



UNL • FACULTAD
DE INGENIERÍA Y
CIENCIAS HÍDRICAS

CÁLCULO NUMÉRICO

ENTREGABLE N° 1

Tema: Métodos iterativos

Alumno: Bargas, Santiago

Ejercicio 1:

Para realizar el sistema de ecuaciones lineales dado, se utilizó la siguiente función:

```
1 function [A,b] = genMatriz(n);
2     A= zeros(n);
3     A(1,1)=1;
4     A(100,100)=1;
5     b= ones(n,1) / n^2;
6     b(1,1)=0;
7     b(100,1)=0;
8     for i=2:n-1
9         A(i,i-1)=-1;
10        A(i,i)=2;
11        A(i,i+1)=-1;
12    endfor
13 endfunction
```

A continuación se muestra la solución para $N=100$ con los métodos de Jacobi, Gauss-Seidel, SOR y eliminación de Gauss. En donde para cada método iterativo se determinó el número de iteraciones necesarias considerando una cota para el residuo de $1e-6$.

Para comprobar que los métodos iterativos convergen se utilizó el siguiente teorema:

Para que la sucesión

$$x^{(k)} = Gx^{(k-1)} + c$$

converja a x es necesario y suficiente que

$$\rho(G) < 1$$

donde ρ es el radio espectral de la matriz de iteración.

Se utilizó el siguiente código en octave para calcular el radio espectral de la matriz de iteración:

```
1 function [rhoA] = RadioEspectral(A,metodo,w)
2 % descomponemos la matriz A para calcular la matriz de iteracion
3 [L D U]=DescomponerMatriz(A);
4
5 % elegimos el metodo
6 % Jacobi
7
8 if(metodo=='ja')
9     [rhoA pos]=max(abs(eig(-inv(D)*(L+U))));
10
11 % Gauss-Seidel
12 else
13 if(metodo=='gs')
14     [rhoA pos]=max(abs(eig(-inv(D+L)*(U))));
15
16 else
17     [rhoA pos]=max(abs(eig(-inv(D+w*L)*((w-1)*D+w*U))));
18
19 endif
20 endif
21
22 endfunction
```

La matriz A puede descomponerse de manera aditiva:

$$A = L_s + D_s + U_s$$

D_s matriz diagonal con la diagonal de A ($d_{ii} = a_{ii}$)

L_s matriz triangular inferior con los términos de A excluyendo la diagonal ($l_{ij} = a_{ij}; i > j$)

U_s matriz triangular superior con los terminas de A excluyendo la diagonal ($u_{ij} = a_{ij}; i < j$)

```
1 function [L D U]=DescomponerMatriz(A)
2 n=length(A);
3 v=diag(A);
4 L=tril(A,-1);
5 U=triu(A,1);
6 D=diag(v);
7 endfunction
```

Metodo de Eliminacion Gaussiana:

No fue posible encontrar solución a través de este método, y no existen iteraciones ya que es un método directo y no iterativo.

Método de Jacobi:

Convergencia del método de Jacobi

- Un método iterativo converge si $\|G\| < 1$
- En la iteración de Jacobi: $G = (I - M^{-1}A) = (I - D_s^{-1}A)$
- Los elementos diagonales de $(D_s^{-1}A)$ valen 1
- Una norma para G: $\|G\| = \max_{1 \leq i \leq n} \sum_{j=1; j \neq i}^n |a_{ij}/a_{ii}|$
- Si una matriz es diagonalmente dominante, entonces el método de Jacobi converge para cualquier vector inicial

Para el método de Jacobi en Octave utilizamos la siguiente función:

```

1 function [x,r,it] = Jacobi(A,b,x0,maxit,tol)
2     tic();
3     n=length(A(1,:));
4     x=x0;
5     it=0;
6
7     while (it < maxit)
8         for i=1:n
9             x(i) = ( b(i) - A(i,1:i-1)*x0(1:i-1) - A(i,i+1:n)*x0(i+1:n) ) / A(i,i);
10        endfor
11
12        r(it+1) = norm(b-(A*x));
13
14        if r(it+1) < tol %then
15            break;
16        endif
17
18        x0=x;
19        it=it+1;
20
21    endwhile
22

```

En la ventana de comandos se escribió:

```
>> [xJacobi,rJacobi,itJacobi] = Jacobi(A,b,x0,1000000000000,1e-6)
```

donde los resultados fueron:

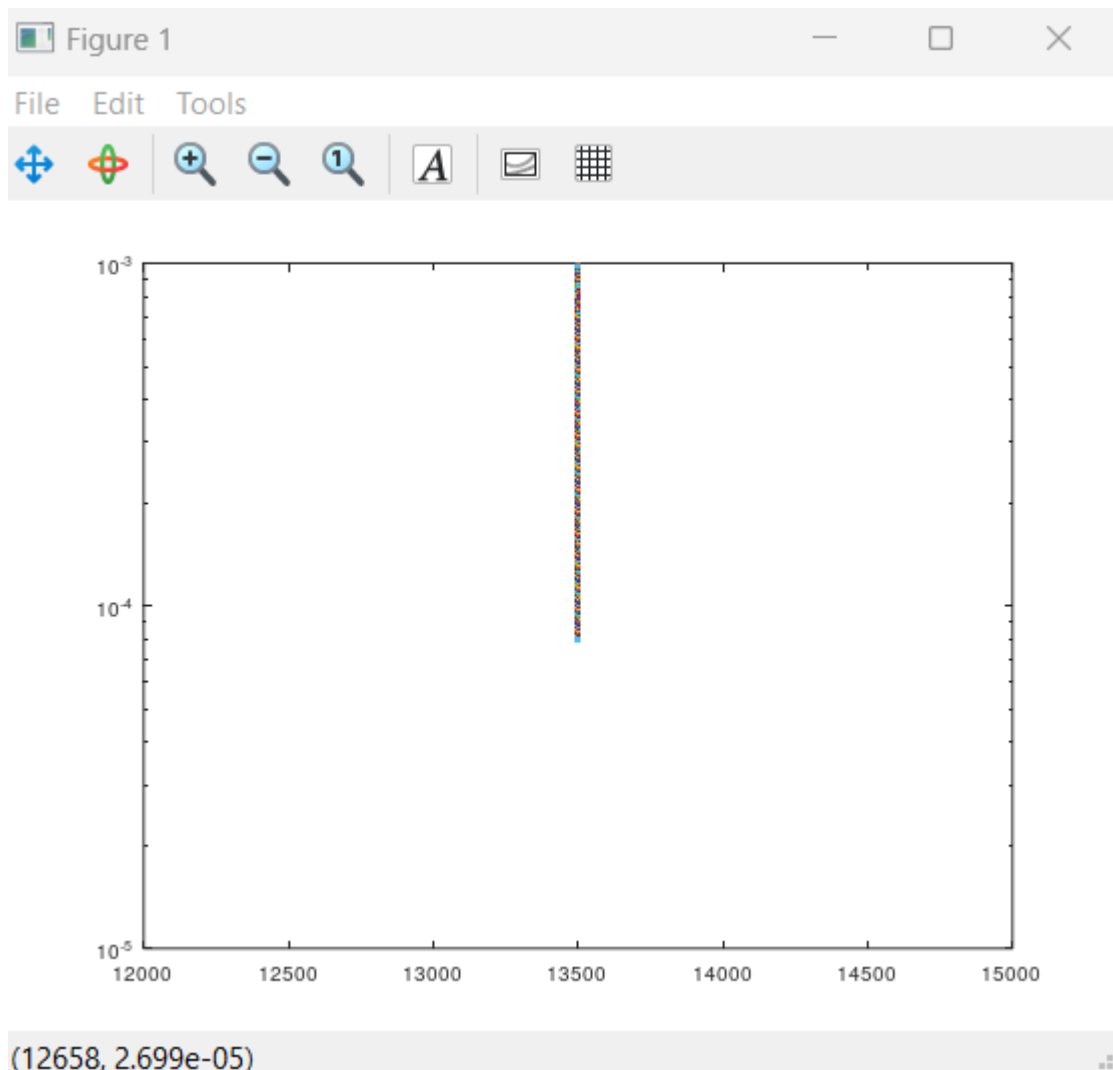
itJacobi	double	1x1	13498
rJacobi	double	1x13499	[9.8234e-04, 9.7...
xJacobi	double	100x1	[0; 4.8955e-03; ...

Como podemos observar este método necesito 13498 iteraciones para poder llegar a la convergencia.

Este método converge y lo podemos demostrar calculando el radio espectral de la matriz de iteración:

```
radioEspectralJacobi = 0.9995
```

El gráfico del residuo es el siguiente:



donde es la acumulación de errores por paso.

Método de Gauss-Seidel:

- La fórmula de iteración Gauss-Seidel:

$$Mx^{(k)} = Nx^{(k-1)} + b$$

con

$$M = D_s + L_s$$

$$N = -U_s$$

- En la iteración k , la componente i de x se calcula:

$$x_i^{(k)} = [b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^n a_{ij}x_j^{(k-1)}] / a_{ii}$$

CONVERGENCIA DEL MÉTODO DE GAUSS-SEIDEL

- Si una matriz es diagonalmente dominante entonces el método Gauss-Seidel converge

- Si una matriz es simétrica, definida positiva, entonces el método de Gauss-Seidel converge

Para el método de Gauss-Seidel en Octave utilizamos la siguiente función:

```

1 function [x, r, it, t] = GaussSeidel(A, b, x0, maxit, tol)
2     tic();
3     n = length(A(1,:));
4     x = x0; # debe inicializarse x
5     it = 0;
6     while ( it < maxit )
7         for i = 1:n
8             x(i) = (b(i) - A(i,1:i-1) * x(1:i-1) - A(i,i+1:n) * x0(i+1:n) ) / A(i,i);
9         endfor
10
11         r(it+1) = norm(A*x - b); # norm(x-x0)
12
13         if r(it+1) < tol %then
14             break;
15         endif
16
17         x0 = x;
18         it = it + 1;
19
20
21     endwhile
22     t = toc();
23 endfunction
24

```

En la ventana de comandos se escribió:

```
>> [xGaussSeidel,rGaussSeidel,itGaussSeidel] = GaussSeidel(A,b,x0,100000,1e-6)
```

donde los resultados fueron:

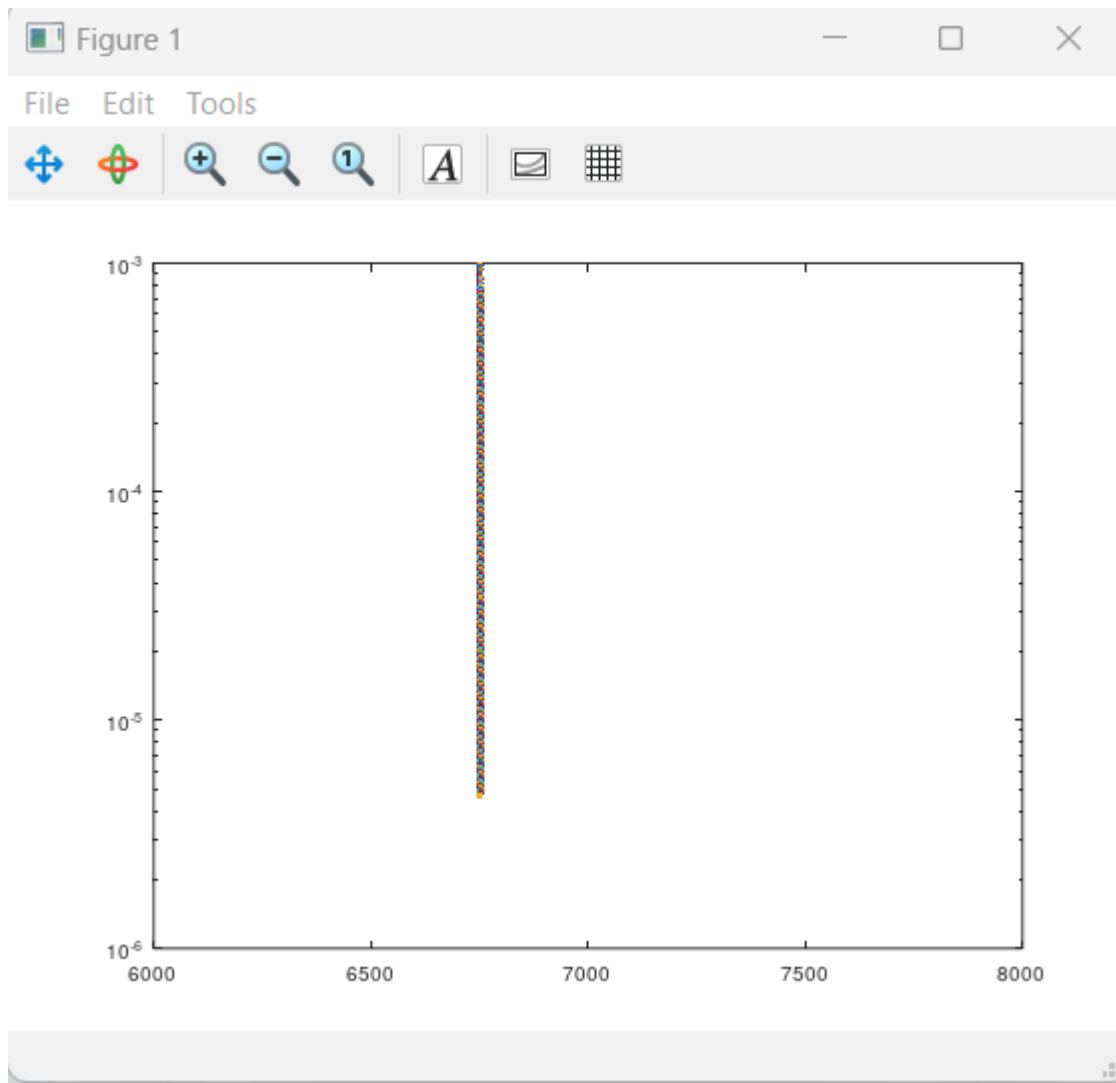
itGaussSeidel	double	1x1	6750
rGaussSeidel	double	1x6751	[9.8022e-04, 9.7...
xGaussSeidel	double	100x1	[0; 4.8954e-03; ...

Como podemos observar este método necesito 6750 iteraciones para poder llegar a la convergencia.

Este método converge y lo podemos demostrar calculando el radio espectral de la matriz de iteración:

```
radioEspectralGaussSeidel = 0.9990
```

El gráfico del residuo es el siguiente:



Método SOR:

- La fórmula de iteración SOR:

$$M_{\omega} x^{(k)} = N_{\omega} x^{(k-1)} + \omega b$$

con $M = D_s + \omega L_s$

$$N = (1 - \omega)D_s - \omega U_s$$

donde ω es un parámetro propio del método

- En la iteración k , la componente i de x se calcula:

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

- Si $\omega = 1$ SOR \rightarrow Gauss – Seidel
- Si $\rho(M_\omega^{-1}N_\omega) < 1$ SOR converge
- Si $a_{ii} \neq 0, \forall i$ entonces $\rho((M_\omega^{-1}N_\omega)) \geq |\omega - 1|$ luego debería ser $0 < \omega < 2$
- Si $0 < \omega < 1 \rightarrow$ método sub – relajación
- Si $1 < \omega < 2 \rightarrow$ método sobre – relajación
- Para una matriz se puede hallar un ω óptimo que minimice el radio espectral

Para el método SOR en Octave utilizamos la siguiente función:

```

1 function [x, rh, it, t] =sor(A, b, x0, maxit, tol, w)
2     tic();
3     n = length(A(1,:));
4     x = x0; # debe inicializarse x
5     it = 0;
6     while ( it < maxit )
7         for i = 1:n
8             x(i) = (1-w) * x0(i) + w * ( b(i) - A(i,1:i-1)*x(1:i-1)...
9                 - A(i,i+1:n)*x0(i+1:n) ) / A(i,i);
10        endfor
11        rh(it+1) = norm(A*x - b); # norm(x-x0)
12        if rh(it+1) < tol
13            break;
14        endif
15        x0 = x;
16        it = it +1;
17    endwhile
18    t = toc();
19    endfunction

```

Calculamos el ω óptimo con la formula:
$$\frac{2}{1+\sqrt{(1-\text{resp}(T_j))^2}}$$

donde resp es el radio espectral de la matriz de iteración de Jacobi. El ω óptimo es 1.9

En la ventana de comandos se escribió:

```
>> [xSOR, rSOR, itSOR, tSOR] =sor(A, b, x0, 100000, 1e-6, 1.9)
```

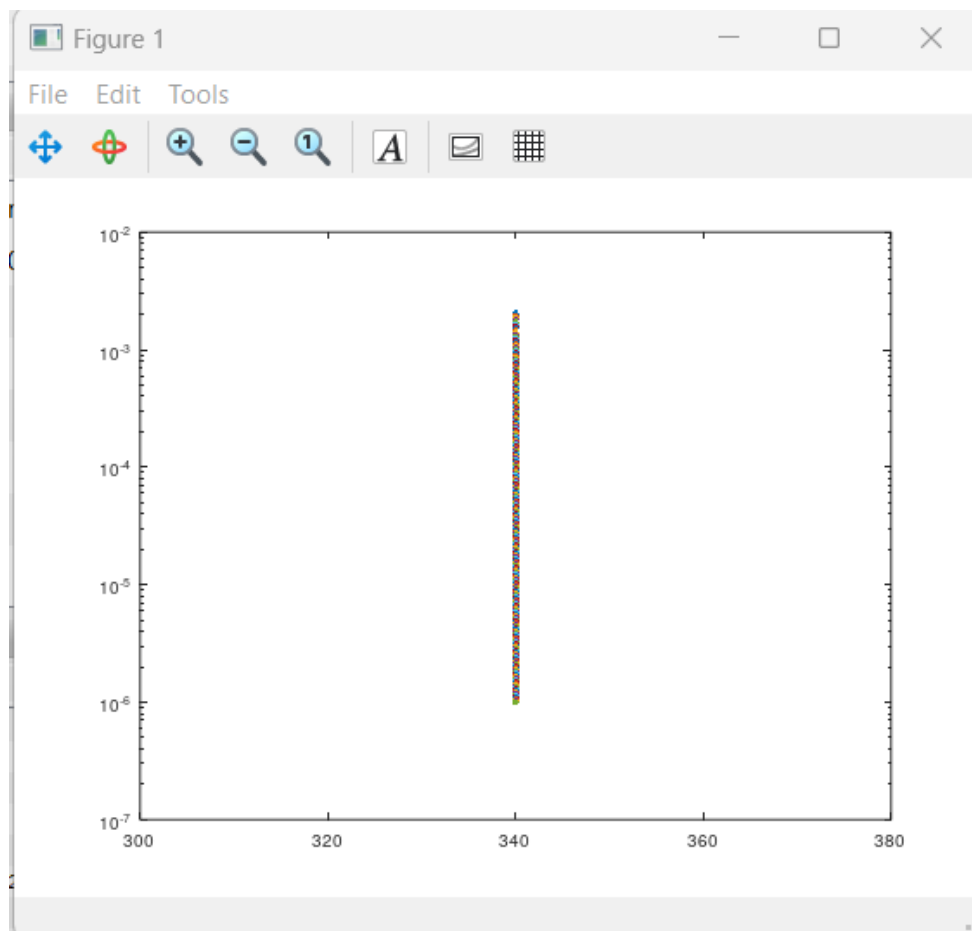
donde los resultados fueron:

itSOR	double	1x1	340
rSOR	double	1x341	[2.0369e-03, 1.9...
xSOR	double	100x1	[0; 4.8945e-03; ...

Como podemos observar este método necesito 340 iteraciones para poder llegar a la convergencia.

Este método converge ya que con $w=1$ el método SOR pasa a ser de Gauss-Seidel, y como vimos antes Gauss-Seidel converge

Su gráfica del residuo es:



Ejercicio 2

Escribimos en ChatGPT: Necesito un ejemplo de un sistema de 5 ecuaciones lineales para el cual el método de Jacobi converja y el método de Gauss Seidel no lo haga.

A lo cual respondió:

Claro, aquí tienes un ejemplo de un sistema de 5 ecuaciones lineales en 5 incógnitas donde el método de Jacobi converge pero el método de Gauss-Seidel no lo hace:

Sistema de ecuaciones:

$$\begin{aligned}2x + y &= 5 \\x + 3y + z &= 5 \\y + 4z + w &= 5 \\z + 5w + v &= 5 \\v + 6w &= 5\end{aligned}$$

En este caso, el método de Jacobi convergerá a una solución, pero el método de Gauss-Seidel no lo hará, ya que la matriz A no cumple con las condiciones necesarias para la convergencia del método de Gauss-Seidel, que requiere que la matriz sea estrictamente diagonal dominante. Puedes probar ambos métodos en Octave utilizando estas matrices y vectores para confirmar este comportamiento.

Para comprobar que este sistema tiene solución única hallamos el determinante de la matriz A, el cual es:

```
A =  
  
    2    1    0    0    0  
    1    3    1    0    0  
    0    1    4    1    0  
    0    0    1    5    1  
    0    0    0    1    6  
  
>> det(A)  
ans = 492.00
```

y como $\det(A) \neq 0$ sabemos que tiene solución única

Para verificar que lo que nos dice ChatGPT es cierto, lo comprobamos en octave, utilizamos la función para calcular el radio espectral de la matriz de iteraciones

```
radioEspectralJacobi = 0.5216
```

Como el radio espectral de la matriz de iteraciones de Jacobi es menor a 1, el método converge.

Ahora vemos para el método de Gauss-Seidel

```
|radioEspectralGauss = 0.2721
```

Como el radio espectral de la matriz de iteraciones de Gauss-Seidel es menor a 1, el método converge.

Por lo tanto ChatGPT está equivocado con el ejemplo que nos brinda.