$, which is not stationary for (1.1). Suppose also that φ_k satisfies (6.6) and that*

$$\Delta < \min\{c_3/(\kappa\epsilon), \Delta_1\} = \Delta_5 \tag{6.9}$$

for $k_i > k_7$, where c_3, ϵ and Δ_1 are defined as in Lemma 6.3. Then $P_{red}^{opt} > \kappa\varphi(x_{k_i})$.

Proof. Suppose, for the purpose of obtaining a contradiction, that $P_{red}^{opt} \leq \kappa\varphi(x_k)$ for some $k_i > k_7$. Then, from (6.7), we have

$$c_3 \min\{\Delta, \Delta_1\} \leq P_{red}^{opt} \leq \kappa\varphi(x_{k_i}) \leq \kappa\epsilon\Delta^2,$$

which is impossible because of (6.9). Thus $P_{red}^{opt} > \kappa\varphi(x_{k_i})$ must hold. ∎

The purpose of the next four lemmas is to prove that there exists a sufficiently small trust region radius so the step is always accepted and Δ is not reduced further at step 1.7.1 of algorithm 3.1.

The first lemma will be used to show the relation between the predicted reduction of the infeasibility and Δ.

LEMMA 6.7. *Suppose that assumption A1 holds and that x_k is not φ-stationary. Then, there exists $\Delta_6, c_4 > 0$ such that*

$$P_{red}^{fsb}(x_k, s_c) \geq c_4\Delta_k, \tag{6.10}$$

if $\Delta_k \in (0, \Delta_6)$.

Proof. Since x_k is not φ-stationary, we have that $d_n \neq 0$. Thus, we can define

$$t^{max}(\Delta) = \max\{t > 0 \mid [x_k, x_k + td_n] \in \Omega \ \text{ and } \ \|td_n\| \leq 0.8\Delta\}.$$

Clearly, if $\Delta \leq 1.25\|d_n\|$, then $\|t^{max}(\Delta)d_n\| = 0.8\Delta$ (since $x_k + d_n \in \Omega$). Now, define

$$c = -\frac{1}{2}\frac{d_n^T \nabla\varphi(x_k)}{\|d_n\|} > 0.$$

By some elementary properties of one-dimensional quadratics, there exists $\Delta_6 \in (0, \|d_n\|]$ such that

$$M(0) - M(t^{max}(\Delta)) \geq -\frac{1}{2}d_n^T \nabla(x_k)t^{max}(\Delta) = c\|d_n\|t^{max}(\Delta) = 0.8c\Delta.$$

for all $\Delta \in (0, \Delta_6)$. Therefore, for the normal step s_n computed at step 1.3.3 of algorithm 3.1, we have

$$M(0) - M(s_n) \geq 0.72c\Delta_k.$$

But, since $A(x_k)s_c = A(x_k)s_n$, we deduce from (1.4) that

$$P_{red}^{fsb}(x_k, s_c) \geq 0.72c\Delta_k$$

and the desired inequality follows. □

In order to prove that $x_k + s_c$ will be accepted, we need to consider how $\overline{\varphi}$ and $\overline{f}$ are computed. Let us begin using the previous lemma to show that, for a small Δ, $\overline{\varphi}_i$, defined in (2.4), will depend on the predicted reduction of the infeasibility.

LEMMA 6.8. *Suppose that A1 holds and that x_k is not φ-stationary. Then there exists $\Delta_7 > 0$ such that*

$$\gamma_c P_{red}^{fsb}(x_k, s_c) > \gamma_f \varphi(x_k),$$

if $\varphi(x_k) < \epsilon_h \Delta_k^2$ and $\Delta_k \in (0, \Delta_7)$.

Proof. Lemma 6.7 ensures that

$$\frac{\gamma_c}{\gamma_f} P_{red}^{fsb}(x_k, s_c) \geq \frac{\gamma_c}{\gamma_f} c_4 \Delta_k > 0.$$

Defining $\Delta_7 = \min\{\gamma_c c_4/(\gamma_f \epsilon_h), \Delta_6\}$, where Δ_6 is given in Lemma 6.7, we have that

$$\frac{\gamma_c}{\gamma_f} P_{red}^{fsb}(x_k, s_c) \geq \epsilon_h \Delta_7 \Delta > \epsilon_h \Delta^2 \geq \varphi(x_k),$$

for all $\Delta \in (0, \Delta_7)$, so the desired result follows. □

Using Lemma 6.7 again, we can also show that $\overline{f}$, defined in (2.5), will depend on P_{red}^{opt} if De is sufficiently small.

LEMMA 6.9. *Suppose that A1, A2 and A4 hold, that $\{x_{k_i}\}$ is an infinite subsequence that converges to the feasible and regular point $x^* \in \Omega$, which is not stationary for (1.1), and that $\overline{\varphi}_k$ is given by (2.4). Then there exists $\Delta_8 > 0$ such that*

$$\gamma_f P_{red}^{opt}(H_k, x_k, s_c) \geq (\varphi(x_k) - \overline{\varphi}_k),$$

if $\varphi(x_k) < \min\{\epsilon_h, \epsilon\}\Delta_k^2$ and $\Delta_k \in (0, \Delta_8)$, where ϵ is defined as in Lemma 6.3.

Proof. From Lemma 6.3 we deduce that, if $\Delta_k \in (0, \Delta_1]$, then

$$\gamma_f P_{red}^{opt}(H_k, x_k, s_c) \geq \gamma_f c_3 \Delta_k.$$

Now, defining $\Delta_8 = \min\{\gamma_f c_3/\epsilon_h, \Delta_1\}$, we have

$$\gamma_f P_{red}^{opt}(H_k, x_k, s_c) \geq ep_h \Delta_8 \Delta \geq \epsilon_h \Delta_k^2 \geq \varphi(x_k) \geq (\varphi(x_k) - \overline{\varphi}_k)$$

and the desired conclusion follows. □

Lemma 6.6 assures that P_{red}^{opt} is sufficiently large when Δ is small. Let us prove now that the actual reduction of the merit function is sufficiently large so the second condition used in step 1.7 of algorithm 3.1 to decrease the trust region radius is never satisfied.

LEMMA 6.10. *Suppose that A1 and A4 hold and that $\theta_k < \theta_k^{sup}$, where θ_k is defined by (2.7) and θ_k^{sup} is defined by (2.8). Then there exists $\Delta_9 > 0$ such that*

$$A_{red}(x_k, s_c, \theta_k^{sup}) < \gamma_g P_{red}(H_k, x_k, s_c, \theta_k^{sup})$$

for all $\Delta_k \in (0, \Delta_9)$.

Proof. If $\theta_k < \theta_k^{sup}$, then $P_{red}(H_k, x_k, s_c) \geq (1/2) P_{red}^{fsb}$. This inequality, together with (6.10), gives that

$$P_{red}(H_k, x_k, s_c) \geq \frac{c_4}{2} \Delta_k,$$

for all $\Delta_k \in (0, \Delta_6)$, where c_4 and Δ_6 are defined in Lemma 6.7. But, from A1 and A4, we also have that

$$|A_{red}(\Delta_k) - P_{red}(\Delta_k)| \leq c_5 \Delta_k^2.$$

for some $c_5 > 0$. From the last two inequalities, we deduce that

$$\frac{|A_{red}(\Delta_k) - P_{red}(\Delta_k)|}{P_{red}(\Delta_k)} = \left| \frac{A_{red}(\Delta_k)}{P_{red}(\Delta_k)} - 1 \right| \leq \frac{2c_5}{c_4} \Delta_k. \tag{6.11}$$

Therefore, defining $\Delta_9 = \min\{(1-\gamma_g)c_4/(2c_5), \Delta_6\}$, we obtain the required result. □

In our last lemma, we will use the previous results to prove that, if $\delta \to 0$, there is no infinite subsequence that converges to a point that is not stationary for (1.1).

LEMMA 6.11. *Suppose that A1, A2, A3 and A4 hold. Suppose also that the limit points of the infinite sequence $\{x_k\}$ are feasible and regular and that $\lim_{k\to\infty} \Delta_k = 0$. Then there exists a limit point of $\{x_k\}$ that is a stationary point of (1.1).*

Proof. Assumption A3 implies that there exists a convergent subsequence $\{x_{k_i}\}$. Let us suppose, for the purpose of obtaining a contradiction, that the limit point of this subsequence is not stationary for (1.1).

Since $\lim_{i\to\infty} \varphi(x_{k_i}) = 0$ and, at the begining of iteration k, the trust region radius satisfies $\Delta_{k_i} \geq \Delta_{min}$, there must exist $k_8 \geq k_3$ (defined in Lemma (6.3)) such that, for $k_i > k_8$, the condition $\varphi(x_{k_i}) \leq \epsilon \Delta_{k_i}^2$ is satisfied, so (6.8) holds.

But, from (2.7) and (2.3), we have that $\theta_{k_i} < 1$, so Lemma 6.10 applies and (6.11) also holds if $\Delta_{k_i} < \Delta_9$.

Thus, supposing that $k_i > k_8$ and $\Delta_{k_i} < \Delta_9$, the point $x_{k_i} + s_c$ would only be rejected and, consequently, the trust region radius would only be reduced if $f(x_{k_i} + s_c) \geq \mathcal{P}(\mathcal{F}_{k_i}, \varphi(x_{k_i} + s_c))$.

Now, we need to consider separately two mutually exclusive situations. First, let us suppose that x_{k_i} is feasible. In this case, Lemma 4.2 assures that, for $\Delta < \overline{\Delta}_4$ (defined in (4.5)), the step is accepted and the trust region radius need not to be reduced further.

On the other hand, if x_{k_i} is not φ-stationary, Lemmas 6.8 and 6.9 assure that for $\Delta_{k_i} \le \min\{\Delta_7, \Delta_8\} = \Delta_{10}$, the definition of $\overline{\eta}$ given in (2.9) holds, so $0 < \overline{\eta} < 1$ and $f(x_{k_i} + s_c) > \mathcal{P}(\mathcal{F}_{k_i}, \varphi(x_{k_i} + s_c))$ is equivalent to (2.6).

Now, following the same steps used in Lemma 6.10, we can use (6.11) one more time to show that, when $\Delta_{k_i} < \min\{(1-\overline{\eta})c_4/(2c_5), \Delta_6\} = \Delta_{11}$, where Δ_6 is defined in Lemma 6.7, inequality (2.6) is satisfied and Δ_{k_i} is not reduced at step 1.7.1 of Algorithm 3.1.

Therefore, $\Delta_{k_i} \ge \alpha_R \min\{\overline{\Delta}_4, \Delta_9, \Delta_{10}, \Delta_{11}\}$, which contradicts the hypotesis that $\lim_{k\to\infty} \Delta_k = 0$, so we conclude that the limit point of the subsequence $\{x_{k_i}\}$ is a stationary point of (1.1). □

Finally, let us state a theorem that puts together all of the results presented so far.

THEOREM 6.12. *Suppose that A1, A2, A3 and A4 hold and that* $\{x_k\}$ *is an infinite sequence generated by algorithm 3.1. Then either the restoration converges to a* φ*-stationary but infeasible point of (1.1), or* $\lim_{k\to\infty} \varphi(x_k) = 0$. *Moreover, if the restoration always succeeds and all of the limit points of* $\{x_k\}$ *are regular, there exists a limit point* x^* *that is a stationary point for (1.1). In particular, if all of the* φ*-stationary points area feasible and regular, then there exists a subsequence of* $\{x_k\}$ *that converges to a feasible, regular ans stationary point of (1.1).*

Proof. This result is a direct consequence of Lemmas 5.1, 5.2, 6.4, 6.5 and 6.11. □

7. Conclusions. In this paper, we depict the general framework of an SQP algorithm that uses a piecewise linear merit function to accept and reject steps. This approach combines ideas from both merit functions and the filter introduced by Fletcher and Leiffer in [7].

The use of several penalty parameters defined automatically by the previous iterates avoids the premature reduction of θ as well as the zigzagging that can occur when a nonmonotone strategy is used to update this parameter. The new method is also less tolerant than the filter method, since we do not accept points that marginally reduce the infeasibility or the objective function.

As the next steps of this work, we intend to test the algorithm with some problems from the CUTEr library and devise a strategy to circunvent the Maratos effect.

Acknowledgments. The author would like to thank Prof. José Mario Martínez for his insightful comments.

REFERENCES

[1] C. AUDET AND J. E. DENNIS, *A pattern search filter method for nonlinear programming without derivatives*, Tech. Report TR00-09, Dep. Computational and Applied Mathematics, Rice University, Houston, TX, 2000.

[2] R. H. BYRD, J. C. GILBERT AND J. NOCEDAL, *A trust region method based on interior point techniques for nonlinear programming*, Math. Programming, 89 (2000), pp. 149–185.

[3] R. H. BYRD, M. E. HRIBAR AND J. NOCEDAL, *An interior point algorithm for large-scale nonlinear programming*, SIAM J. Optim., 9 (2000), pp. 877–900.

[4] C. M. CHIN AND R. FLETCHER, *On the global convergence of an SLP-filter algorithm that takes EQP steps*, Math. Programming, 96 (2003), pp. 161–177.

[5] A. CONN, N. I. M. GOULD AND P. L. TOINT, *Trust Region Methods*, SIAM, Philadelphia, PA, 2000.

[6] R. FLETCHER, N. I. M. GOULD, S. LEYFFER, P. L. TOINT AND A. WÄCHTER, *Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming*, SIAM J. Optim., 13 (2002), pp. 635–659.

[7] R. FLETCHER AND S. LEYFFER, *Nonlinear programming without a penalty function*, Math. Programming, 91 (2002), pp. 239–269.

[8] R. Fletcher, S. Leyffer and P. L. Toint, *On the global convergence of a filter-SQP algorithm*, SIAM J. Optim., 13 (2002), pp. 44–59.
[9] F. A. M. Gomes, M. C. Maciel and J. M. Martínez, *Nonlinear programming algorithms using trust regions and augmented Lagrangians with nonmonotone penalty parameters*, Math. Programming, 84 (1999), pp. 161–200.
[10] C. Gonzaga, E. Karas and M. Vanti, *A globally convergent filter method for nonlinear programming*, SIAM J. Optim., 14 (2003), pp. 646–669.
[11] N. I. M. Gould, S. Leyffer and P. L. Toint, *A multidimensional filter algorithm for nonlinear equations and nonlinear least-squares*, Tech. Report 03/15, Dep. Mathematics, University of Namur, Namur, Belgium, 2003.
[12] N. I. M. Gould, C. Sainvitu and P. L. Toint, *A filter-trust-region method for unconstrained optimization*, Tech. Report 04/03, Dep. Mathematics, University of Namur, Namur, Belgium, 2004.
[13] N. I. M. Gould and P. L. Toint, *Global convergence of a hybrid trust-region SQP-filter algorithm for general nonlinear programming*, Tech. report 01/07, Dep. Mathematics, University of Namur, Namur, Belgium, 2001.
[14] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, NY, 1999.
[15] T. D. Plantenga, *A trust-region method for nonlinear programming based on primal interior point techniques*, SIAM J. Sci. Comput., 20, (1999), pp. 282–305.
[16] M. Ulbrich, S. Ulbrich and L. N. Vicente, *A globally convergent primal-dual interior point filter method for nonconvex nonlinear programming*, Math. Programming, to appear.
[17] A. Wächter and L. T. Biegler, *Global and local convergence of line search filter methods for nonlinear programming*, Tech. Report CAPD B-01-09, Dep. Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, 2001.
[18] A. Wächter and L. T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, IBM Research Report RC23149, IBM T. J. Watson Research Center, Yorktown Heights, NY, 2004.