

Informe Bloque II

Computación Científica

Aingeru García Blas

Índice

1. Introducción	2
2. Gauss Seidel y la norma infinito	3
3. Convergencia inmediata - Gauss Seidel y Jacobi	7
4. Gauss-Seidel y SOR	10
4.1. Comentario sobre el último ejercicio	15

1. Introducción

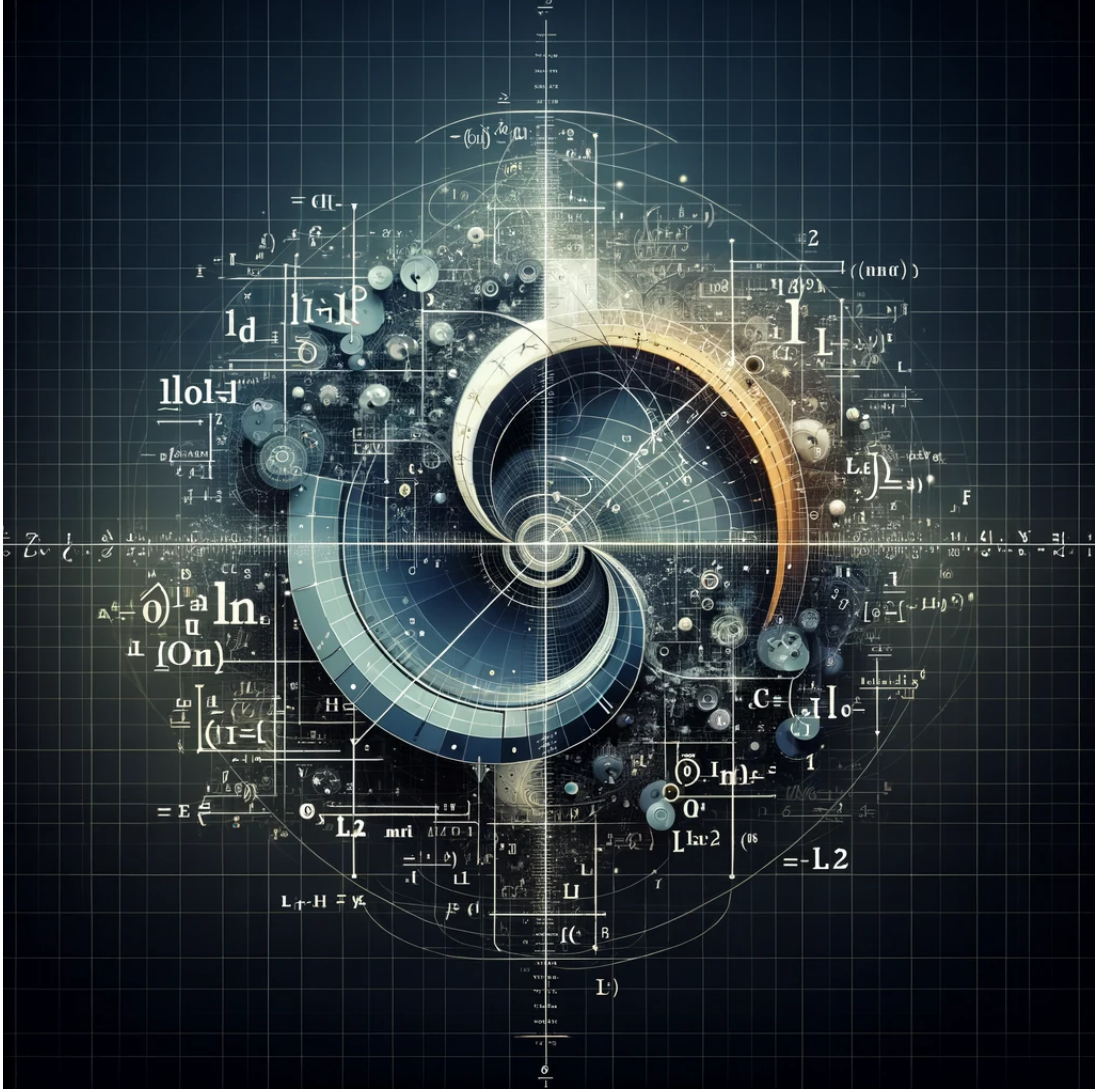


Figura 1: Algoritmos de iteración - punto fijo

Continuando con nuestro viaje por la computación científica, nos enfocamos ahora en los algoritmos de iteración de punto fijo. Estos métodos son esenciales para resolver ecuaciones complejas de una manera eficiente. A ellos dió comienzo el método de Jacobi, un enfoque clásico que resuelve cada variable de forma independiente, utilizando los valores anteriores en cada iteración.

El desarrollo de estos algoritmos no se detuvo con Jacobi. Una evolución importante llegó con el método de Gauss-Seidel, donde se emplean los valores actualizados tan pronto como están disponibles. Esto mejora la rapidez de convergencia hacia la solución, demostrando cómo pequeñas modificaciones pueden tener un gran impacto en la eficiencia del método.

Además, tenemos el método SOR (Successive Over Relaxation), que introduce un factor de relajación en el método de Gauss-Seidel para acelerar aún más la convergencia. Este ajuste nos muestra la importancia de la experimentación en la búsqueda de soluciones más rápidas y precisas.

Estos algoritmos son más que herramientas matemáticas; representan la evolución continua de técnicas para entender y modelar el mundo. No solo resolvemos problemas, sino que también avanzamos en nuestra comprensión del entorno. Sigamos adelante con entusiasmo en la exploración de estas técnicas.

2. Gauss Seidel y la norma infinito

Enunciado:

Dado el sistema de ecuaciones:

$$\begin{aligned}4x_1 - x_2 - x_3 &= -4 \\ -x_1 + 4x_2 - x_4 &= 0 \\ -x_1 + 4x_3 - x_4 &= 4 \\ -x_2 - x_3 + 4x_4 &= -4\end{aligned}$$

Realizar 3 iteraciones mediante el algoritmo de Gauss-Seidel.

Breve introducción a Gauss-Seidel

El método de Gauss-Seidel es un procedimiento iterativo para resolver sistemas de ecuaciones lineales. Se basa en la actualización sucesiva de las aproximaciones de las soluciones, utilizando la fórmula:

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

donde a_{ij} son los coeficientes de la matriz del sistema, b_i son los términos independientes, y $x_j^{(k+1)}$ y $x_j^{(k)}$ son los valores actuales y anteriores de las variables, respectivamente.

Nótese que, en el proceso desarrollado en con éste algoritmo, siempre cogemos el **mejor y más actualizado** valor para cada variable. Esto es una diferencia notable en cuanto a su predecesor **Jacobi**.

Explicado ésto, comencemos con el ejercicio.

Inicialización del método

Usamos el vector de aproximación inicial $\bar{X}_i = \frac{b_i}{a_{ii}}$, donde \mathbf{A} es la matriz de coeficientes y \mathbf{b} es el vector de términos constantes.

$$\bar{\mathbf{X}}^{(0)} = \left[\frac{-4}{4}, \frac{0}{4}, \frac{4}{4}, \frac{-4}{4} \right] = [-1, 0, 1, -1]$$

Formulación del esquema de iteración

Reorganizamos cada ecuación para despejar la variable de la diagonal principal (estrictamente dominante) e incluimos los superíndices correspondientes:

$$\begin{aligned}x_1^{(k+1)} &= \frac{1}{4}(x_2^{(k)} + x_3^{(k)} - 4) \\x_2^{(k+1)} &= \frac{1}{4}(x_1^{(k+1)} + x_4^{(k)}) \\x_3^{(k+1)} &= \frac{1}{4}(x_1^{(k+1)} + x_4^{(k)} + 4) \\x_4^{(k+1)} &= \frac{1}{4}(x_2^{(k+1)} + x_3^{(k+1)} - 4)\end{aligned}$$

Iteraciones

Realizamos las iteraciones del método de Gauss-Seidel:

Iteración 1, k=0

$$\begin{aligned}x_1^{(1)} &= \frac{1}{4}(0 + 1 - 4) = -0,75 \\x_2^{(1)} &= \frac{1}{4}(-0,75 - 1) = -0,4375 \\x_3^{(1)} &= \frac{1}{4}(-0,75 - 1 + 4) = 0,5625 \\x_4^{(1)} &= \frac{1}{4}(-0,4375 + 0,5625 - 4) = -0,96875\end{aligned}$$

Iteración 2, k=1

$$\begin{aligned}x_1^{(2)} &= \frac{1}{4}(-0,4375 + 0,5625 - 4) = -0,96875 \\x_2^{(2)} &= \frac{1}{4}(-0,96875 - 0,96875) = -0,484375 \\x_3^{(2)} &= \frac{1}{4}(-0,96875 - 0,96875 + 4) = 0,515625 \\x_4^{(2)} &= \frac{1}{4}(-0,484375 + 0,515625 - 4) = -0,9921875\end{aligned}$$

Iteración 3, k=2

$$\begin{aligned}x_1^{(3)} &= \frac{1}{4}(-0,484375 + 0,515625 - 4) = -0,9921875 \\x_2^{(3)} &= \frac{1}{4}(-0,9921875 - 0,9921875) = -0,49609375 \\x_3^{(3)} &= \frac{1}{4}(-0,9921875 - 0,9921875 + 4) = 0,50390625 \\x_4^{(3)} &= \frac{1}{4}(-0,49609375 + 0,50390625 - 4) = -0,99804688\end{aligned}$$

Resumiendo, después de las distintas iteraciones obtenemos:

- Iteración 1 (k=1):

$$\bar{\mathbf{X}}^{(1)} = [-0,75, -0,4375, 0,5625, -0,96875]$$

- Iteración 2 (k=2):

$$\bar{\mathbf{X}}^{(2)} = [-0,96875, -0,484375, 0,515625, -0,9921875]$$

- Iteración 3 (k=3):

$$\bar{\mathbf{X}}^{(3)} = [-0,9921875, -0,49609375, 0,50390625, -0,99804688]$$

Cálculo del error relativo con norma infinito

Calculamos el error relativo entre la tercera y la segunda iteración, y también entre la tercera iteración y la solución exacta $\bar{\mathbf{S}}$, usando la norma infinito. Para ello, recordemos el concepto de norma y las 3 formas vistas en clase:

Podemos ver la norma como una función que asigna una longitud o tamaño a los vectores de un espacio. Esta función, denotada habitualmente como $\|\cdot\|$, cumple con ciertas propiedades que la hacen una herramienta fundamental en el análisis de espacios vectoriales. La norma puede ser vista como una **vara de medir** en estos espacios, proporcionando una forma de cuantificar la magnitud o distancia de los vectores.

Las 3 vertientes vistas en clase son las siguientes:

Veamos primero la **Norma Euclidiana**:

$$\|\mathbf{X}\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

De igual manera, la **norma de 1** (o norma L^1) suma los valores absolutos de las componentes del vector, reflejando una medida diferente de su tamaño:

$$\|\mathbf{X}\|_1 = |x_1| + |x_2| + \cdots + |x_n|$$

Por último, la norma que usaremos en este ejercicio: **norma infinito** (o norma L^∞) se define como el valor absoluto máximo de las componentes del vector, proporcionando otra perspectiva sobre su magnitud:

$$\|\mathbf{X}\|_\infty = \max(|x_1|, |x_2|, \dots, |x_n|)$$

Error entre $\bar{\mathbf{X}}^{(3)}$ y $\bar{\mathbf{X}}^{(2)}$

Primero, calculamos la diferencia entre las iteraciones:

$$\bar{\mathbf{X}}^{(3)} - \bar{\mathbf{X}}^{(2)} = \begin{bmatrix} -0,9921875 \\ -0,49609375 \\ 0,50390625 \\ -0,99804688 \end{bmatrix} - \begin{bmatrix} -0,96875 \\ -0,484375 \\ 0,515625 \\ -0,9921875 \end{bmatrix}$$

El resultado es:

$$\bar{\mathbf{X}}^{(3)} - \bar{\mathbf{X}}^{(2)} = \begin{bmatrix} -0,0234375 \\ -0,01171875 \\ -0,01171875 \\ -0,00585938 \end{bmatrix}$$

Luego, calculamos el error relativo:

$$\text{Error Relativo} = \frac{\|\bar{\mathbf{X}}^{(3)} - \bar{\mathbf{X}}^{(2)}\|_{\infty}}{\|\bar{\mathbf{X}}^{(3)}\|_{\infty}} = \frac{0,0234375}{0,99804688} \approx 0,0234$$

Error entre $\bar{\mathbf{X}}^{(3)}$ y \mathbf{S}

La solución exacta es $\mathbf{S} = [-1, -0,5, 0,5, -1]$. Calculamos la diferencia:

$$\bar{\mathbf{X}}^{(3)} - \mathbf{S} = \begin{bmatrix} -0,9921875 \\ -0,49609375 \\ 0,50390625 \\ -0,99804688 \end{bmatrix} - \begin{bmatrix} -1 \\ -0,5 \\ 0,5 \\ -1 \end{bmatrix}$$

El resultado es:

$$\bar{\mathbf{X}}^{(3)} - \mathbf{S} = \begin{bmatrix} 0,0078125 \\ 0,00390625 \\ 0,00390625 \\ 0,00195312 \end{bmatrix}$$

Finalmente, el error relativo es:

$$\text{Error Relativo} = \frac{\|\bar{\mathbf{X}}^{(3)} - \mathbf{S}\|_{\infty}}{\|\mathbf{S}\|_{\infty}} = \frac{0,0078125}{1} \approx 0,0079$$

3. Convergencia inmediata - Gauss Seidel y Jacobi

Enunciado:

Dado el sistema lineal $\mathbf{A}\bar{\mathbf{x}} = \bar{\mathbf{b}}$, donde

$$\mathbf{A} = \begin{pmatrix} 0,96326 & 0,81321 \\ 0,81321 & 0,68654 \end{pmatrix}, \quad \bar{\mathbf{b}} = \begin{pmatrix} 0,88824 \\ 0,74988 \end{pmatrix}$$

y el vector inicial $\bar{\mathbf{X}}^{(0)} = \begin{pmatrix} 0,33116 \\ 0,70000 \end{pmatrix}$.

Se pide resolver el sistema con el método de Gauss-Seidel y Jacobi, a una precisión de $p = 5$.

Formulación del método

Aunque la matriz \mathbf{A} no tiene una diagonal dominante, procedemos a reorganizar el sistema para obtener un esquema iterativo de Gauss-Seidel, ya que la condición de diagonal dominante es **suficiente pero no necesaria**.

Sistema de Ecuaciones

A continuación, transformamos los elementos proporcionados en su sistema de ecuaciones:

El sistema original es:

$$\begin{aligned} 0,96326x_1 + 0,81321x_2 &= 0,88824 \\ 0,81321x_1 + 0,68654x_2 &= 0,74988 \end{aligned}$$

Transformación para algoritmos de punto fijo

Para aplicar algoritmos de iteración de punto fijo, transformamos el sistema de ecuaciones lineales a una forma-ecuación que se ajusta a estos métodos. La forma general de un sistema adaptado a algoritmos de punto fijo es:

$$\bar{\mathbf{X}} = \mathbf{T}\bar{\mathbf{X}} + \bar{\mathbf{C}}$$

Donde $\bar{\mathbf{X}}$ representa el vector de variables, \mathbf{T} es una matriz que define las relaciones entre estas variables, y $\bar{\mathbf{C}}$ es un vector constante.

Esta transformación implica reescribir el sistema de ecuaciones de tal manera que cada ecuación se convierte en una expresión que define una variable en términos de las otras. El objetivo es obtener un esquema iterativo en el cual el nuevo valor de \mathbf{X} en cada iteración se calcula en función de los valores previos. Esta estructura es fundamental para la implementación de algoritmos de punto fijo, permitiendo una aproximación progresiva a la solución del sistema, iteración a iteración.

Despejamos pues x_1 y x_2 para obtener el esquema iterativo:

$$x_1^{(k+1)} = \frac{1}{0,96326}(-0,81321x_2^{(k)} + 0,88824)$$

$$x_2^{(k+1)} = \frac{1}{0,68654}(-0,81321x_1^{(k+1)} + 0,74988)$$

Iteraciones

Con el vector inicial $\mathbf{x}^{(0)}$, realizamos las iteraciones:

Iteración 1, k=0

$$x_1^{(1)} = \frac{1}{0,96326}(-0,81321 \times 0,70000 + 0,88824) = 0,33115981$$

$$x_2^{(1)} = \frac{1}{0,68654}(-0,81321 \times 0,33115981 + 0,74988) = 0,69999931$$

Iteración 2, k=1

$$x_1^{(2)} = \frac{1}{0,96326}(-0,81321 \times 0,69999931 + 0,88824) = 0,33116039$$

$$x_2^{(2)} = \frac{1}{0,68654}(-0,81321 \times 0,33116039 + 0,74988) = 0,69999863$$

Como el enunciado menciona que $p=5$, únicamente se debe de considerar las 5 cifras más significativas, pese a haber dejado más. Con lo que, se considera que tras iterar la aproximación no cambia.

Método de Jacobi

El método de Jacobi, siendo el precursor de los algoritmos de punto fijo, se basa en la actualización simultánea de todas las variables. Aunque no es el más eficiente, se le respeta por ser el *ancestro* de estos métodos y es siempre mencionado en las introducciones al contexto de éste tipo de herramientas.

Esquema Iterativo de Jacobi

El esquema iterativo de Jacobi para este sistema es el siguiente:

$$x_1^{(k+1)} = \frac{1}{0,96326}(-0,81321x_2^{(k)} + 0,88824)$$

$$x_2^{(k+1)} = \frac{1}{0,68654}(-0,81321x_1^{(k)} + 0,74988)$$

Observemos que en éste caso, no utilizamos el mejor y más actualizado valor para cada variable en las ecuaciones - lo que realizamos en Gauss-Seidel y otros, es fruto de **optimización del algoritmo original de Jacobi**.

Iteraciones de Jacobi

Realizamos una iteración del método de Jacobi partiendo del mismo vector inicial.

Iteración 1, k=0

$$x_1^{(1)} = \frac{1}{0,96326}(-0,81321 \times 0,70000 + 0,88824) = 0,33115981$$

$$x_2^{(1)} = \frac{1}{0,68654}(-0,81321 \times 0,33116 + 0,74988) = 0,69999909$$

Análisis de la convergencia

En ambos métodos, Gauss-Seidel y Jacobi, observamos una convergencia inmediata de las iteraciones. Esto significa que los valores de las variables se estabilizan rápidamente hacia la solución exacta del sistema, incluso después de solo una o dos iteraciones. La convergencia inmediata se puede atribuir a varios factores:

1. Los valores iniciales proporcionados están muy cerca de la solución real (o incluso son la propia solución), lo que permite una rápida aproximación a la solución exacta.
2. El sistema de ecuaciones puede tener propiedades particulares que facilitan una convergencia rápida, como una buena condición de la matriz o una relación adecuada entre los coeficientes.

Estos resultados indican que en ciertas condiciones, especialmente con una buena elección del vector inicial y propiedades favorables del sistema, los métodos iterativos pueden converger rápidamente, minimizando la necesidad de múltiples iteraciones.

4. Gauss-Seidel y SOR

Enunciado:

Consideramos el sistema de ecuaciones lineales dado por:

$$\begin{aligned} -2x_1 + x_2 &= -1 \\ x_1 - 2x_2 + x_3 &= 0 \\ x_2 - 2x_3 + x_4 &= 0 \\ x_3 - 2x_4 &= 0 \end{aligned}$$

- Resolver mediante Gauss-Seidel tomando $\bar{X}_i = \frac{b_i}{a_{ii}}$ como vector inicial y Tolerancia

$$T = 0,5 \times 10^{-3}$$

- Realizar 3 iteraciones mediante el algoritmo de SOR

Inicialización

El vector inicial \bar{X}_i se calcula como $\bar{X}_i = \frac{b_i}{a_{ii}}$. Por lo tanto, tenemos:

$$\begin{aligned} x_1^{(0)} &= \frac{-1}{-2} = 0,5 \\ x_2^{(0)} &= \frac{0}{-2} = 0 \\ x_3^{(0)} &= \frac{0}{-2} = 0 \\ x_4^{(0)} &= \frac{0}{-2} = 0 \end{aligned}$$

Formulación del esquema iterativo:

Veamos como resulta el esquema para este caso, tras eliminar los signos negativos:

$$\begin{aligned} x_1^{(k+1)} &= \frac{1}{2}(x_2^{(k)} + 1) \\ x_2^{(k+1)} &= \frac{1}{2}(+x_1^{(k+1)} + x_3^{(k)}) \\ x_3^{(k+1)} &= \frac{1}{2}(+x_2^{(k+1)} + x_4^{(k)}) \\ x_4^{(k+1)} &= \frac{1}{2}(+x_3^{(k+1)}) \end{aligned}$$

Iteración 1, k=0

Aplicamos la fórmula de Gauss-Seidel:

$$\begin{aligned}x_1^{(1)} &= \frac{x_2^{(0)}}{2} = \frac{1}{2} = 0,5 \\x_2^{(1)} &= \frac{x_1^{(1)} + x_3^{(0)}}{2} = \frac{0,5 + 0}{2} = 0,25 \\x_3^{(1)} &= \frac{x_2^{(1)} + x_4^{(0)}}{2} = \frac{1}{2} \cdot \frac{1}{4} = 0,125 \\x_4^{(1)} &= \frac{x_3^{(1)}}{2} = \frac{0,125}{2} = 0,0625\end{aligned}$$

Cálculo de la Tolerancia para $k = 0$

En métodos iterativos, el cálculo de la tolerancia es crucial para determinar el momento adecuado para detener las iteraciones. La tolerancia se calcula como sigue:

$$\text{Tolerancia} \geq \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_\infty}{\|\mathbf{x}^{(k+1)}\|_\infty}$$

donde $\|\cdot\|_\infty$ denota la norma infinito, $\mathbf{x}^{(k)}$ es el vector de la iteración actual y $\mathbf{x}^{(k+1)}$ es el vector de la siguiente iteración.

Si esta tolerancia es menor que un umbral preestablecido, entonces se considera que el método ha convergido y se puede detener el proceso iterativo. En este ejercicio, utilizamos una tolerancia T definida como:

$$T = 0,5 \times 10^{-3}$$

Dado el vector inicial $\mathbf{x}^{(0)} = [\frac{1}{2}, 0, 0, 0]$ y el resultado de la iteración $\mathbf{x}^{(1)} = [0,5, 0,25, 0,125, 0,0625]$, calculamos la tolerancia:

$$\text{Tolerancia} \geq \frac{\|\mathbf{x}^{(1)} - \mathbf{x}^{(0)}\|_\infty}{\|\mathbf{x}^{(1)}\|_\infty}$$

El cálculo:

$$\text{Tolerancia} \geq \frac{\| [0,5, 0,25, 0,125, 0,0625] - [\frac{1}{2}, 0, 0, 0] \|_\infty}{\| [0,5, 0,25, 0,125, 0,0625] \|_\infty} = \frac{0,25}{0,5} = 0,5$$

Comparando este resultado con la tolerancia establecida de $0,5 \times 10^{-3}$, observamos que el criterio de parada no se cumple en esta iteración.

Iteración 2, k=1

Repetimos el procedimiento:

$$\begin{aligned}x_1^{(2)} &= \frac{x_2^{(1)} + 1}{2} = \frac{5}{8} = 0,625 \\x_2^{(2)} &= \frac{x_1^{(2)} + x_3^{(1)}}{2} = \frac{0,625 + 0,125}{2} = 0,375 \\x_3^{(2)} &= \frac{x_2^{(2)} + x_4^{(1)}}{2} = \frac{0,375 + 0,0625}{2} = 0,21875 \\x_4^{(2)} &= \frac{x_3^{(2)}}{2} = \frac{0,21875}{2} = 0,109375\end{aligned}$$

Calculamos la tolerancia para la Iteración 2:

$$\text{Tolerancia} \geq \frac{\|\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\|_{\infty}}{\|\mathbf{x}^{(2)}\|_{\infty}}$$

Con:

$$\text{Tolerancia} \geq \frac{\| [0,625, 0,375, 0,21875, 0,109375] - [0,5, 0,25, 0,125, 0,0625] \|_{\infty}}{\| [0,625, 0,375, 0,21875, 0,109375] \|_{\infty}} = \frac{0,125}{0,625} = 0,2$$

Iteración 3, k=2

Una vez más, aplicamos la fórmula:

$$\begin{aligned}x_1^{(3)} &= \frac{x_2^{(2)} + 1}{2} = \frac{1 + 0,375}{2} = 0,6875 \\x_2^{(3)} &= \frac{x_1^{(3)} + x_3^{(2)}}{2} = \frac{0,6875 + 0,21875}{2} = 0,453125 \\x_3^{(3)} &= \frac{x_2^{(3)} + x_4^{(2)}}{2} = \frac{0,453125 + 0,109375}{-2} = 0,28125 \\x_4^{(3)} &= \frac{x_3^{(3)}}{22} = \frac{0,28125}{2} = 0,140625\end{aligned}$$

Para el cálculo de la tolerancia en la iteración 3, usamos la fórmula:

$$\text{Tolerancia} \geq \frac{\|\mathbf{x}^{(3)} - \mathbf{x}^{(2)}\|_{\infty}}{\|\mathbf{x}^{(3)}\|_{\infty}}$$

$$\text{Tolerancia} \geq \frac{\| [0,6875, 0,453125, 0,28125, 0,140625] - [0,625, 0,375, 0,21875, 0,109375] \|_{\infty}}{\| [0,6875, 0,453125, 0,28125, 0,140625] \|_{\infty}} = \frac{0,078125}{0,6875} = 0,11$$

Comentario sobre éste resultado al final del ejercicio.

Desarrollo del método SOR

Consideramos el sistema de ecuaciones lineales dado por:

$$\begin{aligned}-2x_1 + x_2 &= -1 \\ x_1 - 2x_2 + x_3 &= 0 \\ x_2 - 2x_3 + x_4 &= 0 \\ x_3 - 2x_4 &= 0\end{aligned}$$

El método SOR (Successive Over-Relaxation) se aplica según el siguiente esquema, donde $\omega = 1,2$:

Esquema del Método SOR

El método SOR (Successive Over-Relaxation) para un sistema de ecuaciones lineales se puede expresar según el siguiente esquema, donde ω es el factor de relajación:

Para la i -ésima ecuación del sistema, la fórmula de actualización para la variable x_i en la iteración $k+1$ es:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{\text{coef}(x_i)} \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right)$$

Donde:

- ω es el factor de relajación.
- $\text{coef}(x_i)$ es el coeficiente de x_i en la i -ésima ecuación.
- a_{ij} son los coeficientes de la matriz del sistema.
- b_i es el término constante de la i -ésima ecuación.
- $x_j^{(k)}$ es el valor de la variable x_j en la iteración k .

Este esquema se aplica de manera iterativa para todas las variables x_i del sistema hasta alcanzar la convergencia deseada. Es el *patrón* que sacamos de aplicar el algoritmo y observar **las estructuras de recurrencia**, con lo que evitamos tener que operar cada vez: Veamos como queda para este caso:

$$\begin{aligned}x_1^{(k+1)} &= x_1^{(k)} + \frac{\omega}{-2}(-1 + 2x_1^{(k)} - x_2^{(k)}) \\ x_2^{(k+1)} &= x_2^{(k)} + \frac{\omega}{-2}(-x_1^{(k+1)} + 2x_2^{(k)} - x_3^{(k)}) \\ x_3^{(k+1)} &= x_3^{(k)} + \frac{\omega}{-2}(-x_2^{(k+1)} + 2x_3^{(k)} - x_4^{(k)}) \\ x_4^{(k+1)} &= x_4^{(k)} + \frac{\omega}{-2}(-x_3^{(k+1)} + 2x_4^{(k)})\end{aligned}$$

Iteración 1:

$$\begin{aligned}x_1^{(1)} &= x_1^{(0)} + \frac{1,2}{-2} \left(-1 + 2x_1^{(0)} - x_2^{(0)} \right) \\&= 0,5 + \frac{1,2}{-2} (-1 + 2 \cdot 0,5 - 0) \\&= 0,5 \\x_2^{(1)} &= x_2^{(0)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_1^{(1)} + 1 \cdot x_3^{(0)} \right) \\&= 0 + \frac{1,2}{-2} (0 - 1 \cdot 0,5 + 0) \\&= 0,3 \\x_3^{(1)} &= x_3^{(0)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_2^{(1)} + 1 \cdot x_4^{(0)} \right) \\&= 0 + \frac{1,2}{-2} (0 - 1 \cdot 0,3 + 0) \\&= 0,18 \\x_4^{(1)} &= x_4^{(0)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_3^{(1)} \right) \\&= 0 + \frac{1,2}{-2} (0 - 1 \cdot 0,18) \\&= 0,108\end{aligned}$$

Iteración 2:

$$\begin{aligned}x_1^{(2)} &= x_1^{(1)} + \frac{1,2}{-2} \left(-1 + 2x_1^{(1)} - x_2^{(1)} \right) \\&= 0,5 + \frac{1,2}{-2} (-1 + 2 \cdot 0,5 - 0,3) \\&= 0,68 \\x_2^{(2)} &= x_2^{(1)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_1^{(2)} + 1 \cdot x_3^{(1)} \right) \\&= 0,3 + \frac{1,2}{-2} (0 - 1 \cdot 0,68 + 0,18) \\&= 0,6 \\x_3^{(2)} &= x_3^{(1)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_2^{(2)} + 1 \cdot x_4^{(1)} \right) \\&= 0,18 + \frac{1,2}{-2} (0 - 1 \cdot 0,6 + 0,108) \\&= 0,4752 \\x_4^{(2)} &= x_4^{(1)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_3^{(2)} \right) \\&= 0,108 + \frac{1,2}{-2} (0 - 1 \cdot 0,4752) \\&= 0,39312\end{aligned}$$

Iteración 3:

$$\begin{aligned}
x_1^{(3)} &= x_1^{(2)} + \frac{1,2}{-2} \left(-1 + 2x_1^{(2)} - x_2^{(2)} \right) \\
&= 0,68 + \frac{1,2}{-2} (-1 + 2 \cdot 0,68 - 0,6) \\
&= 0,824 \\
x_2^{(3)} &= x_2^{(2)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_1^{(3)} + 1 \cdot x_3^{(2)} \right) \\
&= 0,6 + \frac{1,2}{-2} (0 - 1 \cdot 0,824 + 0,4752) \\
&= 0,80928 \\
x_3^{(3)} &= x_3^{(2)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_2^{(3)} + 1 \cdot x_4^{(2)} \right) \\
&= 0,4752 + \frac{1,2}{-2} (0 - 1 \cdot 0,80928 + 0,39312) \\
&= 0,724896 \\
x_4^{(3)} &= x_4^{(2)} + \frac{1,2}{-2} \left(0 - 1 \cdot x_3^{(3)} \right) \\
&= 0,39312 + \frac{1,2}{-2} (0 - 1 \cdot 0,724896) \\
&= 0,8280576
\end{aligned}$$

4.1. Comentario sobre el último ejercicio

En este ejercicio he tenido un problema; el profesor mencionó que con 3 iteraciones del algoritmo Gauss-Seidel debería de bastar para llegar al rango o umbral de tolerancia establecido.

Sin embargo, tras hacerlo varias veces y probar diferentes mecanismos y despejes para producir diferentes esquemas e incluso continuar iterando; no he conseguido lograr que la tolerancia de una iteración con la anterior sea menor que la establecida por el enunciado.

Estoy convencido de que habré cometido algún error (posiblemente) ínfimo en algún lugar, pero desafortunadamente no he conseguido encontrar donde. No dispongo de mucho tiempo y he tomado la decisión de entregarlo de ésta manera, muy a mi pesar. No obstante, no me doy por vencido y esta lucha no ha terminado; durante los próximos días volveré a abordarlo con ojos frescos con intención de por fin encontrar la solución.

«No he fracasado. He encontrado 10,000 maneras que no funcionan»

–Thomas A. Edison

13 de Noviembre de 2023