Numerical Optimization

Lecture Notes #26
Nonlinear Equations: Practical Methods

Peter Blomgren,
⟨blomgren.peter@gmail.com⟩

Department of Mathematics and Statistics
Dynamical Systems Group
Computational Sciences Research Center
San Diego State University
San Diego, CA 92182-7720

http://terminus.sdsu.edu/

Fall 2018

Outline

1. Nonlinear Equations...
   - Breakdown of Global Convergence: Blowup / Cycling
   - Toward Increased Robustness
2. Practical Line Search Methods
   - Convergence
   - Algorithm
   - Convergence Rate
3. Practical Trust-Region Methods
   - Fundamentals
   - Algorithm
   - Convergence

San Diego State University

Breakdown of Global Convergence

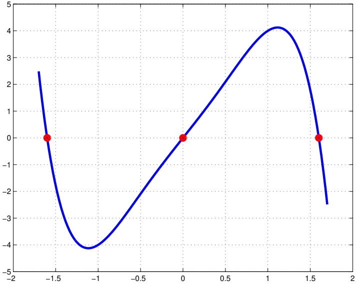
As we have seen, both **Newton's** and **Broyden's method** with unit step $\alpha_k \equiv 1$, must be started "close enough" to the solution $\bar{\mathbf{r}}(\bar{\mathbf{x}}^*) = 0$ in order to converge.

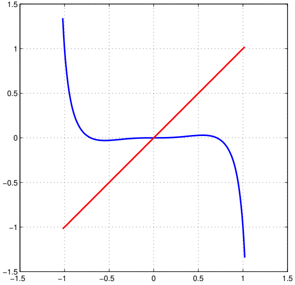
Broyden's method also requires the more restrictive $\|B_0 - J(\bar{\mathbf{x}}^*)\| \leq \epsilon$.

- When started too far away from the solution, components of the unknowns $\bar{\mathbf{x}}_k$, or function vector $\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)$, or the Jacobian $J(\bar{\mathbf{x}}_k)$ may blow up; — this sort of breakdown is easy to **identify**. (But not necessarily easy to fix...)
- A type of breakdown that is not as easy to detect is **cycling**, where the sequence of iterates $\{\bar{\mathbf{x}}_k\}$ repeat, *i.e.* $\bar{\mathbf{x}}_{k+m} = \bar{\mathbf{x}}_k$, for some $m \geq 1$. Clearly, the larger m is, the harder it is to detect cycling (especially in finite precision, where we have $\bar{\mathbf{x}}_{k+m} \approx \bar{\mathbf{x}}_k$).

SAN DIEGO STATE UNIVERSITY

Example: Cycling





The function $r(x) = -x^5 + x^3 + 4x$ has three non-degenerate real roots. Since the roots are non-degenerate, we expect the fixed point iteration defined by the Newton iteration

$$x \leftarrow x - \frac{f(x)}{f'(x)} = x - \frac{-x^5 + x^3 + 4x}{-5x^4 + 3x^2 + 4}$$

to converge quadratically.

Example: Cycling

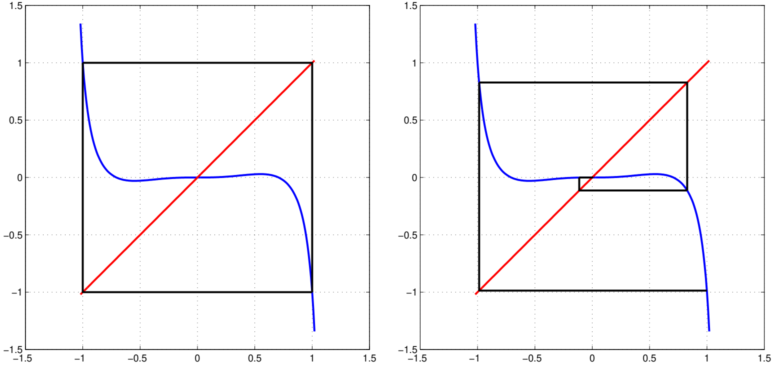
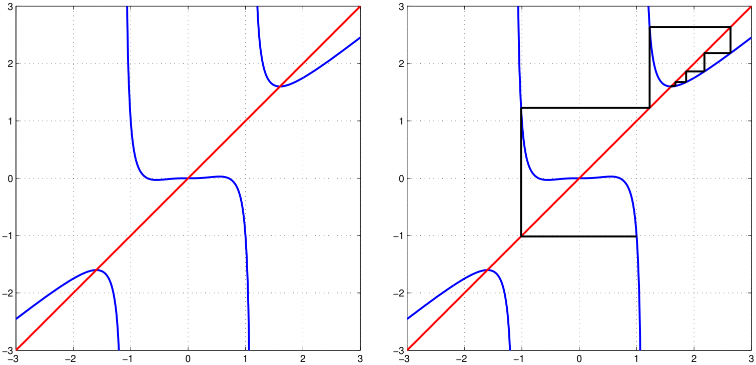


Figure: If we start the iteration in $x_0 = 1$, then the Newton iteration cycles $\{1, -1, 1, -1, \dots\}$ (left figure). On the right we see the rapid convergence to the root $x^* = 0$, for the iteration started at $x_0 = 0.999$.

SAN DIEGO STATE UNIVERSITY

Example: Cycling



If we start the iteration in $x_0 = 1.001$, then the Newton iteration escapes out to the root $1.6004\ldots$.

Cycling is quite an exotic occurrence.

San Diego State University

Sidenote: Convergence of Newton's Method can be Surprisingly Complex

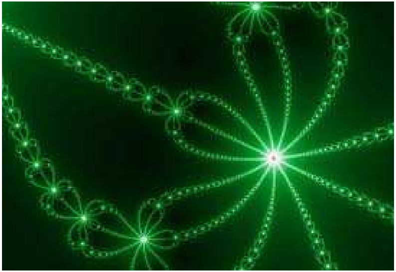


Figure: A Newton 4th power fractal. **Credit:** `fractalfoundation.org`, Annamarie M.

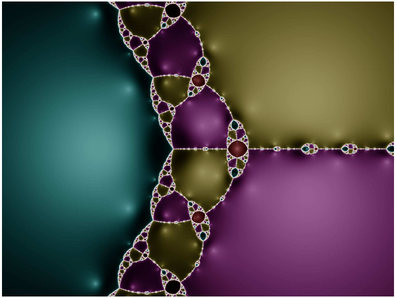


Figure: The Juila set (in white) for Newton's method applied to $f(z) = z^3 - 2z + 2$. Start values in the cyan, pink, yellow shaded regions converge to one of the three zeros of $f(z)$. Values from the red/black regions do not converge, they are attracted by a cycle of period 2. **Credit:** Wikimedia commons.

Increased Robustness

We can make both Newton's and Broyden's method more robust by using them in a line-search or trust-region framework.

However, in order to use these frameworks, we must define a scalar-valued **merit function** with which we measure progress toward the solution.

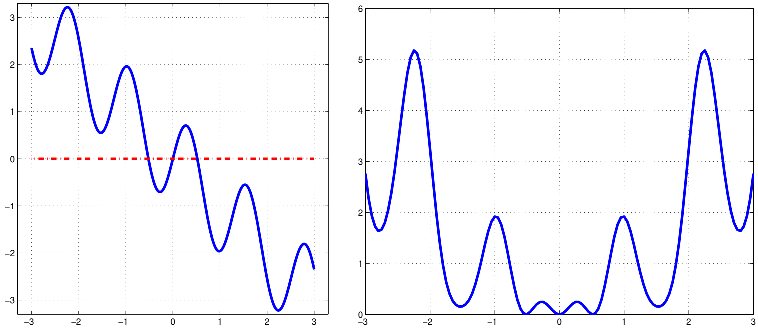
The most widely used merit function is the sum-of-squares,

$$f(\bar{\mathbf{x}}) = \frac{1}{2}\|\bar{\mathbf{r}}(\bar{\mathbf{x}})\|^2 = \frac{1}{2}\sum_{i=1}^{n} r_i^2(\bar{\mathbf{x}}).$$

Root of $\bar{\mathbf{r}}(\bar{\mathbf{x}}) = 0 \Rightarrow$ Local minimizer of $f(\bar{\mathbf{x}})$.

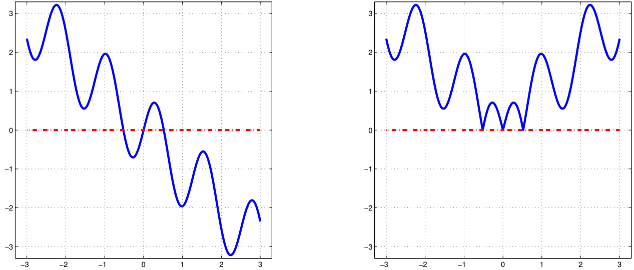
Local minimizer of $f(\bar{\mathbf{x}}) \not\Rightarrow$ Root of $\bar{\mathbf{r}}(\bar{\mathbf{x}}) = 0$.

Roots and Local Minima



Consider the non-linear function $r(x) = \sin(5x) - x$ (pictured to the left) and the associated sum-of-squares objective $f(x) = \frac{1}{2}(\sin(5x) - x)^2$ (pictured to the right). In this range we have gone from three roots, to seven local minima.

Other Merit Functions — The l_1-norm



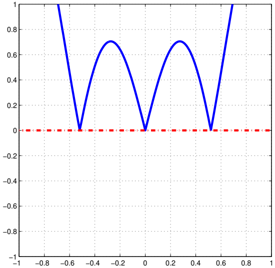


Figure: The l_1-norm (sum-of-absolute-values) gives us an alternative objective, with its own problems — the derivative does not exist in the optimal points...

Practical Line Search Methods

We can build algorithms with global convergence properties by applying the line search approach to the sum-of-squares merit function $f(\bar{\mathbf{x}}) = \frac{1}{2}\|\bar{\mathbf{r}}(\bar{\mathbf{x}})\|^2$.

Note: Convergence is global in the sense that we guarantee convergence to a stationary point for $f(\bar{\mathbf{x}})$, *i.e.* a point $\bar{\mathbf{x}}^*$ such that $\nabla f(\bar{\mathbf{x}}^*) = 0$.

From a point $\bar{\mathbf{x}}_k$, the search direction $\bar{\mathbf{p}}_k$ must be a descent direction for $f(\bar{\mathbf{x}})$, *i.e.*

$$\cos\theta_k = \frac{-\bar{\mathbf{p}}_k^T \nabla f(\bar{\mathbf{x}}_k)}{\|\bar{\mathbf{p}}_k^T\| \, \|\nabla f(\bar{\mathbf{x}}_k)\|} > 0.$$

Then we use a line search procedure to identify a step α_k, satisfying *e.g.* the **Wolfe conditions**.

San Diego State University

Practical Line Search Methods — Convergence

Theorem

Suppose that $J(\bar{\mathbf{x}})$ is Lipschitz continuous in a neighborhood $\mathcal{D}$ of the level set $\mathcal{L}(\bar{\mathbf{x}}_0) = \{\bar{\mathbf{x}} \in \mathbb{R}^n \,:\, f(\bar{\mathbf{x}}) \leq f(\bar{\mathbf{x}}_0)\}$. Suppose that a line search algorithm is applied and that the search directions $\bar{\mathbf{p}}_k$ satisfy $\cos\theta_k > 0$, and the step lengths α_k satisfy the Wolfe conditions. Then the **Zoutendijk condition** *holds, i.e.*

$$\sum_{k=0}^{\infty} \cos^2\theta_k \|J_k^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 < \infty$$

As long as we can bound $\cos\theta_k \geq \delta > 0$, this guarantees that $\|J_k^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \to 0$.

Further, if $\|J(\bar{\mathbf{x}})^{-1}\|$ is bounded on $\mathcal{D}$, then $\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) \to 0$.

San Diego State University

Practical Line Search Methods

Newton / inexact Newton

We take a look at the search directions generated by Newton and inexact Newton line-search methods — is the condition $\cos\theta_k \geq \delta > 0$ satisfied???

When the Newton-step is well defined, it is a descent direction for $f(\cdot)$ whenever $\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) \neq 0$, since

$$\bar{\mathbf{p}}_k^T \nabla f(\bar{\mathbf{x}}_k) = -\bar{\mathbf{p}}_k^T J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k) = -\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 < 0,$$

and we have

$$\begin{aligned} \cos\theta_k &= -\frac{\bar{\mathbf{p}}_k^T \nabla f(\bar{\mathbf{x}}_k)}{\|\bar{\mathbf{p}}_k^T\| \, \|\nabla f(\bar{\mathbf{x}}_k)\|} &= \frac{\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2}{\|J(\bar{\mathbf{x}}_k)^{-1}\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \, \|J(\bar{\mathbf{x}}_k)^T\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|} \\ &\geq \frac{1}{\|J(\bar{\mathbf{x}}_k)^{-1}\| \, \|J(\bar{\mathbf{x}}_k)^T\|} &= \frac{1}{\kappa(J(\bar{\mathbf{x}}_k))} = \frac{|\lambda|_{\min}}{|\lambda|_{\max}}. \end{aligned}$$

If the **condition number** $\kappa(J(\bar{\mathbf{x}}_k))$ is uniformly bounded, we have $\cos\theta_k \geq \delta > 0$. When $\kappa(J(\bar{\mathbf{x}}_k))$ is large, the Newton direction may cause poor performance, since $\cos\theta_k \rightsquigarrow 0$.

SAN DIEGO STATE UNIVERSITY

If $J(\bar{\mathbf{x}})$ is **ill-conditioned** (close to singular), then we must modify the Newton step in order to ensure that $\cos\theta_k \geq \delta > 0$ holds.

For instance, we can add a $\tau_k I$ to $J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)$, and define the modified Newton step to be

$$\bar{\mathbf{p}}_k = -\left[J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k) + \tau_k I\right]^{-1} J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)$$

Usually, we do not want to do this explicitly. Instead we use the fact that the Cholesky factor of $J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k) + \tau_k I$ is identical to R^T, where R is the upper triangular factor of the **QR-factorization** of the matrix

$$\begin{bmatrix} J(\bar{\mathbf{x}}_k) \\ \sqrt{\tau_k} I \end{bmatrix}.$$

This factorization can be implemented in such a way that repeating the factorization for an updated value of $\tau_k^{[\mu+1]} = \tau_k^{[\mu]} + \epsilon$ is cheap.

Practical Line Search Methods

The inexactness does not compromise the global convergence behavior:

For an inexact Newton step, $\bar{\mathbf{p}}_k$, we have,

$$\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}_k\| \leq \eta_k \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|.$$

Squaring this inequality gives

$$2\bar{\mathbf{p}}_k^T J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 + \|J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}_k\|^2 \leq \eta_k^2 \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2$$

$$\Rightarrow \quad \bar{\mathbf{p}}_k^T \nabla \bar{\mathbf{r}}(\bar{\mathbf{x}}_k) = \bar{\mathbf{p}}_k^T J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k) \leq \left[\frac{\eta_k^2 - 1}{2}\right] \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2.$$

We also have,

$$\|\bar{\mathbf{p}}_k\| \leq \|J(\bar{\mathbf{x}}_k)^{-1}\| \left[\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}_k\| + \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \right] \leq (\eta_k + 1)\|J(\bar{\mathbf{x}}_k)^{-1}\| \, \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|,$$

$$\|\nabla \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| = \|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \leq \|J(\bar{\mathbf{x}}_k)\| \, \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|.$$

Putting it all together...

San Diego State University

Practical Line Search Methods

We can now write down an estimate for $\cos\theta_k$ for the inexact Newton directions

$$\cos\theta_k = -\frac{\bar{\mathbf{p}}_k^T \nabla \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)}{\|\bar{\mathbf{p}}_k\| \, \|\nabla \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|} \geq \frac{1-\eta_k^2}{2(1+\eta_k)\|J(\bar{\mathbf{x}}_k)\| \|J(\bar{\mathbf{x}}_k)^{-1}\|} \geq \frac{1-\eta_k}{2\kappa(J(\bar{\mathbf{x}}_k))}.$$

This is the same bound (with a different constant) as the bound for Newton's method.

— **Hence, inexact Newton converges when Newton's method does.**

San Diego State University

Line Search Newton for Nonlinear Equations — Algorithm

Given $\delta \in (0,1)$ and c_1, c_2 with $0 < c_1 < c_2 < \frac{1}{2}$, and $\bar{\mathbf{x}}_0 \in \mathbb{R}^n$:

Algorithm: Line Search Newton for Nonlinear Equations

while($\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| > \epsilon$)
  if $\bar{\mathbf{p}} = -J(\bar{\mathbf{x}}_k)^{-1}\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)$ satisfies $\cos\theta_k \geq \delta$
    Accept $\bar{\mathbf{p}}_k = \bar{\mathbf{p}}$
  else
    Search for $\bar{\mathbf{p}}_k(\tau_k)$ satisfying $\cos\theta_k(\tau_k) \geq \delta$
    $\bar{\mathbf{p}}_k(\tau_k) = -\left[J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k) + \tau_k I\right]^{-1} J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)$
  endif
  if $\alpha = 1$ satisfies the Wolfe conditions
    $\alpha_k = 1$
  else
    Perform a line-search to find $\alpha_k > 0$ satisfying
    the Wolfe conditions.
  endif
  $\bar{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_k + \alpha_k \bar{\mathbf{p}}_k$
endwhile(k = k + 1)

Line Search Newton for Nonlinear Equations

Theorem

Suppose that a line search algorithm that uses Newton search directions yields a sequence $\{\bar{\mathbf{x}}_k\}$ *that converges to* $\bar{\mathbf{x}}^*$*, where* $\bar{\mathbf{r}}(\bar{\mathbf{x}}^*) = 0$ *and* $J(\bar{\mathbf{x}}^*)$ *is non-singular. Suppose also that there is an open neighborhood* $\mathcal{D}$ *of* $\bar{\mathbf{x}}^*$ *such that the components* $r_i(\bar{\mathbf{x}})$ *are twice differentiable, with* $\|\nabla r_i(\bar{\mathbf{x}})\|$ *bounded for* $\bar{\mathbf{x}} \in \mathcal{D}$*. If the unit step length* α_k *is accepted whenever it satisfies the Wolfe conditions, with* $c_2 < \frac{1}{2}$*, then the convergence is Q-***quadratic***; that is* $\|\bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}^*\| = \mathcal{O}(\|\bar{\mathbf{x}}_k - \bar{\mathbf{x}}^*\|^2)$*.*

Note: This theorem applies to **any** algorithm which eventually uses the Newton search direction.

Practical Trust-Region Methods

The most commonly used trust-region method for nonlinear equations is simply "standard trust-region" applied to the merit function $f(\bar{\mathbf{x}}) = \frac{1}{2}\|\bar{\mathbf{r}}(\bar{\mathbf{x}})\|^2$, using $B_k = J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)$ as the approximate Hessian in the model function $m_k(\bar{\mathbf{p}})$. (Levenberg-Marquardt style...)

Global convergence follows directly from previously proved theorems for convergence of trust-region methods.

Rapid local convergence can be shown under the assumption that the Jacobian $J(\bar{\mathbf{x}})$ is Lipschitz continuous.

In the next few slides we take a closer look at the trust-region method for nonlinear equations.

Practical Trust-Region Methods

Our model function is given by

$$\begin{aligned} m_k(\bar{\mathbf{p}}) &= \frac{1}{2}\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}\|_2^2 \\ &= f(\bar{\mathbf{x}}_k) + \bar{\mathbf{p}}^T J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + \frac{1}{2}\bar{\mathbf{p}}^T J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}. \end{aligned}$$

As usual we generate the step $\bar{\mathbf{p}}_k$ by solving the sub-problem

$$\bar{\mathbf{p}}_k = \arg\min_{\bar{\mathbf{p}} \in \mathbb{R}^n} m_k(\bar{\mathbf{p}}), \quad \text{subject to } \|\bar{\mathbf{p}}\| \leq \Delta_k.$$

We can express ρ_k, the ratio of actual to predicted reduction as

$$\rho_k = \frac{\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 - \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k + \bar{\mathbf{p}}_k)\|^2}{\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 - \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}_k\|^2}.$$

Trust Region for Nonlinear Equations — Algorithms...

Given $\overline{\Delta} > 0$, $\Delta_0 \in (0, \overline{\Delta})$ and $\eta \in [0, \frac{1}{4})$

Algorithm: Trust Region for Nonlinear Equations

while($\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| > \epsilon$)

  $\bar{\mathbf{p}}_k = \arg\min_{\bar{\mathbf{p}} \in \mathbb{R}^n} m_k(\bar{\mathbf{p}})$, subject to $\|\bar{\mathbf{p}}\| \leq \Delta_k$ [TR]

  $\rho_k = \frac{\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 - \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k + \bar{\mathbf{p}}_k)\|^2}{\|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^2 - \|\bar{\mathbf{r}}(\bar{\mathbf{x}}_k) + J(\bar{\mathbf{x}}_k)\bar{\mathbf{p}}_k\|^2}$

  if($\rho_k < \frac{1}{4}$)

    $\Delta_{k+1} = \frac{1}{4}\|\bar{\mathbf{p}}_k\|$

  else

    if($\rho_k > \frac{3}{4}$ and $\|\bar{\mathbf{p}}_k\| = \Delta_k$)

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

    else

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

    endif

  endif

  if($\rho_k > \eta$) { $\bar{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_k + \bar{\mathbf{p}}_k$ } else { $\bar{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_k$ } endif

endwhile(k = k + 1)

Trust Region for Nonlinear Equations

We take a closer look at the solution of the subproblem [TR] using the dogleg method.

The **Cauchy point** is given by

$$\bar{\mathbf{p}}_k^c = -\tau_k \left(\frac{\Delta_k}{\|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|} \right) J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k),$$

where

$$\tau_k = \min \left\{ 1, \frac{\|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\|^3}{\Delta_k \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)(J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)) J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)} \right\}.$$

For the **full step**, we use the fact that the model Hessian $B_k = J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)$ is symmetric *semi*-definite; when $J(\bar{\mathbf{x}}_k)$ has full rank we get

$$\bar{\mathbf{p}}_k^J = -\left[J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)\right]^{-1} \left[J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\right] = -J(\bar{\mathbf{x}}_k)^{-1} \bar{\mathbf{r}}(\bar{\mathbf{x}}_k).$$

Trust Region for Nonlinear Equations

The dogleg selection of $\bar{\mathbf{p}}_k$ is given by:

Algorithm: Dogleg Selection

```
Calculate p̄_k^c
if( ||p̄_k^c|| = Δ_k )
  p̄_k = p̄_k^c
else
  Calculate p̄_k^J
  if( ||p̄_k^J|| < Δ_k )
    p̄_k = p̄_k^J
  else
    p̄_k = p̄_k^c + τ(p̄_k^J − p̄_k^c), where τ ∈ [0,1] : ||p̄_k|| = Δ_k
  endif
endif
```

Trust Region for Nonlinear Equations

Exact Solution

From previous results we know that the exact solution of the subproblem [TR] has the form

$$\bar{\mathbf{p}}_k = -\left[J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k) + \lambda_k I\right]^{-1} \left[J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\right]$$

for some $\lambda_k \geq 0$, and that $\lambda_k = 0$ if $\|\bar{\mathbf{p}}_k^J\| \leq \Delta_k$.

Note that this is the same linear system that gives the Levenberg-Marquardt step $\bar{\mathbf{p}}_k^{\text{LM}}$ in the discussion on nonlinear least squares.

In a sense the LM-approach for non-linear equations is a special case of the LM-approach for nonlinear least squares problems.

We can identify an approximation of λ_k using the Cholesky factorization, *e.g.* `modelhess` in the project code; alternatively we can base the search on the QR-factorization.

Trust Region for Nonlinear Equations

The dogleg method has the **advantage** over methods trying to attain the exact solution to the subproblem in that **only one linear system needs to be solved per iteration**.

Global convergence for the trust-region algorithm is described in the two following theorems (which should look somewhat familiar...): –

SAN DIEGO STATE UNIVERSITY

Trust Region for Nonlinear Equations — Convergence, 1 of 2

Theorem

Let $\eta = 0$ in the trust-region algorithm. Suppose that $J(\bar{\mathbf{x}})$ is continuous in a neighborhood $\mathcal{D}$ of the level set $\mathcal{L}(\bar{\mathbf{x}}_0) = \{\bar{\mathbf{x}} \in \mathbb{R}^n \,:\, f(\bar{\mathbf{x}}) \leq f(\bar{\mathbf{x}}_0)\}$ and that $\|J(\bar{\mathbf{x}})\|$ is bounded above on $\mathcal{L}(\bar{\mathbf{x}}_0)$. Suppose in addition that all approximate solutions of the trust-region subproblem satisfy ($c_1 > 0$, $\gamma \geq 1$)

$$m_k(0) - m_k(\bar{\mathbf{p}}_k) \geq c_1 \|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \min\left\{\Delta_k, \frac{J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)}{J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)}\right\},$$

$$\|\bar{\mathbf{p}}_k\| \leq \gamma \Delta_k.$$

We then have that

$$\lim\inf_{k\to\infty} \|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| = 0$$

Trust Region for Nonlinear Equations — Convergence, 2 of 2

Theorem

Let $\eta \in (0, \frac{1}{4})$ *in the trust-region algorithm. Suppose that* $J(\bar{\mathbf{x}})$ *is Lipschitz continuous in a neighborhood* $\mathcal{D}$ *of the level set* $\mathcal{L}(\bar{\mathbf{x}}_0) = \{\bar{\mathbf{x}} \in \mathbb{R}^n \,:\, f(\bar{\mathbf{x}}) \leq f(\bar{\mathbf{x}}_0)\}$ *and that* $\|J(\bar{\mathbf{x}})\|$ *is bounded above on* $\mathcal{L}(\bar{\mathbf{x}}_0)$. *Suppose in addition that all approximate solutions of the trust-region subproblem satisfy* ($c_1 > 0$, $\gamma \geq 1$)

$$m_k(0) - m_k(\bar{\mathbf{p}}_k) \geq c_1 \|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| \min\left\{\Delta_k, \frac{J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)}{J(\bar{\mathbf{x}}_k)^T J(\bar{\mathbf{x}}_k)}\right\},$$

$$\|\bar{\mathbf{p}}_k\| \leq \gamma \Delta_k.$$

We then have that

$$\lim_{k \to \infty} \|J(\bar{\mathbf{x}}_k)^T \bar{\mathbf{r}}(\bar{\mathbf{x}}_k)\| = 0$$

Trust Region for Nonlinear Equations

Finally, we state a result regarding the convergence rate. Note that the result requires exact solution of the subproblem.

Theorem

Suppose that the sequence $\{\bar{\mathbf{x}}_k\}$ generated by the trust-region algorithm converges to a non-degenerate solution $\bar{\mathbf{x}}^*$ of the problem $\bar{\mathbf{r}}(\bar{\mathbf{x}}) = 0$. Suppose also that $J(\bar{\mathbf{x}})$ is Lipschitz continuous in an open neighborhood $\mathcal{D}$ of $\bar{\mathbf{x}}^*$ and that the trust-region subproblem is solved exactly for all sufficiently large k. Then the sequence $\{\bar{\mathbf{x}}_k\}$ **converges quadratically** *to $\bar{\mathbf{x}}^*$.*

Thus we can design a globally convergent method which converges quadratically! — **Robustness** and **Speed** in the same algorithm!

San Diego State University

Index

breakdown of the Newton iteration
- blowup, 3
- cycling, 4

nonlinear equations
- practical line search methods, 11
- practical trust region methods, 19

SAN DIEGO STATE UNIVERSITY