Eliminación de Gauss

- Un método basado en esta transformación del sistema de ecuaciones es el método de Eliminación de Gauss
- Se basa en realizar una serie de transformaciones sobre el sistema de modo de obtener un sistema equivalente, con matriz triangular.
- En un sistema de n ecuaciones se efectuan $n-1$ pasos de eliminación.
- La matriz $\mathbf{A}$ del sistema, va siendo transformada en matrices $\mathbf{A}^{(\mathbf{k})}$ en cada paso k de la eliminación.
- El vector de términos independientes va transformándose también en sucesivos vectores $\mathbf{b}^{(\mathbf{k})}$
- Se introducirá el método a través de un ejemplo sencillo.

Eliminación de Gauss

Ejemplo:

$$\begin{array}{l} E_1 \\ E_2 \\ E_3 \\ E_4 \end{array} \qquad \begin{bmatrix} 6 & -2 & 2 & 4 \\ 12 & -8 & 6 & 10 \\ 3 & -13 & 9 & 3 \\ -6 & 4 & 1 & -18 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 34 \\ 27 \\ -38 \end{bmatrix}$$

Se propone transformar el sistema en varios pasos

En el primer paso se propone un sistema:

$$\begin{array}{l} E_1 \\ E_2 - 2E_1 \\ E_3 - \frac{1}{2}E_1 \\ E_4 - (-1)E_1 \end{array} \qquad \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & -12 & 8 & 1 \\ 0 & 2 & 3 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ 21 \\ -26 \end{bmatrix}$$

Eliminación de Gauss

En el segundo paso se propone un sistema:

$$\begin{array}{l} E_1 \\ E_2 \\ E_3 - 3E_2 \\ E_4 - (-\frac{1}{2})E_2 \end{array} \qquad \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 4 & -13 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -21 \end{bmatrix}$$

En el tercer paso:

$$\begin{array}{l} E_1 \\ E_2 \\ E_3 \\ E_4 - 2E_3 \end{array} \qquad \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -3 \end{bmatrix}$$

Eliminación de Gauss

- El sistema ha quedado transformado en uno equivalente con matriz triangular superior, que es más sencillo para resolver.
- La solución del sistema triangular se realiza por retrosustitución, dando:

$$\mathbf{x} = \begin{bmatrix} 1 \\ -3 \\ -2 \\ 1 \end{bmatrix}$$

Eliminación de Gauss

- Se forma una sucesión de sistemas con matrices $\mathbf{A}^{(k)}$
- En el paso k la $k-esima$ fila queda inalterada, igual que las filas superiores a ella. Se modifican las filas inferiores.
- En el paso k el elemento a_{kk} se denomina *pivote*. La fila k se llama *fila pivote* y la columna k *columna pivote*

Eliminación de Gauss

$$
\mathbf{A}^{(k)} = \begin{array}{cc}
 & \begin{array}{cccccccc} \phantom{a_{11}^{(k)}} & & \phantom{a_{1,k-1}^{(k)}} & col.k & & col.j & & \phantom{a_{1n}^{(k)}} \end{array} \\
\begin{array}{c} \\ \\ fila\ k \\ \\ fila\ i \\ \\ \\ \end{array} &
\begin{bmatrix}
a_{11}^{(k)} & \dots & a_{1,k-1}^{(k)} & a_{1k}^{(k)} & \dots & a_{1j}^{(k)} & \dots & a_{1n}^{(k)} \\
\vdots & \vdots & \vdots & & \ddots & \vdots & \vdots & \vdots \\
0 & \dots & 0 & a_{kk}^{(k)} & \dots & a_{kj}^{(k)} & \dots & a_{kn}^{(k)} \\
\vdots & \vdots & & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & \dots & 0 & a_{ik}^{(k)} & \dots & a_{ij}^{(k)} & \dots & a_{in}^{(k)} \\
\vdots & \vdots & & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & \dots & 0 & a_{nk}^{(k)} & \dots & a_{nj}^{(k)} & \dots & a_{nn}^{(k)}
\end{bmatrix}
\end{array}
$$

Eliminación de Gauss

Los elementos de la matriz $\mathbf{A}^{(k+1)}$ se calculan:

$$a_{ij}^{(k+1)} = \begin{cases} a_{ij}^{(k)} & si \quad i \leq k \\ a_{ij}^{(k)} - \left(\frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \right) a_{kj}^{(k)} & si \quad i \geq k+1 \;\; y \;\; j \geq k+1 \\ 0 & si \quad i \geq k+1 \;\; y \;\; j \leq k \end{cases}$$

Algoritmo para eliminación de Gauss

Input: n, $\tilde{\mathbf{A}} = [\mathbf{A}\ \mathbf{b}]$
Output: $\mathbf{x}$, o mensaje de error

Eliminacion

for $k = 1, 2, \ldots n-1$ do
  for $i = k+1, k+2, \ldots n$ do
    $m \leftarrow a_{ik}/a_{kk}$
    $a_{ik} \leftarrow 0$
    for $j = k+1, k+2, \ldots n+1$ do
      $a_{ij} \leftarrow a_{ij} - m\ a_{kj}$
    end
  end
end

```
function [x] = gauss1(A,b)
n=length(b);
A=[A b];
for k=1:n-1
for i=k+1:n
m = A(i,k)/A(k,k);
A(i,k)=0;
for j=k+1:n+1
A(i,j)=A(i,j)-m*A(k,j);
endfor
endfor
endfor
```

Algoritmo para eliminación de Gauss (cont.)

if $a_{nn} = 0 \quad \rightarrow$ mens. error: 'no hay sol. unica'

```
if (A(n,n)==0) disp('no hay sol. unica'), endif
```

Retrosustitucion

	function x=sust_atras1(A)
$x_n \ = \ a_{n,n+1}/a_{nn}$	x(n)=A(n,n+1)/A(n,n);
for $i = n-1, n-2, \ldots 1$ do	for i=n-1:-1:1
$\quad s \ \leftarrow \ a_{i,n+1}$	s=A(i,n+1);
$\quad$ for $j = i+1, i+2, \ldots n$ do	for j=i+1:n
$\qquad s \ \leftarrow \ s \ - \ a_{ij}x_j$	s=s-A(i,j)*x(j);
$\quad$ end	endfor
$\quad x_i \ \leftarrow \ s/a_{ii}$	x(i)=s/A(i,i);
end	endfor

```
function [x] = gauss1(A,b)
n=length(b);
A=[A b];
% Eliminacion
for k=1:n-1
  for i=k+1:n
    m = A(i,k)/A(k,k);
    A(i,k)=0;
    for j=k+1:n+1
      A(i,j) = A(i,j)-m*A(k,j);
    endfor
  endfor
endfor
if (A(n,n)==0)
  disp('no hay sol. unica')
endif
x=sust_atras1(A); %retrosustitucion
```

```
function x=sust_atras1(A)
x=A(:,end); %necesario para que x sea columna
n=length(x); %definimos n por ser una variable local
x(n)=A(n,n+1)/A(n,n);
for i = n-1:-1:1
  s = A(i,n+1);
  for j = i+1:n
    s = s - A(i,j)*x(j);
  endfor
  x(i) = s/A(i,i);
endfor
```