最佳化方法與應用
MA5037-*

Chapter 4. Trust Region Methods

Introduction

Line search methods and trust-region methods both generate steps with the help of a quadratic model of the objective function, but they use this model in different ways. Line search methods use it to generate a search direction, and then focus their efforts on finding a suitable step length α along this direction. Trust-region methods define a region around the current iterate within which they trust the model to be an adequate representation of the objective function, and then choose the step to be the approximate minimizer of the model in this region. In effect, they choose the direction and length of the step simultaneously. If a step is not acceptable, they reduce the size of the region and find a new minimizer. In general, the direction of the step changes whenever the size of the trust region is altered.

Introduction

The size of the trust region is critical to the effectiveness of each step. If the region is too small, the algorithm misses an opportunity to take a substantial step that will move it much closer to the minimizer of the objective function. If too large, the minimizer of the model may be far from the minimizer of the objective function in the region, so we may have to reduce the size of the region and try again.

Introduction

In practical algorithms, we choose the size of the region according to the performance of the algorithm during previous iterations. If the model is consistently reliable, producing good steps and accurately predicting the behavior of the objective function along these steps, the size of the trust region may be increased to allow longer, more ambitious, steps to be taken. A failed step is an indication that our model is an inadequate representation of the objective function over the current trust region. After such a step, we reduce the size of the region and try again.

Introduction

Figure 1 (in the next slide) illustrates the trust-region approach on a function f of two variables in which the current point x_k and the minimizer x_* lie at opposite ends of a curved valley. The quadratic model function m_k, whose elliptical contours are shown as dashed lines, is constructed from function and derivative information at x_k and possibly also on information accumulated from previous iterations and steps. A line search method based on this model searches along the step to the minimizer of m_k (shown), but this direction will yield at most a small reduction in f, even if the optimal step length is used. The trust-region method steps to the minimizer of m_k within the dotted circle (shown), yielding a more significant reduction in f and better progress toward the solution.

Introduction

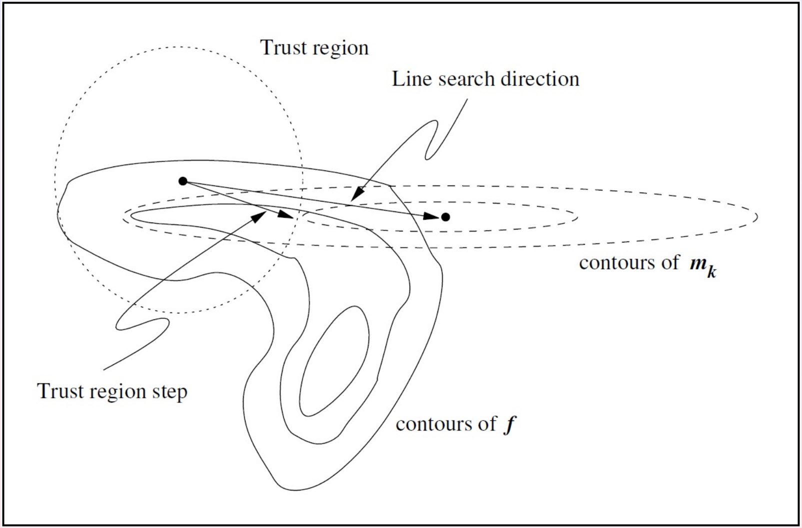

Figure 1: Trust-region and line search steps.

Introduction

In this chapter, we will assume that the model function m_k that is used at each iterate x_k is quadratic. Moreover, m_k is based on the Taylor-series expansion of f around x_k, which is

$$f(x_k + p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} (\nabla^2 f)(x_k + tp) p \,, \tag{1}$$

where $f_k = f(x_k)$ and $g_k = (\nabla f)(x_k)$, and t is some scalar in the interval $(0, 1)$. By using an approximation B_k to the Hessian in the second-order term, m_k is defined as follows:

$$m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \,, \tag{2}$$

where B_k is some symmetric matrix. The difference between $m_k(p)$ and $f(x_k + p)$ is $\mathcal{O}(\|p\|^2)$, which is small when $\|p\|$ is small.

Introduction

In this chapter, we will assume that the model function m_k that is used at each iterate x_k is quadratic. Moreover, m_k is based on the Taylor-series expansion of f around x_k, which is

$$f(x_k + p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} (\nabla^2 f)(x_k + tp) p \,, \tag{1}$$

where $f_k = f(x_k)$ and $g_k = (\nabla f)(x_k)$, and t is some scalar in the interval $(0, 1)$. By using an approximation B_k to the Hessian in the second-order term, m_k is defined as follows:

$$m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \,, \tag{2}$$

where B_k is some symmetric matrix. The difference between $m_k(p)$ and $f(x_k + p)$ is $\mathcal{O}(\|p\|^2)$, which is small when $\|p\|$ is small.

Introduction

When B_k is equal to the true Hessian $(\nabla^2 f)(x_k)$, the approximation error in the model function m_k is $\mathcal{O}(\|p\|^3)$, so this model is especially accurate when $\|p\|$ is small. This choice $B_k \equiv (\nabla^2 f)(x_k)$ leads to the trust-region Newton method, and will be discussed further in Section 4.4. In other sections of this chapter, we emphasize the generality of the trust-region approach by assuming little about B_k except **symmetry** and **uniform boundedness**.

Introduction

To obtain each step, we seek a solution of the sub-problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \Delta_k\,, \tag{3}$$

where $\Delta_k > 0$ is the **trust-region radius**. The trust-region approach requires us to solve a sequence of sub-problems (3) in which the objective function and constraint (which can be written as $p^{\mathrm{T}} p \leqslant \Delta_k^2$) are both quadratic. When B_k is positive definite and $\|B_k^{-1} g_k\| \leqslant \Delta_k$, the solution of (3) is easy to identify – it is simply the unconstrained minimum $p_k^B = -B_k^{-1} g_k$ of the quadratic $m_k(p)$. In this case, we call p_k^B the **full step**. The solution of (3) is not so obvious in other cases, but it can usually be found without too much computational expense. In any case, as described below, we need only an approximate solution to obtain convergence and good practical behavior.

Introduction

To obtain each step, we seek a solution of the sub-problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \Delta_k\,, \tag{3}$$

where $\Delta_k > 0$ is the **trust-region radius**. The trust-region approach requires us to solve a sequence of sub-problems (3) in which the objective function and constraint (which can be written as $p^{\mathrm{T}} p \leqslant \Delta_k^2$) are both quadratic. When B_k is positive definite and $\|B_k^{-1} g_k\| \leqslant \Delta_k$, the solution of (3) is easy to identify – it is simply the unconstrained minimum $p_k^B = -B_k^{-1} g_k$ of the quadratic $m_k(p)$. In this case, we call p_k^B the **full step**. The solution of (3) is not so obvious in other cases, but it can usually be found without too much computational expense. In any case, as described below, we need only an approximate solution to obtain convergence and good practical behavior.

Introduction

To obtain each step, we seek a solution of the sub-problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \Delta_k\,, \qquad (3)$$

where $\Delta_k > 0$ is the **trust-region radius**. The trust-region approach requires us to solve a sequence of sub-problems (3) in which the objective function and constraint (which can be written as $p^{\mathrm{T}} p \leqslant \Delta_k^2$) are both quadratic. When B_k is positive definite and $\|B_k^{-1} g_k\| \leqslant \Delta_k$, the solution of (3) is easy to identify – it is simply the unconstrained minimum $p_k^B = -B_k^{-1} g_k$ of the quadratic $m_k(p)$. In this case, we call p_k^B the **full step**. The solution of (3) is not so obvious in other cases, but it can usually be found without too much computational expense. In any case, as described below, we need only an approximate solution to obtain convergence and good practical behavior.

Introduction

- **Outline of the Trust-Region Approach**

One of the key ingredients in a trust-region algorithm is the strategy for choosing the trust-region radius Δ_k at each iteration. We base this choice on the agreement between the model function m_k and the objective function f at previous iterations. Given a step p_k we define the ratio

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}; \tag{4}$$

the numerator is called the **actual reduction**, and the denominator is the **predicted reduction** (that is, the reduction in f predicted by the model function).

Introduction

- **Outline of the Trust-Region Approach**

One of the key ingredients in a trust-region algorithm is the strategy for choosing the trust-region radius Δ_k at each iteration. We base this choice on the agreement between the model function m_k and the objective function f at previous iterations. Given a step p_k we define the ratio

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}\,; \tag{4}$$

the numerator is called the **actual reduction**, and the denominator is the **predicted reduction** (that is, the reduction in f predicted by the model function).

Introduction

Note that since the step p_k is obtained by minimizing the model m_k over a region that includes $p = 0$, the predicted reduction will always be non-negative. Hence, if ρ_k is negative, the new objective value $f(x_k + p_k)$ is greater than the current value $f(x_k)$, so the step must be rejected.

On the other hand, if ρ_k is close to 1, there is good agreement between the model m_k and the function f over this step, so it is safe to **expand** the trust region for the next iteration. If ρ_k is positive but significantly smaller than 1, we do not alter the trust region, but if it is close to zero or negative, we **shrink** the trust region by reducing Δ_k at the next iteration.

Introduction

Note that since the step p_k is obtained by minimizing the model m_k over a region that includes $p = 0$, the predicted reduction will always be non-negative. Hence, if ρ_k is negative, the new objective value $f(x_k + p_k)$ is greater than the current value $f(x_k)$, so the step must be rejected.

On the other hand, if ρ_k is close to 1, there is good agreement between the model m_k and the function f over this step, so it is safe to **expand** the trust region for the next iteration. If ρ_k is positive but significantly smaller than 1, we do not alter the trust region, but if it is close to zero or negative, we **shrink** the trust region by reducing Δ_k at the next iteration.

Introduction

Note that since the step p_k is obtained by minimizing the model m_k over a region that includes $p = 0$, the predicted reduction will always be non-negative. Hence, if ρ_k is negative, the new objective value $f(x_k + p_k)$ is greater than the current value $f(x_k)$, so the step must be rejected.

On the other hand, if ρ_k is close to 1, there is good agreement between the model m_k and the function f over this step, so it is safe to **expand** the trust region for the next iteration. If ρ_k is positive but significantly smaller than 1, we do not alter the trust region, but if it is close to zero or negative, we **shrink** the trust region by reducing Δ_k at the next iteration.

Introduction

Algorithm 4.1 (Trust Region).

Given $\widehat{\Delta} > 0$, $\Delta_0 \in (0, \widehat{\Delta})$, and $\eta \in [0, 1/4)$;

for $k = 0, 1, 2, \cdots$

Obtain p_k by (approximately) solving (3) & evaluate ρ_k from (4);

if $\rho_k < 1/4$

$\Delta_{k+1} = \frac{1}{4}\Delta_k$;

else

if $\rho_k > 3/4$ and $\|p_k\| = \Delta_k$

$\Delta_{k+1} = \min(2\Delta_k, \widehat{\Delta})$;

else

$\Delta_{k+1} = \Delta_k$;

if $\rho_k > \eta$

$x_{k+1} = x_k + p_k$;

else

$x_{k+1} = x_k$;

end (**for**)

Introduction

Here $\widehat{\Delta}$ is an overall bound on the step lengths. Note that the radius is increased only if $\|p_k\|$ actually reaches the boundary of the trust region. If the step stays strictly inside the region, we infer that the current value of Δ_k is not interfering with the progress of the algorithm, so we leave its value unchanged for the next iteration.

To turn Algorithm 4.1 into a practical algorithm, we need to focus on solving the trust-region sub-problem (3). In discussing this matter, we sometimes drop the subscript k and restate the problem as

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

A first step to characterizing exact solutions of (5) is given by the following theorem which shows that the solution p_* of (5) satisfies

$$(B+\lambda \mathrm{I})p_* = -g \tag{6a}$$

for some $\lambda \geqslant 0$.

Introduction

Here $\widehat{\Delta}$ is an overall bound on the step lengths. Note that the radius is increased only if $\|p_k\|$ actually reaches the boundary of the trust region. If the step stays strictly inside the region, we infer that the current value of Δ_k is not interfering with the progress of the algorithm, so we leave its value unchanged for the next iteration.

To turn Algorithm 4.1 into a practical algorithm, we need to focus on solving the trust-region sub-problem (3). In discussing this matter, we sometimes drop the subscript k and restate the problem as

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

A first step to characterizing exact solutions of (5) is given by the following theorem which shows that the solution p_* of (5) satisfies

$$(B + \lambda \mathrm{I})p_* = -g \tag{6a}$$

for some $\lambda \geqslant 0$.

Introduction

Here $\widehat{\Delta}$ is an overall bound on the step lengths. Note that the radius is increased only if $\|p_k\|$ actually reaches the boundary of the trust region. If the step stays strictly inside the region, we infer that the current value of Δ_k is not interfering with the progress of the algorithm, so we leave its value unchanged for the next iteration.

To turn Algorithm 4.1 into a practical algorithm, we need to focus on solving the trust-region sub-problem (3). In discussing this matter, we sometimes drop the subscript k and restate the problem as

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

A first step to characterizing exact solutions of (5) is given by the following theorem which shows that the solution p_* of (5) satisfies

$$(B + \lambda \mathrm{I}) p_* = -g \tag{6a}$$

for some $\lambda \geqslant 0$.

Introduction

Theorem

The vector p_* is a **global** *solution of the trust-region problem*

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

if and only if p_* is feasible and there is a scalar $\lambda \geqslant 0$ such that the following conditions are satisfied:

$$(B+\lambda\mathrm{I})p_* = -g\,, \tag{6a}$$

$$\lambda(\Delta - \|p_*\|) = 0\,, \tag{6b}$$

$$(B+\lambda\mathrm{I}) \text{ is positive semi-definite.} \tag{6c}$$

Remark: The non-negativity of λ and (6b) are part of the **KKT conditions** for the constrained optimization problem (5).

We will delay the proof of this result until Section 4.3.

Introduction

Theorem

The vector p_* is a **global** *solution of the trust-region problem*

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

if and only if p_* is feasible and there is a scalar $\lambda \geqslant 0$ such that the following conditions are satisfied:

$$(B + \lambda\mathrm{I})p_* = -g\,, \tag{6a}$$

$$\lambda(\Delta - \|p_*\|) = 0\,, \tag{6b}$$

$$(B + \lambda\mathrm{I}) \text{ is positive semi-definite.} \tag{6c}$$

Remark: The non-negativity of λ and (6b) are part of the **KKT conditions** for the constrained optimization problem (5).

We will delay the proof of this result until Section 4.3.

Introduction

Theorem

The vector p_* *is a* **global** *solution of the trust-region problem*

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

if and only if p_* *is feasible and there is a scalar* $\lambda \geqslant 0$ *such that the following conditions are satisfied:*

$$(B + \lambda\mathrm{I})p_* = -g\,, \tag{6a}$$

$$\lambda(\Delta - \|p_*\|) = 0\,, \tag{6b}$$

$$(B + \lambda\mathrm{I}) \text{ is positive semi-definite.} \tag{6c}$$

Remark: The non-negativity of λ and (6b) are part of the **KKT conditions** for the constrained optimization problem (5).

We will delay the proof of this result until Section 4.3.

Introduction

The condition (6b) is a complementarity condition that states that at least one of the non-negative quantities λ and $(\Delta - \|p_*\|)$ must be zero. Hence, when the solution lies strictly inside the trust region (as it does when $\Delta = \Delta_1$ in Figure 2 in the next slide), we must have $\lambda = 0$ and so $Bp_* = -g$ with B positive semi-definite, from (6a) and (6c), respectively. In the other cases $\Delta = \Delta_2$ and $\Delta = \Delta_3$, we have $\|p_*\| = \Delta$, and so λ is allowed to take a positive value. Note from (6a) that

$$\lambda p_* = -Bp_* - g = -\nabla m(p_*) \,.$$

Thus, when $\lambda > 0$, the solution p_* is collinear with the negative gradient of m and normal to its contours. These properties can be seen in Figure 2.

Introduction

The condition (6b) is a complementarity condition that states that at least one of the non-negative quantities λ and $(\Delta - \|p_*\|)$ must be zero. Hence, when the solution lies strictly inside the trust region (as it does when $\Delta = \Delta_1$ in Figure 2 in the next slide), we must have $\lambda = 0$ and so $Bp_* = -g$ with B positive semi-definite, from (6a) and (6c), respectively. In the other cases $\Delta = \Delta_2$ and $\Delta = \Delta_3$, we have $\|p_*\| = \Delta$, and so λ is allowed to take a positive value. Note from (6a) that

$$\lambda p_* = -Bp_* - g = -\nabla m(p_*) .$$

Thus, when $\lambda > 0$, the solution p_* is collinear with the negative gradient of m and normal to its contours. These properties can be seen in Figure 2.

Introduction

The condition (6b) is a complementarity condition that states that at least one of the non-negative quantities λ and $(\Delta - \|p_*\|)$ must be zero. Hence, when the solution lies strictly inside the trust region (as it does when $\Delta = \Delta_1$ in Figure 2 in the next slide), we must have $\lambda = 0$ and so $Bp_* = -g$ with B positive semi-definite, from (6a) and (6c), respectively. In the other cases $\Delta = \Delta_2$ and $\Delta = \Delta_3$, we have $\|p_*\| = \Delta$, and so λ is allowed to take a positive value. Note from (6a) that

$$\lambda p_* = -Bp_* - g = -\nabla m(p_*)\,.$$

Thus, when $\lambda > 0$, the solution p_* is collinear with the negative gradient of m and normal to its contours. These properties can be seen in Figure 2.

Introduction

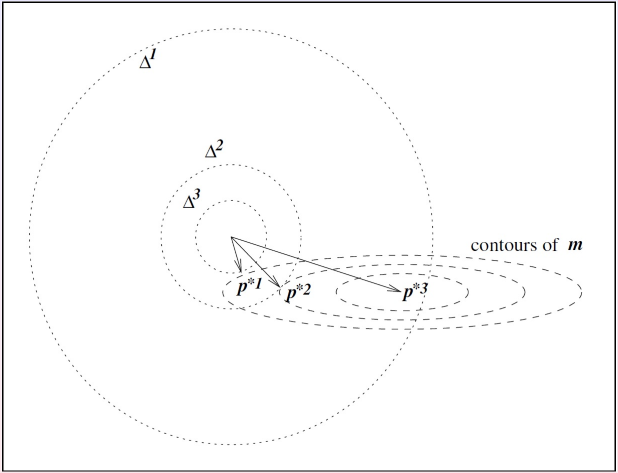

Figure 2: Solution of trust-region sub-problem for different radii Δ^1, Δ^2, Δ^3.

Introduction

In Section 4.1, we describe two strategies for finding approximate solutions of the sub-problem (3), which achieve at least as much reduction in m_k as the reduction achieved by the so-called Cauchy point. This point is simply the minimizer of m_k along the steepest descent direction $-g_k$ subject to the trust-region bound. The first approximate strategy is the dogleg method, which is appropriate when the model Hessian B_k is positive definite. The second strategy, known as two-dimensional subspace minimization, can be applied when B_k is indefinite, though it requires an estimate of the most negative eigenvalue of this matrix. A third strategy, described in Section 7.1, uses an approach based on the conjugate gradient method to minimize m_k, and can therefore be applied when B is large and sparse.

Introduction

In Section 4.1, we describe two strategies for finding approximate solutions of the sub-problem (3), which achieve at least as much reduction in m_k as the reduction achieved by the so-called Cauchy point. This point is simply the minimizer of m_k along the steepest descent direction $-g_k$ subject to the trust-region bound. The first approximate strategy is the dogleg method, which is appropriate when the model Hessian B_k is positive definite. The second strategy, known as two-dimensional subspace minimization, can be applied when B_k is indefinite, though it requires an estimate of the most negative eigenvalue of this matrix. A third strategy, described in Section 7.1, uses an approach based on the conjugate gradient method to minimize m_k, and can therefore be applied when B is large and sparse.

Introduction

In Section 4.1, we describe two strategies for finding approximate solutions of the sub-problem (3), which achieve at least as much reduction in m_k as the reduction achieved by the so-called Cauchy point. This point is simply the minimizer of m_k along the steepest descent direction $-g_k$ subject to the trust-region bound. The first approximate strategy is the dogleg method, which is appropriate when the model Hessian B_k is positive definite. The second strategy, known as two-dimensional subspace minimization, can be applied when B_k is indefinite, though it requires an estimate of the most negative eigenvalue of this matrix. A third strategy, described in Section 7.1, uses an approach based on the conjugate gradient method to minimize m_k, and can therefore be applied when B is large and sparse.

Introduction

Section 4.3 is devoted to a strategy in which an iterative method is used to identify the value of λ for which

$$(B + \lambda \mathrm{I})p_* = -g \tag{6a}$$

is satisfied by the solution of the sub-problem. We prove global convergence results in Section 4.2. Section 4.4 discusses the trust-region Newton method, in which the Hessian B_k of the model function is equal to the Hessian $(\nabla^2 f)(x_k)$ of the objective function. The key result of this section is that, when the trust-region Newton algorithm converges to a point x_* satisfying second-order sufficient conditions, it converges superlinearly.

Introduction

Section 4.3 is devoted to a strategy in which an iterative method is used to identify the value of λ for which

$$(B + \lambda \mathrm{I})p_* = -g \tag{6a}$$

is satisfied by the solution of the sub-problem. We prove global convergence results in Section 4.2. Section 4.4 discusses the trust-region Newton method, in which the Hessian B_k of the model function is equal to the Hessian $(\nabla^2 f)(x_k)$ of the objective function. The key result of this section is that, when the trust-region Newton algorithm converges to a point x_* satisfying second-order sufficient conditions, it converges superlinearly.

§4.1 Algorithms Based on the Cauchy Point

- **The Cauchy Point**

As we saw in Chapter 3, line search methods can be globally convergent even when the optimal step length is not used at each iteration. In fact, the step length α_k need only satisfy fairly loose criteria (such as Wolfe or Goldstein conditions). A similar situation applies in trust-region methods. Although in principle we seek the optimal solution of the sub-problem (3), it is enough for purposes of global convergence to find an approximate solution p_k that lies within the trust region and gives a sufficient reduction in the model. The sufficient reduction can be quantified in terms of the Cauchy point, which we denote by p_k^C and define in terms of the following simple procedure.

§4.1 Algorithms Based on the Cauchy Point

- **The Cauchy Point**

As we saw in Chapter 3, line search methods can be globally convergent even when the optimal step length is not used at each iteration. In fact, the step length α_k need only satisfy fairly loose criteria (such as Wolfe or Goldstein conditions). A similar situation applies in trust-region methods. Although in principle we seek the optimal solution of the sub-problem (3), it is enough for purposes of global convergence to find an approximate solution p_k that lies within the trust region and gives a sufficient reduction in the model. The sufficient reduction can be quantified in terms of the Cauchy point, which we denote by p_k^c and define in terms of the following simple procedure.

§4.1 Algorithms Based on the Cauchy Point

Algorithm 4.2 (Cauchy Point Calculation).

Find the vector p_k^s that solves a linear version of (3); that is,

$$p_k^s = \arg\min_{p\in\mathbb{R}^n} \left(f_k + g_k^{\mathrm{T}} p\right) \qquad \text{s.t.} \quad \|p\| \leqslant \Delta_k\,; \tag{7}$$

Calculate the scalar $\tau_k > 0$ that minimizes $m_k(p_k^s)$ subject to satisfying the trust-region bound; that is,

$$\tau_k = \arg\min_{\tau\geqslant 0} m_k(\tau p_k^s) \qquad \text{s.t.} \quad \|\tau p_k^s\| \leqslant \Delta_k\,; \tag{8}$$

Set $p_k^c = \tau_k p_k^s$.

Remark: Since p_k^s satisfies $\|p_k^s\| = \Delta_k$, the optimization problem (8) is indeed identical to

$$\tau_k = \arg\min_{\tau\geqslant 0} m_k(\tau p_k^s) \qquad \text{s.t.} \quad |\tau| \leqslant 1\,. \tag{8'}$$

§4.1 Algorithms Based on the Cauchy Point

Algorithm 4.2 (Cauchy Point Calculation).

Find the vector p_k^s that solves a linear version of (3); that is,

$$p_k^s = \arg\min_{p\in\mathbb{R}^n} \left(f_k + g_k^{\mathrm{T}} p\right) \qquad \text{s.t.} \quad \|p\| \leqslant \Delta_k\,; \tag{7}$$

Calculate the scalar $\tau_k > 0$ that minimizes $m_k(p_k^s)$ subject to satisfying the trust-region bound; that is,

$$\tau_k = \arg\min_{\tau\geqslant 0} m_k(\tau p_k^s) \qquad \text{s.t.} \quad \|\tau p_k^s\| \leqslant \Delta_k\,; \tag{8}$$

Set $p_k^c = \tau_k p_k^s$.

Remark: Since p_k^s satisfies $\|p_k^s\| = \Delta_k$, the optimization problem (8) is indeed identical to

$$\tau_k = \arg\min_{\tau\geqslant 0} m_k(\tau p_k^s) \qquad \text{s.t.} \quad |\tau| \leqslant 1\,. \tag{8'}$$

§4.1 Algorithms Based on the Cauchy Point

It is easy to write down a closed-form definition of the Cauchy point. For a start, the solution of (7) is simply

$$p_k^s = -\frac{\Delta_k}{\|g_k\|} g_k \,.$$

To obtain τ_k explicitly, we consider the cases of $g_k^{\mathrm{T}} B_k g_k \leqslant 0$ and $g_k^{\mathrm{T}} B_k g_k > 0$ separately.

1. For the former case, the function $m_k(\tau p_k^s)$ decreases monotonically with τ whenever $g_k \neq 0$, so τ_k is simply the largest value that satisfies the trust-region bound, namely, $\tau_k = 1$.
2. For the case $g_k^{\mathrm{T}} B_k g_k > 0$, $m_k(\tau p_k^s)$ is a convex quadratic in τ, so τ_k is either the unconstrained minimizer of this quadratic, $\|g_k\|^3/(\Delta_k g_k^{\mathrm{T}} B_k g_k)$, or the boundary value 1, whichever comes first.

§4.1 Algorithms Based on the Cauchy Point

It is easy to write down a closed-form definition of the Cauchy point. For a start, the solution of (7) is simply

$$p_k^s = -\frac{\Delta_k}{\|g_k\|} g_k \,.$$

To obtain τ_k explicitly, we consider the cases of $g_k^{\mathrm{T}} B_k g_k \leqslant 0$ and $g_k^{\mathrm{T}} B_k g_k > 0$ separately.

1. For the former case, the function $m_k(\tau p_k^s)$ decreases monotonically with τ whenever $g_k \neq 0$, so τ_k is simply the largest value that satisfies the trust-region bound, namely, $\tau_k = 1$.
2. For the case $g_k^{\mathrm{T}} B_k g_k > 0$, $m_k(\tau p_k^s)$ is a convex quadratic in τ, so τ_k is either the unconstrained minimizer of this quadratic, $\|g_k\|^3/(\Delta_k g_k^{\mathrm{T}} B_k g_k)$, or the boundary value 1, whichever comes first.

§4.1 Algorithms Based on the Cauchy Point

It is easy to write down a closed-form definition of the Cauchy point. For a start, the solution of (7) is simply

$$p_k^s = -\frac{\Delta_k}{\|g_k\|} g_k .$$

To obtain τ_k explicitly, we consider the cases of $g_k^T B_k g_k \leqslant 0$ and $g_k^T B_k g_k > 0$ separately.

1. For the former case, the function $m_k(\tau p_k^s)$ decreases monotonically with τ whenever $g_k \neq 0$, so τ_k is simply the largest value that satisfies the trust-region bound, namely, $\tau_k = 1$.
2. For the case $g_k^T B_k g_k > 0$, $m_k(\tau p_k^s)$ is a convex quadratic in τ, so τ_k is either the unconstrained minimizer of this quadratic, $\|g_k\|^3/(\Delta_k g_k^T B_k g_k)$, or the boundary value 1, whichever comes first.

§4.1 Algorithms Based on the Cauchy Point

In summary, we have

$$p_k^C = -\tau_k \frac{\Delta_k}{\|g_k\|} g_k \,, \tag{9}$$

where

$$\tau_k = \begin{cases} 1 & \text{if } g_k^{\mathrm{T}} B_k g_k \leqslant 0 \,, \\ \min\big(\|g_k\|^3/(\Delta_k g_k^{\mathrm{T}} B_k g_k), 1\big) & \text{otherwise} \,. \end{cases} \tag{10}$$

Figure 3 (in the next slide) illustrates the Cauchy point for a sub-problem in which B_k is positive definite. In this example, p_k^C lies strictly inside the trust region. The Cauchy step p_k^C is inexpensive to calculate – no matrix factorizations are required – and is of crucial importance in deciding if an approximate solution of the trust-region sub-problem is acceptable. Specifically, a trust-region method will be globally convergent if its steps p_k give a reduction in the model m_k that is at least some fixed positive multiple of the decrease attained by the Cauchy step.

§4.1 Algorithms Based on the Cauchy Point

In summary, we have

$$p_k^C = -\tau_k \frac{\Delta_k}{\|g_k\|} g_k \,, \tag{9}$$

where

$$\tau_k = \begin{cases} 1 & \text{if } g_k^{\mathrm{T}} B_k g_k \leqslant 0\,, \\ \min\left(\|g_k\|^3/(\Delta_k g_k^{\mathrm{T}} B_k g_k), 1\right) & \text{otherwise}\,. \end{cases} \tag{10}$$

Figure 3 (in the next slide) illustrates the Cauchy point for a sub-problem in which B_k is positive definite. In this example, p_k^C lies strictly inside the trust region. The Cauchy step p_k^C is inexpensive to calculate – no matrix factorizations are required – and is of crucial importance in deciding if an approximate solution of the trust-region sub-problem is acceptable. Specifically, a trust-region method will be globally convergent if its steps p_k give a reduction in the model m_k that is at least some fixed positive multiple of the decrease attained by the Cauchy step.

§4.1 Algorithms Based on the Cauchy Point

In summary, we have

$$p_k^C = -\tau_k \frac{\Delta_k}{\|g_k\|} g_k \,, \tag{9}$$

where

$$\tau_k = \begin{cases} 1 & \text{if } g_k^{\mathrm{T}} B_k g_k \leqslant 0 \,, \\ \min\big(\|g_k\|^3/(\Delta_k g_k^{\mathrm{T}} B_k g_k), 1\big) & \text{otherwise} \,. \end{cases} \tag{10}$$

Figure 3 (in the next slide) illustrates the Cauchy point for a sub-problem in which B_k is positive definite. In this example, p_k^C lies strictly inside the trust region. The Cauchy step p_k^C is inexpensive to calculate – no matrix factorizations are required – and is of crucial importance in deciding if an approximate solution of the trust-region sub-problem is acceptable. Specifically, a trust-region method will be globally convergent if its steps p_k give a reduction in the model m_k that is at least some fixed positive multiple of the decrease attained by the Cauchy step.

§4.1 Algorithms Based on the Cauchy Point

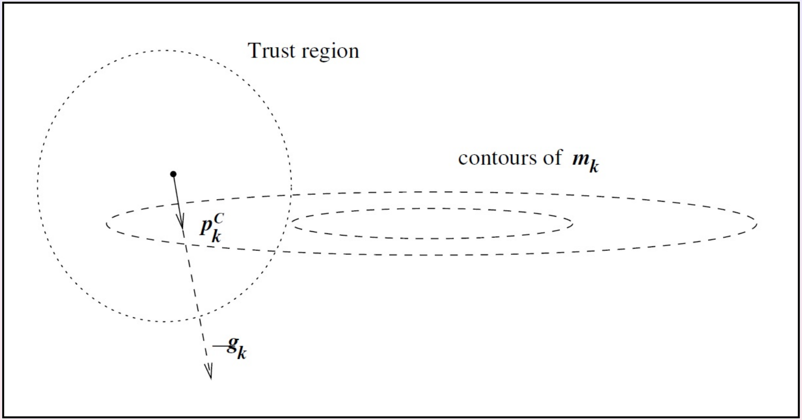

Figure 3: The Cauchy point.

§4.1 Algorithms Based on the Cauchy Point

- **Improving on the Cauchy Point**

Since the Cauchy point p_k^c provides sufficient reduction in the model function m_k to yield global convergence, and since the cost of calculating it is so small, why should we look any further for a better approximate solution of (3)? The reason is that by always taking the Cauchy point as our step, we are simply implementing the steepest descent method with a particular choice of step length. As we have seen in Chapter 3, steepest descent performs poorly even if an optimal step length is used at each iteration.

§4.1 Algorithms Based on the Cauchy Point

- **Improving on the Cauchy Point**

Since the Cauchy point p_k^c provides sufficient reduction in the model function m_k to yield global convergence, and since the cost of calculating it is so small, why should we look any further for a better approximate solution of (3)? The reason is that by always taking the Cauchy point as our step, we are simply implementing the steepest descent method with a particular choice of step length. As we have seen in Chapter 3, steepest descent performs poorly even if an optimal step length is used at each iteration.

§4.1 Algorithms Based on the Cauchy Point

The Cauchy point does not depend very strongly on the matrix B_k, which is used only in the calculation of the step length. Rapid convergence can be expected only if B_k plays a role in determining the direction of the step as well as its length, and if B_k contains valid curvature information about the function. A number of trust-region algorithms compute the Cauchy point and then try to improve on it. The improvement strategy is often designed so that the full step $p_k^B = -B_k^{-1} g_k$ is chosen whenever B_k is positive definite and $\|p_k^B\| \leqslant \Delta_k$. When B_k is the exact Hessian $(\nabla^2 f)(x_k)$ or a quasi-Newton approximation, this strategy can be expected to yield superlinear convergence.

§4.1 Algorithms Based on the Cauchy Point

We now consider three methods for finding approximate solutions to (3) that have the features just described. Throughout this section we will be focusing on the internal workings of a single iteration, so we simplify the notation by dropping the subscript "k" from the quantities Δ_k, p_k, m_k, and g_k and refer to the formulation

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

of the sub-problem. In this section, we denote the solution of (5) by $p_*(\Delta)$, to emphasize the dependence on Δ.

§4.1 Algorithms Based on the Cauchy Point

- **The Dogleg Method**

The first approach we discuss goes by the descriptive title of the dogleg method. It can be used when B is positive definite. To motivate this method, we start by examining the effect of the trust-region radius Δ on the solution $p_*(\Delta)$ of the sub-problem

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

When B is positive definite, we have already noted that the unconstrained minimizer of m is $p^B = -B^{-1}g$.

§4.1 Algorithms Based on the Cauchy Point

When B is positive definite, we have already noted that the unconstrained minimizer of m is $p^B = -B^{-1}g$.

1. When this point is feasible for (5), it is obviously a solution, so we have
$$p_*(\Delta) = p^B\,, \quad \text{when } \Delta \geqslant \|p^B\|. \tag{11}$$
2. When Δ is small relative to p^B, the restriction $\|p\| \leqslant \Delta$ ensures that the quadratic term in m has little effect on the solution of (5). For such Δ, we can get an approximation to $p(\Delta)$ by simply omitting the quadratic term from (5) and writing
$$p_*(\Delta) \approx -\Delta\frac{g}{\|g\|}\,, \quad \text{when } \Delta \text{ is small.} \tag{12}$$
3. For intermediate values of Δ, the solution $p_*(\Delta)$ typically follows a curved trajectory like the one in Figure 4.

§4.1 Algorithms Based on the Cauchy Point

When B is positive definite, we have already noted that the unconstrained minimizer of m is $p^B = -B^{-1}g$.

1. When this point is feasible for (5), it is obviously a solution, so we have
$$p_*(\Delta) = p^B, \quad \text{when } \Delta \geqslant \|p^B\|. \tag{11}$$
2. When Δ is small relative to p^B, the restriction $\|p\| \leqslant \Delta$ ensures that the quadratic term in m has little effect on the solution of (5). For such Δ, we can get an approximation to $p(\Delta)$ by simply omitting the quadratic term from (5) and writing
$$p_*(\Delta) \approx -\Delta \frac{g}{\|g\|}, \quad \text{when } \Delta \text{ is small.} \tag{12}$$
3. For intermediate values of Δ, the solution $p_*(\Delta)$ typically follows a curved trajectory like the one in Figure 4.

§4.1 Algorithms Based on the Cauchy Point

When B is positive definite, we have already noted that the unconstrained minimizer of m is $p^B = -B^{-1}g$.

1. When this point is feasible for (5), it is obviously a solution, so we have
$$p_*(\Delta) = p^B\,, \quad \text{when } \Delta \geqslant \|p^B\|. \tag{11}$$

2. When Δ is small relative to p^B, the restriction $\|p\| \leqslant \Delta$ ensures that the quadratic term in m has little effect on the solution of (5). For such Δ, we can get an approximation to $p(\Delta)$ by simply omitting the quadratic term from (5) and writing
$$p_*(\Delta) \approx -\Delta \frac{g}{\|g\|}\,, \quad \text{when } \Delta \text{ is small.} \tag{12}$$

3. For intermediate values of Δ, the solution $p_*(\Delta)$ typically follows a curved trajectory like the one in Figure 4.

§4.1 Algorithms Based on the Cauchy Point

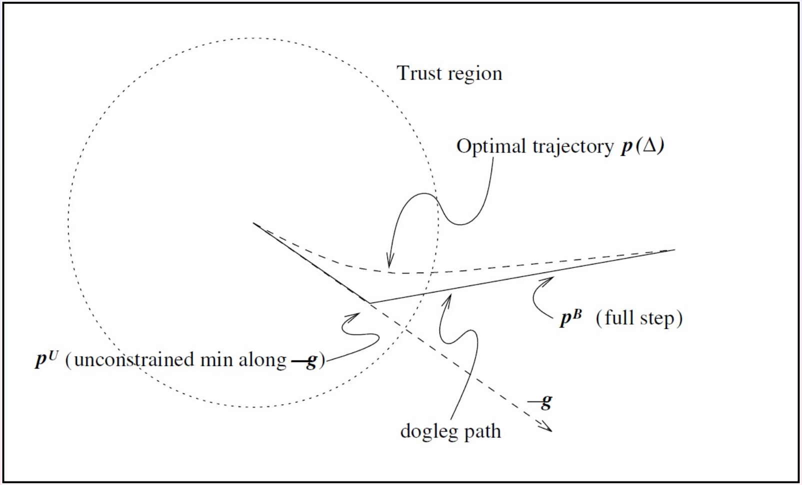

Figure 4: Exact trajectory and dogleg approximation.

§4.1 Algorithms Based on the Cauchy Point

The dogleg method finds an approximate solution by replacing the curved trajectory for $p_*(\Delta)$ with a path consisting of two line segments. The first line segment runs from the origin to the minimizer of m along the steepest descent direction, which is

$$p^U \equiv -\frac{g^{\mathrm{T}} g}{g^{\mathrm{T}} B g} g, \tag{13}$$

while the second line segment runs from p^U to p^B (see Figure 4). Formally, we denote this trajectory by $\tilde{p}(\tau)$ for $\tau \in [0, 2]$, where

$$\tilde{p}(\tau) = \begin{cases} \tau p^U & \text{if } 0 \leqslant \tau \leqslant 1\,, \\ p^U + (\tau - 1)(p^B - p^U) & \text{if } 1 \leqslant \tau \leqslant 2\,. \end{cases} \tag{14}$$

The dogleg method chooses p to minimize the model m along this path, subject to the trust-region bound. The following lemma shows that the minimum along the dogleg path can be found easily.

§4.1 Algorithms Based on the Cauchy Point

The dogleg method finds an approximate solution by replacing the curved trajectory for $p_*(\Delta)$ with a path consisting of two line segments. The first line segment runs from the origin to the minimizer of m along the steepest descent direction, which is

$$p^U \equiv -\frac{g^{\mathrm{T}} g}{g^{\mathrm{T}} B g} g, \tag{13}$$

while the second line segment runs from p^U to p^B (see Figure 4). Formally, we denote this trajectory by $\tilde{p}(\tau)$ for $\tau \in [0, 2]$, where

$$\tilde{p}(\tau) = \begin{cases} \tau p^U & \text{if } 0 \leqslant \tau \leqslant 1, \\ p^U + (\tau - 1)(p^B - p^U) & \text{if } 1 \leqslant \tau \leqslant 2. \end{cases} \tag{14}$$

The dogleg method chooses p to minimize the model m along this path, subject to the trust-region bound. The following lemma shows that the minimum along the dogleg path can be found easily.

§4.1 Algorithms Based on the Cauchy Point

Lemma

Let B be positive definite. Then

1. $\|\widetilde{p}(\tau)\|$ *is an increasing function of* τ, *and*
2. $m(\widetilde{p}(\tau))$ *is a decreasing function of* τ.

Proof.

It is "easy to see" that ① and ② both hold for $\tau \in [0,1]$, so we restrict our attention to the case of $\tau \in [1,2]$.

For ①, define $h(\alpha)$ by

$$h(\alpha) = \frac{1}{2}\|\widetilde{p}(1+\alpha)\|^2 = \frac{1}{2}\|p^U + \alpha(p^B - p^U)\|^2$$
$$= \frac{1}{2}\|p^U\|^2 + \alpha(p^U)^{\mathrm{T}}(p^B - p^U) + \frac{1}{2}\alpha^2\|p^B - p^U\|^2 .$$

Our result is proved if we can show that $h'(\alpha) \geqslant 0$ for $\alpha \in (0,1)$. □

§4.1 Algorithms Based on the Cauchy Point

Lemma

Let B be positive definite. Then

1. $\|\widetilde{p}(\tau)\|$ *is an increasing function of* τ, *and*
2. $m(\widetilde{p}(\tau))$ *is a decreasing function of* τ.

Proof.

It is "easy to see" that ① and ② both hold for $\tau \in [0,1]$, so we restrict our attention to the case of $\tau \in [1,2]$.

For ①, define $h(\alpha)$ by

$$\begin{aligned} h(\alpha) &= \frac{1}{2}\|\widetilde{p}(1+\alpha)\|^2 = \frac{1}{2}\|p^U + \alpha(p^B - p^U)\|^2 \\ &= \frac{1}{2}\|p^U\|^2 + \alpha(p^U)^{\mathrm{T}}(p^B - p^U) + \frac{1}{2}\alpha^2\|p^B - p^U\|^2 . \end{aligned}$$

Our result is proved if we can show that $h'(\alpha) \geqslant 0$ for $\alpha \in (0,1)$. □

§4.1 Algorithms Based on the Cauchy Point

Lemma

Let B be positive definite. Then

1. $\|\widetilde{p}(\tau)\|$ *is an increasing function of* τ*, and*
2. $m(\widetilde{p}(\tau))$ *is a decreasing function of* τ*.*

Proof.

It is "easy to see" that ① and ② both hold for $\tau \in [0,1]$, so we restrict our attention to the case of $\tau \in [1,2]$.

For ①, define $h(\alpha)$ by

$$\begin{aligned} h(\alpha) &= \frac{1}{2}\|\widetilde{p}(1+\alpha)\|^2 = \frac{1}{2}\|p^U + \alpha(p^B - p^U)\|^2 \\ &= \frac{1}{2}\|p^U\|^2 + \alpha(p^U)^{\mathrm{T}}(p^B - p^U) + \frac{1}{2}\alpha^2\|p^B - p^U\|^2 . \end{aligned}$$

Our result is proved if we can show that $h'(\alpha) \geqslant 0$ for $\alpha \in (0,1)$. □

§4.1 Algorithms Based on the Cauchy Point

Proof (cont'd).

Now we compute $h'(\alpha)$ and obtain

$$\begin{aligned} h'(\alpha) &= -(p^U)^{\mathrm{T}}(p^U - p^B) + \alpha\|p^U - p^B\|^2 \geqslant -(p^U)^{\mathrm{T}}(p^U - p^B) \\ &= \frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg} g^{\mathrm{T}}\Big(-\frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg}g + B^{-1}g\Big) \\ &= g^{\mathrm{T}}g \cdot \frac{g^{\mathrm{T}}B^{-1}g}{g^{\mathrm{T}}Bg}\Big[1 - \frac{(g^{\mathrm{T}}g)^2}{(g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g)}\Big]. \end{aligned}$$

By the fact that B is positive definite,

$$\|g^{\mathrm{T}}\sqrt{B}\|^2 = g^{\mathrm{T}}Bg \quad \text{and} \quad \|\sqrt{B^{-1}}g\|^2 = g^{\mathrm{T}}B^{-1}g$$

so that the Cauchy-Schwarz inequality shows that

$$(g^{\mathrm{T}}g)^2 = (g^{\mathrm{T}}\sqrt{B}\sqrt{B^{-1}}g)^2 \leqslant \|g^{\mathrm{T}}\sqrt{B}\|^2\|g^{\mathrm{T}}\sqrt{B^{-1}}\|^2$$

Therefore, $h'(\alpha) \geqslant 0$. □

§4.1 Algorithms Based on the Cauchy Point

Proof (cont'd).

Now we compute $h'(\alpha)$ and obtain

$$\begin{aligned} h'(\alpha) &= -(p^U)^{\mathrm{T}}(p^U - p^B) + \alpha\|p^U - p^B\|^2 \geqslant -(p^U)^{\mathrm{T}}(p^U - p^B) \\ &= \frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg}g^{\mathrm{T}}\Big(-\frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg}g + B^{-1}g\Big) \\ &= g^{\mathrm{T}}g\cdot\frac{g^{\mathrm{T}}B^{-1}g}{g^{\mathrm{T}}Bg}\Big[1 - \frac{(g^{\mathrm{T}}g)^2}{(g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g)}\Big]. \end{aligned}$$

By the fact that B is positive definite,

$$\|g^{\mathrm{T}}\sqrt{B}\|^2 = g^{\mathrm{T}}Bg \quad \text{and} \quad \|\sqrt{B^{-1}}g\|^2 = g^{\mathrm{T}}B^{-1}g$$

so that the Cauchy-Schwarz inequality shows that

$$(g^{\mathrm{T}}g)^2 = (g^{\mathrm{T}}\sqrt{B}\sqrt{B^{-1}}g)^2 \leqslant \|g^{\mathrm{T}}\sqrt{B}\|^2\|g^{\mathrm{T}}\sqrt{B^{-1}}\|^2$$

Therefore, $h'(\alpha) \geqslant 0$. □

§4.1 Algorithms Based on the Cauchy Point

Proof (cont'd).

Now we compute $h'(\alpha)$ and obtain

$$\begin{aligned} h'(\alpha) &= -(p^U)^{\mathrm{T}}(p^U - p^B) + \alpha\|p^U - p^B\|^2 \geqslant -(p^U)^{\mathrm{T}}(p^U - p^B) \\ &= \frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg} g^{\mathrm{T}}\Big(-\frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg}g + B^{-1}g\Big) \\ &= g^{\mathrm{T}}g \cdot \frac{g^{\mathrm{T}}B^{-1}g}{g^{\mathrm{T}}Bg}\Big[1 - \frac{(g^{\mathrm{T}}g)^2}{(g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g)}\Big]. \end{aligned}$$

By the fact that B is positive definite,

$$\|g^{\mathrm{T}}\sqrt{B}\|^2 = g^{\mathrm{T}}Bg \quad \text{and} \quad \|\sqrt{B^{-1}}g\|^2 = g^{\mathrm{T}}B^{-1}g$$

so that the Cauchy-Schwarz inequality shows that

$$(g^{\mathrm{T}}g)^2 = (g^{\mathrm{T}}\sqrt{B}\sqrt{B^{-1}}g)^2 \leqslant (g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g).$$

Therefore, $h'(\alpha) \geqslant 0$. □

§4.1 Algorithms Based on the Cauchy Point

Proof (cont'd).

Now we compute $h'(\alpha)$ and obtain

$$\begin{aligned} h'(\alpha) &= -(p^U)^{\mathrm{T}}(p^U - p^B) + \alpha\|p^U - p^B\|^2 \geqslant -(p^U)^{\mathrm{T}}(p^U - p^B) \\ &= \frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg} g^{\mathrm{T}}\Big(-\frac{g^{\mathrm{T}}g}{g^{\mathrm{T}}Bg}g + B^{-1}g\Big) \\ &= g^{\mathrm{T}}g \cdot \frac{g^{\mathrm{T}}B^{-1}g}{g^{\mathrm{T}}Bg}\Big[1 - \frac{(g^{\mathrm{T}}g)^2}{(g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g)}\Big]. \end{aligned}$$

By the fact that B is positive definite,

$$\|g^{\mathrm{T}}\sqrt{B}\|^2 = g^{\mathrm{T}}Bg \quad \text{and} \quad \|\sqrt{B^{-1}}g\|^2 = g^{\mathrm{T}}B^{-1}g$$

so that the Cauchy-Schwarz inequality shows that

$$(g^{\mathrm{T}}g)^2 = (g^{\mathrm{T}}\sqrt{B}\sqrt{B^{-1}}g)^2 \leqslant (g^{\mathrm{T}}Bg)(g^{\mathrm{T}}B^{-1}g).$$

Therefore, $h'(\alpha) \geqslant 0$. □

§4.1 Algorithms Based on the Cauchy Point

Proof (cont'd).

For ②, we define $\widehat{h}(\alpha) = m(\widetilde{p}(1+\alpha))$ and show that $\widehat{h}'(\alpha) \leqslant 0$ for $\alpha \in (0,1)$. Substitution of

$$\widetilde{p}(1+\alpha) = p^U + \alpha(p^B - p^U)$$

into (5) and differentiation with respect to the argument leads to

$$\begin{aligned} \widehat{h}'(\alpha) &= (p^B - p^U)^{\mathrm{T}}(g + Bp^U) + \alpha(p^B - p^U)^{\mathrm{T}} B(p^B - p^U) \\ &\leqslant (p^B - p^U)^{\mathrm{T}}(g + Bp^U + B(p^B - p^U)) \\ &= (p^B - p^U)^{\mathrm{T}}(g + Bp^B) = 0\,, \end{aligned}$$

giving the result. □

§4.1 Algorithms Based on the Cauchy Point

It follows from this lemma that the path $\widetilde{p}(\tau)$ intersects the trust-region boundary $\|p\| \leqslant \Delta$ at exactly one point if $\|p^B\| \geqslant \Delta$, and nowhere otherwise. Since m is decreasing along the path, the chosen value of p will be at p^B if $\|p^B\| \leqslant \Delta$, otherwise at the point of intersection of the dogleg and the trust-region boundary. In the latter case, we compute the appropriate value of τ by solving the following scalar quadratic equation:

$$\|p^U + (\tau - 1)(p^B - p^U)\|^2 = \Delta^2.$$

§4.1 Algorithms Based on the Cauchy Point

Consider now the case in which the exact Hessian $(\nabla^2 f)(x_k)$ is available for use in the model problem (5):

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

When $(\nabla^2 f)(x_k)$ is positive definite, we can simply set $B = (\nabla^2 f)(x_k)$ (that is, $p^B = -(\nabla^2 f(x_k))^{-1} g_k$) and apply the procedure above to find the Newton-dogleg step. Otherwise, we can define p^B by choosing B to be one of the positive definite modified Hessians described in Section 3.4, then proceed as above to find the dogleg step. Near a solution satisfying second-order sufficient conditions ($(\nabla^2 f)(x_*)$ is positive definite), p^B will be set to the usual Newton step, allowing the possibility of rapid local convergence of Newton's method (see Section 4.4).

§4.1 Algorithms Based on the Cauchy Point

Consider now the case in which the exact Hessian $(\nabla^2 f)(x_k)$ is available for use in the model problem (5):

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

When $(\nabla^2 f)(x_k)$ is positive definite, we can simply set $B = (\nabla^2 f)(x_k)$ (that is, $p^B = -(\nabla^2 f(x_k))^{-1} g_k$) and apply the procedure above to find the Newton-dogleg step. Otherwise, we can define p^B by choosing B to be one of the positive definite modified Hessians described in Section 3.4, then proceed as above to find the dogleg step. Near a solution satisfying second-order sufficient conditions ($(\nabla^2 f)(x_*)$ is positive definite), p^B will be set to the usual Newton step, allowing the possibility of rapid local convergence of Newton's method (see Section 4.4).

§4.1 Algorithms Based on the Cauchy Point

Consider now the case in which the exact Hessian $(\nabla^2 f)(x_k)$ is available for use in the model problem (5):

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta \,. \tag{5}$$

When $(\nabla^2 f)(x_k)$ is positive definite, we can simply set $B = (\nabla^2 f)(x_k)$ (that is, $p^B = -(\nabla^2 f(x_k))^{-1} g_k$) and apply the procedure above to find the Newton-dogleg step. Otherwise, we can define p^B by choosing B to be one of the positive definite modified Hessians described in Section 3.4, then proceed as above to find the dogleg step. Near a solution satisfying second-order sufficient conditions ($(\nabla^2 f)(x_*)$ is positive definite), p^B will be set to the usual Newton step, allowing the possibility of rapid local convergence of Newton's method (see Section 4.4).

§4.1 Algorithms Based on the Cauchy Point

The use of a modified Hessian in the Newton-dogleg method is not completely satisfying from an intuitive viewpoint. A modified factorization perturbs the diagonals of $(\nabla^2 f)(x_k)$ in a somewhat arbitrary manner, and the benefits of the trust-region approach may not be realized. In fact, the modification introduced during the factorization of the Hessian is redundant in some sense because the trust-region strategy introduces its own modification. As we show in Section 4.3, the exact solution of the trust-region problem (3) with $B_k = (\nabla^2 f)(x_k)$ is $-(\nabla^2 f(x_k) + \lambda \mathrm{I})^{-1} g_k$, where λ is chosen large enough to make $(\nabla^2 f(x_k) + \lambda \mathrm{I})$ positive definite, and its value depends on the trust-region radius Δ_k. We conclude that the Newton-dogleg method is most appropriate when the objective function is convex. The techniques described below may be more suitable for the general case.

§4.1 Algorithms Based on the Cauchy Point

The use of a modified Hessian in the Newton-dogleg method is not completely satisfying from an intuitive viewpoint. A modified factorization perturbs the diagonals of $(\nabla^2 f)(x_k)$ in a somewhat arbitrary manner, and the benefits of the trust-region approach may not be realized. In fact, the modification introduced during the factorization of the Hessian is redundant in some sense because the trust-region strategy introduces its own modification. As we show in Section 4.3, the exact solution of the trust-region problem (3) with $B_k = (\nabla^2 f)(x_k)$ is $-(\nabla^2 f(x_k) + \lambda \mathrm{I})^{-1} g_k$, where λ is chosen large enough to make $(\nabla^2 f(x_k) + \lambda \mathrm{I})$ positive definite, and its value depends on the trust-region radius Δ_k. We conclude that the Newton-dogleg method is most appropriate when the objective function is convex. The techniques described below may be more suitable for the general case.

§4.1 Algorithms Based on the Cauchy Point

The use of a modified Hessian in the Newton-dogleg method is not completely satisfying from an intuitive viewpoint. A modified factorization perturbs the diagonals of $(\nabla^2 f)(x_k)$ in a somewhat arbitrary manner, and the benefits of the trust-region approach may not be realized. In fact, the modification introduced during the factorization of the Hessian is redundant in some sense because the trust-region strategy introduces its own modification. As we show in Section 4.3, the exact solution of the trust-region problem (3) with $B_k = (\nabla^2 f)(x_k)$ is $-(\nabla^2 f(x_k) + \lambda \mathrm{I})^{-1} g_k$, where λ is chosen large enough to make $(\nabla^2 f(x_k) + \lambda \mathrm{I})$ positive definite, and its value depends on the trust-region radius Δ_k. We conclude that the Newton-dogleg method is most appropriate when the objective function is convex. The techniques described below may be more suitable for the general case.

§4.1 Algorithms Based on the Cauchy Point

The use of a modified Hessian in the Newton-dogleg method is not completely satisfying from an intuitive viewpoint. A modified factorization perturbs the diagonals of $(\nabla^2 f)(x_k)$ in a somewhat arbitrary manner, and the benefits of the trust-region approach may not be realized. In fact, the modification introduced during the factorization of the Hessian is redundant in some sense because the trust-region strategy introduces its own modification. As we show in Section 4.3, the exact solution of the trust-region problem (3) with $B_k = (\nabla^2 f)(x_k)$ is $-(\nabla^2 f(x_k) + \lambda \mathrm{I})^{-1} g_k$, where λ is chosen large enough to make $(\nabla^2 f(x_k) + \lambda \mathrm{I})$ positive definite, and its value depends on the trust-region radius Δ_k. We conclude that the Newton-dogleg method is most appropriate when the objective function is convex. The techniques described below may be more suitable for the general case.

§4.1 Algorithms Based on the Cauchy Point

The dogleg strategy can be adapted to handle indefinite matrices B, but there is not much point in doing so because the full step p^B is not the unconstrained minimizer of m in this case. Instead, we now describe another strategy, which aims to include directions of negative curvature (that is, directions d for which $d^{\mathrm{T}}Bd < 0$) in the space of candidate trust-region steps.

§4.1 Algorithms Based on the Cauchy Point

- **Two-dimensional Subspace Minimization**

When B is positive definite, the dogleg method strategy can be made slightly more sophisticated by widening the search for p to the entire two-dimensional subspace spanned by p^U and p^B (equivalently, g and $-B^{-1}g$). The sub-problem (5) is replaced by

$$\min_p m(p) = f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad \text{s.t. } \|p\| \leqslant \Delta,\, p \in \operatorname{span}[g, B^{-1}g] \,. \quad (15)$$

This is a problem in two variables that is computationally inexpensive to solve. Clearly, the Cauchy point p^c is feasible for (15), so the optimal solution of this sub-problem yields at least as much reduction in m as the Cauchy point, resulting in global convergence of the algorithm. The two-dimensional subspace minimization strategy is obviously an extension of the dogleg method as well, since the entire dogleg path lies in $\operatorname{span}[g, B^{-1}g]$.

§4.1 Algorithms Based on the Cauchy Point

- **Two-dimensional Subspace Minimization**

When B is positive definite, the dogleg method strategy can be made slightly more sophisticated by widening the search for p to the entire two-dimensional subspace spanned by p^U and p^B (equivalently, g and $-B^{-1}g$). The sub-problem (5) is replaced by

$$\min_p m(p) = f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t. } \|p\| \leqslant \Delta,\ p \in \operatorname{span}[g, B^{-1}g]\,. \tag{15}$$

This is a problem in two variables that is computationally inexpensive to solve. Clearly, the Cauchy point p^C is feasible for (15), so the optimal solution of this sub-problem yields at least as much reduction in m as the Cauchy point, resulting in global convergence of the algorithm. The two-dimensional subspace minimization strategy is obviously an extension of the dogleg method as well, since the entire dogleg path lies in $\operatorname{span}[g, B^{-1}g]$.

§4.1 Algorithms Based on the Cauchy Point

- **Two-dimensional Subspace Minimization**

When B is positive definite, the dogleg method strategy can be made slightly more sophisticated by widening the search for p to the entire two-dimensional subspace spanned by p^U and p^B (equivalently, g and $-B^{-1}g$). The sub-problem (5) is replaced by

$$\min_p m(p) = f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t. } \|p\| \leqslant \Delta,\ p \in \text{span}[g, B^{-1}g]\,. \tag{15}$$

This is a problem in two variables that is computationally inexpensive to solve. Clearly, the Cauchy point p^c is feasible for (15), so the optimal solution of this sub-problem yields at least as much reduction in m as the Cauchy point, resulting in global convergence of the algorithm. The two-dimensional subspace minimization strategy is obviously an extension of the dogleg method as well, since the entire dogleg path lies in span$[g, B^{-1}g]$.

§4.1 Algorithms Based on the Cauchy Point

- **Two-dimensional Subspace Minimization**

When B is positive definite, the dogleg method strategy can be made slightly more sophisticated by widening the search for p to the entire two-dimensional subspace spanned by p^U and p^B (equivalently, g and $-B^{-1}g$). The sub-problem (5) is replaced by

$$\min_p m(p) = f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t. } \|p\| \leqslant \Delta,\ p \in \operatorname{span}[g, B^{-1}g]\,. \tag{15}$$

This is a problem in two variables that is computationally inexpensive to solve. Clearly, the Cauchy point p^c is feasible for (15), so the optimal solution of this sub-problem yields at least as much reduction in m as the Cauchy point, resulting in global convergence of the algorithm. The two-dimensional subspace minimization strategy is obviously an extension of the dogleg method as well, since the entire dogleg path lies in $\operatorname{span}[g, B^{-1}g]$.

§4.1 Algorithms Based on the Cauchy Point

This strategy can be modified to handle the case of indefinite B. When B has negative eigenvalues, the two-dimensional subspace in (15) is changed to

$$\text{span}[g, (B+\alpha \mathrm{I})^{-1} g] \quad \text{for some } \alpha \in (-\lambda_1, -2\lambda_1], \tag{16}$$

where λ_1 denotes the most negative/smallest eigenvalue of B. When $\|(B+\alpha \mathrm{I})^{-1} g\| \leqslant \Delta$, we discard the subspace search of (15), (16) and instead define the step to be

$$p = -(B+\alpha \mathrm{I})^{-1} g + v, \tag{17}$$

where v is a vector that satisfies $v^{\mathrm{T}}(B+\alpha \mathrm{I})^{-1} g \leqslant 0$. (This condition ensures that $\|p\| \geqslant \|(B+\alpha \mathrm{I})^{-1} g\|$.) When B has zero eigenvalues but no negative eigenvalues, we define the step to be the Cauchy point $p = p^C$.

§4.1 Algorithms Based on the Cauchy Point

This strategy can be modified to handle the case of indefinite B. When B has negative eigenvalues, the two-dimensional subspace in (15) is changed to

$$\operatorname{span}[g, (B+\alpha \mathrm{I})^{-1} g] \quad \text{for some } \alpha \in (-\lambda_1, -2\lambda_1], \tag{16}$$

where λ_1 denotes the most negative/smallest eigenvalue of B. When $\|(B+\alpha \mathrm{I})^{-1} g\| \leqslant \Delta$, we discard the subspace search of (15), (16) and instead define the step to be

$$p = -(B+\alpha \mathrm{I})^{-1} g + v, \tag{17}$$

where v is a vector that satisfies $v^{\mathrm{T}}(B+\alpha \mathrm{I})^{-1} g \leqslant 0$. (This condition ensures that $\|p\| \geqslant \|(B+\alpha \mathrm{I})^{-1} g\|$.) When B has zero eigenvalues but no negative eigenvalues, we define the step to be the Cauchy point $p = p^C$.

§4.1 Algorithms Based on the Cauchy Point

When the exact Hessian is available, we can set $B = (\nabla^2 f)(x_k)$, and note that $B^{-1}g$ is the Newton step. Hence, when the Hessian is positive definite at the solution x_* and when x_k is close to x_* and Δ is sufficiently large, the subspace minimization problem (15) will be solved by the Newton step. The reduction in model function m achieved by the two-dimensional subspace minimization strategy often is close to the reduction achieved by the exact solution of (5).

Most of the computational effort lies in a single factorization of B or $B + \alpha \mathrm{I}$ (estimation of α and solution of (15) are less significant), while strategies that find nearly exact solutions of (5) typically require two or three such factorizations (see Section 4.3).

§4.2 Global Convergence

- **Reduction Obtained by the Cauchy Point**

In the preceding discussion of algorithms for approximately solving the trust-region sub-problem, we have repeatedly emphasized that global convergence depends on the approximate solution obtaining at least as much decrease in the model function m as the Cauchy point. In fact, a fixed positive fraction of the Cauchy decrease suffices. We start the global convergence analysis by obtaining an estimate of the decrease in m achieved by the Cauchy point. We then use this estimate to prove that the sequence of gradients $\{g_k\}$ generated by Algorithm 4.1 has an accumulation point at zero, and in fact converges to zero when η is strictly positive.

§4.2 Global Convergence

- **Reduction Obtained by the Cauchy Point**

In the preceding discussion of algorithms for approximately solving the trust-region sub-problem, we have repeatedly emphasized that global convergence depends on the approximate solution obtaining at least as much decrease in the model function m as the Cauchy point. In fact, a fixed positive fraction of the Cauchy decrease suffices. We start the global convergence analysis by obtaining an estimate of the decrease in m achieved by the Cauchy point. We then use this estimate to prove that the sequence of gradients $\{g_k\}$ generated by Algorithm 4.1 has an accumulation point at zero, and in fact converges to zero when η is strictly positive.

§4.2 Global Convergence

Steihaug's algorithm (Algorithm 7.2) produce approximate solutions p_k of the sub-problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \Delta_k\,, \tag{3}$$

that satisfy the following estimate of decrease in the model function:

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0, 1]$. The usefulness of this estimate will become clear in the following two sections. For now, we note that when Δ_k is the minimum value in (18), the condition is slightly reminiscent of the first Wolfe condition: The (least) desired reduction in the model is proportional to the gradient and the size of the step.

Next we show now that the Cauchy point p_k^c satisfies (18), with $c_1 = 1/2$.

§4.2 Global Convergence

Steihaug's algorithm (Algorithm 7.2) produce approximate solutions p_k of the sub-problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \Delta_k\,, \qquad (3)$$

that satisfy the following estimate of decrease in the model function:

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \qquad (18)$$

for some constant $c_1 \in (0, 1]$. The usefulness of this estimate will become clear in the following two sections. For now, we note that when Δ_k is the minimum value in (18), the condition is slightly reminiscent of the first Wolfe condition: The (least) desired reduction in the model is proportional to the gradient and the size of the step.

Next we show now that the Cauchy point p_k^c satisfies (18), with $c_1 = 1/2$.

§4.2 Global Convergence

Lemma

The Cauchy point p_k^c satisfies (18) with $c_1 = 1/2$; that is,

$$m_k(0) - m_k(p_k^c) \geqslant \frac{1}{2}\|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{19}$$

Proof.

For simplicity, we drop the iteration index k in the proof.

We consider first the case $g^{\mathrm{T}}Bg \leqslant 0$. Here, we have

$$\begin{aligned} m(p^c) - m(0) &= m\big(-\frac{\Delta}{\|g\|}g\big) - f \\ &= -\frac{\Delta}{\|g\|}\|g\|^2 + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}g^{\mathrm{T}}Bg \\ &\leqslant -\Delta\|g\| \leqslant -\|g\| \min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big), \end{aligned}$$

and so (19) certainly holds. □

§4.2 Global Convergence

Lemma

The Cauchy point p_k^c satisfies (18) *with* $c_1 = 1/2$*; that is,*

$$m_k(0) - m_k(p_k^c) \geqslant \frac{1}{2}\|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{19}$$

Proof.

For simplicity, we drop the iteration index k in the proof.

We consider first the case $g^{\mathrm{T}}Bg \leqslant 0$. Here, we have

$$\begin{aligned} m(p^c) - m(0) &= m\Big(-\frac{\Delta}{\|g\|}g\Big) - f \\ &= -\frac{\Delta}{\|g\|}\|g\|^2 + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}g^{\mathrm{T}}Bg \\ &\leqslant -\Delta\|g\| \leqslant -\|g\| \min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big), \end{aligned}$$

and so (19) certainly holds. □

§4.2 Global Convergence

Proof (cont'd).

For the next case, consider $g^{\mathrm{T}}Bg > 0$ and

$$\frac{\|g\|^3}{\Delta g^{\mathrm{T}}Bg} > 1\,. \tag{20}$$

Recall from (9) and (10) that for the case $g^{\mathrm{T}}Bg > 0$ the Cauchy point p_k^c is given by

$$p^c = -\min\Big(\frac{\|g\|^3}{\Delta g^{\mathrm{T}}Bg}, 1\Big)\frac{\Delta}{\|g\|}g = -\frac{\Delta}{\|g\|}g\,.$$

Using this fact together with (20), we obtain

$$\begin{aligned} m(p^c) - m(0) &= -\frac{\Delta}{\|g\|}\|g\|^2 + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}g^{\mathrm{T}}Bg \leqslant -\Delta\|g\| + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}\frac{\|g\|^3}{\Delta} \\ &= -\frac{1}{2}\Delta\|g\| \leqslant -\frac{1}{2}\|g\|\min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big) \end{aligned}$$

Therefore, in this case (19) holds too. □

§4.2 Global Convergence

Proof (cont'd).

For the next case, consider $g^{\mathrm{T}}Bg > 0$ and

$$\frac{\|g\|^3}{\Delta g^{\mathrm{T}}Bg} > 1\,. \tag{20}$$

Recall from (9) and (10) that for the case $g^{\mathrm{T}}Bg > 0$ the Cauchy point p_k^c is given by

$$p^c = -\min\Big(\frac{\|g\|^3}{\Delta g^{\mathrm{T}}Bg}, 1\Big)\frac{\Delta}{\|g\|}g = -\frac{\Delta}{\|g\|}g\,.$$

Using this fact together with (20), we obtain

$$\begin{aligned} m(p^c) - m(0) &= -\frac{\Delta}{\|g\|}\|g\|^2 + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}g^{\mathrm{T}}Bg \leqslant -\Delta\|g\| + \frac{1}{2}\frac{\Delta^2}{\|g\|^2}\frac{\|g\|^3}{\Delta} \\ &= -\frac{1}{2}\Delta\|g\| \leqslant -\frac{1}{2}\|g\|\min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big) \end{aligned}$$

Therefore, in this case (19) holds too. □

§4.2 Global Convergence

Proof (cont'd).

In the remaining case, (20) does not hold, and therefore

$$\frac{\|g\|^3}{\Delta g^{\mathrm{T}} B g} \leqslant 1\,. \tag{21}$$

From the formula for the Cauchy point we have

$$p^C = -\min\Big(\frac{\|g\|^3}{\Delta g^{\mathrm{T}} B g}, 1\Big)\frac{\Delta}{\|g\|} g = -\frac{\|g\|^2}{g^{\mathrm{T}} B g} g\,,$$

so

$$\begin{aligned} m(p^C) - m(0) &= -\frac{\|g\|^2}{g^{\mathrm{T}} B g}\|g\|^2 + \frac{1}{2}\frac{\|g\|^4}{(g^{\mathrm{T}} B g)^2} g^{\mathrm{T}} B g = -\frac{1}{2}\frac{\|g\|^4}{g^{\mathrm{T}} B g} \\ &\leqslant -\frac{1}{2}\frac{\|g\|^4}{\|B\|\|g\|^2} = -\frac{1}{2}\frac{\|g\|^2}{\|B\|} \leqslant -\frac{1}{2}\|g\| \min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big)\,, \end{aligned}$$

yielding the desired result (19) once again. □

§4.2 Global Convergence

Proof (cont'd).

In the remaining case, (20) does not hold, and therefore

$$\frac{\|g\|^3}{\Delta g^{\mathrm{T}} B g} \leqslant 1\,. \tag{21}$$

From the formula for the Cauchy point we have

$$p^c = -\min\Big(\frac{\|g\|^3}{\Delta g^{\mathrm{T}} B g}, 1\Big)\frac{\Delta}{\|g\|} g = -\frac{\|g\|^2}{g^{\mathrm{T}} B g} g\,,$$

so

$$\begin{aligned} m(p^c) - m(0) &= -\frac{\|g\|^2}{g^{\mathrm{T}} B g}\|g\|^2 + \frac{1}{2}\frac{\|g\|^4}{(g^{\mathrm{T}} B g)^2} g^{\mathrm{T}} B g = -\frac{1}{2}\frac{\|g\|^4}{g^{\mathrm{T}} B g} \\ &\leqslant -\frac{1}{2}\frac{\|g\|^4}{\|B\|\|g\|^2} = -\frac{1}{2}\frac{\|g\|^2}{\|B\|} \leqslant -\frac{1}{2}\|g\| \min\Big(\Delta, \frac{\|g\|}{\|B\|}\Big)\,, \end{aligned}$$

yielding the desired result (19) once again. □

§4.2 Global Convergence

To satisfy

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0, 1]$, our approximate solution p_k has only to achieve a reduction that is at least some fixed fraction c_2 of the reduction achieved by the Cauchy point. We state the observation formally as a theorem.

Theorem

Let p_k be any vector such that $\|p_k\| \leqslant \Delta_k$ and

$$m_k(0) - m_k(p_k) \geqslant c_2\big[m_k(0) - m_k(p_k^C)\big].$$

Then p_k satisfies (18) with $c_1 = c_2/2$. In particular, if p_k is the exact solution p_k^* of (3) (the minimization of m_k subject to the trust-region bound), then it satisfies (18) with $c_1 = 1/2$.

§4.2 Global Convergence

To satisfy

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0, 1]$, our approximate solution p_k has only to achieve a reduction that is at least some fixed fraction c_2 of the reduction achieved by the Cauchy point. We state the observation formally as a theorem.

Theorem

Let p_k be any vector such that $\|p_k\| \leqslant \Delta_k$ and

$$m_k(0) - m_k(p_k) \geqslant c_2 \big[m_k(0) - m_k(p_k^c)\big] .$$

Then p_k satisfies (18) with $c_1 = c_2/2$. In particular, if p_k is the exact solution p_k^* of (3) (the minimization of m_k subject to the trust-region bound), then it satisfies (18) with $c_1 = 1/2$.

§4.2 Global Convergence

Note that the dogleg and two-dimensional subspace minimization algorithms both satisfy (18) with $c_1 = 1/2$, because they all produce approximate solutions p_k for which

$$m_k(p_k) \leqslant m_k(p_k^C)\,.$$

§4.2 Global Convergence

- **Convergence to Stationary Points**

Global convergence results for trust-region methods come in two varieties, depending on whether we set the parameter η in Algorithm 4.1 to zero or to some small positive value. When $\eta = 0$, (that is, the step is taken whenever it produces a lower value of f), we can show that the sequence of gradients $\{g_k\}$ has a limit point at zero. For the more stringent acceptance test with $\eta > 0$, which requires the actual decrease in f to be at least some small fraction of the predicted decrease, we have the stronger result that $g_k \to 0$.

§4.2 Global Convergence

Algorithm 4.1 (Trust Region).

Given $\hat{\Delta} > 0$, $\Delta_0 \in (0, \hat{\Delta})$, and $\eta \in [0, 1/4)$;

for $k = 0, 1, 2, \cdots$

Obtain p_k by (approximately) solving (3) & evaluate ρ_k from (4);

if $\rho_k < 1/4$

$\Delta_{k+1} = \frac{1}{4}\Delta_k$;

else

if $\rho_k > 3/4$ and $\|p_k\| = \Delta_k$

$\Delta_{k+1} = \min(2\Delta_k, \hat{\Delta})$;

else

$\Delta_{k+1} = \Delta_k$;

if $\rho_k > \eta$

$x_{k+1} = x_k + p_k$;

else

$x_{k+1} = x_k$;

end (for)

§4.2 Global Convergence

In this section we prove the global convergence results for both cases. We assume that the approximate Hessians B_k are uniformly bounded, and that f is bounded from below on the level set

$$S \equiv \left\{x \,\middle|\, f(x) \leqslant f(x_0)\right\}. \tag{22}$$

For later reference, we define an open neighborhood of this set by

$$S(R_0) \equiv \left\{x \,\middle|\, \|x-y\| < R_0 \text{ for some } y \in S\right\},$$

where R_0 is a positive constant.

To allow our results to be applied more generally, we also allow the length of the approximate solution p_k of (3) to exceed the trust-region bound, provided that it stays within some fixed multiple of the bound; that is,

$$\|p_k\| \leqslant \gamma \Delta_k \quad \text{for some constant } \gamma \geqslant 1. \tag{23}$$

§4.2 Global Convergence

In this section we prove the global convergence results for both cases. We assume that the approximate Hessians B_k are uniformly bounded, and that f is bounded from below on the level set

$$S \equiv \left\{x \,\middle|\, f(x) \leqslant f(x_0)\right\}. \tag{22}$$

For later reference, we define an open neighborhood of this set by

$$S(R_0) \equiv \left\{x \,\middle|\, \|x - y\| < R_0 \text{ for some } y \in S\right\},$$

where R_0 is a positive constant.

To allow our results to be applied more generally, we also allow the length of the approximate solution p_k of (3) to exceed the trust-region bound, provided that it stays within some fixed multiple of the bound; that is,

$$\|p_k\| \leqslant \gamma \Delta_k \quad \text{for some constant } \gamma \geqslant 1. \tag{23}$$

§4.2 Global Convergence

Theorem

Consider solving the minimization problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \gamma\Delta_k\,, \tag{3'}$$

using Algorithm 4.1, where $\gamma \geqslant 1$ *is a fixed constant in* (3'). *Suppose that* $\|B_k\| \leqslant \beta$ *for some constant* β, *that* f *is bounded from below on the level set* S *defined by* (22) *and Lipschitz continuously differentiable in the neighborhood* $S(R_0)$ *for some* $R_0 > 0$, *and that all approximate solutions of* (3') *satisfy the inequalities*

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0,1]$. *We then have* $\liminf\limits_{k\to\infty} \|g_k\| = 0$. *Moreover, if* $\eta > 0$, *then* $\lim\limits_{k\to\infty} \|g_k\| = 0$.

§4.2 Global Convergence

Theorem

Consider solving the minimization problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \gamma\Delta_k\,, \tag{3'}$$

using Algorithm 4.1, where $\gamma \geqslant 1$ is a fixed constant in (3'). Suppose that $\|B_k\| \leqslant \beta$ for some constant β, that f is bounded from below on the level set S defined by (22) and Lipschitz continuously differentiable in the neighborhood $S(R_0)$ for some $R_0 > 0$, and that all approximate solutions of (3') satisfy the inequalities

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0,1]$. We then have $\liminf\limits_{k\to\infty} \|g_k\| = 0$. Moreover, if $\eta > 0$, then $\lim\limits_{k\to\infty} \|g_k\| = 0$.

§4.2 Global Convergence

Theorem

Consider solving the minimization problem

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad s.t. \quad \|p\| \leqslant \gamma\Delta_k\,, \tag{3'}$$

using Algorithm 4.1, where $\gamma \geqslant 1$ is a fixed constant in (3'). Suppose that $\|B_k\| \leqslant \beta$ for some constant β, that f is bounded from below on the level set S defined by (22) and Lipschitz continuously differentiable in the neighborhood $S(R_0)$ for some $R_0 > 0$, and that all approximate solutions of (3') satisfy the inequalities

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some constant $c_1 \in (0,1]$. We then have $\liminf_{k\to\infty} \|g_k\| = 0$. Moreover, if $\eta > 0$, then $\lim_{k\to\infty} \|g_k\| = 0$.

§4.2 Global Convergence

Proof (cont'd).

Suppose the contrary (of $\liminf_{k\to\infty} \|g_k\| = 0$) that there are $\varepsilon > 0$ and $K > 0$ such that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

From (18) we have for $k \geqslant K$ that

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

We claim that there exists $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k,

where ρ_k is the ratio between the actual reduction and the model reduction ρ_k given by

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}. \tag{4}$$

□

§4.2 Global Convergence

Proof (cont'd).

Suppose the contrary (of $\liminf_{k\to\infty} \|g_k\| = 0$) that there are $\varepsilon > 0$ and $K > 0$ such that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

From (18) we have for $k \geqslant K$ that

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

We claim that there exists $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k,

where ρ_k is the ratio between the actual reduction and the model reduction ρ_k given by

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}. \tag{4}$$

§4.2 Global Convergence

Proof (cont'd).

Suppose the contrary (of $\liminf_{k\to\infty} \|g_k\| = 0$) that there are $\varepsilon > 0$ and $K > 0$ such that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

From (18) we have for $k \geqslant K$ that

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

We claim that there exists $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k,

where ρ_k is the ratio between the actual reduction and the model reduction ρ_k given by

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}. \tag{4}$$

□

§4.2 Global Convergence

Proof (cont'd).

Suppose (for now) that the claim is valid so that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k.

Recall that in Algorithm 4.1 we have

1. $\Delta_{k+1} = \Delta_k/4$ if $\rho_k < 1/4$;
2. $\Delta_{k+1} \geqslant \Delta_k$ if $\rho_k \geqslant 1/4$.

Therefore, $\Delta_{k+1} \geqslant \Delta_k$ whenever Δ_k falls below the threshold $\bar{\Delta}$ and reduction of Δ_k (by a factor of $1/4$) can occur in our algorithm only if $\Delta_k \geqslant \bar{\Delta}$. We then conclude that

$$\Delta_k \geqslant \min\big(\Delta_K, \bar{\Delta}/4\big) \quad \forall\, k \geqslant K. \tag{26}$$

The contradiction will be based on (26). □

§4.2 Global Convergence

Proof (cont'd).

Suppose (for now) that the claim is valid so that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k.

Recall that in Algorithm 4.1 we have

1. $\Delta_{k+1} = \Delta_k/4$ if $\rho_k < 1/4$;
2. $\Delta_{k+1} \geqslant \Delta_k$ if $\rho_k \geqslant 1/4$.

Therefore, $\Delta_{k+1} \geqslant \Delta_k$ whenever Δ_k falls below the threshold $\bar{\Delta}$ and reduction of Δ_k (by a factor of $1/4$) can occur in our algorithm only if $\Delta_k \geqslant \bar{\Delta}$. We then conclude that

$$\Delta_k \geqslant \min\left(\Delta_K, \bar{\Delta}/4\right) \quad \forall\, k \geqslant K. \tag{26}$$

The contradiction will be based on (26). □

§4.2 Global Convergence

Proof (cont'd).

Suppose (for now) that the claim is valid so that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k.

Recall that in Algorithm 4.1 we have

1. $\Delta_{k+1} = \Delta_k/4$ if $\rho_k < 1/4$;
2. $\Delta_{k+1} \geqslant \Delta_k$ if $\rho_k \geqslant 1/4$.

Therefore, $\Delta_{k+1} \geqslant \Delta_k$ whenever Δ_k falls below the threshold $\bar{\Delta}$ and reduction of Δ_k (by a factor of $1/4$) can occur in our algorithm only if $\Delta_k \geqslant \bar{\Delta}$. We then conclude that

$$\Delta_k \geqslant \min\left(\Delta_K, \bar{\Delta}/4\right) \quad \forall\, k \geqslant K. \tag{26}$$

The contradiction will be based on (26). □

§4.2 Global Convergence

Proof (cont'd).

Suppose (for now) that the claim is valid so that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k.

Recall that in Algorithm 4.1 we have

1. $\Delta_{k+1} = \Delta_k/4$ if $\rho_k < 1/4$;
2. $\Delta_{k+1} \geqslant \Delta_k$ if $\rho_k \geqslant 1/4$.

Therefore, $\Delta_{k+1} \geqslant \Delta_k$ whenever Δ_k falls below the threshold $\bar{\Delta}$ and reduction of Δ_k (by a factor of $1/4$) can occur in our algorithm only if $\Delta_k \geqslant \bar{\Delta}$. We then conclude that

$$\Delta_k \geqslant \min\big(\Delta_K, \bar{\Delta}/4\big) \quad \forall\, k \geqslant K. \tag{26}$$

The contradiction will be based on (26). □

§4.2 Global Convergence

Proof (cont'd).

Let $\mathcal{K} = \{k \geqslant K \,|\, \rho_k \geqslant 1/4\}$. Note that if $k \in \mathcal{K}$, $x_{k+1} = x_k + p_k$. By the definition of ρ_k and inequalities (25), (26), for $k \in \mathcal{K}$ we have

$$f(x_k) - f(x_{k+1}) = f(x_k) - f(x_k + p_k) \geqslant \frac{1}{4}\big[m_k(0) - m_k(p_k)\big]$$
$$\geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big) \geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big).$$

Since $\{f(x_k)\}$ is a decreasing sequence and f is bounded from below,

$$f(x_0) - \inf_S f \geqslant \sum_{k=0}^{\infty} \big[f(x_k) - f(x_{k+1})\big] \geqslant \frac{\#\mathcal{K}}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big),$$

so $\#\mathcal{K} < \infty$. Hence $\rho_k < 1/4$ for all k sufficiently large. Therefore, Δ_k will eventually be multiplied by $1/4$ at every iteration, and we have $\lim\limits_{k\to\infty} \Delta_k = 0$, which contradicts (26). Hence, our original assertion (24) must be false, giving that $\liminf\limits_{k\to\infty} \|g_k\| = 0$. □

§4.2 Global Convergence

Proof (cont'd).

Let $\mathcal{K} = \{k \geqslant K \,|\, \rho_k \geqslant 1/4\}$. Note that if $k \in \mathcal{K}$, $x_{k+1} = x_k + p_k$. By the definition of ρ_k and inequalities (25), (26), for $k \in \mathcal{K}$ we have

$$f(x_k) - f(x_{k+1}) = f(x_k) - f(x_k + p_k) \geqslant \frac{1}{4}\big[m_k(0) - m_k(p_k)\big]$$
$$\geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big) \geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big).$$

Since $\{f(x_k)\}$ is a decreasing sequence and f is bounded from below,

$$f(x_0) - \inf_S f \geqslant \sum_{k=0}^{\infty}\big[f(x_k) - f(x_{k+1})\big] \geqslant \frac{\#\mathcal{K}}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big),$$

so $\#\mathcal{K} < \infty$. Hence $\rho_k < 1/4$ for all k sufficiently large. Therefore, Δ_k will eventually be multiplied by $1/4$ at every iteration, and we have $\lim\limits_{k\to\infty} \Delta_k = 0$, which contradicts (26). Hence, our original assertion (24) must be false, giving that $\liminf\limits_{k\to\infty} \|g_k\| = 0$. □

§4.2 Global Convergence

Proof (cont'd).

Let $\mathcal{K} = \{k \geqslant K \,|\, \rho_k \geqslant 1/4\}$. Note that if $k \in \mathcal{K}$, $x_{k+1} = x_k + p_k$. By the definition of ρ_k and inequalities (25), (26), for $k \in \mathcal{K}$ we have

$$f(x_k) - f(x_{k+1}) = f(x_k) - f(x_k + p_k) \geqslant \frac{1}{4}\big[m_k(0) - m_k(p_k)\big]$$
$$\geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big) \geqslant \frac{1}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big).$$

Since $\{f(x_k)\}$ is a decreasing sequence and f is bounded from below,

$$f(x_0) - \inf_S f \geqslant \sum_{k=0}^{\infty} \big[f(x_k) - f(x_{k+1})\big] \geqslant \frac{\#\mathcal{K}}{4} c_1 \varepsilon \min\Big(\Delta_K, \frac{\bar{\Delta}}{4}, \frac{\varepsilon}{\beta}\Big),$$

so $\#\mathcal{K} < \infty$. Hence $\rho_k < 1/4$ for all k sufficiently large. Therefore, Δ_k will eventually be multiplied by $1/4$ at every iteration, and we have $\lim\limits_{k\to\infty} \Delta_k = 0$, which contradicts (26). Hence, our original assertion (24) must be false, giving that $\liminf\limits_{k\to\infty} \|g_k\| = 0$. □

§4.2 Global Convergence

Proof (cont'd).

Now we establish the claim that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k

under the assumption (of $\liminf_{k\to\infty} \|g_k\| > 0$) that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

so that for $k \geqslant K$

$$m_k(0) - m_k(p_k) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

Let β_1 denote the Lipschitz constant for g on the set $S(R_0)$, and define

$$\bar{\Delta} = \min\Big(\frac{c_1 \varepsilon}{\gamma^2(\beta + \beta_1)}, \frac{R_0}{2\gamma}\Big).$$

The fact that $c_1 \leqslant 1$ and $\gamma \geqslant 1$ imply that $\bar{\Delta} \leqslant \varepsilon/\beta$. □

§4.2 Global Convergence

Proof (cont'd).

Now we establish the claim that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k

under the assumption (of $\liminf\limits_{k\to\infty} \|g_k\| > 0$) that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

so that for $k \geqslant K$

$$m_k(0) - m_k(p_k) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

Let β_1 denote the Lipschitz constant for g on the set $S(R_0)$, and define

$$\bar{\Delta} = \min\Big(\frac{c_1 \varepsilon}{\gamma^2(\beta + \beta_1)}, \frac{R_0}{2\gamma}\Big).$$

The fact that $c_1 \leqslant 1$ and $\gamma \geqslant 1$ imply that $\bar{\Delta} \leqslant \varepsilon/\beta$. □

§4.2 Global Convergence

Proof (cont'd).

Now we establish the claim that there exists a $\bar{\Delta} > 0$ such that

if $\Delta_k \leqslant \bar{\Delta}$ for some particular $k \geqslant K$, then $\rho_k > 1/4$ for such k

under the assumption (of $\liminf_{k\to\infty} \|g_k\| > 0$) that

$$\|g_k\| \geqslant \varepsilon \quad \forall\, k \geqslant K. \tag{24}$$

so that for $k \geqslant K$

$$m_k(0) - m_k(p_k) \geqslant c_1 \varepsilon \min\Big(\Delta_k, \frac{\varepsilon}{\beta}\Big). \tag{25}$$

Let β_1 denote the Lipschitz constant for g on the set $S(R_0)$, and define

$$\bar{\Delta} = \min\Big(\frac{c_1 \varepsilon}{\gamma^2(\beta + \beta_1)}, \frac{R_0}{2\gamma}\Big).$$

The fact that $c_1 \leqslant 1$ and $\gamma \geqslant 1$ imply that $\bar{\Delta} \leqslant \varepsilon/\beta$. □

§4.2 Global Convergence

Proof (cont'd).

Let $k \geqslant K$ satisfy $\Delta_k \leqslant \bar{\Delta}$. Then $\|p_k\| \leqslant \gamma\Delta_k \leqslant R_0/2$ so that

$$x_k + tp_k \in S(R_0) \quad \forall\, t \in [0,1]\,.$$

By Taylor's theorem,

$$f(x_k + p_k) = f(x_k) + g(x_k)^{\mathrm{T}} p_k + \int_0^1 \big[g(x_k + tp_k) - g(x_k)\big]^{\mathrm{T}} p_k\, dt.$$

Therefore, the Lipschitz condition shows that

$$\begin{aligned}
&\big|m_k(p_k) - f(x_k + p_k)\big| \\
&\quad = \Big|\frac{1}{2} p_k^{\mathrm{T}} B_k p_k - \int_0^1 \big[g(x_k + tp_k) - g(x_k)\big]^{\mathrm{T}} p_k\, dt\Big| \\
&\quad \leqslant \frac{\beta}{2}\|p_k\|^2 + \int_0^1 \beta_1 t \|p_k\|^2\, dt \\
&\quad \leqslant \frac{\beta}{2}\|p_k\|^2 + \frac{\beta_1}{2}\|p_k\|^2 \leqslant \frac{\gamma^2}{2}\Delta_k^2(\beta + \beta_1)\,. \qquad (27)
\end{aligned}$$

§4.2 Global Convergence

Proof (cont'd).

Let $k \geqslant K$ satisfy $\Delta_k \leqslant \bar{\Delta}$. Then $\|p_k\| \leqslant \gamma\Delta_k \leqslant R_0/2$ so that

$$x_k + tp_k \in S(R_0) \quad \forall\, t \in [0,1]\,.$$

By Taylor's theorem,

$$f(x_k+p_k) = f(x_k) + g(x_k)^{\mathrm{T}} p_k + \int_0^1 \big[g(x_k+tp_k) - g(x_k)\big]^{\mathrm{T}} p_k\, dt.$$

Therefore, the Lipschitz condition shows that

$$\begin{aligned}
&\big|m_k(p_k) - f(x_k+p_k)\big| \\
&\quad = \Big|\frac{1}{2}p_k^{\mathrm{T}} B_k p_k - \int_0^1 \big[g(x_k+tp_k) - g(x_k)\big]^{\mathrm{T}} p_k\, dt\Big| \\
&\quad \leqslant \frac{\beta}{2}\|p_k\|^2 + \int_0^1 \beta_1 t \|p_k\|^2\, dt \\
&\quad \leqslant \frac{\beta}{2}\|p_k\|^2 + \frac{\beta_1}{2}\|p_k\|^2 \leqslant \frac{\gamma^2}{2}\Delta_k^2(\beta+\beta_1)\,.
\end{aligned} \tag{27}$$

□

§4.2 Global Convergence

Proof (cont'd).

Since $\Delta_k \leqslant \bar{\Delta}$, using (27), the inequality

$$m_k(0) - m_k(p_k) \geqslant c_1 \varepsilon \min\left(\Delta_k, \frac{\varepsilon}{\beta}\right), \tag{25}$$

the definition of $\bar{\Delta} = \min\left(\frac{c_1 \varepsilon}{\gamma^2(\beta + \beta_1)}, \frac{R_0}{2\gamma}\right)$, and the fact that $\bar{\Delta} \leqslant \varepsilon/\beta$ we find that

$$\begin{aligned}|\rho_k - 1| &= \left|\frac{(f(x_k) - f(x_k + p_k)) - (m_k(0) - m_k(p_k))}{m_k(0) - m_k(p_k)}\right| \\ &= \left|\frac{m_k(p_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}\right| \leqslant \frac{\gamma^2 \Delta_k^2 (\beta + \beta_1)}{2c_1 \varepsilon \min(\Delta_k, \varepsilon/\beta)} \\ &= \frac{\gamma^2 \Delta_k^2 (\beta + \beta_1)}{2c_1 \varepsilon \Delta_k} \leqslant \frac{\gamma^2 \bar{\Delta}(\beta + \beta_1)}{2c_1 \varepsilon} \leqslant \frac{1}{2}\end{aligned}$$

which shows that $\rho_k > \frac{1}{4}$. Therefore, the claim is established. □

§4.2 Global Convergence

Proof (cont'd).

Now suppose that $\eta > 0$. Let $m \in \mathbb{N}$ satisfy $g_m \neq 0$. Define

$$\varepsilon = \frac{1}{2}\|g_m\|\,, \quad R = \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big)\,.$$

Since $B[x_m, R] = \{x \,|\, \|x - x_m\| \leqslant R\}$ is contained in $S(R_0)$, by the Lipschitz condition,

$$\|g(x) - g_m\| \leqslant \beta_1\|x - x_m\| \quad \forall\, x \in B[x_m, R]\,.$$

Therefore, if $x \in B[x_m, R]$, we have

$$\|g(x)\| \geqslant \|g_m\| - \|g(x) - g_m\| \geqslant 2\varepsilon - \varepsilon = \varepsilon\,;$$

thus if the entire sequence $\{x_k\}_{k \geqslant m}$ stays inside the ball $B[x_m, R]$, we would have $\|g_k\| \geqslant \varepsilon > 0$ for all $k \geqslant m$, and the reasoning in the previous case shows that this scenario **cannot** occur. Therefore, the sequence $\{x_k\}_{k \geqslant m}$ eventually leaves $B[x_m, R]$. □

§4.2 Global Convergence

Proof (cont'd).

Now suppose that $\eta > 0$. Let $m \in \mathbb{N}$ satisfy $g_m \neq 0$. Define

$$\varepsilon = \frac{1}{2}\|g_m\|, \quad R = \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big).$$

Since $B[x_m, R] = \big\{x \,\big|\, \|x - x_m\| \leqslant R\big\}$ is contained in $S(R_0)$, by the Lipschitz condition,

$$\|g(x) - g_m\| \leqslant \beta_1 \|x - x_m\| \quad \forall\, x \in B[x_m, R].$$

Therefore, if $x \in B[x_m, R]$, we have

$$\|g(x)\| \geqslant \|g_m\| - \|g(x) - g_m\| \geqslant 2\varepsilon - \varepsilon = \varepsilon;$$

thus if the entire sequence $\{x_k\}_{k \geqslant m}$ stays inside the ball $B[x_m, R]$, we would have $\|g_k\| \geqslant \varepsilon > 0$ for all $k \geqslant m$, and the reasoning in the previous case shows that this scenario **cannot** occur. Therefore, the sequence $\{x_k\}_{k \geqslant m}$ eventually leaves $B[x_m, R]$. □

§4.2 Global Convergence

Proof (cont'd).

Now suppose that $\eta > 0$. Let $m \in \mathbb{N}$ satisfy $g_m \neq 0$. Define

$$\varepsilon = \frac{1}{2}\|g_m\|\,, \quad R = \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big)\,.$$

Since $B[x_m, R] = \big\{x \,\big|\, \|x - x_m\| \leqslant R\big\}$ is contained in $S(R_0)$, by the Lipschitz condition,

$$\|g(x) - g_m\| \leqslant \beta_1 \|x - x_m\| \quad \forall\, x \in B[x_m, R]\,.$$

Therefore, if $x \in B[x_m, R]$, we have

$$\|g(x)\| \geqslant \|g_m\| - \|g(x) - g_m\| \geqslant 2\varepsilon - \varepsilon = \varepsilon\,;$$

thus if the entire sequence $\{x_k\}_{k \geqslant m}$ stays inside the ball $B[x_m, R]$, we would have $\|g_k\| \geqslant \varepsilon > 0$ for all $k \geqslant m$, and the reasoning in the previous case shows that this scenario **cannot** occur. Therefore, the sequence $\{x_k\}_{k \geqslant m}$ eventually leaves $B[x_m, R]$. □

§4.2 Global Convergence

Proof (cont'd).

Let the index $\ell \geqslant m$ be such that $x_{\ell+1}$ is the **first** iterate after x_m outside $B[x_m, R]$. Since $\|g_k\| \geqslant \varepsilon$ for $k = m, m+1, \cdots, \ell$, we can use the condition

$$m_k(0) - m_k(p_k) \geqslant c_1 \varepsilon \min\left(\Delta_k, \frac{\varepsilon}{\beta}\right) \tag{25}$$

to conclude

$$\begin{aligned} f(x_m) - f(x_{\ell+1}) &= \sum_{k=m}^{\ell} \left[f(x_k) - f(x_{k+1}) \right] \geqslant \sum_{\substack{k=m \\ x_k \neq x_{k+1}}}^{\ell} \eta \left[m_k(0) - m_k(p_k) \right] \\ &\geqslant \sum_{\substack{k=m \\ x_k \neq x_{k+1}}}^{\ell} \eta c_1 \varepsilon \min\left(\Delta_k, \frac{\varepsilon}{\beta}\right), \end{aligned}$$

where we have limited the sum to the iterations k for which $x_k \neq x_{k+1}$; that is, those iterations on which a step was actually taken. □

§4.2 Global Convergence

Proof (cont'd).

If $\Delta_k \leqslant \varepsilon/\beta$ for all $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, we have

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \sum_{\substack{k=m \\ x_k \neq x_{k+1}}}^{\ell} \Delta_k \geqslant \eta c_1 \varepsilon R = \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big). \quad (28)$$

Otherwise, if $\Delta_k > \varepsilon/\beta$ for some $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, then

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \cdot \frac{\varepsilon}{\beta}. \quad (29)$$

Since $\{f(x_k)\}_{k=0}^{\infty}$ is decreasing and bounded from below, $f(x_k) \searrow f_*$ for some $f_* > -\infty$. Therefore, using (28) and (29), we obtain

$$\begin{aligned} f(x_m) - f_* \geqslant f(x_m) - f(x_{\ell+1}) &\geqslant \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta}, \frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big) \\ &= \frac{1}{4}\eta c_1 \|g_m\| \min\Big(\frac{\|g_m\|}{\beta}, \frac{\|g_m\|}{\beta_1}, R_0\Big) > 0. \end{aligned}$$

Since $f(x_m) - f_* \searrow 0$, we must have $\|g_m\| \to 0$. □

§4.2 Global Convergence

Proof (cont'd).

If $\Delta_k \leqslant \varepsilon/\beta$ for all $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, we have

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \sum_{\substack{k=m \\ x_k \neq x_{k+1}}}^{\ell} \Delta_k \geqslant \eta c_1 \varepsilon R = \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big). \quad (28)$$

Otherwise, if $\Delta_k > \varepsilon/\beta$ for some $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, then

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \cdot \frac{\varepsilon}{\beta}. \quad (29)$$

Since $\{f(x_k)\}_{k=0}^{\infty}$ is decreasing and bounded from below, $f(x_k) \searrow f_*$ for some $f_* > -\infty$. Therefore, using (28) and (29), we obtain

$$f(x_m) - f_* \geqslant f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta}, \frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big)$$

$$= \frac{1}{4} \eta c_1 \|g_m\| \min\Big(\frac{\|g_m\|}{\beta}, \frac{\|g_m\|}{\beta_1}, R_0\Big) > 0.$$

Since $f(x_m) - f_* \searrow 0$, we must have $\|g_m\| \to 0$. □

§4.2 Global Convergence

Proof (cont'd).

If $\Delta_k \leqslant \varepsilon/\beta$ for all $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, we have

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \sum_{\substack{k=m \\ x_k \neq x_{k+1}}}^{\ell} \Delta_k \geqslant \eta c_1 \varepsilon R = \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big). \quad (28)$$

Otherwise, if $\Delta_k > \varepsilon/\beta$ for some $m \leqslant k \leqslant \ell$ with $x_k \neq x_{k+1}$, then

$$f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \cdot \frac{\varepsilon}{\beta}. \quad (29)$$

Since $\{f(x_k)\}_{k=0}^{\infty}$ is decreasing and bounded from below, $f(x_k) \searrow f_*$ for some $f_* > -\infty$. Therefore, using (28) and (29), we obtain

$$f(x_m) - f_* \geqslant f(x_m) - f(x_{\ell+1}) \geqslant \eta c_1 \varepsilon \min\Big(\frac{\varepsilon}{\beta}, \frac{\varepsilon}{\beta_1}, \frac{R_0}{2}\Big)$$

$$= \frac{1}{4}\eta c_1 \|g_m\| \min\Big(\frac{\|g_m\|}{\beta}, \frac{\|g_m\|}{\beta_1}, R_0\Big) > 0.$$

Since $f(x_m) - f_* \searrow 0$, we must have $\|g_m\| \to 0$. □

§4.3 Iterative Solution of the Sub-problem

In this section, we describe a technique that uses the characterization

$$(B + \lambda \mathrm{I}) p_* = -g \quad \text{for some } \lambda \geqslant 0 \tag{6a}$$

of the sub-problem solution, applying Newton's method to find the value of λ which matches the given trust-region radius Δ in

$$\min_{p \in \mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

We also prove the key result concerning the characterization of solutions of (5).

The methods of Section 4.1 make no serious attempt to find the exact solution of the sub-problem (5). They do, however, make some use of the information in the model Hessian B_k, and they have advantages of reasonable implementation cost and nice global convergence properties.

§4.3 Iterative Solution of the Sub-problem

In this section, we describe a technique that uses the characterization

$$(B+\lambda \mathrm{I})p_* = -g \quad \text{for some } \lambda \geqslant 0 \tag{6a}$$

of the sub-problem solution, applying Newton's method to find the value of λ which matches the given trust-region radius Δ in

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad \text{s.t.} \quad \|p\| \leqslant \Delta\,. \tag{5}$$

We also prove the key result concerning the characterization of solutions of (5).

The methods of Section 4.1 make no serious attempt to find the exact solution of the sub-problem (5). They do, however, make some use of the information in the model Hessian B_k, and they have advantages of reasonable implementation cost and nice global convergence properties.

§4.3 Iterative Solution of the Sub-problem

When the problem is relatively small (that is, n is not too large), it may be worthwhile to exploit the model more fully by looking for a closer approximation to the solution of the sub-problem. In this section, we describe an approach for finding a good approximation at the cost of a few factorizations of the matrix B (typically three factorization), as compared with a single factorization for the dogleg and two-dimensional subspace minimization methods. This approach is based on the characterization of the exact solution given in the key theorem (shown in the next slide), together with an ingenious application of Newton's method in one variable. Essentially, the algorithm tries to identify the value of λ for which (6a) is satisfied by the solution of (5).

§4.3 Iterative Solution of the Sub-problem

When the problem is relatively small (that is, n is not too large), it may be worthwhile to exploit the model more fully by looking for a closer approximation to the solution of the sub-problem. In this section, we describe an approach for finding a good approximation at the cost of a few factorizations of the matrix B (typically three factorization), as compared with a single factorization for the dogleg and two-dimensional subspace minimization methods. This approach is based on the characterization of the exact solution given in the key theorem (shown in the next slide), together with an ingenious application of Newton's method in one variable. Essentially, the algorithm tries to identify the value of λ for which (6a) is satisfied by the solution of (5).

§4.3 Iterative Solution of the Sub-problem

When the problem is relatively small (that is, n is not too large), it may be worthwhile to exploit the model more fully by looking for a closer approximation to the solution of the sub-problem. In this section, we describe an approach for finding a good approximation at the cost of a few factorizations of the matrix B (typically three factorization), as compared with a single factorization for the dogleg and two-dimensional subspace minimization methods. This approach is based on the characterization of the exact solution given in the key theorem (shown in the next slide), together with an ingenious application of Newton's method in one variable. Essentially, the algorithm tries to identify the value of λ for which (6a) is satisfied by the solution of (5).

§4.3 Iterative Solution of the Sub-problem

Theorem (Key theorem in Section 4.3)

The vector p_* is a **global** *solution of the trust-region problem*

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

if and only if p_* is feasible and there is a scalar $\lambda \geqslant 0$ such that the following conditions are satisfied:

$$(B + \lambda \mathrm{I}) p_* = -g\,, \tag{6a}$$

$$\lambda(\Delta - \|p_*\|) = 0\,, \tag{6b}$$

$$(B + \lambda \mathrm{I}) \text{ is positive semi-definite.} \tag{6c}$$

§4.3 Iterative Solution of the Sub-problem

The characterization of the key theorem suggests an algorithm for finding the solution p of

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

Either $\lambda = 0$ satisfies (6a) and (6c) with $\|p\| \leqslant \Delta$, or else we define

$$p(\lambda) = -(B + \lambda \mathrm{I})^{-1} g$$

for λ sufficiently large that $B + \lambda\mathrm{I}$ is positive definite and seek a value $\lambda > 0$ such that

$$\|p(\lambda)\| = \Delta\,. \tag{30}$$

This problem is a one-dimensional root-finding problem in the variable λ.

§4.3 Iterative Solution of the Sub-problem

To see that a value of λ with all the desired properties exists, we appeal to the **eigen-decomposition** of B and use it to study the properties of $\|p(\lambda)\|$. Since B is symmetric, there is an orthogonal matrix Q and a diagonal matrix Λ such that $B = Q\Lambda Q^{\mathrm{T}}$, where

$$\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n),$$

and $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$ are the eigenvalues of B. Clearly, $B + \lambda \mathrm{I} = Q(\Lambda + \lambda \mathrm{I})Q^{\mathrm{T}}$, and for $\lambda \neq \lambda_j$, we have

$$p(\lambda) = -Q(\Lambda + \lambda \mathrm{I})^{-1} Q^{\mathrm{T}} g = -\sum_{j=1}^{n} \frac{q_j^{\mathrm{T}} g}{\lambda_j + \lambda} q_j, \tag{31}$$

where q_j denotes the j-th column of Q. Therefore, by orthonormality of $q_1, q_2, \cdots, q_n$, we have

$$\|p(\lambda)\|^2 = \sum_{j=1}^{n} \frac{(q_j^{\mathrm{T}} g)^2}{(\lambda_j + \lambda)^2}. \tag{32}$$

§4.3 Iterative Solution of the Sub-problem

To see that a value of λ with all the desired properties exists, we appeal to the **eigen-decomposition** of B and use it to study the properties of $\|p(\lambda)\|$. Since B is symmetric, there is an orthogonal matrix Q and a diagonal matrix Λ such that $B = Q\Lambda Q^{\mathrm{T}}$, where

$$\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)\,,$$

and $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$ are the eigenvalues of B. Clearly, $B + \lambda \mathrm{I} = Q(\Lambda + \lambda \mathrm{I})Q^{\mathrm{T}}$, and for $\lambda \neq \lambda_j$, we have

$$p(\lambda) = -Q(\Lambda + \lambda \mathrm{I})^{-1} Q^{\mathrm{T}} g = -\sum_{j=1}^{n} \frac{q_j^{\mathrm{T}} g}{\lambda_j + \lambda} q_j\,, \tag{31}$$

where q_j denotes the j-th column of Q. Therefore, by orthonormality of $q_1, q_2, \cdots, q_n$, we have

$$\|p(\lambda)\|^2 = \sum_{j=1}^{n} \frac{(q_j^{\mathrm{T}} g)^2}{(\lambda_j + \lambda)^2}\,. \tag{32}$$

§4.3 Iterative Solution of the Sub-problem

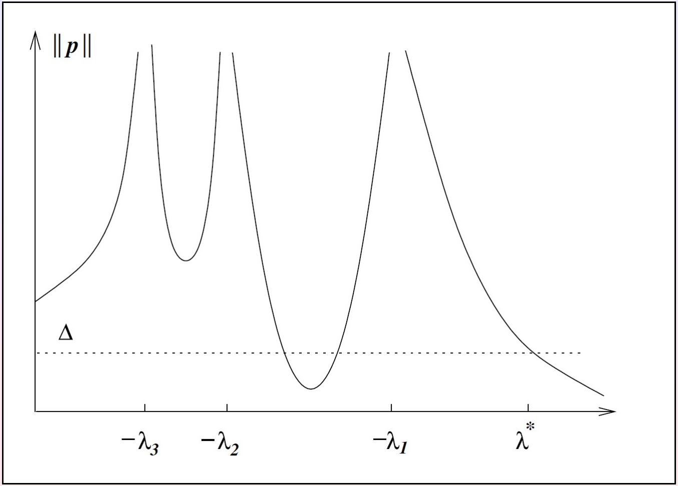

Figure 5: $\|p(\lambda)\|$ as a function of λ.

§4.3 Iterative Solution of the Sub-problem

This expression tells us a lot about $\|p(\lambda)\|$. If $\lambda > -\lambda_1$, we have $\lambda_j + \lambda > 0$ for all $j = 1, 2, \cdots, n$, and so $\|p(\lambda)\|$ is a continuous, non-increasing function of λ on the interval $(-\lambda_1, \infty)$. In fact,

$$\lim_{\lambda\to\infty} \|p(\lambda)\| = 0 \tag{33}$$

and

$$\lim_{\lambda\to-\lambda_j} \|p(\lambda)\| = \infty \quad \text{if } q_j^{\mathrm{T}} g \neq 0. \tag{34}$$

Figure 5 plots $\|p(\lambda)\|$ against λ in a case in which $q_1^{\mathrm{T}} g$, $q_2^{\mathrm{T}} g$, and $q_3^{\mathrm{T}} g$ are all nonzero. Note that the properties (33) and (34) hold and that $\|p(\lambda)\|$ is a non-increasing function of λ on $(-\lambda_1, \infty)$. In particular, when $q_1^{\mathrm{T}} g \neq 0$, there is always a unique value $\lambda_* \in (-\lambda_1, \infty)$ such that $\|p(\lambda_*)\| = \Delta$. Note that there may be other smaller values of λ for which $\|p(\lambda)\| = \Delta$, but these will fail to satisfy (6c).

§4.3 Iterative Solution of the Sub-problem

This expression tells us a lot about $\|p(\lambda)\|$. If $\lambda > -\lambda_1$, we have $\lambda_j + \lambda > 0$ for all $j = 1, 2, \cdots, n$, and so $\|p(\lambda)\|$ is a continuous, non-increasing function of λ on the interval $(-\lambda_1, \infty)$. In fact,

$$\lim_{\lambda \to \infty} \|p(\lambda)\| = 0 \tag{33}$$

and

$$\lim_{\lambda \to -\lambda_j} \|p(\lambda)\| = \infty \quad \text{if } q_j^{\mathrm{T}} g \neq 0. \tag{34}$$

Figure 5 plots $\|p(\lambda)\|$ against λ in a case in which $q_1^{\mathrm{T}} g$, $q_2^{\mathrm{T}} g$, and $q_3^{\mathrm{T}} g$ are all nonzero. Note that the properties (33) and (34) hold and that $\|p(\lambda)\|$ is a non-increasing function of λ on $(-\lambda_1, \infty)$. In particular, when $q_1^{\mathrm{T}} g \neq 0$, there is always a unique value $\lambda_* \in (-\lambda_1, \infty)$ such that $\|p(\lambda_*)\| = \Delta$. Note that there may be other smaller values of λ for which $\|p(\lambda)\| = \Delta$, but these will fail to satisfy (6c).

§4.3 Iterative Solution of the Sub-problem

This expression tells us a lot about $\|p(\lambda)\|$. If $\lambda > -\lambda_1$, we have $\lambda_j + \lambda > 0$ for all $j = 1, 2, \cdots, n$, and so $\|p(\lambda)\|$ is a continuous, non-increasing function of λ on the interval $(-\lambda_1, \infty)$. In fact,

$$\lim_{\lambda\to\infty} \|p(\lambda)\| = 0 \tag{33}$$

and

$$\lim_{\lambda\to-\lambda_j} \|p(\lambda)\| = \infty \quad \text{if } q_j^{\mathrm{T}} g \neq 0. \tag{34}$$

Figure 5 plots $\|p(\lambda)\|$ against λ in a case in which $q_1^{\mathrm{T}} g$, $q_2^{\mathrm{T}} g$, and $q_3^{\mathrm{T}} g$ are all nonzero. Note that the properties (33) and (34) hold and that $\|p(\lambda)\|$ is a non-increasing function of λ on $(-\lambda_1, \infty)$. In particular, when $q_1^{\mathrm{T}} g \neq 0$, there is always a unique value $\lambda_* \in (-\lambda_1, \infty)$ such that $\|p(\lambda_*)\| = \Delta$. Note that there may be other smaller values of λ for which $\|p(\lambda)\| = \Delta$, but these will fail to satisfy (6c).

§4.3 Iterative Solution of the Sub-problem

We now sketch a procedure for identifying the $\lambda_* \in (-\lambda_1, \infty)$ for which $\|p(\lambda_*)\| = \Delta$, which works when $q_1^{\mathrm{T}} g \neq 0$ (and leave the case of $q_1^{\mathrm{T}} g = 0$ later). First, note that when B is positive definite and $\|B^{-1}g\| \leqslant \Delta$, the value $\lambda = 0$ satisfies (6), so the procedure can be terminated immediately with $\lambda_* = 0$. Otherwise, we could use the root-finding Newton's method to find the value of $\lambda > -\lambda_1$ that solves

$$\varphi_1(\lambda) = \|p(\lambda)\| - \Delta = 0. \tag{35}$$

The disadvantage of this approach can be seen by considering the form of $\|p(\lambda)\|$ when λ is greater than, but close to, $-\lambda_1$. For such λ, we can approximate φ_1 by a rational function, as follows:

$$\varphi_1(\lambda) \approx \frac{C_1}{\lambda + \lambda_1} + C_2\,,$$

where $C_1 > 0$ and C_2 are constants.

§4.3 Iterative Solution of the Sub-problem

We now sketch a procedure for identifying the $\lambda_* \in (-\lambda_1, \infty)$ for which $\|p(\lambda_*)\| = \Delta$, which works when $q_1^{\mathrm{T}} g \neq 0$ (and leave the case of $q_1^{\mathrm{T}} g = 0$ later). First, note that when B is positive definite and $\|B^{-1} g\| \leqslant \Delta$, the value $\lambda = 0$ satisfies (6), so the procedure can be terminated immediately with $\lambda_* = 0$. Otherwise, we could use the root-finding Newton's method to find the value of $\lambda > -\lambda_1$ that solves

$$\varphi_1(\lambda) = \|p(\lambda)\| - \Delta = 0. \tag{35}$$

The disadvantage of this approach can be seen by considering the form of $\|p(\lambda)\|$ when λ is greater than, but close to, $-\lambda_1$. For such λ, we can approximate φ_1 by a rational function, as follows:

$$\varphi_1(\lambda) \approx \frac{C_1}{\lambda + \lambda_1} + C_2 \,,$$

where $C_1 > 0$ and C_2 are constants.

§4.3 Iterative Solution of the Sub-problem

Clearly this approximation (and hence φ_1) is highly nonlinear, so the root-finding Newton's method will be unreliable or slow. Better results will be obtained if we reformulate the problem (35) so that it is nearly linear near the optimal λ. By defining

$$\varphi_2(\lambda) = \frac{1}{\Delta} - \frac{1}{\|p(\lambda)\|},$$

using (32) we can show that for λ slightly greater than $-\lambda_1$,

$$\varphi_2(\lambda) \approx \frac{1}{\Delta} - \frac{\lambda + \lambda_1}{C_3}$$

for some $C_3 > 0$. Hence, φ_2 is nearly linear near $-\lambda_1$ (see Figure 6), and the root-finding Newton's method will perform well, provided that it maintains $\lambda > -\lambda_1$.

§4.3 Iterative Solution of the Sub-problem

Clearly this approximation (and hence φ_1) is highly nonlinear, so the root-finding Newton's method will be unreliable or slow. Better results will be obtained if we reformulate the problem (35) so that it is nearly linear near the optimal λ. By defining

$$\varphi_2(\lambda) = \frac{1}{\Delta} - \frac{1}{\|p(\lambda)\|},$$

using (32) we can show that for λ slightly greater than $-\lambda_1$,

$$\varphi_2(\lambda) \approx \frac{1}{\Delta} - \frac{\lambda + \lambda_1}{C_3}$$

for some $C_3 > 0$. Hence, φ_2 is nearly linear near $-\lambda_1$ (see Figure 6), and the root-finding Newton's method will perform well, provided that it maintains $\lambda > -\lambda_1$.

§4.3 Iterative Solution of the Sub-problem

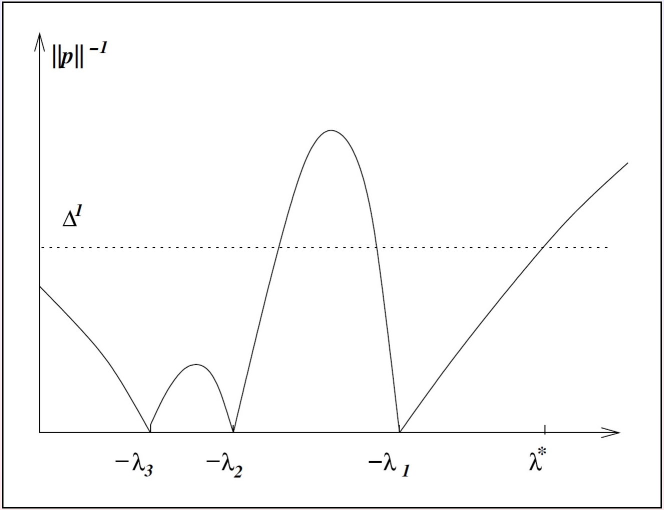

Figure 6: $1/\|p(\lambda)\|$ as a function of λ.

§4.3 Iterative Solution of the Sub-problem

The root-finding Newton's method applied to φ_2 generates a sequence of iterates $\lambda^{(\ell)}$ by setting

$$\lambda^{(\ell+1)} = \lambda^{(\ell)} - \frac{\varphi_2\big(\lambda^{(\ell)}\big)}{\varphi_2'\big(\lambda^{(\ell)}\big)}. \tag{36}$$

After some elementary manipulation, this updating formula can be implemented in the following practical way.

Algorithm 4.3 (Trust Region Sub-problem).

Given $\lambda^{(0)}$, $\Delta > 0$;

for $\ell = 0, 1, 2, \cdots$

Factor $B + \lambda^{(\ell)}\mathrm{I} = R^{\mathrm{T}}R$;

Solve $R^{\mathrm{T}}Rp_\ell = -g$, $R^T q_\ell = p_\ell$;

Set $\lambda^{(\ell+1)} = \lambda^{(\ell)} + \frac{\|p_\ell\|^2}{\|q_\ell\|^2} \cdot \frac{\|p_\ell\| - \Delta}{\Delta}$;

end (for)

§4.3 Iterative Solution of the Sub-problem

The root-finding Newton's method applied to φ_2 generates a sequence of iterates $\lambda^{(\ell)}$ by setting

$$\lambda^{(\ell+1)} = \lambda^{(\ell)} - \frac{\varphi_2\big(\lambda^{(\ell)}\big)}{\varphi_2'\big(\lambda^{(\ell)}\big)}\,. \tag{36}$$

After some elementary manipulation, this updating formula can be implemented in the following practical way.

Algorithm 4.3 (Trust Region Sub-problem).

Given $\lambda^{(0)}$, $\Delta > 0$;

for $\ell = 0, 1, 2, \cdots$

Factor $B + \lambda^{(\ell)}\mathrm{I} = R^{\mathrm{T}}R$;

Solve $R^{\mathrm{T}}Rp_\ell = -g$, $R^T q_\ell = p_\ell$;

Set $\lambda^{(\ell+1)} = \lambda^{(\ell)} + \frac{\|p_\ell\|^2}{\|q_\ell\|^2} \cdot \frac{\|p_\ell\| - \Delta}{\Delta}$;

end (**for**)

§4.3 Iterative Solution of the Sub-problem

Safeguards must be added to this algorithm to make it practical; for instance, when $\lambda^{(\ell)} < -\lambda_1$, the Cholesky factorization $B + \lambda^{(\ell)}\mathrm{I} = R^{\mathrm{T}}R$ will not exist. A slightly enhanced version of this algorithm does, however, converge to a solution of (30) in most cases.

The main work in each iteration of this method is, of course, the Cholesky factorization of $B + \lambda^{(\ell)}\mathrm{I}$. Practical versions of this algorithm do **not** iterate till convergence to the optimal λ is obtained with high accuracy, but are content with an approximate solution that can be obtained in two or three iterations.

§4.3 Iterative Solution of the Sub-problem

- **The hard case**

Recall that in the discussion above, we assumed that $q_1^{\mathrm{T}} g \neq 0$. In fact, the approach described above can be applied even when the most negative eigenvalue is a multiple eigenvalue (that is, $0 > \lambda_1 = \lambda_2 = \cdots$), provided that $Q_1^{\mathrm{T}} g \neq 0$, where Q_1 is the matrix whose columns span the subspace corresponding to the eigenvalue λ_1. When this condition does not hold, the situation becomes a little complicated, because the limit

$$\lim_{\lambda \to -\lambda_1} \|p(\lambda)\| = \infty$$

does not hold and so there may not be a value $\lambda \in (-\lambda_1, \infty)$ such that $\|p(\lambda)\| = \Delta$ (see Figure 7). Moré and Sorensen [214] refer to this case as the hard case.

§4.3 Iterative Solution of the Sub-problem

- **The hard case**

Recall that in the discussion above, we assumed that $q_1^{\mathrm{T}} g \neq 0$. In fact, the approach described above can be applied even when the most negative eigenvalue is a multiple eigenvalue (that is, $0 > \lambda_1 = \lambda_2 = \cdots$), provided that $Q_1^{\mathrm{T}} g \neq 0$, where Q_1 is the matrix whose columns span the subspace corresponding to the eigenvalue λ_1. When this condition does not hold, the situation becomes a little complicated, because the limit

$$\lim_{\lambda \to -\lambda_1} \|p(\lambda)\| = \infty$$

does not hold and so there may not be a value $\lambda \in (-\lambda_1, \infty)$ such that $\|p(\lambda)\| = \Delta$ (see Figure 7). Moré and Sorensen [214] refer to this case as the hard case.

§4.3 Iterative Solution of the Sub-problem

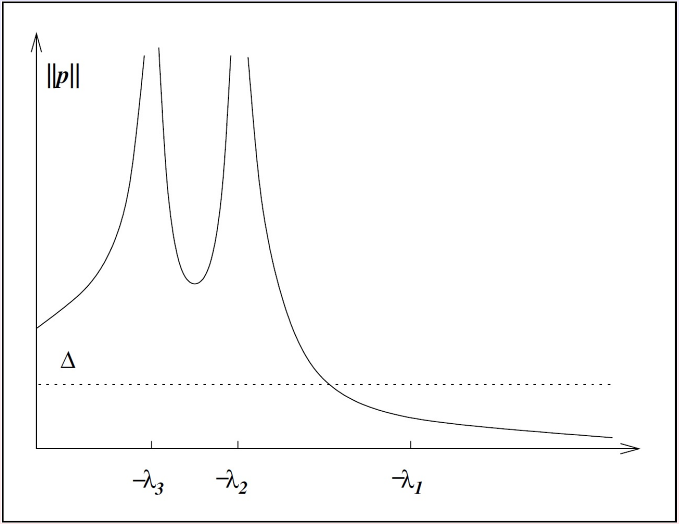

Figure 7: The hard case: $\|p(\lambda)\| < \Delta$ for all $\lambda \in (-\lambda_1, \infty)$.

§4.3 Iterative Solution of the Sub-problem

At first glance, it is not clear how p and λ can be chosen to satisfy (6) in the hard case. Clearly, our root-finding technique will not work, since there is no solution for λ in the open interval $(-\lambda_1, \infty)$. Nevertheless, the key theorem assures us that the right value of λ lies in the interval $[-\lambda_1, \infty)$, so there is only one possibility: $\lambda = -\lambda_1$. To find p, it is not enough to delete the terms for which $\lambda_j = \lambda_1$ from the formula (31) and set

$$p = \sum_{j:\lambda_j \neq \lambda_1} \frac{q_j^{\mathrm{T}} g}{\lambda_j - \lambda_1} q_j \,.$$

Instead, we note that $(B - \lambda_1 \mathrm{I})$ is singular, so there is a vector z such that $\|z\| = 1$ and $(B - \lambda_1 \mathrm{I})z = 0$. In fact, z is an eigenvector of B corresponding to the eigenvalue λ_1, so by orthogonality of Q we have $q_j^{\mathrm{T}} z = 0$ for index j with $\lambda_j \neq \lambda_1$.

§4.3 Iterative Solution of the Sub-problem

It follows from this property that if we set

$$p = \sum_{j:\lambda_j \neq \lambda_1} \frac{q_j^{\mathrm{T}} g}{\lambda_j - \lambda_1} q_j + \tau z \tag{37}$$

for any scalar τ, we have

$$\|p\|^2 = \sum_{j:\lambda_j \neq \lambda_1} \frac{(q_j^{\mathrm{T}} g)^2}{(\lambda_j - \lambda_1)^2} + \tau^2,$$

so it is always possible to choose τ to ensure that $\|p\| = \Delta$. It is easy to check that the conditions (6) holds for this choice of p.

§4.3 Iterative Solution of the Sub-problem

We now give a formal proof of the key result that characterizes the exact solution of (5). The proof relies on the following technical lemma, which deals with the unconstrained minimizers of quadratics and is particularly interesting in the case where the Hessian is positive semi-definite.

Lemma

Let m be the quadratic function defined by

$$m(p) = g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B p \,, \tag{38}$$

where B is any symmetric matrix. Then

1. *m attains a minimum **if and only if** B is positive semi-definite and g is in the range of B. If B is positive semi-definite, then every p satisfying $Bp = -g$ is a global minimizer of m.*
2. *m has a unique minimizer **if and only if** B is positive definite.*

§4.3 Iterative Solution of the Sub-problem

Proof.

We prove each of the three claims in turn.

1. "$\Leftarrow$" Since g is in the range of B, there is a p with $Bp = -g$. For all $w \in \mathbb{R}^n$, we have

$$\begin{aligned} m(p+w) &= g^{\mathrm{T}}(p+w) + \frac{1}{2}(p+w)^{\mathrm{T}}B(p+w) \\ &= (g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp) + g^{\mathrm{T}}w + (Bp)^{\mathrm{T}}w + \frac{1}{2}w^{\mathrm{T}}Bw \\ &= m(p) + \frac{1}{2}w^{\mathrm{T}}Bw \\ &\geqslant m(p) \end{aligned} \tag{39}$$

since B is positive semi-definite. Hence, p is a minimizer of m.

"$\Rightarrow$" Let p be a minimizer of m. Since $\nabla m(p) = Bp + g = 0$, we have that g is in the range of B. Also, we have $\nabla^2 m(p) = B$ positive semi-definite, giving the result. □

§4.3 Iterative Solution of the Sub-problem

Proof.

We prove each of the three claims in turn.

1. "$\Leftarrow$" Since g is in the range of B, there is a p with $Bp = -g$. For all $w \in \mathbb{R}^n$, we have
$$\begin{aligned} m(p+w) &= g^{\mathrm{T}}(p+w) + \frac{1}{2}(p+w)^{\mathrm{T}}B(p+w) \\ &= (g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp) + g^{\mathrm{T}}w + (Bp)^{\mathrm{T}}w + \frac{1}{2}w^{\mathrm{T}}Bw \\ &= m(p) + \frac{1}{2}w^{\mathrm{T}}Bw \\ &\geqslant m(p) \end{aligned} \tag{39}$$
since B is positive semi-definite. Hence, p is a minimizer of m.
"$\Rightarrow$" Let p be a minimizer of m. Since $\nabla m(p) = Bp + g = 0$, we have that g is in the range of B. Also, we have $\nabla^2 m(p) = B$ positive semi-definite, giving the result. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

2. "$\Leftarrow$" The same argument as in ① suffices with the additional point that $w^{\mathrm{T}}Bw > 0$ whenever $w \neq 0$.

   "$\Rightarrow$" We proceed as in ① to deduce that B is positive semi-definite. If B is not positive definite, there is a vector $w \neq 0$ such that $Bw = 0$. Hence, from (39) we have $m(p+w) = m(p)$, so the minimizer is not unique, giving a contradiction. □

We also need/recall the following

Theorem (Lagrange Multiplier)

Let f and g be continuously differentiable functions. Suppose that subject to the constraint $g(x) = 0$ the function f attains its extrema at x_*. If $(\nabla g)(x_*) \neq 0$, then there is a real value λ such that $(\nabla f)(x_*) = \lambda(\nabla g)(x_*)$.

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

2. "$\Leftarrow$" The same argument as in ① suffices with the additional point that $w^{\mathrm{T}}Bw > 0$ whenever $w \neq 0$.

   "$\Rightarrow$" We proceed as in ① to deduce that B is positive semi-definite. If B is not positive definite, there is a vector $w \neq 0$ such that $Bw = 0$. Hence, from (39) we have $m(p+w) = m(p)$, so the minimizer is not unique, giving a contradiction. □

We also need/recall the following

Theorem (Lagrange Multiplier)

Let f and g be continuously differentiable functions. Suppose that subject to the constraint $g(x) = 0$ the function f attains its extrema at x_*. If $(\nabla g)(x_*) \neq 0$, then there is a real value λ such that $(\nabla f)(x_*) = \lambda(\nabla g)(x_*)$.

§4.3 Iterative Solution of the Sub-problem

Theorem (Key theorem in Section 4.3)

The vector p_* *is a* **global** *solution of the trust-region problem*

$$\min_{p\in\mathbb{R}^n} m(p) \equiv f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}Bp \quad s.t. \quad \|p\| \leqslant \Delta\,. \tag{5}$$

if and only if p_* *is feasible and there is a scalar* $\lambda \geqslant 0$ *such that the following conditions are satisfied:*

$$(B+\lambda\mathrm{I})p_* = -g\,, \tag{6a}$$

$$\lambda(\Delta - \|p_*\|) = 0\,, \tag{6b}$$

$$(B+\lambda\mathrm{I}) \text{ is positive semi-definite.} \tag{6c}$$

§4.3 Iterative Solution of the Sub-problem

Proof.

"$\Leftarrow$" Assume first that there is $\lambda \geqslant 0$ such that the conditions (6) are satisfied. By ① in the previous lemma, (6a) and (6c) imply that p_* is a global minimizer of the quadratic function

$$\widehat{m}(p) = f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p = m(p) + \frac{\lambda}{2}p^{\mathrm{T}}p. \tag{40}$$

This implies that $\widehat{m}(p) \geqslant \widehat{m}(p_*)$ for all $p \in \mathbb{R}^n$; thus

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}(p_*^{\mathrm{T}}p_* - p^{\mathrm{T}}p) .$$

Since $\lambda(\Delta - \|p_*\|) = 0$, we have $\lambda(\Delta^2 - p_*^{\mathrm{T}}p_*) = 0$.

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}(\Delta^2 - p^{\mathrm{T}}p).$$

Hence, from $\lambda \geqslant 0$, we have $m(p) \geqslant m(p_*)$ for all p with $\|p\| \leqslant \Delta$. Therefore, p_* is a global minimizer of (5). □

§4.3 Iterative Solution of the Sub-problem

Proof.

"$\Leftarrow$" Assume first that there is $\lambda \geqslant 0$ such that the conditions (6) are satisfied. By ① in the previous lemma, (6a) and (6c) imply that p_* is a global minimizer of the quadratic function

$$\widehat{m}(p) = f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} (B + \lambda \mathrm{I}) p = m(p) + \frac{\lambda}{2} p^{\mathrm{T}} p. \qquad (40)$$

This implies that $\widehat{m}(p) \geqslant \widehat{m}(p_*)$ for all $p \in \mathbb{R}^n$; thus

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2} \left(p_*^{\mathrm{T}} p_* - p^{\mathrm{T}} p \right).$$

Since $\lambda(\Delta - \|p_*\|) = 0$, we have $\lambda(\Delta^2 - p_*^{\mathrm{T}} p_*) = 0$.

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2} (\Delta^2 - p^{\mathrm{T}} p).$$

Hence, from $\lambda \geqslant 0$, we have $m(p) \geqslant m(p_*)$ for all p with $\|p\| \leqslant \Delta$. Therefore, p_* is a global minimizer of (5). □

§4.3 Iterative Solution of the Sub-problem

Proof.

"$\Leftarrow$" Assume first that there is $\lambda \geqslant 0$ such that the conditions (6) are satisfied. By ① in the previous lemma, (6a) and (6c) imply that p_* is a global minimizer of the quadratic function

$$\widehat{m}(p) = f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p = m(p) + \frac{\lambda}{2}p^{\mathrm{T}}p. \tag{40}$$

This implies that $\widehat{m}(p) \geqslant \widehat{m}(p_*)$ for all $p \in \mathbb{R}^n$; thus

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}(p_*^{\mathrm{T}}p_* - p^{\mathrm{T}}p)\,.$$

Since $\lambda(\Delta - \|p_*\|) = 0$, we have $\lambda(\Delta^2 - p_*^{\mathrm{T}}p_*) = 0$.

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}(\Delta^2 - p^{\mathrm{T}}p).$$

Hence, from $\lambda \geqslant 0$, we have $m(p) \geqslant m(p_*)$ for all p with $\|p\| \leqslant \Delta$. Therefore, p_* is a global minimizer of (5). □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

"$\Rightarrow$" For the converse, we assume that p_* is a global solution of (5) and show that there is a $\lambda \geqslant 0$ that satisfies (6). In the case $\|p_*\| < \Delta$, p_* is an unconstrained minimizer of m, and so

$$\nabla m(p_*) = Bp_* + g = 0\,, \quad \nabla^2 m(p_*) = B \text{ positive semi-definite,}$$

and so the properties (6) hold for $\lambda = 0$.

Assume for the remainder of the proof that $\|p_*\| = \Delta$. Then (6b) is immediately satisfied, and p_* also solves the constrained problem

$$\min m(p) \quad \text{subject to} \quad \|p\| = \Delta.$$

By the Lagrange multiplier theorem, there is a λ such that the Lagrangian function defined by $\mathcal{L}(p, \lambda) = m(p) + \frac{\lambda}{2}(p^{\mathrm{T}}p - \Delta^2)$ has a stationary point at p_*. $\square$

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

"$\Rightarrow$" For the converse, we assume that p_* is a global solution of (5) and show that there is a $\lambda \geqslant 0$ that satisfies (6). In the case $\|p_*\| < \Delta$, p_* is an unconstrained minimizer of m, and so

$$\nabla m(p_*) = Bp_* + g = 0\,, \quad \nabla^2 m(p_*) = B \text{ positive semi-definite,}$$

and so the properties (6) hold for $\lambda = 0$.

Assume for the remainder of the proof that $\|p_*\| = \Delta$. Then (6b) is immediately satisfied, and p_* also solves the constrained problem

$$\min m(p) \quad \text{subject to} \quad \|p\| = \Delta.$$

By the Lagrange multiplier theorem, there is a λ such that the Lagrangian function defined by $\mathcal{L}(p, \lambda) = m(p) + \frac{\lambda}{2}(p^{\mathrm{T}}p - \Delta^2)$ has a stationary point at p_*. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

By setting $\nabla_p \mathcal{L}(p_*, \lambda)$ to zero, we obtain

$$Bp_* + g + \lambda p_* = 0 \quad \Rightarrow \quad (B + \lambda \mathrm{I})p_* = -g\,, \tag{41}$$

so that (6a) holds. Since $m(p) \geqslant m(p_*)$ for any p with $p^{\mathrm{T}}p = p_*^{\mathrm{T}}p_* = \Delta^2$, we have for such vectors p that

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}(p_*^{\mathrm{T}}p_* - p^{\mathrm{T}}p)\,. \tag{42}$$

Using (6a), we obtain that

$$\begin{aligned}
&\frac{1}{2}(p - p_*)^{\mathrm{T}}(B + \lambda\mathrm{I})(p - p_*) \\
&= \frac{1}{2}p^{\mathrm{T}}(B + \lambda\mathrm{I})p + \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda\mathrm{I})p_* - p_*^{\mathrm{T}}(B + \lambda\mathrm{I})p \\
&= \frac{1}{2}p^{\mathrm{T}}(B + \lambda\mathrm{I})p - \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda\mathrm{I})p_* - g^{\mathrm{T}}p_* + g^{\mathrm{T}}p
\end{aligned}$$

□

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

By setting $\nabla_p \mathcal{L}(p_*, \lambda)$ to zero, we obtain

$$Bp_* + g + \lambda p_* = 0 \quad \Rightarrow \quad (B + \lambda \mathrm{I})p_* = -g\,, \tag{41}$$

so that (6a) holds. Since $m(p) \geqslant m(p_*)$ for any p with $p^{\mathrm{T}}p = p_*^{\mathrm{T}}p_* = \Delta^2$, we have for such vectors p that

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}\big(p_*^{\mathrm{T}}p_* - p^{\mathrm{T}}p\big). \tag{42}$$

Using (6a), we obtain that

$$\begin{aligned}
&\frac{1}{2}(p - p_*)^{\mathrm{T}}(B + \lambda \mathrm{I})(p - p_*) \\
&\quad = \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p + \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p_* - p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p \\
&\quad = \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p - \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p_* - g^{\mathrm{T}}p_* + g^{\mathrm{T}}p
\end{aligned}$$

□

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

By setting $\nabla_p \mathcal{L}(p_*, \lambda)$ to zero, we obtain

$$Bp_* + g + \lambda p_* = 0 \quad \Rightarrow \quad (B + \lambda \mathrm{I})p_* = -g\,, \tag{41}$$

so that (6a) holds. Since $m(p) \geqslant m(p_*)$ for any p with $p^{\mathrm{T}}p = p_*^{\mathrm{T}}p_* = \Delta^2$, we have for such vectors p that

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}\big(p_*^{\mathrm{T}}p_* - p^{\mathrm{T}}p\big). \tag{42}$$

Using (6a), we obtain that

$$\begin{aligned}
&\frac{1}{2}(p - p_*)^{\mathrm{T}}(B + \lambda \mathrm{I})(p - p_*) \\
&\quad = \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p + \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p_* - p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p \\
&\quad = \frac{1}{2}p^{\mathrm{T}}(B + \lambda \mathrm{I})p - \frac{1}{2}p_*^{\mathrm{T}}(B + \lambda \mathrm{I})p_* - g^{\mathrm{T}}p_* + g^{\mathrm{T}}p
\end{aligned}$$

□

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

so that (42) implies that

$$(p-p_*)^{\mathrm{T}}(B+\lambda \mathrm{I})(p-p_*) \geqslant 0 \quad \text{if } \|p\| = \|p_*\| = \Delta. \tag{43}$$

Since the set of directions

$$\left\{ w \,\middle|\, w = \pm \frac{p-p_*}{\|p-p_*\|} \text{ for some } p \text{ with } \|p\| = \Delta \right\}$$

is dense on the unit sphere, (43) suffices to prove (6c).

It remains to show that one of these Lagrange multipliers λ is non-negative. Because (6a) and (6c) are satisfied by p_*, we have from ① of the previous lemma that p_* minimizes

$$\widehat{m}(p) = f + g^{\mathrm{T}}p + \frac{1}{2}p^{\mathrm{T}}(B+\lambda \mathrm{I})p = m(p) + \frac{\lambda}{2}p^{\mathrm{T}}p, \tag{40}$$

so (42) holds **for all** $p \in \mathbb{R}^n$. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

so that (42) implies that

$$(p - p_*)^{\mathrm{T}}(B + \lambda \mathrm{I})(p - p_*) \geqslant 0 \quad \text{if } \|p\| = \|p_*\| = \Delta. \tag{43}$$

Since the set of directions

$$\left\{ w \,\middle|\, w = \pm \frac{p - p_*}{\|p - p_*\|} \text{ for some } p \text{ with } \|p\| = \Delta \right\}$$

is dense on the unit sphere, (43) suffices to prove (6c).

It remains to show that one of these Lagrange multipliers λ is non-negative. Because (6a) and (6c) are satisfied by p_*, we have from ① of the previous lemma that p_* minimizes

$$\widehat{m}(p) = f + g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}}(B + \lambda \mathrm{I}) p = m(p) + \frac{\lambda}{2} p^{\mathrm{T}} p, \tag{40}$$

so (42) holds **for all** $p \in \mathbb{R}^n$. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

Suppose that there are only negative values of λ that satisfy (6a) and (6c). Then we have from

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}\left(p_*^{\mathrm{T}} p_* - p^{\mathrm{T}} p\right). \tag{42}$$

that $m(p) \geqslant m(p_*)$ whenever $\|p\| \geqslant \|p_*\| = \Delta$. Since we already know that p_* minimizes m for $\|p\| \leqslant \Delta$, it follows that p_* is in fact a global, unconstrained minimizer of m. From ① of the previous lemma it follows that $Bp_* = -g$ and B is positive semi-definite. Therefore conditions (6a) and (6c) are satisfied by $\lambda = 0$, which contradicts our assumption that only negative values of λ can satisfy the conditions. We conclude that $\lambda \geqslant 0$, completing the proof. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

Suppose that there are only negative values of λ that satisfy (6a) and (6c). Then we have from

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}\big(p_*^{\mathrm{T}} p_* - p^{\mathrm{T}} p\big). \tag{42}$$

that $m(p) \geqslant m(p_*)$ whenever $\|p\| \geqslant \|p_*\| = \Delta$. Since we already know that p_* minimizes m for $\|p\| \leqslant \Delta$, it follows that p_* is in fact a global, unconstrained minimizer of m. From ① of the previous lemma it follows that $Bp_* = -g$ and B is positive semi-definite. Therefore conditions (6a) and (6c) are satisfied by $\lambda = 0$, which contradicts our assumption that only negative values of λ can satisfy the conditions. We conclude that $\lambda \geqslant 0$, completing the proof. □

§4.3 Iterative Solution of the Sub-problem

Proof (cont'd).

Suppose that there are only negative values of λ that satisfy (6a) and (6c). Then we have from

$$m(p) \geqslant m(p_*) + \frac{\lambda}{2}\left(p_*^{\mathrm{T}} p_* - p^{\mathrm{T}} p\right). \tag{42}$$

that $m(p) \geqslant m(p_*)$ whenever $\|p\| \geqslant \|p_*\| = \Delta$. Since we already know that p_* minimizes m for $\|p\| \leqslant \Delta$, it follows that p_* is in fact a global, unconstrained minimizer of m. From ① of the previous lemma it follows that $Bp_* = -g$ and B is positive semi-definite. Therefore conditions (6a) and (6c) are satisfied by $\lambda = 0$, which contradicts our assumption that only negative values of λ can satisfy the conditions. We conclude that $\lambda \geqslant 0$, completing the proof. □

§4.3 Iterative Solution of the Sub-problem

- **Convergence of Algorithm Based on Nearly Exact Solutions**

As we noted in the discussion of Algorithm 4.3 (which finds $\lambda^{(\ell)}$ by Newton's root-finding method), the loop to determine the optimal values of λ and p for the sub-problem (5) does not iterate till high accuracy is achieved. Instead, it is terminated after two or three iterations with a fairly loose approximation to the true solution. The inexactness in this approximate solution is measured in a different way from the dogleg and subspace minimization algorithms, in which, for the convergence of the sequence $\{g_k\}$, it requires that the sequence $\{p_k\}$ satisfies

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some positive constant c_1.

§4.3 Iterative Solution of the Sub-problem

- **Convergence of Algorithm Based on Nearly Exact Solutions**

As we noted in the discussion of Algorithm 4.3 (which finds $\lambda^{(\ell)}$ by Newton's root-finding method), the loop to determine the optimal values of λ and p for the sub-problem (5) does not iterate till high accuracy is achieved. Instead, it is terminated after two or three iterations with a fairly loose approximation to the true solution. The inexactness in this approximate solution is measured in a different way from the dogleg and subspace minimization algorithms, in which, for the convergence of the sequence $\{g_k\}$, it requires that the sequence $\{p_k\}$ satisfies

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

for some positive constant c_1.

§4.3 Iterative Solution of the Sub-problem

Instead, we require that

$$m(0) - m(p) \geqslant c_1 \left[m(0) - m(p_*) \right], \qquad (44a)$$

$$\|p\| \leqslant \gamma\Delta, \qquad (44b)$$

for some constants $c_1 \in (0,1]$ and $\gamma > 0$, where p_* is the exact solution of (3). The condition (44a) ensures that the approximate solution achieves a significant fraction of the maximum decrease possible in the model function m. Here we emphasize that it is not necessary to know p_*; there are practical termination criteria that imply (44a).

§4.3 Iterative Solution of the Sub-problem

Instead, we require that

$$m(0) - m(p) \geqslant c_1 \big[m(0) - m(p_*) \big] , \tag{44a}$$

$$\|p\| \leqslant \gamma\Delta , \tag{44b}$$

for some constants $c_1 \in (0,1]$ and $\gamma > 0$, where p_* is the exact solution of (3). The condition (44a) ensures that the approximate solution achieves a significant fraction of the maximum decrease possible in the model function m. Here we emphasize that it is not necessary to know p_*; there are practical termination criteria that imply (44a).

§4.3 Iterative Solution of the Sub-problem

One major difference between (44) and the earlier criterion (18) is that (44) makes better use of the second-order part of $m(\cdot)$; that is, the $p^{\mathrm{T}}Bp$ term. This difference is illustrated by the case in which $g = 0$ while B has negative eigenvalues, indicating that the current iterate x_k is a saddle point. Here, the right-hand side of (18) is zero (indeed, the algorithms we described earlier would terminate at such a point). The right-hand side of (44) is positive, indicating that decrease in the model function is still possible, so it forces the algorithm to move away from x_k.

§4.3 Iterative Solution of the Sub-problem

The close attention that near-exact algorithms pay to the second-order term is warranted only if this term closely reflects the actual behavior of the function f – in fact, the trust-region Newton method, for which $B = (\nabla^2 f)(x)$, is the only case that has been treated in the literature. For purposes of global convergence analysis, the use of the exact Hessian allows us to say more about the limit points of the algorithm than merely that they are stationary points. The following result shows that second-order necessary conditions ($\nabla^2 f(x_*)$ is positive semi-definite) are satisfied at the limit points.

§4.3 Iterative Solution of the Sub-problem

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ be continuous, $x_0 \in \mathbb{R}^n$ be given, S be the level set $\left\{x \,\middle|\, f(x) \leqslant f(x_0)\right\}$. Suppose that f is twice differentiable and bounded from below in S, and f is Lipschitz continuously differentiable in the neighborhood $S(R_0)$ for some $R_0 > 0$. Consider solving the minimization problem

$$\min_{p \in \mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad \text{s.t.} \quad \|p\| \leqslant \Delta_k \tag{3}$$

using Algorithm 4.1, where $B_k = (\nabla^2 f)(x_k)$, and all approximate solutions p_k of (3) satisfy the inequalities

$$m(0) - m(p_k) \geqslant c_1 \left[m(0) - m(p_k^*)\right], \tag{44a}$$

$$\|p_k\| \leqslant \gamma \Delta_k, \tag{44b}$$

where p_k^* is the exact minimizer of (3). Then $\lim\limits_{k \to \infty} \|g_k\| = 0$.

§4.3 Iterative Solution of the Sub-problem

Theorem (cont'd)

If, in addition, the level set S is compact, then either the algorithm terminates at a point x_k at which the second-order necessary conditions – $(\nabla^2 f)(x_k)$ is positive semi-definite – for a local solution hold, or else $\{x_k\}$ has a limit point x_* in S at which the second-order necessary conditions hold.

§4.4 Local Convergence of Trust-Region Newton Methods

Since global convergence of trust-region methods that use exact Hessians $(\nabla^2 f)(x)$ is established above, we turn our attention now to local convergence issues. The key to attaining the fast rate of convergence usually associated with Newton's method is to show that the trust-region bound eventually does not interfere as we approach a solution. Specifically, we hope that near the solution, the (approximate) solution of the trust-region sub-problem is well inside the trust region and becomes closer and closer to the true Newton step. Steps that satisfy the latter property are said to be **asymptotically similar** to the Newton steps.

§4.4 Local Convergence of Trust-Region Newton Methods

Since global convergence of trust-region methods that use exact Hessians $(\nabla^2 f)(x)$ is established above, we turn our attention now to local convergence issues. The key to attaining the fast rate of convergence usually associated with Newton's method is to show that the trust-region bound eventually does not interfere as we approach a solution. Specifically, we hope that near the solution, the (approximate) solution of the trust-region sub-problem is well inside the trust region and becomes closer and closer to the true Newton step. Steps that satisfy the latter property are said to be **asymptotically similar** to the Newton steps.

§4.4 Local Convergence of Trust-Region Newton Methods

We first prove a general result that applies to any algorithm of the form of Algorithm 4.1 that generates steps that are asymptotically similar to Newton steps whenever the Newton steps easily satisfy the trust-region bound. It shows that the trust-region constraint eventually becomes inactive in algorithms with this property and that superlinear convergence can be attained. The result assumes that the exact Hessian $B_k = (\nabla^2 f)(x_k)$ is used in (3) when x_k is close to a solution x_* that satisfies second-order sufficient conditions: $(\nabla^2 f)(x_*)$ is positive definite. Moreover, it assumes that the algorithm uses an approximate solution p_k of (3) that achieves a similar decrease in the model function m_k as the Cauchy point.

§4.4 Local Convergence of Trust-Region Newton Methods

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice Lipschitz continuously differentiable in the level set $S = \{x \,|\, f(x) \leqslant f(x_0)\}$, and the trust-region algorithm based on (3) with $B_k = (\nabla^2 f)(x_k)$ and Algorithm 4.1 chooses steps p_k that satisfy the Cauchy-point-based model reduction criterion

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

and are asymptotically similar to the Newton steps p_k^N in the sense

$$\|p_k - p_k^N\| = o(\|p_k^N\|)\,. \tag{46}$$

If the sequence $\{x_k\}$ converges to x_* at which the second-order sufficient conditions hold, then the trust-region bound Δ_k becomes inactive for all k sufficiently large, and the sequence $\{x_k\}$ converges superlinearly to x_*.

§4.4 Local Convergence of Trust-Region Newton Methods

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice Lipschitz continuously differentiable in the level set $S = \{x \mid f(x) \leqslant f(x_0)\}$, and the trust-region algorithm based on (3) with $B_k = (\nabla^2 f)(x_k)$ and Algorithm 4.1 chooses steps p_k that satisfy the Cauchy-point-based model reduction criterion

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

and are asymptotically similar to the Newton steps p_k^N in the sense

$$\|p_k - p_k^N\| = o(\|p_k^N\|) \,. \tag{46}$$

If the sequence $\{x_k\}$ converges to x_* at which the second-order sufficient conditions hold, then the trust-region bound Δ_k becomes inactive for all k sufficiently large, and the sequence $\{x_k\}$ converges superlinearly to x_*.

§4.4 Local Convergence of Trust-Region Newton Methods

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice Lipschitz continuously differentiable in the level set $S = \{x \mid f(x) \leqslant f(x_0)\}$, and the trust-region algorithm based on (3) with $B_k = (\nabla^2 f)(x_k)$ and Algorithm 4.1 chooses steps p_k that satisfy the Cauchy-point-based model reduction criterion

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right) \tag{18}$$

and are asymptotically similar to the Newton steps $p_k^{\rm N}$ in the sense

$$\|p_k - p_k^{\rm N}\| = o(\|p_k^{\rm N}\|) \,. \tag{46}$$

If the sequence $\{x_k\}$ converges to x_* at which the gradient vanishes and the Hessian is positive definite, then the trust-region bound Δ_k becomes inactive for all k sufficiently large, and the sequence $\{x_k\}$ converges superlinearly to x_*.

§4.4 Local Convergence of Trust-Region Newton Methods

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice Lipschitz continuously differentiable in the level set $S = \{x \,|\, f(x) \leqslant f(x_0)\}$, and the trust-region algorithm based on (3) with $B_k = (\nabla^2 f)(x_k)$ and Algorithm 4.1 chooses steps p_k that satisfy the Cauchy-point-based model reduction criterion

$$m_k(0) - m_k(p_k) \geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \tag{18}$$

and are asymptotically similar to the Newton steps $p_k^{\rm N}$ in the sense

$$\|p_k - p_k^{\rm N}\| = o(\|p_k^{\rm N}\|)\,. \tag{46}$$

If the sequence $\{x_k\}$ converges to x_* at which the gradient vanishes and the Hessian is positive definite, then the trust-region bound Δ_k becomes inactive for all k sufficiently large, and the sequence $\{x_k\}$ converges superlinearly to x_*.

§4.4 Local Convergence of Trust-Region Newton Methods

Proof.

We first show that the unconstrained minimizer p_k^{N}, the Newton direction, satisfies $\|p_k^{\mathrm{N}}\| \leqslant \frac{1}{2}\Delta_k$ for all sufficiently large k.

To begin with, we look for a lower bound on the predicted reduction $m_k(0) - m_k(p_k)$ for all sufficiently large k. We assume that k is large enough that the $o(\|p_k^{\mathrm{N}}\|)$ term in (46) is less than $\|p_k^{\mathrm{N}}\|$.

1. If $\|p_k^{\mathrm{N}}\| \leqslant \Delta_k/2$, we have $\|p_k\| \leqslant \|p_k^{\mathrm{N}}\| + o(\|p_k^{\mathrm{N}}\|) \leqslant 2\|p_k^{\mathrm{N}}\|$.
2. If $\|p_k^{\mathrm{N}}\| > \Delta_k/2$, we have $\|p_k\| \leqslant \Delta_k < 2\|p_k^{\mathrm{N}}\|$.

In both cases, then, we have

$$\|p_k\| \leqslant 2\|p_k^{\mathrm{N}}\| \leqslant 2\|B_k^{-1}\|\|g_k\|,$$

and so $\|g_k\| \geqslant \frac{1}{2}\|p_k\|/\|B_k^{-1}\|$ for k sufficiently large. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof.

We first show that the unconstrained minimizer p_k^{N}, the Newton direction, satisfies $\|p_k^{\mathrm{N}}\| \leqslant \frac{1}{2}\Delta_k$ for all sufficiently large k.

To begin with, we look for a lower bound on the predicted reduction $m_k(0) - m_k(p_k)$ for all sufficiently large k. We assume that k is large enough that the $o(\|p_k^{\mathrm{N}}\|)$ term in (46) is less than $\|p_k^{\mathrm{N}}\|$.

1. If $\|p_k^{\mathrm{N}}\| \leqslant \Delta_k/2$, we have $\|p_k\| \leqslant \|p_k^{\mathrm{N}}\| + o(\|p_k^{\mathrm{N}}\|) \leqslant 2\|p_k^{\mathrm{N}}\|$.
2. If $\|p_k^{\mathrm{N}}\| > \Delta_k/2$, we have $\|p_k\| \leqslant \Delta_k < 2\|p_k^{\mathrm{N}}\|$.

In both cases, then, we have

$$\|p_k\| \leqslant 2\|p_k^{\mathrm{N}}\| \leqslant 2\|B_k^{-1}\|\|g_k\|,$$

and so $\|g_k\| \geqslant \frac{1}{2}\|p_k\|/\|B_k^{-1}\|$ for k sufficiently large. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

Using (18), we find that for k sufficiently large,

$$\begin{aligned} m_k(0) - m_k(p_k) &\geqslant c_1 \|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right) \\ &\geqslant c_1 \frac{\|p_k\|}{2\|B_k^{-1}\|} \min\left(\|p_k\|, \frac{\|p_k\|}{2\|B_k\|\|B_k^{-1}\|}\right) \\ &= c_1 \frac{\|p_k\|^2}{4\|B_k^{-1}\|^2\|B_k\|} \, . \end{aligned}$$

Since $x_k \to x_*$, by the continuity of $(\nabla^2 f)(x)$ and the positive definiteness of $B_* \equiv (\nabla^2 f)(x_*)$ we deduce that the following bound holds for all k sufficiently large:

$$\frac{c_1}{4\|B_k^{-1}\|^2\|B_k\|} \geqslant \frac{c_1}{8\|B_*^{-1}\|^2\|B_*\|} \equiv c_3,$$

where $c_3 > 0$. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

Using (18), we find that for k sufficiently large,

$$\begin{aligned} m_k(0) - m_k(p_k) &\geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \\ &\geqslant c_1 \frac{\|p_k\|}{2\|B_k^{-1}\|} \min\Big(\|p_k\|, \frac{\|p_k\|}{2\|B_k\|\|B_k^{-1}\|}\Big) \\ &= c_1 \frac{\|p_k\|^2}{4\|B_k^{-1}\|^2\|B_k\|}\,. \end{aligned}$$

Since $x_k \to x_*$, by the continuity of $(\nabla^2 f)(x)$ and the positive definiteness of $B_* \equiv (\nabla^2 f)(x_*)$ we deduce that the following bound holds for all k sufficiently large:

$$\frac{c_1}{4\|B_k^{-1}\|^2\|B_k\|} \geqslant \frac{c_1}{8\|B_*^{-1}\|^2\|B_*\|} \equiv c_3,$$

where $c_3 > 0$. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

Using (18), we find that for k sufficiently large,

$$\begin{aligned} m_k(0) - m_k(p_k) &\geqslant c_1 \|g_k\| \min\Big(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\Big) \\ &\geqslant c_1 \frac{\|p_k\|}{2\|B_k^{-1}\|} \min\Big(\|p_k\|, \frac{\|p_k\|}{2\|B_k\|\|B_k^{-1}\|}\Big) \\ &= c_1 \frac{\|p_k\|^2}{4\|B_k^{-1}\|^2\|B_k\|} \geqslant c_3\|p_k\|^2 . \end{aligned}$$

Since $x_k \to x_*$, by the continuity of $(\nabla^2 f)(x)$ and the positive definiteness of $B_* \equiv (\nabla^2 f)(x_*)$ we deduce that the following bound holds for all k sufficiently large:

$$\frac{c_1}{4\|B_k^{-1}\|^2\|B_k\|} \geqslant \frac{c_1}{8\|B_*^{-1}\|^2\|B_*\|} \equiv c_3,$$

where $c_3 > 0$. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

By Lipschitz continuity of $(\nabla^2 f)(x)$ near x_*,

$$\Big|\big[f(x_k) - f(x_k + p_k)\big] - \big[m_k(0) - m_k(p_k)\big]\Big|$$
$$= \Big|\frac{1}{2}p_k^{\mathrm{T}} B_k p_k - \frac{1}{2}p_k^{\mathrm{T}}(\nabla^2 f)(x_k + tp_k)p_k\Big| \leqslant \frac{L}{2}\|p_k\|^3,$$

where $L > 0$ is the Lipschitz constant for $\nabla^2 f$. Hence, by the definition of ρ_k, we have for sufficiently large k that

$$|\rho_k - 1| \leqslant \frac{\|p_k\|^3(L/2)}{c_3\|p_k\|^2} = \frac{L}{2c_3}\|p_k\| \leqslant \frac{L}{2c_3}\Delta_k\,. \tag{47}$$

Now, the trust-region radius can be reduced only if $\rho_k < 1/4$ (or some other fixed number less than 1), so it is clear from (47) that the sequence $\{\Delta_k\}$ is bounded away from zero. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

By Lipschitz continuity of $(\nabla^2 f)(x)$ near x_*,

$$\Big|\big[f(x_k) - f(x_k + p_k)\big] - \big[m_k(0) - m_k(p_k)\big]\Big|$$
$$= \left|\frac{1}{2}p_k^{\mathrm{T}} B_k p_k - \frac{1}{2}p_k^{\mathrm{T}}(\nabla^2 f)(x_k + tp_k)p_k\right| \leqslant \frac{L}{2}\|p_k\|^3,$$

where $L > 0$ is the Lipschitz constant for $\nabla^2 f$. Hence, by the definition of ρ_k, we have for sufficiently large k that

$$|\rho_k - 1| \leqslant \frac{\|p_k\|^3(L/2)}{c_3\|p_k\|^2} = \frac{L}{2c_3}\|p_k\| \leqslant \frac{L}{2c_3}\Delta_k. \tag{47}$$

Now, the trust-region radius can be reduced only if $\rho_k < 1/4$ (or some other fixed number less than 1), so it is clear from (47) that the sequence $\{\Delta_k\}$ is bounded away from zero. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

By Lipschitz continuity of $(\nabla^2 f)(x)$ near x_*,

$$\Big|\big[f(x_k) - f(x_k + p_k)\big] - \big[m_k(0) - m_k(p_k)\big]\Big|$$
$$= \Big|\frac{1}{2}p_k^{\mathrm{T}} B_k p_k - \frac{1}{2}p_k^{\mathrm{T}}(\nabla^2 f)(x_k + tp_k)p_k\Big| \leqslant \frac{L}{2}\|p_k\|^3,$$

where $L > 0$ is the Lipschitz constant for $\nabla^2 f$. Hence, by the definition of ρ_k, we have for sufficiently large k that

$$|\rho_k - 1| \leqslant \frac{\|p_k\|^3(L/2)}{c_3\|p_k\|^2} = \frac{L}{2c_3}\|p_k\| \leqslant \frac{L}{2c_3}\Delta_k\,. \qquad (47)$$

Now, the trust-region radius can be reduced only if $\rho_k < 1/4$ (or some other fixed number less than 1), so it is clear from (47) that the sequence $\{\Delta_k\}$ is bounded away from zero. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

Since $x_k \to x_*$, we have $\|p_k^N\| \to 0$ and therefore $\|p_k\| \to 0$ from (46). Hence, the trust-region bound is inactive for all k sufficiently large, and the bound $\|p_k^N\| \leqslant \frac{1}{2}\Delta_k$ is eventually always satisfied.

To prove superlinear convergence, we use the quadratic convergence of Newton's method to conclude that

$$\|x_k + p_k^N - x_*\| = \mathcal{O}\big(\|x_k - x_*\|^2\big) .$$

This implies that $\|p_k^N\| = \mathcal{O}(\|x_k - x_*\|)$. Therefore, using (46),

$$\begin{aligned}\|x_k + p_k - x_*\| &\leqslant \|x_k + p_k^N - x_*\| + \|p_k^N - p_k\| \\ &= \mathcal{O}\big(\|x_k - x_*\|^2\big) + o(\|p_k^N\|) = o\big(\|x_k - x_*\|\big) ,\end{aligned}$$

thus proving superlinear convergence. □

§4.4 Local Convergence of Trust-Region Newton Methods

Proof (cont'd).

Since $x_k \to x_*$, we have $\|p_k^{\mathrm{N}}\| \to 0$ and therefore $\|p_k\| \to 0$ from (46). Hence, the trust-region bound is inactive for all k sufficiently large, and the bound $\|p_k^{\mathrm{N}}\| \leqslant \frac{1}{2}\Delta_k$ is eventually always satisfied.

To prove superlinear convergence, we use the quadratic convergence of Newton's method to conclude that

$$\|x_k + p_k^{\mathrm{N}} - x_*\| = \mathcal{O}\big(\|x_k - x_*\|^2\big) .$$

This implies that $\|p_k^{\mathrm{N}}\| = \mathcal{O}(\|x_k - x_*\|)$. Therefore, using (46),

$$\begin{aligned}\|x_k + p_k - x_*\| &\leqslant \|x_k + p_k^{\mathrm{N}} - x_*\| + \|p_k^{\mathrm{N}} - p_k\| \\ &= \mathcal{O}\big(\|x_k - x_*\|^2\big) + o(\|p_k^{\mathrm{N}}\|) = o\big(\|x_k - x_*\|\big) ,\end{aligned}$$

thus proving superlinear convergence. □

§4.4 Local Convergence of Trust-Region Newton Methods

Reasonable implementations of the dogleg, subspace minimization, and nearly-exact algorithm of Section 4.3 with $B_k = (\nabla^2 f)(x_k)$ eventually use the steps $p_k = p_k^{\rm N}$ under the conditions of the theorem just established, and therefore converge **quadratically**. In the case of the dogleg and two-dimensional subspace minimization methods, the exact step $p_k^{\rm N}$ is one of the candidates for p_k – it lies inside the trust region, along the dogleg path, and inside the two-dimensional subspace. Since under the assumptions of the theorem, $p_k^{\rm N}$ is the unconstrained minimizer of m_k for k sufficiently large, it is certainly the minimizer in the more restricted domains, so we have $p_k = p_k^{\rm N}$. For the approach of Section 4.3, if we follow the reasonable strategy of checking whether $p_k^{\rm N}$ is a solution of (3) prior to embarking on Algorithm 4.3, then eventually we will also have $p_k = p_k^{\rm N}$ also.

§4.4 Local Convergence of Trust-Region Newton Methods

Reasonable implementations of the dogleg, subspace minimization, and nearly-exact algorithm of Section 4.3 with $B_k = (\nabla^2 f)(x_k)$ eventually use the steps $p_k = p_k^{\rm N}$ under the conditions of the theorem just established, and therefore converge **quadratically**. In the case of the dogleg and two-dimensional subspace minimization methods, the exact step $p_k^{\rm N}$ is one of the candidates for p_k – it lies inside the trust region, along the dogleg path, and inside the two-dimensional subspace. Since under the assumptions of the theorem, $p_k^{\rm N}$ is the unconstrained minimizer of m_k for k sufficiently large, it is certainly the minimizer in the more restricted domains, so we have $p_k = p_k^{\rm N}$.

For the approach of Section 4.3, if we follow the reasonable strategy of checking whether $p_k^{\rm N}$ is a solution of (3) prior to embarking on Algorithm 4.3, then eventually we will also have $p_k = p_k^{\rm N}$ also.

§4.4 Local Convergence of Trust-Region Newton Methods

Reasonable implementations of the dogleg, subspace minimization, and nearly-exact algorithm of Section 4.3 with $B_k = (\nabla^2 f)(x_k)$ eventually use the steps $p_k = p_k^{\mathrm{N}}$ under the conditions of the theorem just established, and therefore converge **quadratically**. In the case of the dogleg and two-dimensional subspace minimization methods, the exact step p_k^{N} is one of the candidates for p_k – it lies inside the trust region, along the dogleg path, and inside the two-dimensional subspace. Since under the assumptions of the theorem, p_k^{N} is the unconstrained minimizer of m_k for k sufficiently large, it is certainly the minimizer in the more restricted domains, so we have $p_k = p_k^{\mathrm{N}}$. For the approach of Section 4.3, if we follow the reasonable strategy of checking whether p_k^{N} is a solution of (3) prior to embarking on Algorithm 4.3, then eventually we will also have $p_k = p_k^{\mathrm{N}}$ also.

§4.5 Other Enhancements

- **Scaling**

As we noted in Chapter 2, optimization problems are often posed with poor scaling – the objective function f is highly sensitive to small changes in certain components of the vector x and relatively insensitive to changes in other components. Topologically, a symptom of poor scaling is that the minimizer x_* lies in a narrow valley, so that the contours of the objective $f(\cdot)$ near x_* tend towards highly eccentric ellipses. Algorithms that fail to compensate for poor scaling can perform badly; see Figure 8 (in the next slide) for an illustration of the poor performance of the steepest descent approach.

§4.5 Other Enhancements

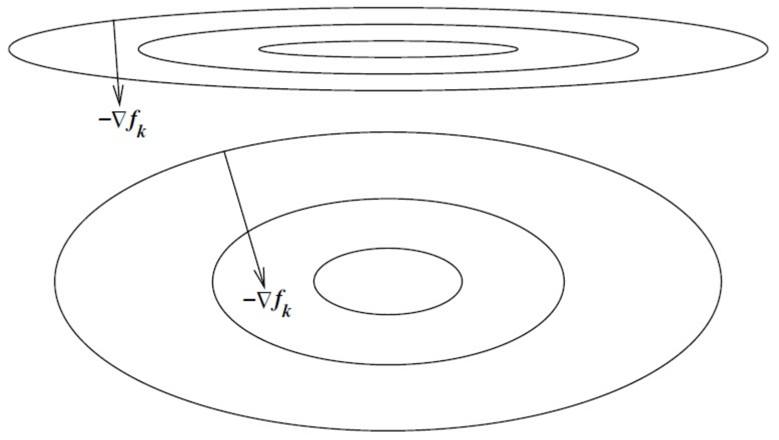

Figure 8: Poorly scaled and well scaled problems, and performance of the steepest descent direction.

§4.5 Other Enhancements

Recalling our definition of a trust region – a region around the current iterate within which the model $m_k(\cdot)$ is an adequate representation of the true objective $f(\cdot)$ – it is easy to see that a spherical trust region may not be appropriate when f is poorly scaled. Even if the model Hessian B_k is exact, the rapid changes in f along certain directions probably will cause m_k to be a poor approximation to f along these directions. On the other hand, m_k may be a more reliable approximation to f along directions in which f is changing more slowly. Since the shape of our trust region should be such that our confidence in the model is more or less the same at all points on the boundary of the region, we are led naturally to consider elliptical trust regions in which the axes are short in the sensitive directions and longer in the less sensitive directions.

§4.5 Other Enhancements

Elliptical trust regions can be defined by

$$\|Dp\| \leqslant \Delta\,, \tag{48}$$

where D is a diagonal matrix with positive diagonal elements, yielding the following scaled trust-region sub-problem:

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad \text{s.t.} \quad \|Dp\| \leqslant \Delta_k\,. \tag{49}$$

When $f(x)$ is highly sensitive to the value of the i-th component x_i, we set the corresponding diagonal element d_{ii} of D to be large, while d_{ii} is smaller for less-sensitive components.

§4.5 Other Enhancements

Elliptical trust regions can be defined by

$$\|Dp\| \leqslant \Delta\,, \tag{48}$$

where D is a diagonal matrix with positive diagonal elements, yielding the following scaled trust-region sub-problem:

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \quad \text{s.t.} \quad \|Dp\| \leqslant \Delta_k\,. \tag{49}$$

When $f(x)$ is highly sensitive to the value of the i-th component x_i, we set the corresponding diagonal element d_{ii} of D to be large, while d_{ii} is smaller for less-sensitive components.

§4.5 Other Enhancements

Information to construct the scaling matrix D may be derived from the second derivatives $\dfrac{\partial^2 f}{\partial x_i^2}$. We can allow D to change from iteration to iteration; most of the theory of this chapter will still apply with minor modifications provided that each d_{ii} stays within some predetermined range $[d_{\rm lo}, d_{\rm hi}]$, where $0 < d_{\rm lo} \leqslant d_{\rm hi} < \infty$. Of course, we do not need D to be a precise reflection of the scaling of the problem, so it is not necessary to devise elaborate heuristics or to perform extensive computations to get it just right.

§4.5 Other Enhancements

Algorithm 4.4 (Generalized Cauchy Point).

Find the vector p_k^s that solves

$$p_k^s = \arg\min_{p\in\mathbb{R}^n} \left(f_k + g_k^{\mathrm{T}} p\right) \qquad \text{s.t.} \quad \|Dp\| \leqslant \Delta_k ; \tag{50}$$

Calculate the scalar $\tau_k > 0$ that minimizes $m_k(\tau p_k^s)$ subject to satisfying the trust-region bound; that is,

$$\begin{aligned} \tau_k &= \arg\min_{\tau>0} m_k(\tau p_k^s) \quad \text{s.t.} \quad \|\tau D p_k^s\| \leqslant \Delta_k ; \\ p_k^C &= \tau_k p_k^s . \end{aligned} \tag{51}$$

For this scaled version, we find that

$$p_k^s = -\frac{\Delta_k}{\|D^{-1} g_k\|} D^{-2} g_k , \tag{52}$$

$$\tau_k = \begin{cases} 1 & \text{if } g_k^{\mathrm{T}} D^{-2} B_k D^{-2} g_k \leqslant 0, \\ \min\left(\dfrac{\|D^{-1} g_k\|^3}{\Delta_k g_k^{\mathrm{T}} D^{-2} B_k D^{-2} g_k}, 1\right) & \text{otherwise.} \end{cases} \tag{53}$$

§4.5 Other Enhancements

Algorithm 4.4 (Generalized Cauchy Point).

Find the vector p_k^s that solves

$$p_k^s = \arg\min_{p\in\mathbb{R}^n} \left(f_k + g_k^{\mathrm{T}} p\right) \qquad \text{s.t.} \quad \|Dp\| \leqslant \Delta_k\,; \tag{50}$$

Calculate the scalar $\tau_k > 0$ that minimizes $m_k(\tau p_k^s)$ subject to satisfying the trust-region bound; that is,

$$\begin{aligned} \tau_k &= \arg\min_{\tau>0} m_k(\tau p_k^s) \quad \text{s.t.} \quad \|\tau D p_k^s\| \leqslant \Delta_k\,; \\ p_k^C &= \tau_k p_k^s\,. \end{aligned} \tag{51}$$

For this scaled version, we find that

$$p_k^s = -\frac{\Delta_k}{\|D^{-1}g_k\|} D^{-2} g_k\,, \tag{52}$$

$$\tau_k = \begin{cases} 1 & \text{if } g_k^{\mathrm{T}} D^{-2} B_k D^{-2} g_k \leqslant 0, \\ \min\left(\dfrac{\|D^{-1}g_k\|^3}{\Delta_k g_k^{\mathrm{T}} D^{-2} B_k D^{-2} g_k}, 1\right) & \text{otherwise.} \end{cases} \tag{53}$$

§4.5 Other Enhancements

A simpler alternative for adjusting the definition of the Cauchy point and the various algorithms of this chapter to allow for the elliptical trust region is simply to rescale the variables p in the sub-problem (49) so that the trust region is spherical in the scaled variables. By defining $\widetilde{p} = Dp$ and by substituting into (49), we obtain that

$$\min_{\widetilde{p}\in\mathbb{R}^n} \widetilde{m}_k(\widetilde{p}) = f_k + g_k^{\mathrm{T}} D^{-1}\widetilde{p} + \frac{1}{2}\widetilde{p}^{\mathrm{T}} D^{-1} B_k D^{-1}\widetilde{p} \quad \text{s.t.} \quad \|\widetilde{p}\| \leqslant \Delta_k\,.$$

The theory and algorithms can now be derived in the usual way by substituting $\widetilde{p}$ for p, $D^{-1}g_k$ for g_k, $D^{-1}B_kD^{-1}$ for B_k and so on.

§4.5 Other Enhancements

- **Trust Region in Other Norms**

Trust regions may also be defined in terms of norms such as

$$\|p\|_1 \leqslant \Delta_k \quad \text{or} \quad \|p\|_\infty \leqslant \Delta_k \,,$$

or their scaled counterparts

$$\|Dp\|_1 \leqslant \Delta_k \quad \text{or} \quad \|Dp\|_\infty \leqslant \Delta_k \,,$$

where D is a positive diagonal matrix as before. Norms such as these offer no obvious advantages for small-medium unconstrained problems, but they may be useful for constrained problems. For instance, for the bound-constrained problem

$$\min_{p\in\mathbb{R}^n} f(x) \quad \text{subject to} \quad x \geqslant 0 \,,$$

the trust-region sub-problem may take the form

$$\min_{p\in\mathbb{R}^n} m_k(p) = f_k + g_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \;\; s.t. \;\; x_k + p \geqslant 0 \,, \|p\| \leqslant \Delta_k \,. \quad (54)$$

§4.5 Other Enhancements

When the trust region is defined by a Euclidean norm, the feasible region for (54) consists of the intersection of a sphere and the non-negative orthant – an awkward object, geometrically speaking. When the ∞-norm is used, however, the feasible region is simply the rectangular box defined by

$$x_k + p \geqslant 0\,, \quad p \geqslant -\Delta_k \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad p \leqslant \Delta_k \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix},$$

so the solution of the sub-problem is easily calculated by using techniques for bound-constrained quadratic programming.

§4.5 Other Enhancements

For large problems, in which factorization or formation the model Hessian B_k is not computationally desirable, the use of a trust region defined by $\|\cdot\|_\infty$ will also give rise to a bound-constrained sub-problem, which may be more convenient to solve than the standard sub-problem (3). To our knowledge, there has not been much research on the relative performance of methods that use trust regions of different shapes on large problems.