Tarea 4-Métodos numéricos

1st Daniel Vallejo Aldana
Maestría en Ciencias de la Computación
Centro de Investigación en Matemáticas
daniel.vallejo@cimat.mx

***Resumen*—El presente trabajo se basa en la solución de sistemas de ecuaciones, en la primera parte se aborda el método de Cholesky, un método analítico diseñado para factorizar matrices positivas definidas A en un producto de matrices triangulares LL^T, en la siguiente parte de este trabajo se propone utilizar este método para resolver el problema elíptico unidimensional. Finalmente, se describen dos métodos iterativos, el método de Jacobi y el método de Gauss-Seidel para aproximar el vector solución de forma iterativa a partir de una solución inicial $\mathbf{x}_0$**

***Index Terms*—Cholesky, Métodos iterativos, Jacobi, Gauss-Seidel, Sistemas de ecuaciones**

I. Introducción

Como se pudo observar en la tarea 3, el problema de resolver sistemas de ecuaciones puede ser atacado con diferentes puntos de vista, uno de ellos es factorizar las matrices de tal forma que utilizando sustitución hacia adelante y sustitución hacia atrás podamos encontrar una solución al sistema $Ax = b$. En el presente trabajo se abordará un caso especial de dichas factorizaciones que se cumple cuando A es una matriz simétrica y definida positiva. Dicha factorización es la factorización de Cholesky, donde A se factoriza de la forma $A = LL^T$ con L una matriz triangular inferior. Otra forma de atacar el problema de resolver sistemas de ecuaciones es mediante el uso de métodos iterativos, donde se propone una solución inicial $\mathbf{x}_0$ y se va actualizando el vector $\mathbf{x}^t$ de acuerdo a una serie de reglas. En el presente trabajo describiremos los métodos de Jacobi y Gauss-Seidel para solución de sistemas de ecuaciones y los compararemos con el método de Cholesky en la solución del problema elíptico unidimensional.

II. Método/Algoritmo

II-A. *Factorización de Cholesky y sus variantes*

II-A1. Método de factorización de Cholesky: De acuerdo a lo visto en la tarea 3, podemos factorizar una matriz A de $n \times n$ en un producto de matrices de la forma $A = LDU$, en particular, para una matriz A positiva definida y simétrica podemos convertir $U = L^T$ podemos hacer $D = I$, por lo que podemos descomponer a la matriz A de la forma $A = LL^T$ donde L es una matriz triangular inferior. Este algoritmo de descomposición de matrices es el algoritmo de Cholesky [1]. De acuerdo a [1], las entradas de la matriz L, se actualizan de la siguiente forma

$$l_{j,i} = \begin{cases} \sqrt{a_{1,1}} \text{ para } i = j = 1 \\ \frac{a_{j,1}}{l_{1,1}} \text{ para } i = 1 j \in \{2, \dots n\} \\ \sqrt{a_{j,i} - \sum_{k=i}^{i-1} l_{i,k}^2} \\ \text{ para } i = j, i, j \in \{2, .., n-1\} \\ \frac{\left(a_{j,i} - \sum_{k=1}^{i-1} l_{j,k} l_{i,k}\right)}{l_{i,i}} \text{ para } j \in \{i+1, n\} \\ i \in \{2, .., n-1\} \\ \sqrt{a_{n,n} - \sum_{k=i}^{n-1} l_{n,k}^2} \text{ para } i = j = n \end{cases}$$

El pseudocódigo para el algoritmo de factorización de Cholesky se muestra a continuación

Algorithm 1 CHOLESKY

Require: A, simétrica positiva definida
Ensure: L tal que $A = LL^T$

$l_{1,1} = \sqrt{a_{1,1}}$
for j from 2 to n **do**
  $l_{j,1} = \frac{a_{j,1}}{l_{1,1}}$
end for
for i from 2 to n-1 **do**
  $l_{i,i} = \sqrt{a_{i,i} - \sum_{k=1}^{i-1} l_{i,k}^2}$
  for j from i+1 to n **do**
    $l_{j,i} = \frac{\left(a_{j,i} - \sum_{k=1}^{i-1} l_{j,k} l_{i,k}\right)}{l_{i,i}}$
  end for
end for
$l_{n,n} = \sqrt{a_{n,n} - \sum_{k=1}^{n-1} l_{n,k}}^2$
return L

II-A2. Método de Cholesky para matrices tridiagonales: Un caso especial de matrices es cuando A es de la forma

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & 0 \\ a_{2,1} & a_{2,2} & & \\ \vdots & & \ddots & a_{n-1,n} \\ 0 & \dots & a_{n,n-1} & a_{n,n} \end{bmatrix}$$

, a este tipo de matrices se les conoce como matrices bandadas, en particular si el ancho de banda es 2 entonces tenemos una matriz tridiagonal. Para este caso especial consideraremos la factorización de Cholesky de la matriz A. Consideremos entonces una matriz L de la siguiente forma

$$\begin{bmatrix} l_{1,1} & 0 & \dots & 0 \\ l_{2,1} & \ddots & 0 & 0 \\ \vdots & l_{i,i-1} & l_{i,i} & 0 \\ l_{n,1} & \dots & \dots & l_{n,n} \end{bmatrix}$$

Haciendo los cálculos de los productos de submatrices podemos ver que

$$l_{j,i} = \begin{cases} \sqrt{a_{1,1}} \text{ para } i = j = 1 \\ \frac{a_{j,1}}{l_{1,1}} \text{ para } i = 1, j \leq bandwidth \\ \sqrt{a_{i,i} - l_{i,i-1}^2} \text{ para } i = j \text{ y } i, j \geq 2 \\ \frac{a_{i,j}}{l_{i,i}} \text{ para } |i - j| = 1, i \neq j \end{cases}$$

Lo anterior nos dice que L es una matriz con estructura y que es de la forma

$$\begin{bmatrix} l_{1,1} & 0 & \dots & 0 \\ l_{2,1} & \ddots & 0 & 0 \\ \vdots & l_{i,i-1} & l_{i,i} & 0 \\ 0 & \dots & \dots & l_{n,n} \end{bmatrix}$$

II-A3. Solución al problema elíptico unidimensional mediante diferencias finitas: Consideremos ahora el problema elíptico unidimensional descrito en la siguiente ecuación

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} = 2 \text{ para } 0 < x < 1 \\ u(0) = 0 \\ u(1) = 2 \end{cases}$$

Utilizando la aproximación por diferencias finitas para la segunda derivada, llamemos u_i al valor de la función u que toma al ser evaluada en el punto x_i dentro de la discretización del intervalo $[0, 1]$ con N puntos. De lo anterior podemos ver que

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

Donde h es la distancia entre los puntos x_i y x_{i+1}, en este caso consideramos $h = \frac{1}{N}$ En este caso u_i se convierten en nuestras incógnitas que queremos estimar para resolver la ecuación diferencial anterior. Trasladando el problema anterior a un problema de solución de sistemas de ecuaciones, podemos ver que para resolver el problema elíptico unidimensional debemos resolver el siguiente sistema

$$\frac{1}{h^2} \begin{bmatrix} -2 & 1 & \dots & 0 \\ 1 & -2 & 1 & 0 \\ \vdots & & \ddots & 1 \\ 0 & \dots & 1 & -2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ \vdots \\ 2 - \frac{2}{h^2} \end{bmatrix}$$

Para poder utilizar el método de factorización de Cholesky para resolver dicho sistema de ecuaciones hay que multiplicar ambos lados de la igualdad por -1. Llamemos A, a la matriz asociada al problema elíptico unidimensional, sea **u** el vector de $N - 2$ incógnitas y sea **b** el vector resultado, entonces el pseudocódigo para resolver este problema es el siguiente

Algorithm 2 RESOLVER-ELIPTICO-1D

```
Require: A, b
Ensure: u
  L = CHOLESKY(-(1/h^2) A)
  s_1=FORWARD-SUBSTITUTION(L,b)
  u=BACKWARD-SUBSTITUTION(L^T,s_1)
  return u
```

II-B. Métodos iterativos para resolver sistemas de ecuaciones

II-B1. Método de Jacobi: De acuerdo a [1], el método iterativo de Jacobi se obtiene resolviendo la i-ésima ecuación en $Ax = b$, siempre que los elementos en la diagonal de A sean distintos de 0. Lo anterior lo podemos escribir como

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

Lo cual nos genera la siguiente expresión para la actualización de la entrada x_i^t a partir del valor x_i^{t-1}.

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

El pseudocódigo para resolver sistemas de ecuaciones mediante el método iterativo de Jacobi es el siguiente

Algorithm 3 JACOBI-METHOD

```
Require: A,b,X0,tol,maxiter
Ensure: Una aproximación de x tal que ||Ax - b|| < tol
  i = 1
  while i ≤ maxiter do
    for j from 1 to n do
      x_i = (1/a_{i,i}) [ Σ_{j=1,j≠i}^{n} (-a_{i,j} X0_j) + b_i ]
    end for
    if ||x - X0|| < tol then
      return x
    end if
    i = i + 1
    X0_i = x_i
  end while
```

La convergencia del método de Jacobi se asegura cuando la matriz A cumple la siguiente propiedad

- A es estrictamente diagonal dominante es decir $|a_{i,i}| > \sum_{i \neq j} |a_{i,j}|$

II-B2. Método de Gauss-Seidel: El método de Gauss-Seidel es una modificación al método anteriormente descrito de Jacobi en donde la actualización del valor de x_i^t está dada de la siguiente forma

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

La convergencia del método de Gauss-Seidel se asegura si la matriz A, cumple alguna de las siguientes propiedades.

- A es definida positiva y simétrica
- A es estrictamente diagonal dominante es decir $|a_{i.i}| > \sum_{i \neq j} |a_{i,j}|$

El pseudocódigo para el algoritmo de Gauss-Seidel se muestra a continuación

Algorithm 4 GAUSS-SEIDEL-METHOD

Require: A,**b**,**X0**,tol,$maxiter$
Ensure: Una aproximación de **x** tal que $\|Ax - b\| < tol$

```
i = 1
while i ≤ maxiter do
    for j from 1 to n do
        x_i = 1/a_{i,i} [ −Σ_{j=1}^{i−1}(a_{i,j} x_j) − Σ_{j=i+1}^{n}(a_{i,j} X0_j) + b_i ]
    end for
    if ||x − X0|| < tol then
        return x
    end if
    i = i + 1
    X0_i = x_i
end while
```

III. Resultados

III-A. Prueba de Factorización de Cholesky

Consideremos el siguiente sistema de ecuaciones

$$\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$$

Utilizando el método de Cholesky para resolver dicho sistema podemos ver que $x_1 = 7$, $x_2 = 10$ y $x_3 = 8$, la matriz L, correspondiente a este problema, es la siguiente

$$\begin{bmatrix} 1{,}414210 & 0 & \\ -0{,}707107 & 1{,}22474 & 0 \\ 0 & -0{,}816497 & 1{,}1547 \end{bmatrix}$$

Notemos que la matriz L mantiene la estructura de matriz bandada como se describió en la sección de metodología.

III-B. Solución del problema elíptico unidimensional

Se utilizó el método de Cholesky para resolver el problema elíptico unidimensional descrito en la sección anterior con 10,50,100,200 nodos. El resultado de las soluciones se muestra en la Figura 1

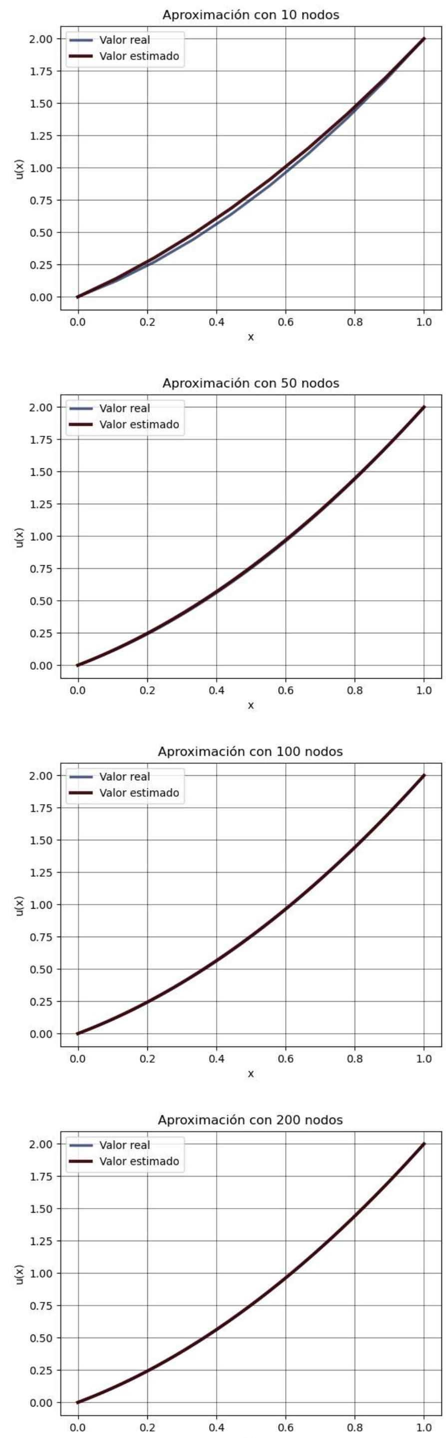

Figura 1. Aproximación de la solución del problema elíptico unidimensional con diferente número de nodos

Podemos notar que a partir de 50 nodos la curva de la solución estimada contra la solución real son casi indistinguibles.

III-C. Solución del problema iterativo unidimensional usando métodos iterativos

De acuerdo a las condiciones enunciadas en la sección de metodología, notemos que la matriz A resultante del planteamiento del problema elíptico unidimensional no es estrictamente diagonal dominante, es solamente diagonal dominante, por lo que no asegura la convergencia al menos para el método de Jacobi ni Gauss-Seidel. Los resultados de los algoritmos sobre este problema se muestra en el cuadro I. La comparación entre los tiempos de ejecución de los algoritmos se encuentran en el cuadro II.

Comparación de error de los diferentes métodos			
N	Cholesky $\frac{\|u-u_{app}\|}{\|u\|}$	Jacobi $\frac{\|u-u_{app}\|}{\|u\|}$	Gauss-Seidel $\frac{\|u-u_{app}\|}{\|u\|}$
10	0.196077	0.196077	0.196077
100	0.0172666	0.429291	0.260102
1000	0.00170614	0.933038	0.907192
10000	**0.001704**	0.990508	0.99584

Cuadro I

CUADRO COMPARATIVO ENTRE LOS DIFERENTES ALGORITMOS PARA ENCONTRAR LA SOLUCIÓN DEL PROBLEMA ELÍPTICO UNIDIMENSIONAL

Comparación de tiempo de los diferentes métodos			
N	Cholesky (segs)	Jacobi (segs)	Gauss-Seidel (segs)
10	1.0442e-05	0.000518612	0.000669626
100	0.00260677	0.144191	0.136149
1000	1.80231	11.2217	10.8268
10000	**1876.43**	227.75	227.59

Cuadro II

CUADRO COMPARATIVO ENTRE LOS DIFERENTES ALGORITMOS PARA ENCONTRAR LA SOLUCIÓN DEL PROBLEMA ELÍPTICO UNIDIMENSIONAL EN CUANTO A TIEMPO DE EJECUCIÓN

Podemos observar que en el caso del método analítico, el aumentar el número de nodos ayuda a obtener una mejor aproximación de la solución a dicho problema, sin embargo, en el caso de los métodos iterativos podemos observar que a medida que aumentamos el número de nodos el error relativo crece considerablemente. Lo anterior puede deberse a que la matriz A no es estrictamente diagonal dominante por lo que entonces depende de la condición inicial $\mathbf{x}_0$ para poder converger. En este caso usamos como $\mathbf{x}_0$ un vector que contiene solamente valores de 0.

IV. CONCLUSIONES

Con base en los resultados obtenidos en el presente trabajo, podemos concluir que el método de Cholesky es un método analítico que tiene un mejor desempeño en cuanto a tiempo de ejecución y error relativo en relación con el método anteriormente estudiados. Así mismo podemos concluir que los métodos iterativos son útiles para matrices grandes, ya que para matrices pequeñas son más tardados que algún método analítico. También podemos concluir que dichos métodos son sensibles a condiciones iniciales si las condiciones de convergencia de la matriz A no se satisfacen completamente. Finalmente, vemos que al aumentar el número de nodos en una matriz, el error relativo con los métodos iterativos crece cuando la convergencia no está asegurada por lo que la condición inicial $\mathbf{x}_0$ cobra mayor relevancia

Referencias

[1] Richard L Burden, J Douglas Faires, and Annette M Burden. *Numerical analysis*. Cengage learning, 2015.