
Título del Reporte

Nombre del primer autor* Nombre del segundo autor[†]

Fecha de entrega del reporte

1. RESUMEN

Sinopsis breve que describa el problema, los métodos numéricos utilizados, los resultados obtenidos y las conclusiones.

2. PLANTEAMIENTO DEL PROBLEMA

Presentación breve y precisa del problema a tratar, indicando los métodos numéricos utilizados para su resolución.

3. METODOLOGÍA

Esta sección debe contener como mínimo lo siguiente:

- descripción de los métodos numéricos utilizados y sus respectivas implementaciones computacionales;
- lista de viñetas con los nombres de los m-archivos realizados;
- presentación formal de los resultados teóricos¹ (lemas, proposiciones, corolarios, teoremas).

*Correo electrónico del primer autor

[†]Correo electrónico del segundo autor

¹En caso de no haber sido presentados en clases de teoría, por favor referenciar en la sección 6.

4. DISCUSIÓN DE LOS RESULTADOS

Presentación y análisis de los resultados computacionales. Se recomienda el uso de gráficos con ejes coordenados claramente identificados y leyendas para distinguir cada línea, utilizando un formato que garantice legibilidad. Utilice tablas con pocas filas y columnas para representar datos relevantes. Gráficos y tablas deben estar numerados y acompañados de una nota explicativa.

A modo ilustrativo, considere los siguientes ejemplos de gráfico y tabla:

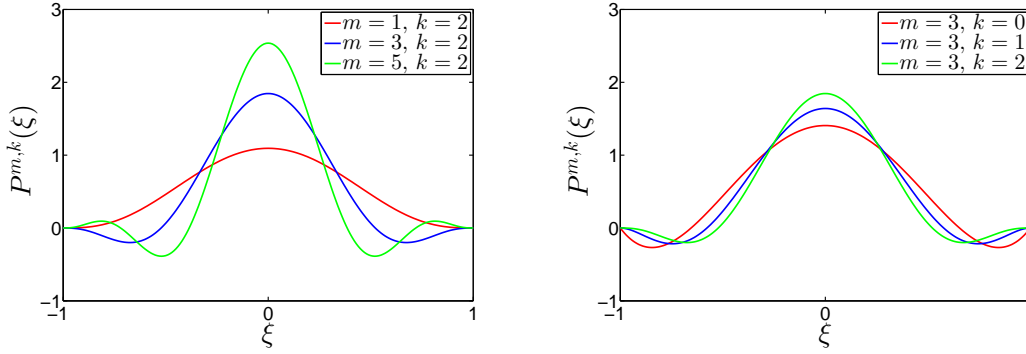


Figura 4.1: Polinomio $P^{m,k}$ para $m = 1, 3, 5$; $k = 2$ (izquierda) y $m = 3$; $k = 0, 1, 2$ (derecha).

N	30	60	120	240	480
ε	0.37	0.31	0.25	0.20	0.17
p	\times	2.67	3.01	3.08	2.73

Tabla 4.1: Número de nodos N , parámetro de escalamiento ε y orden de convergencia p en la norma $\|\cdot\|_{L_w^2}$ para la solución espectral de la ecuación de advección.

5. CONCLUSIONES

Explicación detallada con soporte teórico de los resultados obtenidos.

6. REFERENCIAS BIBLIOGRÁFICAS (OPCIONAL)

Lista de las referencias citadas.

APÉNDICE A LISTA DE PROGRAMAS COMPUTACIONALES

Lista de todos los programas utilizados para generar los resultados presentados en la sección 4. Los códigos deben estar detalladamente comentados, indicando

- nombre del programa;
- nombre del (los) autor(es);
- correo electrónico del (los) autor(es);
- fecha de elaboración del programa;
- descripción del programa; y
- especificación de los datos de entrada y salida del programa.

A modo ilustrativo, considere el siguiente ejemplo de presentación un m-archivo:

```
% -----  
% Nombre del programa:      jacobi.m  
% Autor(es):               Nombre(s) y apellido(s)  
% Email del (los) autor(es):  
% Fecha de elaboracion:     Mayo 11 de 2016  
  
% Breve descripcion del programa: Resuelve el sistema  $a * x = b$  utilizando  
% ----- el metodo iterativo de Jacobi.  
  
% Datos de entrada:  
% -----  
% a: Matriz cuadrada de coeficientes del sistema  
% b: Vector columna que representa el lado derecho del sistema  
% x: Vector columna que contiene el iterado inicial  
  
% Datos de salida:  
% -----  
% x      : Vector columna que aproxima la solucion del sistema  
% resi: Vector columna que contiene en cada coordenada la norma  
%        Euclidea del vector  $a * x - b$ .  
% iter: Numero de iteraciones realizadas.  
% -----  
  
function [ x , resi , iter] = jacobi( a , b , x )  
  
% Maximo de iteraciones  
% -----  
max_iter = 1000;  
  
% Tolerancia para el criterio de parada  
% -----  
epsi = 1e-8;  
  
% Dimension de x  
% -----  
n = length(x);  
  
% Inicio: Iteraciones
```

```

% -----
for k = 1 : max_iter
    % Almacenando el iterado k en y
    % -----
    y = x;
    % Inicio: Calculo del iterado k + 1
    % -----
    for i = 1 : n
        suma = b( i );
        diag = a( i , i );
        % Inicio: Verificando que a(i,i) no sea muy pequeño
        % -----
        if abs ( a ( i , i ) ) < eps
            iter = k;
            return
        end
        % Fin: Verificando que a(i,i) no sea muy pequeño
        % -----
        for j = 1 : n
            if j ~= i
                suma = suma - a(i,j) * y(j);
            end
        end
        x(i) = suma / diag;
    end
    % Fin: Calculo del iterado k + 1
    % -----

    % Vector residual
    % -----
    resi(k) = norm( b - a * x );

    % Criterio de parada
    % -----
    if norm( x - y ) < norm( x ) * epsi
        iter = k;
        return
    end
end

end
% Fin: Iteraciones
% -----

```