Multilevel Algorithms for Nonlinear Equations and Equality Constrained Optimizations

Natalia Alexandrov

CRPC-TR93318
May 1993

Center for Research on Parallel Computation
Rice University
P.O. Box 1892
Houston, TX 77251-1892

Multilevel Algorithms for Nonlinear Equations
and Equality Constrained Optimization

Natalia Alexandrov

May 1993

TR93-20

RICE UNIVERSITY

Multilevel Algorithms for Nonlinear Equations and Equality Constrained Optimization

by

Natalia Alexandrov

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

APPROVED, THESIS COMMITTEE:

John E. Dennis, Jr., Chairman
Noah Harding Professor of Computational and Applied Mathematics

Richard A. Tapia,
Noah Harding Professor of Computational and Applied Mathematics

B. Frank Jones.
Noah Harding Professor of Mathematics

Robert M. Lewis
Computational and Applied Mathematics

Dan C. Sorensen,
Professor of Computational and Applied Mathematics

Houston, Texas
April, 1993

Abstract

Multilevel Algorithms for Nonlinear Equations and Equality Constrained Optimization

by

Natalia Alexandrov

A general trust region strategy is proposed for solving nonlinear systems of equations and equality constrained optimization problems by means of multilevel algorithms. The idea is to use the trust region strategy to globalize the Brent algorithms for solving nonlinear equations, and to extend them to algorithms for solving optimization problems. The new multilevel algorithm for nonlinear equality constrained optimization operates as follows. The constraints are divided into an arbitrary number of blocks dictated by the application. The trial step from the current solution approximation to the next one is computed as a sum of substeps, each of which must predict a Fraction of Cauchy Decrease on the subproblem of minimizing the model of each constraint block, and, finally, the model of the objective function, restricted to the intersection of the null spaces of all the preceding linearized constraints. The models of each constraint block and of the objective function are built by using the function and derivative information at different points. The merit function used to evaluate the step is a modified ℓ_2 penalty function with nested penalty parameters. The scheme for updating the penalty parameters is a generalization of the one proposed by El-Alem. The algorithm is shown to be well-defined and globally convergent under reasonable assumptions. The global convergence theory for the optimization algorithm implies global convergence of the multilevel algorithm for nonlinear equations and a modification of a class of trust region algorithms proposed by Maciel, and Dennis, El-Alem and Maciel.

The algorithms are expected to become flexible tools for solving a variety of optimization problems and to be of great practical use in applications such as multidisciplinary design optimization. In addition, they serve to establish a foundation for the study of the general multilevel optimization problem.

Acknowledgments

First, I wish to thank my parents for their unwavering faith in me and for the continual encouragement and inspiration they have always provided.

Every teacher who has ever taught me has made a direct or an indirect contribution to this work.

I owe a special debt to my advisor, Professor John Dennis. He has always generously shared his ideas and his knowledge with me.

My profound gratitude goes to Michael Lewis for his valiant effort of reading my thesis carefully in a very short time.

I would like to thank all members of my committee for their attention. Special thanks go to Professor Richard Tapia for welcoming me into the program and for sharing his enthusiasm for mathematics. I am very grateful to Professor Frank Jones for his inspired teaching. I wish to acknowledge Professor Dan Sorensen for his valuable comments.

Many thanks go to Mahmoud El-Alem for his time and good advice.

I would like to thank many other people. The professors and students of the Department of Mathematics made my first, difficult year in graduate school a good one. Virginia Torczon has welcomed me into the Department of Mathematical Sciences and has been a warm and supportive influence ever since. I wish to thank Shireen Dadmehr for always being a good friend in the past three years. Karen Williamson and Amr El-Bakry freely contributed their time and advice. Michael Pearlman has swiftly dealt with all my computer problems. I have enjoyed my conversations with Franka Mailian about music and art. Fran, Vivian Choi, and Linda Neyra have always been a great help.

Finally, I would like to thank Antonina Ivanovna, my secondary school teacher of Mathematics, who did much to give me a foundation that has been serving me to this day.

Contents

Abstract ii
Acknowledgments iii
List of Illustrations vii
List of Tables viii

1 Preliminaries **1**
1.1 General Introduction: Multilevel Algorithms for NLEQ and EQC . . 3
1.2 Historical Background 7
1.2.1 The Methods of Brown and Brent 7
1.2.2 Null Space Methods 8
1.3 Theoretical Background 10
1.3.1 Unconstrained Minimization Results 10
1.3.2 Analysis and Algebra 13
1.3.3 Notation 15

2 Multilevel Algorithm for Solving Nonlinear Equations Based on Brent's Method **18**
2.1 Brent's Algorithm for Nonlinear Equations 18
2.2 Globalization of Brent's Method: Multilevel Algorithm 19
2.2.1 The Levenberg-Marquardt Step 21
2.2.2 The Truncated Brent Step 24
2.2.3 Updating the Penalty Parameters 25
2.2.4 Motivation for the Merit Function 27
2.2.5 Step Evaluation and Trust Region Updating 28
2.2.6 Convergence Properties 30

3 Multilevel Algorithm for Equality Constrained Optimization Based on Brent's Method **32**

3.1 Modification of an Existing Algorithm 32
3.2 Extension of Brent's Method to Equality Constrained Optimization: a Multilevel Algorithm 35
3.2.1 The Merit Function and Reductions 36
3.2.2 The Statement of the Algorithm 37
3.2.3 Updating the Penalty Parameters 38
3.2.4 An Alternative Penalty Parameter Scheme 40
3.2.5 The Stopping Criteria 41

4 Global Convergence Theory **43**
4.1 Basic Ingredients of a Global Convergence Proof 43
4.2 Assumptions 45
4.3 Technical Lemmas 46
4.4 The Behavior of the Model 50
4.5 The Behavior of the Penalty Parameters 58
4.6 The Trust Region Radius Is Bounded Below 74
4.7 The Algorithm is Well-Defined 75
4.8 The Global Convergence Theorem 77
4.9 Corollaries 80

5 Implementation **81**
5.1 Trust Region Updating Strategies 82
5.2 Projectors 86
5.2.1 Using QR-Factorization 87
5.2.2 Using Reduced Basis Projectors 88
5.3 Order of Constraints 89
5.4 Scaling 89
5.5 Penalty Parameters 91

6 Applications and Conclusions **92**
6.1 Applications 92
6.2 Research Plan 92
6.3 Summary 94

A Glossary of the Lemmas and Theorems **95**

B Test Problems **100**

Bibliography **105**

Illustrations

1.1 Multilevel algorithm for NLEQ: one trial step. 5
1.2 Brown's method vs. Brent's method for two linear equations. 9

4.1 Flow chart for the proof of Lemma 4.10. 65

Tables

5.1 Three initial radii strategies. Starting point 1. 84

5.2 Three initial radii strategies. Starting point 2. 85

5.3 Multilevel vs. LMDER: function evaluations. 86

5.4 Order sensitivity. 90

Chapter 1

Preliminaries

This study proposes generalizations of the methods of Brown and Brent for solving systems of nonlinear equations; i.e.,

Problem NLEQ:

$$\text{Given } F : \Re^n \to \Re^n,$$
$$\text{find } x^* \in \Re^n, \text{ such that } F(x^*) = 0,$$

where F has Lipschitz continuous first derivatives.

The algorithms for solving systems are further generalized to solve the nonlinear equality constrained optimization problem; i.e.,

Problem EQC:

$$\text{minimize } f(x)$$
$$\text{subject to } C(x) = 0,$$

where $f : \Re^n \to \Re$ and $C : \Re^n \to \Re^m (m \leq n)$ are at least twice continuously differentiable. For this problem, a procedure similar to the one for nonlinear equations is used as a scheme for finding feasible points.

When reduced to their mathematical formulations, many problems of science and engineering may be stated as nonlinear optimization problems or problems that require solving nonlinear systems of equations. Much successful research has been done in developing numerical methods together with the underlying theory for their solution. However, the appearance of increasingly challenging practical problems coupled with growing computational capabilities require the introduction of algorithms that can solve new problems while making good use of powerful computers. We also hope that the new algorithms described in this thesis will allow researchers and industrial users to solve old problems more effectively, and even to recognize the existence of tractable optimization problems where none have been thought to exist.

The algorithms proposed here can be used to solve any general problem NLEQ or EQC, but we consider the following special set of optimization problems as a target.

Optimization problems that arise in engineering and other industrial applications are often of very large dimensions. Their constraint systems are frequently nearly square, with only a relatively few variables that correspond to actual physical control parameters, while the majority are state or behavior variables for the system. Methods that allow solving large problems by solving several problems of smaller dimension are very important to find.

Frequently, in engineering design and other applications, constraints occur naturally in blocks arranged in a specific order. For instance, in multidisciplinary design optimization the blocks may come from different sources such as various analyses or simulation codes. In other applications, computational resources do not allow the entire constraint system to be considered simultaneously. An optimization strategy that permits arbitrary splitting of constraints into blocks and separate processing of them would be very useful for such applications. For problem NLEQ, local Brown-Brent methods form such a strategy. Their natural extension is the basis for the multilevel algorithms proposed here.

In addition to their flexibility in handling an arbitrary number of constraint blocks, our algorithms are attractive because, following the original Brown-Brent methods in their finite-difference derivative form, they require fewer function evaluations than the Newton's method.

Multidisciplinary design optimization is expected to be the most useful specific application of the proposed algorithms.

The remainder of this chapter introduces the new algorithms in general terms, discusses their historical background, explains the notation, and states some well-known theoretical results that will be used without proof throughout the thesis. Chapter 2 discusses the globalization of Brent's method for solving systems of nonlinear equations. Chapter 3 introduces the multilevel algorithm for equality constrained optimization based on Brent's method. Chapter 4 is devoted to global convergence theory for the multilevel algorithms. Chapter 5 discusses the problems of implementation. Finally, Chapter 6 concludes with descriptions of some applications and plans for future research. Appendix A contains a glossary of the lemmas and theorems; Appendix B contains the list of test problems.

1.1 General Introduction: Multilevel Algorithms for NLEQ and EQC

Let us begin by considering the conventional approach to solving problem NLEQ with a trust region method used as a globalization strategy. A complete step of such an algorithm has the following form:

Algorithm 1.1 The Trust Region Algorithm for Nonlinear Equations

Given $F : \Re^n \rightarrow \Re^n$, $x_c \in \Re^n, \delta_c > 0$,

Do until convergence:

(1) Let s_c be an approximate solution of
minimize $\|F(x_c) + \nabla F(x_c)^T s\|^2$
subject to $\|s\| \leq \delta_c$.

(2) Set $x_+ = x_c + s_c$.

(3) Evaluate x_+. If it is acceptable, set $x_c = x_+$.

(4) Update δ_c.

End

To evaluate the computed iterate, the algorithm uses the merit function $f(x) = \|F(x)\|_2^2$.

This is a special case of the trust region approach to the unconstrained optimization problem:

Problem UNC:

$$\text{minimize } f(x)$$
$$x \in \Re^n.$$

Detailed treatment of the trust region approach to UNC can be found in Dennis and Schnabel [15], Sorensen [34], Moré [25], Moré and Sorensen [26], Powell [19], and Shultz, Schnabel and Byrd [33].

The methods proposed in this dissertation are multilevel algorithms. In the case of nonlinear equations, our scheme differs from the conventional algorithm in that its major iteration involves finding an approximate solution of not one quadratic model over a single restricted region, but a sweep of quadratic models, each approximately minimized over its own trust region. Each model approximates the sum of squares of a block of equations, restricted to certain subspaces. Each model is computed at a different point. The case of a single block of equations is included.

In general terms, a multilevel algorithm for nonlinear equations with the trust region strategy will have the following form:

Algorithm 1.2 The General Multilevel Trust Region Algorithm for Nonlinear Equations

Given $F : \Re^n \rightarrow \Re^n$, let the components of F be partitioned into M arbitrary blocks denoted by $F_1, \ldots, F_M$. Let $\delta_1^c > 0, \ldots, \delta_M^c > 0, x_c \in \Re^n$ be given.

Do until convergence:

(1) Compute the trial step $\hat{s}_c$:

Set $y_0 = x_c$.

Do $k = 1, M$:

Let s_k be an approximate solution of the subproblem

minimize $\|F_k(y_{k-1}) + \nabla F_k(y_{k-1})^T s\|^2$

subject to $\|s\| \leq \delta_k^c$ and

s is restricted to an appropriate subspace.

Set $y_k = y_{k-1} + s_k$.

End Do

Set $\hat{s}_c = s_1 + \ldots + s_M$.

(2) Set $x_+ = x_c + \hat{s}_c$.

(3) Evaluate x_+. If it is acceptable, set $x_c = x_+$.

(4) Update $\delta_1^c, \ldots, \delta_M^c$.

End

Thus the principal difference between Algorithm 1.1 and 1.2 is in the computation of the trial step:

- In Algorithm 1.1, the trial step is computed using the function and derivative information for the equation components at a single point.

- In Algorithm 1.2, the trial step is computed using the function and derivative information for the equation components at different points.

This difference makes the conventional merit function $\|F(x)\|_2^2$ used in Algorithm 1.1 inadequate for measuring the progress of Algorithm 1.2 toward a solution. We will introduce a new merit function in Chapter 2 and discuss the reasons for introducing it in more detail. As a consequence, steps 3 of the two algorithms also differ.

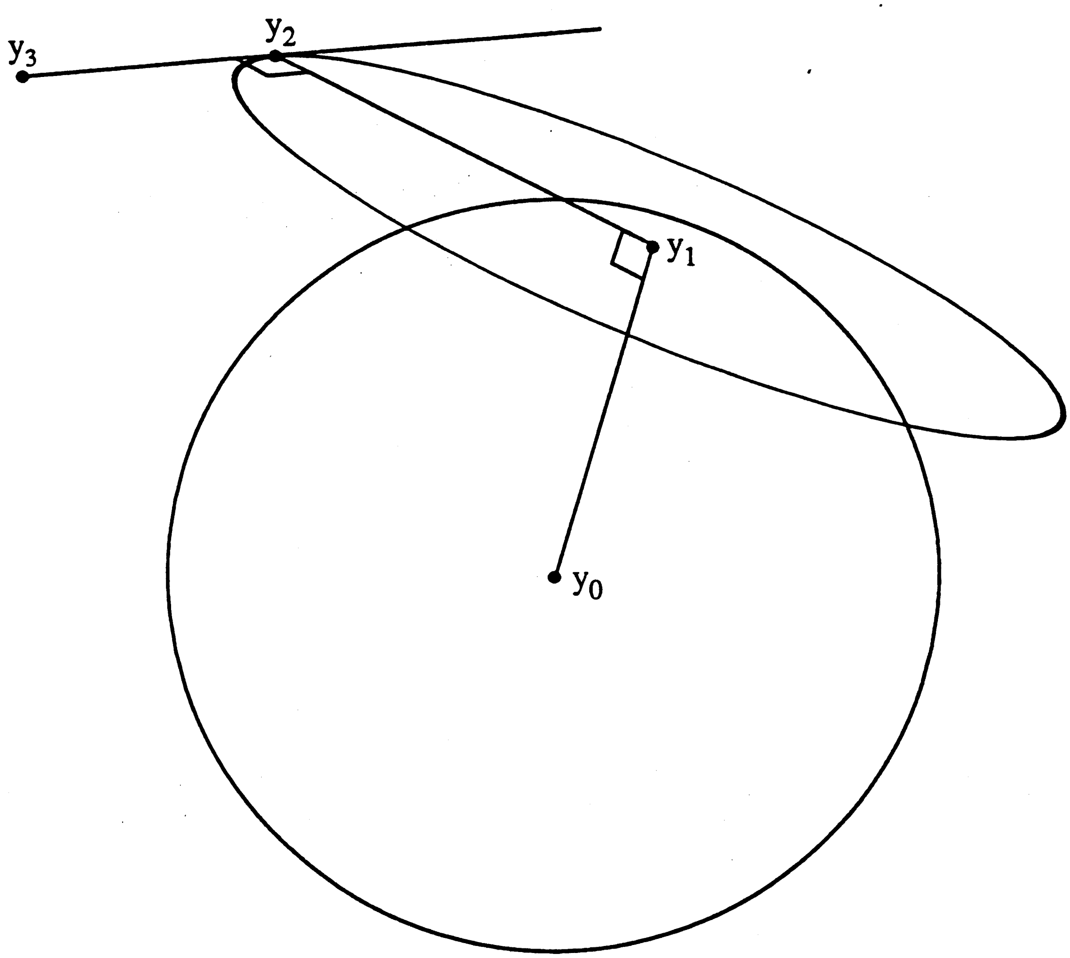

Figure 1.1 Multilevel algorithm for NLEQ: one trial step.

Figure 1.1 illustrates the computation of one trial step of the multilevel algorithm for problem NLEQ with $n = 3, M = 3$:

At $y_0 = x_c$, the model of F_1 at y_0 is minimized over the three-dimensional trust region, producing y_1. Then the model of F_2 at y_1, restricted to the nullspace of $\nabla F_1(y_0)^T$, is minimized in a two-dimensional region to produce y_2. Finally, the model of F_3 at y_2, restricted to the intersection of the null spaces of $\nabla F_1(y_0)^T$ and $\nabla F_2(y_1)^T$, is minimized in a one-dimensional region to yield $y_3 = x_+$.

In the case of equality constrained optimization, the m components of the constraint system are partitioned into M arbitrary blocks. Then a procedure similar to the one for problem NLEQ is used on the constraint blocks. However, when the norm of the model of the last block of constraints is approximately minimized to yield y_M, only m degrees of freedom of the problem are exhausted, leaving $n-m$ degrees of freedom. Then the reduced model of the objective function is approximately minimized over that $(n-m)$-dimensional region to produce $y_{M+1} = x_+$.

In addition to practical considerations discussed in the previous section, this thesis offers several theoretical contributions:

1. We present a global convergence theory for an algorithm based on Brent's method. The algorithm and its theory may be viewed as a generalization of several other algorithms (see Section 1.2).

2. To the author's knowledge, the proposed algorithm for solving nonlinear equations is the first theoretically supported method for globalizing Brent's algorithm via the trust region strategy.

3. In both the algorithms for NLEQ and EQC, the choice of the merit function and the definition of the predicted reduction in the local quadratic model play a crucial role. We introduce new merit functions based on the ℓ_2 penalty function with a multistage penalty parameter updating scheme using the method proposed by El-Alem [8], [9].

4. The work may be considered as a foundation for the study of the general multilevel optimization problem. This will be discussed in more detail in Chapter 6.

1.2 Historical Background

The algorithms proposed here may be viewed as a combination and generalization of the null space methods for solving optimization problems and the generalized elimination methods (Brown-Brent) for solving nonlinear equations.

1.2.1 The Methods of Brown and Brent

Theoretical origins of this research lie in the method for solving nonlinear systems of equations introduced by Brown in [4], [5], [6]. In [3], Brent viewed Brown's method from a different perspective, which allowed Brent to propose a class of methods, among which Brown's original method was a special case. Gay [12] and Martinez [22], [23] provided further modifications and generalizations of the methods.

The following statement of the general Brown-Brent algorithm was condensed from the descriptions in Gay [12] and Dennis [14]. In these works the algorithm is described in terms of one-dimensional blocks.

Denote the components of $F(x)$ by $F_1(x), \ldots, F_n(x)$.

Algorithm 1.3 Local Brown-Brent Algorithm for Nonlinear Systems

Outer Loop: Do until convergence:

$y_0 = x_c$

$H_0 = \Re^n$

Inner Loop: Do $k = 1, n$

1. Form the linearization, L_k about y_{k-1} of F_k restricted to $\bigcap_{i=0}^{k-1} H_i$.
   $L_k = 0$ defines H_k, an $(n-k)$-dimensional hyperplane in $\Re^n$.
2. Move from $y_{k-1} \in \bigcap_{i=0}^{k-1} H_i$ to $y_k \in \bigcap_{i=0}^{k} H_i$.

End Inner Loop

$x_c = y_n$

End Outer Loop

The point y_n of intersection of all the hyperplanes is the point where all the linearizations vanish. The way in which the steps 1-2 of the inner loop are actually done determines the particular kind of Brown-Brent method. In Brent's method, $s_k = y_k - y_{k-1}$ is the shortest ℓ_2 norm step from y_{k-1} to H_k. In Brown's method, s_k is the shortest ℓ_2 norm step from y_{k-1} to H_k parallel the k-th coordinate axis.

When applied to a linear system of equations, i.e., when $F(x) = Ax - b$, Brown's method is equivalent to Gaussian elimination with pivoting about the maximum row element of the reduced matrix [4], while Brent's method is equivalent to factoring A into a product of a lower triangular matrix and an orthogonal matrix [3].

Figure 1.2 illustrates the difference between the two methods.

Brown [4], [6], Brown and Dennis [7], Brent [3], and Gay [12] established local quadratic convergence of variants of the algorithm, both for analytic and finite difference derivatives.

An important feature of the methods is that in their finite-difference derivative form, they require fewer function evaluations than Newton's method. The savings in computation for problem NLEQ can be demonstrated as follows.

The finite-difference Newton's method requires $n+1$ evaluations of each function component $F_k(x)$, resulting in n^2+n component function evaluations. In contrast, each inner loop of a local Brown-Brent method requires $n+1$ evaluations of $F_1(x)$, n evaluations of $F_2(x)$, $\ldots$, 2 evaluations of $F_n(x)$, resulting in $\frac{n^2+3n}{2}$ component function evaluations.

The above description suggests that Brown-Brent based methods perform more efficiently when the linear or the most linear components are placed first in the system. Numerical experience confirms this.

1.2.2 Null Space Methods

The multilevel methods proposed here may be viewed as a generalization of an approach to nonlinear programming known as the null space or generalized elimination approach (see Fletcher [11]).

Different authors refer to different methods as "null space methods", but the general idea of a null space method for equality constrained minimization is to reduce the dimension of the problem by first taking the step intended to solve the constraint equations, and then to minimize the model of the function restricted to the null space of the linearized constraints. The resulting minimization problem is of a lower dimension than the original one.

A well-known local method of this type is the GRG (Generalized Reduced Gradient) algorithm. Details of GRG and other null space methods can be found in Lasdon [17], Fletcher [11], Avriel [1], and Gill, Murray and Wright [29].

A class of global trust region algorithms that use the same general principle of reducing the problem's dimension is known as the class of tangent space methods.

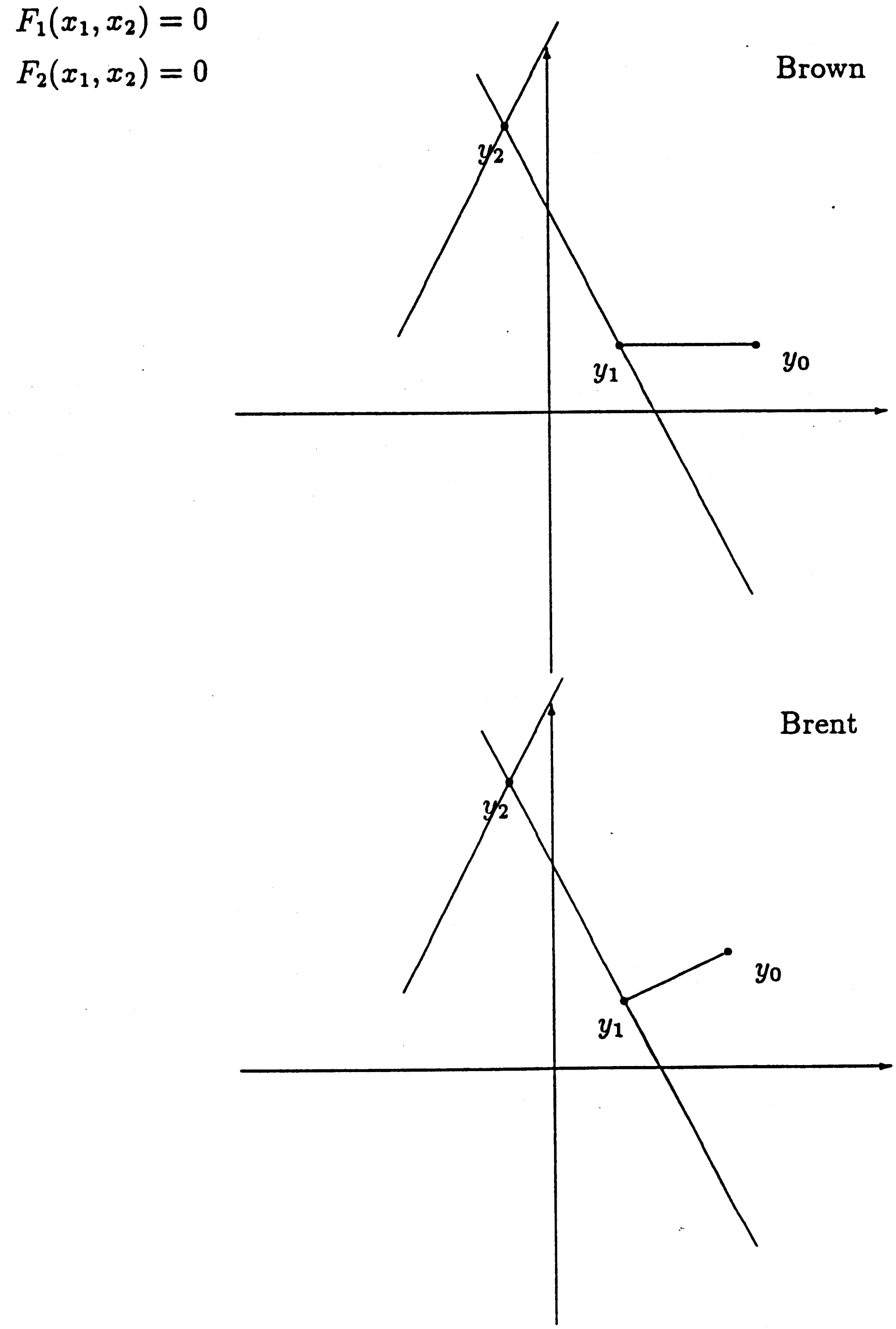

Figure 1.2 Brown's method vs. Brent's method for two linear equations.

The tangent space approach was introduced to avoid the possibility of infeasibility of the constrained trust region subproblem.

Recent work on these methods by Maciel [20] and Dennis, El-Alem and Maciel [16] will be discussed in more detail in Chapter 3. The main feature of the class is that the trial step is computed as a sum of two substeps, the first of which is made toward the linearized constraints in the direction orthogonal to the null space of the constraint Jacobian, while the second is made to minimize the model of the Lagrangian in the null space of the linearized constraints. The function and derivative information is computed at a single point x_c.

The multilevel methods proposed here generalize the tangent space methods in the sense that their trial steps are sums of not two substeps but of as many substeps as there are constraint blocks together with a substep on the model of the objective function with the model information computed at the points resulting from taking the substeps one-by-one.

1.3 Theoretical Background

The global convergence theory for equality constrained optimization algorithms has its basis in the theory for unconstrained optimization. Problem UNC is of direct interest here because the constrained trust region subproblems of the multilevel algorithms are converted into unconstrained trust region subproblems.

In this section we give some well-known results in unconstrained minimization theory without proof, state some useful lemmas from multivariate calculus and linear algebra, and explain the notation. These results can be obtained from numerous sources—the references stated for each result are not exhaustive.

1.3.1 Unconstrained Minimization Results

The solution of the following quadratic subproblem serves as a trial step for problem UNC:

$$\begin{aligned} &\text{minimize } f(x_c) + \nabla f(x_c)^T s + \frac{1}{2} s^T H_c s \\ &\text{subject to } \|s\| \le \delta_c, \end{aligned} \tag{1.1}$$

where $f, \delta_c \in \Re, \nabla f, s \in \Re^n, H_c = H_c^T \in \Re^{n \times n}$, and $\|\cdot\|$ denotes the ℓ_2 norm.

The solution is given by the following lemma.

Lemma 1.1 Let $f : \Re^n \rightarrow \Re$ be twice continuously differentiable; let H_c be symmetric positive definite. Then problem (1.1) is solved by

$$s(\mu) \equiv -(H_c + \mu I)^{-1} \nabla f(x_c) \tag{1.2}$$

for the unique μ such that $\|s(\mu)\| = \delta_c$, unless $\|s(0)\| \leq \delta_c$, in which case $s(0) = -H_c^{-1} \nabla f(x_c)$ is the solution.

Reference: Dennis and Schnabel [15], p. 131.

The subproblem for the nonlinear least squares problem (NLSQ) is a special case of Subproblem (1.1). It has the following form:

$$\begin{aligned} &\text{minimize } \|R(x_c) + \nabla R(x_c)^T s\|_2^2 \\ &\text{subject to } \|s\| \leq \delta_c, \end{aligned} \tag{1.3}$$

where $R : \Re^n \rightarrow \Re^m$ for some $m \leq n$. The solution is given by

Lemma 1.2 The problem NLSQ is solved by

$$s(\mu) \equiv -(\nabla R(x_c) \nabla R(x_c)^T + \mu I)^{-1} \nabla R(x_c) R(x_c), \tag{1.4}$$

where $\mu = 0$ if $\|(\nabla R(x_c) \nabla R(x_c)^T)^{-1} \nabla R(x_c) R(x_c)\| \leq \delta_c$, and $\mu > 0$ otherwise.

References: Dennis and Schnabel [15], p. 227; Moré [24].

The step $s(0)$ is defined by the limiting process

$$s(0) \equiv \lim_{\mu \rightarrow 0^+} s(\mu) = -[\nabla R(x_c)^T]^\dagger R(x_c), \tag{1.5}$$

where $\dagger$ denotes the pseudoinverse of ∇R^T.

The step (1.4) is known as the Levenberg-Marquardt step. It was introduced by Levenberg [18] and Marquardt [21].

The subproblems of the multilevel algorithm for nonlinear equations and for the equality constraint component of the multilevel optimization algorithm are of the form (1.3).

The following lemma describes a useful property of the Levenberg-Marquardt step.

Lemma 1.3 The steps (1.4) and (1.5) are orthogonal to all vectors in the null space of $\nabla R(x_c)^T$.

Reference: Dennis and Schnabel [15], p. 154.

The Merit Function

In order to evaluate a trial step, algorithms use merit functions, which are functions related to the problem in such a way that the improvement in the merit function signifies progress toward the solution of the problem.

For unconstrained minimization, a natural choice for a merit function is the objective function itself. Let

$$\phi(s) = f(x_c) + \nabla f(x_c)^T s + \frac{1}{2} s^T H_c s \tag{1.6}$$

denote the quadratic model of the merit function. We define two related functions.

Definition 1.1 **The actual reduction** is defined as

$$ared_c(s_c) = f(x_c) - f(x_c + s_c), \tag{1.7}$$

and **the predicted reduction** is defined as

$$\begin{aligned} pred_c(s_c) &= \phi(0) - \phi(s_c) \\ &= -\nabla f(x_c)^T(s_c) - \frac{1}{2} s_c^T H_c s_c, \end{aligned} \tag{1.8}$$

so that the predicted reduction in the merit function is an approximation to the actual reduction in the merit function.

The standard way to evaluate the trial step in trust region methods is to consider the ratio of the actual reduction to the predicted reduction. A value lower than a small predetermined value causes the step to be rejected. Otherwise the step is accepted.

Fraction of Cauchy Decrease

For practical considerations, it is important to know how exactly the quadratic subproblem must be solved. For theoretical considerations, it is enough to impose the mildest possible restriction on the step. Usually the trial step is required to satisfy at least the Fraction of Cauchy Decrease condition. Sometimes, to strengthen the global convergence results, the step is required to satisfy a Fraction of Optimal Decrease condition. The latter condition is outside the scope of this thesis. To satisfy the Fraction of Cauchy Decrease condition, s_c must predict at least a fraction of the decrease predicted by the Cauchy step, which is the steepest descent step for the model within the trust region. We must have

$$\phi(s_c) - \phi(0) \le \kappa[\phi(s_c^{CP}) - \phi(0)], \tag{1.9}$$

where

$$s_c^{CP} = -\alpha_c^{CP}\nabla f(x_c) \text{ with}$$

$$\alpha_c^{CP} = \begin{cases} \frac{\|\nabla f(x_c)\|^2}{\nabla f(x_c)^T H_c \nabla f(x_c)} & \text{if } \frac{\|\nabla f(x_c)\|^3}{\nabla f(x_c)^T H_c \nabla f(x_c)} \leq \delta_c \\ \frac{\delta_c}{\|\nabla f(x_c)\|} & \text{otherwise.} \end{cases}$$

Reference: See Dennis and Schnabel [15], pp. 139—141, for details on the Cauchy point.

The Fraction of Cauchy Decrease property implies a weaker condition which has a more convenient form and is frequently used as a technical lemma in the global convergence proofs.

Lemma 1.4 Let s_c satisfy (1.9). Then

$$\phi(0) - \phi(s_c) \geq \frac{\kappa}{2}\|\nabla f(x_c)\| \min\left\{\frac{\|f(x_c)\|}{\|H_c\|}, \delta_c\right\}. \qquad (1.10)$$

References: Powell [19]; Moré [25].

Either (1.9) or (1.10) is necessary to establish global convergence theoretically.

We conclude this section with Powell's global convergence theorem [19] for any unconstrained minimization trust region algorithm.

Theorem 1.1 Let $f : \Re^n \to \Re$ be continuously differentiable and bounded below on the level set $\{x \in \Re^n | f(x) \leq f(x_0)\}$. Assume that $\{H_i\}$ are uniformly bounded above. Let $\{x_i\}$ be the sequence of iterates generated by a trust region algorithm that satisfies (1.9) or (1.10). Then

$$\liminf_{i\to\infty} \|\nabla f(x_i)\| = 0.$$

Detailed treatment of the unconstrained minimization theory and practice can be found in Moré [25], Moré and Sorensen [26], Sorensen [34], and Shultz, Schnabel and Byrd [33].

1.3.2 Analysis and Algebra

Definition 1.2 *(Penrose)* If A is an $m \times n$ matrix then its pseudoinverse $A^\dagger$ is defined as the unique $n \times m$ matrix such that

1. $AA^\dagger A = A$;

2. $A^\dagger A A^\dagger = A^\dagger$;
3. $(AA^\dagger)^T = AA^\dagger$;
4. $(A^\dagger A)^T = A^\dagger A$.

Reference: Campbell and Meyer [32], p. 9.

The following property of the pseudoinverse is the most important one for our purposes.

Lemma 1.5 Let $A \in \Re^{m\times n}, b \in \Re^m$. Then $A^\dagger b$ is the minimum least square (ℓ_2 norm) solution of the problem

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|x\|_2^2 \\ &\text{subject to } x \in \ \text{argmin } \{\|Ax - b\|_2^2\}. \end{aligned} \tag{1.11}$$

Reference: Campbell and Meyer [32], p. 28.

The following lemmas are available from many sources; for example, Dennis and Schnabel [15], and Ortega and Rheinboldt [28]. They are necessary for producing estimates in the proofs and will be stated without additional comments.

Lemma 1.6 Let $F : \Re^n \to \Re^m$ be differentiable in an open convex set $D \subset \Re^n$. Then for any $x, y \in D$,

$$\|F(y) - F(x)\| \le \sup_{0\le t\le 1} \|F'(x + t(y - x))\|\|x - y\|. \tag{1.12}$$

The notation $x \in (x_1, x_2)$ for $x, x_1, x_2 \in \Re^n$ means that $x = x_1 + t(x_2 - x_1)$ for some $t \in (0, 1)$.

Lemma 1.7 Let $f : \Re^n \to \Re$ be differentiable in an open convex set $D \subset \Re^n$. Then for $x \in D$ and any nonzero perturbation $p \in \Re^n$, there exists $z \in (x, x + p)$ such that

$$f(x + p) = f(x) + \nabla f(z)^T p. \tag{1.13}$$

Lemma 1.8 Let $f : \Re^n \to \Re$ be twice continuously differentiable in an open convex set $D \subset \Re^n$. Then for $x \in D$ and any nonzero perturbation $p \in \Re^n$, there exists $z \in (x, x + p)$ such that

$$f(x + p) = f(x) + \nabla f(x)^T p + \frac{1}{2} p^T \nabla^2 f(z) p. \tag{1.14}$$

1.3.3 Notation

From here on, we assume that the equations in problem NLEQ or equality constraints in problem EQC are partitioned into M arbitrary blocks.

Every algorithm proposed here has an outer loop with the loop counter i and an inner loop with the loop counter k. Thus k corresponds to the block number of equations for problem NLEQ or constraints for problem EQP. If the subscript k is used with a constant, that constant refers to the properties of the k-th block of equations or constraints.

Minimization subproblems are solved within the inner loop, yielding intermediate steps. The sum of the intermediate steps produces the total step, the acceptability of which is determined in the outer loop. The following notation is used in the rest of the study.

$\|\cdot\|$ — Unless specified otherwise, all norms are ℓ_2 norms.

x_c, x_+ — Current and next outer loop iterates.

x_i, x_{i+1} — Current and next outer loop iterates, when considered in convergence analysis.

x_* — A local solution of the original problem.

y_k, $k = 0, \ldots, M$ — ($k = 0, \ldots, M+1, M \leq m \leq n$ for optimization.) Inner loop iterates when considered without reference to outer iterations.

y_k^c, y_k^+ — Inner loop iterates when considered with reference to outer iterations.

y_k^i, y_k^{i+1} — Inner loop iterates when considered with reference to outer iterations in convergence analysis.

s_k, $k = 1, \ldots, M$ — ($k = 1, \ldots, M+1, M \leq m \leq n$ for optimization.) Steps computed in the inner loop subproblems.

s_k^i — The subproblem steps when the reference to the outer loop is required.

s_k^c — The same as s_k^i.

s_k^+ — The same as s_k^{i+1}.

s_k^- — The same as s_k^{i-1}.

$\hat{s}_c$ — $\hat{s}_c = s_1 + \ldots + s_M$ in the context of equations, and $\hat{s}_c = s_1 + \ldots + s_{M+1}$ in the context of optimization; $\hat{s}_c$ is the trial step.

δ_k — $k = 1, \ldots, M$ for equations, and $k = 1, \ldots, M+1$ for optimization. This is the trust region radius for subproblem k; δ_k takes on a superscript identical to the ones for s_k when mentioned in the context of the outer loop

$\hat{\delta}_c$ — The radius of the total trust region centered at $x_c = y_0$.

f_c, f_+ — $f(x_c)$, $f(x_+)$.

H_M — In the context of optimization, H_M approximates $\nabla^2 f(y_M)$.

$\phi_M(s)$ — In the context of optimization, this is the quadratic model of f built about the point y_M.

$J_k(x)$ — $\nabla F_k^T(x)$, in the context of equations.

P — A basis for the intersection of the null space of $\nabla C(x)^T$, in the context of optimization.

P_k — A basis for the intersection of the null spaces of $J_1(y_0), \ldots, J_k(y_{k-1})$ in the context of nonlinear equations, and a basis of the intersection of the null spaces of $\nabla C_1(y_0)^T, \ldots, \nabla C_k(y_{k-1})^T$ in the context of optimization.

Q — Special case of P: orthonormal basis for the null space of $\nabla C(x)^T$, in the context of optimization.

Q_k — Special case of P_k: orthonormal basis for the intersection of the null spaces of $J_1(y_0), \ldots, J_k(y_{k-1})$ in the context of nonlinear equations, and an orthonormal basis for the intersection of the null spaces of $\nabla C_1(y_0)^T, \ldots, \nabla C_k(y_{k-1})^T$ in the context of optimization.

Z — Special case of P: reduced basis projector into the null space of $\nabla C(x)^T$, in the context of optimization. (Described in Chapter 5.)

Z_k — Special case of P_k: reduced basis projector into the intersection of the null spaces of $J_1(y_0), \ldots, J_k(y_{k-1})$ in the context of nonlinear equations, and into the intersection of the null spaces of $\nabla C_1(y_0)^T, \ldots, \nabla C_k(y_{k-1})^T$ in the context of optimization. (Described in Chapter 5.)

In general, when we omit superscripts, we refer to the objects within a single outer loop. For example, $C_k(y_{k-1})$ refers to $C_k(y^i_{k-1})$ or $C_k(y^c_{k-1})$.

Additional notation will be introduced as needed.

Chapter 2

Multilevel Algorithm for Solving Nonlinear Equations Based on Brent's Method

2.1 Brent's Algorithm for Nonlinear Equations

One way to present Brent's local method* is as follows:

Algorithm 2.1 Brent's Algorithm for Nonlinear Equations

Let $F \in C^1(\Re^n), x_c$ be given. The components of F are partitioned into M arbitrary blocks.

Outer Loop: Do until convergence:

$y_0 = x_c$.

Inner Loop: Do $k = 1, M$

Solve for s_k:

minimize $\frac{1}{2}\|s\|_2^2$

subject to $s \in$ argmin $\|F_k(y_{k-1}) + J_k(y_{k-1})s\|_2$

$J_j(y_{j-1})s = 0, j = 1, \ldots, k-1.$

$y_k = y_{k-1} + s_k$.

End Inner Loop

$x_c = y_M$.

End Outer Loop

Thus, s_k is the shortest ℓ_2 norm step from the current minor iterate to the hyperplane formed by the restricted linearization of the component block numbered k. The trust region strategy suggests itself as a natural approach to globalizing the local algorithm.

*For simplicity, we use exact derivatives.

2.2 Globalization of Brent's Method: Multilevel Algorithm

The equations are divided into M arbitrary blocks. The inner loop of the multilevel algorithm for problem NLEQ is as follows. The current approximation to a solution of problem NLEQ is y_0. A quadratic Gauss-Newton model of the first block is built about the initial point, y_0, and a step, s_1, bounded by the trust region, is found in such a way that it satisfies a Fraction of Cauchy Decrease condition for this model. The step is taken to yield the point $y_1 = y_0 + s_1$. The quadratic model of the second block of equations, restricted to the null space of the Jacobian of the first block, is built **using the information at the new point.** It is important to emphasize that all the function and derivative information for the second block is computed at the new point y_1. The next step, s_2, bounded by its own trust region, is obtained to satisfy a Fraction of Cauchy Decrease condition for this restricted model of the second block. The step is taken to yield the point y_2. The process of computing steps that satisfy sufficient predicted decrease for the restricted models of progressively smaller dimensions continues. Again, **the model for each block is built by using the function and derivative information at the most recently computed point.** The final step, s_M, is obtained to produce sufficient predicted decrease in the quadratic model, at y_{M-1}, of the last block of equations, restricted to the intersection of the null spaces of the Jacobians of all previous blocks. The final step is taken to yield the next major iterate. The total step from x_c to x_+ is the sum of the substeps in the inner sweep, i.e., $\hat{s}_c = s_1 + \ldots + s_M$. Unless the convergence criterion is met, the algorithm returns to process again the first block of equations. The current version of the algorithm evaluates the total step $\hat{s}_c$. It is possible that future versions of the algorithm will also evaluate intermediate substeps $s_1, \ldots, s_M$. Such a strategy may prevent having to return to the first block of equations. Instead, processing may resume at some intermediate inner iterate y_k.

To measure the progress of the algorithm toward a solution, we introduce a new merit function:

$$
\begin{aligned}
&P(x; \rho_1, \ldots, \rho_{M-1}) = \\
&\quad \|F_M(x)\|^2 + \rho_{M-1}(\|F_{M-1}(x)\|^2 + \ldots + \rho_2(\|F_2(x)\|^2 + \rho_1\|F_1(x)\|^2)) \\
&\quad = \|F_M(x)\|^2 + \sum_{k=1}^{M-1} \Big(\prod_{j=k}^{M-1} \rho_j \Big) \|F_k(x)\|^2,
\end{aligned}
$$

where $\rho_k \geq 1$, $k = 1, \ldots, M$. The initial choice $\rho_k = 1$ is arbitrary and scale-dependent. The only requirement is that $\rho_k \geq 1$. For theoretical purposes, the problem is assumed to be well-scaled.

We shall motivate the choice of the merit function in Section 2.2.4.

At $y_M = x_+ = x_c + \hat{s}_c$, we model each $\|F_k(x_+)\|^2$ by $\|F_k(y_{k-1}) + J_k(y_{k-1})s_k\|^2$, and so we model the merit function at x_+ by

$$
\begin{aligned}
&\mathcal{M}_c(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c) = \\
&\quad \|F_M(y_{M-1}) + J_M(y_{M-1})s_M\|^2 + \rho_{M-1}^c(\|F_{M-1}(y_{M-2}) + J_{M-1}(y_{M-2})s_{M-1}\|^2 \\
&\quad +\rho_{M-2}^c(\|F_{M-2}(y_{M-3}) + J_{M-2}(y_{M-3})s_{M-2}\|^2 + \cdots \\
&\quad +\rho_2^c(\|F_2(y_1) + J_2(y_1)s_2\|^2 + \rho_1^c\|F_1(y_0) + J_1(y_0)s_1\|^2))) \\
&\quad = \|F_M(y_{M-1}) + J_M(y_{M-1})s_M\|^2 + \sum_{k=1}^{M-1}(\prod_{j=k}^{M-1} \rho_j)\|F_k(y_{k-1}) + J_k(y_{k-1})s_k\|^2.
\end{aligned}
$$

Algorithm 2.2 Multilevel Algorithm for Nonlinear Equations

Given $\delta_k > 0, k = 1, \ldots, M, \delta_{max} > 0, \delta_{min} > 0, 0 < \eta_1 < \eta_2 < 1, \alpha_1 \in (0, 1]$, $\alpha_2 > 1, x_c \in \Re^n$.

Outer Loop: Do until convergence:

$y_0 = x_c$.

Inner Loop: Do $k = 1, M$

Compute s_k^{Brent}, the unconstrained Brent step from y_{k-1}.

if $\|s_k^{Brent}\| \leq \delta_k$ **then**

$y_k = y_{k-1} + s_k^{Brent}$.

$s_k = s_k^{Brent}$.

else

Compute s_k that satisfies the Fraction of Cauchy Decrease condition on $\frac{1}{2}\|F_k(y_{k-1}) + J_k(y_{k-1})s\|_2^2$ restricted to the intersection of the null spaces of $J_j(y_{j-1})s = 0, j = 1, \ldots, k-1$, and $\|s\|_2 \leq \delta_k$.

$y_k = y_{k-1} + s_k$.

End if

End Inner Loop

$x_+ = y_M$.

$\hat{s}_c = s_1 + \ldots + s_M$.

Update the penalty parameters using Algorithm 2.3.

Evaluate the step and update the trust region radius:

Compute $P(x_c; \rho_1^c, \ldots, \rho_{M-1}^c), P(x_+; \rho_1^c, \ldots, \rho_{M-1}^c)$,
$\mathcal{M}_c(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)$,
$ared = P(x_c; \rho_1^c, \ldots, \rho_{M-1}^c) - P(x_+; \rho_1^c, \ldots, \rho_{M-1}^c)$,
$pred = P(x_c; \rho_1^c, \ldots, \rho_{M-1}^c) - \mathcal{M}_c(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)$.
Use Algorithm 2.4 to evaluate the step and to update the trust region radii.

End Outer Loop

The subproblems of the algorithm can be solved by using the QR decomposition of $J_1(y_0)$ to find a basis for its null space and then by updating the QR decomposition for subsequent subproblems to find a basis, Q_{k-1}, for the intersection of the null spaces of $J_1(y_0), \ldots, J_k(y_{k-1})$. The procedure for finding a basis for the intersection of null spaces is discussed in Chapter 5. A change of variables, $v = Q_{k-1}s$, converts the constrained subproblems to unconstrained ones. In the test version of the algorithm, the reduced quadratic subproblems are solved by the subroutine GQTPAR (Moré and Sorensen [26]). As the inner sweep progresses, smaller and smaller dimensional problems are solved.

We attempt to take the unconstrained Brent step first, because we would like to retain local quadratic convergence of Brent's local algorithm, that is, once an iterate comes sufficiently close to a solution, we would like the algorithm to take unconstrained Brent steps until convergence occurs. At present, we are not certain that our merit function will allow us to recognize this situation. Local convergence analysis is beyond the scope of this thesis, but it is included in the research plans of the immediate future.

Once the subproblems with null space constraints are converted into unconstrained trust-region subproblems, the steps may be chosen in any manner, as long as they satisfy the Fraction of Cauchy Decrease condition. Let us consider two choices: the optimal step and the truncated Brent step.

2.2.1 The Levenberg-Marquardt Step

The following proposition establishes the relation between the Levenberg-Marquardt step and the Brent step.

Proposition 2.1 Let s_k be the Levenberg-Marquardt step from y_{k-1} to y_k with trust region radius δ_k. Then s_k is equal to the unconstrained Brent step on the shifted equation $G_k(y) \equiv F_k(y) - [F_k(y_{k-1}) + J_k(y_{k-1})s_k] = 0$.

Proof:

Let $k = 1$. The Levenberg-Marquardt step from y_0 to y_1 is a solution of the subproblem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|F_1(y_0) + J_1(y_0)s\|^2 \\ \text{subject to} & \|s\| \leq \delta_1. \end{array}$$

By Lemma 1.2, the solution is

$$s_1 = s_1(\mu) = -[J_1(y_0)^T J_1(y_0) + \mu I]^{-1} J_1(y_0)^T F_1(y_0).$$

Now let us compute the Brent step from y_0 on the shifted equation

$$G_1(y) \equiv F_1(y) - [F_1(y_0) + J_1(y_0)s_1] = 0.$$

The linearization of $G_1(y)$ at y_0 is

$$\begin{aligned} L_1(s) &= G_1(y_0) + \nabla G_1(y_0)^T s \\ &= -J_1(y_0)s_1 + J_1(y_0)s, \end{aligned}$$

where $s = y - y_0$. The unconstrained Brent step is the solution of the subproblem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|s\|^2 \\ \text{subject to} & J_1(y_0)s - J_1(y_0)s_1 = 0, \end{array}$$

and by Lemma 1.5 we have

$$s_1^{Brent} = J_1(y_0)^{\dagger} J_1(y_0)s_1.$$

If $s_1(0)$ is inside the trust region, then

$$s_1(0) = -J_1(y_0)^{\dagger} F_1(y_0)$$

and

$$s_1^{Brent} = -J_1(y_0)^{\dagger} J_1(y_0) J_1(y_0)^{\dagger} F_1(y_0) = -J_1(y_0)^{\dagger} F_1(y_0) = s_1(0),$$

by a property of pseudoinverses (Definition 1.3).

Now, for $\mu > 0$,

$$s_1^{Brent} = J_1(y_0)^{\dagger} J_1(y_0)s_1 = -J_1(y_0)^{\dagger} J_1(y_0)[J_1(y_0)^T J_1(y_0) + \mu I]^{-1} J_1(y_0)^T F_1(y_0).$$

By Lemma 1.3, s_1 is orthogonal to the null space of $J_1(y_0)$, i.e., $s_1 \in (\mathcal{N}[J_1(y_0)])^\perp$ or $s_1 \in R(J_1(y_0)^T)$. Therefore,

$$s_1 = J_1(y_0)^\dagger J_1(y_0) s_1 = s_1^{Brent}.$$

So, for both $\mu = 0$ and $\mu > 0$, the Levenberg-Marquardt step from y_0 to y_1 is the same as the unconstrained Brent step on the shifted equation.

Taking the step yields $y_1 = y_0 + s_1$ and y_1 resides in the hyperplane $J_1(y_0)s - J_1(y_0)s_1 = 0$.

Now we linearize $F_2(x) = 0$ restricted to this hyperplane, which must also contain y_2. Thus, s_2 must lie in the null space of $J_1(y_0)$, i.e., it must satisfy

$$J_1(y_0)s = 0$$

In particular, let Q_1 be an orthonormal basis for $\mathcal{N}[J_1(y_0)]$; then change variables to $s = Q_1 v$ and linearize F_2 about y_1:

$$F_2(y_1 + Q_1 v) \sim F_2(y_1) + J_2(y_1)Q_1 v.$$

We obtain s_2 as a Levenberg-Marquardt step on

$$\begin{array}{ll} \text{minimize} & \|F_2(y_1) + J_2(y_1)Q_1 v\|^2 \\ \text{subject to} & \|v\| \le \delta_2, \end{array}$$

i.e.,

$$s_2 = Q_2[(J_2(y_1)Q_1)^T(J_2(y_1)Q_1) + \mu I]^{-1}[J_2(y_1)Q_1]^T F_2(y_1).$$

The shifted equation is

$$G_2(y) \equiv F_2(y) - [F_2(y_1) + J_2(y_1)s_2].$$

Linearizing this equation restricted to $J_1(y_0)s = 0$ and following an argument identical to the one for $k = 1$, we arrive to the conclusion that s_2 is also the unconstrained Brent step on the shifted equation.

Taking into account successive restrictions to $\bigcap_{j=1}^{k-1}[\mathcal{N}(\nabla C_j(y_{j-1})^T)]$, the reasoning is unchanged for any k, which proves the proposition. $\square$

Thus, each subproblem step in the inner sweep is a Brent step from the current minor iterate to the hyperplane formed by the linearization of the next equation, "shifted" to be inside or on the boundary of the trust region.

2.2.2 The Truncated Brent Step

The subproblem steps can also be approximated by attempting the unconstrained Brent step first, keeping it if it is within the trust region and shortening it to lie within the trust region if it is not, i.e.,

$$s_k = \frac{\delta_k * s_k^{Brent}}{\|s_k^{Brent}\|}.$$

This strategy preserves the sufficient predicted decrease property of the step, which is shown in the following proposition.

Proposition 2.2 Let s_k^{Brent} be the unconstrained Brent step from y_{k-1} to y_k with trust region radius δ_k. If $\|s_k^{Brent}\| \leq \delta_k$, then let $s_k = s_k^{Brent}$. Otherwise let

$$s_k = \frac{\delta_k * s_k^{Brent}}{\|s_k^{Brent}\|}. \tag{2.1}$$

Then s_k satisfies the Fraction of Cauchy Decrease condition on subproblem k.

Proof:

If $s_k = s_k^{Brent}$, then the conclusion is obvious. So for $\zeta_k = \frac{\delta_k}{\|s_k^{Brent}\|}$ assume that $0 < \zeta_k < 1$. We can express s_k^{Brent} as $s_k^{Brent} = Q_{k-1}v_k$, where v_k is the Levenberg-Marquardt step for the problem

$$\begin{array}{ll} \text{minimize} & \|F_k(y_{k-1}) + J_k(y_{k-1})Q_{k-1}v\|^2 \\ \text{subject to} & \|v\| \leq \|s_k^{Brent}\|. \end{array}$$

Therefore,

$$\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{Brent}\| \leq \|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{BCP}\|,$$

where $s_k^{BCP} = Q_{k-1}u_k$ and u_k solves

$$\begin{array}{ll} \text{minimize} & \|F_k(y_{k-1}) + J_k(y_{k-1})Q_{k-1}u\|^2 \\ \text{subject to} & \|u\| \leq \|s_k^{Brent}\| \\ & u = -\alpha(J_k(y_{k-1})Q_{k-1})^T F_k(y_{k-1}), \qquad \alpha \geq 0. \end{array}$$

Clearly, since the unconstrained Brent step lies outside the trust region,

$$A = \{s \mid \|s\| \leq \delta_k, s = -\alpha J_k(y_{k-1})^T F_k(y_{k-1}), \alpha \geq 0\}$$

is a proper subset of

$$B = \{s \mid \|s\| \leq \|s_k^{Brent}\|, s = -\alpha J_k(y_{k-1})^T F_k(y_{k-1}), \alpha \geq 0\},$$

and since

$$\min \{\|F_k(y_{k-1}) + J_k(y_{k-1})s\|, s \in A\} \geq \min \{\|F_k(y_{k-1}) + J_k(y_{k-1})s\|, s \in B\},$$

we have

$$\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{BCP}\| \leq \|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{CP}\|,$$

which implies that

$$\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{Brent}\| \leq \|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{CP}\|,$$

Now we have

$$\begin{aligned}
\|F_k(y_{k-1}) + J_k(y_{k-1})s_k\| &= \|F_k(y_{k-1}) + J_k(y_{k-1})\zeta_k s_k^{Brent}\| \\
&= \|F_k(y_{k-1}) + J_k(y_{k-1})\zeta_k s_k^{Brent} \\
&+ \zeta_k F_k(y_{k-1}) - \zeta_k F_k(y_{k-1})\| \\
&\leq \zeta_k\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{Brent}\| + (1-\zeta_k)\|F_k(y_{k-1})\| \\
&\leq \zeta_k\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{BCP}\| + (1-\zeta_k)\|F_k(y_{k-1})\| \\
&\leq \zeta_k\|F_k(y_{k-1}) + J_k(y_{k-1})s_k^{CP}\| + (1-\zeta_k)\|F_k(y_{k-1})\|
\end{aligned}$$

which proves the proposition. □

For very large problems, shortening the unconstrained Brent step may be preferable to solving quadratic subproblems.

2.2.3 Updating the Penalty Parameters

To update the penalty function parameters, we use the following procedure.

Algorithm 2.3 Penalty Parameter Updating Algorithm (Done on completion of each inner sweep of minimization problems.)

At the beginning of Algorithm 2.2, set $\rho_1^- = \ldots = \rho_{M-1}^- = 1$ and choose $\beta \in (0,1)$.

1. Compute $Fpred_1(s_1) = \|F_1(y_0)\|^2 - \|F_1(y_0) + J_1(y_0)s_1\|^2$
2. Compute $Fpred_2(s_1, s_2; \rho_1^-) =$

   $\|F_2(y_0)\|^2 - \|F_2(y_1) + J_2(y_1)s_2\|^2 + \rho_1^- Fpred_1(s_1)$

   if $Fpred_2(s_1, s_2; \rho_1^-) \geq \frac{\rho_1^-}{2}[\|F_1(y_0)\|^2 - \|F_1(y_0) + J_1(y_0)s_1\|^2]$ **then**

   $\rho_1^c = \rho_1^-$

   $Fpred_2(s_1, s_2; \rho_1^c) = Fpred_2(s_1, s_2; \rho_1^-)$

else

$\rho_1^c = \bar{\rho}_1 + \beta,$

where $\bar{\rho}_1 = \frac{2[\|F_2(y_1)+J_2(y_1)s_2\|^2 - \|F_2(y_0)\|^2]}{Fpred_1(s_1)}$

Compute $Fpred_2(s_1, s_2; \rho_1^c)$

end if

3. Compute $Fpred_3(s_1, s_2, s_3; \rho_1^c, \rho_2^-) =$

$[\|F_3(y_0)\|^2 - \|F_3(y_2) + J_3(y_2)s_3\|^2] + \rho_2^- Fpred_2(s_1, s_2; \rho_1^c)$

if $Fpred_3(s_1, s_2, s_3; \rho_1^c, \rho_2^-) \geq \frac{\rho_2^-}{2} Fpred_2(s_1, s_2; \rho_1^c)$ **then**

$\rho_2^c = \rho_2^-$

$Fpred_3(s_1, s_2, s_3; \rho_1^c, \rho_2^c) = Fpred_3(s_1, s_2, s_3; \rho_1^c, \rho_2^-)$

else

$\rho_2^c = \bar{\rho}_2 + \beta,$

where $\bar{\rho}_2 = \frac{2[\|F_3(y_2)+J_3(y_2)s_3\|^2 - \|F_3(y_0)\|^2]}{Fpred_2(s_1, s_2; \rho_1^c)}$

Compute $Fpred_3(s_1, s_2, s_3; \rho_1^c, \rho_2^c)$

end if

Continue the process until all $M - 1$ penalty parameters are updated.

End

This penalty parameter updating scheme generalizes the scheme proposed in El-Alem [8], [9]. It ensures that our merit function has an essential property, namely, that unless an iterate is optimal, the predicted reduction should always be positive.

Note that without updating the penalty parameters we can be assured of the positive predicted reduction from x_c only for the first block of equations, i.e., only $Fpred_1(s_1)$ is definitely positive without additional considerations. To ensure that $Fpred_2(s_1, s_2; \rho_1)$ is positive, we may have to increase ρ_1. Now that $Fpred_2(s_1, s_2; \rho_1)$ is positive, we use it to ensure that the next partial predicted reduction is positive, and so on. Thus the predicted reduction of the first block is the most heavily penalized one. This is another argument for placing the linear equations or constraints first, if at all possible.

The properties of the penalty parameters are discussed in more detail in the next two chapters.

It should be emphasized that the step computation is completely independent of the penalty parameter computation.

2.2.4 Motivation for the Merit Function

Once the connection between the truncated Brent step and the Levenberg-Marquardt step became clear, the statement of the algorithm followed naturally. However, discovering an appropriate merit function required some effort.

Our original intention was to construct a good algorithm and to apply existing theory, e.g., Shultz, Schnabel and Byrd [33], to prove its global convergence. To accomplish this, we wanted our merit function to have the property that the predicted reduction associated with a step be the reduction in the merit function caused by this step when the merit function is applied to the model problem.

Toward this aim, we conjectured that the conventional merit function $f(x) = \frac{1}{2}\|F(x)\|_2^2$ would have the required property. Since the underlying problem was unchanged this conjecture seemed reasonable. So, an explanation of why the conventional merit function has proved inadequate is in order.

The first indication that we were too optimistic came with the question, "What is our model problem?" To answer it, we attempted to express the model of $f(x)$ in the form of a quadratic. We approximated our merit function at x_+ by $\Psi_c(s_1, \ldots, s_M) \equiv \sum_{k=1}^{M} \|F_k(y_{k-1}) + J_k(y_{k-1})s_k\|^2$ and noticed that

$$\begin{aligned}
\Psi_c(s_1, \ldots, s_M) &= \sum_{k=1}^{M} \|F_k(y_{k-1}) + J_k(y_{k-1})s_k\|^2 \\
&= \sum_{k=1}^{M} \|F_k(y_{k-1}) + J_k(y_{k-1})(s_k + \ldots + s_M)\|^2 \\
&= \|\bar{F}_c + \bar{J}_c\hat{s}_c\|^2
\end{aligned}$$

where

$$\bar{F}_c = [F_1(y_0), F_2(y_1) - J_2(y_1)s_1, \ldots, F_M(y_{M-1}) - J_M(y_{M-1})(s_1 + \ldots + s_{M-1})]^T$$

and

$$\bar{J}_c = [J_1(y_0), J_2(y_1), \ldots, J_M(y_{M-1})]^T.$$

So, consider the quadratic

$$\bar{\Psi}_c(s) = \frac{1}{2}\|\bar{F}_c + \bar{J}_c s\|^2,$$

where s is the step from $x_c = y_0$ to $x_+ = y_M$. With the merit function $f(x) = \frac{1}{2}\|F(x)\|_2^2$, the actual and predicted reductions become $ared = f_c - f_+$, $pred = \bar{\Psi}_c(0) -$

$\bar{\Psi}_c(\hat{s}_c)$. This merit function / model combination seemed promising, but we could not prove that the total step, $\hat{s}_c$, produced a Fraction of Cauchy Decrease in the model.

Other definitions of the model and predicted reduction have been attempted. As a rule, several of the main ingredients needed for a global convergence proof could be proven, but not all of them.

We believe that these difficulties arise because the conventional merit function does not take into consideration the multilevel structure of the algorithm, in particular, that the equation blocks are modelled at different points. The function $f(x) = \frac{1}{2}\|F(x)\|_2^2$ treats all blocks equally, without taking into consideration the order in which minimization proceeds.

The problem can be summarized as follows:

- The result of the k-th minimization subproblem predicts decrease for the k-th component from point y_{k-1} to point y_k. It predicts no change for all previous blocks. However, there is no prediction at all about how $s_1 + \ldots + s_k$ changes and likely increases the norms of the blocks numbered $k+1, \ldots, M$.

The above mentioned attempts to apply existing theory to our algorithm brought us to the conclusion that the merit function must take into account the multilevel structure of the scheme. The modified ℓ_2 penalty function, described in the previous section does just that, by penalizing for the possible predicted increase in the equation blocks $k, \ldots, M$, that may have occured during inner loop iterations $1, \ldots, k-1$.

2.2.5 Step Evaluation and Trust Region Updating

It is an interesting and challenging aspect of the proposed algorithms that they can accommodate (theoretically, if not practically) several ways to evaluate the trial step. As a related issue, several strategies for updating the trust region radius are also possible. This subject will be discussed in more detail in the chapter on implementation. Here we would like to point out the possibilities and to say that there is a strong indication that no single strategy may solve all problems equally well.

In the current version of the algorithm, the total step, $\hat{s}_c$, is evaluated via the merit function outside the inner loop. It is also possible to use the merit function each substep within the inner loop or to use a hybrid strategy with both the total step evaluation outside the inner loop and the substep evaluation within inner loop.

Possible trust region radius updating strategies follow.

1. In the first scheme, all radii are updated simultaneously by the same factor. A competing strategy suggests simultaneous updating of all radii by different factors, based on individual subproblem performance.

2. In the second scheme, simultaneous updating of all radii, following an inner sweep in which all subproblems are solved once, is contrasted with individual updating of the radii within an inner sweep in which subproblems may be solved several times to achieve a required actual to predicted reduction ratio for all subproblems. The latter strategy seems reasonable because it treats the radii of the subproblems individually, based on their possibly wildly varying behavior.

For the purposes of this thesis, the following strategy is used:

- **The total trial step is evaluated outside the inner loop.**
- **All individual trust region radii are equal and are updated simultaneously by the same factor.**

In the context of equations, the last statement means that the total trust region radius centered at x_c satisfies

$$\hat{\delta}_c = \sqrt{M}\,\delta_k \text{ for all } k. \tag{2.2}$$

In the context of optimization, we have

$$\hat{\delta}_c = \sqrt{M+1}\,\delta_k \text{ for all } k. \tag{2.3}$$

The values δ_{min} and δ_{max} refer, respectively, to the smallest and largest values allowed for the subproblem trust region radii.

We would like to emphasize that the simultaneous expansion or contraction of the trust region radii is not a technical requirement. It is just a current updating strategy, and it is convenient for the purposes of the global convergence proof. However, all that the convergence theory requires is that the norm of the total step $\hat{s}_c = s_1 + \ldots + s_M$ be less than or equal to $\hat{\delta}_c$, while the norm of each substep $s_k, k = 1, \ldots, s_M$ is less than or equal to some fraction of $\hat{\delta}_c$.

The algorithm for evaluating the step and updating the trust region radii follows.

Algorithm 2.4 Step Evaluation / Trust Region Update

Given $\delta_k > 0, k = 1, \ldots, M$ (or $k = 1, \ldots, M+1$ for optimization), $\delta_{max} > 0, \delta_{min} > 0, 0 < \eta_1 < \eta_2 < 1, \alpha_1 \in (0, 1]$, $\alpha_2 > 1, x_c \in \Re^n$, *ared*, *pred*,

Compute $r = \frac{ared}{pred}$.
if $r < \eta_1$ **then** (step not accepted)
  $\delta_k = \alpha_1 * \delta_k$.
else if $r \geq \eta_2$ **then** (step accepted)
  $\delta_k = \min\{\delta_{max}, \max\{\delta_{min}, \alpha_2 * \delta_k\}\}$.
  $x_c = x_+$.
else (step accepted)
  $\delta_k = \max\{\delta_{min}, \delta_k\}$.
  $x_c = x_+$.
end if

We note that if the step is not accepted, the trust region radii are decreased without any safeguard. However, if the step is accepted, the next trust region radius is set to be no smaller than a predetermined positive value δ_{min}. This strategy is extremely important in the global convergence theory. It ensures that the trust region radius is bounded away from zero and hence that the penalty parameters are bounded from above. This technique was introduced in Maciel [20].

2.2.6 Convergence Properties

Let $\{x_i\}$ denote the sequence of iterates generated by Algorithm 2.2, and let $\{\hat{s}_i\}$ be the corresponding sequence of steps.

The following assumptions are made for the algorithms for solving nonlinear systems of equations:

AE1: There exists a convex set, $\Omega \in \Re^n$, such that x_i and $x_i + \hat{s}_i$ are in Ω for all i.

AE2: $J_k(x), k = 1, \ldots, M$, are uniformly bounded in norm in Ω.

AE3: $J(x)$ has full rank for all $x \in \Omega$. This implies that $J_k(x), k = 1, \ldots, M$, also have full rank.

AE4: The projectors into intersections of the null spaces of the equation blocks are uniformly bounded in norm.

The following result holds.

Theorem 2.1 Under assumptions AE1—AE4, given any $\epsilon_{tol} > 0$, the algorithm will terminate because

$$\|F_k(x_i)\| \leq \epsilon_{tol} \tag{2.4}$$

will be simultaneously satisfied for some i and all $k = 1, \ldots, M$.

The proof of this theorem is a corollary of the global convergence theorem for equality constrained optimization in Chapter 4. It will be discussed there. In Chapter 3, we will discuss in more detail the termination criterion that requires the norms of all component blocks to be less than or equal to a certain tolerance parameter.

Chapter 3

Multilevel Algorithm for Equality Constrained Optimization Based on Brent's Method

We study the optimization algorithms in two stages. First, we propose a modification of an existing algorithm, and then we use this modification to propose a general multilevel algorithm.

3.1 Modification of an Existing Algorithm

In Dennis, El-Alem, Maciel [16] and Maciel [20], the authors present a global convergence theory for a class of algorithms for solving problem EQC. These algorithms use the trust region approach as a globalization strategy.

The general algorithm suggested in [16] and [20] computes the trial step as a sum of two components, normal and tangential†. The normal component, s^n, satisfies a Fraction of Cauchy Decrease (FCD) condition on the quadratic model of the constraints, $\frac{1}{2}\|C_c + \nabla C_c^T s\|^2$. The tangential component, s^t, satisfies an FCD condition on the quadratic model of the Lagrangian, $q_c(x_c + s^n + s)$, restricted to the null space of ∇C_c^T. The Lagrange multiplier and Hessian estimates are only assumed to be bounded. To evaluate the trial step, $s_c = s_c^n + s_c^t$, the algorithm uses the augmented Lagrangian. The penalty parameter is updated via the method proposed by El-Alem [8].

Some of the interesting and surprising results are that the convergence theory goes through even when the Lagrange multipliers are taken to be zero and the model Hessians are arbitrary but bounded.

As an initial step toward our goal of a multilevel algorithm, we propose a modification of the general algorithm in [16]. In essence, we replace q_c by a quadratic model of the objective function, but we construct the model at $x_c + s^n$, not at x_c.

†The step may be computed as a single step, but it is crucial for the theory that the step can be represented as a sum of two components that satisfy the appropriate conditions.

Our motivation is the hope that the model at $x_c + s^n$ will lead to a better s^t because it is more local. The modified algorithm follows.

Algorithm 3.1 EQC Algorithm for a Single Block of Constraints

Let $\mathcal{P}(x;\rho) = f(x) + \frac{\rho}{2}\|C(x)\|^2$.
Let $\phi_{\frac{1}{2}}(s) = f(x_{\frac{1}{2}} + \nabla f(x_{\frac{1}{2}})^T s + \frac{1}{2}s^T H_{\frac{1}{2}} s$.

Step 0. Initialize:
Given x_0, compute Z_0, a basis for $\mathcal{N}(\nabla C(x_0)^T)$.
Choose $\delta_{min}, \delta_{max}, \epsilon_{tol} > 0$.
Set $\rho_0 = 1$ and $\beta > 0$.

Step 1. Test for convergence:
if $\|Z_c^T \nabla f_c\| + \|C_c\| \leq \epsilon_{tol}$ **then** exit
end if

Step 2. Compute a trial step:
if $C_c = 0$ **then**

1. Compute a step, s_c^t, that satisfies an FCD condition on the quadratic model, $\phi_c(s)$, of the objective function about x_c.
2. Set $s_c = s_c^t$.

else

1. Compute a step, s_c^n, that satisfies an FCD condition on the quadratic model of the constraints about x_c.
2. Set $x_{\frac{1}{2}} = x_c + s_c^n$.
3. Compute a step, s_c^t, that satisfies an FCD condition on the quadratic model, $\phi_{\frac{1}{2}}(s)$, of the objective function about $x_{\frac{1}{2}}$ and also satisfies the condition $\nabla C_c^T s = 0$. **(This is the major modification.)**
4. Set $s_c = s_c^n + s_c^t$.

end if

Step 3. Update the penalty parameter:

Compute $pred_c(s_c; \rho_-)$.

if $pred_c(s_c; \rho_-) \geq \frac{\rho_-}{2}[\|C_c\|^2 - \|C_c + \nabla C_c^T s_c\|^2]$ **then**

$\rho_c = \rho_-$.

else

$\rho_c = \bar{\rho}_c + \beta$, where

$\bar{\rho}_c = \frac{2[\phi_{\frac{1}{2}}(0) - \phi_{\frac{1}{2}}(s_c^t)]}{\|C_c\|^2 - \|C_c + \nabla C_c^T s_c\|^2}$.

end if

Step 4. Evaluate the step:

Compute $ared_c(s_c; \rho_c) = \mathcal{P}(s_c; \rho_c) - \mathcal{P}(s_+; \rho_c)$.

Compute $pred_c(s_c; \rho_c) = \mathcal{P}(x_c; \rho_c) - \phi_{\frac{1}{2}}(s_c^t) - \frac{\rho_c}{2}\|\|C_c + \nabla C_c^T s_c\|^2$.

Evaluate the step and update the trust region radius using Algorithm 2.4.

if the step is accepted **then**

Set $x_+ = x_c + s_c$.

go to Step 1.

else

go to Step 2.

end if

end

The differences between Algorithm 3.1 and the algorithm proposed in [16] and [20] are as follows:

1. Algorithm 3.1 uses the ℓ_2 penalty function as a merit function instead of the augmented Lagrangian. The definitions of the actual and predicted reductions and the penalty parameter updating scheme are changed accordingly.

2. The tangential component of the step is evaluated by minimizing the quadratic model of the objective function, restricted to the null space of ∇C_c^T, about $x_{\frac{1}{2}}$, instead of minimizing the quadratic model of the Lagrangian, similarly restricted, about x_c, i.e., the quadratic model of the objective function is built from the information at $x_{\frac{1}{2}}$, instead of x_c.

The global convergence theorem for Algorithm 3.1 is a corollary of the theorem proved for the multilevel algorithm in Chapter 4.

In this context, it is clear why the Lagrange multipliers can be taken to be zero. We are solving two minimization subproblems, each with its own starting point. The second subproblem has only the null space constraint and thus is really an unconstrained problem.

This intermediate result also provides an insight into the role played by the penalty parameter for a two-step method. It is important to observe that in the global convergence proofs of both Algorithm 3.1 and the algorithm proposed in [16] and [20], the penalty parameter has a function in addition to the usual one of penalizing the constraint violations: it penalizes the possible increase in the Lagrangian or in the objective function caused by the substep s^n.

The same reasoning as in the case of nonlinear equations applies here. At the first stage, the minimization is performed only on the quadratic model of the constraints, without any reference to the model of the Lagrangian or the objective function. It is possible that the normal component, s_c^n, produced independently, decreases the predicted value of the constraint model but increases the value of the Lagrangian or the function. In fact, s_c^n does not predict anything about the behavior of the Lagrangian model in the same sense as it predicts the behavior of the constraint model. The only "prediction" that the theory provides is a negative lower bound on the effect of s_c^n on the function. In such a case, the penalty parameter serves to ensure that total reduction predicted in the model of the merit function by the total step is positive. It does this by putting an appropriately heavy weight on the decrease predicted for the already processed components.

3.2 Extension of Brent's Method to Equality Constrained Optimization: a Multilevel Algorithm

The inner loop of the multilevel algorithm for problem EQC based on the method of Brent can be described as the following extension of the algorithm for nonlinear equations. The constraint system of the problem is divided into M arbitrary blocks. In practice, this block decomposition is obvious in most cases. The current approximation to a solution of problem EQC is y_0. A quadratic Gauss-Newton model of the first block of constraints is built about the initial point, y_0, and a step, s_1, bounded by the trust region, is found in such a way that it satisfies a Fraction of Cauchy Decrease for this model. The step is taken to yield the point $y_1 = y_0 + s_1$. The process continues along the same lines as Algorithm 2.2 for solving nonlinear systems. When all the constraint blocks have been processed, $n - m$ degrees of freedom still remain. The remaining variables are used in building a model of the objective function, so that the final substep, s_{M+1}, is obtained to produce sufficient predicted decrease in the quadratic model at y_M of the objective function, restricted to the intersection of the null spaces of the Jacobians of all constraint blocks. The final step is taken to yield the next major iterate, i.e., the next approximation to a solution of problem EQC. Unless the convergence criterion is met, the algorithm returns to process again the first block of constraints at the new point.

3.2.1 The Merit Function and Reductions

The new merit function is an extension of the merit function used for nonlinear systems:

$$\begin{aligned}
\bar{\mathcal{P}}(x;\rho_1,\ldots,\rho_M) &= f(x) + \rho_M(\|C_M(x)\|^2 \\
&\quad \rho_{M-1}(\|C_{M-1}(x)\|^2 + \rho_{M-2}(\|C_{M-2}(x)\|^2 + \ldots + \rho_2(\|C_2(x)\|^2 + \rho_1\|C_1(x)\|^2)))) \\
&= f(x) + \sum_{k=1}^{M}(\prod_{j=k}^{M}\rho_j)\|C_k(x)\|^2,
\end{aligned}$$

where $\rho_k \geq 1$, $k = 1,\ldots,M$.

We define the actual reduction as

$$\begin{aligned}
ared_c(s_1,\ldots,s_{M+1};\rho_1,\ldots,\rho_M) &= \bar{\mathcal{P}}(x_c) - \bar{\mathcal{P}}(x_+) \\
&= \bar{\mathcal{P}}(y_0) - \bar{\mathcal{P}}(y_M).
\end{aligned} \tag{3.1}$$

Let

$$\phi_M(s) \equiv f(y_M) + \nabla f(y_M)^T s + \frac{1}{2}s^T H_M s, \tag{3.2}$$

where H_M is either the Hessian of f at y_M or an approximation to it. The only assumption imposed on H_M is that it must be uniformly bounded in norm from above.

The predicted reduction models the actual reduction and is defined as

$$pred_c(s_1,\ldots,s_{M+1};\rho_1,\ldots,\rho_M) = \tag{3.3}$$

$$f(y_0) + \sum_{k=1}^{M}(\prod_{j=k}^{M}\rho_j)\|C_k(y_0)\|^2 \tag{3.4}$$

$$-[\phi_M(s_{M+1}) + \sum_{k=1}^{M}(\prod_{j=k}^{M}\rho_j)\|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2]$$
$$= f(y_0) - [f(y_M) + \nabla f(y_M)^T s_{M+1} + \frac{1}{2}s_{M+1}^T H_M s_{M+1}]$$
$$+ \sum_{k=1}^{M}(\prod_{j=k}^{M}\rho_j)[\|C_k(y_0)\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2].$$

Notice again that the model information is computed at $x_c = y_0$ only for the first block of constraints.

3.2.2 The Statement of the Algorithm

The formal description of the algorithm follows.

Let the constraints be enumerated in the following way. The constraints in the first block are numbered from $n_1 = 1$ to $n_2 - 1$; the constraints in the second block are numbered from n_2 to $n_3 - 1$, and so on, until the constraints in the last block are numbered from n_M to $n_{M+1} = m$.

Algorithm 3.2 Multilevel Algorithm for Equality Constrained Optimization

Given $\delta_k > 0, k = 1, \ldots, M, \delta_{max} > 0, \delta_{min} > 0, 0 < \eta_1 < \eta_2 < 1, \alpha_1 \in (0, 1]$, $\alpha_2 > 1, x_c \in \Re^n$.

Outer Loop: Do until convergence:

$y_0 = x_c$.

Compute the trial step.

Inner Loop: Do $k = 1, M$

Compute the unconstrained Brent step s_k^{Brent} from y_{k-1}.

If $\|s_k^{Brent}\| \leq \delta_k$ **then**

$y_k = y_{k-1} + s_k^{Brent}$.

$s_k = s_k^{Brent}$.

else

Compute s_k that satisfies the Fraction of Cauchy Decrease condition on $\frac{1}{2}\|F_k(y_{k-1}) + J_k(y_{k-1})s\|_2^2$ restricted to the intersection of the null spaces of $J_j(y_{j-1})s = 0, j = 1, \ldots, k-1$, and $\|s\|_2 \leq \delta_k$. ($s_k = \frac{\delta_k s_k^{Brent}}{\|s_k^{Brent}\|}$ is allowed, by Proposition 2.2).

$y_k = y_{k-1} + s_k$.

End if

End Inner Loop

Compute s_{M+1} to satisfy the Fraction of Cauchy Decrease condition on the subproblem: minimize $\phi_M(s_{M+1})$ restricted to the intersection of the null spaces of $J_j(y_{j-1})s = 0, j = 1, \ldots, M$, and $\|s\|_2 \leq \delta_M$.

$y_{M+1} = y_M + s_{M+1}$.

$x_+ = y_{M+1}$.

The trial step is: $\hat{s}_c = s_1 + \ldots + s_{M+1}$.

Update the penalty parameters using Algorithm 3.3.

Evaluate the step and update the trust region radius using Algorithm 2.4.

If the step is accepted, set $x_c = x_+$.

End Outer Loop

We should note that an option is to eliminate only a subset of constraints via the described procedure. In this case, the rest of the constraints and the objective function would be restricted to the intersection of the null spaces of the Jacobians of the processed constraints, and the resulting reduced optimization problem would be solved by a chosen method. The discussion of this approach is left for later work.

3.2.3 Updating the Penalty Parameters

To update the penalty function parameters, we use the following procedure.

Algorithm 3.3 Penalty Parameter Updating Algorithm (Done on completion of each inner sweep of minimization problems.)

At the beginning of Algorithm 3.2, set $\rho_1^- = \ldots = \rho_M^- = 1$ and choose $\beta \in (0,1)$.

1. Compute $cpred_1(s_1) = \|C_1(y_0)\|^2 - \|C_1(y_0) + \nabla C_1(y_0)^T s_1\|^2$.
2. **Do** $k = 1, M-1$

   Update ρ_k.

   Compute

   $$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) = [\|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 + \rho_k^- cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c).$$

   if $cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) \geq$

$\frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c)$ **then**

$\rho_k^c = \rho_k^-$.

$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^c) =$

$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-)$.

else

$\rho_k^c = \bar{\rho}_k + \beta$,

where $\bar{\rho}_k = \frac{2[\|C_{k+1}(y_k)+\nabla C_{k+1}(y_k)^T s_{k+1}\|^2 - \|C_{k+1}(y_0)\|^2]}{cpred_k(s_1,\ldots,s_k;\rho_1^c,\ldots,\rho_{k-1}^c)}$.

Compute $cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^c)$.

end if

end Do

3. Update ρ_M.

Compute

$pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) =$

$[f(y_0) - \phi_M(s_{M+1})] + \rho_M^- cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)$.

if $pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) \geq$

$\frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)$ **then**

$\rho_M^c = \rho_M^-$.

$pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^c) = pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-)$.

else

$\rho_M^c = \rho_M^- + \beta$,

where $\rho_M^- = \frac{2[\phi_M(s_{M+1}) - f(y_0)]}{cpred_M(s_1,\ldots,s_M;\rho_1^c,\ldots,\rho_{M-1}^c)}$.

Compute $pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^c)$.

end if

End

The reasoning behind this update is simple: we want the predicted reduction to be positive. We can achieve this, for example, if for each substep s_k, the predicted reduction accumulated by the step $s_1 + \ldots + s_k$ is at least a fraction of the predicted decrease accumulated by the step $s_1 + \ldots + s_{k-1}$; i.e.,

$$cpred_{k+1} = \|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 + \rho_k \, cpred_k \geq \frac{\rho_k}{2} cpred_k \tag{3.5}$$

or

$$\|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 \geq -\frac{\rho_k}{2} cpred_k.$$

Therefore, ρ_k must satisfy

$$\rho_k \geq \frac{2[\|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 - \|C_{k+1}(y_0)\|^2]}{cpred_k}. \tag{3.6}$$

So, if (3.4) is not satisfied, we set ρ_k to the right-hand side of (3.6) plus a small number $\beta \in (0,1)$. To check, after ρ_k is updated we have

$$\begin{aligned} cpred_{k+1} - \frac{\rho_k}{2} cpred_k &= \|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 \\ &+ \left[\frac{2[\|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 - \|C_{k+1}(y_0)\|^2]}{cpred_k} + \beta\right] cpred_k \\ &- \left[\frac{[\|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 - \|C_{k+1}(y_0)\|^2]}{cpred_k} + \frac{\beta}{2}\right] cpred_k \\ &= \frac{\beta}{2} cpred_k \\ &\geq 0. \end{aligned}$$

Thus

$$cpred_{k+1} \geq \frac{\rho_k^c}{2} cpred_k$$

will always hold. This fact gives us the following lemma.

Lemma 3.1 Let the penalty parameters be updated according to Algorithm 3.3. Then the partial and the total predicted reductions satisfy

$$\begin{aligned} cpred_{k+1} &\geq \frac{\rho_k}{2} cpred_k & (3.7) \\ &\geq \frac{\prod_{j=1}^{k} \rho_j}{2^k} cpred_1 & (3.8) \end{aligned}$$

and

$$\begin{aligned} pred_c &\geq \frac{\rho_M}{2} cpred_M & (3.9) \\ &\geq \frac{\prod_{j=1}^{M} \rho_j}{2^M} cpred_1. & (3.10) \end{aligned}$$

This very useful lemma is used in the proof of convergence.

3.2.4 An Alternative Penalty Parameter Scheme

As mentioned earlier, the only step that is naturally assured to predict a positive decrease from the value at x_c of the norm of its block of constraints is s_1, which is why it is the most heavily weighted substep. The accumulated predicted reduction has to be made positive with the help of the penalty parameters.

An alternative scheme is to place the penalty parameter only on the first block of constraints, i.e., to have a merit function of the form

$$\tilde{\mathcal{P}}(x) = f(x) + \sum_{j=2}^{M} \|C_j(x)\|^2 + \rho\|C_1(x)\|^2$$

with an appropriate scheme for updating ρ.

While theoretically this scheme would not be fundamentally different from the one we adopted, in practice it is expected to have more severe problems with conditioning as ρ grows large.

3.2.5 The Stopping Criteria

In the beginning of this study, the termination criterion similar to the one in [16], [20] was used; namely

$$\|P_M^T \nabla f(y_M)\| + \sum_{k=1}^{M} \|C_k(y_{k-1})\| \le \varepsilon_{tol}. \tag{3.11}$$

It proved inadequate for the purposes of the convergence theory.

While we still use the first order necessary conditions for problem EQC to terminate the algorithm, we now require that

$$\begin{aligned} \|C_1(y_0)\| &\le \epsilon_{tol} \\ \|C_2(y_1)\| &\le \epsilon_{tol} \\ &\cdots \\ \|C_M(y_{M-1})\| &\le \epsilon_{tol} \\ \|P_M^T \nabla f(y_M)\| &\le \epsilon_{tol} \end{aligned} \tag{3.12}$$

hold simultaneously.

Lemma 4.2 in Chapter 4 will show that

$$\|s_k\| = O(\|C_k(y_{k-1})\|).$$

Thus, if $\|C_k(y_{k-1})\|$ is small, $\|s_k\|$ will be small and the inner loop iterates y_k will be close to each other, and in the limit we will show that at least a subsequence of the generated sequence of the outer loop iterates will converge to a stationary point of problem EQC.

The tolerance parameters ϵ_{tol} need not be the same, but for convenience, they are taken to be the same throughout this thesis.

The reason for requiring a stronger stopping criterion is that (3.11) does not differentiate between the individual $\|C_k(y_{k-1})\|$. It is essential for the convergence proof to determine how close to feasibility an iterate must be in order for the penalty parameters not to be increased. This is a measure of feasibility versus optimality. The stopping criterion (3.11) allows only the total feasibility to be measured and thus to determine when ρ_M does not have to be increased. But even if ρ_M is not increased, $\rho_1, \ldots, \rho_{M-1}$ may have to be increased because of the relative sizes of the component block norms. Criterion (3.11) does not allow us to measure **relative feasibility** of one block of constraints with respect to the others.

Several stopping criteria provide this capability, for example, we may require that

$$
\begin{aligned}
\|C_1(y_0)\| + \|C_2(y_1)\| &\leq \epsilon_{tol}^1 \\
\|C_1(y_0)\| + \|C_2(y_1)\| + \|C_3(y_2)\| &\leq \epsilon_{tol}^2 \\
&\ldots \\
\|C_1(y_0)\| + \ldots + \|C_M(y_{M-1})\| &\leq \epsilon_{tol}^{M-1}, \\
\|P_M^T \nabla f(y_M)\| + \|C_1(y_0)\| + \ldots + \|C_M(y_{M-1})\| &\leq \epsilon_{tol}^M,
\end{aligned}
$$

where $\epsilon_{tol}^1 < \epsilon_{tol}^2 < \ldots < \epsilon_{tol}^{M-1} < \epsilon_{tol}^M$, hold simultaneously. However, this is an awkward and artificial criterion, while (3.12) is simple and reasonable.

Chapter 4

Global Convergence Theory

In this chapter we present the global convergence theory for Algorithm 3.2. The general reasoning follows the lines of that in Dennis, El-Alem, Maciel [16] and Maciel [20]. Of course, many modifications are necessary in order to accommodate the multilevel nature of the subproblem solving procedure. However, two crucial properties of the algorithm allow us to retain the basic flow of the argument: the substeps generated by the subproblems are orthogonal to each other and the merit function together with the penalty parameter updating strategy ensure that the overall model behavior has satisfactory reduction properties.

Algorithm 3.1 is an intermediate step between the algorithm of [16] and [20] and our multilevel algorithm. We have pointed out that it differs from the algorithm of [16] and [20] in a crucial way, namely, the model of the constraints and the model of the objective function (or the Lagrangian) are built using information at different points. However, given the convergence theory of [16] and [20], only moderate effort is required to prove convergence for Algorithm 3.1 alone. The situation changes drastically when the constraint system is partitioned into an arbitrary number of blocks, each having its own penalty parameter. Whereas in Algorithm 3.1 and the class of algorithms in [16] and [20] we have to balance total feasibility with optimality, in a multilevel algorithm we must balance optimality with total feasibility and with relative feasibility among the individual constraint blocks, to account for the behavior of individual penalty parameters.

4.1 Basic Ingredients of a Global Convergence Proof

Before beginning the proofs, let us give an overview of the necessary steps that usually comprise a global convergence proof and point out the consequences of our algorithm's distinguishing characteristics for the proof.

The first three ingredients are identical to those required for a typical analysis of an unconstrained minimization algorithm.

1. The trial step must be shown to satisfy a sufficient predicted decrease condition, usually the FCD condition, frequently expressed in the practically useful form of Lemma 1.4. Our algorithm assumes that the substeps satisfy the FCD condition on the subproblems. It remains for us to show that the total step from x_c to x_+ satisfies a suitable decrease condition.

2. The difference between the actual and predicted reduction must be bounded above by at least a constant multiple of the square of the total step norm. This is easily shown for Algorithm 3.2. Some terms of the actual estimate for Algorithm 3.2 involve products of penalty parameters. These terms involve higher powers of the total norm step. This technical detail is necessary for the global convergence proof.

3. The algorithm must be shown to be well-defined, i.e., we must prove that the ratio of the actual reduction to predicted reduction can be made greater than a given $\eta_1 \in (0, 1)$ after a finite number of trial step computations. Given 2, it is easy to show that as the trust region radius approaches zero, the ratio of the actual reduction to predicted reduction approaches one. For the algorithm to be well-defined we must show that the ratio of the predicted to actual reduction approaches one faster than the trust region radius goes to zero. This is easily established for our algorithm.

   An algorithm for constrained optimization that uses penalty parameters in its merit function requires the fourth ingredient.

4. The penalty parameter in the merit function must be shown to be bounded. The technique is to prove that the product of the penalty parameter and the trust region radius is bounded by a constant independent of the iterates. The sequence of the trust region radii is then shown to be bounded away from zero. Here a crucial role is played by the trust region updating technique introduced in [20]: after a successful iteration and before starting the next iteration, the trust region radius is set to be no smaller than a pre-defined value. This way of updating allows us to prove that the sequence of penalty parameters is bounded from above.

The method for updating the penalty parameter ensures that the sequence of penalty parameters is nondecreasing [‡], which, together with its boundedness, allows us to conclude that the penalty parameter sequence converges and, moreover, remains constant after a finite number of increases. This fact is used in the global convergence theorem.

Because of the multilevel nature of our algorithm, proving boundedness of the penalty parameters requires extensive modifications of the theory in [16] and [20].

4.2 Assumptions

Let $\{x_i\}$ denote the sequence of outer iterates generated by any of the proposed algorithms, and let $\{s_i\}$ be the corresponding sequence of steps. Let $\{y_k^i\}$ be the sequence of iterates generated in the inner loops of the algorithms, and let $\{s_k^i\}$ be the corresponding substeps.

The following assumptions are made for the optimization problems:

AO1: There exists a convex set. $\Omega \in \Re^n$, such that x_i, $x_i + s_i$, y_k^i, and $y_k^i + s_k^i$ are in Ω for all i and all k.

AO2: $f, C \in C^2(\Omega)$.

AO3: $\nabla C(x)$ has rank m for all $x \in \Omega$. This implies that each $\nabla C_k(x), k = 1, \ldots, M$, has full rank for all $x \in \Omega$. This is a strong assumption, but it is a standard practice to require it for the sake of convergence proofs. Practical experience suggests that the breakdown of this assumption does not necessarily diminish the efficacy of our algorithm.

AO4: The projectors $P_k, k = 1, \ldots, M$, and P are continuous functions. This assumption is necessary for proving global convergence. Methods for constructing continuous orthonormal null space bases are described in [35], [31], [30] and will be discussed in more detail in Chapter 5. In addition, these matrices are assumed to be bounded away from zero in norm.

AO5: $f(x)$, $\nabla f(x)$, $\nabla^2 f(x)$, H_M, $C(x)$, $\nabla C(x)$, $\nabla C_k(x)$, $\nabla^2 C_j(x), j = 1, \ldots, m$, $\{[P_{k-1}^T \nabla C_k(x)]^T [P_{k-1}^T \nabla C_k(x)]\}^{-1}, k = 1, \ldots, M$, are all uniformly bounded in

‡The global convergence theory for algorithms with nonmonotone penalty parameters has been investigated by Mahmoud El-Alem [10].

norm for all $x \in \Omega$, i.e., there are positive constants $\sigma_1, \ldots, \sigma_{10}$, such that
$\|f(x)\| \leq \sigma_1$,
$\|\nabla f(x)\| \leq \sigma_2$,
$\|\nabla^2 f(x)\| \leq \sigma_3$,
$\|H_M\| \leq \sigma_4$ for all inner loops,
$\|C(x)\| \leq \sigma_5$,
$\|\nabla C(x)\| \leq \sigma_6$,
$\|\nabla^2 C_j(x)\| \leq \sigma_7, j = 1, \ldots, m$,
$\|\{[P_{k-1}^T \nabla C_k(x)]^T [P_{k-1}^T \nabla C_k(x)]\}^{-1}\| \leq \sigma_8, k = 1, \ldots, M$, $(P_0 \equiv I)$,
$\|P_k\| \leq \sigma_9$,
$\|P\| \leq \sigma_{10}$.

The bounds σ_5 and σ_6 on the constraint system and its gradient, respectively, imply the same bounds on the component blocks and component gradients of the system.

Note that if instead of generic P_k and P, we use orthonormal Q_k and Q to project into intersections of constraint block null spaces, the boundedness assumption is automatically satisfied. Reduced basis projectors Z_k, Z, which we will discuss in Chapter 5, also satisfy the boundedness assumption by virtue of the assumptions on the gradients and hessians of the constraint blocks.

4.3 Technical Lemmas

The following simple lemma is important to the proof of global convergence.

Lemma 4.1 Let $\hat{s}_c = s_1 + \ldots + s_{M+1}$ be a trial step generated by Algorithm 3.2. Then for $N \leq M + 1$ the following inequality holds:

$$\|\sum_{k=j}^{N} s_k\| \leq \|\hat{s}_c\|, \quad 1 \leq j \leq N.$$

Proof:

By construction, the substeps $s_1, \ldots, s_{M+1}$ are orthogonal to each other. Therefore, by the Pythagorean theorem we have

$$\begin{aligned} \|s_j + \ldots + s_N\|^2 &= \|s_j\|^2 + \ldots + \|s_N\|^2 \\ &\leq \|s_1\|^2 + \ldots + \|s_{M+1}\|^2 \end{aligned}$$

$$
\begin{aligned}
&= \|s_1 + \ldots + s_{M+1}\|^2 \\
&= \|\hat{s}_c\|^2,
\end{aligned}
$$

and the result is proved. □

The following lemma bounds the substeps in terms of the norms of the constraint blocks. It will allow us to conclude that in the limiting case our termination criterion (3.12) will cause convergence to a stationary point of problem EQC.

Lemma 4.2 Let s_k, $1 \leq k \leq M$, be a substep generated by solving the k-th subproblem of an inner loop. Then, under assumptions AO1—AO5, there exists a positive constant $\mathcal{K}_1$, independent of k and the outer loop index i, such that

$$\|s_k\| \leq \mathcal{K}_1 \|C_k(y_{k-1})\|. \tag{4.1}$$

Proof:

If $C_k(y_{k-1}) = 0$, then $s_k = 0$ and (4.1) holds. If $C_k(y_{k-1}) \neq 0$, then we obtain s_k by approximately solving for v_k (i.e., requiring that v_k satisfy the FCD condition) the problem

$$
\begin{aligned}
&\text{minimize } \|C_k(y_{k-1}) + (P_{k-1}^T \nabla C_k(y_{k-1}))^T v\|^2 \\
&\text{subject to } \|P_{k-1} v\|^2 \leq \delta_k^2,
\end{aligned}
$$

where P_{k-1} is a projector into $\bigcap_{j=1}^{k-1} [\mathcal{N}(\nabla C_j(y_{j-1})^T)]$, and then setting $s_k = P_{k-1} v_k$.

Our strategy of taking the unconstrained Brent step whenever it is within the individual trust region ensures that

$$
\begin{aligned}
\|s_k\| &\leq \|s_k^{Brent}\| \\
&= \|P_{k-1}[\nabla C_k(y_{k-1})^T P_{k-1}]^\dagger C_k(y_{k-1})\| \\
&= \|P_{k-1} P_{k-1}^T \nabla C_k(y_{k-1})[(P_{k-1}^T \nabla C_k(y_{k-1}))^T P_{k-1}^T \nabla C_k(y_{k-1})]^{-1} C_k(y_{k-1})\| \\
&\leq \sigma_9^2 \sigma_8 \sigma_6 \|C_k(y_{k-1})\| \\
&\equiv \mathcal{K}_1 \|C_k(y_{k-1})\|.
\end{aligned}
$$

This concludes the proof. □

The following lemma offers a way of expressing the sufficient predicted decrease condition imposed on the solutions of the inner loop subproblems.

Lemma 4.3 Let $s_1, \dots, s_{M+1}$, be the trial substeps satisfying the Fraction of Cauchy Decrease condition imposed on the inner loop subproblems $1, \dots, M+1$ of the outer loop i. Then, under assumptions AO1—AO5, there exist positive constants $\mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4$, independent of k and the outer loop index i, such that

$$\begin{aligned} &\|C_k(y_{k-1})\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2 \geq \\ &\frac{1}{2}\mathcal{K}_2\|C_k(y_{k-1})\| \min\{\mathcal{K}_3\|C_k(y_{k-1})\|, \delta_k\}, \end{aligned} \tag{4.2}$$

where $k = 1, \dots, M$,

and

$$\begin{aligned} &\phi_M(0) - \phi_M(s_{M+1}) \equiv \\ &f(y_M) - [f(y_M) + \nabla f(y_M)^T s_{M+1} + \frac{1}{2}s_{M+1}^T H_M s_{M+1}] \geq \\ &\frac{1}{2}\|P_M^T \nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T \nabla f(y_M)\|, \delta_{M+1}\}. \end{aligned} \tag{4.3}$$

where P_M is a projector into $\bigcap_{j=1}^{M}[\mathcal{N}(\nabla C_j(y_{j-1})^T)]$.

Proof:

For $1 \leq k \leq M$, letting $P_0 = I$, we find a step v_k that satisfies a Fraction of Cauchy Decrease condition on the problem

$$\begin{aligned} &\text{minimize } \|C_k(y_{k-1}) + (P_{k-1}^T \nabla C_k(y_{k-1}))^T v\|^2 \\ &\text{subject to } \|P_{k-1}v\|^2 \leq \delta_k^2, \end{aligned}$$

and then set the substep $s_k = P_{k-1}v_k$. Therefore, by Lemma 1.4 we have

$$\begin{aligned} &\|C_k(y_{k-1})\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2 \geq \\ &\quad \frac{1}{2}\|P_{k-1}^T \nabla C_k(y_{k-1}) C_k(y_{k-1})\| \min\{\frac{\|P_{k-1}^T \nabla C_k(y_{k-1}) C_k(y_{k-1})\|}{\|P_{k-1}^T \nabla C_k(y_{k-1}) \nabla C_k(y_{k-1})^T P_{k-1}\|}, \delta_k\}. \end{aligned}$$

Since

$$\|P_{k-1}^T \nabla C_k(y_{k-1}) C_k(y_{k-1})\| \geq \frac{1}{\mathcal{K}_1}\|C_k(y_{k-1})\|$$

and

$$\begin{aligned} \|P_{k-1}^T \nabla C_k(y_{k-1}) \nabla C_k(y_{k-1})^T P_{k-1}\| &\leq \|P_{k-1}\|^2 \|\nabla C_k(y_{k-1})\|^2 \\ &\leq \sigma_9^2 \sigma_6^2, \end{aligned}$$

we obtain (4.2) by setting

$$\mathcal{K}_2 = \frac{1}{\mathcal{K}_1} \quad \text{and} \quad \mathcal{K}_3 = \frac{1}{\mathcal{K}_1}\frac{1}{\sigma_6^2\sigma_9^2}.$$

Letting $\mathcal{K}_4 = \frac{1}{\sigma_4}$ yields (4.3) as a direct application of Lemma 1.4 to subproblem $M+1$.

This concludes the proof. □

The next lemma accounts for the effect of each substep, $s_k, 1 \leq k \leq M$, on the constraint blocks numbered $k+1,\ldots,M$ and on the objective function. As mentioned previously, this is a crucial estimate since the computation of s_k guarantees a reduction in the Gauss-Newton model of $\|C_k\|$ from y_{k-1} and predicts no increase for the models of $\|C_j\|, j = 1,\ldots,k-1$, but does not take into account any information about $\|C_j\|, j = k+1,\ldots,M$ or the objective function.

Lemma 4.4 Under assumptions AO1—AO5, there exist positive constants $\mu_1,\ldots,\mu_M$, independent of the iterates, such that

$$\|C_k(y_0)\|^2 - \|C_k(y_{k-1})\|^2 \geq -\mu_{k-1}\sum_{j=1}^{k-1}\|C_j(y_{j-1})\|, \quad k = 2,\ldots,M, \tag{4.4}$$

and

$$f(y_0) - f(y_M) \geq -\mu_M \sum_{j=1}^{M}\|C_j(y_{j-1})\|. \tag{4.5}$$

Proof:

By Lemma 1.8, for some $z_k \in (y_0, y_{k-1})$ we have

$$\begin{aligned}
&\|C_k(y_0)\|^2 - \|C_k(y_{k-1})\|^2 \equiv \|C_k(y_0)\|^2 - \|C_k(y_0 + s_1 + \ldots + s_{k-1})\|^2 \\
&\quad = \|C_k(y_0)\|^2 - \|C_k(y_0)\|^2 - 2[\nabla C_k(y_0)C_k(y_0)]^T(s_1 + \ldots + s_{k-1}) \\
&\quad -\frac{1}{2}(s_1 + \ldots + s_{k-1})^T[\nabla C_k(z_k)\nabla C_k(z_k)^T \\
&\quad + \sum_{j=n_k}^{n_{k+1}-1} C_j(z_k)\nabla^2 C_j(z_k)](s_1 + \ldots + s_{k-1}) \\
&\quad \geq -\{2\|\nabla C_k(y_0)\|\|C_k(y_0)\| + [\frac{1}{2}\|\nabla C_k(z)\|^2 + \sum_{j=n_k}^{n_{k+1}-1}\|C_j(z_k)\|\|\nabla^2 C_j(z_k)\|] \\
&\quad \times\|s_1 + \ldots + s_{k-1}\|\}\|s_1 + \ldots + s_{k-1}\| \\
&\quad \geq -\{2\sigma_6\sigma_5 + [\frac{1}{2}\sigma_6^2 + \sum_{j=n_k}^{n_{k+1}-1}\sigma_5\sigma_6^2](k-1)\delta_{max}\}\,\|s_1 + \ldots + s_{k-1}\|
\end{aligned}$$

$$\geq -\{2\sigma_6\sigma_5 + [\frac{1}{2}\sigma_6^2 + \sum_{j=n_k}^{n_{k+1}-1} \sigma_5\sigma_r^2](k-1)\delta_{max}\}\,\mathcal{K}_1 \sum_{j=1}^{k-1} \|C_j(y_{j-1})\|$$

$$\equiv -\mu_{k-1} \sum_{j=1}^{k-1} \|C_j(y_{j-1})\|.$$

The last inequality follows from Lemma 4.2, and so (4.4) is established.

Similarly, for some $z_M \in (y_0, y_M)$,

$$\begin{aligned}
f(y_0) - f(y_M) &\equiv f(y_0) - f(y_0 + s_1 + \ldots + s_M) \\
&= f(y_0) - f(y_0) - \nabla f(y_0)^T(s_1 + \ldots + s_M) \\
&- \frac{1}{2}(s_1 + \ldots + s_M)^T \nabla^2 f(z_M)(s_1 + \ldots + s_M) \\
&\geq -[\|\nabla f(y_0)\| + \frac{1}{2}\|\nabla^2 f(z_M)\|\|s_1 + \ldots + s_M\|]\|s_1 + \ldots + s_M\| \\
&\geq -[\sigma_2 + \frac{1}{2}\sigma_3 M\delta_{max}]\mathcal{K}_1 \sum_{j=1}^{M} \|C_j(y_{j-1})\| \\
&\geq -[\sigma_2 + \frac{1}{2}\sigma_3 M\delta_{max} + \mu_{M-1}]\mathcal{K}_1 \sum_{j=1}^{M} \|C_j(y_{j-1})\| \\
&\equiv -\mu_M \sum_{j=1}^{M} \|C_j(y_{j-1})\|,
\end{aligned}$$

which concludes the proof. $\square$

The last inequality of the proof was used to ensure that by construction,

$$\mu_1 < \mu_2 < \ldots < \mu_M.$$

This fact will be needed in a later proof.

4.4 The Behavior of the Model

The following lemma provides a workable expression of the Fraction of Cauchy Decrease conditions similar to the one in Lemma 1.4.

Lemma 4.5 Let $s_1, \ldots, s_{M+1}$ be the substeps generated at the current iterate $x_c = y_0$. Then under assumptions AO1—AO5, the partial and the total predicted reductions satisfy the following estimates:

$$cpred_k(s_1, \ldots, s_k; \rho_1, \ldots, \rho_{k-1}) \geq \tag{4.6}$$

$$\frac{1}{2}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\}$$
$$-\mu_{k-1}\sum_{j=1}^{k-1}\|C_j(y_{j-1})\|$$
$$+\rho_{k-1}cpred_{k-1}(s_1,\ldots,s_{k-1};\rho_1,\ldots,\rho_{k-2})$$

and

$$\begin{aligned}
&pred_c(s_1,\ldots,s_{M+1};\rho_1,\ldots,\rho_M)\geq \\
&\quad\frac{1}{2}\|P_M^T\nabla f(y_M)\|\min\{\mathcal{K}_4\|P_M^T\nabla f(y_M)\|,\delta_{M+1}\} \\
&\quad-\mu_M\sum_{j=1}^{M}\|C_j(y_{j-1})\| \\
&\quad+\rho_M cpred_M(s_1,\ldots,s_M;\rho_1,\ldots,\rho_{M-1})
\end{aligned}\tag{4.7}$$

Proof:

In the proof we omit the arguments in the predicted reduction expressions.

$$\begin{aligned}
cpred_k &= \|C_k(y_0)\|^2-\|C_k(y_{k-1})+\nabla C_k(y_{k-1})^Ts_k\|^2+\rho_{k-1}cpred_{k-1} \\
&= [\|C_k(y_0)\|^2-\|C_k(y_{k-1})\|^2] \\
&+ [\|C_k(y_{k-1})\|^2-\|C_k(y_{k-1})+\nabla C_k(y_{k-1})^Ts_k\|^2] \\
&+ \rho_{k-1}cpred_{k-1}.
\end{aligned}$$

Applying Lemma 4.3 and Lemma 4.4 to the right-hand side yields

$$\begin{aligned}
cpred_k &\geq \frac{1}{2}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\} \\
&- \mu_{k-1}\sum_{j=1}^{k-1}\|C_j(y_{j-1})\| \\
&+ \rho_{k-1}cpred_{k-1}.
\end{aligned}$$

Similarly, for the total predicted reduction we have

$$\begin{aligned}
pred &= f(y_0)-o_M(s_{M+1})+\rho_M cpred_M \\
&= [f(y_0)-f(y_M)] \\
&+ [f(y_M)-\phi_M(s_{M+1})] \\
&+ \rho_M cpred_M
\end{aligned}$$

$$\begin{aligned} &\geq \frac{1}{2}\|P_M^T \nabla f(y_M)\| \min\{\mathcal{K}_4 \|P_M^T \nabla f(y_M)\|, \delta_{M+1}\} \\ &- \mu_M \sum_{j=1}^{M} \|C_j(y_{j-1})\| \\ &+ \rho_M cpred_M, \end{aligned}$$

which concludes the proof. □

The following standard lemma provides an upper bound on the error between the actual reduction and the predicted reduction by repeated application of Lemma 1.6, Lemma 1.8, and the Cauchy-Schwarz inequality.

Lemma 4.6 Let $pred \equiv pred_c(s_1, \ldots, s_{M+1}; \rho_1, \ldots, \rho_M)$ and let $ared \equiv ared_c(s_1, \ldots, s_{M+1}; \rho_1, \ldots, \rho_M)$. Under assumptions AO1—AO5, there exist positive constants $\mathcal{K}_5$, $\mathcal{K}_6$, and $\mathcal{K}_7$, and $\nu_k, k = 1, \ldots, M$, independent of the iterates, such that

$$\begin{aligned} |ared - pred| &\leq \mathcal{K}_5 \|\hat{s}_c\|^2 \\ &+ \mathcal{K}_6 (\prod_{j=1}^{M} \rho_j) \|\hat{s}_c\|^3 + \sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j) \nu_k \|C_k(y_{k-1})\| \|\hat{s}_c\|^2 \end{aligned} \tag{4.8}$$

and

$$|ared - pred| \leq \mathcal{K}_7 (\prod_{j=1}^{M} \rho_j) \|\hat{s}_c\|^2. \tag{4.9}$$

Proof:

We have

$$\begin{aligned} |ared - pred| &= |f(y_0) - f(y_0 + s_1 + \ldots + s_{M+1}) \\ &+ \sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j)[\|C_k(y_0)\|^2 - \|C_k(y_0 + s_1 + \ldots + s_{M+1})\|^2] \\ &- \{f(y_0) - \phi_M(s_{M+1}) \\ &+ \sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j)[\|C_k(y_0)\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2]\}| \\ &= |\underbrace{-f(y_{M+1}) + f(y_M) + \nabla f(y_M)^T s_{M+1} + \frac{1}{2} s_{M+1}^T H_M s_{M+1}}_{A} \\ &+ \underbrace{\sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j)[-\|C_k(y_{M+1})\|^2 + \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2]\,|}_{B} \\ &\leq |A| + |B|. \end{aligned}$$

By Lemma 1.8, we have

$$\begin{aligned}
|A| &= |-f(y_M) - \nabla f(y_M)^T s_{M+1} - \frac{1}{2} s_{M+1}^T \nabla^2 f(z) s_{M+1} \\
&+ f(y_M) + \nabla f(y_M)^T s_{M+1} + \frac{1}{2} s_{M+1}^T H_M s_{M+1}| \\
&= \frac{1}{2} |s_{M+1}^T [H_M - \nabla^2 f(z)] s_{M+1}| \\
&\leq \frac{1}{2} \|H_M - \nabla^2 f(z)\| \|s_{M+1}\|^2 \\
&\leq \frac{1}{2} [\|H_M\| + \|\nabla^2 f(z)\|] \|s_{M+1}\|^2 \\
&\leq \frac{1}{2} [\sigma_3 + \sigma_4] \|s_{M+1}\|^2,
\end{aligned}$$

where $z \in (y_M, y_{M+1})$.

By Lemma 4.1,

$$\|s_{M+1}\| \leq \|\hat{s}_c\|.$$

Therefore,

$$|A| \leq \frac{1}{2} [\sigma_3 + \sigma_4] \|\hat{s}_c\|^2.$$

Now, since $\nabla C_k(y_{k-1})^T s_j = 0$ for $j = k+1, \ldots, M+1$, we have by Lemma 1.8

$$\begin{aligned}
|B| &= |\sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j) [\|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T (s_k + \ldots + s_{M+1})\|^2 \\
&- \|C_k(y_{k-1} + s_k + \ldots + s_{M+1})\|^2]| \\
&= |\sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j) \{\|C_k(y_{k-1})\|^2 + 2[\nabla C_k(y_{k-1}) C_k(y_{k-1})]^T (s_k + \ldots + s_{M+1}) \\
&+ \|\nabla C_k(y_{k-1})(s_k + \ldots + s_{M+1})\|^2 \\
&- \|C_k(y_{k-1})\|^2 - 2[\nabla C_k(y_{k-1}) C_k(y_{k-1})]^T (s_k + \ldots + s_{M+1}) \\
&- [\|\nabla C_k(z_k)^T (s_k + \ldots + s_{M+1})\|^2 \\
&- \frac{1}{2} (s_k + \ldots + s_{M+1})^T [\sum_{i=n_k}^{n_{k+1}-1} C_i(z_k) \nabla^2 C_i(z_k)] (s_k + \ldots + s_{M+1})]\}| \\
&= |\sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j) \{\|\nabla C_k(y_{k-1})^T (s_k + \ldots + s_{M+1})\|^2 \\
&- \|\nabla C_k(z_k)^T (s_k + \ldots + s_{M+1})\|^2 \\
&- \frac{1}{2} (s_k + \ldots + s_{M+1})^T [\sum_{i=n_k}^{n_{k+1}-1} C_i(z_k) \nabla^2 C_i(z_k)] (s_k + \ldots + s_{M+1})\}|,
\end{aligned}$$

where $z_k \in (y_{k-1}, y_{M+1}), k = 1, \ldots, M$.

Using Lemma 1.6, we have

$$\begin{aligned}
|B| &\leq \sum_{k=1}^{M}(\prod_{j=k}^{M} \rho_j) M \sigma_6 \sigma_7 \|s_k + \ldots + s_{M+1}\|^3 \\
&+ \sum_{k=1}^{M}(\prod_{j=k}^{M} \rho_j)(n_{k+1} - n_k + 1)\sigma_7 \|s_k + \ldots + s_{M+1}\|^2 \|C_k(y_{k-1})\| \\
&\leq (\prod_{j=1}^{M} \rho_j) M \sigma_6 \sigma_7 \|\hat{s}_c\|^3 \\
&+ \sum_{k=1}^{M}(\prod_{j=k}^{M} \rho_j)(n_{k+1} - n_k + 1)\sigma_7 \|\hat{s}_c\|^2 \|C_k(y_{k-1})\| \\
&\equiv (\prod_{j=1}^{M} \rho_j) \mathcal{K}_6 \|\hat{s}_c\|^3 + \sum_{k=1}^{M}(\prod_{j=k}^{M} \rho_j) \nu_k \|\hat{s}_c\|^2 \|C_k(y_{k-1})\|.
\end{aligned}$$

Denoting $\frac{1}{2}[\sigma_3 + \sigma_4]$ by $\mathcal{K}_5$ and putting A and B together, we have

$$\begin{aligned}
|ared - pred| &\leq \mathcal{K}_5 \|\hat{s}_c\|^2 \\
&+ (\prod_{j=1}^{M} \rho_j) \mathcal{K}_6 \|\hat{s}_c\|^3 + \sum_{k=1}^{M}(\prod_{j=k}^{M} \rho_j) \nu_k \|\hat{s}_c\|^2 \|C_k(y_{k-1})\|,
\end{aligned}$$

which establishes (4.8).

Continuing,

$$\begin{aligned}
|ared - pred| &\leq [\frac{\mathcal{K}_5}{\prod_{j=1}^{M} \rho_j} + \mathcal{K}_6 \delta_{max} + \sum_{j=k}^{M} \nu_k \sigma_5] \\
&\times (\prod_{j=1}^{M} \rho_j) \|\hat{s}_c\|^2 \\
&\leq [\mathcal{K}_5 + \mathcal{K}_6 \delta_{max} + \sum_{j=k}^{M} \nu_k \sigma_5] \\
&\times (\prod_{j=1}^{M} \rho_j) \|\hat{s}_c\|^2 \\
&\equiv \mathcal{K}_7 (\prod_{j=1}^{M} \rho_j) \|\hat{s}_c\|^2
\end{aligned}$$

since $\rho_k \geq 1$ for all k, which establishes (4.9) and completes the proof. □

Both the algorithm in [16], [20] and Algorithm 3.2 have the property that if an iterate is feasible, the penalty parameters are not increased. **We must show that if the iterates are sufficiently close to feasibility, then the penalty parameters will not be increased either.**

The lemma that answers this question is crucial for the proof of convergence. In the process of proving the lemma, it became apparent that the standard termination criterion was not satisfactory for the theory of a multilevel algorithm. This was discussed in Section 3.2.5.

Lemma 4.7 Assume that Algorithm 2.3 does not terminate because for some $k = 1, \ldots, M$, the termination condition is not satisfied, i.e.. $\|C_k(y_{k-1})\| > \epsilon_{tol}$. If $\sum_{j=1}^{k-1} \|C_j(y_{j-1})\| \leq \omega_{k-1}\hat{\delta}_c$, where ω_{k-1} is a positive constant that satisfies

$$\omega_{k-1} = \frac{\mathcal{K}_2 \epsilon_{tol}}{4\sqrt{M+1}\sum_{j=k-1}^{M} \mu_j} \min\{\frac{\sqrt{M+1}\mathcal{K}_3\epsilon_{tol}}{\delta_{max}}, 1\}, \tag{4.10}$$

then

$$\begin{aligned} &cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^-) \geq \\ &\quad \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\| \min\{\mathcal{K}_3\|C_k(y_{k-1})\|, \delta_k\} \\ &\quad +\rho_{k-1}^- cpred_{k-1}(s_1, \ldots, s_{k-1}; \rho_1^c, \ldots, \rho_{k-2}^c). \end{aligned} \tag{4.11}$$

If the algorithm does not terminate because the $(M+1)$-st termination criterion is not satisfied, i.e., $\|P_M^T \nabla f(y_M)\| > \epsilon_{tol}$, and if $\sum_{j=1}^{M} \|C_j(y_{j-1})\| \leq \omega_M \hat{\delta}_c$, where ω_M is a positive constant that satisfies

$$\omega_M = \frac{\epsilon_{tol}}{4\sqrt{M+1}\mu_M} \min\{\frac{\sqrt{M+1}\mathcal{K}_4\epsilon_{tol}}{\delta_{max}}, 1\}, \tag{4.12}$$

then

$$\begin{aligned} &pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) \geq \\ &\quad \frac{1}{4}\|P_M^T \nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T \nabla f(y_M)\|, \delta_{M+1}\} \\ &\quad +\rho_M^- cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c). \end{aligned} \tag{4.13}$$

Proof:

For this proof let

$$\begin{aligned} cpred_k &= cpred_k(s_1,\ldots,s_k;\rho_1^c,\ldots,\rho_{k-1}^-), \\ cpred_{k-1} &= cpred_{k-1}(s_1,\ldots,s_{k-1};\rho_1^c,\ldots,\rho_{k-2}^c). \end{aligned}$$

For $k=2,\ldots,M$, we have by Lemma 4.5,

$$\begin{aligned} cpred_k &\geq \frac{1}{2}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\} \\ &- \mu_{k-1}\sum_{j=1}^{k-1}\|C_j(y_{j-1})\| + \rho_{k-1}^- cpred_{k-1}. \end{aligned}$$

Since $\|C_k(y_{k-1})\| > \epsilon_{tol}$, we have

$$\begin{aligned} cpred_k &\geq \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\} \\ &+ \frac{1}{4}\mathcal{K}_2\epsilon_{tol}\min\{\mathcal{K}_3\epsilon_{tol},\delta_k\} \\ &- \mu_{k-1}\sum_{j=1}^{k-1}\|C_j(y_{j-1})\| + \rho_{k-1}^- cpred_{k-1}. \end{aligned}$$

Since $\delta_k = \frac{\hat{\delta}_c}{\sqrt{M+1}}$, we have

$$\begin{aligned} cpred_k &\geq \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\} \\ &+ \frac{\mathcal{K}_2\epsilon_{tol}\hat{\delta}_c}{4\sqrt{M+1}}\min\{\frac{\sqrt{M+1}\mathcal{K}_3\epsilon_{tol}}{\delta_{max}},1\} \\ &- \mu_{k-1}\omega_{k-1}\hat{\delta}_c + \rho_{k-1}^- cpred_{k-1}. \end{aligned}$$

Since

$$\begin{aligned} \omega_{k-1} &= \frac{\mathcal{K}_2\epsilon_{tol}}{4\sqrt{M+1}\sum_{j=k-1}^{M}\mu_j}\min\{\frac{\sqrt{M+1}\mathcal{K}_3\epsilon_{tol}}{\delta_{max}},1\} \\ &\leq \frac{\mathcal{K}_2\epsilon_{tol}}{4\sqrt{M+1}\mu_{k-1}}\min\{\frac{\sqrt{M+1}\mathcal{K}_3\epsilon_{tol}}{\delta_{max}},1\}, \end{aligned}$$

we have

$$\begin{aligned} cpred_k &\geq \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\|\min\{\mathcal{K}_3\|C_k(y_{k-1})\|,\delta_k\} \\ &+ \rho_{k-1}^- cpred_{k-1}, \end{aligned}$$

which proves (4.11).

Similarly, for $k = M + 1$, let $pred = pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-)$. Then by Lemma 4.5,

$$\begin{aligned} pred &\geq \frac{1}{2}\|P_M^T \nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T \nabla f(y_M)\|, \delta_{M+1}\} \\ &- \mu_M \sum_{j=1}^{M} \|C_j(y_{j-1})\| + \rho_{k-1}^- cpred_{k-1}, \end{aligned}$$

and since $\|P_M^T \nabla f(y_M)\| > \epsilon_{tol}$ and $\delta_{M+1} = \frac{\hat{\delta}_c}{\sqrt{M+1}}$, we have

$$\begin{aligned} pred &\geq \frac{1}{4}\|P_M^T \nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T \nabla f(y_M)\|, \delta_{M+1}\} \\ &+ \frac{\epsilon_{tol}\hat{\delta}_c}{4\sqrt{M+1}} \min\{\frac{\sqrt{M+1}\mathcal{K}_4\epsilon_{tol}}{\delta_{max}}, 1\} \\ &- \mu_M \omega_M \hat{\delta}_c + \rho_M^- cpred_M. \end{aligned}$$

Since

$$\omega_M = \frac{\epsilon_{tol}}{4\sqrt{M+1}\mu_M} \min\{\frac{\sqrt{M+1}\mathcal{K}_4\epsilon_{tol}}{\delta_{max}}, 1\},$$

we have (4.13), which concludes the proof. □

Thus we have a sufficient condition for any of the penalty parameters not to be increased.

Note that by construction $\omega_1 < \omega_2 < \ldots < \omega_M$, which is necessary for a later proof.

The following lemma provides a useful bound on the predicted reduction if the iterate is sufficiently close to feasibility.

Lemma 4.8 Assume that Algorithm 3.2 does not terminate because for some $k = 2, \ldots, M$, the termination criterion is not satisfied, i.e., $\|C_k(y_{k-1})\| > \epsilon_{tol}$, and that $\sum_{j=1}^{k-1} \|C_j(y_{j-1})\| \leq \omega_{k-1}\hat{\delta}_c$, where ω_{k-1} is defined in Lemma 4.7. Then there exists a positive constant τ_{k-1} such that

$$cpred_k(s_1, \ldots, s_k; \rho_1, \ldots, \rho_{k-1}) \geq \tau_{k-1}\hat{\delta}_c. \tag{4.14}$$

If $\|P_M^T \nabla f(y_M)\| > \epsilon_{tol}$ and $\sum_{j=1}^{M} \|C_j(y_{j-1})\| \leq \omega_M\hat{\delta}_c$, with ω_M defined in Lemma 4.7, then there exists a positive constant τ_M such that

$$pred(s_1, \ldots, s_{M+1}; \rho_1, \ldots, \rho_M) \geq \tau_M\hat{\delta}_c. \tag{4.15}$$

Proof:

In this proof, let

$$cpred_k = cpred_k(s_1, \ldots, s_k; \rho_1, \ldots, \rho_{k-1})$$

and let

$$pred = pred(s_1, \ldots, s_{M+1}; \rho_1, \ldots, \rho_M).$$

By Lemma 4.7,

$$\begin{aligned} cpred_k &\geq \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\| \min\{\mathcal{K}_3\|C_k(y_{k-1})\|, \delta_k\} + \rho_{k-1} cpred_{k-1} \\ &\geq \frac{1}{4}\mathcal{K}_2\|C_k(y_{k-1})\| \min\{\mathcal{K}_3\|C_k(y_{k-1})\|, \delta_k\} \\ &\geq \frac{1}{4}\mathcal{K}_2\epsilon_{tol} \min\{\mathcal{K}_3\epsilon_{tol}, \delta_k\} \\ &\geq \frac{1}{4}\mathcal{K}_2\epsilon_{tol}\frac{\hat{\delta}_c}{\sqrt{M+1}} \min\{\frac{\mathcal{K}_3\epsilon_{tol}}{\delta_{max}}, 1\}. \end{aligned}$$

Setting $\tau_{k-1} = \frac{1}{4\sqrt{M+1}}\mathcal{K}_2\epsilon_{tol}\min\{\frac{\mathcal{K}_3\epsilon_{tol}}{\delta_{max}}, 1\}$, we have

$$cpred_k \geq \tau_{k-1}\hat{\delta}_c.$$

Similarly,

$$\begin{aligned} pred &\geq \frac{1}{4}\|P_M^T\nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T\nabla f(y_M)\|, \delta_{M+1}\} + \rho_M cpred_M \\ &\geq \frac{1}{4}\|P_M^T\nabla f(y_M)\| \min\{\mathcal{K}_4\|P_M^T\nabla f(y_M)\|, \delta_{M+1}\} \\ &\geq \frac{1}{4}\epsilon_{tol} \min\{\mathcal{K}_4\epsilon_{tol}, \delta_{M+1}\} \\ &\geq \frac{1}{4}\frac{\epsilon_{tol}}{\sqrt{M+1}} \min\{\frac{\mathcal{K}_4\epsilon_{tol}}{\delta_{max}}, 1\}\hat{\delta}_c \\ &\equiv \tau_M\hat{\delta}_c, \end{aligned}$$

which completes the proof. □

4.5 The Behavior of the Penalty Parameters

As mentioned previously, to prove global convergence we must have the boundedness of the penalty parameters. This is achieved by establishing an upper bound on the product of the penalty parameters with the trust region radius, and then showing that the radii are bounded below. The penalty parameter sequences are shown to be nondecreasing, which, together with their boundedness from above, will allow us to conclude that the penalty parameters tend to a limit, and, moreover, stay constant after a finite number of outer iterations. The limit is shown to exist, but its explicit expression is not known.

In order to prove some of the lemmas below, a nested approach is required to take into account the multilevel nature of the algorithm.

The following lemma summarizes the properties of the penalty parameters that follow directly from the method of their updating.

Lemma 4.9 Let $\{\rho_k^i\}, k = 1, \ldots, M$, be the sequences of penalty parameters generated by the algorithm.

1. If ρ_k^i is not increased. then

$$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) \geq \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \tag{4.16}$$

and if ρ_M^i is not increased, then

$$pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) \geq \frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c). \tag{4.17}$$

2. If ρ_k^i is increased, it is increased by at least β, i.e., if

$$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) < \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c),$$

then $\rho_k^c - \rho_k^- \geq \beta$, and if

$$pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) < \frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c),$$

then $\rho_M^c - \rho_M^- \geq \beta$.

3. The sequences $\{\rho_k^i\}$ are nondecreasing.

Proof:

1. For $k = 1, \ldots, M-1$, if ρ_k is not increased, then

$$\rho_k^c = \rho_k^- \geq \frac{2[\|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 - \|C_{k+1}(y_0)\|^2]}{cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c)}$$

$$\|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 + \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \geq 0$$

$$\begin{aligned} \|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 &+ \rho_k^- cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \\ &\geq \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \end{aligned}$$

or

$$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) \geq \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c).$$

Similarly, if ρ_M is not increased, then

$$\rho_M^c = \rho_M^- \geq \frac{2[f(y_M) + \nabla f(y_M)^T s_{M+1} + \frac{1}{2} s_{M+1}^T H_M s_{M+1} - f(y_0)]}{cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)}$$

$$\begin{aligned} f(y_0) - [f(y_M) + \nabla f(y_M)^T s_{M+1} &+ \frac{1}{2} s_{M+1}^T H_M s_{M+1}] \\ &+ \frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c) \geq 0 \end{aligned}$$

$$\begin{aligned} f(y_0) - [f(y_M) &+ \nabla f(y_M)^T s_{M+1} + \frac{1}{2} s_{M+1}^T H_M s_{M+1}] \\ &+ \rho_M cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c) \geq \frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c) \end{aligned}$$

or

$$pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-) \geq \frac{\rho_M^-}{2} cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c).$$

2. By part 1, if

$$cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-) < \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c),$$

then the penalty parameter is increased. We have

$$\begin{aligned} \|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2 &+ \rho_k^- cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \\ &\leq \frac{\rho_k^-}{2} cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c) \end{aligned}$$

or

$$2[\|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2] \leq -\rho_k^- cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c)$$

and so

$$\frac{2[\|C_{k+1}(y_0)\|^2 - \|C_{k+1}(y_k) + \nabla C_{k+1}(y_k)^T s_{k+1}\|^2]}{cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c)} \leq -\rho_k^-.$$

By the updating procedure, the left-hand side of the inequality is $-\bar{\rho}_k^c$. Hence

$$\bar{\rho}_k^c \geq \rho_k^- \text{ or}$$

$$\rho_k^c - \beta \geq \rho_k^- \text{ and finally}$$

$$\rho_k^c - \rho_k^- \geq \beta.$$

Identical reasoning yields

$$\rho_M^c - \rho_M^- \geq \beta$$

if we replace $cpred_{k+1}(s_1, \ldots, s_{k+1}; \rho_1^c, \ldots, \rho_{k-1}^c, \rho_k^-)$ and $cpred_k(s_1, \ldots, s_k; \rho_1^c, \ldots, \rho_{k-1}^c)$ with $pred(s_1, \ldots, s_{M+1}; \rho_1^c, \ldots, \rho_{M-1}^c, \rho_M^-)$ and $cpred_M(s_1, \ldots, s_M; \rho_1^c, \ldots, \rho_{M-1}^c)$, respectively.

3. The method of updating ρ_k and part 2 directly imply that the sequences of penalty parameters are nondecreasing.

This concludes the proof. □

The next lemma establishes a relation between the trust region radii and the penalty parameters.

Lemma 4.10 For each $k = 1, \ldots, M$, there exists a constant Γ_k, independent of the iterates, such that if ρ_k^i is increased, then

$$\rho_k^i \, \delta_k^i \leq \Gamma_k. \tag{4.18}$$

Proof:

Since ρ_k^i and δ_k^i refer to a single outer iteration, we shall omit the superscripts i on all the entities in this proof.

Let $cpred_k$ denote $cpred_k(s_1, \ldots, s_k; \rho_1, \ldots, \rho_{k-1})$ for all $k = 1, \ldots, M$.

Let us consider the most general case first, i.e., suppose ρ_M is increased. We call this case the most general, because it requires the longest branching argument. The arguments for all other ρ_k start at a lower level of the argument "tree".

ρ_M is increased according to the formula

$$\begin{aligned}\rho_M &= \frac{2[\phi_M(s_{M+1})-\phi_0(0)]}{cpred_M}+\beta\\ &= \frac{2\overbrace{[\phi_M(s_{M+1})-f(y_0)]}^{A}}{cpred_M}+\beta\end{aligned}$$

Applying the triangle inequality, Lemma 1.8, and Assumption AO5, we have for some $z \in (y_M, y_{M+1})$,

$$\begin{aligned}|A| &= |\phi_M(s_{M+1})-f(y_{M+1})+f(y_{M+1})-f(y_0)|\\ &= |\phi_M(s_{M+1})-f(y_{M+1})|+|f(y_{M+1})-f(y_0)|\\ &\le \frac{1}{2}|s_{M+1}^T(H_M-\nabla^2 f(z))s_{M+1}|+\sup_{0\le t\le 1}\|\nabla f(y_0)+t(y_{M+1}-y_0))\|\,\underbrace{\|y_{M+1}-y_0\|}_{\hat{s}_c}\\ &\le \frac{1}{2}[\|H_M\|+\|\nabla^2 f(z)\|]s_{M+1}\|^2+\sigma_2\|\hat{s}_c\|\\ &\le \frac{1}{2}(\sigma_3+\sigma_4)\|\hat{s}_c\|^2+\sigma_2\|\hat{s}_c\|,\end{aligned}$$

and we have

$$\rho_M cpred_M \le \underbrace{2\sigma_2\|\hat{s}_c\|+(\sigma_3+\sigma_4)\|\hat{s}_c\|^2}_{B}+\beta cpred_M. \tag{4.19}$$

Since ρ_M is increased, by Lemma 4.6, $\sum_{j=1}^{M}\|C_j(y_{j-1})\| > \omega_M\hat{\delta}_c$. Let us consider two cases.

$\boxed{\text{Case 1:}}$ $\sum_{j=1}^{M-1}\|C_j(y_{j-1})\| \le \omega_{M-1}\hat{\delta}_c$.

Applying Lemma 4.7 to the left-hand side of (4.19), we have

$$\begin{aligned}&\rho_M[\frac{1}{4}\mathcal{K}_2\|C_M(y_{M-1})\|\min\{\mathcal{K}_3\|C_M(y_{M-1})\|,\delta_M\}+\rho_{M-1}cpred_{M-1}]\\ &\quad\le B+\beta[\|C_M(y_0)\|^2-\|C_M(y_{M-1})+\nabla C_M(y_{M-1})^T s_M\|^2+\rho_{M-1}cpred_{M-1}]\end{aligned}$$

or

$$\begin{aligned}&\frac{1}{4}\rho_M\mathcal{K}_2\|C_M(y_{M-1})\|\min\{\mathcal{K}_3\|C_M(y_{M-1})\|,\delta_M\}\\ &\quad\le B+\beta\underbrace{[\|C_M(y_0)\|^2-\|C_M(y_{M-1})+\nabla C_M(y_{M-1})^T s_M\|^2]}_{D}\\ &\quad+\rho_{M-1}(\beta-1)cpred_{M-1}\\ &\quad\le B+\beta D,\end{aligned}$$

since $\beta - 1 < 0$ ($\beta \in (0,1)$).

Now for some $z_M \in (y_{M-1}, y_M)$,

$$
\begin{aligned}
|D| &= \left| \|C_M(y_0)\|^2 - \|C_M(y_{M-1}) + \nabla C_M(y_{M-1})^T s_M\|^2 \right| \\
&\le \left| \|C_M(y_0)\|^2 - \|C_M(y_M)\|^2 \right| \\
&+ \left| \|C_M(y_M)\|^2 - \|C_M(y_{M-1}) + \nabla C_M(y_{M-1})^T s_M\|^2 \right| \\
&\le \sup_{0\le t\le 1} \|\nabla C_M(y_0 + t(y_M - y_0)) C_M(y_0 + t(y_M - y_0))\| \|y_M - y_0\| \\
&+ |s_M^T[\nabla C_M(y_{M-1})\nabla C_M(y_{M-1})^T - \frac{1}{2}\nabla C_M(z_M)\nabla C_M(z_M)^T \\
&- \frac{1}{2}\sum_{j=n_M}^{m} C_j(z_M)\nabla^2 C_j(z_M)] s_M| \\
&\le \sigma_5\sigma_6\|s_k + \ldots + s_M\| \\
&+ [\|\nabla C_M(y_{M-1})\|^2 + \frac{1}{2}\|\nabla C_M(z_M)\|^2 + \frac{1}{2}\sum_{j=n_M}^{m} \|C_j(z_M)\|\|\nabla^2 C_j(z_M)\|]\|s_M\|^2 \\
&\le \sigma_5\sigma_6\|s_k + \ldots + s_M\| + (\frac{3}{2}\sigma_6^2 + \frac{1}{2}\sum_{j=n_M}^{m}\sigma_6\sigma_7)\|s_M\|^2 \\
&\le \sigma_5\sigma_6\|\hat{s}_c\| + (\frac{3}{2}\sigma_6^2 + \frac{1}{2}\sum_{j=n_M}^{m}\sigma_6\sigma_7)\|\hat{s}_c\|^2.
\end{aligned}
$$

Putting B and D together, we have

$$
\begin{aligned}
&\frac{1}{4}\mathcal{K}_2\|C_M(y_{M-1})\| \min\{\mathcal{K}_3\|C_M(y_{M-1})\|, \delta_M\} \\
&\quad\le (2\sigma_2 + \beta\sigma_5\sigma_6)\|\hat{s}_c\| + [\sigma_3 + \sigma_4 + \beta(\frac{3}{2}\sigma_6^2 + \frac{1}{2}(m - n_M + 1)\sigma_6\sigma_7)]\|\hat{s}_c\|^2 \\
&\quad\le \underbrace{\{2\sigma_2 + \beta\sigma_5\sigma_6 + [\sigma_3 + \sigma_4 + \beta(\frac{3}{2}\sigma_6^2 + \frac{1}{2}(m - n_M + 1)\sigma_6\sigma_7)]\delta_{max}\sqrt{M+1}\}}_{E}\,\hat{\delta}_c.
\end{aligned}
$$

Since $\sum_{j=1}^{M}\|C_j(y_{j-1})\| > \omega_M\hat{\delta}_c$ and $\sum_{j=1}^{M-1}\|C_j(y_{j-1})\| \le \omega_{M-1}\hat{\delta}_c$, we have

$$\|C_M(y_{M-1})\| > (\omega_M - \omega_{M-1})\hat{\delta}_c > 0,$$

where the last inequality follows from $\omega_M > \omega_{M+1}$. Hence

$$\frac{1}{4}\rho_M\mathcal{K}_2(\omega_M - \omega_{M-1})\hat{\delta}_c \min\left\{\mathcal{K}_3(\omega_M - \omega_{M-1})\hat{\delta}_c, \frac{\hat{\delta}_c}{\sqrt{M+1}}\right\} \le E\hat{\delta}_c$$

$$\rho_M\hat{\delta}_c^2 \underbrace{[\frac{1}{4}\mathcal{K}_2(\omega_M - \omega_{M-1}) \min\left\{\mathcal{K}_3(\omega_M - \omega_{M-1}), \frac{1}{\sqrt{M+1}}\right\}}_{G} \le E\hat{\delta}_c$$

or

$$\rho_M \hat{\delta}_c G \leq E$$

and we have

$$\rho_M \hat{\delta}_c \leq \frac{E}{G}.$$

Denoting

$$\frac{E}{G} = \frac{2\sigma_2 + \beta\sigma_5\sigma_6 + [\sigma_3 + \sigma_4 + \beta(\frac{3}{2}\sigma_6^2 + \frac{1}{2}\sum_{j=n_M}^{m}\sigma_6\sigma_7)]\delta_{max}\sqrt{M+1}}{\frac{1}{4}\mathcal{K}_2(\omega_M - \omega_{M-1})\min\{\mathcal{K}_3(\omega_M - \omega_{M-1}), \frac{1}{\sqrt{M+1}}\}} = \Gamma_M^1,$$

we have

$$\rho_M \hat{\delta}_c \leq \Gamma_{M_1}.$$

$\boxed{\textbf{Case 2:}}$ $\sum_{j=1}^{M-1} \|C_j(y_{j-1})\| > \omega_{M-1}\hat{\delta}_c$.
Again consider two cases.
$\boxed{\textbf{Case 2.1:}}$ $\sum_{j=1}^{M-2} \|C_j(y_{j-1})\| \leq \omega_{M-2}\hat{\delta}_c$.
We have

$$\rho_M cpred_M \leq B + \beta cpred_M.$$

Applying Lemma 3.1 once to the left-hand side gives

$$\begin{aligned} \frac{\rho_M \rho_{M-1}}{2} cpred_{M-1} &\leq B + 3[\|C_M(y_0)\|^2 - \|C_M(y_{M-1}) + \nabla C_M(y_{M-1})^T s_M\|^2] \\ &+ 3\rho_{M-1} cpred_{M-1}, \end{aligned}$$

and since $\rho_j \geq 1$ for all j,

$$\frac{\rho_M}{2} cpred_{M-1} \leq B + D\hat{\delta}_c + \beta\rho_{M-1} cpred_{M-1}.$$

And now, following the argument similar to the one for Case 1 leads us to

$$\rho_M \hat{\delta}_c \leq \Gamma_{M_2}.$$

$\boxed{\textbf{Case 2.2:}}$ $\sum_{j=1}^{M-2} \|C_j(y_{j-1})\| > \omega_{M-2}\hat{\delta}_c$.
Again consider two cases.
$\boxed{\textbf{Case 2.2.1:}}$ $\sum_{j=1}^{M-3} \|C_j(y_{j-1})\| \leq \omega_{M-3}\hat{\delta}_c$ gives us

$$\rho_M \hat{\delta}_c \leq \Gamma_{M_3}.$$

$\boxed{\textbf{Case 2.2.2:}}$ $\sum_{j=1}^{M-3} \|C_j(y_{j-1})\| > \omega_{M-3}\hat{\delta}_c$ again gives us two cases to consider.

$$\vdots$$

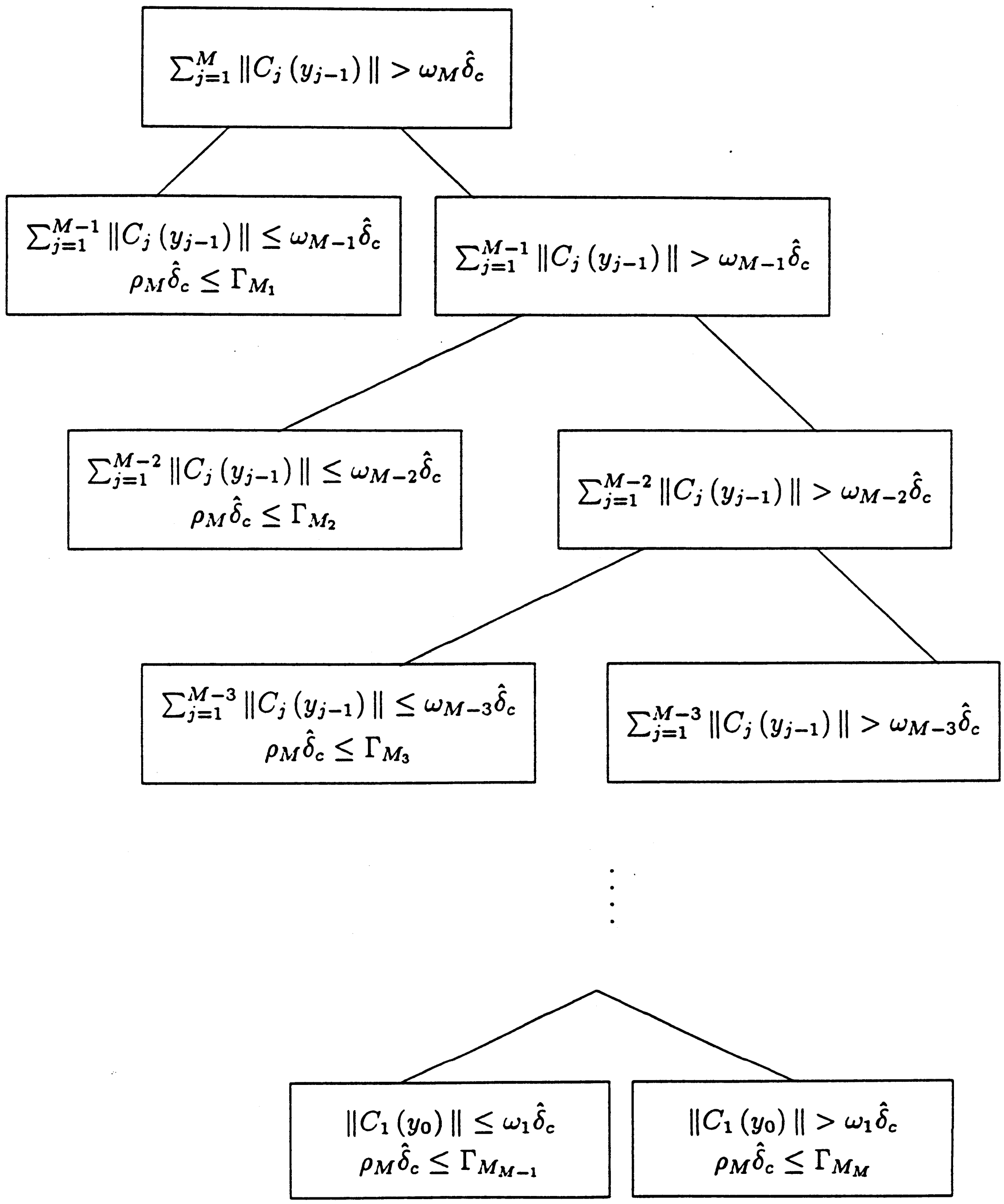

Figure 4.1 Flow chart for the proof of Lemma 4.10.

Continuing the process of branching, we obtain upper bounds on $\rho_M\hat{\delta}_c$ until we reach the branch on $\|C_1(y_0\|$.

For $\|C_1(y_0)\| \leq \omega_1\hat{\delta}_c$, the reasoning is identical, resulting in

$$\rho_M\hat{\delta}_c \leq \Gamma_{M_{M-1}}.$$

Finally, for $C_1(y_0) > \omega_1\hat{\delta}_c$, repeated application of Lemma 3.1 to (4.19) yields

$$\begin{aligned}\frac{\rho_1\rho_2\ldots\rho_M}{2^M}cpred_1 &\leq B + cpred_M \\ &= B + \beta[\|C_M(y_0)\|^2 - \|C_M(y_{M-1}) + \nabla C_M(y_{M-1})^T s_M\|^2] \\ &+ \beta\sum_{k=1}^{M-1}(\prod_{j=k}^{M-1}\rho_j)[\|C_k(y_0)\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2].\end{aligned}$$

Again, taking into account $\rho_j \geq 1$ for all j and applying Lemma 4.3 to the left-hand side, we have

$$\begin{aligned}&\frac{\rho_M}{2^{M+1}}\mathcal{K}_2\|C_1(y_0)\|\min\{\mathcal{K}_3\|C_1(y_0)\|, \delta_1\} \\ &\leq B + \beta\sum_{k=1}^{M}[\|C_k(y_0)\|^2 - \|C_k(y_{k-1}) + \nabla C_k(y_{k-1})^T s_k\|^2].\end{aligned}$$

Adding and subtracting $\|C_k(y_{k-1})\|^2$ from the right-hand side, using assumption AO5, and taking into account $\|C_1(y_0)\| > \omega_1\hat{\delta}_c$, we have

$$\begin{aligned}&\frac{\rho_M}{2^{M+1}}\mathcal{K}_2\omega_1\hat{\delta}_c\min\{\mathcal{K}_3\omega_1\hat{\delta}_c, \frac{\hat{\delta}_c}{\sqrt{M+1}}\} \\ &\leq B + \beta\sum_{k=1}^{M}[\sigma_5\sigma_6\|s_1 + \ldots + s_k\| + (\frac{3}{2}\sigma_6^2 + \frac{1}{2}\sum_{j=1}^{M}\sigma_6\sigma_7)\|s_k\|^2] \\ &B + \beta M\sigma_5\sigma_6\|\hat{s}_c\| + \beta M(\frac{3}{2}\sigma_6^2 + \frac{1}{2}M\sigma_6\sigma_7)\|\hat{s}_c\|^2] \\ &B + [\beta M\sigma_5\sigma_6 \ + \beta M(\frac{3}{2}\sigma_6^2 + \frac{1}{2}M\sigma_6\sigma_7)\sqrt{M+1}\,\delta_{max}]\hat{\delta}_c,\end{aligned}$$

or, replacing B with its value,

$$\begin{aligned}&\rho_M\hat{\delta}_c^2[\frac{1}{2^{M+1}}\mathcal{K}_2\omega_1\min\{\mathcal{K}_3\omega_1, \frac{1}{\sqrt{M+1}}\} \\ &\leq \{2\sigma_2 + \beta M\sigma_5\sigma_6 + [\sigma_3 + \sigma_4 + \beta M(\frac{3}{2}\sigma_6^2 + \frac{1}{2}M\sigma_6\sigma_7)]\sqrt{M+1}\,\delta_{max}\}\hat{\delta}_c.\end{aligned}$$

Denoting

$$\Gamma_{M_M} = \frac{\{2\sigma_2 + \beta M\sigma_5\sigma_6 + [\sigma_3 + \sigma_4 + \beta M(\frac{3}{2}\sigma_6^2 + \frac{1}{2}M\sigma_6\sigma_7)]\sqrt{M+1}\,\delta_{max}\}}{\frac{1}{2^{M+1}}\mathcal{K}_2\omega_1\min\{\mathcal{K}_3\omega_1, \frac{1}{\sqrt{M+1}}\}}$$

we have

$$\rho_M \hat{\delta}_c^2 \leq \Gamma_{M_M} \hat{\delta}_c$$

or

$$\rho_M \hat{\delta}_c \leq \Gamma_{M_M}.$$

Now setting $\Gamma_M = \max\{\Gamma_{M_1}, \ldots, \Gamma_{M_M}\}$, we have

$$\rho_M \hat{\delta}_c \leq \Gamma_M.$$

For $\rho_k, k = 1, \ldots, M-1$, the proofs start with the formula for updating that particular ρ_k, and then employ the same branching argument as for ρ_M. For example, suppose ρ_{M-1} is increased. This is done according to the formula

$$\rho_{M-1} = \frac{\|C_M(y_{M-1}) + \nabla C_M(y_{M-1})^T s_M\|^2 - \|C_M(y_0)\|^2}{cpred_{M-1}} + \beta.$$

Now we apply the triangle inequality, Lemma 1.8, and Assumption AO5 to obtain an upper bound on the numerator of the fraction, so that we have

$$\rho_{M-1} cpred_{M-1} \leq X + \beta cpred_{M-1},$$

where X is an expression that consists of a constant times $\|\hat{s}_c\|$ plus a constant times $\|\hat{s}_c\|^2$. From this inequality the argument proceeds analogously to Case 2 of the argument for ρ_M.

This concludes the proof. □

The Penalty Parameters Are Bounded Above

In Dennis, El-Alem, and Maciel [16], the boundedness of the trust region radii from below and hence the penalty parameters from above is established as two separate results.

Here, only after this pair of facts is proven for ρ_1 can we prove it for ρ_2, and then for ρ_3, etc. So, the two statements are combined in the following lemma. For ρ_1, however, the reasoning is similar to that of [16].

Lemma 4.11 Under assumptions AO1—AO5,

1. there exists a positive constant $\bar{\delta}$ such that if the algorithm does not terminate and any of the penalty parameters are increased, then we have

$$\hat{\delta}_c \geq \bar{\delta}. \tag{4.20}$$

2. For $k = 1, \ldots, M$, $\{\rho_k^i\}$ converges to a limit ρ_k^*. Moreover, there exist positive integers $i_{\rho_k}, k = 1, \ldots, M$, such that $\rho_k^i = \rho_k^*$ for all $i \geq i_{\rho_k}$.

Proof:

The general idea of the proof is to use the two upper bounds on the difference between the actual reduction and the predicted reduction from Lemma 4.6 and the lower bound on the predicted reduction from Lemma 3.1 to obtain an upper bound on the ratio of the difference of the reductions to the predicted reduction in terms of the step norm. We require both estimates from Lemma 4.6, because in order to obtain a suitable estimate, we have to eliminate its dependence on the penalty parameters. Appropriate powers of the step norm in the bounds given by Lemma 4.6 serve just this purpose.

We will first consider the case $k = 1$. Assume that ρ_1 is increased. Then by Lemma 4.7, $\|C_1(y_0)\| > \omega_1 \hat{\delta}_c$.

Let $\hat{s}_c = s_1 + \ldots + s_{M+1}$ be the trial step generated by the outer loop i and let $\hat{s}_-$ be the last acceptable step. Let the rejected steps between $\hat{s}_-$ and $\hat{s}_c$ be numbered $i_1, i_2, \ldots, i_L$, so that we have the sequence

$$\hat{s}_-, \underbrace{\hat{s}_{i_1}, \hat{s}_{i_2}, \ldots, \hat{s}_{i_L}}_{unacceptable}, \hat{s}_{i_{L+1}} = \hat{s}_c.$$

The current trial step $\hat{s}_c$ can be either acceptable or unacceptable.

There are three possibilities:

1. There are no unacceptable steps between $\hat{s}_-$ and $\hat{s}_c$, i.e., $\hat{s}_{i_1} = \hat{s}_c$.
2. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ for all $l = 1, \ldots, L+1$.
3. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ only for some of $l = 1, \ldots, L+1$.

Case 1: If there are no unacceptable steps between $\hat{s}_-$ and $\hat{s}_c$, then the method of updating the trust region radius ensures that

$$\hat{\delta}_c \geq \max\{\hat{\delta}_-, \delta_{min}\} \geq \delta_{min}. \tag{4.21}$$

Case 2: In this case, for all the unacceptable steps, $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$. Let

$$\begin{aligned} ared_{i_l} &\equiv ared_{i_l}(s_1^{i_l}, \ldots, s_{M+1}^{i_l}; \rho_1^{i_l}, \ldots \rho_M^{i_l}), \\ pred_{i_l} &\equiv pred_{i_l}(s_1^{i_l}, \ldots s_{M+1}^{i_l}; \rho_1^{i_l}, \ldots \rho_M^{i_l}). \end{aligned}$$

By Lemma 4.6,

$$|ared_{i_l} - pred_{i_l}| \le \mathcal{K}_7 (\prod_{j=1}^{M} \rho_j^{i_l}) \|\hat{s}_{i_l}\|^2.$$

By Lemma 3.1 and Lemma 4.3, we have

$$\begin{aligned} pred_{i_l} &\ge \frac{1}{2^M}(\prod_{j=1}^{M} \rho_j^{i_l})[\|C_1(y_0)\|^2 - \|C_1(y_0) + \nabla C_1(y_0)^T s_1^{i_l}\|^2 \\ &\ge \frac{1}{2^{M+1}}(\prod_{j=1}^{M} \rho_j^{i_l})\mathcal{K}_2 \|C_1(y_0)\| \min\{\mathcal{K}_3 \|C_1(y_0)\|, \delta_1^{i_l}\}. \end{aligned}$$

Since $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ and $\hat{\delta}_{i_l} = \sqrt{M+1}\,\delta_1^{i_l}$,

$$pred_{i_l} \ge \frac{1}{2^{M+1}}(\prod_{j=1}^{M} \rho_j^{i_l})\mathcal{K}_2 \|C_1(y_0)\| \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}\hat{\delta}_{i_l}.$$

Therefore, since $\|\hat{s}_{i_l}\| \le \hat{\delta}_{i_l}$,

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \le \frac{2^{M+1}\mathcal{K}_7 \|\hat{s}_{i_l}\|}{\mathcal{K}_2 \|C_1(y_0)\| \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}.$$

Since all the steps between $\hat{s}_-$ and $\hat{s}_c$ are rejected,

$$\frac{ared_{i_l}}{pred_{i_l}} < \eta_1,$$

where $\eta_1 \in (0,1)$ is defined in Algorithm 3.2, or

$$1 - \frac{ared_{i_l}}{pred_{i_l}} > 1 - \eta_1.$$

Hence

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} > 1 - \eta_1$$

and

$$\frac{2^{M+1}\mathcal{K}_7 \|\hat{s}_{i_l}\|}{\mathcal{K}_2 \|C_1(y_0)\| \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}} > 1 - \eta_1,$$

which yields

$$\|\hat{s}_{i_l}\| \ge \frac{(1-\eta_1)\mathcal{K}_2 \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}{2^{M+1}\mathcal{K}_7}\|C_1(y_0)\|, l = 1, \ldots, L.$$

We have $\hat{\delta}_c \equiv \hat{\delta}_{i_{L+1}} \geq \alpha_1 \|\hat{s}_{i_L}\|$, where α_1 is the trust region radius updating factor in the case of a trial step rejection. Since $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_1}$, it follows that

$$\hat{\delta}_c \geq \alpha_1 \|\hat{s}_{i_L}\| \geq \alpha_1 [\frac{(1-\eta_1)\mathcal{K}_2 \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}{2^{M+1}\mathcal{K}_7}] \omega_1 \hat{\delta}_{i_1}. \tag{4.22}$$

But since $\hat{\delta}_{i_1}$ is the first step after a successful step,

$$\hat{\delta}_{i_1} = \max\{\hat{\delta}_-, \delta_{min}\} \geq \delta_{min}.$$

Therefore

$$\hat{\delta}_c \geq \frac{\omega_1 \delta_{min} \alpha_1 (1-\eta_1)\mathcal{K}_2 \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}{2^{M+1}\mathcal{K}_7} \equiv \mathcal{K}_8.$$

Case 3: In this case only for some of the unacceptable steps $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$.

Let J be the largest index such that $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_{i_l}$. Since after each step is rejected, the trust region radius is decreased, the following situation holds:

$$\hat{s}_-, \underbrace{\overbrace{\hat{s}_{i_1}, \hat{s}_{i_2}, \ldots, \hat{s}_{i_J}}^{\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_{i_l}}, \overbrace{\hat{s}_{i_{J+1}}, \ldots, \hat{s}_{i_L}}^{\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}}}_{unaccepted}, \hat{s}_{i_{L+1}} = \hat{s}_c,$$

i.e., once $C_1(y_0) > \omega_1 \hat{\delta}_{i_J}$, then $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ holds for all $l = J+1, \ldots, L+1$.

The case $\hat{s}_{i_{J+1}} = \hat{s}_c$ is covered by Case 2, so that we have

$$\hat{\delta}_c \geq \alpha_1 \|\hat{s}_{i_J}\|.$$

If $\hat{s}_{i_{J+1}} \neq \hat{s}_c$, then from (4.22) we have

$$\hat{\delta}_c \geq \frac{(1-\eta_1)\omega_1 \alpha_1 \mathcal{K}_2 \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}{2^{M+1}\mathcal{K}_7} \|\hat{s}_{i_J}\|$$

because $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ for all $l = J+1, \ldots, L+1$. Letting

$$\mathcal{K}_9 = \min\{\frac{(1-\eta_1)\omega_1 \alpha_1 \mathcal{K}_2 \min\{\mathcal{K}_3 \omega_1, \frac{1}{\sqrt{M+1}}\}}{2^{M+1}\mathcal{K}_7}, \alpha_1\},$$

we have

$$\hat{\delta}_c \geq \mathcal{K}_9 \|\hat{s}_{i_J}\|.$$

Now, keeping the same notation for the actual and predicted reductions as in Case 2, by Lemma 4.6, we have

$$\begin{aligned} |ared_{i_l} - pred_{i_l}| &\leq \mathcal{K}_5 \|\hat{s}_{i_l}\|^2 \\ &+ \mathcal{K}_6 (\prod_{j=1}^{M} \rho_j) \|\hat{s}_{i_l}\|^3 + \sum_{k=1}^{M} (\prod_{j=k}^{M} \rho_j) \nu_k \|C_k(y_{k-1})\| \|\hat{s}_{i_l}\|^2. \end{aligned}$$

By Lemma 3.1 and 4.8, we have

$$\begin{aligned} pred_{i_l} &\geq \frac{\rho_2 \cdots \rho_M}{2^{M-1}} cpred_2^{i_l} \\ &\geq \frac{\rho_2 \cdots \rho_M}{2^{M-1}} \tau_1 \hat{\delta}_{i_l}. \end{aligned}$$

And since $\rho_j \geq 1$ for all j, we have

$$\begin{aligned} &\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \leq \\ &\frac{2^{M-1}[\mathcal{K}_5 \|\hat{s}_{i_l}\|^2 + \mathcal{K}_6 \rho_1 \|\hat{s}_{i_l}\|^3 + \sum_{k=2}^{M} \nu_k \|\hat{s}_{i_l}\|^2 \|C_k(y_{k-1})\| + \rho_1 \nu_1 \|\hat{s}_{i_l}\|^2 \|C_1(y_0)\|]}{\tau_1 \hat{\delta}_{i_l}} \\ &\leq \frac{2^{M-1}[\mathcal{K}_5 + \sigma_5 \sum_{k=2}^{M} \nu_k + \mathcal{K}_6 \rho_1 \|\hat{s}_{i_l}\| + \rho_1 \nu_1 \|C_1(y_0)\|] \|\hat{s}_{i_l}\|^2}{\tau_1 \hat{\delta}_{i_l}}. \end{aligned}$$

But $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_{i_l}$ and $\|\hat{s}_{i_l}\| \leq \hat{\delta}_{i_l}$. Hence

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \leq \frac{2^{M-1}[\mathcal{K}_5 + \sigma_5 \sum_{k=2}^{M} \nu_k + \mathcal{K}_6 \rho_1 \hat{\delta}_{i_l} + \rho_1 \nu_1 \omega_1 \hat{\delta}_{i_l}] \|\hat{s}_{i_l}\|^2}{\tau_1 \hat{\delta}_{i_l}}.$$

But $\rho_1 \hat{\delta}_{i_l} \leq \Gamma_1$. Hence

$$\begin{aligned} \frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} &\leq \frac{2^{M-1}[\mathcal{K}_5 + \sigma_5 \sum_{k=2}^{M} \nu_k + \mathcal{K}_6 \Gamma_1 + \Gamma_1 \nu_1 \omega_1] \|\hat{s}_{i_l}\|^2}{\tau_1 \hat{\delta}_{i_l}} \\ &\leq \underbrace{\frac{2^{M-1}[\mathcal{K}_5 + \sigma_5 \sum_{k=2}^{M} \nu_k + \mathcal{K}_6 \Gamma_1 + \Gamma_1 \nu_1 \omega_1]}{\tau_1}}_{\mathcal{K}_{30}} \|\hat{s}_{i_l}\|. \end{aligned}$$

Since $\hat{s}_{i_l}$ is rejected,

$$(1 - \eta_1) \leq \frac{2^{M-1} \mathcal{K}_{30}}{\tau_1} \|\hat{s}_{i_l}\|,$$

Therefore,

$$\|\hat{s}_{i_l}\| \geq \frac{(1 - \eta_1) \tau_1}{2^{M-1} \mathcal{K}_{30}} \tag{4.23}$$

and so

$$\hat{\delta}_c \geq \frac{\mathcal{K}_9 (1 - \eta_1) \tau_1}{2^{M-1} \mathcal{K}_{30}}.$$

Defining

$$\bar{\delta}_1 = \min \left\{ \delta_{min}, \mathcal{K}_8, \frac{\mathcal{K}_9 (1 - \eta_1) \tau_1}{2^{M-1} \mathcal{K}_{30}} \right\},$$

we have

$$\hat{\delta}_c \geq \bar{\delta}_1,$$

which proves the first part of the lemma for $k = 1$.

Now we know that $\{\rho_1^i\}$ is a nondecreasing sequence; we wish to show that it is bounded from above.

If ρ_1^i is increased, then by Lemma 4.10 and the preceding result on the boundedness of $\hat{\delta}_c$ from below we have

$$\rho_1^i \leq \frac{\Gamma_1}{\hat{\delta}_c} \leq \frac{\Gamma_1}{\bar{\delta}_1}.$$

Hence $\{\rho_1^i\}$ is a bounded nondecreasing sequence and therefore it converges to some $\rho_1^* < \infty$, i.e.,

$$\lim_{i \to \infty} \rho_1^i = \rho_1^*.$$

By Lemma 4.9, if ρ_1^i is increased, it is increased by at least $\beta > 0$. Therefore, since $\{\rho_1^i\}$ converges to a finite number, the number of increases has to be finite, i.e.,

$$\rho_1^i = \rho_1^{i_{\rho_1}}$$

for some index i_{ρ_1} and all $i \geq i_{\rho_1}$. Hence both results of the lemma are established for ρ_1.

Next consider the case $k = 2$. Now we assume that ρ_2 is increased and show that $\hat{\delta}_c$ is bounded from below.

If ρ_1 is increased, then the case $k = 1$ applies. So, assume that ρ_1 is not increased. Since ρ_2 is increased, by Lemma 4.7,

$$\|C_1(y_0)\| + \|C_2(y_1)\| > \omega_2 \hat{\delta}_c.$$

If $\|C_1(y_0)\| > \omega_1 \hat{\delta}_c$, then this situation is covered by the case $k = 1$. So, assume that $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_c$. Then $\|C_2(y_1)\| > (\omega_2 - \omega_1)\hat{\delta}_c > 0$, since $\omega_2 > \omega_1$. Since $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_c$, using Lemma 4.7 and $\|C_2(y_1)\| > (\omega_2 - \omega_1)\hat{\delta}_c$, we obtain the estimate

$$\begin{aligned}
cpred_2 &\geq \frac{1}{4}\mathcal{K}_2 \|C_2(y_1)\| \min\{\mathcal{K}_3 \|C_2(y_1)\|, \frac{\hat{\delta}_c}{\sqrt{M+1}}\} + \rho_1 cpred_1 \\
&\geq \frac{1}{4}\mathcal{K}_2 \|C_2(y_1)\| \min\{\mathcal{K}_3 \|C_2(y_1)\|, \frac{\hat{\delta}_c}{\sqrt{M+1}}\} \\
&\geq \frac{1}{4}\mathcal{K}_2 \|C_2(y_1)\| \min\{(\omega_2 - \omega_1)\mathcal{K}_3, \frac{1}{\sqrt{M+1}}\}\hat{\delta}_c.
\end{aligned}$$

Now we consider three cases:

1. There are no unacceptable steps between $\hat{s}_-$ and $\hat{s}_c$, i.e. $\hat{s}_{i_1} = \hat{s}_c$.

2. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_2(y_1)\| > (\omega_2 - \omega_1)\hat{\delta}_{i_l}$ for all $l = 1, \ldots, L+1$.

3. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_2(y_1)\| > (\omega_2 - \omega_1)\hat{\delta}_{i_l}$ only for some of $l = 1, \ldots, L+1$.

Case 1: The reasoning is identical to the one for the case of $k = 1$.

Case 2: In this case, following the same reasoning as for $k = 1$, we now use

$$pred_{i_l} \geq \frac{\rho_2 \ldots \rho_M}{2^{M-1}} cpred_2$$

from Lemma 3.1 to arrive to the estimate

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \leq \frac{2^M \mathcal{K}_7 \rho_1 \|\hat{s}_{i_l}\|}{\mathcal{K}_2 \|C_2(y_1)\| \min\{\mathcal{K}_3(\omega_2 - \omega_1), \frac{1}{\sqrt{M+1}}\}}.$$

However, we proved that $\rho_1 \leq \rho_1^*$, so

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \leq \frac{2^M \mathcal{K}_7 \rho_1^*}{\mathcal{K}_2 \|C_2(y_1)\| \min\{\mathcal{K}_3(\omega_2 - \omega_1), \frac{1}{\sqrt{M+1}}\}} \|\hat{s}_{i_l}\|,$$

and the rest of the argument is identical to that of the case $k = 1$.

Case 3: Here again we use $\rho_1 \leq \rho_1^*$ to remove the dependence of the estimates on the penalty parameters, and the argument proceeds identically to the case $k = 1$ to yield

$$\hat{\delta}_c \geq \bar{\delta}_2.$$

Now, if ρ_2 is increased, we have

$$\rho_2^i \leq \frac{\Gamma_2}{\hat{\delta}_c} \leq \frac{\Gamma_2}{\bar{\delta}_2}$$

and we obtain that $\rho_2^i \to \rho_2^*$ and that there exists an index i_{ρ_2} such that

$$\rho_2^i = \rho_2^{i_{\rho_2}}$$

for all $i \geq i_{\rho_2}$.

$$\vdots$$

Continuing this procedure, at the general step k, we have the estimate

$$\frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \leq \frac{2^{K+1} \mathcal{K}_7 \rho_1 \ldots \rho_{k-1}}{\mathcal{K}_2 \|C_k(y_{k-1})\| \min\{\mathcal{K}_3(\omega_k - \omega_{k-1}), \frac{1}{\sqrt{M+1}}\}} \|\hat{s}_{i_l}\|.$$

But at this point, $\rho_1, \ldots, \rho_{k-1}$ have been shown to be bounded by $\rho_1^*, \ldots, \rho_{k-1}^*$, respectively, thus eliminating the dependence of the estimates on the penalty parameters. The rest of the argument proceeds identically to the case $k = 1$ to show that $\hat{\delta}_c$ is bounded and this result is then used to show that the sequence $\{\rho_k^i\}$ is bounded from above by ρ_k^*.

Setting $\bar{\delta} = \min\{\bar{\delta}_1, \ldots, \bar{\delta}_M\}$ concludes the proof. □

4.6 The Trust Region Radius Is Bounded Below

We have shown that the total trust region radius $\hat{\delta}_c$ is bounded away from zero if any of the penalty parameters are increased. Now we will show that $\hat{\delta}_c$ is always bounded away from zero. The trust region updating strategy ensures that $\hat{\delta}_c$ is bounded from above.

> **Theorem 4.1** Assume that the algorithm does not terminate. Then under assumptions AO1—AO5, there exists a constant $\hat{\delta}_* > 0$, independent of the iterates, such that
>
> $$\hat{\delta}_i \geq \hat{\delta}_* \quad \text{for all } i. \tag{4.24}$$

Proof:

The proof follows the same lines as in Lemma 4.11, but we do not have to consider separate cases based on which penalty parameter is increased. Regardless of which termination criterion is not satisfied, using the notation of Lemma 4.11 we consider three cases:

1. There are no unacceptable steps between $\hat{s}_-$ and $\hat{s}_c$, i.e. $\hat{s}_{i_1} = \hat{s}_c$.
2. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ for all $l = 1, \ldots, L+1$.
3. $\hat{s}_{i_1} \neq \hat{s}_c$ and $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ only for some of $l = 1, \ldots, L+1$.

Case 1: By the same reasoning as in Lemma 4.11 we have

$$\hat{\delta}_c \geq \max\{\hat{\delta}_-, \delta_{min}\} \geq \delta_{min}.$$

Case 2: The proof is identical to that of Lemma 4.11.

Case 3: Here $\|C_1(y_0)\| > \omega_1 \hat{\delta}_{i_l}$ does not hold for at least one $l = 1, \ldots, L$. As in Lemma 4.11, letting J be the largest index such that $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_{i_l}$ for all $i = i_1, \ldots, i_J$, we have

$$\hat{\delta}_c \geq \mathcal{K}_9 \|\hat{s}_{i_J}\|. \tag{4.25}$$

By Lemma 4.6, $|ared_{i_l} - pred_{i_l}| \leq \mathcal{K}_7 (\prod_{j=1}^M \rho_j) \|\hat{s}_{i_l}\|^2$. By Lemma 4.8, $cpred_2 \geq \tau_1 \hat{\delta}_{i_l}$ and so by Lemma 3.1,

$$pred_{i_l} \geq (\frac{\prod_{j=2}^M}{2^{M-1}} \rho_j^{i_l}) cpred_2 \geq (\frac{\prod_{j=2}^M}{2^{M-1}} \rho_j)^{i_l} \tau_1 \hat{\delta}_{i_l}.$$

Therefore, since $\hat{s}_{i_l}$ is rejected,

$$\begin{aligned}
(1 - \eta_1) &\leq \frac{|ared_{i_l} - pred_{i_l}|}{pred_{i_l}} \\
&\leq \frac{2^{M-1} \mathcal{K}_7 \rho_1^{i_l} \|\hat{s}_{i_l}\|^2}{\tau_1 \hat{\delta}_{i_l}} \\
&\leq \frac{2^{M-1} \mathcal{K}_7 \rho_1^* \|\hat{s}_{i_l}\|}{\tau_1}.
\end{aligned}$$

Hence

$$\|\hat{s}_{i_l}\| \geq \frac{(1 - \eta_1)\tau_1}{\mathcal{K}_7 \rho_1^*}.$$

Then by (4.25) we have

$$\hat{\delta}_c \geq \frac{(1 - \eta_1)\tau_1 \mathcal{K}_9}{\mathcal{K}_7 \rho_1^*}.$$

Letting

$$\hat{\delta}_* = \min\left\{ \frac{(1 - \eta_1)\tau_1 \mathcal{K}_9}{\mathcal{K}_7 \rho_1^*}, \delta_{min} \right\},$$

we have $\hat{\delta}_c \geq \hat{\delta}_*$, which concludes the proof. □

4.7 The Algorithm is Well-Defined

The following theorem guarantees that the algorithm is well defined, i.e., that after a finite number of outer loop iterations an acceptable step $\hat{s}_c$ with

$$\frac{ared}{pred} \geq \eta_1$$

will be found.

Theorem 4.2 Unless the current iterate x_c satisfies the termination criteria of the algorithm, an acceptable step $\hat{s}_c$ from x_c will be found after a finite number of trials.

Proof:

In this proof we omit the arguments in the reduction functions *ared* and *pred*.

Assume that the algorithm does not terminate. Regardless of which termination criterion is violated, it suffices to consider two cases.

Case 1: $\|C_1(y_0)\| > \omega_1 \hat{\delta}_c$, where ω_1 is defined in Lemma 4.7.

By Lemmas 3.1 and 4.3, we have

$$\begin{aligned} pred &\geq \frac{1}{2^M}(\prod_{j=1}^{M} \rho_j) cpred_1 \\ &\geq \frac{1}{2^{M+1}}(\prod_{j=1}^{M} \rho_j)\mathcal{K}_2\|C_1(y_0)\| \min\{\mathcal{K}_3\|C_1(y_0)\|, \frac{\hat{\delta}_c}{\sqrt{M+1}}\} \\ &\geq \frac{1}{2^{M+1}}(\prod_{j=1}^{M} \rho_j)\mathcal{K}_2\|C_1(y_0)\| \min\{\mathcal{K}_3\omega_1, \frac{1}{\sqrt{M+1}}\}\hat{\delta}_c. \end{aligned}$$

By Lemma 4.6, the last inequality gives us

$$\begin{aligned} \frac{|ared - pred|}{pred} &\leq \frac{2^{M+1}\mathcal{K}_7\|\hat{s}_c\|^2}{\mathcal{K}_2\|C_1(y_0)\| \min\{\mathcal{K}_3\omega_1, \frac{1}{\sqrt{M+1}}\}\hat{\delta}_c} \\ &\leq \frac{2^{M+1}\mathcal{K}_7}{\mathcal{K}_2\|C_1(y_0)\| \min\{\mathcal{K}_3\omega_1, \frac{1}{\sqrt{M+1}}\}}\hat{\delta}_c. \end{aligned}$$

The last line shows that

$$\left|\frac{ared}{pred} - 1\right|$$

approaches 0 as $\hat{\delta}_c$ becomes smaller. Therefore the criterion

$$\frac{ared}{pred} \geq \eta_1 \in (0, 1)$$

will be satisfied after a finite number of outer loop iteration.

Case 2: $\|C_1(y_0)\| \leq \omega_1 \hat{\delta}_c$.

By Lemma 3.1 and Lemma 4.8, we have

$$\begin{aligned} pred &\geq \frac{1}{2^{M-1}}(\prod_{k=2}^{M} \rho_k) cpred_2 \\ &\geq \frac{1}{2^{M-1}}(\prod_{k=2}^{M} \rho_k)\tau_1\hat{\delta}_c. \end{aligned}$$

Hence

$$\begin{aligned} \frac{|ared - pred|}{pred} &\leq \frac{2^{M-1}\mathcal{K}_7\rho_1\hat{\delta}_c^2}{\tau_1\hat{\delta}_c} \\ &\leq \frac{2^{M-1}\mathcal{K}_7\rho_1^*}{\tau_1}\hat{\delta}_c. \end{aligned}$$

Again the ratio goes to 0 with decreasing $\hat{\delta}_c$, so the acceptance criterion will be satisfied after finitely many trials. □

4.8 The Global Convergence Theorem

In the global convergence result, we show that if the objective function is bounded below, then the sequence of iterates generated by Algorithm 3.2 has a subsequence convergent to a stationary point of the equality constrained minimization problem.

The proof proceeds in stages, block-by-block, but the argument for each component of the proof is almost identical to the argument in Dennis, El-Alem, and Maciel [16].

Theorem 4.3 Assume that $f(x)$ is bounded below. Then, given any $\epsilon_{tol} > 0$, the algorithm will terminate because

$$\begin{aligned} \|C_k(y_{k-1}^i)\| &\leq \epsilon_{tol}, k = 1, \ldots, M \text{ and} && (4.26) \\ \|P_M^T \nabla f(y_M)\| &\leq \epsilon_{tol} && (4.27) \end{aligned}$$

will hold simultaneously for some i.

Proof:

Since $\rho_k, k = 1, \ldots, M$, are bounded below by 1 and the norms of the constraint blocks are bounded below by 0, the boundedness of f implies that the merit function $\bar{\mathcal{P}}(x)$ is bounded below.

Assume that the algorithm does not terminate.

Case 1: $k = 1$.

Assume that there exists $\theta > 0$ such that $\|C_1(y_0^i)\| > \theta$ for all i. Let

$$J \geq \max\{i_{\rho_1}, \ldots, i_{\rho_M}\},$$

that is, J is an index for which every ρ_k has reached its upper bound. We have for all $y \in \Omega$.

$$\|C_1(y)\| \geq \|C_1(y_0^J)\| - \|C_1(y) - C_1(y_0^J)\| \geq \|C_1(y_0^J)\| - \sigma_6\|y - y_0^J\|.$$

Let $a = \frac{\|C_1(y_0^J)\|}{2\sigma_6}$ and consider a ball $\mathcal{B}$ of radius a centered at y_0^J. We have for all $y \in \mathcal{B}$

$$\|C_1(y)\| \geq \frac{1}{2}\|C_1(y_0^J)\|.$$

Now assume that for all $i > J, y_0^i \in \mathcal{B}$. Then from Lemmas 3.1 and 4.3 we have

$$\begin{aligned} pred_i &\geq \frac{\rho_1^i \cdots \rho_M^i}{2^M}[\|C_1(y_0^i)\|^2 - \|C_1(y_0^i) + \nabla C_1(y_0^i)^T s_1^i\|^2] \\ &\geq \frac{\mathcal{K}_2}{2^{M+1}}\|C_1(y_0^i)\| \min\{\mathcal{K}_3\|C_1(y_0^i)\|, \frac{\hat{\delta}_i}{\sqrt{M+1}}\} \\ &\geq \frac{\mathcal{K}_2}{2^{M+1}}\|C_1(y_0^J)\| \min\{\mathcal{K}_3\|C_1(y_0^J)\|, \frac{\hat{\delta}_i}{\sqrt{M+1}}\} \\ &\geq \frac{\mathcal{K}_2\theta}{2^{M+1}} \min\{\mathcal{K}_3\theta, \frac{\hat{\delta}_-}{\sqrt{M+1}}\} \\ &\equiv \mathcal{K}_{10} > 0. \end{aligned}$$

where the last inequality follows from the boundedness of $\hat{\delta}_i$ from below.

Recall that $\bar{\mathcal{P}}$ denotes our merit function introduced in Section 3.2.1.

Since the algorithm is well-defined, we will find an acceptable step after finitely many outer loop iterations, and for all y_0^i in the ball $i \geq J$, we have

$$\bar{\mathcal{P}}_i - \bar{\mathcal{P}}_{i+1} = ared_i \geq \eta_1 pred_i \geq \mathcal{K}_{10}. \tag{4.28}$$

which means that $\bar{\mathcal{P}}$ is increased by a positive value infinitely many times. This contradicts the assumption that f and therefore $\bar{\mathcal{P}}$ are bounded from below. Thus, eventually $\{y_0^i\}$ must leave $\mathcal{B}$.

Let l be the smallest index among the i's which are greater than J such that y_0^{l+1} is outside $\mathcal{B}$. Since $y_0^{l+1} \neq y_0^J$, there exists at least one acceptable step between the iterates J and l. For these acceptable steps we have

$$\bar{\mathcal{P}}_J - \bar{\mathcal{P}}_{l+1} = \sum_{t=J}^{l}(\bar{\mathcal{P}}_t - \bar{\mathcal{P}}_{t+1}) = \sum_{t=J}^{l} ared_t \geq \sum_{t=J}^{l} \eta_1 pred_t. \tag{4.29}$$

And now by (4.28),

$$\bar{\mathcal{P}}_J - \bar{\mathcal{P}}_{l+1} \geq \eta_1\mathcal{K}_{10} \equiv \mathcal{K}_{11} > 0. \tag{4.30}$$

The sequence $\bar{\mathcal{P}}_i$ has a limit $\bar{\mathcal{P}}_-$, since it is decreasing and bounded below. Let $J \to \infty$; then $l = l(J) \to \infty$ and (4.30) gives us a contradiction. Therefore, $\|C_1(y_0^i)\|$ cannot be bounded away from zero, i.e., there exists a subsequence $\{i_j\}$ of $\{i\}$ with

$$\lim_{i_j \to \infty} \|C_1(y_0^{i_j})\| = 0.$$

Case 2: $k = 2$.

Now assume that $\|C_2(y_1^i)\|$ is bounded away from 0, i.e., that there exists ξ such that $\|C_2(y_1^i)\| > \xi$ always holds. Since $\|C_1(y_0^{i_j})\|$ goes to zero, and the the sequence of trust region radii is bounded below, there exists an iteration $J_1 > 0$, such that for all $i_j \geq J_1, \|C_1(y_0^{i_j})\| \leq \omega_1 \hat{\delta}_{i_j}$, where ω_1 is given in Lemma 4.7. Then by Lemma 3.1 and Lemma 4.8, we have

$$\begin{aligned} pred_{i_j} &\geq (\prod_{p=2}^{M} \rho_p) \frac{1}{2^{M-1}} cpred_2^{i_j} \\ &\geq \frac{1}{2^{M-1}} cpred_2^{i_j} \\ &\geq \frac{1}{2^{M-1}} \tau_1 \hat{\delta}_{i_j} \\ &\geq \frac{1}{2^{M-1}} \tau_1 \hat{\delta}_{*}. \end{aligned}$$

But for any l among the i_j's greater than i_{ρ_1}, we have

$$\bar{\mathcal{P}}_l - \bar{\mathcal{P}}_{l+1} = ared_l \geq \eta_1 pred_l \geq \frac{1}{2^{M-1}} \eta_1 \tau_1 \hat{\delta}_{*} > 0. \tag{4.31}$$

As we take the limit as l goes to infinity, we obtain a contradiction. Therefore, $\|C_2(y_1^i)\|$ cannot be bounded away from zero and there exists a subsequence $\{i_{j_l}\}$ of i_j such that

$$\lim_{i_{j_l} \to \infty} \|C_2(y_1^{i_{j_l}})\| = 0.$$

Now suppose that $\|C_3(y_2^i)\|$ is bounded away from zero. Since both $\|C_2(y_1^{i_{j_l}})\|$ and $\|C_1(y_0^{i_{j_l}})\|$ go to zero, there exists an index $J_2 > 0$, such that for all $i_{j_l} > J_2$,

$$\|C_1(y_0^{i_{j_l}})\| + \|C_2(y_1^{i_{j_l}})\| \leq \omega_2 \hat{\delta}_{i_{j_l}}$$

and the argument proceeds as for $k = 1$ and $k = 2$.

We apply the same argument to each subsequent block of components and, finally to $\|P_M^T \nabla f(y_M)\|$. For each case we show that there exists a subsequence of indices for which the norm of each block and finally the norm of the projected gradient go to zero.

This means that

$$\liminf_{i \to \infty} [\|P_M^T \nabla f(y_M^i)\| + \sum_{k=1}^{M} \|C_k(y_{k-1}^i)\|] = 0,$$

which contradicts the assumption that the algorithm does not terminate and concludes the proof. □

By Lemma 4.2, the norms of the substeps are bounded from above by constant multiples of the norms of the constraint blocks. Therefore, as the norm of each block goes to zero, so does the norm of the substep. Since all the projection matrices are assumed to be continuous and bounded above and below, we have the following immediate corollary of Theorem 4.3.

Corollary 4.1 Under the assumptions of Theorem 4.3,

$$\liminf_{i\to\infty}[\|P^T\nabla f(x_i)\| + \|C(x_i)\|] = 0.$$

Thus, at least a subsequence of the generated sequence of iterates converges to a stationary point of problem EQC.

4.9 Corollaries

We can now conclude that the multilevel algorithm for nonlinear equations and Algorithm 3.1 are also globally convergent.

- **Algorithm 2.2. (Nonlinear Equation Solver):** The assumption on the boundedness of f from below is automatically satisfied in the case of nonlinear equations, since $\|F(x)\|^2 \geq 0$. The only assumption that we place on H_M in the optimization algorithm is that it must be bounded from above. The role of H_M in the equation solver is played by $J_M(y_{M-1})^T J_M(y_{M-1})$ which is assumed to be bounded from above.

  Therefore, we can consider Theorem 2.1 proven.

- **Algorithm 3.1:** Algorithm 3.1 is just Algorithm 3.2 with a single block of constraints and thus a single penalty parameter. Therefore, all arguments concerning Algorithm 3.2 apply to Algorithm 3.1.

The global convergence theory presented here is applicable to the theory presented in [16] in the case that Lagrange multipliers are taken to be zero.

Chapter 5

Implementation

In this chapter we discuss the implementation problems that must be resolved for the algorithm to become robust. We support some of the conclusions by the results of preliminary testing.

Preliminary testing was done only for the equation solver. The following test set consists of sixteen nonlinear equation problems extracted from the Moré, Garbow, Hillstrom [27] test set and one linear problem from Noble and Daniel [2]:

1. Rosenbrock Function
2. Powell Singular Function
3. Powell Badly Scaled Function
4. Wood Function
5. Helical Valley Function
6. Watson Function
7. Chebyquad Function
8. Brown Almost-Linear Function
9. Discrete Boundary Value Function
10. Discrete Integral Equation Function
11. Trigonometric function
12. Variably Dimensioned Function
13. Broyden Tridiagonal Function
14. Broyden Banded Function

15. Freudenstein and Roth Function

16. Box 3-dimensional Function

17. Linear Function

The problems are stated in Appendix B. Since all the testing was done for the problem of solving nonlinear systems of equations, "equations" rather than "constraints", will be mentioned frequently in the following discussion.

Notation: "max" in the tables signifies that the problem failed to converge in 500 major iterations; † means that the algorithm converged to a stationary point of $\|F(x)\|^2$ which is not a root of $F(x)$.

5.1 Trust Region Updating Strategies

In standard trust region globalization methods, the trust region updating strategies are directly related to the step evaluation procedures. The accumulated wisdom based on computational experience suggests halving the size of the trust region if the step is unacceptable, leave it unchanged if the step performs "reasonably" well, and to double it if the step produces a very good ratio between the actual and the predicted reductions in the merit function.

In the case of the multilevel algorithms, while it is clear how to evaluate the trial step, it is not at all clear how the trust region radii should be updated. The theory does not object to the trust region radii being increased. but it does require that they not be decreased unless necessary. The current strategy of contracting or expanding the subproblem trust regions simultaneously based on the ratio of the total actual reduction to the predicted reduction works for most problems, but with this strategy the method is not as robust the single block trust region method.

With the current updating strategy, the method proved to be sensitive to the initial trust region setting. We attempt three different initial settings:

1. The initial value is set to the length of the longest unconstrained Brent substep for all the subproblems.

2. The initial value is set to the shortest unconstrained Brent substep for all the subproblems.

3. Individual initial values of the length of the unconstrained Brent substeps are set for every subproblem.

Tables 5.1 and 5.2 compare the number of iterations to convergence for these three settings for two different starting points. The last column contains the number of iterations for the standard one-block case.

The strategy of internal doubling has not yet proven effective because it is applicable only to the first substep.

Further trials are necessary, but preliminary testing indicates that there is a correlation between the algorithm's behavior with respect to the trust region initial setting and the relative nonlinearity of the equations. If a comparatively large block of linear equations precedes a small nonlinear block, the algorithm benefits from a large initial radius setting. When all the equations are nonlinear or linear and nonlinear equations are mixed, a small initial setting gives better results.

The problem is, of course, that different components may exhibit very different behaviors based on their nonlinearity. It is possible that a poor predicted versus actual behavior of one component may force unnecessary shrinking of the trust region radii for all subproblems. One approach to preventing this situation is a strategy that will allow substep evaluation and trust region updating within the inner loop. If the total step is rejected, such a strategy will permit us to return to the last successful inner iterate instead of returning to the first iterate. A disadvantage of the intermediate substep evaluation is a possible increase in the number of function evaluations.

Table 5.3 compares the number of function and Jacobian evaluations for one of the multilevel algorithm variations (the initial radius is set to the minimum Brent step norm) to the subroutine LMDER from the MINPACK package. For test purposes, we use exact derivatives. With the exception of Problems 3 and 12, the number of evaluations is comparable, but at present the LMDER code is much more robust than the multilevel algorithm with respect to different starting points. However, we should point out that the multilevel algorithm is coded exactly as it is stated in the thesis and it is very far from being a "production" code. Table 5.3 is provided only to give a very general idea of the number of function evaluations in a very preliminary stage of the code.

To conclude, the related issues of the step evaluation and trust region updating are still under investigation.

Problem No.	δ_k^{init} = max	δ_k^{init} = min	δ_k^{init} = indiv	One Block
1	2	2	2	17
2	11	12	11	10
3	50	max	50	max
4	68	37	53	22
5	9	7	8	9
6	363	51	244	13
7	3	4	3	4
8	3	3	3	6
9	3	5	3	5
10	3	3	3	3
11	4	4	4	4
12	max	88	max	20
13	4	5	4	5
14	5	5	5	5
15	7	7	7	7
16	16	6	45	4
17	1	2	2	1

Table 5.1 Three initial radii strategies. Starting point 1.

Problem No.	$\delta_k^{init} =$ max	$\delta_k^{init} =$ min	$\delta_k^{init} =$ indiv	One Block
1	4	2	2	6
2	15	14	15	13
3	5	6	5	$2^{\dagger}$
4	53	37	31	39
5	12	7	15	7
6	max	71	max	90
7	14	5	8	33
8	6	5	4	18
9	7	6	7	10
10	4	4	4	4
11	6	6	6	7
12	max	7	max	26
13	8	7	8	7
14	11	6	11	11
15	28	8	24	13
16	50	7	6	5
17	2	2	2	1

Table 5.2 Three initial radii strategies. Starting point 2.

Problem No.	Mult. nfev	Mult. njev	LMDER nfev	LMDER njev
1	2	2	21	16
2	11	11	25	25
3	50	50	18	16
4	37	37	32	27
5	7	7	11	8
6	51	51	45	37
7	4	4	8	6
8	3	3	14	12
9	5	5	6	5
10	3	3	5	4
11	4	4	8	8
12	88	88	22	21
13	5	5	6	5
14	5	5	7	6
15	7	7	8	7
16	6	6	6	5
17	2	2	3	2

Table 5.3 Multilevel vs. LMDER: function evaluations.

5.2 Projectors

In order to resrict the models of the constraints and the objective function to the intersection of the null spaces of the previously processed constraint blocks, we have to find a matrix whose columns form a basis for the intersection of the appropriate null spaces.

The null space of a matrix and the orthogonal projection onto it are uniquely defined. However, a basis for the null space is not uniquely determined.

The global convergence theory requires for the basis matrix to be continuous, bounded away from zero, and bounded from above. In practice, the matrix should be easily computed.

In this section we consider two specific choices for the generic matrix P used in the description of the algorithm.

5.2.1 Using QR-Factorization

Let $A \in \Re^{m \times n}, m \leq n$. A common approach to computing a basis for the null space of A is based on the QR decomposition of A^T:

$$A^T = R\,Q\,\Pi^T \equiv [Q_1|Q_2]\,R\,\Pi^T,$$

where $Q \in \Re^{n \times n}$ is orthonormal, $R \in \Re^{n \times m}$ is upper-triangular, and $\Pi \in \Re^{m \times m}$ is a column pivoting permutation matrix.

Let $rank(A^T) = r$. Then the first r columns of Q, denoted by Q_1, form an orthonormal basis for the column space of A^T, while the last $n - r$ columns, denoted by Q_2 form an orthonormal basis for the null space of A. The QR decomposition allows us to estimate the numerical rank of A.

Now consider two matrices $A \in \Re^{p \times n}$ and $B \in \Re^{q \times n}$. To find an orthonormal basis for the intersection of $\mathcal{N}(A)$ and $\mathcal{N}(B)$, we could just compute the null space of the matrix C formed by appending B to A:

$$C = \begin{bmatrix} A \\ B \end{bmatrix}$$

However, it is more efficient to compute the basis based on the following theorem proved in Golub and Van Loan [13], p. 583:

> **Theorem 5.1** Given $A \in \Re^{p \times n}$ and $B \in \Re^{q \times n}$, let Z be an orthonormal basis for $\mathcal{N}(A)$. If W is an orthonormal basis for $\mathcal{N}(BZ)$, then the columns of ZW form an orthonormal basis for $\mathcal{N}(A) \bigcap \mathcal{N}(B)$.

The use of QR factorization has several advantages:

1. Orthogonal transformations are stable and do not cause deterioration in conditioning of the problem.

2. While theoretically A must have full rank, in practice the drop in the rank does not seem to affect the performance adversely. The QR factorization allows us to compute the basis for the null space when A does not have full rank.

3. Adding rows to A does not necessitate a complete recomputation of the basis matrix. For instance, Theorem 5.1 above can be used to update Q.

4. The basis matrices are trivially bounded.

5. The software for the QR factorization is widely available.

There are also drawbacks.

1. For the proof of convergence, we have to assume that the basis matrix is continuous. Coleman and Sorensen [35] pointed out that the usual method of computing the QR factorization via Housholder transformations may not result in a continuous basis matrix. They suggest three variations on the standard methods that ensure the continuity of the computed basis, but the region in which the basis is continuous is not under the user's control. In addition, discontinuities may occur in arbitrarily small regions about A.

   Given a continuous function $A(x)$, Byrd and Schnabel [31] showed that it is not possible in general to compute an orthonormal basis $Q(x)$ of A as a continuous function of x, although it is possible to do so in special cases.

   In practice, however, the modifications proposed by Coleman and Sorensen and by Gill, Murray, Saunders, Stewart and Wright [30] will be helpful in improving the continuity properties of the orthonormal basis.

2. The QR factorization software for large, sparse problems has not been fully developed.

3. The QR factorization is expensive.

5.2.2 Using Reduced Basis Projectors

The second approach to finding a basis for $\mathcal{N}(A)$ arises from partitioning $A \in \Re^{m\times n}$ with $rank(A) = m$ as

$$A = [B|N],$$

where B is an $m \times m$ nonsingular matrix. Here we assume that the columns of A are interchanged so that B is formed by the first m columns of A.

Then the matrix

$$Z = \begin{bmatrix} -B^{-1}N \\ I_{n-m} \end{bmatrix}$$

forms a basis for the null space of A.

Given our assumptions AO1-AO5, Z for the Jacobians of the constraint blocks are clearly continuous and bounded.

There is a number of drawbacks.

1. As the algorithm progresses, we cannot guarantee that, once selected, the same m columns of A will stay linearly independent. Thus pivoting may be required. While this may be a workable option, it has not been considered yet either theoretically or practically.

2. Not just in theory, but also in practice A must have full rank for the invertibility of B.

3. Software for computing Z is not as readily available as for computing the QR factorization.

5.3 Order of Constraints

Testing even on a limited set of problems confirms the expectation that the proposed algorithms are sensitive to the order of constraints or equations.

Although the methods do work on the problems composed only of nonlinear—and sometimes highly nonlinear—equations, they seem ideally suited for the problems in which a large block of linear or nearly linear equations preceeds the more nonlinear of the equations.

For the test problems of dimension $n \leq 4$, all permutations of the equations were tested. Table 5.4 shows that the difference in the number of iterations to convergence for different permutations can be quite dramatic.

Optimal partitioning of equations is a research topic. Martinez [22], [23], investigated some equation arrangements. At present there is no rule for arranging the equations other than attempting to place the most linear ones first. However, the "linear first" placement may result in great improvements in efficiency. Thus, if the algorithm is to be used over and over again, it is a worthwhile undertaking to establish the best possible ordering for a particular problem.

5.4 Scaling

The initial setting of the penalty parameters to 1 is arbitrary and presupposes that the problem is well-scaled, i.e.. that the constraint blocks are well scaled with respect to each other and to the objective function. If this is not true in practice, then the constraints should be re-scaled or the initial choice of the penalty parameter setting should be adjusted. The only requirement is that it must be greater than or equal to one.

Perm. No.	Pr.1	Pr.2	Pr.3	Pr.4	Pr.5	Pr.15	Pr.16
1	2	11	50	68	9	7	16
2	8	11	13	53	10	6	7
3		11		29	6		17
4		10		31	6		6
5		11		43	11		12
6		10		37	6		8
7		11		31			
8		11		50			
9		11		75			
10		11		33			
11		11		48			
12		11		48			
13		11		32			
14		10		41			
15		11		39			
16		11		62			
17		10		45			
18		10		56			
19		11		57			
20		10		25			
21		11		46			
22		11		41			
23		10		42			
24		10		14			

Table 5.4 Order sensitivity.

5.5 Penalty Parameters

It is well known that too rapidly growing penalty parameters will adversely affect the performance of the algorithm. Preliminary testing has indicated that in the problems which are stopped because they reach the maximum allowable limit of 500 iterations the penalty parameters become large quickly.

Robust trust region algorithms with nonmonotonic penalty parameters have been investigated by El-Alem [10]. The nonmonotonic updating scheme must be incorporated into the multilevel algorithms.

Chapter 6

Applications and Conclusions

In this chapter we say a few words about the expected applications of the proposed multilevel algorithms, state the plan of research, and then conclude with the highlights of the dissertation.

6.1 Applications

The proposed algorithms are expected to be very useful in the scientific and engineering applications that require their constraint systems to be processed by blocks.

The design of complex engineering systems is by nature a multicriteria optimization problem. The design projects are distinguished by very large numbers of variables, constraints, and expensive analyses. To solve the problem, it is necessary to break it into disciplines, each of which produces its own optimal design. The discipline designs are then incorporated into a total design. The multilevel methods proposed here would allow researchers to integrate constraints obtained from different sources.

To solve the multicriteria optimization problem, it is necessary to decide when an iterate is optimal. One of the approaches to optimality is the statement of the multicriteria problem as a multilevel optimization problem, i.e., the problem of minimizing a function on a feasible set, which is an optimal set for another function, and so on. In such an approach, the user places priorities on the optimization problems that are to be solved sequentially. We believe that the multilevel algorithms proposed here will serve as a beginning for a detailed study of the general multilevel optimization problem.

6.2 Research Plan

The multilevel algorithms proposed in this thesis together with their global convergence theory will serve as a basis for investigating several promising directions.

- For practical purposes, it is important to extend the material of this thesis to multilevel algorithms based on Brown's method. We are interested in Brown-based algorithms, because they will be less expensive than the Brent-based algorithms and they will provide a better opportunitiy for taking advantage of sparsity. The theory should also allow us to prove convergence for the entire class of Brown-Brent based methods.

- Another theoretical issue that must be studied is the local convergence properties of the multilevel algorithms. We believe that it may be necessary to modify the merit function to include the Lagrange multipliers in order to prove convergence for the optimization algorithm. We suspect that although the algorithm attempts to take unconstrained Brent steps whenever possible, the current merit function may not recognize these steps as acceptable ones near a solution.

- The issue of implementation is extremely important if the algorithm is to become practically usable. We must investigate different step evaluation and trust region updating strategies, both from theoretical and from practical points of view. Extensive testing of the algorithms will be done for small and for large problems.

- The next step is to consider extensions of the multilevel algorithms to bound constraints and inequality constraints.

- The new algorithm for constrained optimization is a multilevel optimization algorithm with a very special structure: all functions, but the outermost one, are norms squared of the constraint blocks. In the algorithm for nonlinear equations, all functions are of this form.

  A natural extension would be a multilevel optimization algorithm for solving the general multilevel optimization problem (MLO). The extension is difficult to accomplish. A crucial feature of our algorithm is that, despite its multilevel optimization structure, some of its crucial theoretical and practical features are based on the equality constrained optimization problem in its conventional form EQC. For example, we use the first order necessary optimality conditions for problem EQC as a convergence criterion for our multilevel algorithm. The regularity assumption for the constraints plays a vital role in the theory. No such clear-cut, familiar foundation exists for the general multilevel optimization problem. Theoretically and practically, it is not clear how the feasible set should be described or even how optimality should be recognized. Despite the difficulties, we believe that the new algorithms provide sufficient foundation to initiate a study of problem MLO.

6.3 Summary

In this study, we presented a class of multilevel algorithms for solving the nonlinear equations problem and the equality constrained optimization problem.

The new algorithms are extensions of the local algorithms of Brown and Brent for solving nonlinear systems of equations.

The main practical appeal of the multilevel algorithms is that in the case of equality constrained optimization, they allow the user to partition the constraint system arbitrarily and to process the blocks of constraints separately. In their finite-difference derivative form, they require fewer function evaluations than the Newton's method.

The multilevel methods differ from the standard trust region algorithms in that their major iteration involves finding an approximate minimizer of not one quadratic model over a single restricted region, but a sweep of quadratic models, each approximately minimized over its own trust region. Each model is computed at a different point.

We presented a global convergence theory for the algorithm optimization based on Brent's method. The theory implies convergence of the nonlinear equations solver, which, to the author's knowledge, is the first theoretically supported method for globalizing Brent's algorithm via the trust region strategy. The global convergence theory was made possible by the introduction of the merit function with nested penalty parameters updated by a modification of the scheme proposed by El-Alem [8].

The algorithms are expected to be applicable to the problem of the multidisciplinary design optimization and to serve as a foundation for the study of the general multilevel optimization problem.

Appendix A

Glossary of the Lemmas and Theorems

In this appendix we list all the Lemmas, Theorems, and Corollaries by chapter, in the order of appearance. Each is followed by a brief description.

IN CHAPTER 1 ...

The lemmas of this chapter are some well-known results in unconstrained minimization, analysis, and algebra stated without proof and used throughout this dissertation.

Lemma 1.1

This lemma gives the solution of the quadratic subproblem used in the trust region approach to unconstrained minimization.

Lemma 1.2

This lemma gives the solution of the quadratic subproblem used in the trust region approach to the nonlinear least squares problem. It is a special case of Lemma 1.1.

Lemma 1.3

The Levenberg-Marquardt step is orthogonal to the nullspace of the Jacobian of residuals.

Lemma 1.4

This lemma expresses the Fraction of Cauchy Decrease condition in the form of a useful inequality.

Theorem 1.1

This is Powell's global gonvergence theorem for the trust region algorithms for unconstrained minimization.

Lemma 1.5

This gives the least norm solution of the linear least squares problem.

Lemma 1.6

This lemma provides an estimate on the difference between two values of a continuous function.

Lemma 1.7

This lemma gives us a linear approximation of a differentiable function.

Lemma 1.8

This lemma gives us a quadratic approximation of a twice continuously differentiable function.

IN CHAPTER 2 ...

The propositions of this chapter describe some properties of two possible subproblem solutions.

Proposition 2.1

This is a statement of the relationship between the Levenberg-Marquardt step and the unconstrained Brent step for the subproblems of the multilevel nonlinear equation solver.

Proposition 2.2

This result ensures that if the unconstrained Brent step, whose norm is greater than the trust region radius, is shortened to the length of the trust region radius, it has the Fraction of Cauchy Decrease property with respect to the inner-loop subproblem.

Theorem 2.1

This is a global convergence theorem for the multilevel nonlinear equations solver.

IN CHAPTER 3 ...

Lemma 3.1

This lemma shows that the strategy for updating the penalty parameters guarantees that the partial and full predicted reductions satisfy the Fraction of Cauchy Decrease condition.

IN CHAPTER 4 ...

This chapter contains the main theory of the dissertation.

Lemma 4.1

This lemma ensures that the total trial step is no shorter than the sum of the substeps that comprise it.

Lemma 4.2

This lemma bounds the norm of each substep from above by a constant times the norm of the constraint block that gave rise to this particular substep.

Lemma 4.3

This lemma expresses the Fraction of Cauchy Decrease condition in a workable form for each subproblem of the inner loop.

Lemma 4.4

This lemma establishes a lower bound on the effect of the step s_k on $\|C_j(y_{j-1})\|^2, j = k+1,\ldots,M$ and on the objective function.

Lemma 4.5

This lemma places a lower bound on the partial and total predicted decreases.

Lemma 4.6

This lemma places an upper bound on the difference between the actual reduction and the predicted reduction which approximates it.

Lemma 4.7

This lemma determines how close to feasibility the inner loop iterates must be in order for the penalty parameters not to be increased.

Lemma 4.8

This lemma bounds the partial and total predicted reductions from below by a constant times the trust region radius when the inner loop iterates are sufficiently close to feasibility.

Lemma 4.9

This lemma summarizes the basic properties of the penalty parameters that follow from the strategy for their updating.

Lemma 4.10

This lemma stablishes a relationship between the trust region radii and the penalty parameters.

Lemma 4.11

This lemma bounds the trust region radii from below when the penalty parameters are increased and, at the same time, it bounds the penalty parameters from above.

Theorem 4.1

This theorem states that the trust region radii are bounded below.

Theorem 4.2

This theorem states that the algorithm is well-defined, i.e., that, given a current outer iterate x_c, an acceptable step will be found in a finite number of outer iterates.

Theorem 4.3

This is the main result of the dissertation. The theorem ensures that the algorithm will terminate because at some outer iterate the stopping criterion will be satisfied.

Corollary 4.1

As a consequence of Theorem 4.3 and the properties of the substeps, we conclude that at least a subsequence of the sequence of iterates generated by the algorithm converges to a stationary point of the equality constrained minimization problem.

Appendix B

Test Problems

The first sixteen of the following problems were taken from the Moré, Garbow, and Hillstrom [27] test set. The last one was taken from Noble and Daniel [2].

Rosenbrock Function

$n = 2$

$$\begin{aligned} F_1(x) &= 10(x_2 - x_1^2) \\ F_2(x) &= 1 - x_1 \\ x_0 &= (-1.2, 1) \\ f &= 0 \text{ at } (1, 1) \end{aligned}$$

Powell Singular Function

$n = 4$

$$\begin{aligned} F_1(x) &= x_1 + 10x_2 \\ F_2(x) &= \sqrt{5}(x_3 - x_4) \\ F_3(x) &= (x_2 - 2x_3)^2 \\ F_4(x) &= \sqrt{10}(x_1 - x_4)^2 \\ x_0 &= (3, -1, 0, 1) \\ f &= 0 \text{ at } (0, 0, 0, 0) \end{aligned}$$

Powell Badly Scaled Function

$n = 2$

$$\begin{aligned} F_1(x) &= 10^4 x_1 x_2 - 1 \\ F_2(x) &= e^{-x_1} + e^{-x_2} - 1.0001 \end{aligned}$$

$$\begin{aligned} x_0 &= (0,1) \\ f &= 0 \text{ at } (1.098\ldots 10^{-5}, 9.106\ldots) \end{aligned}$$

Wood Function

$n = 4$

$$\begin{aligned} F_1(x) &= -200x_1(x_4 - x_3^2) - (1 - x_1) \\ F_2(x) &= 200(x_2 - x_1^2) + 20(x_2 - 1) + 19.8(x_4 - 1) \\ F_3(x) &= -180x_3(x_4 - x_3^2) - (1 - x_3) \\ F_4(x) &= 180(x_4 - x_3^2) + 20.2(x_4 - 1) + 19.8(x_2 - 1) \\ x_0 &= (-3, -1, -3, -1) \\ f &= 0 \text{ at } (1, 1, 1, 1) \end{aligned}$$

Helical Valley Function

$n = 3$

$$\begin{aligned} F_1(x) &= 10[x_3 - 10\theta(x_1, x_2)] \\ F_2(x) &= 10[\sqrt{x_1^2 + x_2^2} - 1] \\ F_3(x) &= x_3 \\ x_0 &= (-1, 0, 0) \\ f &= 0 \text{ at } (1, 0, 0), \end{aligned}$$

where

$$\theta(x_1, x_2) = \begin{cases} \frac{1}{2\pi} \arctan\left(\frac{x_2}{x_1}\right), & \text{if } x_1 > 0 \\ \frac{1}{2\pi} \arctan\left(\frac{x_2}{x_1}\right) + 0.5, & \text{if } x_1 < 0 \end{cases}$$

Watson Function

$n = 6$

$$\begin{aligned} F_i(x) &= \sum_{j=2}^{n} (j-1) x_j t_i^{j-2} - (\sum_{j=1}^{n} x_j t_i^{j-1})^2 - 1 \\ x_0 &= (-3, -1, -3, -1), \end{aligned}$$

where $t_i = \frac{i}{29}, i = 1, \ldots, n$.

Chebyquad Function

$n = 5$

$$\begin{aligned} F_i(x) &= \frac{1}{n}\sum_{j=1}^{n} T_i(x_j) - \int_0^1 T_i(x)dx \\ x_0 &= (\xi_j), \end{aligned}$$

where $\xi_j = \frac{j}{n+1}$ and T_i is the i-th Chebyshev polynomial shifted to the interval $[0,1]$, i.e.

$$\begin{aligned} \int_0^1 T_i(x)dx &= 0 \text{ for } i \text{ odd}, \\ \int_0^1 T_i(x)dx &= \frac{-1}{i^2-1} \text{ for } i \text{ even}. \end{aligned}$$

Brown Almost-Linear Function

$n = 10$

$$\begin{aligned} F_i(x) &= x_i + \sum_{j=1}^{n} x_j - (n+1) \\ F_n(x) &= (\prod_{j=1}^{n} x_j) - 1 \\ x_0 &= (\frac{1}{2}, \dots, \frac{1}{2}), \\ F &= 0 \text{ at } (\alpha, \dots, \alpha, \alpha^{1-n}), \end{aligned}$$

where α satisfies $n\alpha^n - (n+1)\alpha^{n-1} + 1 = 0$; in particular, $\alpha = 1$.

Discrete Boundary Value Problem

$n = 10$

$$\begin{aligned} F_i(x) &= 2x_i - x_{i-1} + \frac{h^2(x_i + t_i + 1)^3}{2} \\ x_0 &= (\xi_j), \end{aligned}$$

where $\xi_j = t_j(t_j - 1), h = \frac{1}{n+1}, t_i = ih$.

Discrete Integral Equation Function

$n = 10$

$$\begin{aligned} F_i(x) &= x_i + \frac{h[(1-t_i)\sum_{j=1}^{i} t_j(x_j+t_j+1)^3 + t_i)\sum_{j=i+1}^{n}(1-t_j)(x_j+t_j+1)^3]}{2} \\ x_0 &= (\xi_j), \end{aligned}$$

where $\xi_j = t_j(t_j-1), h = \frac{1}{n+1}, t_i = ih.$

Trigonometric function

$n = 10$

$$\begin{aligned} F_i(x) &= n - \sum_{j=1}^{n} \cos x_j + i(1-\cos x_i) - \sin x_i \\ x_0 &= (\frac{1}{n}, \ldots, \frac{1}{n}). \end{aligned}$$

Variably Dimensioned Function

$n = 10$

$$\begin{aligned} t_1 &= \sum_{j=1}^{n} j(x_j - 1) \\ t_2 &= t_1(1+2t_1^2) \\ F_i(x) &= x_i - 1 + it_2 \\ x_0 &= (\xi_j) \\ F = 0 &= \text{ at } (1, \ldots, 1) \end{aligned}$$

where $\xi_j = 1 - \frac{j}{n}$.

Broyden Tridiagonal Function

$n = 10$

$$\begin{aligned} F_i(x) &= (3-2x_i)x_i - x_{i-1} - 2x_{i+1} + 1 \\ x_0 &= (-1, \ldots, -1). \end{aligned}$$

Broyden Banded Function

$n = 10$

$$\begin{aligned} F_i(x) &= x_i(2+5x_i^2)+1-\sum_{j\in J_i} x_j(1+x_j) \\ x_0 &= (-1,\ldots,-1), \end{aligned}$$

where $J_i = \{j | j \neq i, \max\{1, i-m_l\} \leq j \leq \min\{n, i+m_u\}\}$ and $m_l = 5, m_u = 1$.

Freudenstein and Roth Function

$n = 2$

$$\begin{aligned} F_1(x) &= -13 + x_1 + ((5-x_2)x_2 - 2)x_2 \\ F_2(x) &= -29 + x_1 + ((x_2+1)x_2 - 14)x_2 \\ x_0 &= (0.5, -2) \\ F &= 0 \text{ at } (5, 4) \end{aligned}$$

Box 3-dimensional Function

$n = 3$

$$\begin{aligned} F_i(x) &= e^{-t_i x_1} - e^{-t_i x_2} - x_3[e^{-t_i} - e^{-10t_i}] \\ x_0 &= (0, 10, 20) \\ F &= 0 \text{ at } (1, 10, 1), (10, 1, -1) \\ & \quad \text{and whenever } x_1 = x_2 \text{ and } x_3 = 0, \end{aligned}$$

where $t_i = 0.1i$.

Linear Function 1

$n = 3$

$$\begin{aligned} F_1(x) &= 3x_1 - x_2 + 2x_3 - 12 \\ F_2(x) &= x_1 + 2x_2 + 3x_3 - 11 \\ F_3(x) &= 2x_1 - 2x_2 - x_3 - 2 \\ x_0 &= (-5, -5, -5) \\ F &= 0 \text{ at } (3, 1, 2). \end{aligned}$$

Bibliography

[1] M. Avriel. *Nonlinear Programming.* Prentice-Hall, 1976.

[2] Ben Noble and James W. Daniel. *Applied Linear Algebra.* Prentice-Hall, 1988.

[3] Richard P. Brent. Some efficient algorithms for solving systems of nonlinear equations. *SIAM J. Numerical Analysis*, 10(2):327—344, April 1973.

[4] Kenneth M. Brown. *A Quadratically Convergent Method for Solving Simultaneous Nonlinear Equations.* PhD thesis, Purdue University, 1966.

[5] Kenneth M. Brown. Algorithm 316. Solution of simultaneous nonlinear equations. *Communications ACM*, 10:728—729, 1967.

[6] Kenneth M. Brown. A quadratically convergent Newton-like method based on Gaussian elimination. *SIAM J. Numerical Analysis*, 6(4):560—569, December 1969.

[7] Kenneth M. Brown and J. E. Dennis, Jr. On the second order convergence of Brown's derivative-free method for solving simultaneous nonlinear equations. Research Report 71-7, Yale University, Department of Computer Science, New Haven, Connecticut, June 1971.

[8] M. M. El-Alem. *A Global Convergence Theory for a Class of Trust Region Algorithms for Constrained Optimization.* PhD thesis, Rice University, Department of Mathematical Sciences, 1988.

[9] M. M. El-Alem. A global convergence theory for the Celis-Dennis-Tapia trust region algorithm for constrained optimization. *SIAM J. Numerical Analysis*, 28:266—290, 1991.

[10] M. M. El-Alem. Robust trust-region algorithm with non-monotonic penalty parameter scheme for constrained optimization. Technical Report TR92-30, Rice

University, Department of Computational and Applied Mathematics, Houston, Texas 77251, September 1992.

[11] R. Fletcher. *Practical Methods of Optimization.* Wiley, 1989.

[12] David M. Gay. *Brown's Method and Some Generalizations, with Applications to Minimization Problems.* PhD thesis, Cornell University, 1975.

[13] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1989.

[14] J. E. Dennis, Jr. Non-linear least squares and equations. In *The State-of-the-Art in Numerical Analysis*, 1976.

[15] J. E. Dennis, Jr. and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[16] John E. Dennis, Mahmoud El-Alem, and Maria Cristina Maciel. A global convergence theory for general trust-region-based algorithms for equality constrained optimization. Technical Report TR92-28, Rice University, Department of Computational and Applied Mathematics, Houston, Texas 77251, September 1992.

[17] L. S. Lasdon and A. D. Waren. Generalized reduced gradient software for linearly and nonlinearly constrained problems. In H. J. Greenberg, editor, *Design and Implemetation of Optimization Software*, pages 335—362. Sijthoff and Noordhoff, Netherlands, 1978.

[18] K. Levenberg. A method for the solution of certain problems in least squares. *Quart. Appl. Math.*, 2:164—168, 1944.

[19] M. J. D. Powell. Convergence properties of a class of minimization algorithms. In O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, editors, *Nonlinear Programming 2.* Academic Press, 1975.

[20] María Cristina Maciel. *A Global Convergence Theory for a General Class of Trust Region Algorithms for Equality Constrained Optimization.* PhD thesis, Rice University, Department of Computational and Applied Mathematics, 1993.

[21] D. Marquardt. An algorithm for least squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, 11:431—441, 1963.

[22] José Mario Martínez. Generalization of the methods of Brent and Brown for solving nonlinear simultaneous equations. *SIAM J. Numerical Analysis*, 1(3):434—448, June 1979.

[23] José Mario Martínez. Solving nonlinear simultaneous equations with a generalization of Brent's method. *BIT*, 20:501—510, 1980.

[24] J. J. Moré. The Levenberg-Marquardt algorithm: Implementation and theory. In G. A. Watson, editor, *Lecture Notes in Mathematics 630*, pages 105—116. Springer-Verlag, 1978.

[25] J. J. Moré. Recent developments in algorithms and software for trust region methods. In A. Bachem, M. Grötschel, and B. Korte, editors, *Mathematical Programming: The State of the Art.* Springer-Verlag, 1983.

[26] J. J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. Stat. Comput.*, 4(3):553—572, September 1983.

[27] J. J. Moré, Burton S. Garbow, and Kenneth E. Hillstrom. Testing unconstrained optimization software. *ACM Transactions on Mathematical Software*, 7(1):17—41, March 1981.

[28] J. Ortega and W. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, 1970.

[29] Philip E. Gill, Walter Murray, and Margaret H. Wright. *Practical Optimization.* Academic Press, 1981.

[30] Philip E. Gill, Walter Murray, Michael A. Saunders, G. W. Stewart, and Margaret H. Wright. Properties of a representation of a basis for the null space. *Mathematical Programming*, 33:172—186, 1985.

[31] Richard H. Byrd and Robert B. Schnabel. Continuity of the null space basis and constrained optimization. *Mathematical Programming*, 35:32—41, 1986.

[32] S. L. Campbell and C. P. Meyer, Jr. *Generalized Inverses of Linear Transformations.* Dover Publications, 1991.

[33] Gerald A. Shultz, Robert B. Schnabel, and Richard H. Byrd. A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties. *SIAM J. Numerical Analysis*, 22(1):47—67, February 1985.

[34] D. C. Sorensen. Newton's method with a model trust region modification. *SIAM J. Numerical Analysis*, 19(2):409—426, April 1982.

[35] Thomas F. Coleman and Danny C. Sorensen. A note on the computation of an orthonormal basis for the null space of a matrix. *Mathematical Programming*, 29:234—242, 1984.