Métodos Numéricos para Ingeniería
Factorización de Crout para sistemas lineales tridiagonales

Introducción

Para resolver el siguiente sistema lineal $n \times n$, donde se supone que tiene una solución única:

$$\begin{array}{rccccccccccc} E_1: & a_{11}x_1 & + & a_{12}x_2 & & & & & & & = & b_1 \\ E_2: & a_{21}x_1 & + & a_{22}x_2 & + & a_{23}x_3 & & & & & = & b_2 \\ \vdots & & & & & \vdots & & & & & \vdots & \\ E_{n-1}: & & & & & a_{n-1,n-2}x_{n-2} & + & a_{n-1,n-1}x_{n-1} & + & a_{n-1,n}x_n & = & b_{n-1} \\ E_n: & & & & & & & a_{n,n-1}x_{n-1} & + & a_{nn}x_n & = & b_n \end{array} \quad (0.1)$$

Para $x = (x_1, x_2, \ldots, x_n)^t$, dado $A = (a_{ij})$ con $i, j = 1, 2, \ldots, n$ y $b = (b_i)$ para $i = 1, 2, \ldots, n$.

Escribiendo el sistema (0.1) en forma matricial, se tiene:

$$Ax = b.$$

El algoritmo que resuelve el sistema de ecuaciones se muestra en la siguiente sección.

Factorización de Crout para sistemas lineales tridiagonales

ENTRADA la dimensión n; las entradas de la matriz ampliada A.

SALIDA la solución $x_1, \ldots, x_n$.

(Los pasos 1-3 configuran y resuelven $Lz = b$.)

Paso 1 Determine $l_{11} = a_{11}$;
$u_{12} = a_{12}/l_{11}$;
$z_1 = a_{1,n+1}/l_{11}$.

Paso 2 Para $i = 2, \ldots, n-1$ determine $l_{i,i-1} = a_{i,i-1}$; (i-ésima fila de L.)
$l_{ii} = a_{ii} - l_{i,i-1}u_{i-1,i}$;
$u_{i,i+1} = a_{i,i+1}/l_{ii}$; ((i + 1)-ésima columna de U.)
$z_i = \left(a_{i,n+1} - l_{i,i-1}z_{i-1}\right)/l_{ii}$.

Paso 3 Determine $l_{n,n-1} = a_{n,n-1}$; (enésima fila de L.)
$l_{nn} = a_{nn} - l_{n,n-1}u_{n-1,n}$;
$z_n = \left(a_{n,n+1} - l_{n,n-1}z_{n-1}\right)/l_{nn}$.

(Pasos 4 y 5 resuelven $Ux = z$.)

Paso 4 Determine $x_n = z_n$.

Paso 5 Para $i = n-1, \ldots, 1$ determine $x_i = z_i - u_{i,i+1}x_{i+1}$.

Paso 6 **SALIDA** $(x_1, \ldots, x_n)$;
PARE.

Problema

Considere uns sistema matricial del la forma $A(\alpha)\,x = b(\alpha)$, donde

$$A(\alpha)_{ij} = \begin{cases} \sin((i+j)\alpha)\cos((i+j)\alpha) & \text{si } i = j, \\ -\sin(\alpha) & \text{si } j = i-1 \text{ y } i \text{ impar}, \\ \cos(\alpha) & \text{si } j = i-1 \text{ y } i \text{ par}, \\ 1 & \text{si } j = i+1, \\ 0 & \text{en cualquier otro caso.} \end{cases}$$

con $i, j \in \{1, 2, \ldots, n\}$ y el vector $b(\alpha)$ es el siguiente,

$$b(\alpha)_i = \begin{cases} \sin(i\alpha) & \text{si } i \text{ es par}, \\ \cos(i\alpha) & \text{si } i \text{ es impar.} \end{cases}$$

con $i \in \{1, 2, \ldots, n\}$

Actividades

Objetivo: Programar una función en Python que construya la matriz $A(\alpha)$ y el vector $b(\alpha)$ en función de los parámetros de entrada α y n. Además, implementar el algoritmo de factorización de Crout para sistemas lineales tridiagonales conforme a las especificaciones del documento proporcionado.

1. Implemente en Python el algoritmo de factorización de Crout para sistemas lineales tridiagonales siguiendo estrictamente las instrucciones del documento. No se deben utilizar variantes alternativas del algoritmo.

2. Desarrolle una función en Python que reciba como parámetros de entrada los valores de α y n, y que construya la matriz $A(\alpha)$ y el vector $b(\alpha)$.

3. Utilizando el algoritmo de factorización de Crout para sistemas lineales tridiagonales, resuelva el sistema lineal $A(\alpha)\,x = b(\alpha)$ para los valores $\alpha = 1$ y $n = 10$. Calcule el error relativo entre $A(\alpha)\,x$ y $b(\alpha)$ para la solución encontrada en el paso anterior. ¿Podemos afirmar que el algoritmo proporciona una buena aproximación para los valores de n y α? Justifique su respuesta. Incluya esta justificación como comentario en la implementación del algoritmo.