IBM Research Report

A shorter-step trust region algorithm for the minimization of nonlinear partially separable functions

Johara Shahabuddin
IBM Research Division
IBM India Research Lab
Block I, I.I.T. Campus, Hauz Khas
New Delhi - 110016. India.

A SHORTER-STEP TRUST REGION ALGORITHM FOR THE MINIMIZATION OF NONLINEAR PARTIALLY SEPARABLE FUNCTIONS.

JOHARA SHAHABUDDIN*

Abstract. In trust region algorithms for nonlinear minimization, the fit between the objective function and its model is tested in each iteration to update the trust region radius. This radius restricts the step length equally in all directions. However, for a partially separable function, the accuracy of the model may be different for the various parts of the objective function. One would like to allow longer steps in the subspaces of the more accurately modeled parts, with the expectation that the extra flexibility will give faster convergence.

The excellent idea of structuring the trust region for partially separable problems belongs to [A. R. Conn, Nick Gould, A. Sartenaer, and Ph. L. Toint, Convergence properties of minimization algorithms for convex constraints using a structured trust region, *SIAM J. Optim.*, 6 (1996), pp. 1059–1086]. They prove global first order convergence for convex-constrained problems. Their trust region update mechanism and second order analysis are complex.

The sufficient decrease condition in Conn et al. is changed so that the exact minimizer of the model within the trust region will always satisfy it. However, we add another condition on the step that is needed to guarantee global convergence. New and simpler update mechanisms for the trust region radii are investigated. First order convergence is proved for the convex-constrained problem. Second order convergence results are proved for the unconstrained case.

Key words. trust region algorithm, partial separability, unconstrained, convex constraints, global convergence, structured problem, nonlinear programming, large-scale programming

AMS subject classifications. 90C30, 65K05

*IBM-India Research Lab, Indian Institute of Technology, Hauz Khas, New Delhi 110 016, India (sjohara@in.ibm.com)

1. Introduction.

Among nonlinear programming approaches, trust region algorithms are known to have strong convergence properties, globally converging to a local minimum. The main idea in these algorithms is to adjust the maximum allowed length of each iterative step depending on how accurately the nonlinear objective function is modeled by a quadratic. This allowed step length is known as the trust region radius. Each step is computed as an exact or approximate minimizer of a quadratic model of the problem within a spherical region defined by the trust region radius. The convex-constrained minimization problem that we are interested in is:

$$(P) \qquad \min_{x \in X} f(x),$$

where X is a closed convex subset of $\Re^n$, and $f : \Re^n \to \Re$.

Partially separable optimization problems may arise from any system that is modeled as a set of loosely connected subsystems. There has been much recent research trying to improve the efficiency of the algorithms used to optimize partially separable functions, as in [1], [7], [8], [13].

A nonlinear function $f(x)$, $x \in \Re^n$ is defined to be **partially separable** if it can be written as

$$f(x) = \sum_{i=1}^{p} f_i(x),$$

where each $f_i(x),\ i = 1, \ldots, p$, is a nonlinear function with a large invariant subspace. The functions $f_i(x),\ i = 1, \ldots, p$, are known as **element functions**.

A simple instance of an element function with a large invariant subspace is one which depends on only a few of the variables, as is often the case in large problems. We do not specify how large a large invariant subspace needs to be.

Usual trust region algorithms maintain a single trust region radius that restricts the step length equally in all directions. But is the function equally nonlinear in all directions? The original structured trust region idea, as proposed in [3], responds to the observation that not all parts of a partially separable function are equally nonlinear, and the step can be allowed to be longer in the parts that are more accurately modeled. One would expect that the extra flexibility allowed would lead to faster convergence.

The structure of the new trust region is intuitive, and we use the same structure as in [3]. There is a separate trust region radius for each of the elemental subspaces. The intersection of the *elemental trust regions* thus defined is then the structured trust region.

However, the strong convergence results that go with trust region methods do not carry over to the structured case easily. The trust region strategy loses some robustness in the process of structuring. This well-known strategy is: when the trust region size decreases, the function and the model are more in agreement. We look more carefully, and we see that only when the quadratic terms predominate over higher order terms, will this agreement happen. And these terms predominate only in good directions such as the negative gradient direction, directions of negative curvature, or the Newton direction. There are other directions where these terms may be negligible in comparison with higher order terms, and as the trust region size decreases, the function and the model may diverge from agreement.

Unstructured approaches have a spherical trust region, where only the good directions are ever chosen for a step. In contrast, structured trust regions may be shaped to be skewed against such directions after a step is taken. For example, the trust region shape may allow only a tiny step in the direction of the negative gradient, while allowing a long step in an orthogonal direction. Allowing such steps, as we see in this paper, prevents us from proving that all limit points of the sequence of iterates generated by this algorithm are first order critical, although it does allow us to prove that at least one limit point must be first order critical.

Thus, the typical class of sufficient decrease conditions used for trust region algorithms does not guarantee convergence of a structured approach. (Sufficient decrease conditions put a lower bound on the decrease in the quadratic model that an acceptable approximate solution must satisfy.) Unstructured trust region algorithms are globally convergent (converge to a local minimum from any starting point) only if the step satisfies such a condition.

We propose two conditions to overcome this problem, one of which is a sufficient decrease condition, while the second is an additional restriction on the step direction. In [3], a sufficient decrease condition is used that the exact minimum of the model in the iterative subproblem cannot always attain. Thus, an additional restriction on the step is hidden within it. This hidden condition is explicitly stated by our second condition. (Despite the apparent similarity, our conditions on the step were proposed independently of theirs, after we had pointed out to them that an earlier condition they had been using was unsuitable.)

Ideally, instead of conditions on the step, the trust region update mechanism should naturally, as it does for the unstructured approaches, bias the trust region shape towards the good directions. So we experimented with the update mechanisms as well. Five simpler and more intuitive update mechanisms than in [3] are described here. Our results have been proved for only one of update mechanisms so far. But the other ones are shown to have good characteristics.

The next section has the basic notation and assumptions, and concludes with our structured trust region algorithm. In the third section are the two conditions we need on the step. In the fourth are the new trust region update mechanisms and some results about them. The fifth section has proofs of first order convergence results for the convex-constrained minimization problem, modeled on the analysis in [3]. The sixth has second order convergence results for the unconstrained case, modeled on the analysis in [9]. Our aim is to evaluate structured approaches, and it is appropriate to begin with the simpler unconstrained minimization problem for the more complicated second order analysis.

2. The Shorter-Step Algorithm. The problem (P) is solved iteratively, with x_0 as the given starting point. In each iteration, $f(x_k+s)-f(x_k), k=0,1,2,\ldots$ is modeled in terms of its gradient and Hessian. The model, denoted by $m_k(s)$, is approximately minimized at s_k. The trust regions of the elements are then updated, and so is x_k. An iteration where $x_{k+1}=x_k+s_k$ is called a *successful iteration*. Otherwise, $x_{k+1}=x_k$, and the iteration is called *unsuccessful*. If the iteration is unsuccessful, trust region sizes of some of the elements will be reduced, and the same model will be minimized over the new trust region in the next iteration.

The l_2-norm is used throughout this paper unless otherwise specified. (For other norms, the convergence proofs remain valid with changes in values of the appropriate constants.)

We begin with a feasibility assumption on the convex feasible region X of the minimization problem (P):

Assumption 2.1. *X has a non-empty interior.*

The following basic assumptions are needed on f:

Assumption 2.2. *The function f is bounded below on the set $\mathcal{L} := \{x \in X : \ f(x) \le f(x_0)\}$.*

Assumption 2.3. *Each f_i, $i=1,\ldots,p$, and hence f, is twice continuously differentiable on an open set containing $\mathcal{L}$.*

Assumption 2.4. *There exists a positive constant $\chi_H \ge 1$ such that $\|\nabla^2 f(x)\| \le \chi_H$ and $\|\nabla^2 f_i(x)\| \le \chi_H$, $i=1,\ldots,p$, on an open set containing $\mathcal{L}$.*

Define $g_k := \nabla f(x_k)$. Let B_k be an approximation to the Hessian $\nabla^2 f(x_k)$.

DEFINITION 2.5. *The* **overall model** *$m_k(s)$ of $f(x_k+s)-f(x_k)$ is defined as:*

$$m_k(s) := g_k^T s + \frac{1}{2} s^T B_k s,$$

Each element function is modeled in terms of the first three terms of its Taylor series. Let $g_{i,k} := \nabla f_i(x_k)$. Let $B_{i,k}$ be an approximation to $\nabla^2 f_i(x_k)$ such that $\sum_{i=1}^p B_{i,k} = B_k$.

DEFINITION 2.6. *The* **elemental model** *for $f_i(x_k+s)-f_i(x_k)$ is defined as follows:*

$$m_{i,k}(s) := g_{i,k}^T s + \frac{1}{2} s^T B_{i,k} s.$$

Gradient and criticality measure. We assume that the exact derivative g_k is available to simplify our analysis. The derivative is sometimes generalized to the following approximation as in [3], [9]: $\|g_k - \nabla f(x_k)\| \le \kappa_x \Delta_{\min,k}$, where $\Delta_{\min,k}$ is the minimum of all the elemental trust region radii, and κ_x is a nonnegative constant. Our analysis would continue to hold for such an approximation.

We define $\alpha(x_k)$ as a *criticality measure* for the problem (P) as follows:

$$\alpha(x_k) = \alpha_k := \left| \min_{(x_k+d)\in X} \frac{g_k^T d}{\|d\|} \right|. \tag{2.1}$$

Notice that when X is convex, $\alpha_k = 0$ if and only if x_k satisfies first order criticality conditions for the problem (P). (See [3] for a proof of this.) If $X = \Re^n$ (the problem is unconstrained) then it is easy to see that $\alpha_k = \|g_k\|$.

Two functions are defined next. The first is a generalization of the criticality measure α_k, and the second one is a path that follows the negative gradient projected onto the feasible region X.

$$\alpha_k(t) := |\min_{\substack{x_k+d\in X\\ \|d\|\le t}} g_k^T d\,|, \tag{2.2}$$

$$d_k(t) := \arg \min_{\substack{x_k+d\in X\\ \|d\|\le t}} g_k^T d.$$

We need the following lemmas about the criticality measure (for proofs, see Lemmas 2.2 and 3.1 in [4]).

LEMMA 2.7. *If Assumptions 2.3 and 2.1 hold then for all $k \ge 0$, the function $t \mapsto \alpha_k(t)$ is continuous and nondecreasing and the function $t \mapsto \frac{\alpha_k(t)}{t}$ is non-increasing for $t \ge 0$.*

LEMMA 2.8. *If Assumptions 2.3 and 2.1 hold then the function α is continuous with respect to its argument.*

We need α to be uniformly continuous on $\mathcal{L}$. This certainly holds if $\mathcal{L}$ is bounded (by the lemma above), or if the problem is unconstrained, since then $\alpha(x) = \|\nabla f(x)\|$ whose derivative is bounded by Assumption 2.4. Otherwise, assume the following:

Assumption 2.9. *The function $\alpha(x)$ is uniformly continuous in an open set containing $\mathcal{L}$.*

Hessian approximation and Rayleigh quotient. Several different assumptions related to the Hessian approximations B_k have been used by earlier authors. Most trust region algorithms assume that $\|B_k\|$ is uniformly bounded, as in [2], [6], [9]. Powell ([11]) allowed the bound on $\|B_k\|$ to grow linearly with the iteration number k, while still obtaining the same convergence results. His work motivates an assumption about B_k used in a series of trust region algorithms by Conn, Gould and Toint ([3], [4], [12]), which is weaker than assuming it to be uniformly bounded, and which holds when certain quasi-Newton updates are used. This is the assumption used in our first order convergence analysis, described in the rest of this section.

We define the *generalized Rayleigh quotient* of a function f at x along $s \ne 0$:

$$\omega(f, x, s) := \frac{2}{\|s\|^2}[f(x+s) - f(x) - \nabla f(x)^T s].$$

Because of the assumption that $\nabla^2 f$ is bounded, $|\omega(f_i, x, s)| \le L_h$ for all i if x and $x + s$ lie in $\mathcal{L}$, where $L_h \ge 1$ is a positive constant. We define a version of the generalized Rayleigh quotient of m_k:

$$\beta_k := 1 + \max_{q=1,\ldots,k}(\max(|\omega(m_q, 0, s_q)|, \max_{i=1,\ldots,p}(|\omega(m_{i,q}, 0, s_q)|))).$$

Assumption 2.10. $\sum_{k=0}^{\infty} \frac{1}{\beta_k} = +\infty.$

The trust region structure. We define $\Delta_{i,k}, i = 1, \ldots, p$ to be the trust region radii for the p element functions. These are updated in each iteration and together define the overall trust region structure in the following manner:

DEFINITION 2.11. *The null space N of a function $f(x)$ is defined to be the set $\{v \mid f(x+v) = f(x)\}$.*

DEFINITION 2.12. *The range space R of a function $f(x)$ is defined to be the subspace orthogonal to N in R^n.*

Let R_i denote the range space of an element function $f_i, i = 1, \ldots, p$. Elemental models $m_{i,k}$ have the same range space R_i as f_i, for all i, k.

Let $P_{R_i}(s)$ denote the projection of a vector s onto R_i. The constraints in the subproblem (SP) then define the structured trust region as the intersection of *elemental trust regions*. We solve the problem (P) by approximately solving a sequence of subproblems of this form.

$$(SP) \qquad \min_{x\in X} m_k(s) = g_k^T s + \frac{1}{2} s^T B_k$$
$$\|P_{R_i}(s)\| \le \Delta_{i,k}, \quad i = 1, \ldots, p.$$

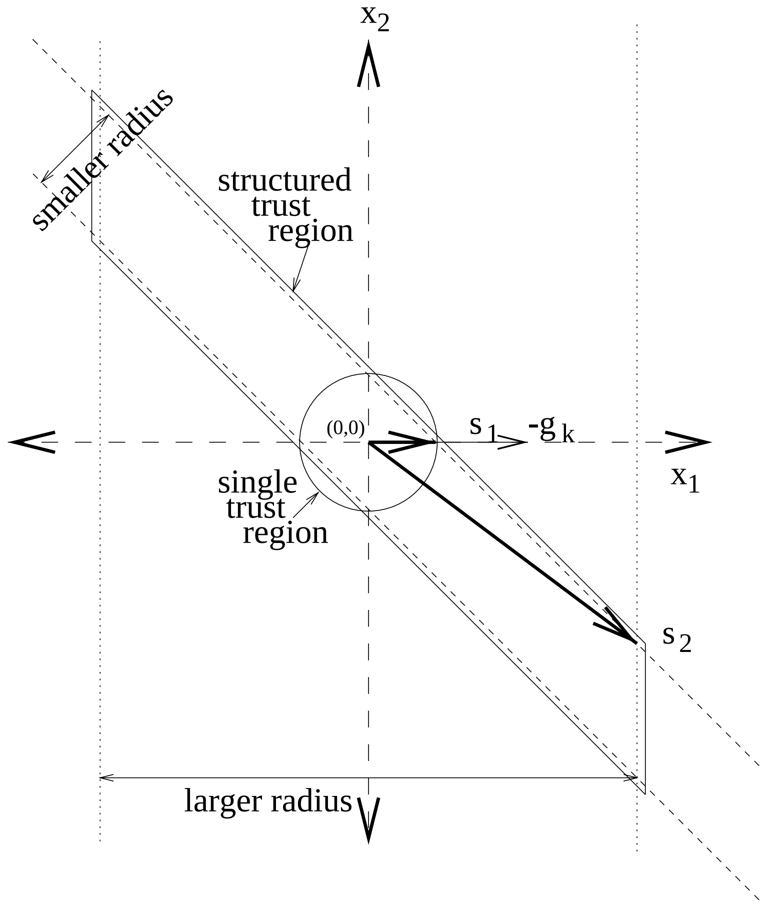

FIG. 2.1. *The allowed step s_2 is likely to be longer than the unstructured trust region step s_1, if the trust region is structured.*

Notice that $m_k(s) = \sum_{i=1}^{p} m_{i,k}(s)$.

We need definitions of the following functions of the trust region radii $\Delta_{i,k}$:

$$
\begin{aligned}
\Delta_{\min,k} &:= \min_{i=1,\ldots,p} (\Delta_{i,k}), \\
\Delta_{\max,k} &:= \max_{i=1,\ldots,p} (\Delta_{i,k}), \\
\Delta_{g,k} &:= \max\{\|s\| : s = tg_k,\ t \le 0,\ \|P_{R_i}(s)\| \le \Delta_{i,k},\ \forall i = 1,\ldots,p\}.
\end{aligned} \tag{2.3}
$$

Notice that $\Delta_{\min,k} \le \Delta_{g,k} \le \sqrt{n}\Delta_{\max,k}$. The function $\Delta_{g,k}$ represents the maximum allowed length of a step in the negative gradient direction, not considering the feasible region X of the original minimization problem.

An example. With the trust region structure and subproblem defined, we can now show how a longer step may be allowed by the above trust region structure. Refer to Figure 2.1 where a possible trust region configuration for the objective function $f(x) = x_1^2 + e^{(x_1+x_2)^2}$ is shown. The elemental range spaces are $R_1 = \mathcal{S}((1,0)')$ and $R_2 = \mathcal{S}((1,1)')$, where $\mathcal{S}(v_1,\ldots,v_m)$ stands for the span of m vectors $v_1,\ldots,v_m$. Since $f_1(x) = x_1^2$ is modeled perfectly by a quadratic, its trust region radius would be large. The function $f_2(x) = e^{(x_1+x_2)^2}$ is nonlinear and one would expect a small trust region radius. The intersection of these two trust region radii for the range spaces of $f_1(x)$ and $f_2(x)$ would give rise to the trust region shown, allowing us to take the step s_2 in the figure, assuming that a high negative gradient dominates the subproblem (SP) solution. If the size of the unstructured trust region was determined by the more nonlinear of the two elements (the circle in the figure), then only the smaller step s_1 in the figure would be allowed.

We now present the shorter-step algorithm. The algorithm follows the general form of the classical, unstructured, trust region algorithm. Two steps are a little different. In step 2 we must compute an approximate solution that satisfies another condition besides the usual sufficient decrease condition. This solution may be obtained by any standard technique for obtaining an approximate solution. Then in step 4, the multiple elemental radii are updated, rather than a single trust region radius.

2.13. **The shorter-step algorithm.** *Given $0 < \mu_1 \le \mu_2 < 1$, a feasible x_0, and starting values for the trust region sizes, the kth iteration takes the following form:*

1. *Given x_k, calculate g_k and B_k. Stop if x_k is a local minimum.*
2. *Solve subproblem (SP) approximately, to get s_k satisfying both the* sufficient decrease condition *(3.1) and the* shorter-step condition *(3.2).*
3. *Evaluate $f(x_k + s_k)$, and hence r_k.*
4. *Update the trust region radii according to one of the mechanisms in Section 4:*
   If $r_k \ge \mu_2$, some of the elemental radii increase.
   If $\mu_1 \le r_k < \mu_2$, increase some of the elemental radii, and decrease some of them.
   If $r_k < \mu_1$, some of the elemental radii decrease.
5. *If $r_k \ge \mu_1$ set $x_{k+1} = x_k + s_k$; else $x_{k+1} = x_k$.*

Other possible ways to structure the trust region. There are some simpler but less flexible methods than the ones to be presented, that are likely to be first and second order globally convergent.

First, look at the following subproblem, similar to (SP) that has been described in detail in [6], [9]. Assume that the elemental subspaces are spanned by basis vectors corresponding to the variables used by the respective element functions. Of the trust region radii that affect a given variable, let the minimum radius be denoted by Δ_k^j, $j = 1, \ldots, n$. This is taken as the trusted length in that coordinate direction. Define D_k to be an n-by-n diagonal matrix with $1/\Delta_k^j$, $j = 1, \ldots, n$, as its diagonal entries. The trust region here is ellipsoidal.

$$\begin{aligned} \min m_k(s) &= g_k^T s + \frac{1}{2} s^T B_k s \\ s^T D_k^T D_k s &\le 1. \end{aligned}$$

The original use of D_k is as a scaling matrix. Good convergence results have been proved for the above subproblem with suitable assumptions on D_k. But this subproblem is hard to solve if D_k is ill-conditioned. Thus, this approach to structuring has limited flexibility because it would not allow widely differing trust region sizes.

A second way to structure would be to impose an upper bound on $\Delta_{\max,k}/\Delta_{\min,k}$ (e.g., by allowing only reductions in $\Delta_{i,k}$'s that correspond to a large enough $\Delta_{\max,k}$). This approach is not very different from using a variable and bounded scaling matrix to define the trust region constraint, as discussed above.

Yet a third way to structure would be to impose the condition that $s_k^T g_k \ge \kappa \|s_k\| \|g_k\|$ (this would take the place of our condition (3.2)), where κ is a small positive fraction and thus obtain first order convergence. Then $s_k^T \nabla^2 f(x_k) s_k \ge -\kappa \lambda_k \|s_k\|^2$, where λ_k is the smallest eigenvalue of $\nabla^2 f(x_k)$, obtains second order convergence. Our attempt is to find more general conditions than either of these for s_k to satisfy. (See [3] for a discussion on this.)

3. The Conditions on the Step. Before we go on we need some terms that are used to compare changes in the values of the functions and their models:

$$\begin{aligned} \delta f_k &= f(x_k) - f(x_k + s_k), \\ \delta f_{i,k} &= f_i(x_k) - f_i(x_k + s_k), \\ \delta m_k &= -m_k(s_k), \\ \delta m_{i,k} &= -m_{i,k}(s_k), \\ r_k &= \delta f_k / \delta m_k, \\ r_{i,k} &= \delta f_{i,k} / \delta m_{i,k}. \end{aligned}$$

The step s_k must minimize $m_k(s)$ approximately, so as to satisfy the following sufficient decrease condition.

$$\delta m_k \ge \kappa \alpha_k \min\left(\frac{\alpha_k}{\beta_k}, \Delta_{\min,k}, 1\right), \tag{3.1}$$

where $\kappa > 0$ is a constant, α_k is the criticality measure defined in (2.1), and $\Delta_{\min,k}$ is as defined in (2.3).

This sufficient decrease condition is fairly typical of the ones existing in the literature. Also, there always exists a step that satisfies it. For a proof, see [4]. One point at which it is achieved is the point defined below.

DEFINITION 3.1. *The* **generalized Cauchy point** s_{kc} *is defined as the minimizer of* $m_k(d_k(t))$ *over* t.

In other words, s_{kc} is the minimizer of $m_k(s)$ along the projected gradient path $d_k(t)$, defined in (2.2). See Section 2.2 in [3] for justification. (In the unconstrained case, the projected gradient path is simply the negative gradient direction, and here s_{kc} is known as the **Cauchy point**.)

We need one other condition on the step, which we motivate next.

Motivation for the shorter-step condition. To explain why another condition on the step is required, we revert to the unconstrained situation, and compare the behavior of the normal unstructured trust region, against a structured trust region.

Unstructured trust region algorithms converge whenever $\delta m_k \geq -m_k(s_{kc})$, (e.g., see [9],) where s_{kc} is the Cauchy point as defined above. In other words, for some $\kappa > 0$:

$$\delta m_k \geq \kappa \|g_k\| \min(\frac{\|g_k\|}{\beta_k}, \Delta_k).$$

For a structured trust region, $\delta m_k \geq -m_k(s_{kc})$ would translate to, for some $\kappa > 0$:

$$\delta m_k \geq \kappa \|g_k\| \min(\frac{\|g_k\|}{\beta_k}, \Delta_{g,k}),$$

where $\Delta_{g,k}$ is as defined in (2.3). Now suppose that $\Delta_{g,k}$ is small enough that the term that dominates in the above condition is the $\Delta_{g,k}$ term.

In the unstructured trust region case, $\|s_k\| \leq \Delta_k = \Delta_{g,k}$, from the definition of $\Delta_{g,k}$. Now on an iteration where $r_k < \mu_1$ and s_k is orthogonal to the gradient, the trust region size reduces equally in all directions. With $\|s_{k+1}\|$ encouraged to be smaller by this, the solution to the subproblem is likely to be dominated more by the first order term in m_k and less by its higher-order terms, and since the trust region is symmetric, s_{k+1} is likely to be more parallel to the gradient, and thus we expect that $r_{k+1} \geq r_k$.

However, in a particular iteration of a structured trust region algorithm, this mechanism of improving the accuracy of fit between the function and model by reducing trust region size does not carry over. Here one may find that $r_k < \mu_1$, $\Delta_{g,k+1} < \Delta_{g,k}$ (of course, according to Algorithm 2.13, $\|g_{k+1}\| = \|g_k\|$), while $\|s_{k+1}\| > \|s_k\|$, because $\Delta_{\max_k}$ is due to a well-modeled element and thus did not decrease. In this way, $\|s_{k+1}\|$ may be more orthogonal to g_{k+1} than s_k is to g_k, even if it is shorter in some elemental subspaces.

Thus we do not any longer expect that r_{k+1} be greater than r_k (since the first order term is may not become more dominant) for bad iterations. The inequality in the trust region sizes resulting from this may tend to encourage successive steps to be turned away from the negative gradient direction, causing worse and worse fits. Thus, there is no guarantee that reduction of trust region sizes would lead to a chance of a better fit between function and model, unless we do it cleverly enough. If we did not reduce the sizes appropriately, the algorithm may stall far from $\|g_k\| = 0$.

Ideally, we should solve this problem by showing the relative trust region sizes in the various directions to be well-behaved (these fluctuate according to the trust region update mechanism). We were unable to show this for our update mechanisms, and found that we needed another condition, which we call the **shorter-step condition** to ensure first order convergence:

$$\delta m_k \geq \kappa' \|s_k\| \alpha_k, \tag{3.2}$$

where κ' is a positive constant. We call this the *shorter-step condition*. It is not too restrictive with a small constant of proportionality κ'. In our implementations, we have not come across a case where our choice of step violates this condition.

In the unconstrained case, where $\alpha_k = \|g_k\|$, condition (3.2) is weaker than assuming that s_k is bounded away from orthogonality to g_k. We show below that for any feasible x_k, there exists an s_k fulfilling both these conditions simultaneously. Unfortunately, the exact minimizer of the subproblem does not satisfy (3.2) and so somewhat limits our choice of step. However for the unstructured trust region subproblem, both the exact minimizer as well as the Cauchy point always satisfy (3.2).

Is it possible to satisfy the sufficient decrease (3.1) and the shorter-step (3.2) conditions simultaneously? The following lemma shows that a projected gradient step inside a region of radius $\Delta_{\min,k}$ satisfies both the conditions at once.

LEMMA 3.2. *Let d denote $d_k(t)$ as defined in (2.2), where $t := \min(1, \Delta_{\min,k})$. Let s_k minimize $m_k(s)$ as defined in (2.5) over $s \in \{t'd \ : \ 0 < t' \le 1\}$. Then s_k is feasible in (SP) and satisfies both conditions (3.1) and (3.2), for κ and κ' at most $1/2$.*

Proof. We have $\|s_k\| \le \|d\| \le t = \min(1, \Delta_{\min,k}) \le 1$. This and (2.2) show that s_k is feasible in (SP).

Case 1. $d^T B_k d \le 0$. We see that the decrease in the model corresponding to s_k is $\delta m_k(s_k) \ge -g_k^T d = \alpha_k(t)$. But by Lemma 2.7, $\alpha_k(t)/t \ge \alpha_k(1)/1 = \alpha_k$. Thus, $\delta m_k(s_k) \ge t\alpha_k$. So (3.1) is satisfied for any $\kappa \le 1$. Notice that $t \ge \|s_k\|$; hence (3.2) is also satisfied for any $\kappa' \le 1$. So both the conditions hold in this case.

Case 2. $d^T B_k d > 0$. We see that the change in the model corresponding to $s_k = t'd$ is $\delta m_k(s_k) = -t' g_k^T d - \frac{t'^2}{2} d^T B_k d$. If t' were unconstrained, the exact maximizer of $\delta m_k(t'd)$ would be $t' = t_* := \frac{-g_k^T d}{d^T B_k d}$.

Now if $t_* \ge 1$, then $s_k = d$ and $d^T B_k d \le -g_k^T d$ so that $\delta m_k \ge -\frac{1}{2} g_k^T d = \frac{\alpha_k(t)}{2} \ge \frac{1}{2} t\alpha_k$, satisfying both conditions, for any $\kappa, \kappa' \le 1/2$. Else, $t_* < 1$ and $s_k = t_* d$ so that $\delta m_k = \frac{(g_k^T d)^2}{2 d^T B_k d} = -t_* g_k^T d/2$ (from the definition of t_*) $= \alpha_k(t)\|s_k\|/2\|d\| \ge \alpha_k(t)\|s_k\|/2t \ge \alpha_k \|s_k\|/2$, as before. So condition (3.2) is satisfied for any $\kappa' \le 1/2$. Now $\delta m_k = \frac{(g_k^T d)^2}{2 d^T B_k d} \ge \frac{\alpha_k(t)^2}{2t^2 \beta_k} \ge \frac{\alpha_k^2}{2\beta_k}$, satisfying condition (3.1) for any $\kappa \le 1/2$. ☐

4. Trust Region Update Mechanisms. The sizes of the elemental trust region radii in each iteration must be updated, so that a good match between the elemental functions and models is maintained. An algorithm that cannot guarantee that at least one trust region radius decreases in an iteration where there is a bad match between function and model, may stall. Also, at least one trust region radius should increase if need be (if the step lies well within a trust region, the radius need not be increased), when the function is modeled accurately, in order to allow the most flexibility possible in a given step (but this is not essential to the theory). Let us look at the difficulties in ensuring conditions such as these.

We classify the iteration depending on how well the function is modeled. As before, a *successful* iteration is one where $r_k \ge \mu_1$, so that $x_{k+1} = x_k + s_k$. Otherwise the iteration is *unsuccessful*. Let $0 < \mu_1 \le \mu_2 < 1$, as in Algorithm 2.13. A *bad iteration* is one where $r_k < \mu_1$. A *good iteration* is one where $r_k \ge \mu_2$. An *adequate iteration* is one in which $\mu_1 \le r_k < \mu_2$.

Similarly, we divide elements into three categories. Those with the best match between element function and its model are called *good elements*, those with the worst fit are called *bad elements*, while all the others are known as *adequate elements*.

The relative error problem. We simplify the following discussion by taking $\mu_1 = \mu_2 = \mu$. One would expect a naive version of Algorithm 2.13 that simply imitates the unstructured trust region algorithm, to update the radii in the following manner in each iteration: solve (SP) and then check the ratios $r_{i,k}$. Decrease the radius for which $r_{i,k} \le \mu$, $i = 1, \dots, p$, where $\mu \in (0,1)$, is a constant. Else increase the radius, if need be (there is no need to increase the radius if the actual step length is much smaller than the radius in a given elemental subspace).

However, this strategy may not be globally convergent. This is because some elements $\delta m_{i,k}$ may be negative, in contrast to the unstructured case where δm_k is always positive. Conn, Gould and Toint point out this complication in [3], and call it the *relative error problem*.

To understand what happens, refer to Figure 4.1. When the function is modeled accurately, the point $(\delta f_{i,k}, \delta m_{i,k})$ is close to the line with slope equal to 1. Thus, the point 2 in the figure represents a better fit between function and model, than the point 1, and the point 3 has a better fit than the point 4. Now let us apply the usual criterion to increase the trust region radius: $r_{i,k} \ge .25$. Point 3 is on the acceptable side of the line of slope .25, and point 4 on the other side when $\delta m_{i,k}$ is positive. However, when $\delta m_{i,k}$ is negative, the line allows point 1 to be acceptable but not point 2, even though point 2 has the better fit.

The picture suggests the following form to replace the one above:

1. If $r_k < \mu$ then
   (i) If $\delta m_{i,k} \ge 0$, and $r_{i,k} < \alpha_+$, $\alpha_+ \in (0,1)$, then decrease the trust region radius $\Delta_{i,k}$.
   (ii) If $\delta m_{i,k} < 0$, and $r_{i,k} \ge \alpha_-$, $\alpha_- \in (1,\infty)$, then decrease the trust region radius $\Delta_{i,k}$.
2. If $r_k \ge \mu$ and if conditions (i) and (ii) are both false, increase the trust region radius $\Delta_{i,k}$.

This separation criterion, depicted in Figure 4.2, is still not safe from stalling. Stalling can happen because of the non-convexity in the separation shown in the figure. For an arbitrary choice of slopes for the two lines separating good and bad elements, there is no way to ensure that at least one trust region radius decreases whenever there is a bad iteration. Indeed, suppose conditions (i) and (ii) are

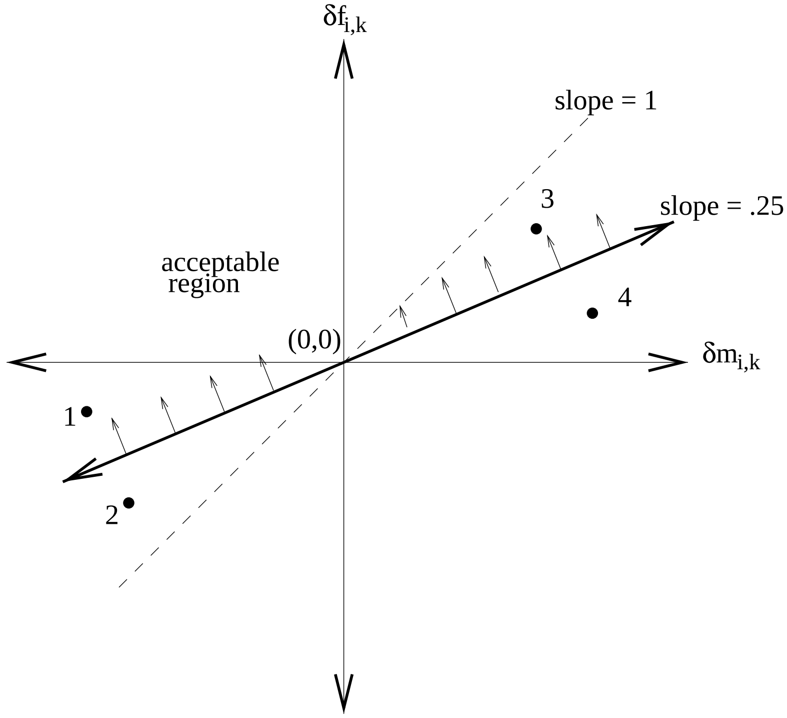

FIG. 4.1. *A naive separation of elements can go wrong: Even if $\delta f_{i,k} = \delta m_{i,k}$, a perfect fit, the ith trust region radius would reduce if $\delta m_{i,k} < 0$.*

both false for all i. Then,

$$\delta f_k = \sum_{i=1}^{p} \delta f_{i,k} \geq \alpha_+ \sum_{i:\delta m_{i,k}>0} \delta m_{i,k} + \alpha_- \sum_{i:\delta m_{i,k}<0} \delta m_{i,k},$$

which does not guarantee that $r_k > 0$. So we could have an iteration where $r_k \leq 0$, and none of the elemental trust regions $\Delta_{i,k}$ decrease. And this means stalling at the point x_k. In other words, even when the ratios $\delta f_{i,k}/\delta m_{i,k}$ are close to 1, the overall ratio $\delta f_k/\delta m_k$ is not guaranteed to be close to 1. Since elemental differences (or errors) between function and model are individually small but the overall difference (or error) may be large, this difficulty is called the *relative error problem.* This happens whenever there is a large cancellation between positive and negative $\delta m_{i,k}$'s.

We have found a way to calculate the slopes α_+ and α_- so as to guarantee that when both conditions (i) and (ii) are false for all i, then $r_k \geq \mu$. We describe this method in the next subsection.

Figure 4.3 illustrates another way to avoid relative error, which has been used in [3]. Their criterion to separate good elements from bad elements is quite complicated, but the basic idea is the one depicted in the figure:

1. If $r_k \geq \mu$ and if $\delta f_{i,k} \geq \delta m_{i,k} - \frac{(1-\mu)}{p}\delta m_k$ then increase the trust region radius $\Delta_{i,k}$.
2. Else if $r_k < \mu$, decrease $\Delta_{i,k}$.

The dark line in the figure separates the good elements from the bad elements, and the distance of the line from the origin is dependent on the value of δm_k.

To summarize this subsection, we must look for separation criteria that ensure that at least one trust region gets reduced in a bad iteration, and, if possible, at least one trust region is increased in a good iteration. There are two types of difficulties that can happen with an incorrect separation criterion — first, the algorithm could stall (i.e., stay stuck at the same non-optimal x_k, with no decrease in trust region radius), and second, it could cycle (i.e., stay stuck at the same non-optimal x_k, with the same pattern of increases and decreases in trust region radii repeated in a cycle). We have two possible forms for such separation criteria as depicted by Figures 4.2 and 4.3. In the next subsection we look at separation criteria motivated by this discussion.

Earlier separation criteria. A lot of flexibility is allowed in the choice of separation criterion and the way trust region radii are updated, and many variants of these schemes exist. We first briefly describe some of these different updating methods to give a sense of the substrate we are building on. We then go on to propose our separation criteria, proving some simple lemmas that are needed for the convergence theory that follows in later chapters, showing that at least one trust region radius decreases in a bad iteration.

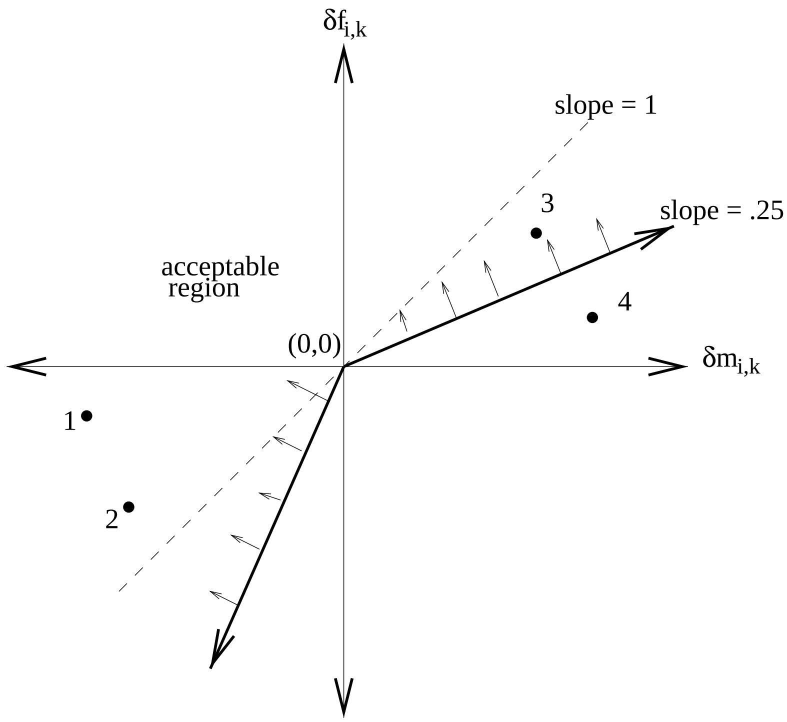

FIG. 4.2. *The dark boundary separates the good and bad elements. Trust region radii for elements that map onto the right of this line, are decreased. The proposed sloped criteria for separation of elements are of this form.*

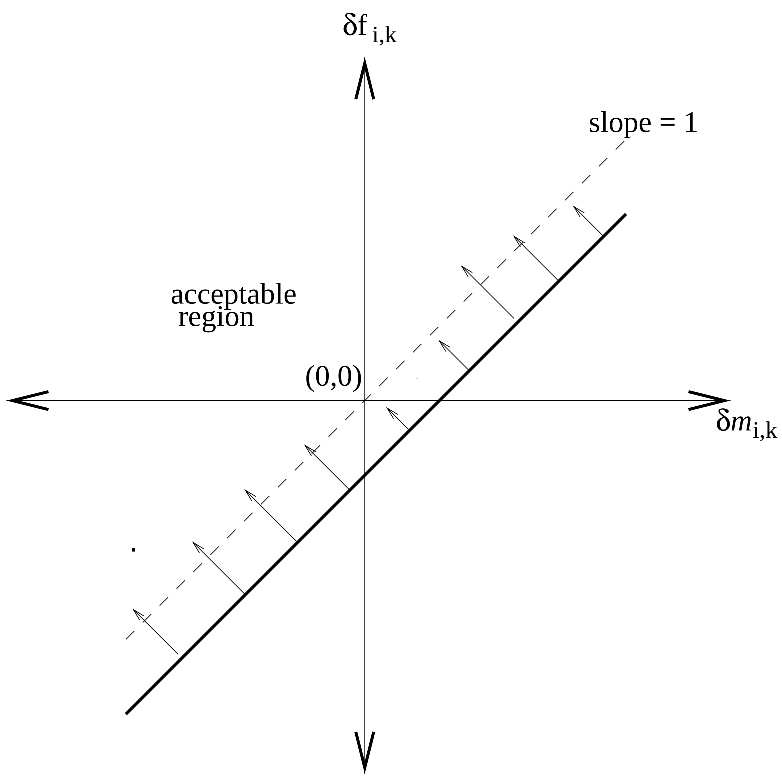

FIG. 4.3. *A line parallel to the line of slope=1 is used to distinguish good and bad elements. This is the basis for the proposed parallel separation criterion.*

Here are some of the separation criteria used for unstructured trust region algorithms in the past.

Fletcher [6]:

1. If $r_k < .25$ then $\Delta_{k+1} = \|s_k\|/4$,
2. if $r_k > .75$, and $\|s_k\| = \Delta_k$ then $\Delta_{k+1} = 2\Delta_k$,
3. otherwise $\Delta_{k+1} = \Delta_k$.

Moré [9]:
Let $0 < \mu_1 < \mu_2 < 1$ and $0 < \gamma_1 < 1 < \gamma_2$ be given.

1. If $r_k \le \mu_1$ then $\Delta_{k+1} \in (0, \gamma_1 \Delta_k]$,
2. if $r_k \in (\mu_1, \mu_2)$ then $\Delta_{k+1} \in [\gamma_1 \Delta_k, \Delta_k]$,
3. if $r_k \ge \mu_2$ then $\Delta_{k+1} \in [\Delta_k, \gamma_2 \Delta_k]$.

LANCELOT [5]:
Let $0 < \mu_1 < \mu_2 < 1$ and $0 < \gamma_1 < \gamma_0 < 1 \le \gamma_2$ be given,
$\gamma = \max\{\gamma_1, \frac{(1-\mu_2) g_k^T \|s_k\|}{(1-\mu_2)(f_k + g_k^T \|s_k\|) - \mu_2 \delta m_k - \delta f_k}\}$ (this formula comes from trying to interpolate to find a good step), $\gamma_1^{(k)} = \max[\gamma_1, \gamma_0 \|s_k\| / \Delta_k]$, and $\gamma_2^{(k)} = \max[1, \gamma_2 \|s_k\| / \Delta_k]$.

1. If $r_k < 0$ then $\Delta_{k+1} = \gamma \Delta_k$,
2. if $r_k \in (0, \mu_1)$ then $\Delta_{k+1} = \gamma_1^{(k)} \Delta_k$,
3. if $r_k \in [\mu_1, \mu_2)$ then $\Delta_{k+1} = \Delta_k$,
4. if $r_k \ge \mu_2$ then $\Delta_{k+1} = \gamma_2^{(k)} \Delta_k$.

Notice the flexibility of Moré's scheme in updating the trust region radius between iterations, where the constants are restricted only to the extent that they lie within certain intervals. It has been seen that the algorithm is insensitive to the value of the constants [6]. Another noticeable fact is that the algorithms try to cut off a bad step (a step where $r_k < 0$) and to increase the trust region size (on a good iteration) only if it is likely to lead to a longer step (you can see this wherever Δ_{k+1} is dependent on s_k in the above criteria), to prevent excessive and unnecessary blowup in the size of the radius. All this goes to say that the trust region updates are fairly heuristic, subject to a minimal requirement: the radius must decrease on a bad iteration.

Conn, Gould and Toint in [3], divide elements into *negligible* elements and *meaningful* elements. They choose the constants so as to ensure that at least one trust region radius must be decreased on a bad iteration. They allow trust regions to increase or decrease in the good iterations, and allow only decreases in bad iterations, to prevent cycling as discussed above.

Conn, Gould and Toint [3]:
Let $0 < \gamma_1 \le \gamma_2 < 1 \le \gamma_3$, $0 < \eta_1 \le \eta_2 < \eta_3 < 1$, $0 < \mu_1 < \mu_2 < 1$.
Define the set of negligible elements as $N_k := \{i \in 1, \ldots, p \;:\; |\delta m_{i,k}| \le \frac{\mu_1}{p} \delta m_k\}$.
Define the set of meaningful elements as its complement $M_k := \{1, \ldots, p\} \setminus N_k$.

1. For $i \in M_k$,
   (i) if $r_k \ge \eta_1$ and $\delta f_{i,k} \ge \delta m_{i,k} - \frac{1-\eta_3}{p} \delta m_k$ then $\Delta_{i,k+1} \in [1, \gamma_3] \Delta_{i,k}$,
   (ii) if $\delta m_{i,k} - \frac{1-\eta_3}{p} \delta m_k > \delta f_{i,k} \ge \delta m_{i,k} - \frac{1-\eta_2}{p} \delta m_k$ then $\Delta_{i,k+1} \in [\gamma_2, 1] \Delta_{i,k}$,
   (iii) if $\delta f_{i,k} < \delta m_{i,k} - \frac{1-\eta_2}{p} \delta m_k$ then $\Delta_{i,k+1} \in [\gamma_1, \gamma_2] \Delta_{i,k}$,
   (iv) otherwise $\Delta_{i,k+1} = \Delta_{i,k}$.
2. For $i \in N_k$,
   (i) if $r_k \ge \eta_1$ and $|\delta f_{i,k}| \le \frac{\mu_2}{p} \delta m_k$ then $\Delta_{i,k+1} = [1, \gamma_3] \Delta_{i,k}$,
   (ii) if $|\delta f_{i,k}| > \frac{\mu_2}{p} \delta m_k$ then $\Delta_{i,k+1} \in [\gamma_1, \gamma_2] \Delta_{i,k}$,
   (iii) otherwise $\Delta_{i,k+1} = \Delta_{i,k}$.

It has been proved in [3] that the above criterion reduces at least one trust region radius on a bad iteration. It is unclear why the authors needed to make the distinction between meaningful and negligible elements. They perhaps wished to have a less restrictive criterion to classify negligible elements as good. But their treatment of meaningful elements (see Figure 4.3) has a natural concession towards elements with a small $\delta m_{i,k}$ — since it always classifies a neighborhood of $(\delta m_{i,k}, \delta f_{i,k}) = (0, 0)$ as 'good'.

New separation criteria. We have come up with three sorts of separation criteria for a structured trust region. Two of them correspond to the pictures in Figures 4.2 and 4.3, while a third is a combination of the two. In this subsection we demonstrate that the sort of stalling described above cannot take place cannot take place for any of these criteria. To prevent cycling, no increases in trust region radii on a bad iteration are allowed.

For ease of description, we adopt the following conventions for the criteria we propose. Let $0 < \gamma_1 \le \gamma_2 < 1 \le \gamma_3 \le \gamma_4$, $0 < \mu_1 \le \mu_2 < 1$. We assign each iteration a parameter τ_k. If $r_k \ge \mu_2$ then $\tau_k = 2$, if $\mu_1 \le r_k < \mu_2$ then $\tau_k = 1$, and otherwise $\tau_k = 0$. We then assign each element a parameter $\tau_{i,k}^1 \in \{0, 1, 2\}$. For each of the three criteria we describe, there is a different method to assign $\tau_{i,k}^1$. We then calculate $\tau_{i,k} = \tau_k + \tau_{i,k}^1$, and use the following method to update the sizes of the elemental trust regions:

If $\tau_{i,k} = 4$ then $\Delta_{i,k+1} = [1, \gamma_4] \Delta_{i,k}$,
if $\tau_{i,k} = 3$ then $\Delta_{i,k+1} = [1, \gamma_3] \Delta_{i,k}$,
if $\tau_{i,k} = 2$ then $\Delta_{i,k+1} = \Delta_{i,k}$,

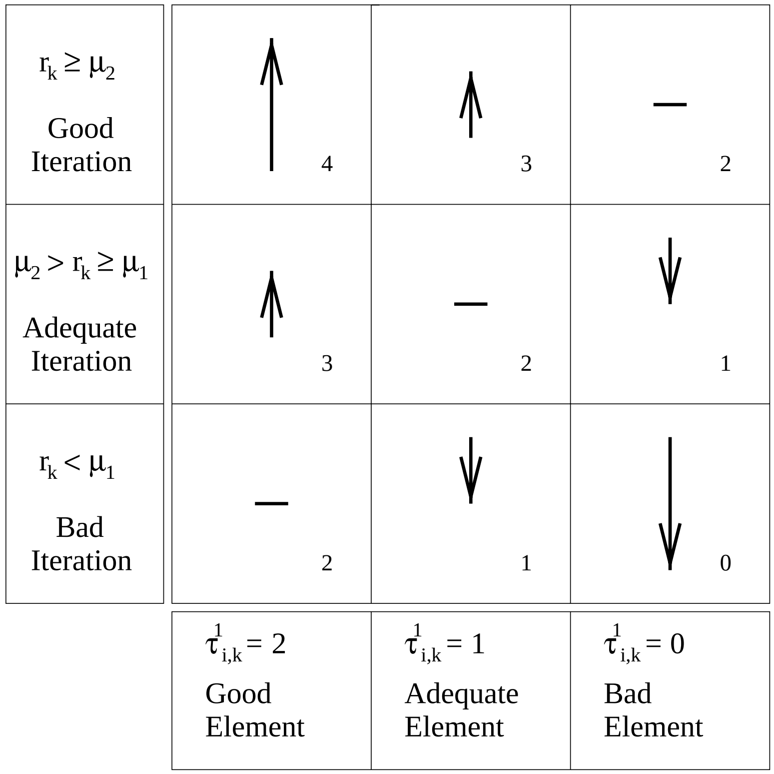

FIG. 4.4. *The general scheme for change in $\Delta_{i,k}$, given the type of iteration (depending on the match between δf_k and δm_k), and the type of element (depending on the match between $\delta f_{i,k}$ and $\delta m_{i,k}$).*

if $\tau_{i,k} = 1$ then $\Delta_{i,k+1} = [\gamma_2, 1]$,
if $\tau_{i,k} = 0$ then $\Delta_{i,k+1} = [\gamma_1, \gamma_2]\Delta_{i,k}$.

See Figure 4.4 for a pictorial representation of how the trust region radii are updated. The arrows show whether $\Delta_{i,k}$ would get decreased or increased, and the number in the bottom right corner of each box shows the value of $\tau_{i,k}$. The τ_k's correspond to good, adequate and bad iterations, and the $\tau^1_{i,k}$'s correspond to good, adequate and bad elements.

We also define the following terms:

$$\begin{aligned} \delta m_+ &:= \sum_{(i:\delta m_{i,k}>0)} \delta m_{i,k}, \\ \delta m_- &:= \sum_{(i:\delta m_{i,k}<0)} \delta m_{i,k}, \qquad (4.1) \\ \rho &:= \delta m_- / \delta m_+. \end{aligned}$$

Note that ρ is negative. Now we can state the criteria.

4.1. **First sloped criterion.**

Let $\alpha_1 = \frac{1}{2}[\mu_1(1+\rho) + \sqrt{\mu_1^2(1+\rho)^2 - 4\rho}]$,
and $\alpha_2 = \frac{1}{2}[\mu_2(1+\rho) + \sqrt{\mu_2^2(1+\rho)^2 - 4\rho}]$.
If $\delta m_{i,k} \geq 0$,
  if $r_{i,k} \geq \alpha_2$ *then* $\tau^1_{i,k} = 2$,
  if $r_{i,k} < \alpha_1$ *then* $\tau^1_{i,k} = 0$,
  else $\tau^1_{i,k} = 1$.
If $\delta m_{i,k} < 0$,
  if $r_{i,k} \leq 1/\alpha_2$ *then* $\tau^1_{i,k} = 2$,
  if $r_{i,k} > 1/\alpha_1$ *then* $\tau^1_{i,k} = 0$,
  else $\tau^1_{i,k} = 1$.

The second criterion has a different way of calculating α_1 and α_2, and is otherwise almost exactly the same as the previous one. We call it the second sloped criterion.

4.2. **Second sloped criterion.**

Let $\alpha_1 = \frac{\mu_1-(2-\mu_1)\rho}{1-\rho}$, *and* $\alpha_2 = \frac{\mu_2-(2-\mu_2)\rho}{1-\rho}$.
If $\delta m_{i,k} \geq 0$,
if $r_{i,k} \geq \alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $r_{i,k} < \alpha_1$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.
If $\delta m_{i,k} < 0$,
if $r_{i,k} \leq 2 - \alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $r_{i,k} > 2 - \alpha_1$ *then* $\tau^1_{i,k} = 0$,
Else $\tau^1_{i,k} = 1$.

Notice that $\mu_1 \leq \alpha_1 \leq 1$ and $\mu_2 \leq \alpha_2 \leq 1$. This is easy to check for the second sloped criterion. For the first, note that α_1 is the positive root of

$$\alpha^2 - \mu_1(1+\rho)\alpha + \rho = 0. \tag{4.2}$$

By considering the values of the left-hand-side at 0, μ_1 and 1, we see that $\mu_1 \leq \alpha_1 \leq 1$. The same holds for subscript 2. In both criteria α_1 and α_2 are monotonic functions of ρ that approach 1 as ρ approaches -1. They approach μ_1 and μ_2 as ρ approaches 0. So we make the criterion to distinguish good elements more strict when there is more cancellation, compensating for it.

Rewriting the criterion for the unstructured trust region method $\delta f_k/\delta m_k > \mu$ as $\frac{\delta m_k - \delta f_k}{\delta m_k} < (1-\mu)$, motivates the next criterion (corresponding to Figure 4.3). We want the error between the change in the function value and the change in the model value to be small relative to the change in the model value. We are tempted to extend it to check the elemental fits in the following way: $\frac{\delta m_{i,k} - \delta f_{i,k}}{\delta m_k} < (1-\mu)/p$, where p is the number of elements. Notice that this update criterion is a part of the update criterion used in [3]. We call this the *parallel criterion*, since the separating line in Figure 4.3 is parallel to the 45-degree line that represents $\delta f_{i,k} = \delta m_{i,k}$.

4.3. **Parallel criterion.**

Let $0 < \mu_1 \leq \mu_2 < 1$.
If $\delta f_{i,k} \geq \delta m_{i,k} - \frac{(1-\mu_2)}{p}\delta m_k$ *then* $\tau^1_{i,k} = 2$,
if $\delta f_{i,k} < \delta m_{i,k} - \frac{(1-\mu_1)}{p}\delta m_k$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.

Finally, we have combined criteria to allow more flexibility in the first and second sloped criteria.

4.4. **First combined criterion.**

Let $\eta_1 = -(1-\mu_1)\rho$, $\eta_2 = -(1-\mu_2)\rho$, $\mu_1' = \mu_1 + \eta_1$, $\mu_2' = \mu_2 + \eta_2$. *Let* α_1, α_2 *be defined as in the first sloped criterion above, except that we replace* μ_1, μ_2 *by* μ_1', μ_2'.
If $\delta m_{i,k} \geq 0$,
if $\delta f_{i,k} \geq \delta m_{i,k} - \frac{\eta_2}{p}\delta m_k$ *or if* $r_{i,k} \geq \alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $\delta f_{i,k} < \delta m_{i,k} - \frac{\eta_1}{p}\delta m_k$ *and if* $r_{i,k} < \alpha_1$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.
If $\delta m_{i,k} < 0$,
if $\delta f_{i,k} \geq \delta m_{i,k} - \frac{\eta_2}{p}\delta m_k$ *or if* $r_{i,k} \leq 1/\alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $\delta f_{i,k} < \delta m_{i,k} - \frac{\eta_1}{p}\delta m_k$ *and if* $r_{i,k} > 1/\alpha_1$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.

4.5. **Second combined criterion.**

Let $\eta_1 = -(1-\mu_1)\rho$, $\eta_2 = -(1-\mu_2)\rho$, $\mu_1' = \mu_1 + \eta_1$, $\mu_2' = \mu_2 + \eta_2$. *Let* α_1, α_2 *be defined as in the second sloped criterion above, except that we replace* μ_1, μ_2 *by* μ_1', μ_2'.
If $\delta m_{i,k} \geq 0$,
if $\delta f_{i,k} \geq \delta m_{i,k} - \frac{\eta_2}{p}\delta m_k$ *or if* $r_{i,k} \geq \alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $\delta f_{i,k} < \delta m_{i,k} - \frac{\eta_1}{p}\delta m_k$ *and if* $r_{i,k} < \alpha_1$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.
If $\delta m_{i,k} < 0$,
if $\delta f_{i,k} \geq \delta m_{i,k} - \frac{\eta_2}{p}\delta m_k$ *or if* $r_{i,k} \leq 2 - \alpha_2$ *then* $\tau^1_{i,k} = 2$,
if $\delta f_{i,k} < \delta m_{i,k} - \frac{\eta_1}{p}\delta m_k$ *and if* $r_{i,k} > 2 - \alpha_1$ *then* $\tau^1_{i,k} = 0$,
else $\tau^1_{i,k} = 1$.

Notice that when $\rho = 0$, both η_1 and η_2 are 0. So $\mu_1' = \mu_1$ and $\mu_2' = \mu_2$. In other words, the above criteria then reduce to the first and second sloped criteria.

When $\rho = -1$, we have $\eta_1 = (1 - \mu_1)$ and $\eta_2 = (1 - \mu_2)$, while μ_1' and μ_2' are both 1. And so, in this case, the above criteria both reduce to the parallel one.

Thus, the criteria reduce to that corresponding to the unstructured trust region method whenever there is no cancellation among $\delta m_{i,k}$'s, and they become more strict (approaching the parallel criterion) for an element with a large $\delta m_{i,k}$, as the amount of cancellation increases.

An analysis of the new criteria. The proof of a lemma that shows that the new separation criteria do satisfy our expectations.

LEMMA 4.6. *For all the criteria described above, if all $\tau_{i,k}^1 \geq 1$ then $\tau_k \geq 1$. And if all $\tau_{i,k}^1 \leq 1$, then $\tau_k \leq 1$.*

Proof. We prove only the first assertion. The proof of the second assertion is similar.
(i) For the first sloped criterion:
If all $\tau_{i,k}^1 \geq 1$, then from (4.1),

$$\delta f_k = \sum_{i=1}^{p} \delta f_{i,k} \geq \alpha_1 \delta m_+ + \frac{1}{\alpha_1} \delta m_-. \tag{4.3}$$

But α_1 is a root of (4.2):

$$\alpha_1^2 - \mu_1 (1 + \frac{\delta m_-}{\delta m_+}) \alpha_1 + \frac{\delta m_-}{\delta m_+} = 0.$$

Since $0 < \mu_1 \leq \alpha_1 \leq 1$ and $\delta m_+ > 0$, α_1 satisfies

$$\alpha_1 \delta m_+ - \mu_1 (\delta m_+ + \delta m_-) + \frac{1}{\alpha_1} \delta m_- = 0,$$

or,

$$\alpha_1 \delta m_+ + \frac{1}{\alpha_1} \delta m_- = \mu_1 \delta m_k.$$

So from (4.3),

$$\delta f_k \geq \mu_1 \delta m_k, \tag{4.4}$$

which is the same as $\tau_k \geq 1$.
(ii) For the second sloped criterion:
We see that α_1 is the solution to

$$\alpha_1 \delta m_+ + (2 - \alpha_1) \delta m_- = \mu_1 \delta m_k.$$

and just as in (4.3), if none of the elements is bad,

$$\sum_{i=1}^{p} \delta f_{i,k} \geq \alpha_1 \delta m_+ + (2 - \alpha_1) \delta m_-.$$

Combining these, once again we get (4.4), which is the same as $\tau_k \geq 1$.
(iii) For the parallel criterion:
If none of the elements is bad,

$$\sum_{i=1}^{p} \delta f_{i,k} \geq \sum_{i=1}^{p} (\delta m_{i,k} - \frac{(1 - \mu_1)}{p} \delta m_k).$$

So, $\delta f_k \geq \delta m_k - (1 - \mu_1) \delta m_k$, and (4.4) holds again.
(iv) For the first combined criterion:
We define the following sets: $\mathcal{I}_+ := \{i : \delta m_{i,k} \geq 0\}$, $\mathcal{I}_- := \{i : \delta m_{i,k} < 0\}$, $\mathcal{I}_1 := \{i \in \mathcal{I}_+ : \delta f_{i,k} < \alpha_1 \delta m_{i,k}\}$, $\mathcal{I}_2 := \{i \in \mathcal{I}_- : \delta f_{i,k} < \frac{1}{\alpha_1} \delta m_{i,k}\}$. Now if all the elements are adequate or good,

$$\delta f_k = \sum_{i=1}^{p} \delta f_{i,k} \geq \sum_{i \in \mathcal{I}_+} \alpha_1 \delta m_{i,k} + \sum_{i \in \mathcal{I}_-} \frac{1}{\alpha_1} \delta m_{i,k} +$$
$$\sum_{i \in \mathcal{I}_1} ((1 - \alpha_1) \delta m_{i,k} - \frac{\eta_1}{p} \delta m_k) + \sum_{i \in \mathcal{I}_2} ((1 - \frac{1}{\alpha_1}) \delta m_{i,k} - \frac{\eta_1}{p} \delta m_k).$$

But $\sum_{i\in\mathcal{I}_+}(\alpha_1\delta m_{i,k}) = \alpha_1\delta m_+$, and $\sum_{i\in\mathcal{I}_-}(\frac{1}{\alpha_1}\delta m_{i,k}) = \frac{1}{\alpha_1}\delta m_-$. Also, $(1-\alpha_1)\delta m_{i,k}$ is nonnegative for $i \in \mathcal{I}_1$ and $(1-\frac{1}{\alpha_1})\delta m_{i,k}$ is nonnegative for $i \in \mathcal{I}_2$. Using the same argument that led to (4.4) from (4.3),

$$\delta f_k \geq \mu_1' \delta m_k - \eta_1 \delta m_k = \mu_1 \delta m_k.$$

(v) For the second combined criterion:
The proof is similar to that of part (iv). □

In the last two sections we looked at the two major issues that any structured trust region approach must address: first, a need to address the relationship between the subproblem minimizer s_k, and directions in which $m_k(s)$ dominates over higher-order terms in the Taylor series of $f_(x_k+s)$; and second, ensuring that the trust region update mechanism satisfies our expectation that (a) the algorithm does not cycle or stall, (b) the trust region radius of at least one element be decreased on a bad iteration, and (c) the trust region radius of at least one element is a candidate for increase on a good iteration.

We then looked at separation criteria used in the past to update the trust region sizes, and then proposed some more of our own, showing how they satisfy the conditions (a), (b), and (c). We show convergence of our algorithm using only the parallel separation criterion.

5. First Order Convergence Analysis. The aim of our first order convergence analysis for the convex-constrained problem (P), is to show that $\lim_{k\to\infty}\alpha_k = 0$.

We make Assumptions 2.1–2.10, 2.9, as discussed before. We also assume that the parallel separation criterion is used. We show that $\lim_{k\to\infty}\alpha_k = 0$.

We begin with a technical lemma that establishes a lower bound on the accuracy of the model.

LEMMA 5.1. *If Assumption 2.4 holds, then there exists a constant $L \geq 1$ such that for each $k = 0,1,2,\ldots$, $|\delta f_k - \delta m_k| \leq L\beta_k\|s_k\|^2$, and for each $i = 1,\ldots,p$, $|\delta f_{i,k} - \delta m_{i,k}| \leq L\beta_k\|s_{i,k}\|^2$, where $s_{i,k}$ is the projection of s_k onto the range space R_i.*

Proof.

$$\begin{aligned}
|\delta f_k - \delta m_k| &= |f(x_k) - f_(x_k+s_k) - m(0) + m(s_k)| \\
&= |g_k^T s_k + \frac{1}{2}\|s_k\|^2\omega(f,x_k,s_k) - g_k^T s_k - \frac{1}{2}\|s_k\|^2\omega(m_k,0,s_k)| \\
&\leq \frac{1}{2}\|s_k\|^2(|\omega(f,x_k,s_k)| + |\omega(m_k,0,s_k)|) \\
&\leq \frac{1}{2}(L_h + \beta_k)\|s_k\|^2 \\
&\leq \frac{1}{2}(L_h + 1)\beta_k\|s_k\|^2 \\
&\leq L\beta_k\|s_k\|^2,
\end{aligned}$$

where $L := \frac{1}{2}(L_h+1)$ and $L_h \leq \chi_H$ is an upper bound on the generalized Rayleigh quotient of f for any choice of x and s.

The proof for the elemental differences $|\delta f_{i,k} - \delta m_{i,k}| \leq L\beta_k\|s_{i,k}\|^2$ is similar. □

The next lemma shows that the step size wont become too small for points away from a critical point. A sequence of three theorems completes the analysis.

LEMMA 5.2. *Consider a sequence of iterates generated by the algorithm and assume that there exists a constant $\epsilon > 0$ such that $\alpha_k \geq \epsilon$ for all k. Then for sufficiently small ϵ, $\Delta_{\min,k} \geq \frac{c_1}{\beta_k}$, where $c_1 = \gamma_1\min(1,\epsilon,\frac{\kappa\epsilon\gamma_1(1-\mu_2)}{Lp})$, for all k.*

Proof. We can suppose ϵ small enough so that $\Delta_{i,0} \geq \frac{c_1}{\beta_0}$ and we satisfy the lemma when $k = 0$. The rest of the proof is by contradiction. Therefore, assume that $\Delta_{\min,k}$ becomes smaller than $\frac{c_1}{\beta_k}$ for the first time on iteration k. Let $\Delta_{\min,k}$ be the trust region radius of the ith element. Since $\Delta_{i,k} < \frac{c_1}{\beta_k}$ for the first time, $\Delta_{i,k-1} \geq \frac{c_1}{\beta_{k-1}}$ so that $\Delta_{i,k-1} > \Delta_{i,k}$ (since $\{\beta_k\}$ is a non-decreasing sequence); similarly, $\Delta_{\min,k-1} > \Delta_{\min,k}$. We will show that $\Delta_{i,k-1} > \Delta_{i,k}$ could not have been possible.

We have $\Delta_{\min,k-1} \leq \Delta_{i,k-1} \leq \frac{\Delta_{\min,k}}{\gamma_1} < \frac{c_1}{\beta_k\gamma_1} \leq \frac{\epsilon}{\beta_k} \leq \frac{\epsilon}{\beta_{k-1}} \leq \frac{\alpha_{k-1}}{\beta_{k-1}}$. This also implies that $\Delta_{\min,k-1} < \frac{c_1}{\beta_k\gamma_1} \leq \frac{1}{\beta_k} \leq 1$. We substitute this into the sufficient decrease condition (3.1), to get $\delta m_{k-1} \geq \kappa\epsilon\Delta_{\min,k-1} \geq \kappa\epsilon\Delta_{\min,k} \geq \kappa\epsilon\gamma_1\Delta_{i,k-1}$.

Now from Lemma 5.1 we have $\delta m_{i,k-1} - \delta f_{i,k-1} \leq L\beta_{k-1}\|s_{i,k-1}\|^2 \leq L\beta_{k-1}\Delta_{i,k-1}^2$ $\leq \frac{L\beta_{k-1}\Delta_{i,k}\delta m_{k-1}}{\gamma_1^2\kappa\epsilon} \leq \frac{(1-\mu_2)}{p}\delta m_{k-1}$. Therefore the ith trust region radius could not have been reduced. ∎

THEOREM 5.3. *For the sequence of iterates generated by the algorithm,*

$$\liminf_{k\to\infty} \alpha_k = 0.$$

Proof. Assume, to obtain a contradiction, that there exists $\epsilon > 0$ such that $\alpha_k > \epsilon$ for all k, and suppose ϵ is small enough so that Lemma 5.2 holds and $\epsilon < 1$. To prove our result, we will try to contradict the assumption that $\sum_{k=1}^{\infty}\frac{1}{\beta_k} = \infty$ by breaking up the sum over specific subsequences of k. Let S denote the index set of successful iterations (where $\delta f_k/\delta m_k \geq \mu_1$) generated by the algorithm. Then $\sum_{k\in S}\delta f_k \geq \mu_1\sum_{k\in S}\delta m_k \geq \mu_1\kappa\epsilon\sum_{k\in S}\min(\frac{\epsilon}{\beta_k}, \Delta_{\min,k}, 1) \geq \mu_1\kappa\epsilon\min(\epsilon, c_1)\sum_{k\in S}\frac{1}{\beta_k}$, applying the sufficient decrease condition (3.1) and the result of Lemma 5.2. So, from the assumption that $f(x)$ is bounded below, we have that $\sum_{k\in S}\frac{1}{\beta_k} < \infty$.

Now let r be an integer such that $\gamma_4\gamma_2^{r-1} < 1$. Define $n_k = |S \cap \{1,\ldots,k\}|$, the number of successful iterations up to iteration $k \geq 1$. Define $\mathcal{F}_1 = \{k : k \leq rn_k\}$ and $\mathcal{F}_2 = \{k : k > rn_k\}$. First we show that $\sum_{k\in\mathcal{F}_1}\frac{1}{\beta_k}$ is finite. If it has only finitely many terms, its convergence is obvious. Otherwise, we may assume that $\mathcal{F}_1$ has an infinite number of elements and then we construct another subsequence $\mathcal{F}_3$ of indices in S in ascending order, with each index repeated r times. Since each $k \in S$ contributes at most r terms, each at least k, to the sequence $\mathcal{F}_1$, the jth term of $\mathcal{F}_3$ is no greater than the jth term of $\mathcal{F}_1$. This and the monotonicity of the sequence $\{\beta_k\}$ give us that $\sum_{k\in\mathcal{F}_1}\frac{1}{\beta_k} \leq \sum_{k\in\mathcal{F}_3}\frac{1}{\beta_k} = r\sum_{k\in S}\frac{1}{\beta_k} < \infty$.

Now we show that $\sum_{k\in\mathcal{F}_2}\frac{1}{\beta_k}$ is finite. We can immediately see that $\Delta_{\min,k} \leq \gamma_4^{n_k}\gamma_2^{k-n_k}\Delta_{\max,0}$. Using the result of Lemma 5.2, we have $\sum_{k\in\mathcal{F}_2}\frac{1}{\beta_k} \leq$
$\frac{\Delta_{\max,0}}{c_1}\sum_{k\in\mathcal{F}_2}(\gamma_4^{n_k}\gamma_2^{k-n_k}) \leq \frac{\Delta_{\max,0}}{c_1}\sum_{k\in\mathcal{F}_2}(\gamma_4\gamma_2^{(r-1)})^{\frac{k}{r}} < \infty$.

Therefore the sum $\sum_{k=0}^{\infty}\frac{1}{\beta_k} = \sum_{k\in\mathcal{F}_1}\frac{1}{\beta_k} + \sum_{k\in\mathcal{F}_2}\frac{1}{\beta_k} < \infty$, which contradicts our assumption. ∎

Here is an example to understand the relationship between $\mathcal{F}_1$ and $\mathcal{F}_3$. Suppose that the successful iterations are $k = (1,4,5,10,\ldots)$ and suppose $r = 2$; then $rn_k =$ $(2,2,2,4,6,6,6,6,6,8,\ldots)$, $\mathcal{F}_1 = (1,2,4,5,6,\ldots)$, and $\mathcal{F}_3 = (1,1,4,4,5,5,\ldots)$. Notice that $r = 2$, that $k = 1,5$ contributed 2 terms each to $\mathcal{F}_1$, and that the other successful k's each contributed fewer than 2 terms.

THEOREM 5.4. *If the algorithm generates an infinite sequence of successful iterates, then* $\lim_{k\in S}\alpha_k = 0$, *where S denotes the sequence of successful iterations.*

Proof. Once again, the proof is by contradiction. Assume that there exists $\epsilon_1 \in (0,1)$ and a subsequence $\{q_j\}$ of successful iteration indices such that, for all j, $\alpha_{q_j} \geq \epsilon_1$. Let $\epsilon_2 \in (0,\epsilon_1)$. Theorem 5.3 guarantees the existence of another subsequence $\{l_j\}$ such that $\alpha_k \geq \epsilon_2$ for $q_j \leq k < l_j$ and $\alpha_{l_j} < \epsilon_2$. We now look at the subsequence whose indices are in $K = \{k : k \in S, q_j \leq k < l_j\}$. For $k \in K$ we have, from (3.2) and the fact that iterations in K are successful, that $\delta f_k \geq \mu_1\kappa'\epsilon_2\|s_k\|$. From this we have:

$$\begin{aligned}
\|x_{q_j} - x_{l_j}\| &\leq \sum_{k\in K}\|x_{k+1} - x_k\| \\
&= \sum_{k\in K}\|s_k\| \\
&\leq \frac{1}{\mu_1\epsilon_2\kappa'}\sum_{k\in K}(f(x_k) - f(x_{k+1})) \\
&= \frac{1}{\mu_1\epsilon_2\kappa'}(f(x_{q_j}) - f(x_{l_j})). \qquad (5.1)
\end{aligned}$$

But Assumption 5.1 implies that the right-hand side of (5.1) converges to zero as j tends to infinity. Hence, by Assumption 2.9 on the uniform continuity of α_k, $|\alpha_{q_j} - \alpha_{l_j}| \leq \frac{1}{2}(\epsilon_1 - \epsilon_2)$ for j sufficiently large. Thus, $\alpha_{q_j} \leq \alpha_{l_j} + \frac{1}{2}(\epsilon_1 - \epsilon_2) \leq \frac{1}{2}(\epsilon_1 + \epsilon_2) < \epsilon_1$, which contradicts our original assumption. ∎

THEOREM 5.5. *If the set of successful iterations (i.e., iterations where $x_{k+1} = x_k + s_k$) generated by the algorithm is finite, then all its iterates x_k are equal to some x_* for k large enough, and x_* is critical.*

Proof. From the algorithm, a finite number of successful iterations means that x_k is unchanged for k large enough, and that $x_* = x_j$, where $j-1$ is the index of the last successful iteration. Now if $\alpha_j > 0$, we can apply the result of Theorem 5.3 to get a contradiction. Hence $\alpha(x_*) = \alpha_j = 0$. □

6. Second Order Convergence for the Unconstrained Case. The unconstrained minimization problem $(P1)$ is as follows

$$(P1) \qquad \min_{x \in \Re^n} f(x),$$

where $f : \Re^n \to \Re$ is a partially separable function.

Here the structured subproblem takes the following form:

$$(SP1) \qquad \begin{aligned} &\min m_k(s) = g_k^T s + \frac{1}{2} s^T B_k s \\ &\|P_{R_i}(s)\| \le \Delta_{i,k}, \quad i = 1, \dots, p. \end{aligned}$$

We use the Assumptions 2.1–2.4, with $X = \Re^n$ for our second order convergence analysis.

Our second order analysis replaces Assumption 2.10 by the stronger Assumption 6.1, as below:

Assumption 6.1. *There exists a positive constant $\chi_B \ge 1$ such that $\|B_k\| \le \chi_B$ and $\|B_{i,k}\| \le \chi_B$, $i = 1, \dots, p$, for all k.*

We begin with a lemma corresponding to Lemma 5.1, needed for second order convergence.

LEMMA 6.2. *If Assumptions 2.4 and 6.1 hold, $|\delta f_k - \delta m_k| \le L\|s_k\|^2$ and $|\delta f_{i,k} - \delta m_{i,k}| \le L\|s_{i,k}\|^2$, for all $i = 1, \dots, p$ and all k, where $L := \frac{1}{2}(L_h + \chi_B) \ge 1$, where χ_B is an upper bound on $\|B_k\|$ and $\|B_{i,k}\|$.*

Proof. The proof is similar to that of Lemma 5.1. □

We need to further strengthen our assumptions about the second derivative.

Assumption 6.3. $B_k = \nabla^2 f(x_k)$ *and* $B_{i,k} = \nabla^2 f_i(x_k)$, $i = 1, \dots, p$.

Since we continue to assume that all the second derivatives are bounded from the first order convergence theory, Assumption 6.1 about the boundedness of B_k automatically applies (replacing the weaker Assumption 2.10). In addition, we assume:

Assumption 6.4. $\nabla^2 f_{i,k}$ *is Lipschitz continuous with a constant L_c for all $i = 1, \dots, p$.*

Secondly, there are changes in the conditions we need the step to satisfy. Note that in the sufficient decrease condition (3.1), we can now replace α_k by $\|g_k\|$:

$$\delta m_k \ge \kappa \|g_k\| \min\left(\frac{\|g_k\|}{\beta_k}, \Delta_{\min,k}, 1\right). \tag{6.1}$$

The shorter-step condition (3.2) is tightened to the following pair of conditions, ensuring sufficient decrease when there is a direction of negative curvature.

$$\begin{aligned} \delta m_k &\ge \kappa' \|g_k\| \|s_k\| \\ \delta m_k &\ge -\kappa' \lambda_k \|s_k\|^2, \end{aligned} \tag{6.2}$$

where κ' is a small positive constant whose value depends on the method used to find an approximate solution to the subproblem in step 2 of the algorithm, and λ_k denotes the minimum eigenvalue of $\nabla^2 f(x)$. Also, for (6.2) we have used the same constant κ' as in (3.2). If we had a solution method that satisfied the two conditions for two different constants, we would simply choose the minimum of the two constants to be κ' for all of our results to hold. We now show that conditions (6.1) and (6.2) are satisfiable simultaneously.

LEMMA 6.5. *There exists a step s_k within the trust region that satisfies conditions (6.1) and (6.2) simultaneously for κ and κ' at most $\frac{1}{2}$, such that $\|s_k\| \geq \Delta_{\min,k}$ whenever $-\lambda_k \Delta_{\min,k} \geq \|g_k\|$.*

Proof.

Case 1. When $-\lambda_k \Delta_{\min,k} \geq \|g_k\|$ the following step s_k is taken: let q_k be a unit length eigenvector corresponding to λ_k such that $q_k^T g_k \leq 0$, without loss of generality. Let s_k be the step that minimizes $m_k(s)$ in the two-dimensional subspace spanned by g_k and q_k subject to $\|s_k\| \leq \Delta_{\min,k}$. We will show that each of the three conditions in the statement of the lemma hold.

Now $\delta m_k(s_k) \geq \delta m_k(q_k \Delta_{\min,k}) \geq -q_k^T g_k \Delta_{\min,k} - \frac{1}{2}\lambda_k \Delta_{\min,k}^2$ (since $q_k^T \nabla^2 f(x_k) q_k = \lambda_k$) $\geq -\frac{1}{2}\lambda_k \Delta_{\min,k}^2$ (from our assumption that $q_k^T g_k < 0$). Since $\|s_k\| \leq \Delta_{\min,k}$, s_k satisfies the second part of (6.2) for $\kappa' \leq 1/2$.

Also, from $\delta m_k(s_k) \geq -\frac{1}{2}\lambda_k \Delta_{\min,k}^2$ and $-\lambda_k \Delta_{\min,k} \geq \|g_k\|$, we have $\delta m_k(s_k) \geq \kappa' \|g_k\| \Delta_{\min,k}$ (for $\kappa' \leq 1/2$) $\geq \kappa' \|g_k\| \|s_k\|$ (given that $\|s_k\| \leq \Delta_{\min,k}$), satisfying the first part of (6.2).

Because the problem is unconstrained, notice that the step from Lemma 3.2 satisfies (6.1), lies along the direction $-g_k$ and has norm less than $\Delta_{\min,k}$. It is thus a feasible alternative to s_k. Since s_k must give at least as much decrease as this feasible alternative, it satisfies (6.1) as well.

Since s_k lies on the boundary of the trust region, we see that $\|s_k\| \geq \Delta_{\min,k}$ is also satisfied.

Case 2. When $-\lambda_k \Delta_{\min,k} < \|g_k\|$ we take what we call the *reduced* Cauchy step, defining it to be minimizer of the model along the negative gradient direction within the trust region and with the further restriction that $\|s_k\| \leq \Delta_{\min,k}$.

With this choice of step (the same as the step in Lemma 3.2 applied to the unconstrained case, as in Case 1, if $\Delta_{\min,k} \geq 1$) it is possible to satisfy (6.1) and the first part of (6.2).

If $\lambda_k \geq 0$, the second part of (6.2) is trivially satisfied. If not, from the first half of (6.2) and $-\lambda_k \Delta_{\min,k} < \|g_k\|$, we have $\delta m_k \geq \kappa' \|g_k\| \|s_k\| \geq -\kappa' \lambda_k \Delta_{\min,k} \|s_k\| \geq -\kappa' \lambda_k \|s_k\|^2$, satisfying the second part of (6.2). ☐

Finally we state a technical lemma proved in [10].

LEMMA 6.6. *Let x_* be an isolated limit point of a sequence $\{x_k\}$ in $\Re^n$. If $\{x_k\}$ does not converge then there is a subsequence $\{x_{l_j}\}$ of successful iterations which converges to x_* and an $\epsilon > 0$ such that*

$$\|x_{l_j+1} - x_{l_j}\| \geq \epsilon.$$

The next theorem contains the main result in this section.

THEOREM 6.7. *Let s_k satisfy conditions (6.1) and (6.2), with $\|s_k\| \geq \Delta_{\min,k}$ when $-\lambda_k \Delta_{\min,k} \geq \|g_k\|$, at each iteration. If $\{x_k\}$ is the sequence generated by Algorithm 2.13 with the parallel separation criterion, then the following are true:*

(a) The sequence $\{g_k\}$ converges to zero.

(b) If $\{x_k\}$ is bounded then there is a limit point x_* with $\nabla^2 f(x_*)$ positive semidefinite.

(c) If x_* is an isolated limit point of $\{x_k\}$ then $\nabla^2 f(x_*)$ is positive semidefinite.

(d) If $\nabla^2 f(x_*)$ is nonsingular for some limit point x_* of $\{x_k\}$, then $\nabla^2 f(x_*)$ is positive definite, $\{x_k\}$ converges to x_*, all iterations are eventually successful, and $\{\Delta_{\min,k}\}$ is bounded away from zero.

Proof.

(a) This follows from the first order theory in the last section.

(b) The proof is by contradiction. Assume that there is a $\epsilon_1 > 0$ such that for all k large enough, say $k \geq k_0$, $-\lambda_k \geq \epsilon_1$. We will show that this contradicts the assumption that f is bounded. We begin by showing that $\Delta_{\min,k} \geq c_2$ for all $k \geq k_0$ (also by contradiction), where $c_2 := \frac{(1-\mu_2)\gamma_1 \kappa' \epsilon_1}{p L_c}$. We choose ϵ_1 to be small enough that $\Delta_{\min,k_0} \geq c_2$. Now suppose $\Delta_{\min,k} < c_2$ for the first time on the kth iteration, $k \geq k_0$. Consider the ith element, where $\Delta_{i,k} = \Delta_{\min,k}$. We have $\Delta_{i,k-1} \leq \Delta_{\min,k}/\gamma_1$. From (6.2) and the mean-value theorem,

$$\begin{aligned} \frac{|\delta f_{i,k-1} - \delta m_{i,k-1}|}{\delta m_{k-1}} &\leq \frac{\|s_{i,k-1}\|^2 \max_{\xi \in [0,1]} \|\nabla^2 f_i(x_{k-1} + \xi s_{k-1}) - \nabla^2 f_i(x_{k-1})\|}{-\kappa' \lambda_k \|s_{k-1}\|^2} \\ &\leq \frac{L_c \|s_{i,k-1}\|^3}{\kappa' \epsilon_1 \|s_{k-1}\|^2} \quad \text{(by Lipschitz continuity)} \end{aligned}$$

$$
\begin{aligned}
&\le \frac{L_c \Delta_{i,k-1}}{\kappa' \epsilon_1} \qquad (\text{since} \|s_{i,k-1}\| \le \|\Delta_{i,k-1}\|) \\
&\le \frac{L_c \Delta_{\min,k}}{\gamma_1 \kappa' \epsilon_1} \\
&\le \frac{1-\mu_2}{p}.
\end{aligned}
$$

Therefore by the parallel separation criterion the ith element is not a candidate for reduction of its trust region, or $\Delta_{\min,k} < c_2$ is not possible. If we had only a finite number of steps where $r_k \ge \mu_1$, $\Delta_{\min,k}$ would converge to zero. Since it cannot, we must have an infinite number of successful steps, where $\Delta_{\min,k}$ and $-\lambda_k$ are bounded away from zero for all sufficiently large k. Since $\|g_k\|$ converges to zero, for large enough k we have $-\lambda_k \Delta_{\min,k} > \|g_k\|$, and so $\|s_k\| \ge \Delta_{\min,k}$. Thus, for all subsequent successful steps $\delta f_k \ge \mu_1 \delta m_k \ge \mu_1 \kappa' \epsilon_1 \|s_k\|^2 \ge \mu_1 \kappa' \epsilon_1 \Delta_{\min,k}^2$, which contradicts the boundedness of f.

(c) If $\{x_k\}$ converges to x_*, the result follows from (b). If $\{x_k\}$ does not converge then Lemma 6.6 applies, and yields a subsequence $\{x_{l_j}\}$ converging to x_* with $\|x_{l_j+1} - x_{l_j}\| \ge \epsilon$. Notice that the sequence $\{l_j\}$ contains only successful iterations. But $\delta m_{l_j} \ge -\kappa' \hat{\lambda}_{l_j} \|x_{l_j+1} - x_{l_j}\|^2$ implies that $\delta f_{l_j} \ge \mu_1 \delta m_{l_j} \ge -\kappa' \mu_1 \hat{\lambda}_{l_j} \epsilon^2$, where $\hat{\lambda}_{l_j} := \min(\lambda_{l_j}, 0)$. Since f is bounded, $\{\hat{\lambda}_{l_j}\}$ must converge to zero and so $\nabla^2 f(x_*)$ is positive semidefinite.

(d) If $\nabla^2 f(x_*)$ is nonsingular for a limit point x_*, then x_* is an isolated limit point by (a). Hence $\nabla^2 f(x_*)$ is positive definite from parts (b) and (c). To prove the rest we go to the following variant of this theorem.

□

THEOREM 6.8. *Let $\{x_k\}$ be the sequence generated by the algorithm under the same conditions on the step as in Theorem 6.7. If x_* is a limit point of $\{x_k\}$ with $\nabla^2 f(x_*)$ positive definite then $\{x_k\}$ converges to x_*, all iterations are eventually successful, and $\{\Delta_{\min,k}\}$ is bounded away from zero.*

Proof. We first prove that $\{x_k\}$ converges to x_*. Choose $\epsilon > 0$ and $h > 0$ so that the minimum eigenvalue of $\nabla^2 f(x)$ is at least ϵ for $\|x - x_*\| \le h$. Since the change in the value of the model δm_k is nonnegative, we have $\|g_k\|\|s_k\| \ge -g_k^T s_k \ge \frac{1}{2} s_k^T \nabla^2 f(x_k) s_k \ge \frac{1}{2}\lambda_k \|s_k\|^2$, where λ_k is the minimum eigenvalue of $\nabla^2 f(x_k)$. Thus $\|x_k - x_*\| \le h$ implies that

$$
\frac{1}{2}\epsilon \|s_k\| \le \|g_k\|. \tag{6.3}
$$

Theorem 6.7 guarantees that $\{g_k\}$ converges to zero, and thus there is an index k_1 for which $\|g_k\| \le \frac{1}{4}\epsilon h$ for all $k \ge k_1$. Hence, (6.3) shows that if $\|x_k - x_*\| \le \frac{1}{2}h$ for $k \ge k_1$, then $\|x_{k+1} - x_*\| \le h$.

Since $g_* = 0$, from the Taylor series expansion of f about x_* we have

$$
f(x) - f(x_*) = (x - x_*)^T \nabla^2 f(x_* + \xi x)(x - x_*)/2,
$$

where $0 \le \xi \le 1$. This implies that for $\frac{1}{2}h < \|x - x_*\| \le h$, $\nabla^2 f(x_* + \xi x)$ is positive definite and $f(x) - f(x_*) \ge \frac{1}{2}\epsilon \|x - x_*\|^2 > \frac{1}{8}\epsilon h^2$. Thus, there exists an index $k_2 > k_1$ such that $\|x_{k_2} - x_*\| \le h/2$ and $f(x_{k_2}) \le f(x_*) + \frac{1}{8}\epsilon h^2$. Applying (6.3) to x_{k_2} and x_{k_2+1}, we get $\|x_{k_2+1} - x_{k_2}\| \le h/2$. But then $\|x_{k_2+1} - x_*\| \le h$. Now $f(x_*) + \frac{1}{2}\epsilon \|x_{k_2+1} - x_*\|^2 \le f_(x_{k_2+1}) \le f(x_{k_2}) \le f(x_*) + \frac{1}{8}\epsilon h^2$, implying that $\|x_{k_2+1} - x_*\| \le h/2$.

Hence, $\|x_k - x_*\| \le h/2$ for $k \ge k_2$. But since h can be chosen arbitrarily small, $\{x_k\}$ converges to x_*.

We now prove that all iterations are successful. From (6.2) and (6.3), there exists an $\epsilon_1 > 0$ with $\delta m_k \ge \epsilon_1 \|s_k\|^2$ for all sufficiently large k. Here we can use an argument similar to that in part (b) of the last theorem to get $\frac{|\delta f_k - \delta m_k|}{\delta m_k} \le \frac{L_c \|s_k\|}{\epsilon_2}$, and hence that $\{|r_k - 1|\}$ converges to zero. Hence all iterations are eventually successful and $\{\Delta_{\min,k}\}$ is bounded away from zero. □

7. Conclusions. The unstructured trust region size has some intrinsic robustness that is forfeited for a relatively unreliable $\Delta_{\min,k}$ when we try to structure. Our algorithm introduces a condition to compensate for this. It is shown to converge in a first and second order sense under general and unrestrictive assumptions. However, our new condition on the step takes the place of a better trust region update mechanism.

8. Acknowledgement. This work was done as part of my Ph.D. at the School of Operations Research and Industrial Engineering at Cornell University, Ithaca, New York. I thank Prof. Michael J. Todd, who was my Ph.D. advisor there.

REFERENCES

[1] A. Bouaricha, and J. J. Moré, *Impact of partial separability on large-scale optimization*, Comput. Optim. Appl., 7 (1997), pp. 27–40.

[2] T. F. Coleman, and Y. Li, *An interior trust region approach for nonlinear minimization subject to bounds*, SIAM J. Optim., 6 (1996), pp. 418–445.

[3] A. R. Conn, N. Gould, A. Sartenaer, and Ph. L. Toint, *Convergence properties of minimization algorithms for convex constraints using a structured trust region*, SIAM J. Optim., 6 (1996), pp. 1059–1086.

[4] A. R. Conn, N. 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 J. Optim., 3 (1993), pp. 164–221.

[5] A. R. Conn, Nick Gould, and Ph. L. Toint, LANCELOT*: A Fortran Package for Large-Scale Nonlinear Optimization*, Springer-Verlag, 1992.

[6] R. Fletcher, *Practical Methods of Optimization*, Second ed., Practical Methods of Optimization, John Wiley, New York, 1987.

[7] A. Griewank, and Ph. L. Toint, *On the existence of convex decompositions of partially separable functions*, Math. Programming, 28 (1984), pp. 25–49.

[8] M. Lescrenier, *Partially separable optimization and parallel computing*, Ann. Oper. Res., 14 (1988), pp. 213–224.

[9] J. J. Moré, *Recent developments in algorithms and software for trust region methods*, in Mathematical Programming: The State of the Art, A. Bachem, M. Grötschel and B. Korte, eds., Springer-Verlag, 1982.

[10] J. J. Moré, and D. Sorensen, *Computing a trust region step*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 553–572.

[11] M. J. D. Powell, *On the global convergence of trust region algorithms for unconstrained minimization*, Math. Programming, 29 (1984), pp. 297–303.

[12] Ph. L. Toint, *Global convergence of a class of trust region methods for non-convex minimization in Hilbert space*, IMA J. Numer. Anal., 8 (1988), pp. 231–252.

[13] Ph. L. Toint, *Global convergence of the partitioned BFGS algorithm for convex partially separable optimization*, Math. Programming, 36 (1986), pp. 290–306.