Numerical Methods for Nonlinear Equations

Homer F. Walker
Mathematical Sciences Department
Worcester Polytechnic Institute
Worcester, MA 01609-2280

This work was partially funded by the U.S. Department of Energy's ASCI program and by NSF grant DMS-9727128 to Worcester Polytechnic Institute and was carried out at Sandia National Laboratories, operated for the U.S. Department of Energy under contract no. DE-AC04-94AL85000.

Outline

Foreword

The following are lecture notes from a short course given at Sandia National Laboratories in Albuquerque, New Mexico in the summer of 2001. Part 1 (Methods for General Problems) was covered July 23–27; Part 2 (Methods for Large-Scale Problems) was covered August 15–17. The notes have been cleaned up and corrected in minor ways, but are for the most part as originally delivered.

I would like to thank John Shadid and Roger Pawlowski for their instrumental role in conceiving and arranging the short course and for their great help in pulling it off. I would also like to acknowledge the influence of the classic book of Dennis and Schnabel [32], which strongly guided the developments in Part I and provided a standard to which to aspire throughout.

Homer Walker
Worcester, Massachusetts
February, 2002

Slide 1

Numerical Methods for Nonlinear Equations

Homer Walker
Mathematical Sciences Department
Worcester Polytechnic Institute
Summer, 2001

Slide 2

Introduction to Part 1
Methods for General Problems

Problem: $F(x_*) = 0, \quad F : I\!R^n \to I\!R^n.$

Problem: $\min_{x \in R^n} f(x), \quad f : I\!R^n \to I\!R^1.$

Recast as $\nabla f(x_*) = 0.$

We will study *iterative* methods for finding *some* solution.

Theorems are rarely the strongest possible. Proofs will usually be off-line.

Slide 3

Topic 1

Methods for Problems in One Variable

a. Basic "pure" methods.

   i. The bisection method.

   ii. Newton's method.

   iii. The secant method.

b. Practical hybrid methods.

Slide 4

a. Basic "pure" methods.

Bisection Method:

Given a, b, such that $F(a) \cdot F(b) < 0$.

Until termination, do:

Set $c \equiv \dfrac{a+b}{2}$.

If $F(c) \cdot F(b) < 0$, $a \leftarrow c$; else $b \leftarrow c$.

If F is continuous on the initial $[a_0, b_0]$, then there is an $x_* \in [a_0, b_0]$ such that $F(x_*) = 0$ and

$$|c_k - x_*| \leq \frac{b_0 - a_0}{2^{k+1}} \ .$$

Slide 5

Taylor series: $0 = F(x_*) = F(x) + F'(x)(x_* - x) + o(x_* - x)$

$\Rightarrow x_* \approx x - F'(x)^{-1}F(x)$.

Newton's Method:

Given an initial x.

Until termination, do:

$x \leftarrow x - F'(x)^{-1}F(x)$

If F is *Lipschitz continuously differentiable* near x_* such that $F(x_*) = 0$ and $F'(x_*) \neq 0$, then for x_0 sufficiently near x_*, $x_k \to x_*$ with

$$|x_{k+1} - x_*| \leq C|x_k - x_*|^2 .$$

Slide 6

Secant Method:

Given initial x, x_-.

Until termination, do:

$$x_+ \leftarrow x - \left(\frac{F(x) - F(x_-)}{x - x_-}\right)^{-1} F(x)$$

$$x_- \leftarrow x, \quad x \leftarrow x_+$$

If F is Lipschitz continuously differentiable near x_* such that $F(x_*) = 0$ and $F'(x_*) \neq 0$, then for x_0, x_{-1} sufficiently near x_*, $x_k \to x_*$ with

$$|x_{k+1} - x_*| \leq C|x_k - x_*| \cdot |x_{k-1} - x_*| .$$

Slide 7

Comparison.

- **Bisection:**
  - — One F-evaluation per iteration.
  - — Guaranteed convergence (with strong assumption).
  - — **Slow** convergence (r-*linear*).

- **Newton:**
  - — One F-evaluation and one F'-evaluation per iteration.
  - — Only *local* convergence in general (may **diverge** without a good initial guess).
  - — **Very fast** local convergence (q-quadratic).

- **Secant:**
  - — One F-evaluation per iteration.
  - — Only *local* convergence in general.
  - — **Fast** local convergence (q-superlinear).

Slide 8

b. Practical Hybrid Methods.

We can construct "hybrid" methods that combine features to retain desirable properties, eliminate undesirable ones.

Brent's Algorithm [10, 11].

- Combines aspects of bisection and secant methods, with additional features to safeguard against worst cases.
- "Enclosure" method, one F-evaluation per iteration (no F'-evaluations), usually converges at least as fast as the secant method.
- Given a tolerance $\delta > 0$, terminates with an approximate solution within 2δ of an actual solution.
- A good implementation is subroutine ZEROIN from Forsythe, Malcolm, Moler [45], available through Netlib (`www.netlib.org`).

Slide 9

Brent's Algorithm.

- Initially: Have a, b such that $F(a) \cdot F(b) \leq 0$, stopping tolerance $\delta > 0$.
- At each iteration: Have a, b, c (initially $c = a$) such that
  - ▷ $F(b) \cdot F(c) \leq 0 \Rightarrow$ solution lies between b and c;
  - ▷ $|F(b)| \leq |F(c)| \Rightarrow b$ is the current approximate solution;
  - ▷ either a, b and c are distinct *or* $a = c$.
- Iteration: If $|b - c| \leq 2\delta$, stop with $b \approx x_*$; else
  - ▷ Try a new b given by
    - — linear interpolation (secant step) if $a = c$;
    - — inverse quadratic interpolation if a, b, and c are distinct.
  - ▷ Modify if necessary so the step is neither too short nor too long.
  - ▷ Update a, b, and c.

Slide 10

Summary.

- Different "pure" methods have different properties.
  - — Robustness: likelihood of convergence to a solution.
  - — Speed: rate of *local* convergence.
  - — Expense *per iteration*: function and perhaps derivative evaluations; in higher dimensions, arithmetic and storage as well.
- No need to stick to "pure" methods.
  - — We can combine/augment them with auxiliary procedures to obtain features we like.
- For a particular application:
  - — Feasibility and robustness are overriding.
  - — Given these, we want an optimal balance of convergence speed and cost per iteration.

Slide 11

Topic 2

Newton's Method for Problems in Several Variables

a. Formulation and properties.

b. Stopping and scaling.

c. Finite-difference Newton's method.

Slide 12

a. Formulation and properties.

Problem: $F(x_*) = 0, \quad F : \mathbb{R}^n \to \mathbb{R}^n.$

Note:

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}, \qquad F(x) = \begin{pmatrix} F_1(x) \\ \vdots \\ F_n(x) \end{pmatrix},$$

$$F'(x) = J(x) = \left(\frac{\partial F_i(x)}{\partial x_j} \right) \in \mathbb{R}^{n\times n}.$$

ASSUME THROUGHOUT: *F is continuously differentiable.*

Slide 13

Taylor series: $0 = F(x_*) = F(x) + F'(x)(x_* - x) + o(x_* - x)$

$\Rightarrow x_* \approx x - F'(x)^{-1}F(x)$.

Newton's Method:

Given an initial x.

Until termination, do:

$x \leftarrow x - F'(x)^{-1}F(x)$

Slide 14

Somewhat more realistically ...

Newton's Method:

Given an initial x.

Iterate:

Decide whether to stop or continue.

Solve $J(x)s = -F(x)$.

Update $x \leftarrow x + s$.

Cost per iteration (in general, full-matrix case) ...

- one F-evaluation, one J-evaluation,
- $O(n^3)$ arithmetic operations,
- $O(n^2)$ storage

Slide 15

"Solve" step: Often this is "approximately solve," or "solve an approximate equation."

- May "perturb" $J(x)$ to mollify ill-conditioning or (in optimization case) indefiniteness; see [32, §5.5, §6.5].
- May replace $J(x)$ with an approximate Jacobian, as in *finite-difference Newton's method* and *quasi-Newton methods*.
- May approximately solve with an iterative linear algebra method, as in *Newton iterative (truncated Newton) methods*.

Slide 16

- *Quadratic local convergence*.

Theorem: *Suppose F is Lipschitz continuously differentiable at x_*, and that $F(x_*) = 0$ and $J(x_*)$ is nonsingular. Then for x_0 sufficiently near x_*, $\{x_k\}$ produced by Newton's method is well-defined and converges to x_* with*

$$\|x_{k+1} - x_*\| \leq C\|x_k - x_*\|^2$$

for a constant C independent of k.

- Iterates may *diverge* if x_0 is not near a solution.
- Convergence is typically *mesh independent* for discretized PDE problems (see, e.g., [2], [1]).

Slide 17

Newton's method is *scale independent*, as follows:

Suppose $\hat{x} = Ax$ for $A \in I\!R^{n \times n}$. Set $\hat{F}(\hat{x}) = F(A^{-1}\hat{x})$.

Then $\hat{J}(\hat{x}) = \hat{F}'(\hat{x}) = F'(A^{-1}\hat{x})A^{-1}$, and for

$$x_+ = x - J(x)^{-1}F(x), \qquad \hat{x}_+ = \hat{x} - \hat{J}(\hat{x})^{-1}\hat{F}(\hat{x}),$$

we have

$$\begin{array}{ccc} x & \stackrel{A}{\longrightarrow} & \hat{x} = Ax \\ \downarrow & & \downarrow \\ x_+ & \stackrel{A^{-1}}{\longleftarrow} & \hat{x}_+ = Ax_+ \end{array}$$

(But scaling may affect other algorithmic features.)

Slide 18

Considerations for optimization.

Problem: $\min_{x \in R^n} f(x), \quad f : I\!R^n \to I\!R^1.$

Recast as $\nabla f(x_*) = 0$.

Apply Newton's method with $F(x) = \nabla f(x)$.

Note: $J(x) = \nabla^2 f(x)$ is symmetric, possibly positive definite.

Note: Iterates may diverge or converge to a point that is not a local minimizer.

b. Stopping and scaling.

Stopping.

Slide 19

Make general comments and outline general criteria.

Problem-specific criteria are often superior in practice.

Questions (cf. [32]):

- *Have we solved the problem?*
- *Have we bogged down?*
- *Have we run out of time, patience, or money?*

- *Have we run out of time, patience, or money?*

Test: Stop if the iteration number reaches some *itmax*.

- *Have we solved the problem?*

Slide 20

Test: Stop if $\|F(x)\| \leq tol_F$.

Note: Near a solution, *this gives a bound on the error*

$$\|x - x_*\| \leq \|J(x)^{-1}\| \cdot \|J(x)(x - x_*)\| \approx \|J(x)^{-1}\| \cdot \|F(x) - F(x_*)\| \leq \|J(x)^{-1}\| tol_F.$$

Caution: tol_F must be carefully chosen to reflect the scale of F. A *scaled norm* may be most appropriate if the components of F differ greatly in scale.

Useful variation ([64]): Stop if $\|F(x)\| \leq tol_{Rel}\|F(x_0)\| + tol_{Abs}$.

- *Have we bogged down?*

Test: Stop if $\|s\| \le tol_x$.

Slide 21

Similar cautions about scaling apply. There is a similiar useful variation.

Note: For $s = J(x)^{-1}F(x)$,

$$\|s\| = \|J(x)^{-1}\left[F(x) - F(x_*)\right]\| \approx \|J(x)^{-1}J(x)(x - x_*)\| = \|x - x_*\|.$$

So near a solution, *this serves as a test on the error in the approximate solution*.

Scaling.

Often, components of F or x differ greatly in magnitude.

Despite the scale independence of "pure" Newton's method, this can create difficulties, e.g.: in stopping tests, solving for the Newton step, certain "globalization" procedures (later).

Slide 22

Often useful to *rescale*: Possibilities (see [32]) . . .

- Choose different units for the components of F or x to improve scaling.
- Apply diagonal scaling matrices and solve the rescaled problem:
  - — Choose $D_x = \text{diag}\,(d_{11}, \ldots, d_{nn})$ so that d_{ii} is a "typical" value of x_i. Similarly choose D_F.
  - — Set $\hat{x} = D_x^{-1}x$, $\hat{F}(\hat{x}) = D_F^{-1}F(D_x\hat{x})$.
  - — Solve $\hat{F}(\hat{x}) = 0$.

Slide 23

c. Finite-difference Newton's method.

Often, analytic evaluation of $J(x)$ is undesirable or infeasible.

We can use instead a *finite-difference approximation*. See [32] for a theoretical treatment. Focus on the practical aspects here.

Approximate J(x) using . . .

- *forward differences*

$$J(x)e_j = \frac{1}{\delta}\left[F(x+\delta e_j) - F(x)\right] + O(\delta), \qquad j = 1, \ldots, n,$$

- *central differences*

$$J(x)e_j = \frac{1}{2\delta}\left[F(x+\delta e_j) - F(x-\delta e_j)\right] + O(\delta^2), \qquad j = 1, \ldots, n.$$

Slide 24

Choosing δ.

- The goal is to choose δ to roughly balance truncation and floating point error.
- Fairly well-justified choices can be made for scalar functions. The justifications weaken with vector functions. Nothing is foolproof.

Choices used in [84] that approximately minimize bounds on the relative error in the difference approximations are . . .

- ▷ $\delta = \left[(1 + \|x\|)\epsilon_F\right]^{1/2}$ for forward differences,
- ▷ $\delta = \left[(1 + \|x\|)\epsilon_F\right]^{1/3}$ for central differences,

where ϵ_F denotes the relative error in F-evaluations ("function precision").

Main underlying assumption: F and its derivatives up to orders two, respectively three, have about the same scale.

Slide 25

Typically, if F has k accurate digits,

— forward differences give $\approx k/2$ accurate digits,

— central differences give $\approx 2k/3$ accurate digits.

A crude, often-used heuristic is ...

▷ $\delta = \epsilon^{1/2}$ for forward differences,

▷ $\delta = \epsilon^{1/3}$ for central differences,

where ϵ is machine epsilon.

In practice, the convergence of finite-difference Newton iterates is usually (but not always) very nearly the same as that of Newton iterates.

For many sparse (especially banded) Jacobians, one can greatly reduce F-evaluations with the Curtis-Powell-Reid trick [26].

Slide 26

Special considerations for optimization.

If derivatives of f are unavailable, then it may be necessary to evaluate $F = \nabla f$ itself using finite-differences.

Accuracy may be an important issue.

- Finite-difference approximations of J must use relatively inaccurate F-values.
- Since $\nabla f = 0$ at an optimizer, finite-difference evaluation of F and J may suffer increasing loss of accuracy through **cancellation**.

It may be necessary to use central differences, especially near an optimizer.

Slide 27

Topic 3

Globally Convergent Modifications of Newton's Method

a. Criteria for global convergence.

b. Backtracking methods.

c. Trust region methods.

Slide 28

a. Criteria for global convergence.

We will explore criteria on a sequence of iterates that **make it likely** that it will converge to a solution.

Later, we'll see how to modify Newton steps (or closely related steps) so that these criteria are satisfied.

Important notes:

- There is no way to ensure that iterates will **always** converge to a solution of **every** problem.
- The goal is to enhance the likelihood of convergence to **some** solution, not to any distinguished solution such as a global optimizer.

Slide 29

Suppose we have $\{x_k\}$.

Ideally: We want conditions on $\{x_k\}$ that imply $x_k \to x_*$ such that $F(x_*) = 0$.

Reasonable: Require $\|F(x_{k+1})\| < \|F(x_k)\|$.

This is not enough!

Examples: Take $F(x) = 1 - x^2$ and $x_0 = 0$. Define ...

- $x_k = x_{k-1} + 2^{-(k+1)}$ for $k = 1, 2, \ldots$.

  See: $x_k \to \frac{1}{2}$, $F(x_k) \not\to 0$. (Steps too short!)

- $x_k = -x_{k-1} + (-1)^{k-1} 2^{-k}$ for $k = 1, 2, \ldots$.

  See: $F(x_k) \to 0$ but $\{x_k\}$ has limit points $\pm 1 \Rightarrow$ no limit. (Steps too long!)

Slide 30

Criteria based on actual/predicted norm reduction.

Given $x \in I\!R^n$ and a step $s \in I\!R^n$, define

- $ared \equiv \|F(x)\| - \|F(x+s)\|$, the *actual reduction* of $\|F\|$;
- $pred \equiv \|F(x)\| - \|F(x) + J(x)s\|$, the *predicted reduction* of $\|F\|$,
- $relpred \equiv \begin{cases} pred/\|F(x)\|, & \text{if } F(x) \neq 0 \\ 1, & \text{if } F(x) = 0 \end{cases}$, the *relative predicted reduction*.

Note: $pred$ is the reduction in $\|F\|$ "predicted" by $F(x) + J(x)s$, the *local linear model* of F at x.

Slide 31

Theorem [37, Cor. 3.6]: *Suppose* $\{x_k\}$ *is such that for each* k *and* $s_k \equiv x_{k+1} - x_k$,

$$ared_k \geq t \cdot pred_k \geq 0$$

for some $t \in (0, 1)$ *independent of* k. *If* $\sum_{k=0}^{\infty} relpred_k = \infty$, *then* $F(x_k) \to 0$. *If also* $\{x_k\}$ *has a limit point* x_* *such that* $J(x_*)$ *is nonsingular, then* $F(x_*) = 0$ *and* $x_k \to x_*$.

Proof (easy part): Suppose $F(x_k) \not\to 0$. Note that $\{\|F(x_k)\|\}$ is monotone decreasing. Then there is an $\epsilon > 0$ such that $\|F(x_k)\| \geq \epsilon$ for all k, and

$$\begin{aligned}
\|F(x_0)\| &\geq \|F(x_0)\| - \|F(x_{k+1})\| = \sum_{j=0}^{k} ared_j \\
&\geq t \cdot \sum_{j=0}^{k} pred_j = t \cdot \sum_{j=0}^{k} relpred_j \|F(x_j)\| \\
&\geq t \cdot \epsilon \cdot \sum_{j=0}^{k} relpred_j
\end{aligned}$$

It follows that $\sum_{j=0}^{\infty} relpred_j < \infty$.

Slide 32

Under the assumptions of the theorem, if $\sum_{k=0}^{\infty} relpred_k = \infty$, then exactly one of the following holds:

- $\|x_k\| \to \infty$;
- $\{x_k\}$ has one or more limit points, and J is singular at each of them;
- $x_k \to x_*$ such that $F(x_*) = 0$ and $J(x_*)$ is nonsingular.

Easy examples show ...

— *it is possible for each of these to hold;*

— *it may not be possible to satisfy* $\sum_{k=0}^{\infty} relpred_k = \infty$.

Slide 33

Directly verifying $\sum_{k=0}^{\infty} relpred_k = \infty$ may be difficult/impossible.

Plan: We will construct algorithms that begin each iteration with the Newton step or something closely related, then modify it if necessary to obtain a step satisfying $ared \geq t \cdot pred \geq 0$. The controlled nature of the modifications *implicitly* ensures $\sum_{k=0}^{\infty} relpred_k = \infty$, when possible.

There is no need to verify explicitly that $\sum_{k=0}^{\infty} relpred_k = \infty$.

Remark: For nonlinear equations, the condition $ared \geq t \cdot pred$ is a special case of more general tests considered in [86], [39], and [40].

Slide 34

Ared/pred criteria for optimization.

Such criteria have been considered for $\min_{x \in \mathbf{R}^n} f(x)$ in [75] and [97].

These require $ared \geq t \cdot pred$, where

- $ared \equiv f(x) - f(x+s)$
- $pred \equiv -\nabla f(x)^T s - \frac{1}{2} s^T \nabla^2 f(x) s$.

Note: $pred$ is the reduction in $f(x)$ "predicted" by

$$f(x) + \nabla f(x)^T s + \frac{1}{2}\, s^T \nabla^2 f(x) s,$$

the *local quadratic model* of f.

Convergence results in [75], [97] are in the context of *trust region methods*.

Slide 35

Goldstein–Armijo type criteria.

Develop these first for optimization: $\min_{x\in \mathbb{R}^n} f(x)$.

Def.: $s \in \mathbb{R}^n$ is a *descent direction* for f at $x \in \mathbb{R}^n$ if $\nabla f^T(x) s < 0$.

Note: *The Newton step* $s^N = -\nabla^2 f(x)^{-1}\nabla f(x)$ *is a descent direction if* $\nabla^2 f(x)$ *is positive definite.*

Goldstein–Armijo conditions [52], [4]: For $0 < \alpha < \beta < 1$ and a descent direction s,

- $f(x+s) \le f(x) + \alpha \nabla f(x)^T s$ (the α-condition),
- $\nabla f(x+s)^T s \ge \beta \nabla f(x)^T s$ (the β-condition).

The condition $0 < \alpha < \beta < 1$ ensures that there exist steps that satisfy these conditions (see [32, Th. 6.3.2]). In practice, we need $0 < \alpha < \frac{1}{2}$ so the Newton step will satisfy them near a minimizer (see [32, Th 6.3.4]).

Slide 36

Theorem [109], [110]: *Suppose* $f : \mathbb{R}^n \to \mathbb{R}^1$ *is continuously differentiable and that* $\{x_k\}$ *is such that each* $s_k \equiv x_{k+1} - x_k$ *is a descent direction satisfying the two Goldstein–Armijo conditions. Suppose also that* $\|\nabla f(x_{k+1}) - \nabla f(x_k)\| \le \lambda \|s_k\|$ *for some* λ *independent of* k. *Then either* $f(x_k) \to -\infty$ *or*

$$\nabla f(x_k)^T \left(\frac{s_k}{\|s_k\|}\right) \to 0.$$

Plan: We will construct algorithms that begin each iteration with the Newton step or something closely related and ultimately produce an acceptable s_k that is a descent direction and such that $\left(\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|}\right)^T \left(\frac{s_k}{\|s_k\|}\right)$ is bounded away from zero. Then the theorem lends itself to strong global convergence statements.

Slide 37

Theorem: *Suppose the assumptions of the previous theorem hold and $f : \mathbb{R}^n \to \mathbb{R}^1$ is twice continuously differentiable. If $\{x_k\}$ has a limit point x_* such that $\nabla^2 f(x_*)$ is positive-definite and, for some $\epsilon > 0$,*

$$-\left(\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|}\right)^T \left(\frac{s_k}{\|s_k\|}\right) \geq \epsilon$$

whenever x_k is sufficiently near x_*, then x_* is a local minimizer of f. If also there is a C such that $\|s_k\| \leq C\,\|\nabla f(x_k)\|$ whenever x_k is sufficiently near x_*, then $x_k \to x_*$.

Slide 38

A variation is ...

Moré–Thuente conditions [76]: For $0 < \alpha \leq \beta < 1$ and a descent direction s,

- $f(x+s) \leq f(x) + \alpha \nabla f(x)^T s$,
- $|\nabla f(x+s)^T s| \leq \beta |\nabla f(x)^T s|$

The second is *stronger* than the Goldstein–Armijo β condition, may be harder to satisfy.

Advantage: It may prevent taking some steps with larger function values.

Slide 39

Adaptation for nonlinear equations.

For solving $F(x_*) = 0$, apply the Goldstein–Armijo or Moré–Thuente tests with $f(x) \equiv \frac{1}{2}\|F(x)\|_2^2$.

Note: $\nabla f(x) = J(x)^T F(x)$.

Further remarks:

- The Goldstein–Armijo β-condition is usually ignored in practice. Steps are tested with respect to the α-condition only.
- For nonlinear equations, the α-condition implies that the condition $ared \geq t \cdot pred$ holds with $t = \alpha$ [37, Prop. 2.1].

Slide 40

b. Backtracking methods.

We will now explore a first way of modifying Newton steps (or closely related steps) to obtain steps that satisfy acceptability criteria.

Backtracking idea: *If a step is not acceptable,* **shorten it** *as necessary to obtain a step that is.*

Def.: $s \in \mathbb{R}^n$ is an *inexact Newton step* if $\|F(x) + J(x)\, s\| < \|F(x)\|$.

- If $F(x) = 0$, then $s = 0$ may also be considered an inexact Newton step.
- The Newton step $s^N = -J(x)^{-1}F(x)$ is an inexact Newton step.

Slide 41

Lemma: *Suppose s is an inexact Newton step and $F(x) \neq 0$. Then for fixed $t \in (0,1)$, $ared(\lambda s) > t \cdot pred(\lambda s) > 0$ for sufficiently small $\lambda > 0$.*

Proof: For $\lambda \in (0,1)$,

$$\begin{aligned} pred(\lambda s) &\equiv \|F(x)\| - \|F(x) + J(x)(\lambda s)\| \\ &= \|F(x)\| - \|(1-\lambda)F(x) + \lambda\big(F(x) + J(x)\, s\big)\| \\ &\geq \|F(x)\| - (1-\lambda)\|F(x)\| - \lambda\|F(x) + J(x)\, s\| \\ &= \lambda \cdot pred(s) > 0, \end{aligned}$$

Also,

$$\begin{aligned} ared(\lambda s) &\equiv \|F(x)\| - \|F(x + \lambda s)\| \\ &\geq \|F(x)\| - \|F(x) + J(x)(\lambda s)\| + o(\lambda) \\ &= pred(\lambda s) + o(\lambda) > t \cdot pred(\lambda s) \end{aligned}$$

for sufficiently small $\lambda > 0$.

Slide 42

Our basic backtracking method is ...

Newton's Method with Backtracking:

Given $t \in (0,1)$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.

Evaluate $F(x)$.

Iterate:

- Decide whether to stop or continue.
- Solve $J(x)s = -F(x)$.
- Evaluate $F(x+s)$.
- While $ared < t \cdot pred$, do:
  - Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.
  - Update $s \leftarrow \theta s$, re-evaluate $F(x+s)$.
- Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.

Slide 43

The reduction $s \leftarrow \theta s$ with $\theta \in [\theta_{\min}, \theta_{\max}]$ is **"safeguarded" backtracking**:

- $\theta \leq \theta_{\max}$ ensures that the backtracking loop will terminate with an acceptable step.
- $\theta \geq \theta_{\min}$ ensures that steps will not be shorter than necessary.

The algorithm can easily be rephrased to allow for more general "inexact Newton" steps.

If the initial step is the exact Newton step $s^N = -J(x)^{-1}F(x)$, and if subsequently $s = \lambda s^N$ for $0 \leq \lambda < 1$, then $pred = \lambda \|F(x)\|$ and

$$ared \geq t \cdot pred \iff \|F(x+s)\| \leq \big(1 - t \cdot \lambda\big)\|F(x)\|.$$

Slide 44

Recast the algorithm as ...

Newton's Method with Backtracking:

Given $t \in (0,1)$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.
Evaluate $F(x)$.
Iterate:
  Decide whether to stop or continue.
  Solve $J(x)s = -F(x)$.
  Evaluate $F(x+s)$; set $\lambda = 1$.
  While $\|F(x+s)\| > \big(1 - t \cdot \lambda\big)\|F(x)\|$, do:
    Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.
    Update $s \leftarrow \theta s$, $\lambda \leftarrow \theta\lambda$, and re-evaluate $F(x+s)$.
  Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.

Slide 45

Theorem [37, Cor. 6.2]: *Suppose $\{x_k\}$ is a sequence produced by the algorithm. If x_* is a limit point of $\{x_k\}$ such that $J(x_*)$ is nonsingular, then $F(x_*) = 0$, $x_k \to x_*$, and $s_k \equiv x_{k+1} - x_k = -J(x_k)^{-1}F(x_k)$ for all sufficiently large k.*

- If $x_k \to x_*$ and $s_k = -J(x_k)^{-1}F(x_k)$ for all large k, then the convergence is that of Newton's method (probably quadratic).
- No explicit assumption $\sum_{k=0}^{\infty} relpred_k = \infty$ is necessary; this is shown to hold in the proof of the theorem.
- Exactly one of the following must hold:
  - ▷ $\|x_k\| \to \infty$;
  - ▷ $\{x_k\}$ has one or more limit points, and J is singular at each of them;
  - ▷ $x_k \to x_*$ such that $F(x_*) = 0$, $J(x_*)$ is nonsingular, and the convergence is eventually that of Newton's method.

Slide 46

These allow drawing nice corollaries by making assumptions about F.

Corollary: For $x_0 \in \mathbb{R}^n$, suppose $\mathcal{L}(x_0) \equiv \{x : \|F(x)\| \le \|F(x_0)\|\}$ is bounded and J is nonsingular everywhere on $\mathcal{L}(x_0)$. Then there exists $x_* \in \mathcal{L}(x_0)$ such that $F(x_*) = 0$ and $x_k \to x_*$, and the convergence is that of Newton's method.

Corollary: Suppose F is *norm-coercive* on $\mathbb{R}^n$, i.e., $\|F(x)\| \to \infty$ as $\|x\| \to \infty$. If J is nonsingular everywhere on $\mathbb{R}^n$, then F maps $\mathbb{R}^n$ *onto* $\mathbb{R}^n$, i.e., for every $y \in \mathbb{R}^n$, there is an $x \in \mathbb{R}^n$ such that $F(x) = y$.

In fact, a much stronger result is that a continuous $F : \mathbb{R}^n \to \mathbb{R}^n$ is norm-coercive on $\mathbb{R}^n$ if and only if it is onto and one-to-one on $\mathbb{R}^n$; see [81, §5.3.8].

Slide 47

Practical implementation.

- *Choose t small, e.g., $t = 10^{-4}$*, so a step will be accepted if there is minimal (but still adequate) progress.
- *Choose $\theta_{\min} = .1$, $\theta_{\max} = .5$*, arbitrary but typical practice (cf. [32]).
- *Choosing $\theta \in [\theta_{\min}, \theta_{\max}]$*.

Crude but always possible: $\theta = \frac{1}{2}$.

There are more sophisticated possibilities if $\|\cdot\|$ is an inner-product norm, i.e., $\|v\| = \sqrt{\langle v, v \rangle}$ for $v \in \mathbb{R}^n$. (Example: $\|v\|_2 = \sqrt{v^T v}$.)

Slide 48

Suppose $\|v\| = \sqrt{\langle v, v \rangle}$ for $v \in \mathbb{R}^n$.

Idea: Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize $g(\theta) \equiv \|F(x + \theta s)\|^2$.

This is exact line search; usually too expensive.

Alternative [32]: *Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a quadratic or cubic that interpolates g.*

Suppose we have $s = \lambda s^N$, $\quad 0 < \lambda \leq 1$, $\quad s^N = -J(x)^{-1} F(x)$.

Then $g'(\theta) = 2 \langle F(x + \theta s), J(x + \theta s)\, s \rangle$, and

$$g'(0) = 2 \langle F(x), J(x)\, s \rangle = -2\lambda \|F(x)\|^2 < 0.$$

Note: s^N is a *descent direction* for $\|F\|$ at x.

Slide 49

First step reduction: We have $g(0)$, $g'(0)$, $g(1)$. Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a quadratic that interpolates these.

Subsequent step reductions: We have $g(0)$, $g'(0)$, $g(1)$, and a third value of g; Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a cubic that interpolates these.

Minimizing the quadratic is simple; minimizing the cubic is a bit more involved. See [32] for details.

Slide 50

Details of the quadratic interpolation.

We want $p(\theta)$ such that

- $p(0) = g(0) = \|F(x)\|^2$,
- $p'(0) = g'(0) = -2\lambda\|F(x)\|^2$,
- $p(1) = g(1) = \|F(x+s)\|^2$.

Setting $\rho = \|F(x+s)\|/\|F(x)\|$, we have

$$p(\theta) = \|F(x)\|^2 - 2\lambda\|F(x)\|^2\theta + \|F(x)\|^2(\rho^2 - 1 + 2\lambda)\theta^2$$

Note: $p''(\theta) = 2\|F(x)\|^2(\rho^2 - 1 + 2\lambda)$.

- If $p''(\theta) \leq 0$, take $\theta = \theta_{\max}$.
- If $p''(\theta) > 0$, we have $p'(\theta) = 0 \iff \theta = \dfrac{\lambda}{\rho^2 - 1 + 2\lambda}$, so choose this θ, correcting if necessary to be in $[\theta_{\min}, \theta_{\max}]$.

Slide 51

Newton's Method with Quadratic Minimization Backtracking:

Given $t \in (0,1)$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.
Evaluate $F(x)$.
Iterate:
  Decide whether to stop or continue.
  Solve $J(x)s = -F(x)$.
  Evaluate $F(x+s)$; set $\lambda = 1$.
  While $\rho \equiv \|F(x+s)\|/\|F(x)\| > 1 - t \cdot \lambda$, do:
    If $\delta \equiv \rho^2 - 1 + 2\lambda \leq 0$, set $\theta = \theta_{\max}$.
    Else do:
      Set $\theta = \lambda/\delta$.
      If $\theta > \theta_{\max}$, $\theta \leftarrow \theta_{\max}$.
      If $\theta < \theta_{\min}$, $\theta \leftarrow \theta_{\min}$.
    Update $s \leftarrow \theta s$, $\lambda \leftarrow \theta\lambda$, and re-evaluate $F(x+s)$.
  Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.

Slide 52

Backtracking for optimization.

General idea: For $\min_{x \in \mathbb{R}^n} f(x)$, just adapt the previous Newton algorithms, replacing F with ∇f and J with $\nabla^2 f$ and using either the *ared*/*pred* conditions (cf. [75], [97]) or the Goldstein–Armijo conditions to determine acceptability of steps.

But there are some important considerations.

Slide 53

Newton's Method with Backtracking:

Given $t \in (0,1)$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.

Evaluate $f(x)$ and $\nabla f(x)$.

Iterate:

Decide whether to stop or continue.

Solve $\nabla^2 f(x)s = -\nabla f(x)$.

Evaluate $f(x+s)$.

While $ared < t \cdot pred$, do:

Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.

Update $s \leftarrow \theta s$, re-evaluate $f(x+s)$.

Update $x \leftarrow x + s$ and $f(x) \leftarrow f(x+s)$

Evaluate $\nabla f(x+s)$ and update $\nabla f(x) \leftarrow \nabla f(x+s)$.

Slide 54

In the *backtracking* . . .

- $ared \equiv f(x) - f(x+s)$ and $pred \equiv -\nabla f(x)^T s - \frac{1}{2} s^T \nabla^2 f(x) s$.
- We can substitute the Goldstein–Armijo α-condition $f(x+s) \le f(x) + \alpha \nabla f(x)^T s$ for the $ared/pred$ condition.
  - — We need $0 < \alpha < \frac{1}{2}$ so the Newton step will be acceptable near a minimizer [32, Th. 6.3.4].
  - — The β-condition is typically not used. Instead, starting with the Newton step (or a nearby step – see below) and modifying it with safeguarded backtracking ensure that steps aren't too short.
- In choosing $\theta \in [\theta_{\min}, \theta_{\max}]$, we minimize a quadratic/cubic interpolating polynomial as before.
  - — First reduction: Minimize over $[\theta_{\min}, \theta_{\max}]$ a quadratic $p(\theta)$ satisfying $p(0) = f(x)$, $p(1) = f(x+s)$, $p'(0) = \frac{d}{d\theta} f(x+\theta s)\big|_{\theta=0} = \nabla f(x)^T s$.
  - — Subsequent reductions: Minimize either this quadratic or a cubic that interpolates also a past value of f; see [32].
- In the while-loop, we only need to re-evaluate f, not ∇f.

Slide 55

In the *solve step* . . .

Very important: $s^N = -\nabla^2 f(x)^{-1} \nabla f(x)$ *is guaranteed to be a descent direction* $\iff$ $\nabla^2 f(x)$ *is positive definite.*

Away from a minimizer, we may need to *perturb* $\nabla^2 f(x)$ to obtain a symmetric positive definite B so that $s = -B^{-1} \nabla f(x)$ is a descent direction.

Idea (see [32, Alg. A5.5.1] for details):

- Begin the Cholesky decomposition of $\nabla^2 f(x)$.
- As necessary, add positive diagonal elements to obtain $L L^T = \nabla^2 f(x) + D$, where $D = \text{diag}\,(d_1, \dots, d_n)$ is such that each $d_i \geq 0$ and $L L^T$ is well-conditioned.
- Then use the Gerschgorin Theorem to compute δ less than the smallest eigenvalue of $\nabla^2 f(x)$.
- Then take $\mu = \min\{|\delta|,\ \max\{d_i\}\}$ and set $B = \nabla^2 f(x) + \mu I$.
- Finally, solve $Bs = -\nabla f(x)$ to obtain a descent direction s.

Slide 56

c. Trust region methods.

We will now explore a second way of determining acceptable steps, the *trust region* approach.

First, note possible shortcomings of the backtracking approach.

Backtracking initially tries the Newton step s^N, chosen so that $F(x) + J(x)\, s^N = 0$. Then the steplength is reduced as necessary until an acceptable step if found.

If F is "badly behaved," . . .

- Many steplength reductions may be required, entailing unproductive effort.
- The step may achieve relatively little reduction in $\|F\|$, compared to other steps of the same length but different directions.

Slide 57

Expand: Suppose $\|\cdot\| = \|\cdot\|_2$ and set $f(x) \equiv \frac{1}{2}\|F(x)\|_2^2$. Then $\nabla f(x) = J(x)^T F(x)$ and

$$\left(\frac{\nabla f(x)}{\|\nabla f(x)\|_2}\right)^T \left(\frac{s^N}{\|s^N\|_2}\right) = \left(\frac{J(x)^T F(x)}{\|J(x)^T F(x)\|_2}\right)^T \left(\frac{-J(x)^{-1} F(x)}{\|-J(x)^{-1} F(x)\|_2}\right)$$

$$= \frac{-\|F(x)\|_2^2}{\|J(x)^T F(x)\|_2 \, \|J(x)^{-1} F(x)\|_2}.$$

For an unfortunate combination of $F(x)$ and $J(x)$, this can be $\approx 1/\kappa_2(J(x))$, where $\kappa_2(J(x)) \equiv \|J(x)\|_2 \, \|J(x)^{-1}\|_2$.

So, *if $J(x)$ is ill-conditioned, s^N may be a very weak descent direction for $\|F\|$*.

Slide 58

The trust region idea.

At each iteration ...

- We have $\delta > 0$ such that we "trust" the local linear model $F(x) + J(x)\, s$ within the region $N_\delta(x) \equiv \{x + s : \ \|s\| \le \delta\}$.
- Choose a step s *ideally* to minimize $\|F(x) + J(x)\, s\|$ over all steps of length $\le \delta$.
- If this step is not acceptable, reduce δ and try again.
- Once an acceptable step has been found, consider adjusting δ for the next step.

Slide 59

General Trust Region Method [37, §4]**:**

Given $0 < t \leq u < 1$, $\delta > 0$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.
Evaluate $F(x)$.
Iterate:
  Decide whether to stop or continue.
  Choose $s \in \arg\min_{\|w\| \leq \delta} \|F(x) + J(x)\, w\|$.
  Evaluate $F(x+s)$.
  While $ared < t \cdot pred$, do:
    Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.
    Update $\delta \leftarrow \theta\delta$.
    Choose a new $s \in \arg\min_{\|w\| \leq \delta} \|F(x) + J(x)\, w\|$; re-evaluate $F(x+s)$.
  Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.
  If $ared \geq u \cdot pred$, choose $\theta \geq 1$; else choose $\theta \geq \theta_{\min}$.
  Update $\delta \leftarrow \theta\delta$.

Slide 60

- This is a *long* way from a practical algorithm. *The biggest issue will be approximating* $s \in \arg\min_{\|w\| \leq \delta} \|F(x) + J(x)\, w\|$.

Proposition: *If* $J(x)$ *is nonsingular and* $s \in \arg\min_{\|w\| \leq \delta} \|F(x) + J(x)\, w\|$, *then*

(i) $\|s^N\| \leq \delta \Rightarrow s = s^N$, (ii) $\|s^N\| \geq \delta \Rightarrow \|s\| = \delta$.

Proof: Since s^N is the unique global minimizer of $\|F(x) + J(x)\, w\|$, (i) is immediate. To show (ii), suppose $\|s^N\| \geq \delta$ and $s < \delta$. Then $\|F(x) + J(x)\, s\| > 0$ and, for $0 < \epsilon < 1$,

$$h_\epsilon \equiv J(x)^{-1} \left\{ -\epsilon \left[F(x) + J(x)\, s \right] \right\}$$

satisfies

$$\|F(x) + J(x)\,(s + h_\epsilon)\| = (1 - \epsilon)\|F(x) + J(x)\, s\| < \|F(x) + J(x)\, s\|.$$

Since $s < \delta$, $\|s + h_\epsilon\| < \delta$ for sufficiently small $\epsilon > 0$, yielding a contradiction.

Slide 61

- The algorithm can break down in the while-loop if $J(x)$ is singular.

Example: For $F : \mathbb{R}^1 \to \mathbb{R}^1$ given by $F(x) = 1 + x^2$, if $x = 0$, then for any s we have $pred = 0$ and $ared < 0$.

- The algorithm does not break down if $J(x)$ is nonsingular.

Proposition: *Suppose* $J(x)$ *is nonsingular and* $\|J(x+s) - J(x)\| \le \lambda \|s\|$ *for* $\|s\| \le \delta$. *Then for* $s \in \arg\min_{\|w\| \le \delta} \|F(x) + J(x)\, w\|$,

$$ared \ge \left[1 - \frac{\lambda \|J(x)^{-1}\|}{2} \delta\right] \, pred.$$

Remarks:

- ▷ The while-loop terminates successfully no later than when $\delta \le \frac{2(1-t)}{\lambda \|J(x)^{-1}\|}$.
- ▷ Lipschitz continuity of J is not necessary but gives prettier result.

Slide 62

Handy Lemma: *Suppose* $\|J(x+s) - J(x)\| \le \lambda \|s\|$ *whenever* $\|s\| \le \delta$. *Then*

$$\|F(x+s) - F(x) - J(x)\, s\| \le \frac{\lambda}{2} \|s\|^2$$

whenever $\|s\| \le \delta$.

Proof:

$$\begin{aligned} \|F(x+s) - F(x) - J(x)\, s\| &= \left\| \int_0^1 \frac{d}{dt} F(x+ts)\, dt - J(x)\, s \right\| \\ &= \left\| \left\{ \int_0^1 [J(x+ts) - J(x)] \, dt \right\} s \right\| \\ &= \left\{ \int_0^1 \lambda t \|s\| \, dt \right\} \|s\| = \frac{\lambda}{2} \|s\|^2. \end{aligned}$$

Slide 63

Proof of the Proposition:

$$\begin{aligned} ared &= \|F(x)\| - \|F(x+s)\| \geq \|F(x)\| - \|F(x) + J(x)\,s\| - \|F(x+s) - F(x) - J(x)\,s\| \\ &\geq pred - \frac{\lambda}{2}\|s\|^2. \end{aligned}$$

Since $\|s\| \leq \delta$, the desired inequality then follows from ...

Claim: $\|s\| \leq \|J(x)^{-1}\| \cdot pred$.

Case 1: If $\|s^N\| \leq \delta$, then $s = s^N$ and

$$\|s\| = \|s^N\| = \| - J(x)^{-1}F(x)\| \leq \|J(x)^{-1}\|\,\|F(x)\| = \|J(x)^{-1}\| \cdot pred.$$

Case 2: If $\|s^N\| > \delta$, then

$$\begin{aligned} pred &= \|F(x)\| - \|F(x) + J(x)\,s\| \geq \|F(x)\| - \|F(x) + J(x)\left(\frac{\|s\|}{\|s^N\|}\,s^N\right)\| \\ &= \|F(x)\| - \|\left(1 - \frac{\|s\|}{\|s^N\|}\right)F(x)\| = \|F(x)\| - \left(1 - \frac{\|s\|}{\|s^N\|}\right)\|F(x)\| \\ &= \frac{\|s\|}{\|s^N\|}\|F(x)\| = \frac{\|F(x)\|}{\| - J(x)^{-1}F(x)\|}\|s\| \geq \frac{1}{\|J(x)^{-1}\|}\|s\|, \end{aligned}$$

and again $\|s\| \leq \|J(x)^{-1}\| \cdot pred$.

Slide 64

- For $s \in \arg\min_{\|w\|\leq\delta} \|F(x) + J(x)\,w\|$, ...
  - ▷ We have $pred \equiv \|F(x)\| - \|F(x) + J(x)\,s\| \geq 0$. Thus an accepted step satisfies $ared \geq t \cdot pred \geq 0$.
  - ▷ If $J(x)$ is nonsingular and $F(x) \neq 0$, then $pred > 0$.

    Proof: If $\|s^N\| \geq \delta$, then $s = s^N$ and $pred = \|F(x)\|$. If $\|s^N\| < \delta$, then

$$\begin{aligned} pred &= \|F(x)\| - \|F(x) + J(x)\,s\| \leq \|F(x)\| - \|F(x) + J(x)\,(\delta\|s^N\|^{-1}s^N)\| \\ &= \|F(x)\| - \|(1 - \delta\|s^N\|^{-1})F(x)\| = \delta\|s^N\|^{-1}\|F(x)\| > 0. \end{aligned}$$

- If $\|v\|^2 = \langle v, v\rangle$ for $v \in I\!R^n$ and $pred > 0$ for any s, then s is a *descent direction* for $\|F\|$ at x.

  Proof:
$$\begin{aligned} \left.\frac{d}{d\theta}\|F(x+\theta s)\|^2\right|_{\theta=0} &= 2\,\langle F(x), J(x)\,s\rangle \\ &= 2\,\langle F(x), F(x) + J(x)\,s\rangle - 2\|F(x)\|^2 \\ &\leq 2\|F(x)\|\,\|F(x) + J(x)\,s\| - 2\|F(x)\|^2 \\ &= -2\|F(x)\| \cdot pred < 0. \end{aligned}$$

Slide 65

Def.: $x \in \mathbb{R}^n$ is a *stationary point* of $\|F\|$ if $\|F(x)\| \leq \|F(x) + J(x)\, s\|$ for every $s \in \mathbb{R}^n$.

Note: If $F(x) \neq 0$, then x is a stationary point $\iff$ there exists no inexact Newton step.

Theorem [37, Th. 4.4]: *Suppose $\{x_k\}$ is a sequence produced by the General Trust Region Method. Then every limit point of $\{x_k\}$ is a stationary point of $\|F\|$. If x_* is a limit point of $\{x_k\}$ such that $J(x_*)$ is nonsingular, then $F(x_*) = 0$, $x_k \to x_*$, and $s_k \equiv x_{k+1} - x_k = -J(x_k)^{-1}F(x_k)$ for all sufficiently large k.*

Slide 66

Practical trust region algorithms.

"Easy" details are much as in backtracking ...

- *Choose t small, e.g., $t = 10^{-4}$*.
- *Choose $\theta_{\min} = .1$, $\theta_{\max} = .5$*.
- *Choosing $\theta \in [\theta_{\min}, \theta_{\max}]$*.

Suppose we have an unsatisfactory trial step s. The approach to choosing θ is similar to that in backtracking, but the goal is to reduce δ rather than $\|s\|$.

First, do: *If $\|s\| < \delta$, update $\delta \leftarrow \|s\|$*.

This may save pointless passes through the while-loop.

Slide 67

Choosing $\theta \in [\theta_{\min}, \theta_{\max}]$ (cont.)

Suppose $||v||^2 = \langle v, v \rangle$ for $v \in I\!R^n$.

As before, choose θ by minimizing a quadratic or cubic polynomial that interpolates $g(\theta) \equiv ||F(x + \theta s)||^2$.

First step reduction: We have $g(0)$, $g'(0)$, $g(1)$. Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a quadratic that interpolates these.

Subsequent step reductions: We have $g(0)$, $g'(0)$, $g(1)$, and a third value of g; Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a cubic that interpolates these.

As before, see [32] for details.

Slide 68

Details of the quadratic interpolation.

Much as before, but we no longer have $s = \lambda s^N \Rightarrow$ slight differences.

We want $p(\theta)$ such that

- $p(0) = g(0) = ||F(x)||^2$,
- $p'(0) = g'(0) = 2\langle F(x), J(x)s \rangle$,
- $p(1) = g(1) = ||F(x + s)||^2$.

Then $p(\theta) = ||F(x)||^2 + 2\langle F(x), J(x)s \rangle \theta + d\theta^2$, where

$$d = ||F(x + s)||^2 - ||F(x)||^2 - 2\langle F(x), J(x)s \rangle .$$

So ...

- If $p''(\theta) = 2d \leq 0$, take $\theta = \theta_{\max}$.
- If $p''(\theta) = 2d > 0$, we have $p'(\theta) = 0 \iff \theta = -\langle F(x), J(x)s \rangle / d$, so choose this θ, correcting if necessary to be in $[\theta_{\min}, \theta_{\max}]$.

- *Updating δ for the next step*.

Follow [32] and prescribe ...

(i) $\delta \leftarrow 2\delta$ if *good agreement* between F and the local linear model,

(ii) $\delta \leftarrow \delta$ if *so-so agreement* ...,

(iii) $\delta \leftarrow \delta/2$ if *poor agreement* ....

Slide 69

Suppose we have u with $t \leq u < 1$.

Choose v with $t \leq v \leq u < 1$.

Updating procedure:

(i) $\delta \leftarrow 2\delta$ if $ared \geq u \cdot pred$;

(ii) $\delta \leftarrow \delta$ if $u \cdot pred > ared \geq v \cdot pred$;

(iii) $\delta \leftarrow \delta/2$ if $v \cdot pred > ared$.

Recommendations in [32]: $u = .75$, $v = .1$ with $t = 10^{-4}$.

- *Determining the trust region step*.

Issue: $s \in \arg\min_{\|w\| \leq \delta} \|F(x) + J(x)\, w\|$ *cannot be determined exactly*.

Our task is to determine an adequate approximation at reasonable cost.

Slide 70

Begin by characterizing this (exact) s.

- Already know $\|s^N\| \leq \delta \Rightarrow s = s^N$ and $\|s^N\| > \delta \Rightarrow \|s\| = \delta$.

Assume throughout: $\|\cdot\| = \|\cdot\|_2$

Similar developments hold for any other inner-product norm.

Slide 71

Lemma: *If $J(x)$ is nonsingular, then $s \in \arg\min_{\|w\|_2 \le \delta} \|F(x) + J(x)\, w\|_2$ is given by*

$$s = s(\mu) \equiv -\left[J(x)^T J(x) + \mu I\right]^{-1} J(x)^T F(x)$$

for a unique $\mu \ge 0$, as follows:

$$\begin{cases} \|s^N\|_2 \le \delta & \Rightarrow \mu = 0, \\ \\ \|s^N\|_2 > \delta & \Rightarrow \mu > 0 \text{ uniquely determined by } \|s(\mu)\|_2 = \delta. \end{cases}$$

Proof: Case 1: If $\|s^N\|_2 \le \delta$, then $s = s^N = s(0)$.

Case 2: If $\|s^N\|_2 > \delta$, then we know $\|s\|_2 = \delta$. Setting $\ell(s) \equiv \frac{1}{2}\|F(x) + J(x)\, s\|_2^2$, we also know $\nabla \ell(s) = J(x)^T F(x) + J(x)^T J(x) s \ne 0$ since $s \ne s^N$, and we must have $\nabla \ell(s) = -\mu s$ for some $\mu > 0$ since $s \in \arg\min_{\|w\|_2 \le \delta} \|F(x) + J(x)\, w\|_2$. It follows that $\left[J(x)^T J(x) + \mu I\right] s = -J(x)^T F(x)$, i.e., $s = s(\mu)$, for <u>some</u> μ such that $\|s(\mu)\|_2 = \delta$. It follows from the Proposition and Corollary below that this μ is unique.

Slide 72

Proposition:

(i) $s(\mu)$ *is differentiable and* $s'(\mu) = -\left[J(x)^T J(x) + \mu I\right]^{-1} s(\mu)$.

(ii) $\phi(\mu) \equiv \|s(\mu)\|_2^2 = s(\mu)^T s(\mu)$ *is differentiable and*

$$\phi'(\mu) = 2 s(\mu)^T s'(\mu) = -2 s(\mu)^T \left[J(x)^T J(x) + \mu I\right]^{-1} s(\mu) < 0.$$

Proof: Suppose $A(\mu)$ is any differentiable, invertible matrix-valued function of a scalar μ. Then for small $\Delta\mu \ne 0$,

$$\begin{aligned} \frac{1}{\Delta\mu}\left\{A(\mu+\Delta\mu)^{-1} - A(\mu)^{-1}\right\} &= A(\mu+\Delta\mu)^{-1}\left\{\frac{A(\mu) - A(\mu+\Delta\mu)}{\Delta\mu}\right\} A(\mu)^{-1} \\ &\to -A(\mu)^{-1} A'(\mu) A(\mu)^{-1} \quad \text{as } \Delta\mu \to 0. \end{aligned}$$

Applying this with $A(\mu) = J(x)^T J(x) + \mu I$ and noting that $A'(\mu) = I$, we conclude that (i) holds, and (ii) follows immediately.

Corollary: $\|s(\mu)\|_2$ *is monotone decreasing in* μ, *with* $\|s(0)\|_2 = \|s^N\|_2$ *and* $\lim_{\mu\to\infty} \|s(\mu)\|_2 = 0$.

Slide 73

Summary observations.

- $s(\mu) \equiv -\left[J(x)^T J(x) + \mu I\right]^{-1} J(x)^T F(x)$ traces out a differentiable curve of trust region steps.
- For $\delta \geq \|s^N\|_2$, the step is $s(0) = s^N$.
- As $\delta \to 0$, $\mu \to \infty$ and $\|s(\mu)\|_2 \to 0$ monotonically.
- For small δ, μ is large and $s(\mu) \approx -\frac{1}{\mu} J(x)^T F(x)$, a short step in the steepest descent direction for $\|F\|_2$ at x.
- Fundamental practical difficulty: *We cannot determine exactly an $s(\mu)$ such that $\|s(\mu)\|_2 = \delta$.*

Slide 74

Approach 1: **The Levenberg–Marquardt (“hook” step) approach**.

Idea: Determine $s = s(\mu)$ *exactly* for a μ such that $\|s(\mu)\|_2$ is *approximately* δ.

Implementation: Set $\Phi(\mu) \equiv \|s(\mu)\|_2 - \delta$, and use a special iteration to approximately solve $\Phi(\mu) = 0$.

- There is no need for great accuracy. The recommendation in [32, §6.4.1] is to terminate the iteration as soon as $\frac{3}{4}\delta \leq \|s(\mu)\|_2 \leq \frac{3}{2}\delta$.
- Each iteration requires $O(n^3)$ arithmetic operations; this may be expensive.
- See [32, §6.4.1] for further details.

Slide 75

Approach 2: **The dogleg approach**.

Idea: Determine s such that $\|s\|_2 = \delta$ *exactly* on a curve that *approximates* the $s(\mu)$-curve $\{s(\mu) : \ 0 \le \mu < \infty\}$.

Implementation: Approximate the $s(\mu)$-curve with the *dogleg curve* Γ_{DL}, the polygonal curve connecting $s = 0$, $s = s^{SD}$ (defined below), and $s = s^N$. Then determine s on the dogleg curve such that $\|s\|_2 = \delta$ (easily done).

Def.: s^{SD} is the minimizer of $\ell(s) \equiv \frac{1}{2}\|F(x) + J(x)\,s\|_2^2$ in the steepest descent direction $-\nabla\ell(0) = -J(x)^T F(x)$, the *steepest descent point*.

Easily determined:

$$s^{SD} = -\frac{\|-J(x)^T F(x)\|_2^2}{\|J(x)J(x)^T F(x)\|_2^2} J(x)^T F(x).$$

Slide 76

- The $s(\mu)$-curve and Γ_{DL} both begin at $s = 0$ and end at $s = s^N$.
- Since $-\nabla\ell(0) = -J(x)^T F(x)$ and $s(\mu) \approx -\frac{1}{\mu} J(x)^T F(x)$ for small μ, *the $s(\mu)$-curve and Γ_{DL} are tangent at $s = 0$.*
- The tangent direction $-\nabla\ell(0) = -J(x)^T F(x)$ is also the steepest descent direction for $\|F\|_2$ at x.
- **Facts** (see [32, §6.4.2]): Along the dogleg curve ...

  (i) $\|F(x) + J(x)\,s\|_2$ is monotone strictly decreasing,

  (ii) $\|s\|_2$ is monotone strictly increasing.

Corollary: *For $0 \le \delta \le \|s^N\|_2$, there is a unique $s \in \Gamma_{DL}$ such that $\|s\|_2 = \delta$; furthermore, $s = \arg\min_{w \in \Gamma_{DL},\ \|w\|_2 \le \delta} \|F(x) + J(x)\,w\|_2$.*

Slide 77

Computing the dogleg step:

Assume $s^N = -J(x)^{-1}F(x)$ has already been computed. Then ...

1. If $\|s^N\|_2 \le \delta$, then $s = s^N$.
2. If $\|s^N\|_2 > \delta$, then do:
   i. Compute s^{SD}.
   ii. If $\|s^{SD}\|_2 \ge \delta$, then $s = \frac{\delta}{\|s^{SD}\|_2} s^{SD}$.
   iii. If $\|s^{SD}\|_2 < \delta$, then $s = s^{SD} + \tau(s^N - s^{SD})$, where τ is uniquely determined by $\|s^{SD} + \tau(s^N - s^{SD})\|_2 = \delta$.

Computing τ: We want $\|s^{SD} + \tau(s^N - s^{SD})\|_2^2 - \delta^2 = 0$, i.e., $a\tau^2 + 2b\tau + c = 0$, where $a = \|s^N - s^{SD}\|_2^2$, $b = (s^{SD})^T(s^N - s^{SD})$, and $c = \|s^{SD}\|_2^2 - \delta^2$. We know $a > 0$ and $c < 0$; also $b > 0$ (see [32, §6.4.2]). We want $0 < \tau < 1$, so τ is given by the "+" root in the quadratic formula: $\tau = \left(-b + \sqrt{b^2 - ac}\right)/a$. However, to avoid possible loss of significance through cancellation, use the alternative formula $\tau = -c/\left(b + \sqrt{b^2 - ac}\right)$.

Slide 78

Computing the Dogleg Step:

Given $F(x)$, $J(x)$, $s^N = -J(x)^{-1}F(x)$, and $\delta > 0$.
If $\|s^N\|_2 \le \delta$, set $s = s^N$.
Else do:
  Compute $s^{SD} = -\frac{\|J(x)^T F(x)\|_2^2}{\|J(x)J(x)^T F(x)\|_2^2} J(x)^T F(x)$.
  If $\|s^{SD}\|_2 \ge \delta$, set $s = \frac{\delta}{\|s^{SD}\|_2} s^{SD}$.
  Else do:
    Evaluate $a = \|s^N - s^{SD}\|_2^2$, $b = (s^{SD})^T(s^N - s^{SD})$, $c = \|s^{SD}\|_2^2 - \delta^2$
     and $\tau = \frac{-c}{b + \sqrt{b^2 - ac}}$.
    Set $s = s^{SD} + \tau(s^N - s^{SD})$.

The outline of the dogleg method on the next slide uses quadratic minimization in reducing the trust region radius δ. Cubic minimization can be used after the first reduction if desired.

Recommendations: $t = 10^{-4}$, $v = .1$, $u = .75$, $\theta_{\min} = .1$, and $\theta_{\max} = .5$; initial δ = norm of the initial Newton step.

Slide 79

The Dogleg Method:

Given $0 < t \le v \le u < 1$, $\delta > 0$, $0 < \theta_{\min} < \theta_{\max} < 1$, and an initial x.
Evaluate $F(x)$.
Iterate:
  Decide whether to stop or continue.
  Compute s to be the dogleg step for δ.
  Evaluate $F(x+s)$.
  While $ared < t \cdot pred$, do:
    If $\|s^N\|_2 < \delta$, update $\delta \leftarrow \|s^N\|_2$.
    If $d \equiv \|F(x+s)\|_2^2 - \|F(x)\|_2^2 - 2F(x)^T J(x)s \le 0$, set $\theta = \theta_{\max}$.
    Else do:
      Set $\theta = -F(x)^T J(x)s/d$.
      If $\theta > \theta_{\max}$, $\theta \leftarrow \theta_{\max}$; if $\theta < \theta_{\min}$, $\theta \leftarrow \theta_{\min}$.
    Update $\delta \leftarrow \theta\delta$.
    Update s to be the dogleg step for the new δ; re-evaluate $F(x+s)$.
  Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.
  If $ared \ge u \cdot pred$ and $\|s^N\|_2 > \delta$, update $\delta \leftarrow 2\delta$.
  Else if $ared < v \cdot pred$, update $\delta \leftarrow \delta/2$.

Slide 80

Concluding remarks:

- A *double dogleg* variation, introduced in [31], is recommended in [32]. This introduces an additional point on the curve to provide an "earlier bias" toward the Newton direction.

- Which is better — Levenberg–Marquardt or dogleg?
  - ▷ Levenberg–Marquardt gives marginally better approximations of the exact trust-region step but requires $O(n^3)$ arithmetic beyond that required to evaluate s^N.
  - ▷ Dogleg methods require only $O(n^2)$ arithmetic but produce slightly inferior approximate trust-region steps.
  - ▷ Dogleg methods are usually preferred when linear algebra costs are dominant, i.e., when n is large or function evaluations are cheap.

Slide 81

Topic 4

Quasi-Newton (Secant Update) Methods

a. General principles and properties; the Broyden update.

b. Other updates.

c. Special methods for large-scale problems: sparsity preserving updates, limited-memory methods, considerations for PDE problems.

Slide 82

a. General principles and properties; the Broyden update.

Quasi-Newton methods are often considered to be anything of the general form

$$x_+ = x - B^{-1}F(x), \qquad B \approx J(x).$$

- This includes Newton's method, finite-difference Newton's method, modified Newton (chord) methods, etc.
- Here, we will follow common traditional usage and use "quasi-Newton method" to refer to "secant update methods" (cf. [32]).
- In practice, we must augment the general form with "globalizations," but we will consider only this "local" form here.

Slide 83

Motivation: Standard Newton's method

$$x_+ = x - J(x)^{-1}F(x)$$

has very fast (usually quadratic) local convergence, but ...

- evaluating $J(x) \Rightarrow$ up to n^2 scalar function evaluations, may be infeasible;
- solving $J(x)\, s = -F(x) \Rightarrow$ up to $O(n^3)$ arithmetic operations.

General goal: Develop quasi-Newton methods in which $B \approx J(x)$ is maintained by *updating* to incorporate enough information to give adequately fast convergence while avoiding most arithmetic and function-evaluation expense.

Specific goal: Develop methods for general problems that require at each iteration only $O(n^2)$ *arithmetic operations* and *no J-evaluations*, and which exhibit *superlinear* local convergence.

Slide 84

For guidance, consider the secant method for $F: \mathbb{R}^1 \rightarrow \mathbb{R}^1$.

$$x_+ = x - B^{-1}F(x), \qquad B = \frac{F(x) - F(x_-)}{x - x_-}.$$

Looks promising: Doesn't require $J(x)$, exhibits superlinear convergence.

This suggests: In general, require that B_+ satisfy the *secant equation*

$$\boxed{B_+\, s = y, \quad \text{where } s = x_+ - x, \quad y = F(x_+) - F(x).}$$

Note: This uniquely determines B_+ only if $n = 1$.

Slide 85

How to determine B_+ when $n > 1$?

Least-change principle: *Make the least possible change in B to obtain B_+ that satisfies the secant condition.*

Later, we may augment this with the requirement that B_+ satisfy not only the secant condition but also specified "auxiliary conditions" that reflect the structure of $J(x)$, such as symmetry or a particular pattern of sparsity.

Rationale: B presumably has useful information about $J(x)$; alter this as little as possible while incorporating new information expressed in the secant condition.

Slide 86

How to interpret the least-change principle?

"Minimal-rank" interpretation: Make a change in B of the *lowest possible rank* to obtain B_+ satisfying the secant condition.

The rank-one updates $B_+ = B + uv^T$ that satisfy the secant condition are of the form

$$B_+ = B + \frac{(y - Bs)w^T}{w^T s}, \qquad w^T s \neq 0.$$

for w such that $w^T s \neq 0$.

The most successful of these is the *Broyden update* [17]

$$B_+ = B + \frac{(y - Bs)s^T}{s^T s},$$

regarded as the *most effective update for general problems*.

Slide 87

Shortcomings: The minimal-rank interpretation ...

- fails to distinguish the Broyden update from other rank-one updates,
- is inappropriate for deriving some updates, such as the rank-n sparsity-preserving updates,
- does not lend itself to understanding and analysis of method behavior.

Slide 88

"Minimal-norm" interpretation: Make the least possible change in B *as measured in an appropriate matrix norm* to obtain B_+ satisfying the secant condition.

To implement this, we need an inner-product matrix norm.

Use the Frobenius norm and inner product: For A, $B \in \mathbb{R}^{n\times n}$, ...

$$\langle A, B\rangle_{\mathcal{F}} = \sum_{1\le i,j\le n} A_{ij}B_{ij} = \text{trace } \{AB^T\},$$

$$\|A\|_{\mathcal{F}} = \langle A, A\rangle_{\mathcal{F}}{}^{1/2} = \sqrt{\sum_{1\le i,j\le n} A_{ij}^2} = \text{trace } \{AA^T\}^{1/2}.$$

Slide 89

Define $\mathcal{Q}(y,s) \equiv \left\{M \in I\!R^{n\times n}: \;\; Ms = y\right\}$.

Goal: To obtain $B_+ \in \mathcal{Q}(y,s)$ for which $\|B_+ - B\|_{\mathcal{F}}$ is minimal, i.e.,

$$B_+ = \arg\min_{\bar{B}\in\mathcal{Q}(y,s)} \|\bar{B} - B\|_{\mathcal{F}}.$$

Proposition:

$$\mathcal{Q}(y,s) = \left\{\frac{ys^T}{s^Ts} + M: \;\; Ms = 0\right\} = \frac{ys^T}{s^Ts} + \mathcal{N},$$

where $\mathcal{N} = \left\{M \in I\!R^{n\times n}: \;\; Ms = 0\right\}$, *the annihilators of* s.

Slide 90

Note:

- $\mathcal{N}$ is a subspace of $I\!R^{n\times n}$.
- $\frac{ys^T}{s^Ts} \in \mathcal{Q}(y,s)$ and $\frac{ys^T}{s^Ts} \in \mathcal{N}^\perp$, i.e., $\left\langle \frac{ys^T}{s^Ts}, M\right\rangle_{\mathcal{F}} = 0$ whenever $M \in \mathcal{N}$.

Thus $\mathcal{Q}(y,s)$ is an *affine subspace* of $I\!R^{n\times n}$, with

— *parallel subspace* $\mathcal{N}$,

— *normal element* $\frac{ys^T}{s^Ts}$.

Slide 91

Then

$$B_+ = \arg\min_{\bar{B}\in\mathcal{Q}(y,s)} \|\bar{B} - B\|_{\mathcal{F}} = \mathcal{P}_{\mathcal{Q}(y,s)} B,$$

where $\mathcal{P}_{\mathcal{Q}(y,s)}$ is *orthogonal projection* onto $\mathcal{Q}(y,s)$ with respect to $\langle\cdot,\cdot\rangle_{\mathcal{F}}$.

In general: If $\mathcal{A} = n + \mathcal{S}$ is an affine subspace with normal n and parallel subspace $\mathcal{S}$, then

$$P_{\mathcal{A}} v = n + P_{\mathcal{S}} v$$

So

$$B_+ = \mathcal{P}_{\mathcal{Q}(y,s)} B = \frac{ys^T}{s^T s} + P_{\mathcal{N}} B.$$

Slide 92

Proposition: *For any* $B \in \mathbb{R}^{n\times n}$,

$$P_{\mathcal{N}} B = B\left[I - \frac{ss^T}{s^T s}\right].$$

Proof: Just verify that

(i) $P_{\mathcal{N}}^2 = P_{\mathcal{N}}$, (ii) $\text{Range } P_{\mathcal{N}} = \mathcal{N}$, (iii) $\langle P_{\mathcal{N}} A, B\rangle_{\mathcal{F}} = \langle A, P_{\mathcal{N}} B\rangle_{\mathcal{F}}$.

Then we have

$$B_+ = \frac{ys^T}{s^T s} + B\left[I - \frac{ss^T}{s^T s}\right] = B + \frac{(y - Bs)s^T}{s^T s},$$

the **Broyden update** again.

Slide 93

Broyden's Method:

Given initial x and B.

Evaluate $F(x)$.

Iterate:

Decide whether to stop or continue.

Solve $Bs = -F(x)$.

Evaluate $F(x+s)$ and set $y = F(x+s) - F(x)$.

Update $B \leftarrow B + \frac{(y - Bs)s^T}{s^T s}$.

Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.

Slide 94

Properties:

- Doesn't require $J(x)$; only requires one F-evaluation per iteration.
- Can be implemented in $O(n^2)$ arithmetic operations per iteration after an initial $O(n^3)$ investment.
  - ▷ Form B^{-1} initially; update at each iteration using the Sherman–Morrison–Woodbury formula [96], [111].
  - ▷ Better: Form $B = QR$ initially; update the Q and R factors at each iteration [49], [36].

  See [32, §8.3] for details.

Slide 95

- *Superlinear local convergence*.

Theorem [21], [33]: *Suppose F is Lipschitz continuously differentiable at x_*, and that $F(x_*) = 0$ and $J(x_*)$ is nonsingular. Then for x_0 sufficiently near x_* and B_0 sufficiently near $J(x_*)$, $\{x_k\}$ produced by Broyden's method is well-defined and converges q-superlinearly to x_*. Moreover, $\{B_k\}$ and $\{B_k^{-1}\}$ are well-defined and bounded.*

Slide 96

b. Other updates.

The Broyden update imposes only the secant condition at each iteration.

We might also want to impose special structure, e.g,

— symmetry,

— positive definiteness,

— sparsity,

— etc.

Extend the least-change approach leading to the Broyden update to a general procedure that will allow incorporating such structure **and** will lead to methods with *local superlinear convergence*.

Slide 97

Suppose we have

- $\langle \cdot, \cdot \rangle$, $\| \cdot \| = \langle \cdot, \cdot \rangle^{1/2}$ on $\mathbb{R}^{n \times n}$,

  — in practice, usually $\langle \cdot, \cdot \rangle_{\mathcal{F}}$, $\| \cdot \|_{\mathcal{F}}$ or "weighted" versions,

- an *affine subspace* $\mathcal{A} \in \mathbb{R}^{n \times n}$ that reflects known structure of J,

  — in practice, usually a *subspace*.

Proposition: *If $\mathcal{A} \cap \mathcal{Q}(y, s) \neq \emptyset$, then $\mathcal{A} \cap \mathcal{Q}(y, s)$ is an affine subspace.*

Proposition: *If $J(x) \in \mathcal{A}$ for all x, then $\mathcal{A} \cap \mathcal{Q}(y, s) \neq \emptyset$.*

Slide 98

If $\mathcal{A} \cap \mathcal{Q}(y, s) \neq \emptyset$, take

$$\boxed{B_+ = \mathcal{P}_{\mathcal{A} \cap \mathcal{Q}(y,s)} B}$$

where $\mathcal{P}_{\mathcal{A} \cap \mathcal{Q}(y,s)}$ is the orthogonal projection onto $\mathcal{A} \cap \mathcal{Q}(y, s)$.

- If $\mathcal{A} \cap \mathcal{Q}(y, s) = \emptyset$, see [33], [34].

- B_+ is the **least-change secant update** of B in $\mathcal{A}$ with respect to $\| \cdot \|$.

Slide 99

Illustration: Suppose we want a *symmetry-preserving update*, i.e.,

$$B = B^T \quad \Longrightarrow \quad B_+ = B_+^T.$$

Take $\quad \|\cdot\| = \|\cdot\|_{\mathcal{F}}, \quad \mathcal{A} = \mathcal{S} \equiv \{M \in I\!R^{n\times n} : \; M = M^T\}$.

We want $B_+ = \mathcal{P}_{\mathcal{S}\cap\mathcal{Q}(y,s)} B$.

In general: If $\mathcal{A}_1 = n_1 + \mathcal{S}_1$ and $\mathcal{A}_2 = n_2 + \mathcal{S}_2$, then

- $\mathcal{A}_1 \cap \mathcal{A}_2 = n + \mathcal{S}_1 \cap \mathcal{S}_2$.
- the normal n is characterized by $n \in \mathcal{A}_1 \cap \mathcal{A}_2$, $n \perp \mathcal{S}_1 \cap \mathcal{S}_2$.
- $\mathcal{P}_{\mathcal{A}_1\cap\mathcal{A}_2} v = n + \mathcal{P}_{\mathcal{S}_1\cap\mathcal{S}_2} v$.

Slide 100

Recall: $\quad \mathcal{Q}(y,s) = \dfrac{ys^T}{s^Ts} + \mathcal{N}$, where $\mathcal{N} = \{M \in I\!R^{n\times n} : \; Ms = 0\}$.

So:

$$B_+ = \mathcal{P}_{\mathcal{S}\cap\mathcal{Q}(y,s)} B = N + \mathcal{P}_{\mathcal{S}\cap\mathcal{N}} B,$$

where $N \in \mathcal{S} \cap \mathcal{Q}(y,s)$ and $N \perp \mathcal{S} \cap \mathcal{N}$.

Claim 1:

$$\mathcal{P}_{\mathcal{S}\cap\mathcal{N}} B = \begin{cases} \left(I - \frac{ss^T}{s^Ts}\right)\left(\frac{B+B^T}{2}\right)\left(I - \frac{ss^T}{s^Ts}\right) & \text{for general } B, \\ \left(I - \frac{ss^T}{s^Ts}\right) B \left(I - \frac{ss^T}{s^Ts}\right) & \text{if } B = B^T. \end{cases}$$

Proof: Just verify that

(i) $P^2_{\mathcal{S}\cap\mathcal{N}} = P_{\mathcal{S}\cap\mathcal{N}}$, (ii) Range $P_{\mathcal{S}\cap\mathcal{N}} = \mathcal{S}\cap\mathcal{N}$, (iii) $\langle P_{\mathcal{S}\cap\mathcal{N}} A, B\rangle_{\mathcal{F}} = \langle A, P_{\mathcal{S}\cap\mathcal{N}} B\rangle_{\mathcal{F}}$.

Slide 101

Claim 2:

$$N = \frac{sy^T + ys^T}{s^T s} - \frac{s^T y \; ss^T}{(s^T s)^2}.$$

Proof: We want $N \in \mathcal{S} \cap \mathcal{Q}(y, s)$ and $N \perp \mathcal{S} \cap \mathcal{N}$, so

$$\begin{aligned} 0 &= \mathcal{P}_{\mathcal{S}\cap\mathcal{N}} N = \left(I - \frac{ss^T}{s^T s}\right) N \left(I - \frac{ss^T}{s^T s}\right) \\ &= N - \frac{ss^T N}{s^T s} - \frac{N ss^T}{s^T s} + \frac{ss^T}{s^T s} N \frac{ss^T}{s^T s} \\ &= N - \frac{sy^T}{s^T s} - \frac{ys^T}{s^T s} + \frac{s^T y \; ss^T}{(s^T s)^2} . \end{aligned}$$

Slide 102

From Claims 1 and 2, we obtain

$$B_+ = N + \mathcal{P}_{\mathcal{S}\cap\mathcal{N}} B = B + \frac{(y - Bs)s^T + s(y - Bs)^T}{s^T s} - \frac{s^T(y - Bs)ss^T}{s^T s},$$

the *Powell symmetric Broyden update* [85].

Slide 103

There are *many variations* on the least-change secant update theme.

There are least-change **inverse** secant updates, in which we make a minimal-norm change *in* B^{-1} to obtain an matrix satisfying the secant condition and any auxiliary conditions, expressed as

$$B_+^{-1} = \mathcal{P}_{\mathcal{A} \cap \mathcal{Q}(s,y)} B^{-1} .$$

So far, we have considered only **fixed-scale** updates, in which the inner-product norm is does not depend on iteration-dependent information.

There are also **(iteratively) rescaled** updates, in which the inner-product norm is updated at each iteration to reflect current information about natural problem scaling.

Slide 104

I. Least-change secant updates. $\quad B_+ = \mathcal{P}_{\mathcal{A} \cap \mathcal{Q}(y,s)} B.$

A. Fixed-scale updates obtained with $\|M\| = \|M\|_{\mathcal{F}} = \sqrt{\mathrm{trace}\ \{MM^T\}}$.

1. $\mathcal{A} = \boldsymbol{R}^{n \times n} \Longrightarrow$ Broyden update, a.k.a. "good" or "first" Broyden update [17]

$$B_+ = B + \frac{(y - Bs)s^T}{s^T s}.$$

2. $\mathcal{A} = \{M \in \boldsymbol{R}^{n \times n} : \; M = M^T\} \Longrightarrow$ Powell symmetric Broyden (PSB) update [85]

$$B_+ = B + \frac{(y - Bs)s^T + s(y - Bs)^T}{s^T s} - \frac{s^T(y - Bs)ss^T}{(s^T s)^2}.$$

3. $\mathcal{A} =$ sparse matrices $\Longrightarrow$ Schubert or sparse Broyden update [93],[20]

$$B_+ = B + \sum_{i=1}^{n} \left(s_i^T s_i\right)^+ e_i e_i^T \,(y - Bs)\, s_i^T ,$$

where e_i is the i^{th} standard unit basis vector, s_i is the vector obtained by imposing on s the sparsity pattern of the i^{th} row of matrices in $\mathcal{A}$, and $(s_i^T s_i)^+ = (s_i^T s_i)^{-1}$ if $s_i \neq 0$ and $(s_i^T s_i)^+ = 0$ if $s_i = 0$.

4. $\mathcal{A} =$ sparse symmetric matrices $\Longrightarrow$ Marwil–Toint update [74], [99] (see also [32]).

Slide 105

I. Least-change secant updates (cont.). $B_+ = \mathcal{P}_{\mathcal{A}\cap\mathcal{Q}(y,s)}B$.

B. Rescaled updates obtained with $\|M\| = \|M\|_W \equiv \sqrt{\text{trace}\,\{W^{-1}MW^{-1}M^T\}}$, where $W = W^T > 0$, $Ws = y$ (assuming $y^T s > 0$).

1. $\mathcal{A} = \boldsymbol{R}^{n\times n} \Longrightarrow$ Pearson update [83].

2. $\mathcal{A} = \{M : M = M^T\} \Longrightarrow$ Davidon-Fletcher-Powell (DFP) update [27], [44]

$$B_+ = B + \frac{(y-Bs)y^T + y(y-Bs)^T}{y^T s} - \frac{s^T(y-Bs)yy^T}{(y^T s)^2}.$$

Slide 106

II. Least-change inverse secant updates. $B_+^{-1} = \mathcal{P}_{\mathcal{A}\cap\mathcal{Q}(s,y)}B^{-1}$.

A. Fixed-scale updates obtained with $\|M\| = \|M\|_{\mathcal{F}} = \sqrt{\text{trace}\,\{MM^T\}}$.

1. $\mathcal{A} = \boldsymbol{R}^{n\times n} \Longrightarrow$ "second" ("bad") Broyden update [17]

$$B_+^{-1} = B^{-1} + \frac{(s-B^{-1}y)y^T}{y^T y}, \quad \text{or} \quad B_+ = B + \frac{(y-Bs)y^T B}{y^T Bs}.$$

2. $\mathcal{A} = \{M : M = M^T\} \Longrightarrow$ Greenstadt update [54]

$$B_+^{-1} = B^{-1} + \frac{(s-B^{-1}y)y^T + y(s-B^{-1}y)^T}{y^T y} - \frac{y^T(s-B^{-1}y)}{(y^T y)^2}.$$

3. $\mathcal{A} =$ sparse or sparse symmetric matrices $\Longrightarrow$ analogues of sparse Broyden and Marwil-Toint updates (never developed, no applications).

Slide 107

II. Least-change inverse secant updates (cont.). $B_+^{-1} = \mathcal{P}_{\mathcal{A}\cap\mathcal{Q}(s,y)} B^{-1}$.

B. Rescaled updates obtained with $\|M\| = \|M\|_W \equiv \sqrt{\text{trace}\,\{W^{-1}MW^{-1}M^T\}}$, where $W = W^T > 0$, $Wy = s$ (assuming $y^T s > 0$).

1. $\mathcal{A} = \boldsymbol{R}^{n\times n} \Longrightarrow$ McCormick update (see [83])

$$B_+^{-1} = B^{-1} + \frac{(s - B^{-1}y)s^T}{y^T s}, \quad \text{or} \quad B_+ = B + \frac{(y - Bs)s^T B}{s^T Bs}.$$

3. $\mathcal{A} = \{M : M = M^T\} \Longrightarrow$ Broyden-Fletcher-Goldfarb-Shanno (BFGS) update [18], [19], [42], [51], [95],

$$B_+^{-1} = B^{-1} + \frac{(s - B^{-1}y)s^T + s(s - B^{-1}y)^T}{y^T s} - \frac{y^T(s - B^{-1}y)ss^T}{(y^T s)^2},$$

or

$$B_+ = B + \frac{yy^T}{y^T s} - \frac{Bss^T B}{s^T Bs}.$$

Slide 108

A very general local convergence analysis for methods using fixed-scale and rescaled least-change [inverse] secant updates is given in [33]. For almost all situations, the following is the main point (see [33] for precise results):

"Theorem" [33]: *Suppose F is Lipschitz continuously differentiable at x_* and that $F(x_*) = 0$ and $J(x_*)$ is nonsingular. If $J(x) \in \mathcal{A}$ [or $J(x)^{-1} \in \mathcal{A}$] for all x near x_*, then for x_0 sufficiently near x_* and B_0 sufficiently near $J(x_*)$, $\{x_k\}$ produced by a quasi-Newton method using a fixed-scale or rescaled least-change [inverse] secant update method converges q-superlinearly to x_*.*

This provides good theoretical support for using these updates, but doesn't provide a basis for preferring any one over any other.

Some updates are more successful than others in practice.

Slide 109

General recommendations.

- Use the ("good") **Broyden update** for general nonlinear equations.
- Use the **BFGS update** for unconstrained minimization.
- And keep in mind ...

— the DFP update,

— the Powell symmetric Broyden update,

— the sparse Broyden update.

And there are many additional possibilities, such as *"partially computed"* Jacobians used, e.g., in nonlinear least-squares methods [30], [33].

Slide 110

c. Special methods for large-scale problems.

The two major quasi-Newton approaches are ...

— *sparsity-preserving updates*

— *limited-memory (implicit) updating*

Slide 111

Sparsity-preserving updates.

The known updates are ...

- The Schubert or sparse Broyden update [93],[20]

$$B_+ = B + \sum_{i=1}^{n} \left(s_i^T s_i\right)^+ e_i e_i^T (y - Bs)\, s_i^T,$$

where e_i is the i^{th} standard unit basis vector, s_i is the vector obtained by imposing on s the sparsity pattern of the i^{th} row of matrices in $\mathcal{A}$, and $(s_i^T s_i)^+ = (s_i^T s_i)^{-1}$ if $s_i \neq 0$ and $(s_i^T s_i)^+ = 0$ if $s_i = 0$.

- The Marwil–Toint sparse symmetric update [74], [99] (see also [32]).

Methods using these enjoy *local q-superlinear convergence* [33, Th. 3.5].

Slide 112

Limited memory (implicit) updating.

The popular low-rank updates are desirable but straightforward implementation entails *full matrices*.

We will develop limited-memory updating for the Broyden update

$$B_+ = B + \frac{(y - Bs)s^T}{s^T s}.$$

With the Sherman–Morrison–Woodbury formula [96], [111], we have

$$\begin{aligned} B_+^{-1} &= B^{-1} + \frac{(s - B^{-1}y)s^T B^{-1}}{s^T B^{-1} y} \\ &= \left\{ I + \frac{(s - B^{-1}y)s^T}{s^T B^{-1} y} \right\} B^{-1}. \end{aligned}$$

Slide 113

By extension, if we start with B_0 and generate $B_1, \ldots, B_k$ through Broyden updating, we have

$$B_k^{-1} = \left[I + v_k w_k^T\right] \ldots \left[I + v_1 w_1^T\right] B_0^{-1}, \qquad (\star)$$

where for $i = 1, \ldots, k$,

$$v_i = \frac{s_{i-1} - B_{i-1}^{-1} y_{i-1}}{s_{i-1}^T B_{i-1}^{-1} y_{i-1}}, \qquad w_i = s_{i-1}.$$

Limited memory idea:

- Choose B_0; obtain and store a factorization.
- For $1 \leq k \leq \text{some } k_0$, create and store v_k, w_k and apply B_k^{-1} using $(\star)$.

Slide 114

Issues:

- How many vector pairs ...
  - — are needed for good performance?
  - — can we afford to store?
- What do we do when we reach the maximum number?

Note: $(\star)$ can be recast as

$$B_k^{-1} = B_0^{-1} + \sum_{i=1}^{k} \tilde{v}_i \tilde{w}_i^T$$

for appropriate $\{\tilde{v}_i, \tilde{w}_k\}$.

For more, see [14], [79], [80] and the references therein.

Slide 115

Considerations for PDE problems.

Discretized PDE problems are perhaps the most frequently encountered large-scale problems.

Straightforward implementations of quasi-Newton methods are often disappointing.

Ultimate superlinear convergence notwithstanding, the convergence of iterates often slows intolerably as the mesh is refined.

Fundamental problem: Quasi-Newton methods do not generally exhibit local superlinear convergence in function spaces!

A promising approach is (1) formulate the problem in a function space setting so that the quasi-Newton method exhibits local superlinear convergence, then (2) apply the corresponding method to the discretized problem.

See [59], [91], [56], [55], [57], [58], [65], [66], [67], [70], [70], [68], [69], [62].

Slide 116

Topic 5
Other Methods

a. Fixed-point iteration.

b. Path following (continuation, homotopy) methods.

Slide 117

a. Fixed-point iteration.

Fixed-Point Problem: $x_* = G(x_*), \quad G : \mathbb{R}^n \to \mathbb{R}^n.$

Note: $x_* = G(x_*) \iff F(x_*) = 0$, where $F(x) \equiv x - G(x)$.

Thus every fixed-point problem can be recast as a zero-finding problem, and vice versa.

However, many problems occur naturally in fixed point form, and are most easily treated in that form.

Slide 118

The natural iteration is . . .

Fixed-Point Iteration:
Given an initial x.
Until termination, do:
$x \leftarrow G(x)$

also known as *functional iteration*, *Picard iteration*, *successive substition*, . . .

We will develop some fairly standard results for this iteration typical of those found in many introductory numerical analysis texts, e.g., [6].

Slide 119

Assume throughout that $\|\cdot\|$ denotes a norm on $I\!R^n$ and also *the induced norm on* $I\!R^{n\times n}$, defined by

$$\|M\| \equiv \max_{x\neq 0} \frac{\|Mx\|}{\|x\|} = \max_{\|x\|=1} \|Mx\|.$$

Def.: G is a *contraction mapping* on $\mathcal{D} \in I\!R^n$ if there is a $\gamma \in [0,1)$ such that

$$\|G(x) - G(y)\| \leq \gamma \|x - y\|$$

for all x, $y \in \mathcal{D}$.

Note:

- G is a contraction mapping on $\mathcal{D} \iff G$ is Lipschitz continuous on $\mathcal{D}$ with Lipschitz constant $\gamma < 1$.
- G is a contraction mapping on $\mathcal{D}$ if it is differentiable and $\|G'(x)\| \leq \gamma < 1$ on $\mathcal{D}$. (More on this later.)

Slide 120

Theorem 1: *Suppose G is a contraction mapping on a closed set $\mathcal{D}$ and $G(\mathcal{D}) \subseteq \mathcal{D}$. Then there is a unique $x_* \in \mathcal{D}$ such that $x_* = G(x_*)$. Moreover, for any $x_0 \in \mathcal{D}$, the fixed-point iterates converge to x_* with $\|x_{k+1} - x_*\| \leq \gamma \|x_k - x_*\|$ for each k.*

Proof: Suppose $x_0 \in \mathcal{D}$ is given. If we have defined $x_k \in \mathcal{D}$ for some k, then $x_{k+1} = G(x_k) \in \mathcal{D}$; thus by an easy induction, the fixed-point iterates are well-defined and remain in $\mathcal{D}$. We have $\|x_{k+1} - x_k\| \leq \gamma \|x_k - x_{k-1}\| \leq \ldots \leq \gamma^{k-1}\|x_1 - x_0\|$, whence for a positive integer ℓ,

$$\|x_{k+\ell} - x_k\| \leq \left(\gamma^{k+\ell-1} \ldots \gamma^{k-1}\right)\|x_1 - x_0\| = \gamma^{k-1}\left(\sum_{j=0}^{\ell} \gamma^j\right)\|x_1 - x_0\| \leq \frac{\gamma^{k-1}}{1-\gamma}\|x_1 - x_0\|.$$

It follows that $\{x_k\}$ is a Cauchy sequence and, since $\mathcal{D}$ is closed, that there is an $x_* \in \mathcal{D}$ such that $x_k \to x_*$. We have $G(x_*) = \lim_{k\to\infty} G(x_k) = \lim_{k\to\infty} x_{k+1} = x_*$, so x_* is a fixed point. To show uniqueness, suppose $\hat{x}_* = G(\hat{x}_*)$ for some $\hat{x}_* \in \mathcal{D}$. Then

$$\|\hat{x}_* - x_*\| = \|G(\hat{x}_*) - G(x_*)\| \leq \gamma \|\hat{x}_* - x_*\|.$$

Since $\gamma < 1$, this can hold only if $\hat{x}_* = x_*$. Finally, we note that

$$\|x_{k+1} - x_*\| = \|G(x_k) - G(x_*)\| \leq \gamma \|x_k - x_*\|,$$

and the proof is complete.

Slide 121

Theorem 2: *Suppose $x_* = G(x_*)$ and that G is continuously differentiable near x_* with $\|G'(x_*)\| < 1$. Then for any η such that $\|G'(x_*)\| < \eta < 1$, there is a $\delta > 0$ such that if $\|x_0 - x_*\| \le \delta$, then the fixed-point iterates converge to x_* with $\|x_{k+1} - x_*\| \le \eta \|x_k - x_*\|$ for each k.*

Proof: Suppose we have η such that $\|G'(x_*)\| < \eta < 1$, and let $\delta > 0$ be such that $\|G'(x)\| \le \eta$ whenever $x \in N_\delta(x_*) = \{y : \ \|y - x_*\| \le \delta\}$. Then for any $x,\ y \in N_\delta(x_*)$,

$$\begin{aligned}\|G(x) - G(y)\| &= \left\| \int_0^1 \frac{d}{dt} G(y + t(x-y))\, dt \right\| = \left\| \int_0^1 G'(y + t(x-y))(x-y)\, dt \right\| \\ &\le \int_0^1 \|G'(y + t(x-y))\|\, dt\, \|x - y\| \le \eta \|x - y\|.\end{aligned}$$

Thus G is a contraction mapping on $N_\delta(x_*)$. Moreover, if $x \in N_\delta(x_*)$, then $\|G(x) - x_*\| = \|G(x) - G(x_*)\| \le \eta\|x - x_*\| < \delta$, whence $G(x) \in N_\delta(x_*)$. Thus $G(N_\delta(x_*)) \subseteq N_\delta(x_*)$, and the theorem follows from Theorem 1.

Slide 122

The "local" result of Theorem 2 can be refined to make clear that local convergence is norm-independent, even though local q-*linear* convergence is norm-dependent in general. Toward this end, define

- $\sigma(M) = \{\lambda : \ Mx = \lambda x, \ \text{some } x \ne 0\}$, the *spectrum* of M,
- $\rho(M) = \max\limits_{\lambda \in \sigma(M)} |\lambda|$, the *spectral radius* of M.

Proposition: *If $\|\cdot\|$ is a norm on $\mathbb{R}^{n\times n}$ induced by a norm $\|\cdot\|$ on $\mathbb{R}^n$, then $\rho(M) \le \|M\|$ for every $M \in \mathbb{R}^{n \times n}$.*

Proof: If $\lambda \in \sigma(M)$, then there is an $x \ne 0$ such that $Mx = \lambda x$ *and* $\|x\| = 1$. Then $\|M\| \ge \|Mx\| = \lambda \|x\| = |\lambda|$, and it follows that $\|M\| \ge \max_{\lambda \in \sigma(M)} |\lambda| = \rho(M)$.

Lemma [63, p. 12]: *For a given $M \in \mathbb{R}^{n\times n}$ and $\epsilon > 0$, there is a norm $\|\cdot\|$ on $\mathbb{R}^n$ for which the induced norm $\|\cdot\|$ on $\mathbb{R}^{n\times n}$ satisfies $\|M\| \le \rho(M) + \epsilon$.*

Slide 123

The following result shows that local convergence to a fixed point x_* is determined by $\rho(G'(x_*))$ and not by any particular norm.

Theorem 3: *Suppose $x_* = G(x_*)$ and that G is continuously differentiable near x_* with $\rho(G'(x_*)) < 1$. Then for any η such that $\rho(G'(x_*)) < \eta < 1$, there is a norm $\|\cdot\|$ on $\mathbb{R}^n$ and a $\delta > 0$ such that if $\|x_0 - x_*\| \leq \delta$, then the fixed-point iterates converge to x_* with $\|x_{k+1} - x_*\| \leq \eta \|x_k - x_*\|$ for each k.*

Proof: By the above lemma, there is a norm $\|\cdot\|$ on $\mathbb{R}^n$ such that $\|G'(x_*)\| < \eta$ for the induced norm $\|\cdot\|$ on $\mathbb{R}^{n\times n}$. With this norm, the theorem follows from Theorem 2.

Slide 124

Application 1. Suppose we have an ODE initial value problem $y' = f(t, y)$, $y(0) = y_0$. Numerically solving this using a backward differentiation formula method requires solving at the mth time step a system

$$y_m = h\beta_0 f(t_m, y_m) + a_m, \qquad a_m \equiv \sum_{j=1}^{q} \alpha_j y_{m-j},$$

to obtain $y_m \approx y(t_m)$, where h is the time step and β_0, $\alpha_1, \ldots, \alpha_q$ are method coefficients. (See, e.g., [14, §1.1].)

This suggests the fixed-point iteration

$$y_m^{(k+1)} = h\beta_0 f(t_m, y_m^{(k)}) + a_m,$$

with $y_m^{(0)}$ given by an explicit "predictor" method.

Here, $G(y) = h\beta_0 f(t_m, y) + a_m$ and $G'(y) = h\beta_0 f_y(t_m, y)$. Then for a given $\|\cdot\|$, we have $\|G'(y)\| < 1$ and the iteration converges whenever

$$h < \frac{1}{\|\beta_0 f_y(t_m, y)\|}.$$

Slide 125

Application 2. Consider a Newton-like iteration

$$x_+ = x - B(x)^{-1}F(x).$$

This is of fixed-point form, with $G(x) \equiv x - B(x)^{-1}F(x)$.

Near x_* such that $F(x_*) = 0$, we have

$$G'(x) = I - B(x)^{-1}J(x) + O(\|x - x_*\|),$$

so $G'(x_*) = I - B(x_*)^{-1}J(x_*)$. It follows that the iteration is locally convergent to x_* if $\|I - B(x_*)^{-1}J(x_*)\| < 1$ for *some* induced norm on $\mathbb{R}^{n\times n}$, equivalently if $\rho\big(I - B(x_*)^{-1}J(x_*)\big) < 1$.

Note: Taking $B(x) = J(x)$ gives Newton's method, for which

$$G'(x_*) = I - J(x_*)^{-1}J(x_*) = 0.$$

It follows that Newton's method is locally and at least superlinearly convergent.

Slide 126

Remark: Theorems 1 and 2 extend beyond $\mathbb{R}^n$ to statements valid on any complete normed linear space (i.e., any Banach space). The appropriate extension of the notion of differentiability is *Fréchet differentiability*.

Application 3. Suppose we have an ODE initial value problem

$$y' = f(t, y), \qquad y(0) = y_0,$$

and we would like to show a solution exists for $0 \le t \le T$.

Assume: f is continuous and $|f(t,z) - f(t,y)| \le \lambda |z - y|$ wherever needed.

If $y(t)$ exists, then

$$y(t) = y_0 + \int_0^t f(\tau, y(\tau))\, d\tau.$$

Conversely, any continuous y satisfying this is a solution of the IVP.

Slide 127

Denote by $C[0,T]$ the set of continuous functions on $[0,T]$.

For *any* $y \in C[0,T]$, we can define a new function $G(y)$ by

$$G(y) = y_0 + \int_0^t f(\tau, y(\tau))\, d\tau.$$

Clearly $G(y) \in C[0,T]$, so $G : C[0,T] \to C[0,T]$.

For any $y \in C[0,T]$ and $\kappa > 0$, define

$$\|y\| = \max_{t \in [0,T]} e^{-\kappa t} |y(t)|.$$

This is a norm on $C[0,T]$, and $C[0,T]$ is complete in this norm.

Slide 128

Choose $\kappa > \lambda$.

Claim: $G : C[0,T] \to C[0,T]$ is a contraction mapping.

It follows that there is a unique $y \in C[0,T]$ such that

$$y(t) = G(y)(t) = y_0 + \int_0^t f(\tau, y(\tau))\, d\tau, \qquad 0 \le t \le T,$$

and this is the unique solution of our IVP on $[0,T]$.

Slide 129

Proof: For y, $z \in C[0,T]$, we have

$$\begin{aligned} |G(y)(t) - G(z)(t)| &= \left| y_0 + \int_0^t f(\tau, y(\tau))\, d\tau - y_0 - \int_0^t f(\tau, z(\tau))\, d\tau \right| \\ &\leq \int_0^t |f(\tau, y(\tau)) - f(\tau, z(\tau))|\, d\tau \\ &\leq \int_0^t \lambda |y(\tau) - z(\tau)|\, d\tau = \int_0^t \lambda e^{+\kappa\tau} e^{-\kappa\tau} |y(\tau) - z(\tau)|\, d\tau \\ &\leq \left\{ \max_{0 \leq \tau \leq t} e^{-\kappa\tau} |y(\tau) - z(\tau)| \right\} \int_0^t \lambda e^{+\kappa\tau}\, d\tau \leq \|y - z\| \int_0^t \lambda e^{+\kappa\tau}\, d\tau \\ &= \frac{\lambda}{\kappa} \left(e^{\kappa t} - 1 \right) \|y - z\| . \end{aligned}$$

Then

$$e^{-\kappa t} |G(y)(t) - G(z)(t)| \leq \frac{\lambda}{\kappa} \left(1 - e^{-\kappa t} \right) \|y - z\| \leq \frac{\lambda}{\kappa} \|y - z\|.$$

It follows that

$$\|G(y) - G(z)\| = \max_{t \in [0,T]} e^{-\kappa t} |G(y)(t) - G(z)(t)| \leq \frac{\lambda}{\kappa} \|y - z\|,$$

and G is a contraction mapping.

Slide 130

b. Path following (continuation, homotopy) methods.

Path-Following Problem: *Given* $F: \mathbb{R}^n \times \mathbb{R}^1 \to \mathbb{R}^n$, *solve* $F(x, \lambda) = 0$ *over a range of* (x, λ)*-values.*

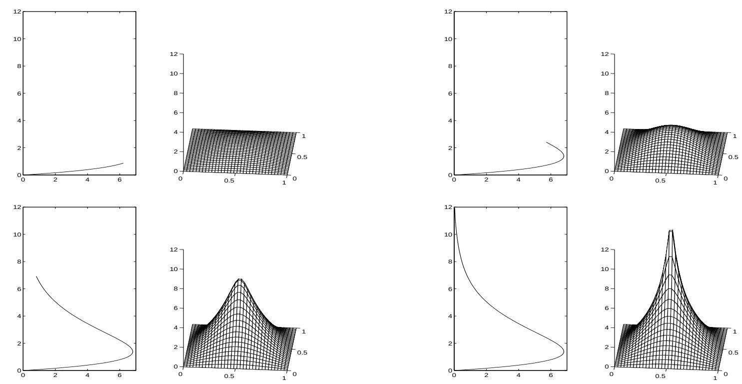
We will consider only the most basic aspects and solution methods.

- For an extensive survey, especially of mathematical aspects, see [3].
- For recent developments in software and algorithms, with many pointers to the literature, see [108].

Slide 131

Example 1: The Bratu (Gelfand) problem (see, e.g., [50], [105]).

$$\Delta u + \lambda e^u = 0 \text{ in } \mathcal{D} = [0,1] \times [0,1], \qquad u = 0 \text{ on } \partial\mathcal{D}$$

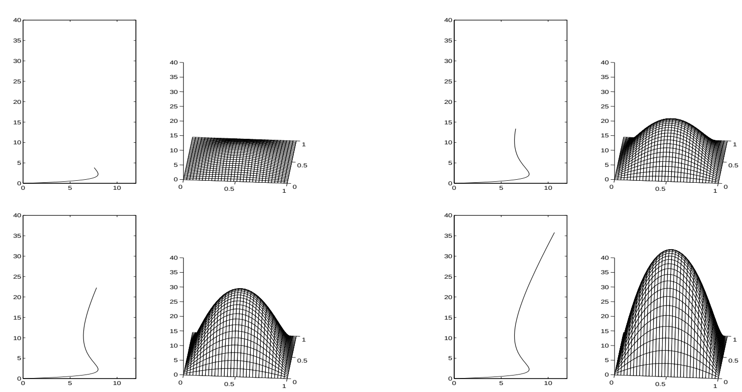


Continuation on the Bratu problem, 32×32 grid. Left: λ vs. $\|u\|_\infty$; right: u.

Slide 132

Example 2: A problem of Chan [22], [105].

$$\Delta u + \lambda \left(1 + \frac{u + u^2/2}{1 + u^2/100}\right) = 0 \text{ in } \mathcal{D} = [0,1] \times [0,1], \qquad u = 0 \text{ on } \partial\mathcal{D}.$$



Continuation on the Chan problem, 32×32 grid. Left: λ vs. $\|u\|_\infty$; right: u.

Slide 133

In considering the path following problem, goals may include . . .

- following the solution curve in detail, possibly to reach otherwise inaccessible solutions,
- determining distinguished points on the curve, such as *turning* or *fold points*, or *bifurcation points*,
- just getting from an initial point to a final point.

Slide 134

Two broad method/problem classes:

Continuation methods are generally associated with problems that involve a natural continuation parameter.

Example: Continuation in the Reynolds number in computational fluid mechanics.

Homotopy methods generally deal with artificially constructed problems that begin with an easy problem and deform it into the problem of real interest.

Example: To solve $F(x_*) = 0$, $F : \boldsymbol{R}^n \to \boldsymbol{R}^n$, construct a *homotopy map* $\rho_a(x, \lambda)$, e.g.,

$$\rho_a(x, \lambda) = \lambda F(x) + (1 - \lambda)(x - a), \qquad 0 \le \lambda \le 1.$$

Then begin with $(x, \lambda) = (a, 0)$ and follow the curve to $(x_*, 1)$.

Slide 135

We will focus on *algorithms for following the curve* that will be useful for all methods.

Convenient notation:

- $\Gamma =$ solution curve.
- $(x, \lambda) = \bar{x} \in I\!\!R^{n+1}$.
- $F(x, \lambda) = F(\bar{x})$, $F'(x, \lambda) = F'(\bar{x}) \in I\!\!R^{n \times (n+1)}$, ...
- $F'(\bar{x}) = [F_x(\bar{x}), F_\lambda(\bar{x})]$, where $F_x(\bar{x}) \in I\!\!R^{n \times n}$ and $F_\lambda(\bar{x}) \in I\!\!R^n$.

We will assume Γ is smooth, will not consider bifurcation.

In particular, we will assume $F'(\bar{x})$ is of full rank n on Γ.

Slide 136

Naive approach: Given a current $(x, \lambda) \in \Gamma$ ...

▷ Increment $\lambda \leftarrow \lambda_+$.

▷ Solve $F(x_+, \lambda_+) = 0$ for x_+.

This breaks down at turning points.

- λ_+ may be such that no x_+ exists.
- F_x is singular at turning points and ill-conditioned nearby.

We will outline methods that treat $\bar{x}$ and $F(\bar{x})$ without distinguishing λ.

Slide 137

Method framework:

1. Determine an initial $\bar{x} \in \Gamma$.
2. Advance along Γ.
   i. Predict the next point.
   ii. Correct to return to Γ.
   iii. Adjust the steplength for the next advance.
3. If necessary, perform a refined computation of the final solution.

Determining the initial $\bar{x} \in \Gamma$ is often problem-dependent and a matter of solving $F(x, \lambda) = 0$ for x, given an initial λ.

For comments on adjusting the steplength and on refined computation of the final solution, see [108].

Slide 138

Predicting the next point.

Techniques typically depend on one or more unit tangents to Γ.

Simple possibility: If $\bar{t}$ is a unit tangent at the current point $\bar{x} \in \Gamma$, then predict $\bar{x}_+ = \bar{x} + h\bar{t}$ as the next point, where h is the current steplength.

More sophisticated [108]: Assume that arclength s is computed along Γ, allowing parametrization in s. Given current and previous points $\bar{x}(s_1)$ and $\bar{x}(s_2)$ on Γ and unit tangents $\bar{t}(s_1)$ and $\bar{t}(s_2)$ at those points, determine an Hermite cubic polynomial $p(s)$ such that

$$\begin{aligned} p(s_1) &= \bar{x}(s_1), \qquad & p'(s_1) = \bar{t}(s_1) \\ p(s_2) &= \bar{x}(s_2), \qquad & p'(s_2) = \bar{t}(s_2) \end{aligned}$$

Then predict $\bar{x}_+ = p(s_1 + h)$, where h is the increment in arclength to the next point.

Slide 139

Correcting to return to Γ.

Corrector iterations typically begin with $\bar{x}_0$ determined by the predictor and produce iterates $\bar{x}_{k+1} = \bar{x}_k + \bar{s}_k$, where

$$F'(\bar{x}_k)\,\bar{s}_k = -F(\bar{x}_k).$$

This is an underdetermined system.

We will outline iterations based on two ways of specifying a unique solution.

Slide 140

The normal flow iteration.

Compute $\bar{s}_k = -F'(\bar{x}_k)^+\,F(\bar{x}_k)$, where "+" denotes Moore–Penrose pseudoinverse, i.e., the minimal-norm solution.

The resulting iteration exhibits local quadratic convergence to Γ [106].

Computing $\bar{s}_k$:

- If direct solution is preferred,
  1. Factor $F'(\bar{x}_k)^T = QR$, where $Q \in I\!R^{(n+1)\times n}$, $R \in I\!R^{n\times n}$.
  2. Solve $R^T w = -F(\bar{x}_k)$.
  3. Form $\bar{s}_k = Qw$.
- If iterative solution is preferred, see [105].

Computing the new unit tangent $\bar{t}$: Given a unit tangent $\bar{t}_0$ at a previous point, compute $\Delta\bar{t} = -F'(\bar{x}_k)^+F'(\bar{x}_k)\bar{t}_0$. Then $F'(\bar{x}_k)(\bar{t}_0 + \Delta\bar{t}) = 0$, hence $\bar{t}_0 + \Delta\bar{t}$ is a tangent. Then take $\bar{t} = (\bar{t}_0 + \Delta\bar{t})/\|\bar{t}_0 + \Delta\bar{t}\|$.

Slide 141

The augmented Jacobian iteration.

Compute $\bar{s}_k$ by

$$\begin{pmatrix} F'(\bar{x}_k) \\ \bar{t}_0^T \end{pmatrix} \bar{s}_k = \begin{pmatrix} -F(\bar{x}_k) \\ 0 \end{pmatrix},$$

where $\bar{t}_0$ is a unit tangent at a previous point.

This iteration also exhibits local quadratic convergence to Γ [106].

Computing $\bar{s}_k$: Straightforward (but watch out for bad scaling); see also [105].

Computing the new unit tangent $\bar{t}$: With $\bar{t}_0$ as above, compute $\Delta\bar{t}$ such that

$$\begin{pmatrix} F'(\bar{x}_k) \\ \bar{t}_0^T \end{pmatrix} \Delta\bar{t} = \begin{pmatrix} -F'(\bar{x}_k)\bar{t}_0 \\ 0 \end{pmatrix}.$$

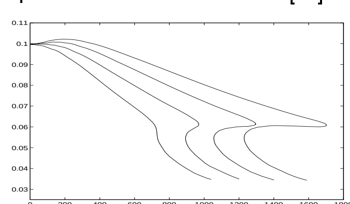
As before, $\bar{t}_0 + \Delta\bar{t}$ is a tangent, so take $\bar{t} = (\bar{t}_0 + \Delta\bar{t})/\|\bar{t}_0 + \Delta\bar{t}\|$.

Slide 142

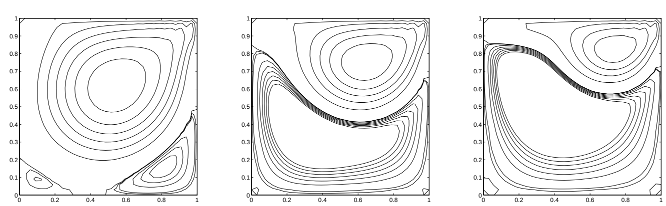
Final Example: The driven cavity problem.

$$(1/Re)\Delta^2\psi + \frac{\partial\psi}{\partial x_1}\frac{\partial}{\partial x_2}\Delta\psi - \frac{\partial\psi}{\partial x_2}\frac{\partial}{\partial x_1}\Delta\psi = 0 \text{ in } \mathcal{D} = [0,1]\times[0,1], \quad \psi = 0 \text{ and } \frac{\partial\psi}{\partial n} = g \text{ on } \partial\mathcal{D},$$

where $g = 0$ on the sides, $g = 1$ on top. The discretization was straightforward centered differences, which results in spurious solutions for low Re [92].



Re vs. $\|\psi\|_\infty$, $m \times m$ grids with $m = 24, 28, 32, 36$.



Solutions on the upper, middle, and lower branches at $Re = 1100$, 32×32 grid.

Slide 143

Introduction to Part 2
Methods for Large-Scale Problems

We'll again consider iterative methods for . . .

Problem: $F(x_*) = 0, \quad F : I\!R^n \rightarrow I\!R^n.$

and also . . .

Problem: $\min_{x \in R^n} f(x), \quad f : I\!R^n \rightarrow I\!R^1,$ recast as $\nabla f(x_*) = 0.$

Assume: Iterative linear algebra methods are preferred.

Main motivation: The case of *very large* n, probably *sparse* $J(x) = F'(x)$.

As before, theorems may not be the strongest possible; proofs are usually off-line.

Slide 144

Topic 6
Inexact Newton and Newton–Krylov Methods

a. Newton-iterative and inexact Newton methods.
   i. Formulation and local convergence.
   ii. Globally convergent methods.
   iii. Choosing the forcing terms.

b. Krylov subspace methods.

c. Newton–Krylov methods.
   i. General considerations.
   ii. Matrix-free implementations.
   iii. Adaptation to path following.

Slide 145

a. Newton-iterative and inexact Newton methods.

The model method will be ...

Newton's Method:

Given an initial x.

Iterate:

Decide whether to stop or continue.

Solve $J(x)s = -F(x)$.

Update $x \leftarrow x + s$.

Here, $J(x) = F'(x) = \left(\frac{\partial F_i(x)}{\partial x_j}\right) \in I\!R^{n\times n}$.

Slide 146

About Newton's method, recall ...

- Major strength: *quadratic local convergence*, which is often *mesh-independent* on discretized PDE problems [2].
- We've previously discussed stopping, scaling, globalization procedures, local and global convergence, etc.

Assume throughout: F is continuously differentiable.

Slide 147

Suppose that *iterative linear algebra methods* are preferred for solving

$$J(x)s = -F(x).$$

The resulting method is a **Newton iterative (truncated Newton) method**.

Key aspect: $J(x)s = -F(x)$ is *solved only approximately*.

Key issues:

- When should we stop the linear iterations?
- How should we globalize the method?
- Which linear solver should we use?

The first two can be well treated in the *strictly more general context* of *inexact Newton methods*.

Slide 148

An **inexact Newton method** [28] is *any* method each step of which reduces the norm of the local linear model of F.

> **Inexact Newton Method [28]:**
>
> Given an initial x.
>
> Iterate:
>
> Decide whether to stop or continue.
>
> Find **some** $\eta \in [0, 1)$ and s that satisfy
>
> $$\|F(x) + J(x)\,s\| \le \eta \|F(x)\|.$$
>
> Update $x \leftarrow x + s$.

Slide 149

- Our previously considered globalized Newton methods are inexact Newton methods.

- A Newton iterative method fits naturally into this framework:

  — Choose $\eta \in [0, 1)$.

  — Apply the iterative linear solver until $\|F(x) + J(x)\, s\| \le \eta \|F(x)\|$.

  ▷ Used in this way, η is called a *forcing term*.

  ▷ The issue of stopping the linear iterations becomes the issue of *choosing the forcing terms*.

Slide 150

- An inexact Newton step exists for *every* $\eta \in [0, 1) \iff F(x) \in \text{range } J(x)$.

- If $J(x)$ is nonsingular, then an inexact Newton step exists for every $\eta \in [0, 1)$.

- If $F(x) \ne 0$, then an inexact Newton step exists for *some* $\eta \in [0, 1) \iff x$ is not a *stationary point** of $\|F\|$.

- If $\|\cdot\|$ is an inner-product norm, then an inexact Newton step exists for *some* $\eta \in [0, 1) \iff F(x) \not\perp \text{range } J(x)$.

* x is a *stationary point* of $\|F\|$ if $\|F(x)\| \le \|F(x) + J(x)\, s\|$ for every s.

Slide 151

Local convergence is controlled by choices of η [28].

Theorem [28]: *Suppose $F(x_*) = 0$ and $J(x_*)$ is invertible. If $\{x_k\}$ is an inexact Newton sequence with x_0 sufficiently near x_*, then*

- $\eta_k \le \eta_{\max} < 1 \quad \Longrightarrow \quad x_k \to x_*$ *q-linearly**,
- $\eta_k \to 0 \quad \Longrightarrow \quad x_k \to x_*$ *q-superlinearly***,

*If also J is Lipschitz continuous*** at x_*, then*

- $\eta_k = O(\|F(x_k)\|) \quad \Longrightarrow \quad x_k \to x_*$ *q-quadratically*****.

* For some $\beta < 1$, $\|x_{k+1} - x_*\|_{J(x_*)} \le \beta \|x_k - x_*\|_{J(x_*)}$ for sufficiently large k, where $\|w\|_{J(x_*)} \equiv \|J(x_*)\, w\|$.

** $\|x_{k+1} - x_*\| \le \beta_k \|x_k - x_*\|$, where $\beta_k \to 0$.

*** For some λ, $\|J(x) - J(x_*)\| \le \lambda \|x - x_*\|$ for x near x_*.

**** For some C, $\|x_{k+1} - x_*\| \le C \|x_k - x_*\|^2$ for all k.

Slide 152

Proof idea:

Suppose $\|F(x) + J(x)\, s\| \le \eta \|F(x)\|$. Set $x_+ = x + s$.

We have $F(x_+) \approx F(x) + J(x)\, s \quad \Longrightarrow \quad \|F(x_+)\| \lessapprox \eta \|F(x)\|$.

Near x_* ...

$$F(x) = F(x) - F(x_*) \approx J(x_*)\,(x - x_*),$$

$$F(x_+) = F(x_+) - F(x_*) \approx J(x_*)\,(x_+ - x_*).$$

So $\|J(x_*)\,(x_+ - x_*)\| \lessapprox \eta \|J(x_*)\,(x - x_*)\|$, i.e.,

$$\|x_+ - x_*\|_{J(x_*)} \lessapprox \eta \|x - x_*\|_{J(x_*)}$$

Globally convergent methods.

A very general method is ...

Global Inexact Newton (GIN) Method [37]:

Given an initial x and $t \in (0, 1)$.

Iterate:

Decide whether to stop or continue.

Find **some** $\eta \in [0, 1)$ and s that satisfy

$$\|F(x) + J(x)\, s\| \le \eta \|F(x)\|.$$

and

$$\|F(x + s)\| \le [1 - t(1 - \eta)]\|F(x)\|.$$

Update $x \leftarrow x + s$.

Slide 153

Recall: Given $x \in I\!R^n$ and a step $s \in I\!R^n$, define

- $ared \equiv \|F(x)\| - \|F(x + s)\|$, the *actual reduction* of $\|F\|$;
- $pred \equiv \|F(x)\| - \|F(x) + J(x)s\|$, the *predicted reduction* of $\|F\|$,

A step s of the GIN method satisfies ...

$$pred \ge (1 - \eta)\|F(x_k)\| \quad \text{and} \quad ared \ge t(1 - \eta)\|F(x_k)\|$$

Compare to our earlier criterion $ared \ge t \cdot pred \ge 0$.

Slide 154

Slide 155

Existence [37, Lem. 3.1, Cor. 3.2]: There exist steps satisfying both

$$\|F(x) + J(x)\, s\| \le \eta \|F(x)\| \quad \text{and} \quad \|F(x+s)\| \le [1 - t(1-\eta)]\|F(x)\|$$

for some $\eta \in [0, 1)$ whenever ...

— there exists *any* inexact Newton step from x,

— x is not a stationary point of $\|F\|$ at which $F(x) \neq 0$.

Slide 156

The main global convergence result is ...

Theorem [37, Th.3.4]: *Suppose $\{x_k\}$ is produced by the GIN method. If $\sum_{k=0}^{\infty}(1-\eta_k) = \infty$, then $F(x_k) \to 0$. If, in addition, x_* is a limit point of $\{x_k\}$ such that $J(x_*)$ is nonsingular, then $F(x_*) = 0$ and $x_k \to x_*$.*

- The analysis previously outlined for steps satisfying $ared_k \ge t \cdot pred_k \ge 0$ is a special case obtained by *defining* $\eta_k \equiv \|F(x_k) + J(x_k)\, s_k\| / \|F(x_k)\|$, which gives $pred_k = (1-\eta_k)\|F(x_k)\|$ and $relpred_k = (1-\eta_k)$.
- The previous argument can be adapted to show the "easy" part:

$$\sum_{k=0}^{\infty}(1-\eta_k) = \infty \quad \Longrightarrow \quad F(x_k) \to 0.$$

Slide 157

As before . . .

- If $\sum_{k=0}^{\infty}(1-\eta_k) = \infty$, then exactly one of the following holds:
  - ▷ $\|x_k\| \to \infty$;
  - ▷ $\{x_k\}$ has one or more limit points, and J is singular at each of them;
  - ▷ $x_k \to x_*$ such that $F(x_*) = 0$ and $J(x_*)$ is nonsingular.
- Easy examples [37, pp. 400-401] show, depending on the problem, . . .
  - — *it is possible for each of these cases to hold;*
  - — *it may not be possible to satisfy* $\sum_{k=0}^{\infty} relpred_k = \infty$.
- Directly verifying $\sum_{k=0}^{\infty}(1-\eta_k) = \infty$ may be difficult/impossible, but we will consider algorithms for which this isn't explicitly required.

Slide 158

Application: Global approximate Newton methods [8].

Global Approximate Newton (GAN) Method [8]:

Given x_0 and $K_0 \geq 0$.

Iterate: For $k = 0, 1, 2, \ldots$

Solve $M_k \bar{s}_k = -F(x_k)$, where $M_k \approx J(x_k)$.

Choose $K_k \in [0, K_0]$.

Set $s_k = \tau_k \bar{s}_k$, where $\tau_k \equiv 1/(1 + K_k\|F(x_k)\|)$.

Set $x_{k+1} = x_k + s_k$.

Slide 159

The global convergence result of [8, §2] is based on the following assumptions:

1. $L(x_0) \equiv \{x | \, \|F(x)\| \leq \|F(x_0)\|\}$ is bounded.
2. J is invertible on $L(x_0)$, each M_k is invertible, and $\|M_k^{-1}\| \leq \kappa$ for all $k \geq 0$.
3. $\|J(y) - J(x)\| \leq \gamma \|y - x\|$ for $x,\, y \in \{u | \, \|u\| \leq \sup_{v \in L(x_0)} \|v\| + \kappa \|F(x_0)\|\}$.
4. $F(x_k) \neq 0$ and $\bar{\eta}_k \equiv \|F(x_k) + J(x_k)\bar{s}_k\| / \|F(x_k)\| \leq \bar{\eta}_0 < 1$ for all $k \geq 0$.
5. For $t \in (0, 1 - \bar{\eta}_0)$, $K_k \geq (\kappa^2 \gamma / 2)(1 - \bar{\eta}_k - t)^{-1} - \|F(x_k)\|^{-1}$ for all $k \geq 0$.

Proposition (cf. [8, p. 285, Th. 1, conclusion(i)]): *Under these assumptions, there exists an x_* such that $F(x_*) = 0$ and $x_k \to x_*$.*

Slide 160

Proof: Set $\eta_k \equiv (1 - \tau_k) + \tau_k \bar{\eta}_k \in [0, 1)$ for each k. Then

$$\|F(x_k) + J(x_k)s_k\| = \|(1 - \tau_k)F(x_k) + \tau_k[F(x_k) + J(x_k)\bar{s}_k]\| \leq \eta_k \|F(x_k)\|.$$

By [8, (2.18), p. 283], we also have $\|F(x_k + s_k)\| \leq (1 - t\tau_k)\|F(x_k)\|$.

Since $1 - t\tau_k \leq 1 - t\tau_k(1 - \bar{\eta}_k) = 1 - t(1 - \eta_k)$, this gives

$$\|F(x_k + s_k)\| \leq [1 - t(1 - \eta_k)] \, \|F(x_k)\|.$$

Thus, *Algorithm GAN is a special case of Algorithm GIN.*

To conclude, note that ...

- $1 - \eta_k = \tau_k(1 - \bar{\eta}_k) \geq \tau_k(1 - \bar{\eta}_0)$ and $\tau_k \equiv \frac{1}{1 + K_k \|F(x_k)\|} \geq \frac{1}{1 + K_0 \|F(x_0)\|}$.
- Consequently, $\sum_{k=0}^{\infty} (1 - \eta_k) = \infty$.
- Assumptions (1) and (2) imply $\{x_k\}$ has a limit point $x_* \in L(x_0)$ with $J(x_*)$ invertible.

It follows from the Theorem that $F(x_*) = 0$ and $x_k \to x_*$.

Slide 161

Move toward practical algorithms with . . .

Inexact Newton Backtracking (INB) Method [37]:

Given an initial x and $t \in (0,1)$, $\eta_{\max} \in [0,1)$, $t \in (0,1)$, and $0 < \theta_{\min} < \theta_{\max} < 1$.

Iterate:

Decide whether to stop or continue.

Choose *initial* $\eta \in [0, \eta_{\max}]$ and s such that

$\|F(x) + J(x)\, s\| \le \eta \|F(x)\|$.

Evaluate $F(x+s)$.

While $\|F(x+s)\| > [1 - t(1-\eta)]\|F(x)\|$, do:

Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.

Update $s \leftarrow \theta s$ and $\eta \leftarrow 1 - \theta(1-\eta)$.

Revaluate $F(x+s)$.

Update $x \leftarrow x + s$ and $F(x) \leftarrow F(x+s)$.

Slide 162

- This clearly lends itself to *Newton iterative implementations*.
- This becomes our previous "basic" backtracking method if initially $\eta = 0$ at each step and we define $\lambda = (1 - \eta)$.
- Can the method break down?
  - ▷ Given an initial $\eta \in [0,1)$, a suitable initial s exists if $J(x)$ is nonsingular.
  - ▷ The while-loop does not break down if $F(x) \neq 0$ or $J(x)$ is nonsingular [37, p. 410].
  - ▷ So *the method does not break down if $J(x)$ is nonsingular*.
- At each step, the final s still satisfies $\|F(x) + J(x)\, s\| \le \eta \|F(x)\|$, even if s and η are modified in the while-loop. Thus, *the method is a special case of the GIN method*.

Slide 163

The global convergence result is . . .

Theorem [37, Th.6.1]: *Suppose $\{x_k\}$ is produced by the INB method. If $\{x_k\}$ has a limit point x_* such that $J(x_*)$ is nonsingular, then $F(x_*) = 0$ and $x_k \to x_*$. Furthermore, the initial s_k and η_k are accepted for all sufficiently large k.*

Possibilities:

- $\|x_k\| \to \infty$.
- $\{x_k\}$ has limit points, and J is singular at each one.
- $\{x_k\}$ converges to x_* such that $F(x_*) = 0$, $J(x_*)$ is nonsingular, and *asymptotic convergence is determined by the initial η_k's.*

Slide 164

A more general possibility is . . .

Piecewise linear backtracking through inexact Newton steps [37, §6].

Idea: At the kth inexact Newton step, given $\eta_{\max} \in [0, 1)$, . . .

- ▹ Choose a forcing term $\eta_k \in [0, \eta_{\max}]$.
- ▹ Select approximate solutions $s_k^{(1)}, \cdots, s_k^{(m_k)}$ satisfying

$$\|F(x_k) + J(x_k)\, s_k^{(j)}\| \le \eta_k^{(j)} \|F(x_k)\|, \qquad j = 1, \cdots, m_k,$$

  where $\eta_{\max} \ge \eta_k^{(1)} > \cdots > \eta_k^{(m_k)} = \eta_k$.

- ▹ If the final approximate solution $s_k^{(m_k)}$ is not acceptable, then determine additional trial steps by *backtracking along the piecewise linear curve* joining 0, $s_k^{(1)}, \cdots, s_k^{(m_k)}$.

See [37, §6] for details.

Slide 165

- The global convergence result is . . .

Theorem [37, Th. 6.3]: *Suppose* $\{x_k\}$ *is an iteration sequence so produced. If* $\{x_k\}$ *has a limit point* x_* *such that* $J(x_*)$ *is nonsingular, then* $F(x_*) = 0$ *and* $x_k \to x_*$*. Furthermore, the initial* s_k *and* η_k *are accepted for all sufficiently large* k*.*

- This approach clearly lends itself to Newton iterative implementations.

- If we can compute $-J(x)^T F(x)$, the steepest descent direction for $\|F\|_2$ at x, then we can adapt this approach to implement a dogleg method in which $s^N = -J(x)^{-1}F(x)$ is replaced by the final approximate solution produced by the linear solver.

Slide 166

Practical implementation of the INB method.

Minor details (most as before):

- Choose $\eta_{\max}$ near 1, e.g., $\eta_{\max} = .9$.

- Choose t small, e.g., $t = 10^{-4}$.

- Choose $\theta_{\min} = .1$, $\theta_{\max} = .5$.

- Take $\|\cdot\|$ to be an inner-product norm, e.g., $\|\cdot\| = \|\cdot\|_2$.

- Choose $\theta \in [\theta_{\min}, \theta_{\max}]$ to minimize a quadratic or cubic that interpolates $\|F(x_k + \theta s_k)\|$.

Slide 167

Choosing the forcing terms.

From [28], we know ...

- $\eta_k \le$ constant $< 1 \implies$ local *linear* convergence.
- $\eta_k \to 0 \implies$ local *superlinear* convergence.
- $\eta_k = O(\|F(x_k)\|) \implies$ local *quadratic* convergence.

These allow practically implementable choices of the η_k's that lead to desirable asymptotic convergence rates.

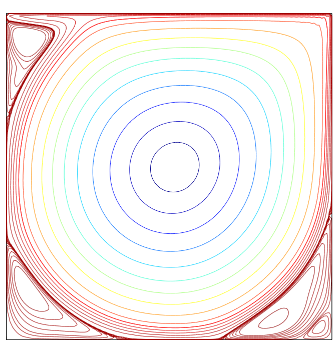
But there remains the danger of **oversolving**, i.e., *imposing an accuracy on an approximate solution s of the Newton equation that leads to significant disagreement between $F(x+s)$ and $F(x) + J(x)\,s$*.

Slide 168

Example: The driven cavity problem.

$$(1/Re)\Delta^2\psi + \frac{\partial\psi}{\partial x_1}\frac{\partial}{\partial x_2}\Delta\psi - \frac{\partial\psi}{\partial x_2}\frac{\partial}{\partial x_1}\Delta\psi = 0 \quad \text{in } \mathcal{D} = [0,1]\times[0,1],$$

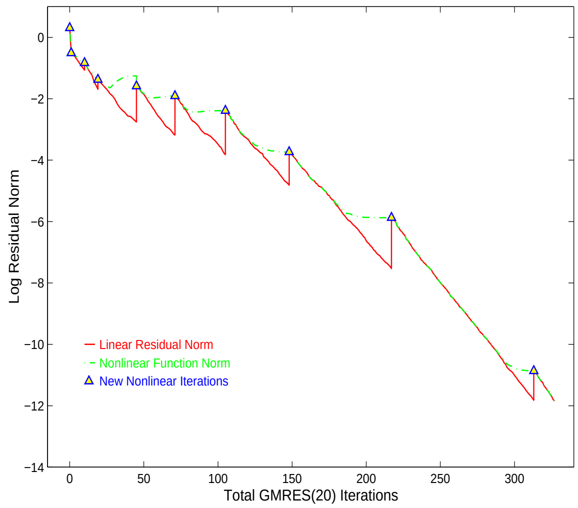
$$\text{On } \partial\mathcal{D},\ \psi = 0 \text{ and } \frac{\partial\psi}{\partial n} = \begin{cases} 1 & \text{on top.} \\ 0 & \text{on the sides and bottom.} \end{cases}$$



Streamlines for $Re = 10,000$.

Slide 169

For $\eta_k = \min\left\{\|F(x_k)\|_2, \frac{1}{k+2}\right\}$ (from [29]), ...



Performance on the driven cavity problem, $Re = 500$. *"Gaps" indicate oversolving.*

Slide 170

Forcing term choices have been proposed in [38] that are aimed at reducing oversolving. The first is ...

Choice 1: Set $\eta_k = \min\{\eta_{\max}, \tilde{\eta}_k\}$, where

$$\tilde{\eta}_k = \frac{\left| \|F(x_k)\| - \|F(x_{k-1}) + J(x_{k-1})\, s_{k-1}\| \right|}{\|F(x_{k-1})\|}$$

- This directly reflects the (dis)agreement between F and its local linear model at the previous step.
- This is invariant under multiplication of F by a scalar.

Slide 171

The local convergence theorem is . . .

Theorem [38, Th.2.2]: *Suppose $F(x_*) = 0$, $J(x_*)$ is nonsingular, and J is Lipschitz continuous at x_*. If $\{x_k\}$ is an inexact Newton sequence with x_0 sufficiently near x_* and with each η_k given by Choice 1, then $x_k \to x_*$ with*

$$\|x_{k+1} - x_*\| \leq \beta \|x_k - x_*\| \|x_{k-1} - x_*\|.$$

for some β independent of k.

- It follows that convergence is . . .
  - r-order $(1 + \sqrt{5})/2$,
  - q-superlinear,
  - two-step q-quadratic.

Slide 172

If we use η_k given by Choice 1 in the backtracking method, we can combine the above local result with the previous global result to obtain . . .

Theorem: *Suppose $\{x_k\}$ is produced by the INB method with each η_k given by Choice 1. If $\{x_k\}$ has a limit point x_* such that $J(x_*)$ is nonsingular and J is Lipschitz continuous at x_*, then $F(x_*) = 0$ and $x_k \to x_*$ with*

$$\|x_{k+1} - x_*\| \leq \beta \|x_k - x_*\| \|x_{k-1} - x_*\|.$$

for some β independent of k.

Slide 173

This and other choices in [38] may become too small too quickly away from a solution.

We recommend safeguards that work against this.

Rationale: If large forcing terms are appropriate at some point, then dramatically smaller forcing terms should be justified over several iterations before usage.

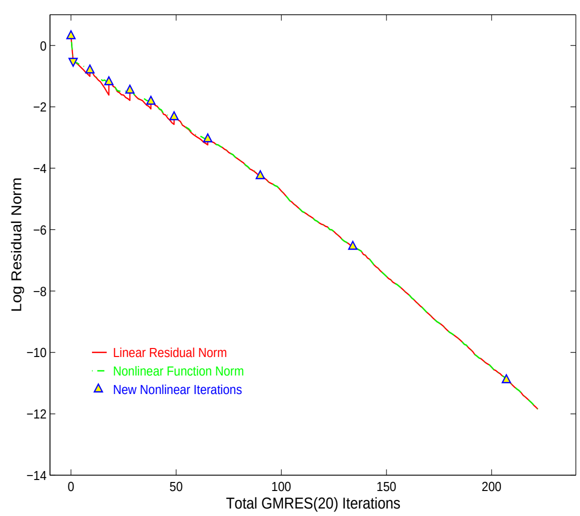
Choice 1 safeguard [38]: Modify η_k by

$$\eta_k \leftarrow \max\{\eta_k, \eta_{k-1}^{(1+\sqrt{5})/2}\}$$

whenever $\eta_{k-1}^{(1+\sqrt{5})/2} > .1$.

Slide 174

For *safeguarded Choice 1* η_k*'s*, ...



Performance on the driven cavity problem, $Re = 500$.
The inverted triangle indicates the safeguard value was used.

Slide 175

Another choice from [38] is ...

Choice 2: Set $\eta_k = \min\{\eta_{\max}, \tilde{\eta}_k\}$, where

$$\tilde{\eta}_k = \gamma \left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|} \right)^{\alpha}, \qquad 0 \le \gamma \le 1, \quad 1 < \alpha \le 2$$

- This is invariant under multiplication of F by a scalar.
- This offers a variety of local convergence rates, determined by γ and α.

Slide 176

The local convergence theorem is ...

Theorem [38, Th.2.3]: *Suppose $F(x_*) = 0$, $J(x_*)$ is nonsingular, and J is Lipschitz continuous at x_*. If $\{x_k\}$ is an inexact Newton sequence with x_0 sufficiently near x_* and with each η_k given by Choice 2, then $x_k \to x_*$ as follows:*

- ▷ *If $\gamma < 1$, then $x_k \to x_*$ with q-order α.*
- ▷ *If $\gamma = 1$, then $x_k \to x_*$ with r-order α and q-order p for every $p \in [1, \alpha)$.*

- In particular, $\alpha = 2$ and $\gamma < 1 \Longrightarrow$ *local quadratic convergence*.

Slide 177

Using η_k given by Choice 2 in the backtracking method and combining the above local result with the previous global result gives ...

Theorem: *Suppose $\{x_k\}$ is produced by the INB method with each η_k given by Choice 2. If $\{x_k\}$ has a limit point x_* such that $J(x_*)$ is nonsingular and J is Lipschitz continuous at x_*, then $F(x_*) = 0$ and $x_k \rightarrow x_*$ as follows:*

- ▷ *If $\gamma < 1$, then $x_k \rightarrow x_*$ with q-order α.*
- ▷ *If $\gamma = 1$,then $x_k \rightarrow x_*$ with r-order α and q-order p for every $p \in [1, \alpha)$.*

Choice 2 safeguard [38]: Modify η_k by

$$\eta_k \leftarrow \max\{\eta_k, \gamma\eta_{k-1}^{\alpha}\}$$

whenever $\gamma\eta_{k-1}^{\alpha} > .1$.

Slide 178

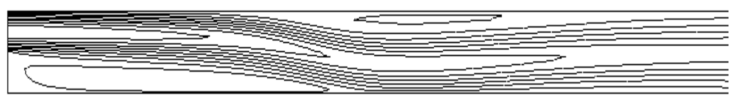
Numerical experiments on CFD problems.

- Joint work with J. N. Shadid and R. S. Tuminaro, Sandia National Labs [94].
- **Goal:** to test the effectiveness of backtracking alone and in combination with various forcing term choices.
- **Problems:** Three 2D CFD benchmark problems and two large scale 3D flow simulations.
- **PDEs:** Low Mach number Navier–Stokes equations with heat and mass transport equations as appropriate.
- **Discretization:** Pressure stabilized streamline upwind Petrov–Galerkin FEM.
- **Software:** INB implementation in the Sandia *MPSalsa* parallel reactive flow code, with GMRES routine and domain-based (overlapping Schwarz) ILU preconditioners from the Sandia *Aztec* package.
- **Machines:** Intel Paragons at Sandia National Labs.

Slide 179

The driven cavity and backward facing step.

$$\mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u}, \qquad \nabla\cdot\mathbf{u} = 0$$



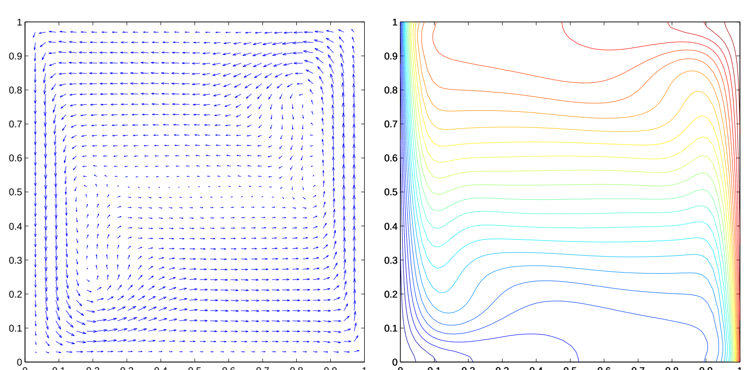
Backward facing step. Streamlines for $Re = 850$.

Slide 180

Thermal convection.

$$\frac{1}{Pr}\mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \nabla^2\mathbf{u} + Ra\,T\hat{\mathbf{g}}, \qquad \nabla\cdot\mathbf{u} = 0, \qquad \mathbf{u}\cdot\nabla T = \nabla^2 T$$

Here, $Pr = 1$.



Thermal convection. Flow and temperature contours at $Ra = 1,000,000$.

Slide 181

2D benchmark problem experiments. A robustness study ...

Forcing Term η_k	Thermal Convection		Lid Driven Cavity		Backward Facing Step	
	Easier	Harder	Easier	Harder	Easier	Harder
Choice 1	0	0	0	0	0	1
	0	1	0	5	0	4
Choice 2 $\alpha = 1.5$, $\gamma = .9$	0	0	1	1	0	3
	0	1	1	4	0	4
Choice 2 $\alpha = 2$, $\gamma = .9$	0	0	0	0	0	2
	0	1	1	5	1	4
10^{-1}	0	0	4	5	1	4
	0	1	5	5	1	2
10^{-4}	0	0	3	4	2	4
	0	1	5	5	3	4

Numbers of failures with backtracking (top rows) and without (bottom rows).

	Easier	Harder
Thermal Convection	$10^3 \leq Ra \leq 10^5$	$Ra = 10^6$
Driven Cavity	$1000 \leq Re \leq 5000$	$6000 \leq Re \leq 10000$
Backward Facing Step	$100 \leq Re \leq 500$	$600 \leq Re \leq 800$

Slide 182

2D benchmark problem experiments. An efficiency study ...

A comparison of Choices 1 and 2 (with backtracking) on problems on which all were successful (see [94] for cases).

Forcing Term η_k	Inexact Newton Steps	Back-tracks	GMRES Iterations	Time (Seconds)
Choice 1	36.5	41.4	4054.1	792.1
Choice 2, $\alpha = 1.5$, $\gamma = .9$	36.3	49.8	4189.6	824.2
Choice 2, $\alpha = 2$, $\gamma = .9$	32.8	48.5	3951.6	779.4

"Backtracks" gives arithmetic means; all other columns give geometric means.

Slide 183

Tilted CVD reactor.

- Navier-Stokes equations plus heat and mass transport. (No chemistry in these experiments.)
- 3D unstructured mesh; $384,200$ unknowns; 220 processors.

With Backtracking

Forcing Term η_k	Inexact Newton Steps	Back-tracks	GMRES Iterations	Time (Seconds)
Choice 1	25	3	1503	924.9
10^{-1}	13	1	1315	593.8
10^{-4}	5	0	1531	444.5

Without Backtracking

Forcing Term η_k	Inexact Newton Steps	Back-tracks	GMRES Iterations	Time (Seconds)
Choice 1	20	0	1052	707.9
10^{-1}	12	0	1051	511.5
10^{-4}	5	0	1531	445.5

Slide 184

Duct flow with contaminant transport.

- Navier-Stokes equations plus mass transport.
- 3D non-uniform mesh; $477,855$ unknowns; 256 processors.
- *Divergence without backtracking* for all forcing terms.
- *No failures with backtracking.*

Forcing Term η_k	Inexact Newton Steps	Back-tracks	GMRES Iterations	Time (Seconds)
Choice 1	28	6	13,450	3554.5
10^{-1}	25	7	15,477	3953.9
10^{-4}	24	6	15,360	3915.1

Performance with backtracking.

Slide 185

Summary observations.

Newton-iterative methods can be very effective on these problems, *but* ...

- A good **forcing term** choice is necessary (but not sufficient).
- **Globalization** (backtracking) is necessary (but not sufficient).
- **Many inexact Newton steps** may be necessary.
- **Very accurate Jacobians** may be necessary.
- **No strategy always works best.**

Slide 186

b. Krylov subspace methods.

Shift gears somewhat to the *linear problem* ...

Problem: $Ax = b, \quad A \in \mathbb{R}^{n\times n}, \quad b \in \mathbb{R}^n.$

Ultimate interest: $J(x)\, s = -F(x)$.

Assume throughout: A is nonsingular.

General references: Survey articles [60], [47]; books [89], [9], [53]

Slide 187

Krylov Subspace Method:

Given x_0, determine ...

$$x_k = x_0 + z_k,$$

$$z_k \in \mathcal{K}_k \equiv \operatorname{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\},$$

Terminology: $\mathcal{K}_k$ is the kth *Krylov subspace*.

There are by now *many* Krylov subspace methods, e.g., ...

CG/CR, GMRES, BCG, CGS, QMR, TFQMR (QMRCGS), QMR-squared, BiCGSTAB, BiCGSTAB2, BiCGSTAB(ℓ), QMRCGSTAB, Arnoldi (FOM/IOM), GMRESR, GCR, GMBACK, MINRES, SYMMLQ, ORTHODIR, ORTHOMIN, ORTHORES, Axelsson, SYMMBK, CGNR, CGNE, LSQR,. . . .

Slide 188

General features.

- *They require only products of A (and sometimes A^T) with vectors.* This property ...

  ... brings out the operator structure of A,

  ... may facilitate exploitation of sparsity, etc.,

  ... may allow *matrix-free* implementations.

- $\mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq \cdots$ and $\dim \mathcal{K}_k \leq k$.

Assume throughout: $r_0 \equiv b - Ax_0 \neq 0$, so $\dim \mathcal{K}_1 = 1$.

Lemma: *If A is nonsingular, then* $\dim \mathcal{K}_{k+1} \leq k \iff A^{-1}b = x_0 + z$ *for some* $z \in \mathcal{K}_k$.

- So a "smart" Krylov subspace method will find the solution in at most n steps — sounds good but may be cold comfort in practice.

Specifying z_k.

There are two *traditional criteria* ...

Minimal residual (MR): Choose $z_k \in \mathcal{K}$ to solve

$$\min_{z \in \mathcal{K}} \|b - A(x_0 + z)\|_2 = \min_{z \in \mathcal{K}} \|r_0 - Az\|_2.$$

Slide 189

Orthogonal residual (OR): Choose $z_k \in \mathcal{K}$ so that

$$r_k \equiv r_0 - Az_k \perp \mathcal{K}_k.$$

- If A is symmetric positive definite, then OR is equivalent to choosing $z_k \in \mathcal{K}_k$ to minimize

$$\|x_0 + z - A^{-1}b\|_A \equiv \sqrt{(x_0 + z - A^{-1}b)^T A(x_0 + z - A^{-1}b)}$$

General properties.

Lemma: *If A is nonsingular and $\dim \mathcal{K}_{k+1} \leq k$, then both MR and OR uniquely characterize $z_k \in \mathcal{K}_k$ such that $x_k = x_0 + z_k = A^{-1}b$.*

But ...

Slide 190

- MR uniquely characterizes z_k for *every* k.
- *Some OR steps may fail to exist* before the solution is found (but OR steps are unique if they exist).

Example: For $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $r_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, the first step fails to exist.

Also . . .

- *The residual norms of an MR method are monotone decreasing*, since

$$\mathcal{K}_{k-1} \subseteq \mathcal{K}_k \quad \Rightarrow \quad \|r_{k-1}^{\mathrm{MR}}\|_2 \geq \|r_k^{\mathrm{MR}}\|_2.$$

Slide 191

- The decrease may not be *strictly* monotone.

Example: For $A = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$ and $r_0 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$,

the method **stagnates** for the first $n-1$ steps.

- *OR residual norms may behave wildly*. Even if OR steps exist, they and their residuals may be dangerously large.

Lemma [13]: $$\|r_k^{\mathrm{OR}}\|_2 = \frac{\|r_k^{\mathrm{MR}}\|_2}{\sqrt{1 - \|r_k^{\mathrm{MR}}\|_2^2 / \|r_{k-1}^{\mathrm{MR}}\|_2^2}}$$

Slide 192

- It follows that $\|r_k^{\mathrm{OR}}\|_2 \geq \|r_k^{\mathrm{MR}}\|_2$ always, with strict inequality until the solution is found.

- If MR makes good progress at the kth step, i.e., $\|r_k^{\mathrm{MR}}\|_2 \ll \|r_{k-1}^{\mathrm{MR}}\|_2$, then $\|r_k^{\mathrm{OR}}\|_2 \approx \|r_k^{\mathrm{MR}}\|_2$ and OR makes good progress as well.

- If MR nearly stagnates, i.e., $\|r_k^{\mathrm{MR}}\|_2 \approx \|r_{k-1}^{\mathrm{MR}}\|_2$, then $\|r_k^{\mathrm{OR}}\|_2 \gg \|r_k^{\mathrm{MR}}\|_2$ and the OR residual is (perhaps dangerously) large.

- These observations underlie "peak-plateau" behavior of OR/MR method pairs [13], [24], [104],[25].

Slide 193

Computing MR and OR steps.

Choose a *basis matrix* $B_k = (b_1, \cdots, b_k)$.

Then $z \in \mathcal{K}_k \iff z = B_k y$ for some $y \in I\!R^k$, and ...

$$\left.\begin{array}{ll} \mathrm{MR}: & y_k = \arg\min_{y \in I\!R^k} \|r_0 - AB_k y\|_2 \\ \mathrm{OR}: & B_k^T r_0 = B_k^T A B_k y_k \end{array}\right\} \quad z_k = B_k y_k.$$

- The *power basis* matrix $B_k = (r_0, Ar_0, \cdots, A^{k-1} r_0)$ is often very ill-conditioned.

Slide 194

Generate a well-conditioned (orthonormal) basis with ...

Arnoldi Process [5]: (standard Gram–Schmidt version)

Given r_0.

Set $\rho_0 \equiv \|r_0\|_2$ and $v_1 \equiv r_0/\rho_0$.

For $k = 1, 2, \cdots,$ do:

  Initialize $v_{k+1} = Av_k$.

  For $i = 1, \cdots, k$, do:

    Set $h_{ik} = v_i^T v_{k+1}$.

    Update $v_{k+1} \leftarrow v_{k+1} - h_{ik} v_i$.

  Set $h_{k+1,k} = \|v_{k+1}\|_2$.

  Update $v_{k+1} \leftarrow v_{k+1}/h_{k+1,k}$.

Slide 195

- For the Arnoldi process, we have ...

$$\text{breakdown} \iff Av_k \in \mathcal{K}_k \iff \dim \mathcal{K}_{k+1} = \dim \mathcal{K}_k = k$$

$$\iff \text{OR and MR give } x_k = A^{-1}b.$$

- Setting

$$V_k \equiv (v_1, \ldots, v_k), \qquad H_k = \begin{pmatrix} h_{11} & \cdots & h_{1k} \\ h_{21} & & \vdots \\ \vdots & \ddots & \\ 0 & \cdots & h_{k+1,k} \end{pmatrix}, \qquad \bar{H}_k = \begin{pmatrix} h_{11} & \cdots & \cdots & h_{1k} \\ h_{21} & & & \vdots \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & h_{k,k-1} & h_{kk} \end{pmatrix},$$

we have

$$\begin{aligned} \text{before breakdown}: \quad & AV_k = V_{k+1}H_k \qquad \text{rank } H_k = k \\ \text{on breakdown}: \quad & AV_k = V_k\bar{H}_k \qquad \bar{H}_k \text{ nonsingular.} \end{aligned}$$

Slide 196

Since $V_k^T V_k = I_k$, it's *easy* to compute MR and OR steps, as follows ...

MR steps: $z_k = V_k y_k$, where

$$\begin{aligned} y_k &= \arg\min_{y \in \mathbb{R}^k} \|r_0 - AV_k y\|_2 \\ &= \begin{cases} \arg\min_{y \in \mathbb{R}^k} \|V_{k+1}(\rho_0 e_1 - H_k y)\|_2 & \text{before breakdown} \\ \arg\min_{y \in \mathbb{R}^k} \|V_k(\rho_0 \bar{e}_1 - \bar{H}_k y)\|_2 & \text{on breakdown} \end{cases} \\ &= \begin{cases} \arg\min_{y \in \mathbb{R}^k} \|\rho_0 e_1 - H_k y\|_2 & \text{before breakdown} \\ \arg\min_{y \in \mathbb{R}^k} \|\rho_0 \bar{e}_1 - \bar{H}_k y\|_2 & \text{on breakdown} \end{cases} \end{aligned}$$

Here, $e = (1, 0, \ldots, 0)^T \in \mathbb{R}^{k+1}$ and $\bar{e} = (1, 0, \ldots, 0)^T \in \mathbb{R}^k$.

Slide 197

OR steps: $z_k = V_k y_k$, where

$$
\begin{aligned}
0 &= V_k^T (r_0 - A V_k y_k) \\
&= \begin{cases} V_k^T V_{k+1}(\rho_0 e_1 - H_k y_k) & \text{before breakdown} \\ V_k^T V_k(\rho_0 \bar{e}_1 - \bar{H}_k y_k) & \text{on breakdown} \end{cases} \\
&= \rho_0 \bar{e}_1 - \bar{H}_k y_k.
\end{aligned}
$$

- Caution: This system is nonsingular on breakdown, but may be singular and have no solution prior to breakdown.

Slide 198

Methods when $A = A^T$.

We have ...

$$A V_k = V_{k+1} H_k \quad \Longrightarrow \quad V_k^T A V_k = V_k^T V_{k+1} H_k = \bar{H}_k$$

Then $A = A^T \quad \Longrightarrow \quad \bar{H}_k = \bar{H}_k^T \quad \Longrightarrow \quad \bar{H}_k$ and H_k are *tridiagonal*.

It follows that $Av_k = h_{k-1,k} v_{k-1} + h_{k,k} v_k + h_{k+1,k} v_{k+1}$, and *we can determine v_{k+1} using only v_{k-1}, v_k, and Av_k!*

- The Arnoldi process becomes the short-recurrence *symmetric Lanczos process* [71], [72].

- **MR and OR can be implemented with short recurrences!**

Slide 199

For A symmetric positive definite, the methods are:

- ▷ OR $\Longrightarrow$ Conjugate Gradient (CG) [61]
- ▷ MR $\Longrightarrow$ Conjugate Residual (CR) [61]

For A symmetric indefinite, the methods are:

- ▷ OR $\Longrightarrow$ SYMMLQ [82]
- ▷ MR $\Longrightarrow$ MINRES [82]

Slide 200

Conjugate Gradient Method [61]:

Given A, b, x, tol, $itmax$.

Set $r = b - Ax$, $\rho^2 = \|r\|_2^2$, $z = 0$, $\beta = 0$.

Iterate: For $itno = 1, \ldots, itmax$, do:
  If $\rho \leq tol$, go to End.
  Update $p \leftarrow r + \beta p$.
  Compute Ap.
  Compute $p^T Ap$ and $\alpha = \rho^2 / p^T Ap$.
  Update $z \leftarrow z + \alpha p$.
  Update $r \leftarrow r - \alpha Ap$.
  Update $\beta \leftarrow \|r\|_2^2 / \rho^2$ and $\rho^2 \leftarrow \|r\|_2^2$.

End: Update $x \leftarrow x + z$.

Slide 201

- In the special case of symmetric positive definite A, OR steps are always defined, and CG doesn't break down.

Theorem [53, Th. 10.2.5], [73]: *With $\kappa_2(A) \equiv \|A\|_2 \|A^{-1}\|_2$, we have*

$$\|x_k - A^{-1}b\|_A \leq 2\|x_0 - A^{-1}b\|_A \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k .$$

- This is almost always *overly pessimistic* but does correctly suggest that conditioning has something to do with convergence and that convergence is fast for well-conditioned A.

Proposition: *If A is symmetric positive definite and has $k \leq n$ distinct eigenvalues, then CG converges in at most k iterations.*

- This correctly suggests that if the eigenvalues of A are clustered around k distinct values, then CG "almost" converges after k iterations.

Slide 202

Preconditioning.

This is a *very* important, *very* vast subject. We will cover only the barest outlines here. See [89], [9], and [53] for more.

Basic idea: Instead of solving $Ax = b$ directly, apply the Krylov solver to a *preconditioned system* that can be solved more efficiently.

Typical approaches include ...

Left: solve $M^{-1}Ax = M^{-1}b$.

Right: solve $AM^{-1}y = b$, then form $x = M^{-1}y$.

Two-sided: solve $M_1^{-1}AM_2^{-1}y = M_1^{-1}b$, then form $x = M_2^{-1}y$.

Slide 203

- The *preconditioners* can be explicitly given as matrices, implicitly defined as operaters, etc.
- The traditional view is that the goal of preconditioning is to improve the conditioning of the system being solved. The real goal is always to reduce time to solution.
- For a preconditioner to be practically worthwhile, speedup in convergence must outweigh the cost of the preconditioner solves.
- Applying a preconditioner with a Krylov method is usually straightforward. Some explanation is necessary in the case of CG.

Slide 204

CG is applicable only to symmetric positive-definite systems, so begin by supposing C is a symmetric positive-definite matrix and applying CG to

$$\tilde{A}\tilde{x} = \tilde{b}, \quad \text{where} \quad \tilde{A} = C^{-1}AC^{-1}, \quad \tilde{b} = C^{-1}b.$$

Once $\tilde{x}$ has been found, recover $x = C^{-1}\tilde{x}$.

Straightforward substitution of these into the CG algorithm, followed by some algebra, results in an algorithm formulated in terms of x, A, b, etc., in which C appears only as C^2.

Setting $M \equiv C^2$ gives *the preconditioned CG algorithm*.

Slide 205

Preconditioned Conjugate Gradient Method:

Given A, b, x, tol, $itmax$, and a symmetric positive-definite M.

Set $r = b - Ax$, $w = M^{-1}r$, $\rho^2 = r^T w$, $z = 0$, $\beta = 0$.

Iterate: For $itno = 1, \ldots, itmax$, do:

If $\rho \le tol$, go to End.
Update $p \leftarrow w + \beta p$.
Compute Ap.
Compute $p^T Ap$ and $\alpha = \rho^2 / p^T Ap$.
Update $z \leftarrow z + \alpha p$.
Update $r \leftarrow r - \alpha Ap$.
Update $w = M^{-1}r$, $\beta \leftarrow r^T w/\rho^2$ and $\rho^2 \leftarrow r^T w$.

End: Update $x \leftarrow x + z$.

Slide 206

Methods for general A.

Simple, old idea: *Apply CG to the normal equations.*

- Applying CG in a straightforward way to $A^T Ax = A^T b$ gives **CGNR**, for **CG** on the **N**ormal equations with **R**esidual minimization over

$$\hat{\mathcal{K}}_k \equiv \text{span}\,\{A^T r_0, (A^T A)A^T r_0, \ldots, (A^T A)^{k-1} A^T r_0\}.$$

  This goes back to [61].

- Applying CG to $AA^T y = b$ and then taking $x = A^T y$ gives **Craig's method** [23], or **CGNE**, meaning **CG** on the **N**ormal equations with **E**rror minimization over $\hat{\mathcal{K}}_k$.

Slide 207

CGNR and CGNE are good methods for *some* problems; see [47], [78].

For many (most?) problems, convergence may be too slow because $\kappa_2(A^TA) = \kappa_2(AA^T) = \kappa_2(A)^2$.

Can we implement MR and OR with short recurrences for general A?

Faber and Manteuffel: NO! [41]

It is shown in [41] that, except for "a few anomolies," the only matrices for which MR or OR can be implemented with short recurrences are those of the form $A = e^{i\theta}(S + \sigma I)$, where $S = S^H$, $\theta \in \mathbb{R}^1$, and $\sigma \in \mathbb{C}^1$.

So we must give up either MR/OR or short recurrences.

Slide 208

First possibility: Stick with MR/OR and give up short recurrences.

We can implement MR/OR with the general Arnoldi process as previously described.

OR leads to the Arnoldi method or FOM (Full Orthogonalization Method) [87].

MR leads to **GMRES**, the Generalized Minimal REsidual Method [90].

Slide 209

Basic operation of standard GMRES.

We have $z_k = V_k y_k$, where $y_k = \arg\min_{y \in R^k} \|\rho_0 e_1 - H_k y\|_2$.

Use *Givens rotations* to factor $J_k \cdots J_1 H_k = \begin{pmatrix} R_k \\ 0 \end{pmatrix}$.

Setting $w = J_k \cdots J_1(\rho_0 e_1)$, we have

$$\|\rho_0 e_1 - H_k y\|_2 = \|J_1^T \cdots J_k^T \left[w - \begin{pmatrix} R_k \\ 0 \end{pmatrix} y \right] \|_2 = \|w - \begin{pmatrix} R_k \\ 0 \end{pmatrix} y\|_2.$$

Write $w = \begin{pmatrix} \bar{w} \\ w_{k+1} \end{pmatrix}$ for $\bar{w} \in I\!R^k$. Then $y_k = R_k^{-1}\bar{w}$ and $z_k = V_k y_k$. Furthermore,

$$\|r_k\|_2 = |w_{k+1}|$$

- **This allows monitoring $\|r_k\|_2$ (for stopping) without having to compute z_k or r_k!**

Slide 210

Basic GMRES properties.

- Monotone decreasing residual norms (but not necessarily strictly decreasing).
- Converges in $\leq n$ iterations (in exact arithmetic) but it *may stagnate* — as long as $r_0 \perp A(\mathcal{K}_k) = \text{span}\,\{Ar_0, \cdots, A^k r_0\}$.
- Carrying out k iterations costs $O(k^2 n)$ arithmetic and requires $O(kn)$ storage, plus k products of A with vectors.
- For most large-scale problems, the method implemented is **GMRES(m)**, which *restarts* as necessary with $x_0 \leftarrow x_m$ after m steps.
- **GMRES(m) may not converge.** It usually works well, but *the choice of m can be very important*.

Slide 211

GMRES(m) [90]: (standard Gram–Schmidt implementation)

Given A, b, x, tol, $itmax$.

Initialize: Set $r = b - Ax$, $v_1 = r/\|r\|_2$, $w = \|r\|_2 e_1 \in \boldsymbol{R}^{m+1}$, $itno = 0$.

Iterate: For $k = 1, \ldots, m$, do:

Set $v_{k+1} = Av_k$; update $itno = itno + 1$.

For $i = 1, \ldots, k$, do:

Set $h_{ik} = v_i^T v_{k+1}$.

Update $v_{k+1} \leftarrow v_{k+1} - h_{ik} v_i$.

Set $h_{k+1,k} = \|v_{k+1}\|_2$.

If $k > 1$, apply $J_{k-1} \cdots J_1$ to $(h_{1k}, \ldots, h_{kk}, h_{k+1,k}, 0, \ldots)^T \in \boldsymbol{R}^{m+1}$.

Determine J_k such that

$$J_k \cdots J_1 (h_{1k}, \ldots, h_{kk}, h_{k+1,k}, 0, \ldots)^T = (r_{1k}, \ldots, r_{kk}, 0, 0 \ldots)^T.$$

If $k = 1$, form $R_1 = (r_{11})$; else form $R_k = \begin{pmatrix} R_{k-1} & r^{(k)} \\ 0 & \end{pmatrix}$, where $r^{(k)} = (r_{ik}) \in \boldsymbol{R}^k$.

Update $w \leftarrow J_k w$. If $|w_{k+1}| \le tol$ or $k = m$ or $itno = itmax$, go to Solve;

else update $v_{k+1} \leftarrow v_{k+1}/h_{k+1,k}$.

Solve: Let k be the final iteration number from Iterate.

Solve $R_k y = \bar{w}$ for y, where $\bar{w} = (w_1, \ldots, w_k)^T$.

Update $x \leftarrow x + (v_1, \ldots, v_k) y$.

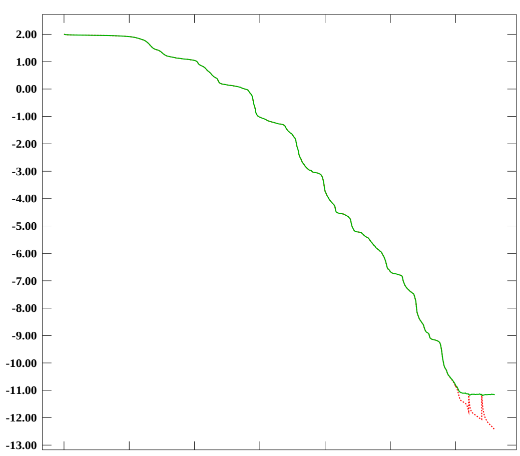
If $|w_{k+1}| \le tol$, accept x; otherwise, return to Initialize.

Slide 212

GMRES performance on a model problem.

$$\Delta u + cu + d\frac{\partial u}{\partial x} = f \quad \text{in } \mathcal{D} = [0,1] \times [0,1], \qquad u = 0 \quad \text{on } \partial\mathcal{D}.$$

- $f \equiv 1$, $c = d = 50$, 100×100 grid ($\Rightarrow n = 10^4$), double precision $\Rightarrow$ machine epsilon $\approx 10^{-16}$.
- GMRES(20), no preconditioning.
- Green (solid): $\log_{10} \|b - Ax_k\|$.
  Red (dotted): $\log_{10} |w_{k+1}|$.



Slide 213

- There have been a number of MR methods mathematically equivalent to GMRES; see, e.g., [47].
- Other GMRES variations include: Householder instead of Gram–Schmidt orthogonalization [102, 103]; "Newton basis" instead of Arnoldi basis [7]; "simpler" GMRES (the least-squares problem emerges in upper-triangular rather than Hessenberg form) [107]; "efficient high accuracy" implementations [100].
- Performance on singular or ill-conditioned systems is treated in [16].
- A variant that allows variable ("flexible") preconditioning, e.g., using an iterative method, is FGMRES [88].
- A related method (built around GMRES) is GMRESR [35].

Slide 214

Second possibility: Give up MR/OR, pursue short recurrences.

Don't forget: According to [41],

- we can't do MR/OR with short recurrences;
- short-recurrence bases can't be orthonormal.

Use the short-recurrence **nonsymmetric Lanczos process** [71] to generate a basis.

Choose $\tilde{r}_0$. (Typically $\tilde{r}_0 = r_0$.) Set $\tilde{\mathcal{K}}_k \equiv \mathrm{span}\,\{\tilde{r}_0, A^T\tilde{r}_0, \ldots, (A^T)^{k-1}\tilde{r}_0\}$.

The nonsymmetric Lanczos process generates basis matrices $V_k = (v_1, \cdots, v_k)$ and $W_k = (w_1, \cdots, w_k)$ for $\mathcal{K}_k$ and $\tilde{\mathcal{K}}_k$, respectively, as follows ...

Slide 215

Nonsymmetric Lanczos Process [71]:

Given r_0 and $\tilde{r}_0$.

Set $v_1 = r_0/\|r_0\|_2, \quad w_1 = \tilde{r}_0/\|\tilde{r}_0\|_2$.

For $k = 1, 2, \cdots$, do:

Set $v_{k+1} = Av_k, \quad w_{k+1} = A^T w_k$.

For $i = 1, \cdots, k$, do:

Set $h_{ik} = w_i^T v_{k+1}/w_i^T v_i, \quad g_{ik} = v_i^T w_{k+1}/w_i^T v_i$.

Update $v_{k+1} \leftarrow v_{k+1} - h_{ik} v_i, \; w_{k+1} \leftarrow w_{k+1} - g_{ik} w_i$.

Set $h_{k+1,k} = \|v_{k+1}\|_2, \quad g_{k+1,k} = \|w_{k+1}\|_2$.

Update $v_{k+1} \leftarrow v_{k+1}/h_{k+1,k}, \quad w_{k+1} \leftarrow w_{k+1}/g_{k+1,k}$.

Slide 216

We have:

- $v_{k+1} \perp \tilde{\mathcal{K}}_k$ and $w_{k+1} \perp \mathcal{K}_k$, so $W_k^T V_k = D_k$ (diagonal).
- $AV_k = V_{k+1} H_k$ and $A^T W_k = W_{k+1} G_k$.
- $W_k^T A V_k = W_k^T V_{k+1} H_k = D_k \bar{H}_k$ and $V_k^T A^T W_k = V_k^T W_{k+1} = D_k \bar{G}_k$.
- $\bar{H}_k = D_k^{-1} \bar{G}_k^T D_k$ and $\bar{G}_k = D_k^{-1} \bar{H}_k^T D_k$.
- $\bar{H}_k$ and $\bar{G}_k$ are tridiagonal.
- **Short recurrences!**

  The inner loop is just "For $i = \max\{1, k-1\}, k$, do:"

Slide 217

How can we use this in a solution method?

One possibility is the following variant on the OR idea.

BCG [72], [43], the **B**i**C**onjugate **G**radient method: Take $x_k = x_0 + z_k$, where $z_k \in \mathcal{K}_k$ is characterized by

$$r_k = r_0 + Az_k \perp \tilde{\mathcal{K}}_k.$$

Slide 218

Biconjugate Gradient Method [72], [43]:

Given x_0, set $q_0 = r_0 = b - Ax_0$.

Choose $\tilde{r}_0 \neq 0$ and set $\tilde{q}_0 = \tilde{r}_0$.

For $k = 1, 2, \cdots$, do:

Compute

$$\delta_k = \tilde{r}_{k-1}^T r_{k-1} / \tilde{q}_{k-1}^T A q_{k-1},$$
$$x_k = x_{k-1} + \delta_k q_{k-1},$$
$$r_k = r_{k-1} - \delta_k A q_{k-1}, \quad \tilde{r}_k = \tilde{r}_{k-1} - \delta_k A^T \tilde{q}_{k-1},$$
$$\gamma_k = \tilde{r}_k^T r_k / \tilde{r}_{k-1}^T r_{k-1},$$
$$q_k = r_k + \gamma_k q_{k-1}, \quad \tilde{q}_k = \tilde{r}_k + \gamma_k \tilde{q}_{k-1}.$$

Slide 219

BCG has *short recurrences* as desired, but ...

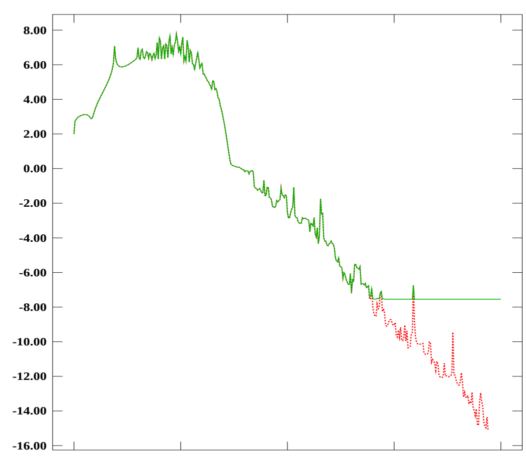
- There is a possibility of *breakdown* if ...
  - — $\tilde{q}_{k-1}^T A q_{k-1} = 0$ (breakdown of the OR criterion),
  - — $\tilde{r}_{k-1}^T r_{k-1} = 0$ (breakdown of the underlying Lanczos process).
- The method needs A^T *products* as well as A products, which may be expensive or infeasible.
- The method may produce *wildly varying residual norms*, which may be unnerving and limit attainable accuracy.

Slide 220

BCG performance on the model problem.

$$\Delta u + cu + d\frac{\partial u}{\partial x} = f \quad \text{in } \mathcal{D} = [0,1] \times [0,1], \qquad u = 0 \quad \text{on } \partial\mathcal{D}.$$

- $f \equiv 1$, $c = d = 50$, 100×100 grid ($\Rightarrow n = 10^4$), double precision $\Rightarrow$ machine epsilon $\approx 10^{-16}$.
- BCG, no preconditioning; directly (green) and recursively (red) evaluated residual norms.
- Green (solid): $\log_{10} \|b - Ax_k\|$.
  Red (dotted): $\log_{10} \|r_k\|$.



Slide 221

First, **address the problem of A^T products.**

Can we develop *"transpose free"* Lanczos methods?

CGS [98], the **C**onjugate **G**radient **S**quared method:

In BCG, A^T only appears only indirectly in the recurrences for the things we care about (x_k and r_k) through inner products $\tilde{q}_k^T A q_k$ and $\tilde{r}_k^T r_k$. There are polynomials ψ_k, ϕ_k such that ...

$$r_k = \psi_k(A) r_0, \qquad \tilde{r}_k = \psi_k(A^T)\tilde{r}_0,$$
$$q_k = \phi_k(A) r_0, \qquad \tilde{q}_k = \phi_k(A^T)\tilde{r}_0.$$

These can be "flipped" across inner products, yielding ...

$$\tilde{r}_k^T r_k = \tilde{r}_0^T \psi_k(A)^2 r_0, \qquad \tilde{q}_k^T A q_k = \tilde{r}_0^T A \phi_k(A)^2 r_0,$$

which are expressions only in A.

Slide 222

Conjugate Gradient Squared Method [98]:

Given x_0, set $p_0 = u_0 = r_0 = b - Ax_0$, $v_0 = Ap_0$.

Choose $\tilde{r}_0$ such that $\rho_0 = \tilde{r}_0^T r_0 \neq 0$.

For $k = 1, 2, \cdots$, do:

Compute

$$\sigma_{k-1} = \tilde{r}_0^T v_{k-1}, \quad \alpha_{k-1} = \rho_{k-1}/\sigma_{k-1},$$
$$q_k = u_{k-1} - \alpha_{k-1} v_{k-1},$$
$$x_k = x_{k-1} + \alpha_{k-1}(u_{k-1} + q_k),$$
$$r_k = r_{k-1} - \alpha_{k-1} A(u_{k-1} + q_k),$$
$$\rho_k = \tilde{r}_0^T r_k, \quad \beta_k = \rho_k/\rho_{k-1},$$
$$u_k = r_k + \beta_k q_k,$$
$$p_k = u_k + \beta_k(q_k + \beta_k p_{k-1}),$$
$$v_k = Ap_k.$$

Slide 223

CGS properties:

- Short recurrences; requires only A products.
- At the kth step, *two* A products are needed, resulting in $x_k \in x_0 + \mathcal{K}_{2k}$.
- Since $r_k^{\mathrm{CGS}} = \psi_k(A)^2 r_0$, the behavior of $r_k^{\mathrm{BCG}} = \psi_k(A) r_0$ tends to be accentuated.

Note: ψ_k is also the kth residual polynomial for CG as well as BCG, i.e., in the symmetric positive-definite case, $r_k^{\mathrm{CG}} = \psi_k(A) r_0$. This accounts for the name "conjugate gradient squared".

Slide 224

Now, **address the problem of wildly varying residual norms.**

One approach: **Bi-CGSTAB** [101], the **B**i-**C**onjugate **G**radient **STAB**ilized method.

The CGS residuals are given by $r_k^{\mathrm{CGS}} = \psi_k(A)^2 r_0$, where ψ_k is the kth BCG residual polynomial, i.e., $r_k^{\mathrm{BCG}} = \psi_k(A) r_0$.

Bi-CGSTAB idea: Consider more general methods with $r_k = \tilde{\psi}_k(A)\psi_k(A) r_0$.

The specific choice of $\tilde{\psi}_k$ in [101] is $\tilde{\psi}_k(t) = (1 - \omega_k t)\tilde{\psi}_{k-1}(t)$, where ω_k is chosen so that $\|r_k^{\mathrm{BiCGSTAB}}\|_2 \equiv \|(1 - \omega_k A)\tilde{\psi}_{k-1}(A)\psi_k(A) r_0\|_2$ is minimal.

Slide 225

Bi-CGSTAB properties:

- Like CGS, ...
  - ▷ Short recurrences; requires only A products.
  - ▷ At the kth step, two A products are needed, resulting in $x_k \in x_0 + \mathcal{K}_{2k}$.
- Typically produces much smoother residual norm behavior than CGS, but the residual norms still behave badly on some problems.
- There are numerous variants; see [60] for up-to-date references.

Slide 226

Another approach: **QMR** [48], the **Q**uasi-**M**inimal **R**esidual method.

For $z \in \mathcal{K}_k$ and the Lanczos basis matrix V_k, we have $z = V_k y$ and $(\rho_0 = \|r_0\|_2)$ ...

$$\|r_0 - Az\|_2 = \|r_0 - AV_k y\|_2 = \|\rho_0 v_1 - V_{k+1} H_k y\|_2 = \|V_{k+1}\left(\rho_0 e_1 - H_k y\right)\|_2.$$

QMR idea: Choose $z_k = V_k y_k$, where $y_k = \arg\min_{y \in R^k} \|\rho_0 e_1 - H_k y\|_2$.

- This would be GMRES if the columns of V_k were the orthonormal Arnoldi vectors.

QMR properties:

- Short recurrences, but requires one A and *one A^T product* per iteration.
- QMR produces residual norm sequences that are fairly smoothly (if not monotonically) decreasing. However, each QMR residual norm is usually about the same size as the best BCG residual norm so far obtained [112], [104].
- There are residual norm bounds ...

  [48]: $\|r_k^{\rm QMR}\|_2 \le \sqrt{k+1}\ \min_{y\in R^k} \|\rho_0 e_1 - H_k y\|_2.$

  [77]: $\|r_k^{\rm QMR}\|_2 \le \kappa_2(V_{k+1})\|r_k^{\rm GMRES}\|_2.$
- Breakdown is considerably alleviated through the *look-ahead Lanczos* process; see [48].
- There are numerous variants, including *transpose-free TFQMR* [46]; see [60] for up-to-date references.

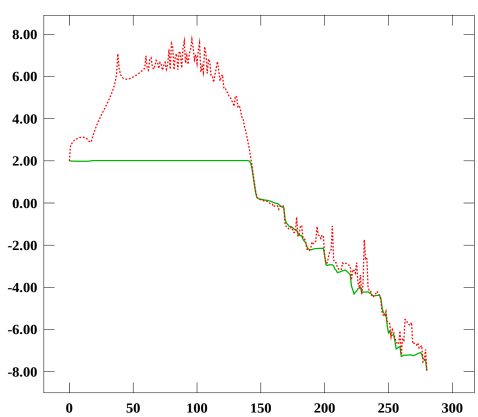
Slide 227

QMR performance on the model problem.

$$\Delta u + cu + d\frac{\partial u}{\partial x} = f \quad \text{in } \mathcal{D} = [0,1]\times[0,1], \qquad u = 0 \quad \text{on } \partial\mathcal{D}.$$

- $f \equiv 1$, $c = d = 50$, 100×100 grid ($\Rightarrow n = 10^4$), double precision $\Rightarrow$ machine epsilon $\approx 10^{-16}$.
- Green (solid): QMR, no preconditioning.
- Red (dotted): BCG, no preconditioning.

The "peak-plateau" behavior can be explained through *residual smoothing* [112], [104].



Slide 228

Slide 229

Summary of major ideas:

- Using the nonsymmetric Lanczos process to obtain short recurrences.
- "Flipping" polynomials across inner products to get rid of A^T products.
- Using the QMR and Bi-CGSTAB ideas to get fairly well-behaved residual norms.
- Using look-ahead Lanczos and similar strategies to alleviate breakdown.

See [60] for up-to-date references.

Slide 230

c. Newton–Krylov methods.

Idea: Implement a Newton iterative method using a *Krylov subspace method* as the linear solver.

- The term appears to have originated with [15].
- Naming conventions: Newton-GMRES, Newton–Krylov–Schwarz (NKS), Newton-Krylov-Multigrid (NKMG), ...
- "Truncated Newton" originated with [29], which outlined an implementation of Newton with CG.

Slide 231

General considerations.

- The linear system is $J(x)\,s = -F(x)$. The usual initial approximate solution is $s_0 = 0$.

- *The linear residual norm $\|F(x) + J(x)\,s\|$ is just the local linear model norm.*

- About **preconditioning** . . .

  - Preconditioning on the right retains compatibility between the norms used in the linear and nonlinear inexact Newton strategies.

  - *Preconditioning on the left may introduce incompatibilities.*

  - It *is* safe to "precondition the problem" on the left, i.e., to solve $M^{-1}F(x) = 0$ for an M that is *used without change throughout the solution process.*

Slide 232

- An MR method (e.g., GMRES) is *optimal* among Krylov subspace methods from the point of view of reducing $\|F(x) + J(x)\,s\|$ and thereby satisfying an inexact Newton condition in a minimal number of iterations.

- The MR property also lends itself to several trust region-like globalizations:

  - A dogleg method within the Krylov subspace [15].

  - Piecewise linear backtracking through residual minimizing steps [37, §8].

  - These strategies are compromised by any deviation from the MR principle, e.g., by *restarting* GMRES.

Slide 233

Considerations for optimization.

The linear system is $\nabla^2 f(x)\, s = -\nabla f(x)$, with exact solution

$$s^N = -\nabla^2 f(x)^{-1} \nabla f(x).$$

$\nabla^2 f(x)$ is symmetric, probably positive-definite near a minimizer.

This suggests using CG or a symmetric-Lanczos variant such as SYMMLQ [82].

Assume $\nabla^2 f(x)$ is SPD and we are applying CG.

- Recall: For $Ax = b$ with symmetric positive definite A, the kth CG iterate minimizes $\|x_0 + z - A^{-1}b\|_A$ over $z \in \mathcal{K}_k$, where $\|v\|_A \equiv \|Av\|_2$ for $v \in I\!R^n$.

Substituting $A \leftarrow \nabla^2 f(x)$, $A^{-1}b \leftarrow s^N$, $x_0 \leftarrow 0$ and $z \leftarrow s$, we have that ...

- *The kth iterate of CG applied to $\nabla^2 f(x)\, s = -\nabla f(x)$ minimizes $\|s - s^N\|_{\nabla^2 f(x)}$ over $s \in \mathcal{K}_k$.*

Slide 234

Recall the *local quadratic model* of f,

$$f(x) + \nabla f(x)^T s + \frac{1}{2}\, s^T \nabla^2 f(x) s.$$

This can be rewritten as

$$f(x) - \frac{1}{2} s^{N^T} \nabla^2 f(s) s^N + \frac{1}{2} \|s - s^N\|_{\nabla^2 f(x)}.$$

It follows that ...

- *The kth CG iterate minimizes the local quadratic model of f over $\mathcal{K}_k$.*
- *Since $r_0 = -\nabla f(x)$, the first CG iterate is the steepest descent step for f at x.*

Slide 235

These observations facilitate trust region-like globalizations.

- The usual dogleg method with $s^N = -\nabla^2 f(x)^{-1} \nabla f(x)$ replaced by the final solution produced by the linear solver.
- Truncated Newton globalizations that employ piecewise-linear backtracking through inexact Newton steps produced by CG, along the lines of that described above.

Slide 236

Matrix-free implementations.

Krylov subspace methods require only products of $J(x)$ — and sometimes $J(x)^T$ — with vectors.

There are possibilities for producing these *without creating and storing $J(x)$.*

One possibility for products involving either $J(x)$ or $J(x)^T$ is **automatic differentiation**.

This is actively being explored in the Mathematics and Computer Science Division, Argonne National Lab. See the ANL Computational Differentiation Project web page , `www-unix.mcs.anl.gov/autodiff/index.html`.

Slide 237

A very widely used technique, applicable when only products involving $J(x)$ are needed, is **finite-difference approximation**.

For a local convergence analysis, see [12].

Given $v \in I\!R^n$, formulas for approximating $J(x)v$ to 1st, 2nd, 4th, and 6th order are ...

$$\frac{1}{\delta}[F(x+\delta v) - F(x)],$$

$$\frac{1}{2\delta}[F(x+\delta v) - F(x-\delta v)],$$

$$\frac{1}{6\delta}\left[8F(x+\frac{\delta}{2}v) - 8F(x-\frac{\delta}{2}v) - F(x+\delta v) + F(x-\delta v)\right],$$

$$\frac{1}{90\delta}\left[256F(x+\frac{\delta}{4}v) - 256F(x-\frac{\delta}{4}v) - 40F(x+\frac{\delta}{2}v) + 40F(x-\frac{\delta}{2}v) + F(x+\delta v) - F(x-\delta v)\right].$$

Slide 238

- In an inexact Newton method, $F(x)$ is already available. Therefore, each of these requires a number of new F-evaluations equal to its order.
- The 1st-order formula is very commonly used; the others very rarely used (although sometimes they're needed).
- If GMRES is used as the solver, a technique in [100] can be applied to achieve the benefits of higher-order differencing at very little cost.
  - ▷ Use the higher order formula at each GMRES restart.
  - ▷ Use first-order differences thereafter until the next restart.
- The 1st, 2nd, and 4th order formulas are offered as options in NITSOL [84]. The technique of [100] is used with GMRES when the higher-order formulas are chosen.

Slide 239

Choosing δ.

- As before ...
  - ▷ We try to choose δ to roughly balance truncation and floating point error.
  - ▷ Fairly well-justified choices can be made for scalar functions. The justifications weaken with vector functions. Nothing is foolproof.
- A choice used in [84] that approximately minimizes a bound on the relative error in the difference approximation is based on ...

$$\delta = \frac{\left[(1 + \|x\|)\epsilon_F\right]^{1/(p+1)}}{\|v\|},$$

where p is the difference order and ϵ_F is the relative error in F-evaluations ("function precision"). The main underlying assumption is that F and its derivatives up to order $p+1$ have about the same scale.

- A crude heuristic is $\delta = \epsilon^{1/(p+1)}$, where ϵ is machine epsilon.

Slide 240

Adaptation to path following.

We previously considered ...

Path-Following Problem: *Given $F : \mathbb{R}^n \times \mathbb{R}^1 \to \mathbb{R}^n$, solve $F(x, \lambda) = 0$ over a range of (x, λ)-values.*

We introduced notation:

- $\Gamma =$ solution curve.
- $(x, \lambda) = \bar{x} \in \mathbb{R}^{n+1}$.
- $F(x, \lambda) = F(\bar{x})$, $F'(x, \lambda) = F'(\bar{x}) \in \mathbb{R}^{n \times (n+1)}$, ...
- $F'(\bar{x}) = [F_x(\bar{x}), F_\lambda(\bar{x})]$, where $F_x(\bar{x}) \in \mathbb{R}^{n \times n}$ and $F_\lambda(\bar{x}) \in \mathbb{R}^n$.

Slide 241

We developed *predictor-corrector* methods for following Γ.

The major linear algebra tasks were

1. Computing a corrector step $\bar{s}$ by solving the *underdetermined system*

$$F'(\bar{x})\,\bar{s} = -F(\bar{x}).$$

2. Computing a unit tangent to the curve.

We will outline ways of applying Krylov subspace methods to these.

As before, assume $F'(\bar{x})$ is of full rank n on Γ.

Slide 242

1. Computing a corrector step.

We had two approaches to computing $\bar{s}$:

- the *normal flow* approach,
- the *augmented Jacobian* approach.

In both cases, we can characterize $\bar{s}$ as satisfying

$$F'(\bar{x})\,\bar{s} = -F(\bar{x}) \quad \text{subject to} \quad \bar{t}^T\bar{s} = 0.$$

- Normal flow: $\bar{t}$ is an approximate unit null vector of $F'(\bar{x})$.
- Augmented Jacobian: $\bar{t}$ is an approximate unit tangent to Γ (i.e., an approximate unit null vector of F') at a previous point.

Slide 243

An obvious approach: Solve $\begin{pmatrix} F'(\bar{x}) \\ \bar{t}^T \end{pmatrix} \bar{s} = \begin{pmatrix} -F(\bar{x}) \\ 0 \end{pmatrix}$.

Potential difficulties:

- Ill-conditioning through unfortunate scaling.
- Krylov iterates may only approximately satisfy $\bar{t}^T \bar{s} = 0$.

We will outline the approach in [105], which avoids these.

Slide 244

The abstract approach:

1. Find $Q \in \mathbb{R}^{(n+1)\times n}$ such that
   a. $\mathrm{range}\,(Q) = \{\bar{t}\}^{\perp}$,
   b. $\|Qy\|_2 = \|y\|_2$ for all $y \in \mathbb{R}^n$.

   Then $F'(\bar{x})Q \in \mathbb{R}^{n\times n}$.
2. Apply the Krylov subspace method to solve approximately $F'(\bar{x})Qy = -F(\bar{x})$ for $y \in \mathbb{R}^n$. Then set $\bar{s} = Qy$.

With this . . .

- $\bar{s} = Qy$ automatically satisfies $\bar{t}^T \bar{s} = 0$ regardless of how well it satisfies $F'(\bar{x})\,\bar{s} = -F(\bar{x})$.
- Since $\|Qy\|_2 = \|y\|_2$ for $y \in \mathbb{R}^n$, conditioning problems are not worsened as long as $\bar{t}$ is an accurate unit null vector.

A concrete implementation:

1. Determine a Householder transformation P such that

$$P\bar{t} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^{n+1}.$$

Slide 245

2. Define Q by $Qv = P\begin{pmatrix} v \\ 0 \end{pmatrix}, \qquad v \in \mathbb{R}^n.$

3. Apply the Krylov solver to $F'(\bar{x})Qy = -F(\bar{x})$, and set $\bar{s} = Qy$.

2. Computing a unit tangent.

We want $\bar{t}$ such that $F'(\bar{x})\bar{t} = 0$.

Slide 246

Suppose we have an initial approximation $\bar{t}_0$.

Then ...

1. Solve $F'(\bar{x})\,\bar{s} = -F'(\bar{x})\,\bar{t}_0$ subject to $\bar{t}_0^T\bar{s} = 0$ as above.

2. Set $\bar{t} = (\bar{t}_0 + \bar{s})/\|\bar{t}_0 + \bar{s}\|_2$.

Slide 247

About **preconditioning** with this approach.

On difficult problems, effective preconditioners have often been determined for the "fixed-parameter" (fixed λ) case. *It would be highly desirable to re-use these for path-following.*

Re-using a left preconditioner $M \in \mathbb{R}^{n\times n}$ is straightforward: Just solve

$$M^{-1}F'(\bar{x})\,\bar{s} = -M^{-1}F(\bar{x})$$

as before.

With right preconditioning, there are at least two ways:

1. Approximately solve $F'(\bar{x}_k)QM^{-1}z_k = -F(\bar{x}_k)$, then set $\bar{s}_k = QM^{-1}z_k$.
2. Writing $\bar{M} = \begin{pmatrix} M & 0 \\ 0 & 1 \end{pmatrix}$, approximately solve $F'(\bar{x}_k)\bar{M}^{-1}Qz_k = -F(\bar{x}_k)$ and set $\bar{s}_k = \bar{M}^{-1}Qz_k$.

In limited experimentation, both seem equally effective.

Slide 248

Numerical experiments [105].

Recall the two example problems ($\mathcal{D} = [0, 1 \times [0, 1])$ . . .

Bratu problem: $\Delta u + \lambda e^u = 0$ in $\mathcal{D} = [0,1] \times [0,1]$, $\quad u = 0$ on $\partial\mathcal{D}$.

Chan [22] problem:

$$\Delta u + \lambda \left(1 + \frac{u + u^2/2}{1 + u^2/100}\right) = 0 \text{ in } \mathcal{D} = [0,1] \times [0,1], \quad u = 0 \text{ on } \partial\mathcal{D}.$$

We applied a simple path following method to these problems:

- Forward Euler predictor, augmented Jacobian corrector iterations.
- Approximate unit tangents were normalized differences of current and previous points on Γ.
- GMRES(40) and BiCGSTAB, preconditioned on the left with a *fast Poisson solver.*

Slide 249

Particular goal: Assess the effectiveness of the preconditioner, especially its *mesh independence.*

The tables show geometric means of successive linear residual norm ratios $\|r_{k+1}\|_2/\|r_k\|_2$ over all Krylov iterations at all corrector steps.

Grid Size	16×16	32×32	64×64	128×128
GMRES(40)	.0291	.0294	.0282	.0285
BiCGSTAB	.0681	.0961	.1091	.1278

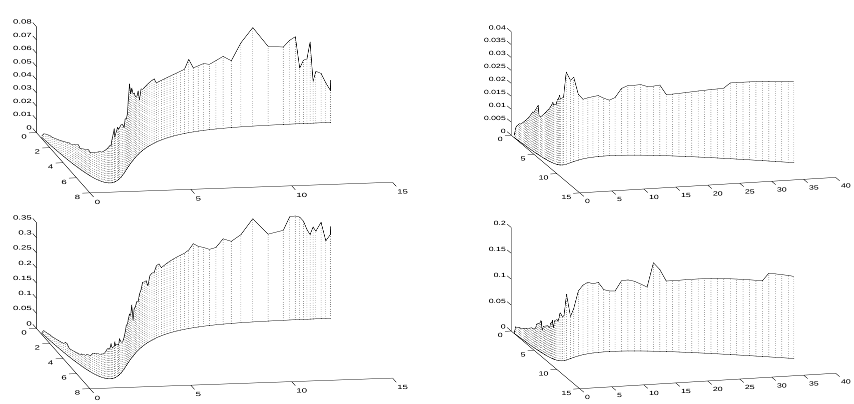
Results for the Bratu problem.

Grid Size	16×16	32×32	64×64	128×128
GMRES(40)	.0207	.0197	.0196	.0205
BiCGSTAB	.0575	.0655	.0789	.0935

Results for the Chan problem.

Slide 250

To show that convergence was not adversely affected near fold points, we plotted geometric means of residual norm ratios $\|r_{k+1}\|_2/\|r_k\|_2$ at each continuation step along the curves, using a 64×64 grid.



Bratu problem (left) and Chan problem (right); GMRES (top) and BiCGSTAB (bottom).

References

[1] E. L. Allgower and K. Böhmer, *Application of the mesh-independence principle to mesh refinement strategies*, SIAM J. Numer. Anal., 24 (1987), pp. 1335–1351.

[2] E. L. Allgower, K. Böhmer, F. A. Potra, and W. C. Rheinboldt, *A mesh-independence principle for operator equations and their discretizations*, SIAM J. Numer. Anal., 23 (1986), pp. 160–169.

[3] E. L. Allgower and K. Georg, *Continuation and path following*, Acta Numerica, (1993), pp. 1–64.

[4] L. Armijo, *Minimization of functions having Lipschitz-continuous first derivatives*, Pacific J. Math., 16 (1966), pp. 1–3.

[5] W. E. Arnoldi, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.

[6] K. E. Atkinson, *An Introduction to Numerical Analysis*, John Wiley and Sons, New York, 1978. 2nd Edition.

[7] Z. Bai, D. Hu, and L. Reichel, *A Newton basis* GMRES *implementation*, Tech. Rep. 91-03, Department of Mathematics, University of Kentucky, April 1991.

[8] R. E. Bank and D. J. Rose, *Global approximate Newton methods*, Numer. Math., 37 (1981), pp. 279–295.

[9] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1993.

[10] R. P. Brent, *An algorithm with guaranteed convergence for finding a zero of a function*, The Computer Journal, 14 (1971), pp. 422–425.

[11] ———, *Algorithms for Minimization without Derivatives*, Prentice Hall Series in Automatic Computation, Englewood Cliffs, NJ, 1973.

[12] P. N. Brown, *A local convergence theory for combined inexact-newton/finite difference projection methods*, SIAM J. Numer. Anal., 24 (1987), pp. 407–434.

[13] ———, *A theoretical comparison of the Arnoldi and* GMRES *algorithms*, SIAM J. Sci. Comput., 20 (1992), pp. 58–78.

[14] P. N. Brown, A. C. Hindmarsh, and H. F. Walker, *Experiments with quasi-newton methods in solving stiff ode systems*, SIAM J. Sci. Stat. Comput., 6 (1985), pp. 297–313.

[15] P. N. Brown and Y. Saad, *Hybrid Krylov methods for nonlinear systems of equations*, SIAM J. Sci. Stat. Comput., 11 (1990), pp. 450–481.

[16] P. N. Brown and H. F. Walker, GMRES *on (nearly) singular systems*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 37–51.

[17] C. G. Broyden, *A class of methods for solving nonlinear simultaneous equations*, Math. Comp., 19 (1965), pp. 577–593.

[18] ———, *A new double-rank minimization algorithm*, AMS Notices, 16 (1969), p. 670.

[19] ———, *The convergence of a class of double-rank minimization algorithms, Parts I and II*, J. Inst. Math. Appl., 6 (1971), pp. 76–90, 222–236.

[20] ———, *The convergence of an algorithm for solving sparse nonlinear systems*, Math. Comp., 25 (1971), pp. 285–294.

[21] C. G. Broyden, J. E. Dennis, Jr., and J. J. Moré, *On the local and superlinear convergence of quasi-newton methods*, J. Inst. Math. Appl., 12 (1973), pp. 223–246.

[22] T. F. Chan, *Newton-like pseudo-arclength methods for computing simple turning points*, SIAM J. Sci. Stat. Comput., 5 (1984), pp. 135–148.

[23] E. J. Craig, *The n-step iteration process*, J. Math. Phys., 34 (1955), pp. 64–73.

[24] J. K. Cullum and A. Greenbaum, *Peaks, plateaus, numerical instabilities in Galerkin and minimal residual pairs of methods for solving* $Ax = b$. preprint, 1994.

[25] ———, *Relations between galerkin and norm-minimizing iterative methods for solving linear systems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 223–247.

[26] A. Curtis, M. J. D. Powell, and J. K. Reid, *On the estimation of sparse jacobian matrices*, J. I. M. A., 13 (1974), pp. 117–120.

[27] W. C. Davidon, *Variable metric methods for minimization*, Tech. Rep. ANL-5990 Rev., Argonne Nat. Labs., 1959.

[28] R. S. Dembo, S. C. Eisenstat, and T. Steihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19 (1982), pp. 400–408.

[29] R. S. Dembo and T. Steihaug, *Truncated Newton algorithms for large-scale optimization*, Math. Prog., 26 (1983), pp. 190–212.

[30] J. E. Dennis, Jr., D. M. Gay, and R. E. Welsch, *An adaptive nonlinear least-squares algorithm*, ACM Trans. Math. Software, (1977).

[31] J. E. Dennis, Jr. and H. H. W. Mei, *Two new unconstrained optimization algorithms which use function and gradient values*, J. Optimization Theory Appl., 28 (1979), pp. 453–482.

[32] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Series in Automatic Computation, Prentice-Hall, Englewood Cliffs, NJ, 1983.

[33] J. E. Dennis, Jr. and H. F. Walker, *Convergence theorems for least change secant update methods*, SIAM J. Numer. Anal., 18 (1981), pp. 949–987.

[34] ———, *Least-change sparse secant update methods with inaccurate secant conditions*, SIAM J. Numer. Anal., 22 (1985), pp. 760–778.

[35] H. A. V. der Vorst and C. Vuik, GMRESR: *a family of nested* GMRES *methods*, Tech. Rep. 91-80, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1991.

[36] P. E.Gill, G. H.Golub, W. Murray, and M. A. Saunders, *Methods for modifying matrix factorizations*, Math. Comp., 28 (1974), pp. 505–535.

[37] S. C. Eisenstat and H. F. Walker, *Globally convergent inexact Newton methods*, SIAM J. Optimization, 4 (1994), pp. 393–422.

[38] ———, *Choosing the forcing terms in an inexact Newton method*, SIAM J. Sci. Comput., 17 (1996), pp. 16–32.

[39] M. El Hallabi, *A Global Convergence Theory for Arbitrary Norm Trust Region Methods for Nonlinear Equations*, PhD thesis, Department of Mathematical Sciences, Rice University, 1987.

[40] M. El Hallabi and R. A. Tapia, *A global convergence theory for arbitrary norm trust-region methods for nonlinear equations*, Tech. Rep. TR87-25, Department of Mathematical Sciences, Rice University, November 1987. Revised July, 1989.

[41] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.

[42] R. Fletcher, *A new approach to variable metric algorithms*, Comput. J., 13 (1970), pp. 317–322.

[43] ———, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis Dundee 1975, G. A. Watson, ed., Lecture Notes in Mathematics 506, Berlin, 1976, Springer, pp. 73–89.

[44] R. Fletcher and M. J. D. Powell, *A rapidly convergent descent method for minimization*, Comput. J., 6 (1963), pp. 163–168.

[45] G. E. Forsythe, M. A. Malcolm, and C. B. Moler, *Computer Methods for Mathematical Computations*, Series in Automatic Computations, Prentice-Hall, Englewood Cliffs, NJ, 1977.

[46] R. W. Freund, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM J. Sci. Comput., 14 (1993), pp. 470–482.

[47] R. W. Freund, G. H. Golub, and N. M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica 1992, 1 (1992), pp. 57–100. Cambridge University Press.

[48] R. W. Freund and N. M. Nachtigal, QMR: *a quasi-minimal residual method for non-Hermitian linear systems*, Numerische Mathematik, 60 (1991), pp. 315–339.

[49] P. E. Gill and W. Murray, *Quasi-newton methods for unconstrained optimization*, J. I. M. A., 9 (1972), pp. 91–108.

[50] R. Glowinski, H. B. Keller, and L. Rheinhart, *Continuation-conjugate gradient methods for the least-squares solution of nonlinear boundary value problems*, SIAM J. Sci. Stat. Comput., 6 (1985), pp. 793–832.

[51] D. Goldfarb, *A family of variable-metric methods derived by variational means*, Math. Comp., 24 (1970), pp. 23–26.

[52] A. A. Goldstein, *Constructive Real Analysis*, Harper and Row, New York, 1967.

[53] G. H. Golub and C. V. Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 1983.

[54] J. Greenstadt, *Variations on variable-metric methods*, Math. Comp., 24 (1970), pp. 1–18.

[55] A. GRIEWANK, *Rates of convergence for secant methods on nonlinear problems in hilbert space*, in Proceedings of the Fourth IIMAS Workshop on Numerical Analysis, J. P. Hennart, ed., Lecture Notes in Mathematics, New York, 1986, Springer, pp. 138–157. held at Guanajuato, México.

[56] ———, *The solution of boundary value problems by broyden based secant methods*, in CTAC-85, Proc. of the Computational Techniques and Applications Conference, J. Noye and R. May, eds., Amsterdam, 1986, North Holland. held at University of Melbourne, August, 1985.

[57] ———, *The local convergence of broyden-like methods on lipschitzian problems in hilbert spaces*, SIAM J. Numer. Anal., 24 (1987), pp. 684–705.

[58] ———, *On the iterative solution of differential and integral equations using secant updating techniques*, in The State of the Art in Numerical Analysis, A. Iserles and M. J. D. Powell, eds., Oxford, 1987, Clarendon Press, pp. 299–324.

[59] W. A. GRUVER AND E. SACHS, *Algorithmic Methods in Optimal Control*, Pitman, London, 1980.

[60] M. H. GUTKNECHT, *Lanczos-type solvers for nonsymmetric linear systems of equations*, Acta Numerica 1997, 6 (1997), pp. 271–397. Cambridge University Press.

[61] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards, 49 (1952), pp. 409–435.

[62] D. M. HWANG AND C. T. KELLEY, *Convergence of broyden's method in banach spaces*, SIAM J. Optimization, 2 (1992), pp. 505–532.

[63] E. ISAACSON AND H. B. KELLER, *Analysis of Numerical Methods*, John Wiley and Sons, New York, 1966.

[64] C. T. KELLEY, *Iterative Methods for Linear and Nonlinear Equations*, Frontiers in Applied Mathematics, SIAM, Philadelphia, 1995.

[65] C. T. KELLEY AND E. W. SACHS, *Broyden's method for approximate solution of nonlinear integral equations*, J. Int. Eqns., 9 (1985), pp. 25–44.

[66] ———, *A quasi-newton method for elliptic boundary value problems*, SIAM J. Numer. Anal., 24 (1987), pp. 516–531.

[67] ———, *Quasi-newton methods and unconstrained optimal control problems*, SIAM J. on Control and Optimization, 25 (1987), pp. 1503–1517.

[68] ———, *A new proof of superlinear convergence for broyden's method in hilbert space*, SIAM J. Optimization, 1 (1991), pp. 146–150.

[69] ———, *Pointwise broyden methods*, SIAM J. Optimization, 3 (1993), pp. 423–441.

[70] ———, *A pointwise quasi-newton method for unconstrained optimal control problems*, Numer. Math., 55 (89), pp. 159–176.

[71] C. LANCZOS, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. National Bureau of Standards, 45 (1950), pp. 255–282.

[72] ———, *Solution of systems of linear equations by minimized iterations*, J. Res. National Bureau of Standards, 49 (1952), pp. 33–53.

[73] D. LUENBERGER, *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, New York, 1973.

[74] E. S. MARWIL, *Exploiting sparsity in Newton-like methods*, PhD thesis, Department of Computer Science, Cornell University, 1978.

[75] J. J. MORÉ AND D. C. SORENSEN, *Computing a trust region step*, SIAM J. Sci. Stat. Comput., 4 (1983), pp. 553–572.

[76] J. J. MORÉ AND D. J. THUENTE, *Line search algorithms with guaranteed sufficient decrease*, ACM Trans. Math. Software, 20 (1994), pp. 286–307.

[77] N. M. NACHTIGAL, *A look-ahead variant of the Lanczos algorithm and its application to the quasi-minimal residual method for nonHermitian linear systems*, PhD thesis, Massachusetts Institute of Technology, 1991.

[78] N. M. NACHTIGAL, S. C. REDDY, AND L. N. TREFETHEN, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 778–795.

[79] J. NOCEDAL, *Theory of algorithms for unconstrained optimization*, Acta Numerica, (1992), pp. 199–242.

[80] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer, New York, 1999.

[81] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[82] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[83] J. D. Pearson, *Variable metric method of minimization*, Comput. J., 12 (1969), pp. 171–178.

[84] M. Pernice and H. F. Walker, NITSOL*: a Newton iterative solver for nonlinear systems*, SIAM J. Sci. Comput., 19 (1998), pp. 302–318.

[85] M. J. D. Powell, *A new algorithm for unconstrained optimization*, in Nonlinear Programming, J. B. Rosen, O. L. Mangasarian, and K. Ritter, eds., New York, 1970, Academic Press, pp. 31–65.

[86] ———, *General algorithms for discrete nonlinear approximation calculations*, in Approximation Theory IV, C. K. Chui, L. L. Schumaker, and J. D. Ward, eds., New York, 1983, Academic Press, pp. 187–218.

[87] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105–126.

[88] ———, *A flexible inner-outer preconditioned* GMRES *algorithm*, Tech. Rep. UMSI 91/279, University of Minnesota Supercomputer Institute, 1991.

[89] ———, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996. This book is currently out of print; an updated version can be downloaded at no cost from http://www-users.cs.umn.edu/~ saad/books.html.

[90] Y. Saad and M. H. Schultz, GMRES*: a generalized minimal residual method for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[91] E. W. Sachs, *Broyden's method in hilbert space*, Math. Prog., 35 (1986), pp. 71–82.

[92] R. Schreiber and H. B. Keller, *Spurious solutions in driven cavity calculations*, J. Comput. Phys., 49 (1983), pp. 165–172.

[93] L. K. Schubert, *Modification of a quasi-newton method for nonlinear equations with a sparse jacobian*, Math. Comp., 24 (1970), pp. 27–30.

[94] J. N. Shadid, R. S. Tuminaro, and H. F. Walker, *An inexact Newton method for fully-coupled solution of the Navier–Stokes equations with heat and mass transport*, J. Comput. Phys., 137 (1997), pp. 155–185.

[95] D. F. Shanno, *Conditioning of quasi-newton methods for function minimization*, Math. Comp., 24 (1970), pp. 647–656.

[96] J. Sherman and W. J. Morrison, *Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix*, Ann. Math. Stat., 20 (1949), p. 621.

[97] G. A. Shultz, R. B. Schnabel, and R. H. Byrd, *A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties*, SIAM J. Numer. Anal., 22 (1985), pp. 47–67.

[98] P. Sonneveld, CGS*, a fast Lanczos–type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 36–52.

[99] P. L. Toint, *On sparse and symmetric matrix updating subject to a linear equation*, Math.Comp., 31 (1977), pp. 954–961.

[100] K. Turner and H. F. Walker, *Efficient high accuracy solutions with* GMRES(m), SIAM J. Sci. Stat. Comput., 13 (1992), pp. 815–825.

[101] H. A. van der Vorst, Bi-CGSTAB*: a fast and smoothly converging variant of* bi-cg *for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 631–644.

[102] H. F. Walker, *Implementation of the* GMRES *method using Householder transformations*, SIAM J. Sci. Stat. Comput., 9 (1988), pp. 152–163.

[103] ———, *Implementations of the* GMRES *method*, Computer Physics Communication, 53 (1989), pp. 311–320.

[104] ———, *Residual smoothing and peak/plateau behavior in Krylov subspace methods*, Applied Numerical Mathematics, 19 (1995), pp. 279–286. Special Issue on Iterative Methods for Linear Equation.

[105] ———, *An adaptation of krylov subspace methods to path following problems*, SIAM J. Sci. Comput., 21 (1999), pp. 1191–1198.

[106] H. F. Walker and L. T. Watson, *Least-change secant update methods for underdetermined systems*, SIAM J. Numer. Anal., 27 (1990), pp. 1227–1262.

[107] H. F. Walker and L. Zhou, *A simpler* GMRES, J. Numer. Lin. Alg. Appl., 1 (1994), pp. 571–581.

[108] L. T. Watson, M. Sosonkina, R. C. Melville, A. P. Morgan, and H. F. Walker, *Hompack90: A suite of fortran90 codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software, 23 (1997), pp. 514–549.

[109] P. Wolfe, *Convergence conditions for ascent methods*, SIAM Rev., 11 (1969), pp. 226–235.

[110] ———, *Convergence conditions for ascent methods.II: Some corrections*, SIAM Rev., 13 (1971), pp. 185–188.

[111] M. A. Woodbury, *Inverting modified matrices*, Tech. Rep. Memorandum Report 42, Statistical Resesarch Group, Princeton University, 1950.

[112] L. Zhou and H. F. Walker, *Residual smoothing techniques for iterative methods*, SIAM J. Sci. Comput., 15 (1994), pp. 297–312.