A Global Convergence Theory for the Celis-Dennis-Tapia Trust Region Algorithm for Constrained Optimization[1]

by

Mahmoud El Alem[2]

Technical Report 88-10, September 1988

Revised May, 1989

---

[1]Research sponsored by SDIO/IST/ARO, AFOSR 85-0243, and DOE DEFG05-86ER25017.

[2]Department of Mathematical Sciences, Rice University, Houston, TX 77251-1892.

A Global Convergence Theory for the Celis-Dennis-Tapia Trust Region Algorithm for Constrained Optimization[1]

by

Mahmoud El-Alem[2]

Abstract. A global convergence theory for a class of trust-region algorithms for solving the equality constrained optimization problem is presented. This theory is sufficiently general that it holds for any algorithm that generates steps that give at least a fraction of Cauchy decrease in the quadratic model of the constraints and uses the augmented Lagrangian as a merit function. This theory is used to establish global convergence of the 1984 Celis-Dennis-Tapia algorithm with a different scheme for updating the penalty parameter. The behavior of the penalty parameter is also discussed.

Key words. Trust region, constrained optimization, Celis-Dennis-Tapia algorithm, globally convergent algorithm, nonlinear programming.

Abbreviated title. A convergence theory for a trust-region algorithm.

AMS(MOS) subject classification. 65K05, 49D37

---

[1] Research sponsored by SDIO/IST/ARO, AFOSR 85-0243, and DOE DEFG05-86ER 25017.

[2] Department of Mathematical Science, Rice University, Houston, TX 77251-1892.

1. Introduction

In this research, we consider the following equality constrained optimization problem

$$\begin{aligned} &\text{minimize } f(x), \\ &\text{subject to } h_i(x) = 0 \qquad i=1,\ldots,m \end{aligned} \tag{EQ}$$

where f and h_i are assumed to be smooth nonlinear functions defined from R^n into R. A more detailed list of assumptions will be explicitly presented later. We will denote by $h(x)$ the vector whose components are $h_i(x)$ $i=1,\ldots,m$. It is convenient to introduce the Lagrangian function $l : R^n \times R^m \to R$ associated with problem (EQ). It is the function:

$$l(x, \lambda) = f(x) + \lambda^T h(x) \tag{1.1}$$

where $\lambda = (\lambda_1, \ldots, \lambda_m)^T$ is the Lagrange multiplier vector. The augmented Lagrangian function $\Phi : R^n \times R^m \times R \to R$ associated with problem (EQ) is the function:

$$\Phi(x,\lambda;r) = l(x, \lambda) + r \; || h(x) ||_2^2 . \tag{1.2}$$

where r is the penalty parameter.

Stating necessary optimality conditions in terms of the Lagrangian function requires a constraint qualification. A satisfactory but somewhat restrictive constraint qualification is the regularity assumption: that is, the vectors $\nabla h_i(x)$ $i=1,\ldots,m$ are linearly independent at the solution. We use the notation $\nabla h(x)$ for the matrix whose columns are $\nabla h_i(x)$ $i=1,\ldots,m$.

The first-order necessary conditions, or Kuhn-Tucker conditions, for a point $x_* \in R^n$ to be a solution of problem (EQ) are that x_* be a feasible point (*i.e.* $h(x_*) = 0$), and that there exists a Lagrange multiplier λ_* such that $\nabla_x l(x_*,\lambda_*) = 0$. Equivalent first-order necessary conditions are that x_* be a feasible point and that $P(x_*)\,\nabla f(x_*) = 0$ where $P(x)$ is the projection onto the null space of $\nabla h(x)^T$, *i.e.*,

$$P(x) = I - \nabla h(x)(\nabla h(x)^T \nabla h(x))^{-1} \nabla h(x)^T \ . \tag{1.3}$$

For a detailed discussion of optimality conditions, see, for example, Fiacco and McCormick (1968).

Problem (EQ) is often solved by the Successive Quadratic Programming (SQP) algorithm. Namely, at the k^{th} iteration, the step is computed by solving the following quadratic programming subproblem:

$$\begin{aligned} &\text{minimize} \quad \nabla_x l(x_k,\lambda_k)^T s + \frac{1}{2}\, s^T B_k\, s \\ &\text{subject to} \quad h(x_k) + \nabla h(x_k)^T s \ = \ 0 \ , \end{aligned} \tag{QP}$$

where B_k is the Hessian of the Lagrangian at (x_k,λ_k) or an approximation to it.

The local convergence analysis for the SQP algorithm has been well established [for example see Tapia (1977),(1978)]. The area of global convergence is currently receiving much attention.

Trust region approaches for unconstrained optimization have proven to be very successful both theoretically and practically. The most natural way to introduce the trust region idea into constrained optimization is to add a constraint which restricts the size of the step in problem (QP). That is, at the k^{th} iteration we solve the following trust-region quadratic programming subproblem:

$$\begin{aligned} &\text{minimize} \quad \nabla_x l(x_k,\lambda_k)^T s + \frac{1}{2}\, s^T B_k\, s \\ &\text{subject to} \quad h(x_k) + \nabla h(x_k)^T s \ = \ 0 \\ &\qquad\qquad || \, s \, ||_2 \ \le \ \Delta_k \ . \end{aligned} \tag{TRQP}$$

However, this approach may lead to inconsistent constraints if $h(x_k) \neq 0$. To overcome this difficulty, two main approaches have been introduced. The first approach is to relax the constraints by considering the following subproblem:

$$\begin{aligned} \text{minimize} \quad & \nabla_x l(x_k,\lambda_k)^T s + \frac{1}{2} s^T B_k s \\ \text{subject to} \quad & \alpha\, h(x_k) + \nabla h(x_k)^T s = 0 \\ & \| s \|_2 \le \Delta_k , \end{aligned}$$

where $0 \le \alpha \le 1$. This approach was first introduced by Vardi (1985). It was also used by Byrd, Schnabel, and Shultz (1987). This approach always leads to a feasible subproblem if α is chosen properly. However, this approach suffers from the disadvantage that the step depends on the unknown parameter α for which there is no clear way of choosing.

The second approach is to add the trust-region constraint to a somewhat different problem. At the k^{th} iteration the step is taken to be the one that minimizes the quadratic model of the Lagrangian subject to some required decrease in $\| h(x_k) + \nabla h(x_k)^T s \|_2$. This idea was first introduced by Celis, Dennis, and Tapia (1985). At each iteration the step is computed by solving the following subproblem:

$$\begin{aligned} \text{minimize} \quad & \nabla_x l(x_k,\lambda_k)^T s + \frac{1}{2} s^T B_k s \\ \text{subject to} \quad & \| h(x_k) + \nabla h(x_k)^T s \|_2 \le \theta_k \\ & \| s \|_2 \le \Delta_k \end{aligned} \qquad \text{(CDT)}$$

where θ_k and Δ_k are positive constants.

Celis, Dennis and Tapia (1985) chose θ_k to be $\| h(x_k) + \nabla h(x_k)^T s_k^{cp} \|_2$, where $s_k^{cp} = -\alpha_k \nabla h(x_k)\, h(x_k)$ is the step to the Cauchy point, *i.e.*, the minimizer in the trust region $\{s : \| s \|_2 \le \Delta_k\}$ of $\| h(x_k) + \nabla h(x_k)^T s \|_2$ along its negative gradient direction. That is, the Celis-Dennis-Tapia step is chosen from the set of steps from x_k that are inside the trust region and give at least as much descent on the 2-norm of the residual of the linearized constraints as does the Cauchy step.

In 1986, Powell and Yuan introduced another way of choosing θ_k. They chose it to be any number that satisfies

$$\theta_k = \min [\ || h(x_k) + \nabla h(x_k)^T s ||_2 : || s ||_2 \leq \sigma \Delta_k \ ] ,$$

for some $0 \leq \sigma \leq 1$. [See Powell and Yuan (1986-b)]

A more general choice of θ_k was suggested by Celis, Dennis, Martinez, Tapia, and Williamson (1989). They chose it to be

$$\theta_k = (1-\tau) \ || h(x_k) ||_2 + \tau \ || h(x_k) + \nabla h(x_k)^T s_k^{cp} ||_2 , \tag{1.4}$$

for some $0 < \tau \leq 1$. Where s_k^{cp} is the step to the Cauchy point and is defined above.

This latter choice of θ_k enforces a fraction of Cauchy decrease on the 2-norm of the linearized constraints in the CDT subproblem. Powell and Yuan's choice of θ_k enforces a fraction of optimal decrease [see Celis, Dennis, Martinez, Tapia, and Williamson (1989)]. The choice given by Celis, Dennis and Tapia (1985) gives at least as much decrease in the 2-norm of the linearized constraints as does the Cauchy step s_k^{cp}. Other choices of θ_k are suggested in Celis, Dennis, Martinez, Tapia, and Williamson (1989). We are going to consider only the choice of θ_k given by (1.4). This choice is appropriate since it insures considerable freedom in the subproblem feasible set, allowing the minimization of the subproblem objective function to pull the iterate toward the optimal point for problem (EQ) rather than progressing too fast toward nonlinear feasibility at the expense of optimality. Our numerical experiments reinforce the validity of this choice. [See Dennis, El-Alem, and Tapia (1989)]

In this paper we consider a trust-region algorithm for solving the equality constrained optimization problem. This algorithm is a variant of the 1984 Celis-Dennis-Tapia trust-region algorithm in that it uses a different scheme for updating the penalty parameter.

The remainder of this paper is organized as follows. In Section 2, we describe in detail the trust-region subproblem that will be considered and the way of computing the trial steps. A scheme for updating the radius of the trust region is presented together with a discussion about the criteria for accepting or rejecting the trial steps. A scheme for updating the penalty parameter is also presented. In Section 3, we present the algorithm. In Section 4, we state the standard assumptions under which our global convergence theory is established. In Section 5, we state our main global convergence results. Sections 6,7 and 8 are devoted to the analysis of the global behavior of our algorithm. Section 9 contains concluding remarks.

Notation:

The trial step at the k^{th} iteration is denoted by $\hat{s}_k$ and its associated Lagrange multiplier by $\hat{\Delta\lambda}_k$. If the step is accepted it will be denoted by s_k and its associated Lagrange multiplier by $\Delta\lambda_k$.

The decomposition of the step $\hat{s}_k$ into a tangential and a normal component is considered. These components are denoted by $\hat{s}_k^t$ and $\hat{s}_k^n$ respectively and are defined by $\hat{s}_k^t = P(x_k)\,\hat{s}_k$ and $\hat{s}_k^n = Q(x_k)\,\hat{s}_k$, where $P(x_k) = I - \nabla h(x_k)(\nabla h(x_k)^T \nabla h(x_k))^{-1} \nabla h(x_k)^T$ and $Q(x_k) = I - P(x_k)$.

The expressions $\nabla^2 h(x_k)\,\Delta\lambda$ and $\nabla^2 h(x_k)\,h(x_k)$ are used to denote $\sum_{i=1}^{m} \Delta\lambda_i\, \nabla^2 h_i(x_k)$ and $\sum_{i=1}^{m} h_i(x_k)\, \nabla^2 h_i(x_k)$ respectively. The matrix B_k denotes $\nabla_x^2 l(x_k,\lambda_k)$ or an approximation to it.

Subscripted values of functions denote evaluation at a particular point. For example f_k means $f(x_k)$.

2. Description of The Algorithm

The algorithm is iterative. At each iteration a trial step $\hat{s}_k$ is obtained by solving a model problem. At the k^{th} iteration, we try to update the estimate of the solution x_k to an improved estimate x_{k+1}. To do this, the step s_k^{QP} and the multiplier $\Delta\lambda_k^{QP}$ are obtained by solving the QP subproblem (see Section 1). If they exist and if s_k^{QP} lies inside the trust region, *i.e.* if $|| s_k^{QP} || \leq \Delta_k$, then we set $\hat{s}_k = s_k^{QP}$ and $\hat{\Delta\lambda}_k = \Delta\lambda_k^{QP}$. Otherwise, the CDT subproblem will be solved (see Section 1). On the other hand, if x_k is feasible, then we solve the TRQP subproblem (see Section 1). This can be summarized in the following scheme:

SCHEME 2.1 Computing the Trial Step

Solve (QP) to get s_k^{QP} and $\Delta\lambda_k^{QP}$ (see Section 1)

If $|| s_k^{QP} ||_2 \leq \Delta_k$

then $\hat{s}_k = s_k^{QP}$

$$\hat{\Delta\lambda}_k = \Delta\lambda_k^{QP}.$$

Else, if x_k is feasible

then solve (TRQP) (see Section 1)

Set $\hat{s}_k = s_k^{TRQP}$

$$\hat{\Delta\lambda}_k = -(\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T [\nabla_x l_k + B_k s_k^{TRQP}].$$

Else, solve (CDT) (see Section 1)

Set $\hat{s}_k = s_k^{CDT}$

$$\hat{\Delta\lambda}_k = -(\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T [\nabla_x l_k + B_k s_k^{CDT}].$$

When x_k is feasible, then $\theta_k = 0$, and the CDT subproblem reduces to the TRQP subproblem. This shows a continuity in the behavior of x_k on switching between the CDT and the TRQP subproblems.

If the trial step is either the TRQP step or the CDT step, our choice of the multiplier requires solving the following linear system in the least-squares sense

$$\nabla h_k \, \hat{\Delta\lambda}_k = -(\nabla l_k + B_k \, \hat{s}_k) .$$

Powell and Yuan (1986-a and 1986-b) have used as a multiplier update formula the least-square multiplier estimate. Using this formula, the following linear system has to be solved in the least-squares sense

$$\nabla h(x_k+\hat{s}_k) \, \hat{\Delta\lambda}_k = -(\nabla f(x_k+\hat{s}_k) - \nabla h(x_k+\hat{s}_k) \, \lambda_k) .$$

Powell and Yuan's choice of $\hat{\Delta\lambda}_k$ is more expensive since it requires a factorization of $\nabla h(x_k+\hat{s}_k)$ at each trial step. Our choice of $\hat{\Delta\lambda}_k$ requires the factorization only when the algorithm moves to a new point after finding an acceptable step. On the other hand, when the SQP step is taken, our multiplier is obtained with no extra cost because it is the SQP multiplier.

Let $\hat{s}_k$ be the step computed by the algorithm and $\hat{\Delta\lambda}_k$ be the corresponding Lagrange multiplier step, we test whether the point $(x_k+\hat{s}_k , \lambda_k+\hat{\Delta\lambda}_k)$ is a better approximation to the solution (x_* , λ_*). In order to do this, we use, as a merit function, the augmented Lagrangian (1.2).

The actual reduction in the merit function in going from (x_k,λ_k) to $(x_k+\hat{s}_k , \lambda_k+\hat{\Delta\lambda}_k)$ is given by

$$Ared_k = \Phi(x_k,\lambda_k;r_k) - \Phi(x_k+\hat{s}_k,\lambda_k+\hat{\Delta\lambda}_k;r_k) .$$

We can write

$$\begin{aligned} Ared_k &= l(x_k,\lambda_k) - l(x_k+\hat{s}_k,\lambda_k) - \hat{\Delta\lambda}_k^T \, h(x_k+\hat{s}_k) \\ &+ r_k \, [\, \| h(x_k) \|_2^2 - \| h(x_k+\hat{s}_k) \|_2^2 \,]. \end{aligned} \tag{2.1}$$

The calculation of the step $\hat{s}_k$ is based on a quadratic approximation of the Lagrangian function and a linear approximation to the constraints. Now by using the same approximations, we can compute the predicted reduction

$$Pred_k = \Phi(x_k,\lambda_k;r_k) - \Psi(x_k,\hat{s}_k,\lambda_k,\hat{\Delta\lambda}_k;r_k),$$

where $\Psi(x_k,\hat{s}_k,\lambda_k,\hat{\Delta\lambda}_k;r_k)$ is an approximation to $\Phi(x_k+\hat{s}_k,\lambda_k+\hat{\Delta\lambda}_k;r_k)$ and is defined by

$$\begin{aligned}\Psi(x_k,\hat{s}_k,\lambda_k,\hat{\Delta\lambda}_k;r_k) &= l(x_k,\lambda_k) + \nabla_x l(x_k,\lambda_k)^T \hat{s}_k + \frac{1}{2}\hat{s}_k{}^T B_k \hat{s}_k \\ &+ \hat{\Delta\lambda}_k^T[\, h(x_k)+\nabla h(x_k)^T \hat{s}_k \,] \\ &+ r_k \;||\, h(x_k) + \nabla h(x_k)^T \hat{s}_k \,||_2^2 .\end{aligned}$$

Hence,

$$\begin{aligned}Pred_k &= -\nabla_x l(x_k,\lambda_k)^T \hat{s}_k - \frac{1}{2}\hat{s}_k{}^T B_k \hat{s}_k - \hat{\Delta\lambda}_k^T(\, h(x_k)+\nabla h(x_k)^T \hat{s}_k \,) \\ &+ r_k \,[\; ||\, h(x_k)\,||_2^2 - ||\, h(x_k) + \nabla h(x_k)^T \hat{s}_k \,||_2^2 \,] .\end{aligned} \tag{2.2}$$

We accept the step and set $x_{k+1} = x_k+s_k$ and $\lambda_{k+1} = \lambda_k+\Delta\lambda_k$, if $\dfrac{Ared_k}{Pred_k} \geq \eta_1$ where $\eta_1 \epsilon (0,1)$ is a small fixed constant.

If the step is rejected, then we set $x_{k+1} = x_k$ and $\lambda_{k+1} = \lambda_k$ and decrease the radius of the trust region by picking $\Delta_{k+1} \epsilon [\, \alpha_1 \,||\, \hat{s}_k \,||_2 , \alpha_2 \,||\, \hat{s}_k \,||_2 \,]$, where $0 < \alpha_1 \leq \alpha_2 < 1$. [See Dennis and Schnabel (1983)].

If the step is accepted, then the trust-region radius is updated by comparing the value of $Ared_k$ with $Pred_k$. Namely, if $\eta_1 \leq \dfrac{Ared_k}{Pred_k} < \eta_2$ where $\eta_2 \epsilon (\eta_1,1)$, then the radius of the trust region is updated by the rule: $\Delta_{k+1} = \min [\, \Delta_k , \alpha_3 \,||\, s_k \,||_2 \,]$ where $\alpha_3 > 1$. However, if $\dfrac{Ared_k}{Pred_k} \geq \eta_2$, then we increase the radius of the trust region by setting

$\Delta_{k+1} = \min[\, \Delta_* \, , \max(\, \Delta_k \, , \alpha_3 \, || \, s_k \, ||_2 \,) \,]$, where Δ_* is a positive constant. This can be summarized in the following scheme:

SCHEME 2.2 Testing the Step and Updating the Trust Region Radius

If $\dfrac{Ared_k}{Pred_k} < \eta_1$,

then set $x_{k+1} = x_k$,

$\lambda_{k+1} = \lambda_k$,

$$\Delta_{k+1} \in [\, \alpha_1 \, || \, \hat{s}_k \, ||_2 \, , \alpha_2 \, || \, \hat{s}_k \, ||_2 \,] \, . \tag{2.3}$$

Else, if $\eta_1 \leq \dfrac{Ared_k}{Pred_k} < \eta_2$

then set $x_{k+1} = x_k + s_k$,

$\lambda_{k+1} = \lambda_k + \Delta\lambda_k$,

$\Delta_{k+1} = \min\, [\Delta_k \, , \alpha_3 \, || \, s_k \, ||_2 \,]$.

Else, set $x_{k+1} = x_k + s_k$,

$\lambda_{k+1} = \lambda_k + \Delta\lambda_k$,

$\Delta_{k+1} = \min[\, \Delta_* \, , \max(\, \Delta_k \, , \alpha_3 \, || \, s_k \, ||_2 \,) \,]$.

Now, we describe our strategy for updating the penalty parameter r . Numerical experiments have suggested that efficient performance of the algorithm is linked to keeping the penalty parameter as small as possible. Our global convergence theory requires that the sequence $\{ r_k \}$ be nondecreasing and that the predicted reduction in the merit function at each iteration be at least as much as a fraction of Cauchy decrease in the 2-norm of the residual of the linearized constraints. The idea now is to keep the penalty parameter as small as possible, subject to satisfying these two conditions needed for our convergence theory. Hence, our strategy will be to start with $r = 1$ and increase it only when necessary to satisfy these two conditions. The following is the scheme that we use for updating the penalty parameter.

SCHEME 2.3 Updating the Penalty Parameter

If

$$Pred_k \geq \frac{r_{k-1}}{2}[\; || \, h_k \, ||_2^2 - || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,] \, ,$$

then set $\quad r_k = r_{k-1}$.

Else, set

$$r_k = 2 \, \frac{\nabla_x l_k^T \hat{s}_k + ½ \, \hat{s}_k^T B_k \hat{s}_k + \hat{\Delta\lambda}_k^T (h_k + \nabla h_k^T \hat{s}_k)}{|| \, h_k \, ||_2^2 - || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2} + \rho \, ,$$

where $\rho > 0$ is a small fixed constant.

The initial choice of the penalty parameter $r = 1$ is scale dependent. Here we assume that the constraints have been made to be well scaled compared to the objective function.

Finally, we discuss the strategy for updating the matrix B_k . If s_k is not an acceptable step, then set $B_{k+1} = B_k$, otherwise compute $B_{k+1} = \nabla^2 f_{k+1} + \nabla^2 g_{k+1} \lambda_{k+1}$ if the exact Hessian is used, or update B_k by some other update formula that satisfies the standard assumption 5. (See Section 4)

3. The Algorithm

The following represents the outline of the Algorithm. It differs from the 1984 Celis-Dennis-Tapia algorithm in its way of updating the penalty parameter in step 3 of the algorithm and in its way of updating the trust region radius in step 4.

Step 0 :

Set $x_0 \in R^n$, $B_0 \in R^{n\times n}$, $\lambda_0 \in R^m$, $r_{-1} = 1$, $\rho > 0$,

$0 < \alpha_1 \leq \alpha_2 < 1 < \alpha_3$, $0 < \eta_1 < \eta_2 < 1$,

$0 < \tau \leq 1$, $\epsilon > 0$, $\Delta_0 > 0$, and $k = 0$.

Step 1 :

If

$$|| P_k \nabla f_k ||_2 + || h_k ||_2 < \epsilon , \tag{3.1}$$

where P_k is defined by (1.3), stop.

Step 2 :

Compute $\hat{s}_k$ and $\hat{\Delta\lambda}_k$ according to Scheme 2.1 above.

Step 3 :

Update the penalty parameter according to Scheme 2.3 above.

Step 4 :

Test the step and update Δ_k according to Scheme 2.2 above.

Step 5 :

Update B_k as above.

Step 6 :

Set $k := k + 1$ and go to step 1.

4. The Standard Assumptions

In this section we state the assumptions under which we prove global convergence.

1) There exists a convex set $\Omega \in R^n$ such that, for all k, x_k and $x_k + \hat{s}_k \in \Omega$.

2) f and $h_i \in C^2(\Omega)$ $i=1,...,m$.

3) $\nabla h(x)$ has full column rank for all $x \in \Omega$.

4) $f(x)$, $h(x)$, $\nabla h(x)$, $\nabla f(x)$, $\nabla^2 f(x)$, $(\nabla h(x)^T \nabla h(x))^{-1}$ and each $\nabla^2 h_i(x)$, for $i=1,...,m$ are all uniformly bounded in norm in Ω.

5) The matrices $\{ B_k , \quad k=1,2,... \}$ have a uniform upper bound.

If Ω were a compact set assumption 4 would follow from continuity.

If the exact Hessian is used, assumption 5 is a strong one, since in most cases it requires that the Lagrange multiplier estimates be uniformly bounded. However, if an approximation to the Hessian of the Lagrangian is used, then any update formula that satisfies the standard assumption 5 can be used. For example, setting B_k be a fixed matrix for all k is a valid one. The question of how to use secant approximations of the Hessian of the Lagrangian in order to produce a more efficient algorithm is a research topic. (See Section 9). Typically, secant updates can be shown to satisfy the standard assumption 5 only as a by-product of the convergence analysis.

The same assumptions as our standard assumptions were used by Byrd, Schnabel, and Shultz (1987) and Powell and Yuan (1986-b).

5. The Global Convergence Theory

In this section we state the main results in our global convergence analysis in order to understand the motivation for the lemmas presented in Sections 6 and 7. These lemmas are necessary to the proofs of our main global convergence results presented in Section 8.

Section 6 is devoted to presenting all results that deal with decrease in the merit function. The behavior of the penalty parameter is discussed in Section 7.

Theorem 5.1

Under the standard assumptions, at any point (x_k , λ_k) generated by the algorithm, either the termination condition of the algorithm will be met or an acceptable step will be found. *i.e.* the condition $\dfrac{Ared_{k+j}}{Pred_{k+j}} \geq \eta_1$ will be satisfied for some j .

The proof of this theorem is given in Section 8. Theorem 5.1 shows that the algorithm is well defined in the sense that it always finds an acceptable step from any point that does not satisfy the termination criteria. From this theorem we see that the algorithm can not loop indefinitely without finding an acceptable step.

Now, we state our main global convergence result, Theorem 5.2.

Theorem 5.2

Under the standard assumptions, the algorithm produces iterates $\{ x_k \}$ which satisfy

$$\liminf_{k\to\infty} \left[\; || \, h_k \, ||_2 + \; || \, P_k \, \nabla f_k \, ||_2 \right] \; = \; 0$$

The proof of this theorem is presented in Section 8. It is well known that a point $x_* \in R^n$ is a stationary point if and only if $h(x_*) = 0$ and $P(x_*)\nabla f(x_*) = 0$. Theorem 5.2 shows that the algorithm will successfully terminate. It means that the Celis-Dennis-Tapia trust-region algorithm for equality constrained optimization generates at least a subsequence converging to a stationary point of the problem.

6. The Decrease in The Model

All results in this section deal with the reduction of the merit function and the predicted reduction of the model.

In the following lemma we use the fact that the step $\hat{s}_k$ is chosen to give at least as much decrease in the 2-norm of the linearized constraints as does the Cauchy step s_k^{cp}.

Lemma 6.1

There exist constants b_1 and b_2 independent of k such that at the k^{th} iteration the predicted decrease in the merit function given by the trial step satisfies

$$Pred_k \; \geq \; \frac{r_k}{2} \; \frac{|| \, h_k \, ||_2}{b_1} \min \left[\; \Delta_k \; , \; \frac{|| \, h_k \, ||_2}{b_2} \; \right] .$$

Proof

First we prove that

$$|| \, h_k \, ||_2^2 \; - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \; \geq \; \frac{|| \, h_k \, ||_2}{b_1} \min \left[\; \Delta_k \; , \; \frac{|| \, h_k \, ||_2}{b_2} \; \right] \qquad (6.1)$$

where b_1 and b_2 are constants independent of k .

When the TRQP step is used, inequality (6.1) is valid a fortiori.

Consider the case when either the QP step or the CDT step is used. From the way of computing the step $\hat{s}_k$ and using the fact that $|| h_k || \; || h_k + \nabla h_k^T s^{cp} ||_2 \leq || h_k ||_2^2$, we have

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \geq || h_k ||_2^2 - \theta_k^2$$

$$\geq \tau^2 [\, || h_k ||_2^2 - || h_k + \nabla h_k^T s_k^{cp} ||_2^2 \,] .$$

$$= \tau^2 [-2 \, h_k^T \, \nabla h_k^T s_k^{cp} - (s_k^{cp})^T \, \nabla h_k \, \nabla h_k^T \, s_k^{cp} \,] .$$

Note that when the QP step is used, we have

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 = || h_k ||_2^2 \geq || h_k ||_2^2 - \theta_k^2 .$$

From the definition of s_k^{cp} , we have $s_k^{cp} = -\alpha_k \, \nabla h_k \, h_k$, where α_k is defined by

$$\alpha_k = \frac{\Delta_k}{|| \nabla h_k \, h_k ||_2} \quad \text{if} \quad \frac{|| \nabla h_k \, h_k ||_2^3}{|| \nabla h_k^T \, \nabla h_k \, h_k ||_2^2} \geq \Delta_k , \tag{6.2-a}$$

otherwise,

$$\alpha_k = \frac{|| \nabla h_k \, h_k ||_2^2}{|| \nabla h_k^T \, \nabla h_k \, h_k ||_2^2} . \tag{6.2-b}$$

Consider the first case. *i.e.*, the case when $s_k^{cp} = -\Delta_k \dfrac{\nabla h_k \, h_k}{|| \nabla h_k \, h_k ||_2}$. In this case, using $\dfrac{|| \nabla h_k \, h_k ||_2^3}{|| \nabla h_k^T \, \nabla h_k \, h_k ||_2^2} \geq \Delta_k$, we have

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \geq \tau^2 [\, 2 \, \Delta_k \, || \nabla h_k \, h_k ||_2 - \Delta_k^2 \frac{|| \nabla h_k^T \, \nabla h_k \, h_k ||_2^2}{|| \nabla h_k \, h_k ||_2^2} \,]$$

$$\geq \tau^2 [\, 2 \, \Delta_k \, || \nabla h_k \, h_k ||_2 - \Delta_k \, || \nabla h_k \, h_k ||_2 \,]$$

$$= \tau^2 \, \Delta_k \; || \nabla h_k \, h_k ||_2 \, . \tag{6.3}$$

Now, consider the second case. We have

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \geq \tau^2 \, [\, 2 \, \frac{|| \nabla h_k \, h_k ||_2^2}{|| \nabla h_k^T \nabla h_k \, h_k ||_2^2} \, || \nabla h_k \, h_k ||_2^2 - \frac{|| \nabla h_k \, h_k ||_2^4}{|| \nabla h_k^T \nabla h_k \, h_k ||_2^4} \, || \nabla h_k^T \nabla h_k \, h_k ||_2^2 \,] \, .$$

Hence,

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \geq \tau^2 \, \frac{|| \nabla h_k \, h_k ||_2^4}{|| \nabla h_k^T \nabla h_k \, h_k ||_2^2} \geq \tau^2 \, \frac{|| \nabla h_k \, h_k ||_2^2}{|| \nabla h_k \, \nabla h_k^T ||_2} \, .$$

From the last inequality and (6.3), we can write

$$|| h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \geq \tau^2 \; || \nabla h_k \, h_k ||_2 \min \, [\, \Delta_k \, , \, \frac{|| \nabla h_k \, h_k ||_2}{|| \nabla h_k \nabla h_k^T ||_2} \,] \, .$$

Now, using the standard assumption 3, we have

$$|| \nabla h_k \, h_k ||_2 \geq \frac{|| h_k ||_2}{|| (\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T ||_2} \, ,$$

The rest of the proof of (6.1) now follows from the standard assumption 4.

From the way of updating the penalty parameter r_k in step 3 of the algorithm, we have

$$Pred_k \geq \frac{r_k}{2} \, [\; || h_k ||_2^2 - || h_k + \nabla h_k^T \hat{s}_k ||_2^2 \,] \, .$$

The proof of the lemma follows immediately from (6.1) and the last inequality.

■

Lemma 6.1 shows that the way we update the penalty parameter ensures that the predicted reduction at each iteration will be at least as much as a fraction of Cauchy decrease in the merit function. This indicates compatibility of the step choice algorithm with the augmented Lagrangian merit function.

Lemma 6.2

Let $\hat{s}_k^t$ and $\hat{s}_k^n$ be the tangential and the normal components of the trial step $\hat{s}_k$ generated by the algorithm. Then, $\hat{s}_k^t$ satisfies

$$(\nabla l_k + B_k \hat{s}_k^n)^T \hat{s}_k^t \le -\frac{1}{2} \| P_k(\nabla l_k + B_k \hat{s}_k^n) \|_2 \min [\overline{\Delta}_k , \frac{\| P_k(\nabla l_k + B_k \hat{s}_k^n) \|_2}{2 \| B_k \|_2}] ,$$

where $\overline{\Delta}_k = \sqrt{\Delta_k^2 - \| \hat{s}_k^n \|_2^2}$.

Proof

The proof follows directly from Lemma 3.2 in Powell and Yuan (1986-b).

Lemma 6.3

For any x_k , $x_k+\hat{s}_k \in \Omega$, we have

$$| Ared_k - Pred_k | \le a_1 \| \hat{s}_k \|_2^2 + r_k [a_2 \| \hat{s}_k \|_2^3 + a_3 \| h_k \|_2 \| \hat{s}_k \|_2^2] ,$$

where a_1 , a_2 , a_3 are constants independent of k .

Proof

From (2.1), (2.2) and the Cauchy-Schwarz inequality, we can write:

$$\begin{aligned} | Ared_k - Pred_k | \le & \; | l(x_k,\lambda_k) + \nabla_x l(x_k,\lambda_k)^T \hat{s}_k + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k - l(x_k+\hat{s}_k , \lambda_k) | \\ & + | \hat{\Delta\lambda}_k^T [h_k + \nabla h_k^T \hat{s}_k - h(x_k+\hat{s}_k)] | \\ & + r_k | \| h_k + \nabla h_k^T \hat{s}_k \|_2^2 - \| h(x_k+\hat{s}_k) \|_2^2 | . \end{aligned}$$

Hence,

$$| Ared_k - Pred_k | \le \frac{1}{2} | \hat{s}_k^T [B_k - \nabla_x^2 l(x_k+\xi_1 \hat{s}_k,\lambda_k)] \hat{s}_k |$$

$$+ \frac{1}{2} \mid \hat{s}_k^T \, [\, \nabla^2 h(x_k+\xi_2 \hat{s}_k) \, \hat{\Delta\lambda}_k \,] \, \hat{s}_k \mid$$

$$+ \, r_k \mid \hat{s}_k^T [\, \nabla h_k \, \nabla h_k^T - \nabla h(x_k+\xi_3 \hat{s}_k) \nabla h^T (x_k+\xi_3 \hat{s}_k) \,] \, \hat{s}_k \mid$$

$$+ \, r_k \mid \hat{s}_k^T \, \nabla^2 h(x_k+\xi_3 \hat{s}_k) \, h(x_k+\xi_3 \hat{s}_k) \, \hat{s}_k \mid ,$$

for some $\xi_1 , \xi_2 , \xi_3 \in (0 , 1)$.

By using the standard assumptions 2, 4 and 5, the form of $\hat{\Delta\lambda}_k$, and the fact that $|| s_k ||_2 \leq \Delta_k$, we get

$$| \, Ared_k - Pred_k \, | \leq a_1 \, || \hat{s}_k ||_2^2 + a_2 \, r_k \, || \hat{s}_k ||_2^3 + a_3 \, r_k \, || \hat{s}_k ||_2^2 \, || h_k ||_2$$

which is the desired result. ■

The result we obtained in the last lemma does not depend on any property of the matrices $\{ B_k \}$ except that their norms have a uniform upper bound, and does not depend on any property of the steps except that they lie inside Ω.

Corollary 6.4

For any x_k , $x_k+\hat{s}_k \in \Omega$, we have

$$| \, Ared_k - Pred_k \, | \leq a_4 \, r_k \, || \hat{s}_k ||_2^2$$

where a_4 is a constant independent of k .

Proof

The proof follows immediately from the last lemma, the fact that $r_k \geq 1$, the fact that $|| s ||_2 \leq \Delta_*$, and the standard assumption 4. ■

Corollary 6.4 shows that, if the penalty parameter is bounded, our definition of predicted reduction implicitly gives an approximation to the merit function that is accurate to within the square of the steplength.

Lemma 6.5

If $\hat{s}_k$ is a trial step generated by the algorithm and $\hat{s}_k^t$ is its tangential component, then

$$(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t \leq 0 .$$

Proof

If $\hat{s}_k$ is the step obtained from the CDT subproblem, then

$$-(\nabla l_k + B_k \hat{s}_k) = \mu \hat{s}_k + \alpha \nabla h_k (h_k + \nabla h_k^T \hat{s}_k)$$

where $\mu , \alpha \geq 0$. See Celis, Dennis, and Tapia (1985). Now

$$-P_k (\nabla l_k + B_k \hat{s}_k) = \mu P_k \hat{s}_k + \alpha P_k [\nabla h_k (h_k + \nabla h_k^T \hat{s}_k)] = \mu \hat{s}_k^t .$$

Hence

$$-(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t = \mu || \hat{s}_k^t ||_2^2 \geq 0 .$$

Now, assume that the step is generated from the TRQP subproblem. Then $\hat{s}_k$ must satisfy

$$(B_k + \mu I)^T \hat{s}_k = -(\nabla l_k + \nabla h_k \hat{\Delta\lambda}_k)$$

where $\mu \geq 0$ with $\mu = 0$ if the step is generated from the QP subproblem, *i.e.* if the trust region constraint is not binding. By multiplying by P_k , we obtain

$$P_k(\nabla l_k + B_k \hat{s}_k) = -P_k \nabla h_k \hat{\Delta\lambda}_k - \mu P_k \hat{s}_k = -\mu \hat{s}_k^t .$$

Hence,

$$(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t = -\mu || \hat{s}_k^t ||_2^2 \leq 0 .$$

This implies that in all cases the lemma is true. ■

Lemma 6.6

There exists a constant b_3 such that, for all k , the normal component $\hat{s}_k^n$

satisfies

$$\| \hat{s}_k^n \|_2 \leq b_3 \| h_k \|_2 .$$

Proof

The proof follows directly from the standard assumptions 3 and 4. ■

Lemma 6.7

Let $\hat{s}_k$ be a step generated by the algorithm and let $\hat{s}_k^t$ and $\hat{s}_k^n$ be its tangential and normal components respectively. Let P_k be defined by (1.3), $\overline{\Delta}_k$ be as in Lemma 6.2 and $\bar{h}_k = \nabla h_k (\nabla h_k^T \nabla h_k)^{-1} h_k$, then

$$\begin{aligned} Pred_k \geq \frac{1}{4} \| P_k(\nabla l_k + B_k \hat{s}_k^n) \|_2 \min [\, \overline{\Delta}_k \, , \frac{\| P_k(\nabla l_k + B_k \hat{s}_k^n) \|_2}{2\, b_4}] \\ - b_5 \| \hat{s}_k \|_2 \; \| h_k \|_2 - | (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k | \\ + r_k [\; \| h_k \|_2^2 - \| h_k + \nabla h_k^T \hat{s}_k \|_2^2 \,] , \end{aligned} \tag{6.4}$$

where b_4 and b_5 are constants independent of k .

Proof

From the definition of $Pred_k$ and $\hat{\Delta\lambda}_k$, we can write

$$\begin{aligned} Pred_k &= -(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k \\ &+ (\nabla l_k + B_k \hat{s}_k)^T \nabla h_k \, (\nabla h_k^T \nabla h_k)^{-1} (h_k + \nabla h_k^T \hat{s}_k) \\ &+ r_k [\; \| h_k \|_2^2 - \| h_k + \nabla h_k^T \hat{s}_k \|_2^2 \,] . \end{aligned}$$

Now, since $\nabla h_k \, (\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T \hat{s}_k = \hat{s}_k^n$, we can write:

$$\begin{aligned} Pred_k &= -(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k \\ &+ (\nabla l_k + B_k \hat{s}_k)^T [\, \bar{h}_k + \hat{s}_k^n \,] \end{aligned}$$

$$+ \; r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,].$$

Using $\hat{s}_k - \hat{s}_k^n = \hat{s}_k^t$ and $-(\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t \geq -\frac{1}{2} (\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t$, we get

$$Pred_k \geq -\frac{1}{2} (\nabla l_k + B_k \hat{s}_k)^T \hat{s}_k^t + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k + (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k$$

$$+ \; r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,]$$

$$\geq -\frac{1}{2} (\nabla l_k + B_k \hat{s}_k^n)^T \hat{s}_k^t - \frac{1}{2} (\hat{s}_k^t)^T B_k \hat{s}_k^t + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k$$

$$+ (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k + r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,] \, ;$$

which can be written as

$$Pred_k \geq -\frac{1}{2} (\nabla l_k + B_k \; \hat{s}_k^n)^T \; \hat{s}_k^t + \frac{1}{2} (\hat{s}_k^t)^T B_k \hat{s}_k^n + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k^n$$

$$+ (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k + r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,] \, .$$

By using Lemma 6.2, we obtain

$$Pred_k \geq \frac{1}{4} \, || \, P_k(\nabla l_k + B_k \hat{s}_k^n) \, ||_2 \min \; [\; \bar{\Delta}_k \; , \; \frac{|| \, P_k(\nabla l_k + B_k \hat{s}_k^n) \, ||_2}{2 \, || \, B_k \, ||_2} \;]$$

$$+ \frac{1}{2} (\hat{s}_k^t)^T B_k \hat{s}_k^n + \frac{1}{2} \hat{s}_k^T B_k \hat{s}_k^n + (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k$$

$$+ \; r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,].$$

But by Lemma 6.6 and the standard assumption 5, we can write

$$Pred_k \geq \frac{1}{4} \, || \, P_k(\nabla l_k + B_k \hat{s}_k^n) \, ||_2 \min \; [\; \bar{\Delta}_k \; , \; \frac{|| \, P_k(\nabla l_k + B_k \hat{s}_k^n) \, ||_2}{2 b_4} \;]$$

$$- (b_4 b_3 \, || \, \hat{s}_k \, ||_2 \, || \, h_k \, ||_2) - | \, (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k \; |$$

$$+ \; r_k \; [\; || \, h_k \, ||_2^2 - \; || \, h_k + \nabla h_k^T \hat{s}_k \, ||_2^2 \,].$$

where b_4 is a constant such that $|| \, B_k \, ||_2 \leq b_4$. Now, by setting $b_5 = b_4 \, b_3$, we obtain the result. ■

In order to prove that the algorithm is making an improvement in the merit function, we have to prove that we will get a positive predicted reduction at each iteration. Toward this end we must prove that the positive quantities in (6.4) are greater than or equal to the absolute value of the negative quantities. If this is not the case, then the algorithm, according to Scheme 2.3, will increase the penalty parameter to ensure that this will be the case. First we need to derive an upper bound on the third quantity. The following lemma will give us this bound.

Lemma 6.8

Let $\bar{h}_k$ be as in Lemma 6.7, then there exist constants a_5 and a_6 such that

$$| (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k | \leq [a_5 || \hat{s}_k ||_2 + a_6 || s_{k-t_k} ||_2] || h_k ||_2$$

where s_{k-t_k} is the last acceptable step.

Proof

We have

$$Q_k(\nabla l_k + B_k \hat{s}_k) = Q_k \nabla f_k + Q_k \nabla h_k \lambda_k + Q_k B_k \hat{s}_k$$

Now

$$Q_k \nabla f_k = \nabla h_k (\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T \nabla f_k = - \nabla h_k \lambda_k^p$$

where $\lambda_k^p = - (\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T \nabla f_k$, and

$$Q_k \nabla h_k = \nabla h_k (\nabla h_k^T \nabla h_k)^{-1} \nabla h_k^T \nabla h_k = \nabla h_k .$$

Since s_{k-t_k} is the last acceptable step, then $x_{k-t_k+1} = x_k$ and $\lambda_{k-t_k+1} = \lambda_k$. We have

$$Q_k \nabla h_k \lambda_k = \nabla h_k \lambda_k = \nabla h_k \lambda_{k-t_k+1}$$

$$= - \nabla h_k [(\nabla h_{k-t_k}^T \nabla h_{k-t_k})^{-1} \nabla h_{k-t_k}^T (\nabla f_{k-t_k} + B_{k-t_k} s_{k-t_k})]$$

$$= \nabla h_k \, [\, \lambda^{\ell}_{k-t_k} - (\, \nabla h^T_{k-t_k} \nabla h_{k-t_k} \,)^{-1} \nabla h^T_{k-t_k} \, B_{k-t_k} \, s_{k-t_k} \,] \, .$$

This implies that

$$\begin{aligned} || \, Q_k(\nabla l_k + B_k \, \hat{s}_k) \, ||_2 \, \le \, & || \, \nabla h_k \, (\lambda^{\ell}_k - \lambda^{\ell}_{k-t_k}) \, ||_2 \\ & + b_1 \, || \, \nabla h_k \, ||_2 \, || \, B_{k-t_k} \, ||_2 \, || \, s_{k-t_k} \, ||_2 \\ & + \, || \, B_k \, ||_2 \, || \, \hat{s}_k \, ||_2 \, , \end{aligned} \tag{6.5}$$

where b_1 is as in Lemma 6.1. Now by using the standard assumptions, there exists a constant b_6, such that

$$\begin{aligned} || \, \nabla h_k \, (\lambda^{\ell}_k - \lambda^{\ell}_{k-t_k}) \, ||_2 \, & \le \, || \, \nabla h_k \, ||_2 \, || \, \lambda^{\ell}_k - \lambda^{\ell}_{k-t_k} \, ||_2 \\ & \le \, b_6 \, || \, x_k - x_{k-t_k} \, ||_2 \, , \end{aligned}$$

and since $x_k = x_{k-t_k+1}$, we have

$$\begin{aligned} || \, \nabla h_k \, (\lambda^{\ell}_k - \lambda^{\ell}_{k-t_k}) \, ||_2 \, & \le \, b_6 \, || \, x_{k-t_k+1} - x_{k-t_k} \, ||_2 \\ & = \, b_6 \, || \, s_{k-t_k} \, ||_2 \, . \end{aligned} \tag{6.6}$$

Substitute (6.6) in (6.5), and by using the standard assumptions 4 and 5, we obtain

$$|| \, Q_k(\nabla l_k + B_k \, \hat{s}_k) \, ||_2 \, \le \, b_7 \, || \, \hat{s}_k \, ||_2 + b_8 \, || \, s_{k-t_k} \, ||_2 \, . \tag{6.7}$$

where b_7 and b_8 are constants independent of k.

Since $Q_k \bar{h}_k = \bar{h}_k$, we have

$$\begin{aligned} | \, (\nabla l_k + B_k \hat{s}_k)^T \, \bar{h}_k \, | \, & = \, | \, [\, Q_k \, (\nabla l_k + B_k \hat{s}_k \,) \,]^T \, \bar{h}_k \, | \\ & \le \, || \, Q_k \, (\nabla l_k + B_k \hat{s}_k) \, ||_2 \, || \, \bar{h}_k \, ||_2 \, . \end{aligned}$$

Now, by using (6.7) and the fact that $|| \, \bar{h}_k \, ||_2 \le b_9 \, || \, h_k \, ||_2$, where $b_9 = \sup_{x \in \Omega} || \, \nabla h(x) (\nabla h(x)^T \nabla h(x))^{-1} \, ||_2$ the proof follows immediately. ■

The following lemma proves the important property that if $\| h_k \|_2$ is small enough, then the penalty parameter will not be increased in step 3 of the algorithm.

Lemma 6.9

Let k index an iteration at which the algorithm does not terminate. If $\| h_k \|_2 \leq c_1 \Delta_k$ where c_1 is a small constant that satisfies

$$c_1 \leq \min \left[\frac{\sqrt{3}}{2\, b_3} , \frac{\epsilon}{3\, \Delta_*} , \frac{\epsilon}{3\, b_5\, \Delta_*} , \frac{\epsilon}{48\,(a_5 + b_5 + a_6)\, \Delta_*} \min \left(1 , \frac{\epsilon}{3\, b_4\, \Delta_*} \right) \right] \quad (6.8)$$

where a_5 and a_6 are as in Lemma 6.8, b_3 is as in Lemma 6.6, b_4 and b_5 are as in Lemma 6.7, and Δ_* is the upper bound on the trust region radius, then

$$Pred_k \geq \frac{r_k}{2} \left[\| h_k \|_2^2 - \| h_k + \nabla h_k^T \hat{s}_k \|_2^2 \right] + \frac{1}{8} \| P_k(\nabla l_k + B_k\, \hat{s}_k^n) \|_2 \min \left[\frac{1}{2}\Delta_k , \frac{\| P_k(\nabla l_k + B_k\, \hat{s}_k^n) \|_2}{2 b_4} \right] .$$

Proof

If k is the index of an iteration at which the algorithm does not terminate, then

$$\| P_k\, \nabla l_k \|_2 + \| h_k \|_2 \geq \epsilon .$$

Now

$$\begin{aligned} \| P_k(\nabla l_k + B_k\, \hat{s}_k^n) \|_2 &\geq \| P_k\, \nabla l_k \|_2 - \| P_k B_k\, \hat{s}_k^n \|_2 \\ &= \| P_k\, \nabla l_k \|_2 - b_5 \| h_k \|_2 \end{aligned} \quad (6.9)$$

But, since $\| h_k \|_2 \leq \frac{1}{3} \epsilon$, it follows that $\| P_k\, \nabla l_k \|_2 \geq \frac{2}{3} \epsilon$. We have

$$\| P_k(\nabla l_k + B_k\, \hat{s}_k^n) \|_2 \geq \frac{2}{3} \epsilon - b_5\, c_1\, \Delta_k \geq \frac{1}{3} \epsilon . \quad (6.10)$$

Now, from Lemma 6.7, Lemma 6.8 and $\| h_k \|_2 \leq c_1 \Delta_k$, we obtain

$$Pred_k \geq \frac{1}{4} \; || \; P_k \; (\nabla l_k + B_k \hat{s}_k^n \;) \; ||_2 \min \; [\; \overline{\Delta}_k \;, \; \frac{|| \; P_k(\nabla l_k + B_k \hat{s}_k^n \;) \; ||_2}{2 \; b_4} \;]$$

$$- \; c_1 \; [\; b_5 \; || \; \hat{s}_k \; ||_2 + (\; a_5 \; || \; \hat{s}_k \; ||_2 + a_6 \; || \; s_{k-t_k} \; ||_2 \;) \;] \; \Delta_k$$

$$+ \; r_k \; [\; || \; h_k \; ||_2^2 - \; || \; h_k + \nabla h_k^T \hat{s}_k \; ||_2^2 \;] \; . \tag{6.11}$$

So, by using (6.10), we can write

$$Pred_k \geq \frac{1}{8} \; || \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2 \min \; [\; \overline{\Delta}_k \;, \; \frac{|| \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2}{2 b_4} \;]$$

$$+ \; \frac{1}{8} (\; \frac{1}{3} \epsilon \;) \min \; [\; \overline{\Delta}_k \;, \; \frac{\epsilon}{6 \; b_4} \;] \; - \; c_1 \; [\; (a_5 + b_5 + a_6) \; \Delta_* \;] \; \Delta_k$$

$$+ \; r_k \; [\; || \; h_k \; ||_2^2 - \; || \; h_k + \nabla h_k^T \hat{s}_k \; ||_2^2 \;] \; . \tag{6.12}$$

Now, since $\overline{\Delta}_k = \sqrt{\Delta_k^2 - || \; \hat{s}_k^n \; ||_2^2}$, and by using Lemma 6.6 and $|| \; h_k \; ||_2 \leq \frac{\sqrt{3}}{2 \; b_3} \Delta_k$, we obtain

$$\overline{\Delta}_k \geq \sqrt{\Delta_k^2 - (3/4) \; \Delta_k^2} = \frac{1}{2} \Delta_k \; .$$

By substituting the last inequality in (6.12), we obtain

$$Pred_k \geq \frac{1}{8} \; || \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2 \min \; [\; \frac{1}{2} \Delta_k \;, \; \frac{|| \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2}{2 b_4} \;]$$

$$+ \; \frac{1}{8} (\; \frac{1}{3} \epsilon \;) \min \; [\; \frac{1}{2} \Delta_k \;, \; \frac{\epsilon}{6 \; b_4} \;] \; - \; c_1 \; [\; (a_5 + b_5 + a_6) \; \Delta_* \;] \; \Delta_k$$

$$+ \; r_k \; [\; || \; h_k \; ||_2^2 - \; || \; h_k + \nabla h_k^T \hat{s}_k \; ||_2^2 \;] \; .$$

Since c_1 satisfies inequality (6.8), we have

$$Pred_k \geq \frac{1}{8} \; || \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2 \min \; [\; \frac{1}{2} \Delta \;, \; \frac{|| \; P_k(\nabla l_k + B_k \; \hat{s}_k^n \;) \; ||_2}{2 b_4} \;]$$

$$+ \; \frac{r_k}{2} \; [\; ||h_k \; ||_2^2 - \; ||h_k + \nabla h_k^T \hat{s}_k \; ||_2^2 \;] \; . \tag{6.13}$$

This is the desired result. ■

If $|| h_k ||_2 \leq c_1 \Delta_k$, then half of the first term in (6.11) will cancel the second and the third terms, and the fourth term need never enter into the calculation. This implies that if we set $r_k = r_{k-1}$, inequality (6.13) remains valid. So, in this case, the algorithm will not increase the penalty parameter.

Lemma 6.10

Let k be the index of an iteration at which the algorithm does not terminate. If $|| h_k ||_2 \leq c_1 \Delta_k$, where c_1 is as in Lemma 6.9, then there exists a constant c_2 such that

$$Pred_k \geq c_2 \Delta_k$$

Proof

From (6.10) and (6.13), we have

$$Pred_k \geq \frac{1}{8} (\frac{1}{3} \epsilon) \min [\frac{1}{2} \Delta_k , \frac{\epsilon}{6 b_4}]$$

$$\geq \frac{1}{48} \epsilon \min [1 , \frac{\epsilon}{3 b_4 \Delta_*}] \Delta_k .$$

The result now follows if we set $c_2 = \frac{1}{48} \epsilon \min [1 , \frac{\epsilon}{3 b_4 \Delta_*}]$. ■

7. The Behavior of The Penalty Parameter

This section is devoted to the study of the behavior of the penalty parameter. Our objective is to prove that the penalty parameter is bounded. This will imply that r_k is fixed for k sufficiently large. This result is very important in proving global convergence of the algorithm.

Lemma 7.1

If k is the index of an iteration at which the penalty parameter r_k is increased, then

$$r_k \min [\ \Delta_k\ ,\ \frac{||\ h_k\ ||_2}{b_2}\ ] \leq a_7\ ||\ \hat{s}_k\ ||_2 + a_8\ ||\ s_{k-t_k}\ ||_2\ ,$$

where a_7 and a_8 are constants independent of k and s_{k-t_k} is the last acceptable step.

Proof

Let k be the index of an iteration at which the penalty parameter is increased, then by step 3 of the algorithm r_k is updated by the following rule:

$$r_k \ = \ 2\ \frac{\nabla_x l_k^T \hat{s}_k\ +\ ½\ \hat{s}_k^T B_k \hat{s}_k\ +\ \hat{\Delta\lambda}_k^T (h_k\ +\ \nabla h_k^T \hat{s}_k)}{||\ h_k\ ||_2^2 -\ ||\ h_k + \nabla h_k^T \hat{s}_k\ ||_2^2} + \rho\ .$$

This can be written as

$$\begin{aligned} \frac{r_k}{2} [\ ||\ h_k\ ||_2^2 -\ ||\ h_k + \nabla h_k^T\ \hat{s}_k\ ||_2^2\ ] &= (\ \nabla l_k + B_k\ \hat{s}_k\ )^T\ \hat{s}_k - \frac{1}{2}\ \hat{s}_k^T\ B_k\ \hat{s}_k \\ &- (\ \nabla l_k + B_k\ \hat{s}_k\ )^T\ (\ \bar{h}_k + \hat{s}_k^n\ ) \qquad (7.1) \\ &+ \frac{\rho}{2} [\ ||\ h_k\ ||_2^2 -\ ||\ h_k + \nabla h_k^T \hat{s}_k\ ||_2^2\ ]\ . \end{aligned}$$

Thus, from (7.1), (6.1), and Lemma 6.5

$$\begin{aligned} \frac{r_k}{2}\ \frac{||\ h_k\ ||_2}{b_1} \min [\ \Delta_k\ ,\ \frac{||\ h_k\ ||_2}{b_2}\ ] &\leq -\frac{1}{2}\ (\hat{s}_k^t)^T B_k\ \hat{s}_k^n - \frac{1}{2}\ \hat{s}_k^T B_k\ \hat{s}_k^n \\ &- (\ \nabla l_k + B_k\ \hat{s}_k\ )^T\ \bar{h}_k \\ &- \rho\ h_k^T\ \nabla h_k^T\ \hat{s}_k\ , \end{aligned}$$

and we can write

$$\frac{r_k}{2}\ \frac{||\ h_k\ ||_2}{b_1} \min [\ \Delta_k\ ,\ \frac{||\ h_k\ ||_2}{b_2}\ ] \leq\ ||\ B_k\ ||_2\ ||\ \hat{s}_k\ ||_2\ ||\ \hat{s}_k^n\ ||_2$$

$$+\rho \ || \nabla h_k ||_2 \ || \hat{s}_k ||_2 \ || h_k ||_2$$

$$+ | (\nabla l_k + B_k \hat{s}_k)^T \bar{h}_k | . \qquad (7.2)$$

Now by using Lemma 6.8,

$$\frac{r_k}{2} \frac{|| h_k ||_2}{b_1} \min [\Delta_k , \frac{|| h_k ||_2}{b_2}] \leq || B_k ||_2 \ || \hat{s}_k ||_2 \ || \hat{s}_k^n ||_2$$

$$+ (a_5 || \hat{s}_k ||_2 + a_6 || s_{k-t_k} ||_2) \ || h_k ||_2$$

$$+ \rho \ || \nabla h_k ||_2 \ || \hat{s}_k ||_2 \ || h_k ||_2 .$$

But, by Lemma 6.6 $|| \hat{s}_k^n || \leq b_3 \ || h_k ||$ and from the standard assumption 4 $|| \nabla h_k ||_2 \leq b_{10}$ where $b_{10} = \sup_{x \in \Omega} || \nabla h(x) ||$,

$$\frac{r_k || h_k ||_2}{2 b_1} \min [\Delta_k , \frac{|| h_k ||_2}{b_2}] \leq (b_4 b_3 + a_5 + \rho b_{10}) || \hat{s}_k ||_2 \ || h_k ||_2$$

$$+ a_6 \ || s_{k-t_k} ||_2 \ || h_k ||_2 .$$

The result follows immediately upon dividing by $\dfrac{|| h_k ||_2}{2 b_1}$. ∎

Corollary 7.2

If k is the index of an iteration at which the algorithm does not terminate and the penalty parameter r_k is increased, then

$$r_k \Delta_k \leq a_9 \ || \hat{s}_k ||_2 + a_{10} \ || s_{k-t_k} ||_2$$

where a_9 and a_{10} are constants independent of k and s_{k-t_k} is the last acceptable step.

Proof

From Lemma 7.1, if k is index of an iteration at which the penalty parameter

r_k increases, then r_k must satisfy the following inequality:

$$r_k \min \left[\Delta_k , \frac{|| h_k ||_2}{b_2} \right] \leq a_7 \; || \hat{s}_k ||_2 + a_8 \; || s_{k-t_k} ||_2$$

From Lemma 6.9 if $|| h_k ||_2 \leq c_1 \Delta_k$, then we do not increase r_k . So, for any iteration at which the penalty parameter increases, we must have

$$|| h_k ||_2 > c_1 \Delta_k ,$$

and we obtain

$$r_k \Delta_k \min \left[1 , \frac{c_1}{b_2} \right] \leq a_7 \; || \hat{s}_k ||_2 + a_8 \; || s_{k-t_k} ||_2 .$$

Hence,

$$r_k \Delta_k \leq a_9 \; || \hat{s}_k ||_2 + a_{10} \; || s_{k-t_k} ||_2 ,$$

and we get the desired result. ■

Under the standard assumptions, at each iteration at which the termination criteria is not satisfied and the penalty parameter is increased, $r_k \Delta_k$ is bounded. However, if we can bound $\frac{|| s_{k-t_k} ||_2}{\Delta_k}$ by a constant independent of k , we obtain an upper bound on r_k itself. In the following lemma we derive a relation between $|| s_{k-t_k} ||_2$ and Δ_k . In Lemma 7.4 we prove that the penalty parameter is bounded.

Lemma 7.3

Let k be the index of any iteration at which the algorithm does not terminate and the penalty parameter r_k is increased, then

$$\Delta_k \geq c_3 \; || s_{k-t_k} ||_2$$

where s_{k-t_k} is the last acceptable step and c_3 is a constant independent of k

and t_k .

Proof

We consider three cases:

First, if $t_k = 1$, *i.e.*, s_{k-1} is the last acceptable step, then from (2.3), we have

$$\Delta_k \geq \alpha_1 \| s_{k-1} \|_2 .$$

The result in this case follows if we set $c_3 = \alpha_1$.

Second, if s_{k-1} is not the last acceptable step and at the same time $\| h_{k-i} \|_2 \geq c_1 \Delta_{k-i}$ for all $i \in [1 , t_k-1]$, in this case, from Corollary 6.4, we have

$$| Ared_{k-i} - Pred_{k-i} | \leq a_4 r_{k-i} \| \hat{s}_{k-i} \|_2^2$$

Now, from Lemma 6.1, we have

$$Pred_{k-i} \geq \frac{r_{k-i}}{2} \frac{\| h_{k-i} \|_2}{b_1} \min [\Delta_{k-i} , \frac{\| h_{k-i} \|_2}{b_2}] ,$$

But since all $k-i$, $i=1,\ldots,t_k-1$ satisfy $\| h_{k-i} \|_2 \geq c_1 \Delta_{k-i} \geq c_1 \| \hat{s}_{k-i} \|_2$, we have

$$Pred_{k-i} \geq \frac{r_{k-i}}{2} \frac{\| h_{k-i} \|_2}{b_1 b_2} \| \hat{s}_{k-i} \|_2 \min [b_2 , c_1] .$$

Hence,

$$\left| \frac{Ared_{k-i} - Pred_{k-i}}{Pred_{k-i}} \right| \leq \frac{2 a_4 b_1 b_2 \| \hat{s}_{k-i} \|_2}{\min [b_2 , c_1] \| h_{k-i} \|_2} .$$

But since all $k-i$, $i=1,\ldots,t_k-1$ index unacceptable steps, we have

$$(1 - \eta_1) < \left| \frac{Ared_{k-i}}{Pred_{k-i}} - 1 \right| , \qquad 1 \leq i \leq t_k-1$$

So, for all $i \in [1 , t_k-1]$, we have

$$|| \hat{s}_{k-i} ||_2 \geq \frac{(1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,]\ || h_{k-i} ||_2 \,.$$

Now, since $x_{k-1} = x_{k-(t_k-1)}$, $h_{k-1} = h_{k-(t_k-1)}$, we have

$$\begin{aligned}
\Delta_k &\geq \alpha_1 \ || \hat{s}_{k-1} ||_2 \\
&\geq \frac{\alpha_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,]\ || h_{k-1} ||_2 \\
&= \frac{\alpha_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,]\ || h_{k-(t_k-1)} ||_2 \\
&\geq \frac{\alpha_1\, c_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,]\ \Delta_{k-(t_k-1)} \\
&\geq \frac{\alpha_1^2\, c_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,]\ || s_{k-t_k} ||_2 \,.
\end{aligned}$$

The result in this case follows by setting

$$c_3 = \frac{\alpha_1^2\, c_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min [\, b_2 \,,\, c_1 \,] \,.$$

Finally, if the step indexed by $k-1$ is not the last acceptable step and not all $i \in [\, 1 \,,\, t_k - 1 \,]$ satisfy $|| h_{k-i} ||_2 \geq c_1 \Delta_{k-i}$, then there exists at least one $j \in [\, 1 \,,\, t_k - 1 \,]$ such that $|| h_{k-j} ||_2 < c_1 \Delta_{k-j}$. Let l be the smallest integer $\in [\, 1 \,,\, t_k - 1 \,]$ such that $|| h_{k-l} ||_2 < c_1 \Delta_{k-l}$. For all $i \in [\, 1 \,,\, l-1 \,]$, we have

$$|| h_{k-i} ||_2 \geq c_1 \Delta_{k-i}$$

As in the first two parts, if we set

$$c_4 = \min [\, \alpha_1 \,,\, \frac{\alpha_1^2\, c_1 (1-\eta_1)}{2\, a_4\, b_1\, b_2} \min (\, b_2 \,,\, c_1 \,) \,] \,, \tag{7.3}$$

we obtain

$$\Delta_k \geq c_4 \ || \hat{s}_{k-l} ||_2 \tag{7.4}$$

where c_4 is given by (7.3). Now, for $k-l$ we have

$$|| h_{k-l} ||_2 < c_1 \Delta_{k-l} . \tag{7.5}$$

From Lemma 6.3 (where we replace k by $k-l$), the inequality $|| \hat{s}_{k-l} ||_2 \leq \Delta_{k-l}$ and the inequality (7.5), we have

$$|Ared_{k-l} - Pred_{k-l}| \leq a_1 || \hat{s}_{k-l} ||_2^2 + r_{k-l} (a_2 + a_3 c_1) || \hat{s}_{k-l} ||_2^2 \Delta_{k-l} . \tag{7.6}$$

If k indexes an iteration at which r_k is increased, then from Corollary 7.2 and the standard assumptions we know that $r_k \Delta_k$ is bounded. By using inequality (7.4), we arrive at

$$r_{k-l} || \hat{s}_{k-l} ||_2 \leq \frac{1}{c_4} r_{k-l} \Delta_k \leq \frac{1}{c_4} r_k \Delta_k \leq m_0 ,$$

where m_0 is a uniform bound. Hence inequality (7.6) can be written as

$$| Ared_{k-l} - Pred_{k-l} | \leq a_1 || \hat{s}_{k-l} ||_2^2 + (a_2 + c_1 a_3) m_0 || \hat{s}_{k-l} ||_2 \Delta_{k-l}$$

$$\leq [a_1 + (a_2 + c_1 a_3) m_0] || \hat{s}_{k-l} ||_2 \Delta_{k-l} .$$

By using Lemma 6.10, we have

$$\left| \frac{Ared_{k-l} - Pred_{k-l}}{Pred_{k-l}} \right| \leq \frac{a_1 + (a_2 + c_1 a_3) m_0}{c_2} || \hat{s}_{k-l} ||_2 .$$

But since the $k-l$ th is not an acceptable step, then

$$(1 - \eta_1) < \left| \frac{Ared_{k-l}}{Pred_{k-l}} - 1 \right| \leq \frac{a_1 + (a_2 + c_1 a_3) m_0}{c_2} || \hat{s}_{k-l} ||_2 .$$

Hence, by using inequality (7.4), we obtain

$$\Delta_k \geq c_4 || \hat{s}_{k-l} ||_2$$

$$> \frac{c_2 c_4}{[a_1 + (a_2 + c_1 a_3) m_0]} (1 - \eta_1) .$$

$$\geq \frac{c_2 c_4 (1 - \eta_1)}{[a_1 + (a_2 + c_1 a_3) m_0] \Delta_*} || s_{k-l_k} ||_2$$

The result then follows if we set

$$c_3 \;=\; \min\;[\; c_4\,,\; \frac{c_2\, c_4\,(\,1-\eta_1\,)}{[\; a_1 + (\; a_2 + c_1\, a_3\,)\; m_0\;]\;\Delta_*}\;]\;.$$

This completes the proof. ■

The following lemma uses Corollary 7.2 and Lemma 7.3 to prove that if each member of the sequence of iterates generated by the algorithm does not satisfy the termination condition in step 1 of the algorithm, then the penalty parameter is bounded.

Lemma 7.4

Under the standard assumptions, if each member of the sequence of iterates generated by the algorithm does not satisfy the termination condition (3.1), then the penalty parameter sequence $\{\; r_k \;\}$ is bounded.

Proof

The proof is by contradiction. Suppose that $\{\; r_k \;\}$ is not bounded. This implies that there exists an infinite subsequence of indices $\{k_j\}$ such that $\{r_{k_j}\}$ is increased. Now, from Lemma 6.9, we never increase the penalty parameter if $||\; h_k \;||_2 \;\le\; c_1\,\Delta_k$. So, $||\; h_{k_j} \;||_2 \;>\; c_1\,\Delta_{k_j}$.

Let m be any integer $\epsilon \{\; k_j \;\}$, then from Corollary 7.2 we can write

$$r_m\;\Delta_m \;\le\; a_9\;||\;\hat{s}_m\;||_2 + a_{10}\;||\;s_{m-t_m}\;||_2\,, \tag{7.7}$$

where s_{m-t_m} is the last acceptable step. On the other hand, from Lemma 7.3 we have

$$||\;s_{m-t_m}\;||_2 \;\le\; \frac{1}{c_3}\,\Delta_m\;.$$

By substituting the last inequality in (7.7), we get

$$r_m \leq a_9 + \frac{a_{10}}{c_3} .$$

Since $a_9 + \frac{a_{10}}{c_3}$ is independent of m, it is an upper bound of the sequence $\{ r_{k_j} \}$ contradicting the assumption that the sequence $\{ r_{k_j} \}$ is increased. This proves the theorem. ■

From the last lemma, we can conclude that for all k, $1 \leq r_k \leq r_*$ where r_* is a constant independent of k.

Since if r_k is increased, it is increased by a quantity $\geq \rho$, then the number of iterations at which the penalty parameter is increased must be finite. Hence, there exists a constant $\bar{k}$ such that

$$r_k = r_{\bar{k}} \qquad \text{for all } k \geq \bar{k} . \tag{7.8}$$

8. The Global Convergence Theory

In this section we present the proofs of our main global convergence results that have been stated in Section 5. We start by restating and then proving Theorem 5.1. First we introduce some notation that will be used in the remainder of this paper.

We call an iteration a successful iteration if the trial step of that iteration was accepted because $\frac{Ared_k}{Pred_k} > \eta_1$. Otherwise, the iteration is said to be unsuccessful.

We denote by $S(k_1 , k_2)$ the set of indices of successful iterations in the interval $[k_1 , k_2]$.

Theorem 5.1

Under the standard assumptions, at any point (x_k , λ_k) generated by the algorithm, either the termination condition of the algorithm will be met or an acceptable step will be found. *i.e.* the condition $\frac{Ared_{k+j}}{Pred_{k+j}} \geq \eta_1$ will be satisfied for some j .

Proof

If the termination condition of the algorithm is satisfied, then there is nothing to prove. Assume that the point (x_k , λ_k) does not satisfy the termination condition in step 1 of the algorithm.

First, we assume that $|| h_k ||_2 > c_1 \Delta_k$ where c_1 is as in Lemma 6.9. Using Lemma 6.1 we obtain

$$Pred_k \geq \frac{r_k}{2} \frac{|| h_k ||_2 \ \Delta_k}{b_1 \ b_2} \min [\ b_2 , c_1 \ ] ,$$

then, using Corollary 6.4, we obtain

$$| \frac{Ared_k}{Pred_k} - 1 | \leq \frac{2 \ a_4 \ b_1 \ b_2}{|| h_k ||_2 \min [\ b_2 , c_1 \ ]} \Delta_k .$$

Now, as Δ_k gets smaller, the quantity $| \frac{Ared_k}{Pred_k} - 1 |$ approaches 0 and hence the condition $\frac{Ared_k}{Pred_k} \geq \eta_1$ will be met after a finite number of trials.

Now, assume that $|| h_k ||_2 \leq c_1 \Delta_k$. Using Corollary 6.4, Lemma 6.10 and $r_k \leq r_*$, we can write

$$| \frac{Ared_k - Pred_k}{Pred_k} | \leq \frac{a_4 \ r_*}{c_2} \Delta_k .$$

So, as Δ_k gets smaller, the quantity $| \frac{Ared_k}{Pred_k} - 1 |$ approaches 0, and hence the condition $\frac{Ared_k}{Pred_k} \geq \eta_1$ will be met after a finite number of trials. This completes the proof. ■

The proof of our main global convergence result, Theorem 5.2, uses the following two lemmas. The first lemma proves that under the standard assumptions, either the algorithm terminates, or converges to a feasible point. The second lemma proves that under the standard assumptions, either the algorithm terminates, or $\| P_k \nabla f_k \|_2 < \epsilon$, for some k sufficiently large, where $\epsilon > 0$ is any given constant. This means that if each member of the sequence of iterates generated by the algorithm does not satisfy the termination condition (3.1), then the sequence $\{ \| P_k \nabla f_k \|_2 \}$ will not be bounded away from zero.

Lemma 8.1

Let the standard assumptions hold. If each member of the sequence of iterates generated by the algorithm does not satisfy the termination condition (3.1), then

$$\lim_{k \to \infty} \| h_k \|_2 = 0 .$$

Proof

Suppose $\limsup_{k \to \infty} \| h_k \|_2 = \epsilon_0 > 0$. Then there exists an infinite sequence of indices $\{ k_j \}$ such that $\| h_k \|_2 \geq \frac{\epsilon_0}{2}$ for all $k \in \{ k_j \}$.

Let $\hat{k}$ be such that $\hat{k} \in \{ k_j \}$, $\hat{k} \geq \bar{k}$, where $\bar{k}$ is the same as in (7.8). Since $h \in C^2$, we have that for some $\beta > 0$ and any $x \in \Omega$

$$\| h(x) \|_2 \geq \| h_{\hat{k}} \|_2 - \| h(x) - h_{\hat{k}} \|_2 \geq \| h_{\hat{k}} \|_2 - \beta \| x - x_{\hat{k}} \|_2 .$$

This implies that for all x that satisfies $\| x - x_{\hat{k}} \|_2 \leq \frac{\| h_{\hat{k}} \|_2}{2\beta}$, we have

$$\| h(x) \|_2 \geq \frac{\| h_{\hat{k}} \|_2}{2} .$$

Let $\sigma = \dfrac{||\ h_{\hat{k}}\ ||_2}{2\ \beta}$ and consider the ball

$$B_\sigma = [\ x : \ ||\ x - x_{\hat{k}}\ ||_2 \leq \sigma\ ]\ .$$

First we will show that eventually the iterate must move outside B_σ .

If $x_k \in B_\sigma$ for all $k \geq \hat{k}$, then from Lemma 6.1 and $r_k \geq 1$,

$$Pred_k \geq \frac{1}{2} \frac{||\ h_k\ ||_2}{b_1} \min [\ \Delta_k\ ,\ \frac{||\ h_k\ ||_2}{b_2}\ ]$$

$$\geq \frac{1}{2} \frac{||\ h_{\hat{k}}\ ||_2}{2\ b_1} \min [\ \Delta_k\ ,\ \frac{||\ h_{\hat{k}}\ ||_2}{2\ b_2}\ ]\ .$$

If all $k \geq \hat{k}$ are not acceptable steps, then we contradict Theorem 5.1. Hence, there exists an infinite sequence of indices indexing successful steps inside the ball. For any such k we have

$$\Phi_k - \Phi_{k+1} = Ared_k \geq \eta_1\ Pred_k$$

$$\geq \frac{\eta_1}{2} \frac{||\ h_{\hat{k}}\ ||_2}{2\ b_1} \min [\ \Delta_k\ ,\ \frac{||\ h_{\hat{k}}\ ||_2}{2\ b_2}\ ]\ . \tag{8.1}$$

Since Φ_k is bounded below and $||\ h_{\hat{k}}\ ||_2 > 0$, inequality (8.1) implies that

$$\liminf_{k \to \infty} \Delta_k = 0 \tag{8.2}$$

Define σ_1 to be a constant that satisfies:

$$\sigma_1 < \min [\ 1\ ,\ \frac{a\ b\ \Delta_{\hat{k}}}{\alpha_1\ r_*\ (\ 1 - \eta_2\ )}\ ,\ \frac{||\ h_{\hat{k}}\ ||_2}{2}\ ]$$

where $a = \max [\ r_*\ ,\ 2\ r_*^2\ a_4\ ]$ and $b = \max [\ b_1\ ,\ b_2\ ]$. Now, because of (8.2), there exist some sufficiently large k such that

$$\Delta_k \leq \frac{\alpha_1\ \sigma_1\ r_*}{a\ b} (\ 1 - \eta_2\ )\ . \tag{8.3}$$

Let m be the first integer greater than $\hat{k}$ such that (8.3) holds. This implies that $m \geq \hat{k}+1$, and using (2.3) we get

$$b \; || \hat{s}_{m-1} ||_2 \leq \frac{b \; \Delta_m}{\alpha_1}$$

$$\leq \frac{\sigma_1 \; r_*}{a} (1 - \eta_2) \tag{8.4}$$

$$\leq \sigma_1 (1 - \eta_2) \leq \sigma_1 \, . \tag{8.5}$$

Now, by using Lemma 6.1 and the fact that $r_{m-1} \geq 1$, we obtain

$$Pred_{m-1} \geq \frac{1}{2} \frac{|| h_{m-1} ||_2}{b_1} \min [\; || \hat{s}_{m-1} ||_2 \, , \frac{|| h_{m-1} ||_2}{b_2}] \, , \tag{8.6}$$

and since $m-1 \geq \hat{k}$, x_{m-1} lies inside the ball B_σ and by using the definition of σ_1 above, we have

$$|| h_{m-1} ||_2 \geq \frac{|| h_{\hat{k}} ||_2}{2} \geq \sigma_1 \, . \tag{8.7}$$

From (8.5) and (8.7) we have

$$b \; || \hat{s}_{m-1} ||_2 \leq \; || h_{m-1} ||_2 \, .$$

By substituting the last inequality and (8.7) into (8.6), we obtain

$$Pred_{m-1} \geq \frac{\sigma_1}{2 \; b} \; || \hat{s}_{m-1} ||_2 \, .$$

But, by Corollary 6.4,

$$| \; Ared_{m-1} - Pred_{m-1} \; | \leq a_4 \; r_* \; || \hat{s}_{m-1} ||_2^2 \, . \tag{8.8}$$

So,

$$| \frac{Ared_{m-1} - Pred_{m-1}}{Pred_{m-1}} | \leq \frac{2 \; a_4 \; b \; r_* \; || \hat{s}_{m-1} ||_2^2}{\sigma_1 \; || \hat{s}_{m-1} ||_2} \, .$$

Now using (8.4), we obtain

$$\left| \frac{Ared_{m-1} - Pred_{m-1}}{Pred_{m-1}} \right| \le \frac{2\, a_4\, r_*^2\, \sigma_1}{\sigma_1\, a} (1 - \eta_2) \le (1 - \eta_2) .$$

This implies that

$$\frac{Ared_{m-1}}{Pred_{m-1}} \ge \eta_2 .$$

Hence from the rule of updating the radius of the trust region, we have

$$\Delta_{m-1} \le \Delta_m .$$

The last inequality implies that $k = m-1$ satisfies (8.3). This contradicts the supposition that m is the smallest such index and means that there is no $m > \hat{k}$ such that (8.3) holds. Hence, for all $k > \hat{k}$, we have

$$\Delta_k > \frac{\alpha_1\, \sigma_1\, r_*}{a\, b} (1 - \eta_2)$$

which contradict (8.2). Hence, eventually $\{ x_k \}$ must leave the ball B_σ for some $k > \hat{k}$. Let $l+1$ be the first integer greater than $\hat{k}$ such that x_{l+1} does not lie inside the ball B_σ . Since $x_{l+1} \neq x_{\hat{k}}$, there must exist at least one acceptable step in the set of iterates indexed $\{ \hat{k},...,l \}$, so by Lemma 6.1,

$$\Phi_{\hat{k}} - \Phi_{l+1} = \sum_{k=\hat{k}}^{l} (\Phi_k - \Phi_{k+1}) \ge \sum_{k \epsilon S(\hat{k},l)} \eta_1 Pred_k$$

$$\ge \sum_{k \epsilon S(\hat{k},l)} \frac{\eta_1}{2} \frac{|| h_{\hat{k}} ||_2}{2 b_1} \min \left[\Delta_k , \frac{|| h_{\hat{k}} ||_2}{2\, b_2} \right] .$$

If $\Delta_k < \dfrac{|| h_{\hat{k}} ||_2}{2\, b_2}$ for all $k \,\epsilon\, S(\hat{k},l)$, then

$$\Phi_{\hat{k}} - \Phi_{l+1} \ge \frac{\eta_1}{2} \frac{|| h_{\hat{k}} ||_2}{2\, b_1} \sum_{k \epsilon S(\hat{k},l)} \Delta_k$$

$$\ge \frac{\eta_1}{2} \frac{|| h_{\hat{k}} ||_2}{2\, b_1} \sigma .$$

Otherwise,

$$\Phi_{\hat{k}} - \Phi_{l+1} \geq \frac{\eta_1}{2} \frac{||\, h_{\hat{k}} \,||_2^2}{4\, b_1\, b_2} .$$

In either case

$$\Phi_{\hat{k}} - \Phi_{l+1} \geq \frac{\eta_1}{2} \frac{||\, h_{\hat{k}} \,||_2}{2\, b_1} \min \left[\sigma , \frac{||\, h_{\hat{k}} \,||_2}{2\, b_2} \right]$$

$$= \frac{\eta_1}{2} \frac{||\, h_{\hat{k}} \,||_2}{2\, b_1} \min \left[\frac{||\, h_{\hat{k}} \,||_2}{2\, \beta} , \frac{||\, h_{\hat{k}} \,||_2}{2\, b_2} \right]$$

$$= \frac{\eta_1}{2} \frac{||\, h_{\hat{k}} \,||_2^2}{4\, b_1} \min \left[\frac{1}{\beta} , \frac{1}{b_2} \right] . \tag{8.9}$$

Since $\{ \Phi_k \}$ is bounded below and is a decreasing sequence, $\{ \Phi_k \}$ converges to some limit Φ_* . Taking the limit as l goes to infinity in inequality (8.9), we obtain

$$\Phi_{\hat{k}} - \Phi_* \geq \frac{\eta_1}{2} \frac{||\, h_{\hat{k}} \,||_2^2}{4\, b_1} \min \left[\frac{1}{\beta} , \frac{1}{b_2} \right] .$$

If we now take the limit as $\hat{k}$ goes to infinity, we obtain

$$0 \geq \frac{\eta_1}{2} \frac{\epsilon_0}{8\, b_1} \min \left[\frac{1}{\beta} , \frac{1}{b_2} \right]$$

which contradicts $\epsilon_0 > 0$. The supposition is wrong and hence the lemma is proved. ■

Lemma 8.2

Let the standard assumptions hold. If each member of the sequence of iterates generated by the algorithm does not satisfy the termination condition (3.1), then

$$\liminf_{k \to \infty} \; ||\, P_k \, \nabla f_k \,||_2 = 0 .$$

Proof

The proof is by contradiction. Suppose that there exists an $\epsilon_0 > 0$ and an integer K such that $|| P_k \nabla f_k ||_2 \geq \epsilon_0$ for all $k \geq K$.

Since, by using (6.9),

$$|| P_k (\nabla l_k + B_k \hat{s}_k^n) ||_2 \geq || P_k \nabla f_k ||_2 - b_5 || h_k ||_2 ,$$

From Lemma 8.1, there exist k_1 sufficiently large such that for all $k \geq k_1$, we have

$$|| h_k ||_2 < \frac{1}{2 b_5} \epsilon_0 .$$

Thus for $k \geq \max [K , k_1]$

$$|| P_k (\nabla l_k + B_k \hat{s}_k^n) ||_2 \geq \frac{1}{2} \epsilon_0 .$$

Now, since from (6.4) and Lemma 6.8,

$$Pred_k \geq \frac{1}{4} || P_k (\nabla l_k + B_k \hat{s}_k^n) ||_2 \min [\overline{\Delta}_k , \frac{|| P_k(\nabla l_k + B_k \hat{s}_k^n) ||_2}{2 b_4}]$$
$$- (b_5 || \hat{s}_k ||_2 || h_k ||_2) - (a_5 || \hat{s}_k ||_2 + a_6 || s_{k-t_k} ||_2) || h_k ||_2 ,$$

and since $|| h_k ||_2$ converges to zero and $|| \hat{s}_k ||_2$ and $|| s_{k-t_k} ||_2$ are bounded, then there exists an integer $k_2 \geq \max [K , k_1]$ such that for all $k \geq k_2$ we have

$$Pred_k \geq \frac{1}{8} || P_k(\nabla l_k + B_k \hat{s}_k^n) ||_2 \min [\frac{1}{2} \Delta_k , \frac{|| P_k(\nabla l_k + B_k \hat{s}_k^n) ||_2}{2 b_4}] .$$

Thus, for all $k \geq k_2$, we have

$$Pred_k \geq \frac{1}{8} \frac{\epsilon_0}{2} \min [\frac{1}{2} \Delta_k , \frac{\epsilon_0}{4 b_4}] .$$

From Theorem 5.1 there exists an infinite sequence of successful iterations. Now, for any successful iteration indexed $k \geq k_2$, we have

$$Ared_k \;\geq\; \eta_1\, Pred_k \;\geq\; \frac{\eta_1}{32}\, \epsilon_0 \min\, [\; \Delta_k \;,\; \frac{\epsilon_0}{2b_4}\;]\; .$$

If $\bar{k}_2 \geq \max\, [\, k_2 \,,\, \bar{k}\,]$, then the last inequality and the assumption that $\{\Phi_k\}$ is bounded below imply that

$$\infty \;>\; \sum_{k=\bar{k}_2}^{\infty} (\; \Phi_k - \Phi_{k+1} \;) \;=\; \sum_{k=\bar{k}_2}^{\infty} Ared_k$$

$$\geq \sum_{k=S(\bar{k}_2\,,\infty)} \frac{\eta_1}{32}\, \epsilon_0 \min\, [\; \Delta_k \;,\; \frac{\epsilon_0}{2b_4}\;]\; .$$

This implies that

$$\liminf_{k\to\infty}\; \Delta_k \;=\; 0\; . \tag{8.10}$$

This means that there exists an integer $k_3 \geq \bar{k}_2$ such that

$$\Delta_k \;\leq\; \frac{\alpha_1\, \sigma_2}{a} (\; 1 - \eta_2\;) \tag{8.11}$$

is satisfied for some $k \geq k_3$, where $a = \max\, [\; 1 \;,\; \frac{32\; a_4\; r_*}{\epsilon_0}\;]$ and σ_2 is defined to be a constant that satisfies

$$\sigma_2 \;<\; \min\, [\; 1 \;,\; \frac{a\;\Delta_{k_3}}{\alpha_1 (\; 1 - \eta_2\;)} \;,\; \frac{\epsilon_0}{2\; b_4}\;]\; .$$

Let m be the first integer greater than k_3 such that (8.11) holds. This implies that $m \;\geq\; k_3{+}1$. So, from (2.3),

$$||\; \hat{s}_{m-1}\; ||_2 \;\leq\; \frac{\Delta_m}{\alpha_1} \;\leq\; \frac{\alpha_1\, \sigma_2}{\alpha_1\; a} (\; 1 - \eta_2\;) \tag{8.12}$$

$$\leq\; \sigma_2 (\; 1 - \eta_2\;) \;\leq\; \sigma_2 \;<\; \frac{\epsilon_0}{2\; b_4}\; .$$

We obtain

$$Pred_{m-1} \;\geq\; \frac{\epsilon_0}{32}\;\; ||\; \hat{s}_{m-1}\; ||_2\; .$$

So, by using Corollary 6.4, (8.12), and the last inequality, we get

$$\left| \frac{Ared_{m-1} - Pred_{m-1}}{Pred_{m-1}} \right| \leq \frac{32\ a_4\ r_*\ \|\ \hat{s}_{m-1}\ \|_2}{\epsilon_0}$$

$$\leq \frac{32\ a_4\ r_*\ \sigma_2}{\epsilon_0\ a} (\ 1 - \eta_2\ )$$

$$\leq \sigma_2 (\ 1 - \eta_2\ )\ \leq\ (\ 1 - \eta_2\ )\ .$$

The last inequality implies that

$$\frac{Ared_{m-1}}{Pred_{m-1}} \geq \eta_2\ .$$

Hence, from the rule of updating the radius of the trust region in Scheme 2.2, we obtain

$$\Delta_{m-1} \leq \Delta_m\ .$$

This implies that $m-1$ satisfies (8.11) which contradicts the assumption that m is the smallest integer $\geq k_3$ such that (8.11) holds. Hence, for all $k \geq k_3$, we have

$$\Delta_k > \frac{\alpha_1\ \sigma_2}{a} (\ 1 - \eta_2\ )\ .$$

The last inequality contradicts (8.10). The supposition is contradicted and hence the lemma is proved. ∎

Now let us again state and then prove, our main global convergence result, Theorem 5.2.

Theorem 5.2

Under the standard assumptions, the algorithm produces iterates $\{\ x_k \}$ which satisfy

$$\liminf_{k \to \infty} \left[\; || \, h_k \, ||_2 + \; || \, P_k \, \nabla f_k \, ||_2 \right] \; = \; 0$$

Proof

The proof follows immediately from Lemma 8.1 and Lemma 8.2. ■

9. Concluding Remarks

We have presented a global convergence analysis for a variant of the 1984 Celis-Dennis-Tapia algorithm in which we use a different scheme for updating the penalty parameter. This scheme ensures that the merit function is decreased at each iteration by at least a fraction of Cauchy decrease. This indicates compatibility with the choice of θ_k in the CDT subproblem.

To force global convergence, we have employed, as a merit function, the augmented Lagrangian which is naturally compatible with the subproblem. For more details, see Celis, Dennis, Martinez, Tapia, and Williamson (1989).

Schittkowski (1983), Gill, Murray, Saunders, and Wright (1986) and Powell and Yuan (1986-a and 1986-b) have also considered this function as a merit function.

Powell and Yuan (1986-a and 1986-b) used the least-squares multiplier estimate to update the estimate of the multiplier λ , and hence they treated it as a function of x rather than a separate variable. They proved several global and local convergence properties using this merit function. We prefer our way of updating the multiplier λ for several reasons including the fact that it is less expensive to calculate than the Powell and Yuan's choice.

For future work, there are many questions that need to be answered.

Although intensive numerical investigation with the CDT algorithm was reported by Celis, Dennis and Tapia (1985), Celis (1985) and Celis, Dennis, Martinez, Tapia, and Williamson (1989), we believe that the implementation of the algorithm must be refined. In particular, an efficient algorithm for solving the CDT subproblem is needed. This will require a closer look at the CDT subproblem and the characteristics of its solution. Currently, this is a topic of research, *e.g.* Yuan (1987) and Zhang (1988), but the problem has not been solved.

A related important question is how to use a secant approximation of the Hessian of the Lagrangian in order to produce a more efficient algorithm. We believe that Tapia (1988) will be of considerable value here.

Another important topic that we expect to consider is how to incorporate inequality constraints into the formulation of the algorithm.

Acknowledgments

This work constitutes a part of the author's doctoral thesis and was supervised by Professors John Dennis and Richard Tapia in the Department of Mathematical Sciences, Rice University, Houston, Texas. He gratefully thanks his advisors for their numerous helpful suggestions.

The author is also greatly indebted to Professor Richard Byrd for his constant help and valuable discussions throughout this research. He is also indebted to Professor Ph. Toint for his valuable suggestions and comments on an earlier version of this paper.

References

R. H. Byrd, R. B. Schnabel and G. A. Shultz (1987), A Trust Region Algorithm for Nonlinearly Constrained Optimization, SIAM J. Numer. Anal., 24 (5), pp. 1152-1170.

R. H. Byrd, E. O. Omojokun, R. B. Schnabel, and G. A. Shultz (1987), Robust Trust Region Methods for Nonlinearly Constrained Optimization, Presented at the 1987 SIAM Conference on Optimization, Houston, TX (in preparation).

M. R. Celis (1985), A Trust Region Strategy for Nonlinear Equality Constrained Optimization, Ph.D. thesis, Rice University.

M. R. Celis, J. E. Dennis, and R. A. Tapia (1985), A Trust Region Strategy for Nonlinear Equality Constrained Optimization, Numerical Optimization 1984 (P. Boggs, R. Byrd and R. Schnabel, ed), SIAM Philadelphia, pp. 71-82.

M. R. Celis, J. E. Dennis, J. M. Martinez, R. A. Tapia, and K. Williamson (1989), An Algorithm Based on a Convenient Trust-Region Subproblem for Nonlinear Programming. (in preparation)

A. R. Conn, N. I. M. Gould, and Ph. L. Toint (1987), Global Convergence of a Class of Trust Region Algorithms for Optimization With Simple Bounds, University of Waterloo Computer Science Dept. Report No 86/1, Waterloo, Canada.

J. E. Dennis, M. M. El-Alem, and R. A. Tapia (1989), Numerical Experience With Some Trust-Region Subproblems For Constrained Optimization, Presented at the Third SIAM Conference on Optimization, April (1989), Boston, MA. (in preparation)

J. E. Dennis and R. B. Schnabel (1983), Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall, Englewood Cliffs, N.J.

M. M. El-Alem (1988), A Global Convergence Theory for a Class of Trust Region Algorithms for Constrained Optimization, Ph. D. thesis, Rice University, May 1988.

A. V. Fiacco and G. P. McCormick (1968), Nonlinear Programming: Sequential Unconstrained Minimization Techniques, John Wiley and Sons, New York.

R. Fontecilla (1986), A Heuristic Algorithm for Nonlinear Programming. (to appear in J. of Opt. Th. and Appl.).

R. Fontecilla (1988), Local Convergence of Secant Methods for Nonlinear Constrained Optimization, SIAM J. Numer. Anal., 25 (3), pp. 692-712.

R. Fontecilla, T. Steihaug, and R. A. Tapia (1987), A Local Convergence Theory for a Class of Quasi-Newton Methods for Constrained Optimization, SIAM J. Numer. Anal., 24 (5), pp. 1133-1151.

D. M. Gay (1983), A Trust Region Approach to Linearly Constrained Optimization, Lecture Notes in Mathematics, No. 1066, (A. Dold and B. Eckmann,ed.), pp. 72-105.

P. E. Gill, W. Murray, M. A. Saunders, and M. Wright (1986), Some Theoretical Properties of an Augmented Lagrangian Merit Function, Report SOL 86-6, Stanford University.

J. J. Moré and D. C. Sorensen (1983), Computing A Trust Region Step, SIAM J. Sci. Stat. Comput., l4, No3, pp. 553-572.

M. J. D. Powell (1986), A method for Nonlinear Constraints in Minimization Calculations, Presented at the 1986 IMA/SIAM Meeting on "The State of the Art in Numerical Analysis".

M. J. D. Powell and Y. Yuan (1986-a), A Recursive Quadratic Programming Algorithm that Uses Differentiable Exact Penalty Functions, Math. Programming, 35, No 3, pp. 265-278.

M. J. D. Powell and Y. Yuan (1986-b), A Trust Region Algorithm for Equality Constrained Optimization, Report DAMTP 1986/NA2, Cambridge, England.

K. Schittkowski (1983), On the Convergence of a Sequential Quadratic Programming Method with an Augmented Lagrangian Line Search Function, Math. Operationsforschung U. Statistik, Ser. Optimization, 14, pp. 197-216.

G. A. Shultz, R. B. Schnabel and R. H. Byrd (1985), A Family of Trust-Region-Based Algorithms for Unconstrained Minimization with Strong Global Convergence Properties, SIAM J. Numer. Anal., 22, No 1, pp. 47-67.

R. A. Tapia (1977), Diagonalized Multiplier Methods and Quasi-Newton Methods for Constrained Optimization, J. of Opt. Th. and Appl., 22, pp. 135-194.

R. A. Tapia (1978), Quasi-Newton Methods for Equality Constrained Optimization: Equivalence of Existing Methods and a New Implementation, Nonlinear Programming 3, (O. L. Mangasarian, R. R. Meyer, and S. M. Robinson eds.), Academic Press, New York, pp. 125-164.

R. A. Tapia (1983), An Introduction to the Algorithms and Theory of Constrained Optimization, Unpublished notes, Rice University, Houston, Tx.

R. A. Tapia (1988), On Secant Update for Use in General Constrained Optimization, Mathematics of computation, 51, No 183, pp. 181-202.

A. Vardi (1985), A Trust Region Algorithm for Equality Constrained Minimization: Convergence Properties and Implementation, SIAM J. Numer. Anal., 22, No 3, pp. 575-591.

Y. Yuan (1987), On a Subproblem of Trust Region Algorithms For Constrained Optimization, Report DAMTP 1987/NA10, Cambridge, England.

Y. Zhang (1988), Computing a Celis-Dennis-Tapia Trust Region Step for Equality Constrained Optimization, TR 88-16, Department of Mathematical Sciences, Rice University.

A Global Convergence Theory for the Celis-Dennis-Tapia Trust Region Algorithm for Constrained Optimization[1]

by

Mahmoud El Alem[2]

Technical Report 88-10, September 1988

Revised May, 1989

---

[1]Research sponsored by SDIO/IST/ARO, AFOSR 85-0243, and DOE DEFG05-86ER25017.

[2]Department of Mathematical Sciences, Rice University, Houston, TX 77251-1892.