Resolución de un Sistema con Intercambiadores de Calor en Serie (Sistemas Lineales)

Métodos Numéricos

Docente: Dra. Alma Xóchitl González Morales

Universidad de Guanajuato – Campus León
División de Ciencias e Ingenierías
León, Guanajuato, México

Noviembre 2025

Resolución de un Sistema con Intercambiadores de Calor en Serie (Sistemas Lineales)

Oliva-Villar David Isaac

División de Ciencias e Ingenierías Campus León, Universidad de Guanajuato

Loma del Bosque 103, 37150 León Guanajuato, México

Licenciatura en Ingeniería Química Sustentable

(Fecha: 16 de Noviembre de 2025)

Resumen

El presente trabajo, presenta el análisis de un sistema con cinco intercambiadores de calor conectados en serie, modelado mediante balances de energía en estado estacionario. El sistema lineal resultante se resolvió utilizando tres métodos numéricos: Gauss-Jordan, Gauss-Seidel y factorización LU, con el objetivo de comparar la precisión, convergencia y eficiencia de cada método aplicados a esta clase de casos presentes en la ingeniería química. Los resultados muestran la evolución de las temperaturas a lo largo de los intercambiadores y permiten evaluar la sensibilidad del sistema frente a variaciones de parámetros como el flujo másico y el coeficiente de transferencia de calor. El análisis presentado aquí evidencia la importancia de seleccionar métodos numéricos adecuados para modelar empleando sistemas lineales en aplicaciones de ingeniería química.

1 Planteamiento del problema

Como ha sido comentado al inicio del documento, el objetivo de este estudio es el analizar un sistema de cinco intercambiadores de calor conectados en serie, por donde circula un fluido caliente que transfiere energía a un fluido frío en estado estacionario, es decir, este flujo no varía con el tiempo. Cada intercambiador se modela mediante un balance de energía, considerando que los flujos másicos son constantes, las capacidades caloríficas no varían con la temperatura, y que la transferencia de calor se realiza de manera eficiente con un coeficiente global U constante y un área de intercambio A conocida para cada unidad.

1.1. Balance de energía en cada intercambiador

Para el intercambiador i, la variación de energía del fluido caliente se puede expresar mediante:

$$\dot{m}c_p(T_{i-1} - T_i) + UA(T_i - T_c) = 0, \quad i = 1, 2, \ldots, 5 \tag{1}$$

donde:

- $\dot{m}$ es el flujo másico del fluido caliente [kg/s],
- c_p es la capacidad calorífica del fluido caliente [J/(kg · K)],
- T_{i-1} y T_i son las temperaturas de entrada y salida del intercambiador i [K],
- T_c es la temperatura del fluido frío [K],
- U es el coeficiente global de transferencia de calor [W/(m^2 · K)],
- A es el área de intercambio del intercambiador i [m^2].

Reordenando la ecuación, se obtiene una forma lineal:

$$-(\dot{m}c_p + UA)T_i + \dot{m}c_pT_{i-1} = -UAT_c \tag{2}$$

Esta expresión genera un sistema lineal de cinco ecuaciones con cinco incógnitas T_1, T_2, T_3, T_4, T_5, donde T_0 es la temperatura conocida de entrada del fluido caliente.

1.2. Matriz del sistema lineal

El sistema lineal se puede escribir en forma matricial como:

$$\begin{bmatrix} -(\dot{m}c_p + UA) & 0 & 0 & 0 & 0 \\ \dot{m}c_p & -(\dot{m}c_p + UA) & 0 & 0 & 0 \\ 0 & \dot{m}c_p & -(\dot{m}c_p + UA) & 0 & 0 \\ 0 & 0 & \dot{m}c_p & -(\dot{m}c_p + UA) & 0 \\ 0 & 0 & 0 & \dot{m}c_p & -(\dot{m}c_p + UA) \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} -UAT_c + \dot{m}c_pT_0 \\ -UAT_c \\ -UAT_c \\ -UAT_c \\ -UAT_c \end{bmatrix} \tag{3}$$

Esta matriz tridiagonal refleja la dependencia de cada intercambiador con su entrada y salida, y permite resolver el sistema mediante métodos numéricos directos e iterativos como Gauss-Jordan, Gauss-Seidel y factorización LU. La elección del método depende del tamaño del sistema y de la estructura de la matriz, siendo los métodos iterativos particularmente eficientes para matrices con estructura tridiagonal.

1.3. Resolución manual del sistema

Para realizar el análisis manual, se consideraron valores representativos de un sistema típico de intercambiadores de calor en serie: un flujo másico $\dot{m} = 1$ kg/s, capacidad calorífica $c_p = 4000$ J/(kg · K), coeficiente global de transferencia de calor $U = 500$ W/(m^2 · K), área de intercambio $A = 10$ m^2, temperatura de entrada del fluido caliente $T_0 = 600$ K y temperatura del fluido frío $T_c = 300$ K. Estos valores permiten observar de forma clara cómo disminuye la temperatura del fluido caliente a lo largo de los intercambiadores, sin complicar demasiado los cálculos.

Aplicando el balance de energía para cada intercambiador i en régimen estacionario:

$$\dot{m}c_p(T_{i-1} - T_i) + UA(T_i - T_c) = 0 \tag{4}$$

y reordenando para expresar T_i como incógnita:

$$-(\dot{m}c_p + UA)T_i + \dot{m}c_p T_{i-1} = -UAT_c \tag{5}$$

Se obtienen los coeficientes numéricos:

$$\dot{m}c_p = 4000, \quad UA = 5000, \quad \dot{m}c_p + UA = 9000$$

Por lo tanto, el sistema de cinco ecuaciones lineales queda:

$$\begin{aligned}
-9000T_1 + 4000 \cdot 600 &= -5000 \cdot 300 \\
-9000T_2 + 4000T_1 &= -5000 \cdot 300 \\
-9000T_3 + 4000T_2 &= -5000 \cdot 300 \\
-9000T_4 + 4000T_3 &= -5000 \cdot 300 \\
-9000T_5 + 4000T_4 &= -5000 \cdot 300
\end{aligned}$$

Resolviendo paso a paso:

$$\begin{aligned}
T_1 &= \frac{-2{,}400{,}000 - 1{,}500{,}000}{-9000} = 433{,}33 \text{ K} \\
T_2 &= \frac{-1{,}500{,}000 + 4000 \cdot 433{,}33}{9000} \approx 359{,}26 \text{ K} \\
T_3 &= \frac{-1{,}500{,}000 + 4000 \cdot 359{,}26}{9000} \approx 326{,}34 \text{ K} \\
T_4 &= \frac{-1{,}500{,}000 + 4000 \cdot 326{,}34}{9000} \approx 311{,}71 \text{ K} \\
T_5 &= \frac{-1{,}500{,}000 + 4000 \cdot 311{,}71}{9000} \approx 305{,}20 \text{ K}
\end{aligned}$$

Estos cálculos muestran cómo la temperatura del fluido caliente disminuye progresivamente a lo largo de los intercambiadores, acercándose a la temperatura del fluido frío, tal como se espera físicamente en un sistema en serie.

Intercambiador	**Temperatura salida T_i [K]**
1	433.33
2	359.26
3	326.34
4	311.71
5	305.20

Cuadro 1: Temperaturas de salida de los cinco intercambiadores de calor en serie.

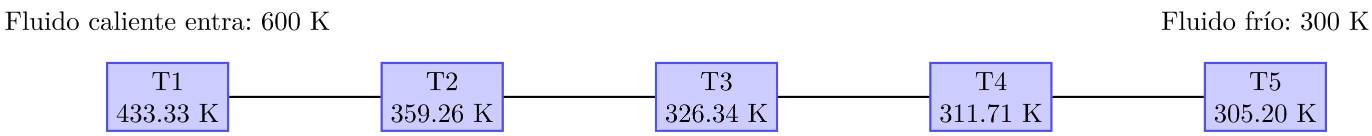
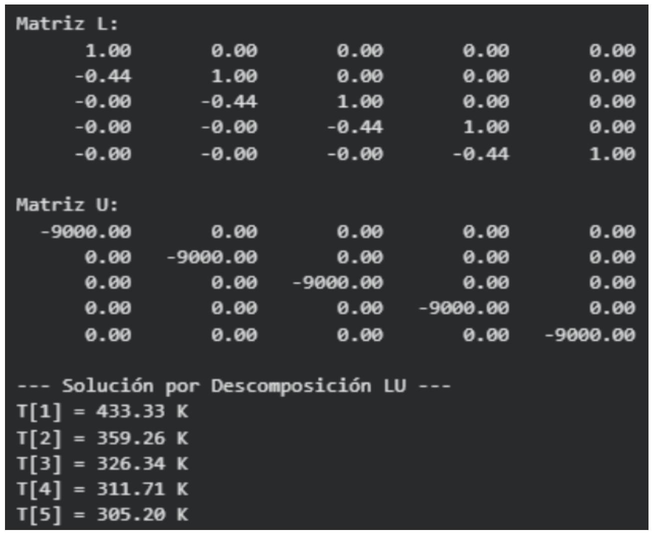


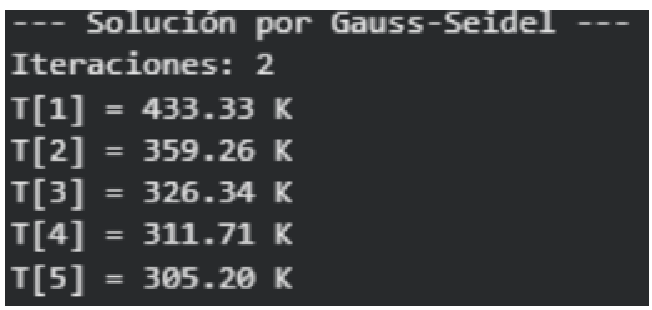
Figura 1: Diagrama en serie de cinco intercambiadores de calor con temperaturas de salida calculadas.

2 Resultados



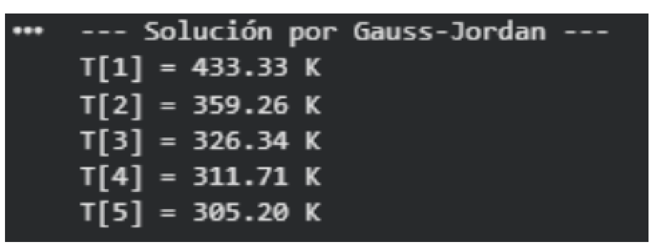
```
Matriz L:
      1.00      0.00      0.00      0.00      0.00
     -0.44      1.00      0.00      0.00      0.00
     -0.00     -0.44      1.00      0.00      0.00
     -0.00     -0.00     -0.44      1.00      0.00
     -0.00     -0.00     -0.00     -0.44      1.00

Matriz U:
  -9000.00      0.00      0.00      0.00      0.00
      0.00  -9000.00      0.00      0.00      0.00
      0.00      0.00  -9000.00      0.00      0.00
      0.00      0.00      0.00  -9000.00      0.00
      0.00      0.00      0.00      0.00  -9000.00

--- Solución por Descomposición LU ---
T[1] = 433.33 K
T[2] = 359.26 K
T[3] = 326.34 K
T[4] = 311.71 K
T[5] = 305.20 K
```

Imagen 1: Resultados obtenidos con el método de factorización LU (Captura de pantalla)



```
--- Solución por Gauss-Seidel ---
Iteraciones: 2
T[1] = 433.33 K
T[2] = 359.26 K
T[3] = 326.34 K
T[4] = 311.71 K
T[5] = 305.20 K
```

Imagen 2: Resultados obtenidos con el método de Gauss-Seidel (Captura de pantalla)



```
--- Solución por Gauss-Jordan ---
T[1] = 433.33 K
T[2] = 359.26 K
T[3] = 326.34 K
T[4] = 311.71 K
T[5] = 305.20 K
```

Imagen 3: Resultados obtenidos con el método de Gauss-Jordan (Captura de pantalla)

3 Conclusiones

Los tres métodos numéricos aplicados (LU, Gauss-Seidel y Gauss-Jordan) dieron resultados que coinciden entre ellos para las temperaturas de salida de los cinco intercambiadores de calor en serie, lo que confirma la consistencia y estabilidad del sistema lineal que se modeló para este trabajo. Las temperaturas disminuyen progresivamente conforme va pasando de un intercambiador a otro, acercándose a la temperatura del fluido frío, lo cuál tiene sentido físicamente hablando, teniendo en cuenta el comportamiento esperado de un sistema en serie. Los métodos directos (LU y Gauss-Jordan) ofrecen soluciones exactas en un número finito de pasos, mientras que el método iterativo (Gauss-Seidel) convergió muy rápidamente, en solo dos iteraciones, mostrando eficiencia en su aplicación cada método aplicado, en este particular caso, todos dan resultados iguales, complementando la solución al sistema entre ellos. Esta coincidencia de resultados permite concluir que la modelación numérica es confiable para analizar y predecir la respuesta de intercambiadores de calor ante variaciones de parámetros como flujo másico o coeficiente de transferencia.

Referencias

[1] D. Q. Kern, *Process Heat Transfer*, McGraw-Hill, 1950. Referencia clásica en diseño de intercambiadores de calor, balances de energía, coeficientes de transferencia y aplicaciones industriales.

[2] Y. A. Çengel y A. J. Ghajar, *Heat and Mass Transfer: Fundamentals and Applications*, 5th edition, McGraw-Hill, 2015. Libro de referencia moderna para transferencia de calor y análisis de intercambiadores.

Códigos empleados para el análisis (lenguaje C)

Listing 1: Factorización LU para 5 intercambiadores de calor en serie

```c
#include <stdio.h>

#define N 5

void imprimirMatriz(const char* nombre, double M[N][N]) {
    printf("\n%s:\n", nombre);
    for (int i = 0; i < N; i++) {
        for (int j = 0; j < N; j++) {
            printf("%10.2f ", M[i][j]);
        }
        printf("\n");
    }
}

void descomposicion_LU() {
    double A[N][N] = {
        {-9000,     0,     0,     0,     0},
        { 4000, -9000,     0,     0,     0},
        {    0,  4000, -9000,     0,     0},
        {    0,     0,  4000, -9000,     0},
        {    0,     0,     0,  4000, -9000}
    };

    double b[N] = {
        -5000*300 - 4000*600,
        -5000*300,
        -5000*300,
        -5000*300,
        -5000*300
    };

    double L[N][N] = {0};
    double U[N][N] = {0};
    double y[N], x[N];

    for (int i = 0; i < N; i++) {
        L[i][i] = 1;

        // Calculo de U
        for (int j = i; j < N; j++) {
            double sum = 0.0;
            for (int k = 0; k < i; k++) sum += L[i][k] * U[k][j];
            U[i][j] = A[i][j] - sum;
        }

        // Calculo de L
        for (int j = i + 1; j < N; j++) {
            double sum = 0.0;
            for (int k = 0; k < i; k++) sum += L[j][k] * U[k][i];
            L[j][i] = (A[j][i] - sum) / U[i][i];
        }
    }
```

```
    imprimirMatriz("Matriz␣L", L);
    imprimirMatriz("Matriz␣U", U);

    // Sustituci n hacia adelante Ly = b
    for (int i = 0; i < N; i++) {
        double sum = 0.0;
        for (int j = 0; j < i; j++) sum += L[i][j] * y[j];
        y[i] = b[i] - sum;
    }

    // Sustituci n hacia atr s Ux = y
    for (int i = N - 1; i >= 0; i--) {
        double sum = 0.0;
        for (int j = i + 1; j < N; j++) sum += U[i][j] * x[j];
        x[i] = (y[i] - sum) / U[i][i];
    }

    printf("\n---␣Soluci n␣por␣Descomposici n␣LU␣---\n");
    for (int i = 0; i < N; i++) {
        printf("T[%d]␣=␣%.2f␣K\n", i+1, x[i]);
    }
}

int main() {
    descomposicion_LU();
    return 0;
}
```

Listing 2: Gauss-Seidel para 5 intercambiadores de calor en serie

```
#include <stdio.h>
#include <math.h>

#define N 5

void gauss_seidel() {
    // Matriz A (coeficientes de T1..T5)
    double A[N][N] = {
        {-9000,     0,     0,     0,     0},
        { 4000, -9000,     0,     0,     0},
        {    0,  4000, -9000,     0,     0},
        {    0,     0,  4000, -9000,     0},
        {    0,     0,     0,  4000, -9000}
    };

    // Vector b (lado derecho del sistema)
    double b[N] = {
        -5000*300 - 4000*600,  // Primera ecuaci n con T0
        -5000*300,
        -5000*300,
        -5000*300,
        -5000*300
    };

    double x[N] = {0, 0, 0, 0, 0};  // Aproximaci n inicial
    double x_old[N];
    double tol = 1e-4;
```

```
    int max_iter = 100;
    int iter = 0;

    while (iter < max_iter) {
        for (int i = 0; i < N; i++) x_old[i] = x[i];

        for (int i = 0; i < N; i++) {
            double sum = 0.0;
            for (int j = 0; j < N; j++) {
                if (i != j) sum += A[i][j] * x[j];
            }
            x[i] = (b[i] - sum) / A[i][i];
        }

        // Calcular error m ximo
        double error_max = 0.0;
        for (int i = 0; i < N; i++) {
            double diff = fabs(x[i] - x_old[i]);
            if (diff > error_max) error_max = diff;
        }

        iter++;
        if (error_max < tol) break;
    }

    printf("---␣Soluci n␣por␣Gauss-Seidel␣---\n");
    printf("Iteraciones:␣%d\n", iter);
    for (int i = 0; i < N; i++) {
        printf("T[%d]␣=␣%.2f␣K\n", i+1, x[i]);
    }
}

int main() {
    gauss_seidel();
    return 0;
}
```

Listing 3: Gauss-Jordan para 5 intercambiadores de calor en serie

```
#include<stdio.h>
#include<math.h>

#define N 5   // N mero de inc gnitas

int main() {
    // Matriz aumentada [A|b]
    double A[N][N+1] = {
        {-9000,     0,     0,     0,     0,  -5000*300 - 4000*600},  // T1
        { 4000, -9000,     0,     0,     0,  -5000*300},              // T2
        {    0,  4000, -9000,     0,     0,  -5000*300},             // T3
        {    0,     0,  4000, -9000,     0,  -5000*300},            // T4
        {    0,     0,     0,  4000, -9000,  -5000*300}             // T5
    };

    for (int paso = 0; paso < N; paso++) {
        // Intercambio de filas para pivotaje parcial
        int max_fila = paso;
```

```
        for (int i = paso + 1; i < N; i++) {
            if (fabs(A[i][paso]) > fabs(A[max_fila][paso])) {
                max_fila = i;
            }
        }
        if (max_fila != paso) {
            for (int j = 0; j < N+1; j++) {
                double temp = A[paso][j];
                A[paso][j] = A[max_fila][j];
                A[max_fila][j] = temp;
            }
        }

        // Verificaci n de pivote
        if (fabs(A[paso][paso]) < 1e-10) {
            printf("Pivote cero en fila %d\n", paso+1);
            return 1;
        }

        // Normalizar fila del pivote
        double pivote = A[paso][paso];
        for (int j = paso; j < N+1; j++) {
            A[paso][j] /= pivote;
        }

        // Eliminar otras filas
        for (int i = 0; i < N; i++) {
            if (i != paso) {
                double factor = A[i][paso];
                for (int j = paso; j < N+1; j++) {
                    A[i][j] -= factor * A[paso][j];
                }
            }
        }
    }

    // Mostrar soluci n
    printf("--- Soluci n por Gauss-Jordan ---\n");
    for (int i = 0; i < N; i++) {
        printf("T[%d] = %.2f K\n", i+1, A[i][N]);
    }

    return 0;
}
```