Solution of Linear Equations

$$
\begin{array}{ccccccccc}
a_{00}x_0 & + & a_{01}x_1 & + & \dots & + & a_{0(n-1)}x_{(n-1)} & = & b_0 \\
a_{10}x_0 & + & a_{11}x_1 & + & \dots & + & a_{1(n-1)}x_{(n-1)} & = & b_1 \\
\vdots & & \vdots & & \vdots & & \vdots & & \vdots \\
a_{(n-1)0}x_0 & + & a_{(n-1)1}x_1 & + & \dots & + & a_{(n-1)(n-1)}x_{(n-1)} & = & b_{(n-1)}
\end{array}
$$

- $\mathbf{Ax} = \mathbf{b}$ in matrix form
  - $\mathbf{A}$ is an $n \times n$ matrix
  - $\mathbf{b}$ and $\mathbf{x}$ are vectors of length n
- Small systems can be solved by *Gaussian elimination*
  - expensive – $\Theta(n^3)$
  - hard to parallelise

Upper/lower triangular form

We can write $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$ where:

- $\mathbf{L} = \{l_{ij} \mid 0 \le i, j \le n-1\}$ is lower-triangular

$$l_{ij} = \begin{array}{ll} a_{ij} & j < i \\ 0 & j \ge i \end{array}$$

- $\mathbf{D} = \{d_{ij} \mid 0 \le i, j \le n-1\}$ is diagonal

$$d_{ij} = \begin{array}{ll} a_{ii} & \\ 0 & j \ne i \end{array}$$

- $\mathbf{U} = \{u_{ij} \mid 0 \le i, j \le n-1\}$ is upper-triangular

$$u_{ij} = \begin{array}{ll} a_{ij} & j > i \\ 0 & j \le i \end{array}$$

Assumption

- We assume $d_{ii} \ne 0$ for all i
  - if not, we can permute the variables of $\mathbf{x}$ or the sequence of equations
  - but there is no solution if this is not possible

Jacobi's Method

- Matrix equation can be written as:

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

- Jacobi iteration is simply defined by:

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

  where $k \ge 0$ and $x^{(0)}$ is an initial "guess"
- Or, in terms of elements:

$$x_i^{(k+1)} = \frac{1}{a_{ii}}(b_i - \sum_{j \ne i} a_{ij}x_j^{(k)}) \quad \text{for } i, j = 0, 1, \ldots, n-1$$

Convergence of Jacobi

- Sufficient condition for convergence is:

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \quad \text{for } 0 \leq i, j \leq n-1$$

  ($\mathbf{A}$ is *strictly diagonally dominant*)

- A common check in practical implementations is:

$$\frac{||x^{(k+1)} - x^{(k)}||_\infty}{||x^{(k+1)}||_\infty} < \varepsilon$$

  where $||x||_\infty = \max_i |x_i|$ (the *infinity-norm*) and ε is a pre-defined threshold (e.g. 10^{-8} or less)

Parallel Jacobi

- Parallel implementation is straightforward:
  - $\mathbf{D}^{-1}\mathbf{b}$ is a constant vector
  - $(\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})\mathbf{x}^{(k)})$ is evaluated by parallel matrix-vector multiplication
  - $\mathbf{x}^{(k)}$ is distributed according to the partition of $\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})$, e.g. :
    - on row-processor for striping
    - on "diagonal" processor for checkerboarding
    - or according to some other partitioning scheme (more on this later!)

Gauss-Seidel Method

- Improve convergence of Jacobi by using *up-to-date information* as it is computed
  - if components of $\mathbf{x}^{(k)}$ are calculated in increasing order of subscript, the dot-product $\mathbf{Lx}$ can use components of $\mathbf{x}$ from the *current* iteration
  - but the dot-product $\mathbf{Ux}$ can't
- Gauss-Seidel iteration is defined by:

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

- Or in terms of the elements of the matrix and vectors:

Convergence of Gauss-Seidel

- Same sufficient condition for convergence (strict diagonal dominance of $\mathbf{A}$) as Jacobi
- Also the same practical test for numerical convergence
- If $\mathbf{A}$ is symmetric, then $\mathbf{A}$ positive-definite is a *necessary and sufficient condition*

Parallel Gauss-Seidel

- Parallelisation is harder for Gauss-Seidel because of the sequentiality of the update process...
- ...although for some sparse matrices you may be able to reorder the equations to allow computation to be done in parallel