

AMSC 600/CMSC 760
Fall 2007
Solution of Sparse Linear Systems
Part 1: Stationary Iterative Methods
Dianne P. O'Leary
©2006, 2007

Solving Sparse Linear Systems: Slow Iterative Methods

The plan:

- Iterative methods:
 - Basic (slow) iterations: Jacobi, Gauss-Seidel, SOR.

Basic iterations

The idea: Given an initial guess $\mathbf{x}^{(0)}$ for the solution to $\mathbf{Ax}^* = \mathbf{b}$, construct a sequence of guesses $\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots\}$ converging to \mathbf{x}^* .

The amount of work to construct each new guess from the previous one should be a small multiple of the number of nonzeros in \mathbf{A} .

Stationary Iterative Methods

These methods grew up in the engineering and mathematical literature. They were **very popular** in the 1960s and are still sometimes used.

Today, they are **almost never** the best algorithms to use (because they take too many iterations), but they are useful **preconditioners** for Krylov subspace methods.

We will define three of them:

- Jacobi (Simultaneous displacement)
- Gauss-Seidel (Successive displacement)
- SOR

Theme: All of these methods split \mathbf{A} as $\mathbf{M} - \mathbf{N}$ for some nonsingular matrix \mathbf{M} . Other splittings of this form are also useful.

The Jacobi iteration

Idea: The i th component of the residual vector \mathbf{r} is defined by

$$r_i = b_i - a_{i1}x_1 - a_{i2}x_2 - \dots - a_{i,i-1}x_{i-1} - a_{ii}x_i - a_{i,i+1}x_{i+1} - \dots - a_{in}x_n.$$

Let's modify x_i to make $r_i = 0$.

Given $\mathbf{x}^{(k)}$, construct $\mathbf{x}^{(k+1)}$ by

$$x_i^{(k+1)} = (b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)}) / a_{ii}, \quad i = 1, \dots, n.$$

Observations:

- We must require \mathbf{A} to have nonzeros on its main diagonal.
- The algorithm is easy to program! We only need to store two \mathbf{x} vectors, $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k+1)}$.
- The iteration may or may not converge, depending on the properties of \mathbf{A} .
- We should only touch the nonzeros in \mathbf{A} – otherwise the work per iteration would be $O(n^2)$ instead of $O(nz)$.
- If we partition \mathbf{A} as $\mathbf{L} + \mathbf{D} + \mathbf{U}$, where \mathbf{D} contains the diagonal entries, \mathbf{U} contains the entries above the diagonal, and \mathbf{L} contains the entries below the diagonal, then we can express the iteration as

$$\mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)}$$

and this is useful for analyzing convergence. ($\mathbf{M} = \mathbf{D}$, $\mathbf{N} = -(\mathbf{L} + \mathbf{U})$)

The Gauss-Seidel iteration

Idea: If we really believe that we have improved the i th component of the solution by our Jacobi iteration, then it makes sense to use its latest value in the iteration:

Given $\mathbf{x}^{(k)}$, construct $\mathbf{x}^{(k+1)}$ by

$$x_i^{(k+1)} = (b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)})/a_{ii}, \quad i = 1, \dots, n.$$

Observations:

- We still require \mathbf{A} to have nonzeros on its main diagonal.
- The algorithm is easier to program, since we only need to keep one \mathbf{x} vector around!
- The iteration may or may not converge, depending on the properties of \mathbf{A} .
- We should only touch the nonzeros in \mathbf{A} – otherwise the work per iteration would be $O(n^2)$ instead of $O(nz)$.
- If we partition \mathbf{A} as $\mathbf{L} + \mathbf{D} + \mathbf{U}$, where \mathbf{D} contains the diagonal entries, \mathbf{U} contains the entries above the diagonal, and \mathbf{L} contains the entries below the diagonal, then we can express the iteration as

$$(\mathbf{D} + \mathbf{L})\mathbf{x}^{(k+1)} = \mathbf{b} - \mathbf{U}\mathbf{x}^{(k)}$$

and this is useful for analyzing convergence. ($\mathbf{M} = \mathbf{D} + \mathbf{L}$, $\mathbf{N} = -\mathbf{U}$)

The SOR (Successive Over-Relaxation) iteration

Idea: People who used these iterations on finite difference matrices discovered that Gauss-Seidel (GS) converged faster than Jacobi (J), and they could improve its convergence rate by [going a little further in the GS direction](#):

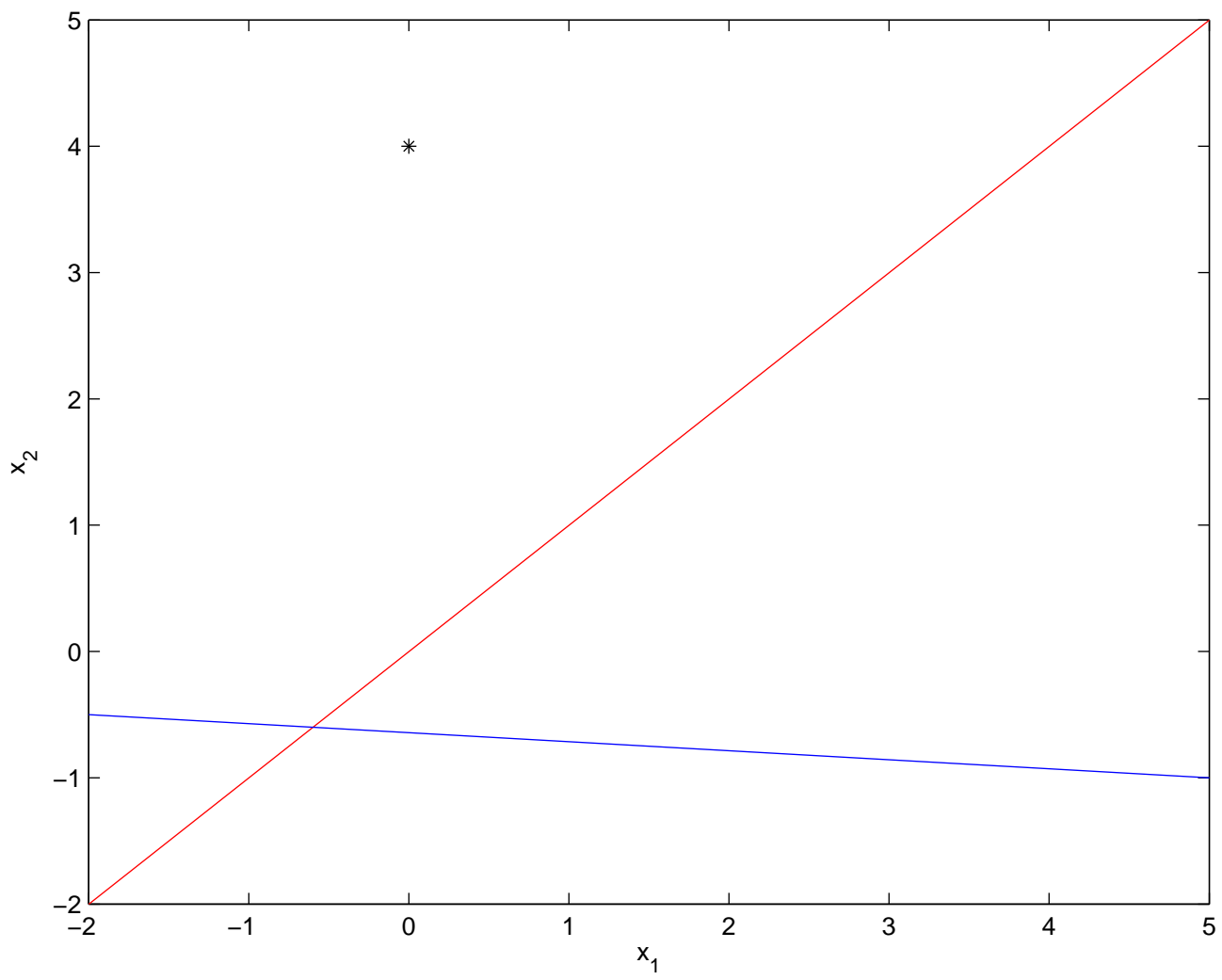
Given $\mathbf{x}^{(k)}$, construct $\mathbf{x}^{(k+1)}$ by

$$\mathbf{x}^{(k+1)} = (1 - \omega)\mathbf{x}^{(k)} + \omega\mathbf{x}^{(k+1)}_{GS}$$

where ω is a number between 1 and 2.

Unquiz: Suppose $n = 2$ and our linear system can be graphed as in the figure. Draw the first 3 Jacobi iterates and the first 3 Gauss-Seidel iterates using the point marked with a star as $\mathbf{x}^{(0)}$. Does either iteration depend on the ordering of the equations or unknowns? []

Convergence of Stationary iterative methods



- All of these iterations can be expressed as

$$\mathbf{x}^{(k+1)} = \mathbf{G}\mathbf{x}^{(k)} + \mathbf{c}$$

where $\mathbf{G} = \mathbf{M}^{-1}\mathbf{N}$ is a matrix that depends on \mathbf{A} and \mathbf{c} is a vector that depends on \mathbf{A} and \mathbf{b} .

- For all of these iterations, $\mathbf{x}^* = \mathbf{G}\mathbf{x}^* + \mathbf{c}$.
- Subtracting, we see that the error $\mathbf{e}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ satisfies

$$\mathbf{e}^{(k+1)} = \mathbf{G}\mathbf{e}^{(k)},$$

and it can be shown that the error converges to zero for any initial $\mathbf{x}^{(0)}$ if and only if all of the eigenvalues of \mathbf{G} lie inside the unit circle.

- Many conditions on \mathbf{A} have been found that guarantee convergence of these methods; see the SIM notes.

Unquiz: Let \mathbf{G} be an $n \times n$ matrix with a full set of linearly independent eigenvectors. Given a vector $\mathbf{x}^{(0)}$, consider the iteration

$$\mathbf{x}^{(k+1)} = \mathbf{G}\mathbf{x}^{(k)} + \mathbf{b},$$

for $k = 0, 1, \dots$

- (a) Suppose we have a vector \mathbf{x}_{true} satisfying

$$\mathbf{x}_{true} = \mathbf{G}\mathbf{x}_{true} + \mathbf{b}.$$

Such a vector is called a **fixed point** of the iteration. Show that if $\mathbf{e}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}_{true}$, then

$$\mathbf{e}^{(k+1)} = \mathbf{G}\mathbf{e}^{(k)}.$$

- (b) Show by induction that

$$\mathbf{e}^{(k)} = \mathbf{G}^k \mathbf{e}^{(0)}.$$

(c) Show that $\mathbf{e}^{(k)} \rightarrow \mathbf{0}$ as $k \rightarrow \infty$ for any initial vector $\mathbf{x}^{(0)}$ if and only if all eigenvalues of \mathbf{G} lie within the unit circle; i.e., if and only if $|\lambda_j| < 1$ for $j = 1, \dots, n$. Hint: To do this, consider expressing $\mathbf{e}^{(0)}$ as

$$\mathbf{e}^{(0)} = \sum_{j=1}^n \alpha_j \mathbf{u}_j$$

where \mathbf{u}_j are the eigenvectors of \mathbf{G} and the values α_j are appropriate coefficients. Now compute $\mathbf{G}^k \mathbf{e}^{(0)}$ and study its convergence. \square

This is enough about slow methods, except to note the following fact.

So far: Stationary iterative methods.

- $\mathbf{Ax} = \mathbf{b}$ is replaced by $\mathbf{x} = \mathbf{Gx} + \mathbf{c}$.
- $\mathbf{x}^{(k+1)} = \mathbf{Gx}^{(k)} + \mathbf{c}$
- If $\mathbf{x}^{(0)} = \mathbf{0}$, then

$$\begin{aligned}\mathbf{x}^{(1)} &= \mathbf{c} \\ \mathbf{x}^{(2)} &\in \text{span}\{\mathbf{c}, \mathbf{Gc}\} \\ \mathbf{x}^{(3)} &\in \text{span}\{\mathbf{c}, \mathbf{Gc}, \mathbf{G}^2\mathbf{c}\} \\ \mathbf{x}^{(k)} &\in \text{span}\{\mathbf{c}, \mathbf{Gc}, \mathbf{G}^2\mathbf{c}, \dots, \mathbf{G}^{k-1}\mathbf{c}\} \\ &\equiv \mathcal{K}_k(\mathbf{G}, \mathbf{c})\end{aligned}$$

and we call $\mathcal{K}_k(\mathbf{G}, \mathbf{c})$ a **Krylov subspace**.

- The work per iteration is $O(nz)$ plus a small multiple of n .
- Note that $\mathcal{K}_k(\mathbf{G}, \mathbf{c}) = \mathcal{K}_k(\hat{\mathbf{G}}, \mathbf{c})$ if $\hat{\mathbf{G}} = \mathbf{I} - \mathbf{G}$.

The idea behind Krylov subspace methods: Instead of making the GS choice (for example) from the Krylov subspace, let's try to pick the **best** vector without doing a lot of extra work.

What is “best”?

- **The variational approach:** Choose $\mathbf{x}^{(k)} \in \mathcal{K}_k(\mathbf{G}, \mathbf{c})$ to minimize

$$\|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{Z}}$$

where $\|\mathbf{y}\|_{\mathbf{Z}}^2 = \mathbf{y}^T \mathbf{Z} \mathbf{y}$ and \mathbf{Z} is a symmetric positive definite matrix.

- **The Galerkin approach:** Choose $\mathbf{x}^{(k)} \in \mathcal{K}_k(\mathbf{G}, \mathbf{c})$ to make the residual $\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{Ax}^{(k)}$ orthogonal to every vector in $\mathcal{K}_k(\mathbf{G}, \mathbf{c})$ for some choice of inner product.

We'll follow up on these ideas in the next set of notes.