## 1.4 The LU Decomposition

It is often advantageous to think of Gaussian elimination as constructing a lower tridiagonal matrix $L$ and an upper triangular matrix $U$, so that $LU = A$. To illustrate this method consider the following *tridiagonal* linear system

$$\begin{pmatrix} b_0 & c_0 & 0 & \cdots & \cdots & 0 \\ a_1 & b_1 & c_1 & & & \vdots \\ 0 & a_2 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & \cdots & \cdots & 0 & a_n & b_n \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ \vdots \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} \qquad (2)$$

We can write this in the form

$$\mathbf{M}.\mathbf{y} = \mathbf{p}$$

and then consider the matrix factorisation

$$\begin{aligned} \mathbf{M} &= \mathbf{LU} \\ &= \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ l_1 & 1 & 0 & & & \vdots \\ 0 & l_2 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & l_{n-1} & 1 & 0 \\ 0 & \cdots & \cdots & 0 & l_n & 1 \end{pmatrix} \begin{pmatrix} d_0 & u_0 & 0 & \cdots & \cdots & 0 \\ 0 & d_1 & u_1 & & & \vdots \\ 0 & 0 & d_2 & u_2 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & 0 & d_{n-1} & u_{n-1} \\ 0 & \cdots & \cdots & 0 & 0 & d_n \end{pmatrix} \\ &= \begin{pmatrix} d_0 & u_0 & 0 & \cdots & \cdots & 0 \\ l_1 d_0 & l_1 u_0 + d_1 & u_1 & & & \vdots \\ 0 & l_2 d_1 & l_2 u_1 + d_2 & u_2 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & l_{n-1} d_{n-2} & l_{n-1} u_{n-2} + d_{n-1} & u_{n-1} \\ 0 & \cdots & \cdots & 0 & l_n d_{n-1} & l_n u_{n-1} + d_n \end{pmatrix} \end{aligned}$$

where $L$ is the lower triangular matrix above and $U$ is the upper triangular matrix above.

Equating the elements of the original tridiagonal matrix $M$ with the elements of the product matrix $LU$, we find that

$$\begin{aligned} u_i &= c_i, \quad i = 0, 1, 2, \ldots\ldots n-1 \\ d_0 &= b_0 \\ l_i &= \frac{a_i}{d_{i-1}}, \quad d_i = b_i - l_i u_{i-1}, \quad i = 1, 2, \ldots\ldots, n \end{aligned} \tag{3}$$

We then solve the original system given by (2) by solving two smaller problems. The tridiagonal system (2) is written in the form

$$\mathbf{M.y} = (\mathbf{LU})\,\mathbf{.y} = \mathbf{L(U.y)} = \mathbf{p}$$

We introduce an intermediate vector $\mathbf{z} = \mathbf{Uy}$ so that (2) becomes

$$\mathbf{Lz} = \mathbf{p}, \quad \mathbf{Uy} = \mathbf{z}.$$

First we solve the problem

$$\mathbf{Lz} = \mathbf{p}$$

for the intermediate vector $\mathbf{z}$, i.e. we solve the system

$$\begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ l_1 & 1 & & & & \vdots \\ 0 & l_2 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & l_{n-1} & 1 & 0 \\ 0 & \cdots & \cdots & 0 & l_n & 1 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ \vdots \\ z_{n-1} \\ z_n \end{pmatrix} = \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} \tag{4}$$

Trivially, the $z_i$ can be found by forward substitution so

$$z_0 = p_0, \quad z_i = p_i - l_i z_{i-1}, \quad i = 1, 2, \ldots\ldots, n$$

This determines the vector $\mathbf{z}$.

Having obtained $\mathbf{z}$ we find $\mathbf{y}$ by solving $\mathbf{Uy} = \mathbf{z}$;

$$\begin{pmatrix} d_0 & u_0 & 0 & \cdots & \cdots & 0 \\ 0 & d_1 & u_1 & & & \vdots \\ 0 & 0 & d_2 & u_2 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & 0 & d_{n-1} & u_{n-1} \\ 0 & \cdots & \cdots & 0 & 0 & d_n \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ \vdots \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ \vdots \\ z_{n-1} \\ z_n \end{pmatrix} \tag{5}$$

The solution of (5) is trivially obtained by backward substitution

$$y_n = \frac{z_n}{d_n}, \quad y_i = \frac{z_i - c_i y_{i+1}}{d_i}, \quad i = n-1, n-2, \ldots\ldots, 2, 1$$

This gives us the solution, $\mathbf{y}$, of our original problem (2), $\mathbf{My} = \mathbf{p}$.

This method can be also extended to decompose non-sparse matrices, i.e. $A = LU$ , thus opening up a wider class of associated methods.

$$= \begin{pmatrix} l_{11} & 0 & \cdots & \cdots & \cdots & 0 \\ l_{21} & l_{22} & 0 & & & \vdots \\ \vdots & l_2 & & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & l_{n-1} & l_{n-1,n-1} & 0 \\ l_{n1} & l_{n2} & \cdots & \cdots & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & \cdots & \cdots & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & \cdots & \cdots & u_{2n} \\ 0 & 0 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & 0 & u_{n-1,n-1} & u_{n-1,n} \\ 0 & \cdots & \cdots & 0 & 0 & u_{nn} \end{pmatrix}$$

$=$

If $l_{ii} = 1 \ \forall \ 1 \leq i \leq n$ as mentioned in the earlier discussion, this ensures a unique solution and is called *Doolittle's method*. *Crout's method* requires $u_{ii} = 1 \ \forall \ 1 \leq i \leq n$.

The case where $l_{ii} = u_{ii} \ \forall \ 1 \leq i \leq n$ is known as *Choleski's method*. This requires the matrix $\mathbf{A}$ to be symmetric and *positive definite*. Hence we can find an upper triangular matrix $\mathbf{U}$ such that $\mathbf{A} = \mathbf{U}^{\mathrm{T}}\mathbf{U}$

A symmetric matrix $A$ is positive definite if for all $\underline{x} \neq \underline{0}$, $\underline{x}^{\mathrm{T}} A \underline{x} > 0$. As an example consider

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 3 \end{pmatrix}$$

Therefore if $\underline{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ then $\underline{x}^{\mathrm{T}} A \underline{x}$ becomes

$$(x,\ y,\ z) \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = (x,\ y,\ z) \begin{pmatrix} x+y \\ x+2y+z \\ y+3z \end{pmatrix}$$

$= x^2 + 2y^2 + 3z^2 + 2xy + 2yz$

$= (x+y)^2 + (y+z)^2 + 2z^2$ (sum of squares)

$\Rightarrow$ for $(x,\ y,\ z) \neq \underline{0}$, the quadratic form above is always positive, hence $A$ is strictly positive definite.

Example:

Consider the matrix $\begin{pmatrix} 2 & -1 & 1 \\ 3 & 3 & 9 \\ 3 & 3 & 5 \end{pmatrix}$ which can be factorised into

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 1.5 & 1 & 0 \\ 1.5 & 1 & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 4.5 & 7.5 \\ 0 & 3 & -4 \end{pmatrix}$$

Ex $A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 2 \\ 0 & 2 & 4 \end{pmatrix}$ +ve definite if $\underline{x}^T A \underline{x} > 0$

$\underline{x} \neq 0$.

$\forall (x, y, z) \neq (0, 0, 0)$

$$(x, y, z) \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 2 \\ 0 & 2 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = (x, y, z) \begin{pmatrix} 2x + y \\ x + 3y + 2z \\ 2y + 4z \end{pmatrix}$$

$$= 2x^2 + 2xy + 3y^2 + 4yz + 4z^2 \quad \text{(quadratic form)}.$$

Can be written $x^2 + (x^2 + 2xy + y^2) + y^2 + (y^2 + 4z^2 + 4yz)$

now we have perfect squares.

$$= x^2 + (x+y)^2 + y^2 + (y+2z)^2$$

always +ve provided $\underline{x} \neq \underline{0}$.

## 1.5 Criteria for invertibility

A system of linear equations is uniquely solvable if and only if the matrix $\mathbf{A}$ is invertible. This in turn is true if any of the following is:

1. If and only if the determinant is nonzero;
2. If and only if all the eigenvalues are nonzero;
3. If (but not only if) it is **strictly diagonally dominant.**

In practise it takes far too long to work out the determinant. The second criterion is often useful though, and there are quite quick methods for working out the eigenvalues. The third method is explained on the next page.

(Note: there are many other criteria for invertibility.)

A matrix $\mathbf{A}$ with entries $A_{ij}$ is strictly diagonally dominant if

$$|A_{ii}| > \sum_{j \neq i} |A_{ij}| .$$

That is, the diagonal element in each row is bigger in modulus that the sum of the moduli of the off-diagonal elements in that row.

For example,

$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ is s.d.d. and so invertible;

$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$ is not s.d.d. but still invertible;

$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ is neither s.d.d. nor invertible.

$\begin{pmatrix} 7 & 2 & 0 \\ 3 & 5 & -1 \\ 0 & 5 & -6 \end{pmatrix}$

$|7| > |2| + |0|$ ✓
$|5| > |3| + |-1|$ ✓
$|-6| > |5| + |0|$ ✓

} Inverse exists.

## 1.6 Iterative Techniques

Seldom used for low dimensional problems - Gaussian elimination preferred.

For large systems with a high percentage of zero entries these are efficient in terms of

1. Computer Storage
2. Computational Time

Consider $(n \times n)$ linear system: $A\underline{x} = \underline{b}$

Initial approximation $\underline{x}^{(0)}$ to solution $\underline{x}$ which generates a sequence of vectors $\{\underline{x}^{(k)}\}_{k=0}^{\infty}$ that converges to $\underline{x}$.

Most techniques involve a process which converts $A\underline{x} = \underline{b}$ to

$$\underline{x} = T\underline{x} + \underline{c}$$

where $T$ is an $(n \times n)$ matrix and $\underline{c}$ is a vector.

The initial vector $\underline{x}^{(0)}$ is selected and a sequence of approximations generated by computing

$$\underline{x}^{(k)} = T\underline{x}^{(k-1)} + \underline{c}$$

for each $k = 1, 2, 3, \ldots$.

Now consider $A\underline{x} = \underline{b}$ with

$$A = \begin{bmatrix} 10 & -1 & 2 & 0 \\ -1 & 11 & -1 & 3 \\ 2 & -1 & 10 & -1 \\ 0 & 3 & -1 & 8 \end{bmatrix}, \quad \underline{b} = \begin{bmatrix} 6 \\ 25 \\ -11 \\ 15 \end{bmatrix}$$

$E_1$ : $10x_1 - x_2 + 2x_3 = 6$

$E_2$ : $-x_1 + 11x_2 - x_3 + 3x_4 = 25$

$E_3$ : $2x_1 - x_2 + 10x_3 - x_4 = -11$

$E_4$ : $3x_2 - x_3 + 8x_4 = 15$

which has an exact solution

$$\underline{x} = \begin{bmatrix} 1 \\ 2 \\ -1 \\ 1 \end{bmatrix}$$

To convert $A\underline{x} = \underline{b}$ to the form $\underline{x} = T\underline{x} + \underline{c}$ ; solve $E_i$ for each $x_i$ $(i = 1, 2, 3, 4)$ to give

$$\left.\begin{aligned} x_1^{(k)} &= \tfrac{1}{10}\left\{x_2^{(k-1)} - 2x_3^{(k-1)} + 6\right\} \\ x_2^{(k)} &= \tfrac{1}{11}\left\{x_1^{(k-1)} + x_3^{(k-1)} - 3x_4^{(k-1)} + 25\right\} \\ x_3^{(k)} &= \tfrac{1}{10}\left\{-2x_1^{(k-1)} + x_2^{(k-1)} + x_4^{(k-1)} - 11\right\} \\ x_4^{(k)} &= \tfrac{1}{8}\left\{-3x_2^{(k-1)} + x_3^{(k-1)} + 15\right\} \end{aligned}\right\} \qquad (6)$$

So (6) can be written as

$$\left.\begin{aligned} x_1^{(k)} &= \tfrac{1}{10}\left\{x_2^{(k-1)} - 2x_3^{(k-1)}\right\} + \tfrac{3}{5} \\ x_2^{(k)} &= \tfrac{1}{11}\left\{x_1^{(k-1)} + x_3^{(k-1)} - 3x_4^{(k-1)}\right\} + \tfrac{25}{11} \\ x_3^{(k)} &= \tfrac{1}{10}\left\{-2x_1^{(k-1)} + x_2^{(k-1)} + x_4^{(k-1)}\right\} - \tfrac{11}{10} \\ x_4^{(k)} &= \tfrac{1}{8}\left\{-3x_2^{(k-1)} + x_3^{(k-1)}\right\} + \tfrac{15}{8} \end{aligned}\right\}$$

$\Rightarrow$

$$T = \begin{bmatrix} 0 & \frac{1}{10} & -\frac{1}{5} & 0 \\ \frac{1}{11} & 0 & \frac{1}{11} & -\frac{3}{11} \\ -\frac{1}{5} & \frac{1}{10} & 0 & \frac{1}{10} \\ 0 & -\frac{3}{8} & \frac{1}{8} & 0 \end{bmatrix} \qquad \underline{c} = \begin{bmatrix} \frac{3}{5} \\ \frac{25}{11} \\ -\frac{11}{10} \\ \frac{15}{8} \end{bmatrix}$$

For initial approximation let $\underline{x}^{(0)} = \underline{0}^{\mathrm{T}}$ and generate $\underline{x}^{(1)}$ by substituting $(0,\ 0,\ 0,\ 0)^{\mathrm{T}}$ in (6) to get $(0.6,\ 2.2727,\ -1.1000,\ 1.8750) = \underline{x}^{(1)}$ and $\underline{x}^{(2)}$ can now be obtained.

In general obtain $\underline{x}^{(k)} = \left(x_1^{(k)},\ x_2^{(k)},\ x_3^{(k)},\ x_4^{(k)}\right)^{\mathrm{T}}$.

By the very nature of iterative techniques, these continue indefinitely, hence one requires a convergence criterion to terminate the computation once the imposed condition is satisfied. So we need to know when to stop iterating, i.e. does the sequence converge to the solution of the given system of equations? To answer this, we need a means of measuring the distance between $n-$ dimensional vectors.

## 1.7 Vector Norms

In two and three dimensions, the size of vectors called the modulus is generally obtained by Pythagoras. So if

$$\underline{v} = (a_1,\ a_2,\ a_3)$$

then the modulus of this vector $|\underline{v}| = \sqrt{(a_1^2+,\ a_2^2,\ a_3^2)}$ which gives us the distance from the origin.

Now consider $\underline{x} = (x_1,\ x_2, \ldots\ldots\ldots,\ x_n)^T$. Let $\mathbb{R}^n$ denote the set of all $n-$ dimensional vectors (so all vector components are real). To define distance in $\mathbb{R}^n$ we use the notion of a *norm*. A pair of double vertical lines $\|\cdot\|$ is used to denote a norm, i.e. size of an $n-$ dimensional vector.

More formally a *vector norm* on $\mathbb{R}^n$ is a function, $\|\cdot\|$, such that

$$\|\cdot\| : \mathbf{R}^n \longrightarrow \mathbf{R}$$

with the following properties:

**(i)** $\|\underline{x}\| \geq 0 \quad \forall \underline{x} \in \mathbb{R}^n$

**(ii)** $\|\underline{x}\| = 0 \quad$ iff $\underline{x} = \underline{0}$

**(iii)** $\|\alpha\underline{x}\| = |\alpha|\,\|\underline{x}\| \quad \forall \alpha \in \mathbb{R}$ and $\underline{x} \in \mathbb{R}^n$

**(iv)** $\|\underline{x} + \underline{y}\| \leq \|\underline{x}\| + \|\underline{y}\| \quad \forall \underline{x},\ \underline{y} \in \mathbb{R}^n$ (Triangle inequality)

If $\underline{x} = (x_1,\ x_2, \ldots\ldots\ldots,\ x_n)^T$, then the vector norm $\|\underline{x}\|_p$ for $p = 1, 2, \ldots$ is defined as

$$\|\underline{x}\|_p = \left\{ \sum_{i=1}^{n} |x_i|^p \right\}^{1/p}$$

Then there are a number of ways of defining a *norm*, depending on the value of $p$.

The special case $\|\underline{x}\|_\infty$ is given by

$$\|\underline{x}\|_\infty = \max_{1 \leq\ i\ \leq\ n} |x_i| \qquad l_\infty \text{ norm.}$$

The most commonly encountered vector norm (also known as the modulus of a vector) is $\|\underline{x}\|_2$ and defined by

$$\|\underline{x}\|_2 = \left\{ \sum_{i=1}^{n} x_i^2 \right\}^{1/2} \qquad l_2 \text{ norm}$$

For obvious reasons this is also called the *Euclidean norm*.

## Examples

1. $\underline{x} = (\sin k,\ \cos k,\ 2^k)^T$ ; $k$ is a positive integer

$\|\underline{x}\|_2 = \sqrt{\sin^2 k + \cos^2 k + 4^k} = \sqrt{1 + 4^k}$

$\|\underline{x}\|_\infty = \max |\sin k,\ \cos k,\ 2^k| = 2^k$

2. $\underline{x} = (2,\ 1,\ 3,\ -4)^T$

$\|\underline{x}\|_2 = \sqrt{30}$ and $\|\underline{x}\|_\infty = 4$

These and other types of vector norms are further summarized in the following table, using a particular vector $\underline{y} = (1,\ 2,\ 3)$

| Name | Symbol | value | numerical value |
|---|---|---|---|
| $l_1$ – norm | $\|\underline{x}\|_1$ | 6 | 6.000 |
| $l_2$ – norm | $\|\underline{x}\|_2$ | $\sqrt{14}$ | 3.742 |
| $l_3$ – norm | $\|\underline{x}\|_3$ | $6^{2/3}$ | 3.302 |
| $l_4$ – norm | $\|\underline{x}\|_4$ | $2^{1/4}\sqrt{7}$ | 3.146 |
| $l_\infty$ – norm | $\|\underline{x}\|_\infty$ | 3 | 3.00 |

If $\underline{x} = (x_1,\ x_2, \ldots\ldots,\ x_n)^T$ , $\underline{y} = (y_1,\ y_2, \ldots\ldots,\ y_n)^T \in \mathbb{R}^n$, then the definitions for $l_2$ and $l_\infty$ can be easily extended to consider distances between $\underline{x}$ and $\underline{y}$

$$\|\underline{x} - \underline{y}\|_2 = \left\{ \sum_{i=1}^{n} (x_i - y_i)^2 \right\}^{1/2}$$

$$\|\underline{x} - \underline{y}\|_\infty = \max_{1 \le i \le n} |x_i - y_i|$$

Having introduced the concept of a vector norm, we can now define the most appropriate form for our convergence criteria, by making use of the $l_\infty$ norm. One such relative condition makes use of the infinity norm $l_\infty$ as a test for convergence, where

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

$\varepsilon$ is the specified tolerance ($> 0$) and $\|\underline{x}\|_\infty = \max_{1 \le i \le n} |x_i|$.

In the numerical example, actual computations yield for $k = 9$ & $10$

| $k = 9$ | $k = 10$ | |
|---|---|---|
| 0.9997 | 1.0001 | based on $\epsilon = 10^{-3}$ |
| 2.0004 | 1.9998 | |
| −1.0004 | −0.9998 | |
| 1.0006 | 0.9998 | |

$$\frac{\left\|\underline{x}^{(10)} - \underline{x}^{(9)}\right\|_\infty}{\left\|\underline{x}^{(10)}\right\|_\infty} = \frac{\left\|(4.\ -6,\ 6,\ -8)\,10^{-4}\right\|_\infty}{1.9998} = 4 \times 10^{-4} < \epsilon$$

Also $\left\|\underline{x}^{(10)} - \underline{x}^{(9)}\right\|_\infty = 2 \times 10^{-4} < \epsilon$

The iterative technique used is called the *JACOBI* method.

It consists of solving the $i^{th}$ equation in $A\underline{x} = \underline{b}$ for $x_i$ (provided $a_{ii} \neq 0$) to obtain

$$x_i = \sum_{j=1}^{n} \left( \frac{-a_{i\,j} x_j}{a_{ii}} \right) + \frac{b_i}{a_{ii}} \qquad i = 1, \ldots\ldots, n \quad ; \quad j \neq i$$

and generating each $x_i^{(k)}$ from components of $\underline{x}^{(k-1)}$ $(k \geq 1)$ by

$$x_i^{(k)} = \frac{1}{a_{ii}} \left[ \sum_{j=1}^{n} \left( -a_{i\,j}\, x_j^{(k-1)} \right) + b_i \right] \qquad i = 1, \ldots\ldots, n \quad ; \quad j \neq i$$

We can now refine this method very simply by using the most up-to-date $x_i$. To compute $x_i^{(k)}$, components of $\underline{x}^{(k-1)}$ are used.

Since for $i > 1$, $x_1^{(k)}$, ....., $x_{i-1}^{(k)}$ have already been computed and are likely to be better approximations to the actual solutions $x_1$, ....., $x_{i-1}$ than $x_1^{(k-1)}$, ....., $x_{i-1}^{(k-1)}$, it is far better to calculate $x_i^{(k)}$ using the most recently calculated values, i.e.

$$x_i^{(k)} = \frac{1}{a_{ii}} \left[ -\sum_{j=1}^{i-1} \left( a_{i\,j}\, x_j^{(k)} \right) - \sum_{j=i+1}^{n} \left( a_{i\,j}\, x_j^{(k-1)} \right) + b_i \right],$$

$$(i = 1, \ldots\ldots, n)$$

This is called the *GAUSS-SEIDEL* method.

Re-doing the previous example using the G-S method iterative technique:

$$
\begin{aligned}
10x_1 - x_2 + 2x_3 &= 6 \\
-x_1 + 11x_2 - x_3 + 3x_4 &= 25 \\
2x_1 - x_2 + 10x_3 - x_4 &= -11 \\
3x_2 - x_3 + 8x_4 &= 15
\end{aligned}
$$

which is written as

$$
\left.
\begin{aligned}
x_1^{(k)} &= \tfrac{1}{10}\left\{x_2^{(k-1)} - 2x_3^{(k-1)} + 6\right\} \\
x_2^{(k)} &= \tfrac{1}{11}\left\{x_1^{(k)} + x_3^{(k-1)} - 3x_4^{(k-1)} + 25\right\} \\
x_3^{(k)} &= \tfrac{1}{10}\left\{-2x_1^{(k)} + x_2^{(k)} + x_4^{(k-1)} - 11\right\} \\
x_4^{(k)} &= \tfrac{1}{8}\left\{-3x_2^{(k)} + x_3^{(k)} + 15\right\}
\end{aligned}
\right\}
$$

Again letting $\underline{x}^{(0)} = \underline{0}^{\mathrm{T}}$

Here
$\underline{x}^{(4)} = (1.0009,\ 2.0003,\ -1.0003,\ 0.9999)$
$\underline{x}^{(5)} = (1.0001,\ 2.0000,\ -1.0000,\ 1.0000)$

$$
\frac{\left\|\underline{x}^{(5)} - \underline{x}^{(4)}\right\|_\infty}{\left\|\underline{x}^{(5)}\right\|_\infty} = 4 \times 10^{-4} \qquad (< \ \epsilon)
$$

We note that Jacobi's method required twice as many iterations for the same level of accuracy.

If $A$ is strictly diagonally dominant than for any choice of $\underline{x}^{(0)}$ the sequence of solutions generated by both Gauss-Seidel and Jacobi converge to the unique solution.

## 1.8 Successive-Over-Relaxation (SOR) Methods

A large part of numerical analysis is based upon the need to improve the efficiency and speed of existing techniques. Although we have seen from the Gauss-Seidel method that it is superior to Jacobi in this regard, we can nevertheless continue to look for improvements in convergence speed. In this section we will briefly look at a method called *successive-over-relaxation*.

Let $\underline{\hat{x}} \in \mathbb{R}^n$ be an approximation to the actual solution for $A\underline{x} = \underline{b}$.

The *residual vector* $\underline{r}$ for $\underline{\hat{x}}$ is

$$\underline{r} = \underline{b} - A\underline{\hat{x}}$$

In Jacobi/Gauss-Seidel methods a residual vector is associated with each calculation of an approximation component to the solution vector.

**Aim:** Generate a sequence of approximations that cause the associated residual vectors to converge rapidly to zero. Then methods involving equations of the form

$$x_i^{(k)} = x_i^{(k-1)} + \omega \frac{r_{ii}^{(k)}}{a_{ii}}$$

are called **relaxation methods** and for certain choices of positive $\omega$ lead to faster convergence.

For $0 < \omega < 1$, the procedures are called **under-relaxation methods** and used for convergence of systems which do not converge by Gauss-Seidel.

For $\omega > 1$, the procedures are called **over-relaxation methods** which are used to accelerate convergence of systems which are convergent by Gauss-Seidel.

**Scheme:** For Gauss-Seidel we have

$$x_i^{(k)} = x_i^{(k-1)}(1-\omega) + \frac{\omega}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1}\left(a_{i\,j}\, x_j^{(k)}\right) - \sum_{j=i+1}^{n}\left(a_{i\,j}\, x_j^{(k-1)}\right)\right]$$

for $(i = 1, \ldots\ldots, n)$.

**How do we choose the value of $\omega$?**

Firstly we introduce a few relevant ideas.

Suppose the matrix $A$ can be written in the form

$$A = D - L - U$$

where $D$ is a matrix with the diagonal elements of $A$ (zero everywhere else).

$L$ is a strictly lower-triangular part of $A$.
$U$ is strictly upper part of $A$

Then for the Jacobi method we have

$$T_j = D^{-1}(L+U)$$

and for Gauss-Seidel

$$T_g = (D-L)^{-1}U$$

**Example:** Consider

$$A = \begin{pmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{pmatrix}$$

So

$$D = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} \rightarrow D^{-1} = \frac{1}{4}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$T_j = D^{-1}(L+U) = D^{-1}(D-A)$$

$$= \begin{pmatrix} \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} \end{pmatrix}\begin{pmatrix} 0 & -3 & 0 \\ -3 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\frac{3}{4} & 0 \\ -\frac{3}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{pmatrix}$$

Given an $n \times n$ matrix $A$ the *spectral radius* $\rho(A)$ is defined as

$$\rho(A) = \max|\lambda_i| \quad i = 1, 2, \ldots\ldots, n$$

where $\lambda$ is the eigenvalue of $A$.

Note: If $\lambda = \alpha + i\beta$ then $|\lambda| = \sqrt{\alpha^2 + \beta^2}$.

In response to the earlier question regarding ? choice of $\omega$, there is no complete answer. However there are a few well kn n results (theorems) which can be used in certain situations.

If $A$ is a positive definite tridiagonal matrix then the optimal choice for $\omega$ is

$$\omega = \frac{2}{1+\sqrt{1-\rho(T_g)}} = \frac{2}{1+\sqrt{1-\rho^2(T_j)}}$$

With this choice of $\omega$, $\rho(T_\omega) = \omega - 1$.

The earlier example gave

$$T_J = \begin{pmatrix} 0 & -\frac{3}{4} & 0 \\ -\frac{3}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{pmatrix}$$

so we have

$$T_J - \lambda I = \begin{pmatrix} -\lambda & -\frac{3}{4} & 0 \\ -\frac{3}{4} & -\lambda & \frac{1}{4} \\ 0 & \frac{1}{4} & -\lambda \end{pmatrix} \rightarrow$$

$$|T_J - \lambda I| = -\lambda\left(\lambda^2 - \tfrac{5}{8}\right)$$

Thus $\rho(T_J) = \sqrt{0.625}$

and

$$\omega = \frac{2}{1 + \sqrt{1 - \rho(T_g)}} = \frac{2}{1 + \sqrt{1 - \rho^2(T_J)}} = \frac{2}{1 + \sqrt{1 - 0.625}}$$

$$\approx 1.24$$

different.

Another theorem states:

If $A$ is positive definite and $0 < \omega < 2$, then the SOR method converges for any starting guess $\underline{x}^{(0)}$

**Exercise:**

Use Gauss-Seidel to solve $A\underline{x} = \underline{b}$, where $A$ is as given above and $\underline{b} = [24,\ 30,\ -24]^T$.

Repeat this using an SOR method with $\omega = 1.25$.