Solución Aproximada de Ecuaciones Diferenciales Parciales utilizando el Método de Elementos Finitos

Fredy Vides

2025

1 Introducción

Considerando la formulación abstracta de un problema de valor de frontera correspondiente a una ecuación diferencial parcial de tipo elíptico, la cual puede expresearse en forma:

$$A\ u = f, \tag{1}$$

para $u, f \in H^1(\Omega)$, y para $\Omega \subset \mathbb{R}^n$ compacto. Si se reformula el problema (1) de forma débil a través de la expresión:

$$a(u, v) = \hat{f}(v), \tag{2}$$

para cada $v \in H_0^1(\Omega)$, donde:

$$\begin{aligned} a(u, v) &:= \langle A\ u, v\rangle_{L^2(\Omega)}\,, \\ \hat{f}(v) &:= \langle f, v\rangle_{L^2(\Omega)}\,. \end{aligned}$$

Como consecuencia de la teoría de proyecciones ortogonales en espacios de Hilbert, es posible proyectar el problema (2) a una representación matricial inducida por una base $\{\phi_j\}_{j=1}^p$ de un cerrado finito dimensional $V_h(\Omega) \subset H^1(\Omega)$. Con base en las consideraciones previas, la formulación matricial de (2) toma la forma:

$$\mathcal{A}_h\ u_h = \hat{f}_h \tag{3}$$

donde

$$u_h = \begin{bmatrix} u_{h,1} \\ u_{h,2} \\ \vdots \\ u_{h,p} \end{bmatrix}$$

y las entradas de las matrices $\mathcal{A}_h, \hat{f}_h$ están determinadas por las expresiones:

$$\begin{aligned} \mathcal{A}_{h,i,j} &:= a(\phi_i, \phi_j), \\ \hat{f}_{h,i} &:= \hat{f}(\phi_i). \end{aligned}$$

En este contexto,

$$\tilde{u}_h := \sum_{j=1}^{p} u_{h,j}\ \phi_j$$

representa la proyección sobre $V_h(\Omega)$ de las solución débil de (1), correspondiente a la solución aproximada de (2).

2 Implementación Computacional

En esta sección consideraremos ejemplos prototípicos que permiten ilustrar la implementación computacional de los elementos presentados en la sección previa.

2.1 Ecuaciones dieferenciales en 1D

Consideremos PVFs de la forma:

$$\begin{cases} -\partial_x^2\ u = f, \\ u(a) = u(b) = 0 \end{cases} \tag{4}$$

Por propiedades elementales de la teoría de integración en $\mathbb{R}$, es posible identificar la formulación débil del PVF (4), la cual estará determinada por la expresión:

$$\int_a^b \partial_x u\ \partial_x v\ dx = \int_a^b f\ v\ dx \tag{5}$$

2.1.1 Cómputo de formas matriciales de referencia

Para encontrar la representación particular de la forma matricial (3) correspondiente al problema (5), iniciamos considerando una partición de $[a, b]$ en subintervalos $I_j = [a_j, b_j]$, con $0 \leq j \leq N-1$ para $N > 0$ denotando el número de elementos. Donde tenemos que $a_0 = a$, $b_{N-1} = b$, $b_j = a_{j+1}$ para $0 \leq j \leq N-2$, y $b_j - a_j = h > 0$, para $h = (b-a)/N$.

Para identificar una base $\{\phi_{h,j}\}$ del cerrado $V_h([a, b])$ correspondiente, consideraremos primero la formulación local correspondiente al intervalo de referencia $I = [-1, 1]$, en el cual, para un orden entero $p \geq 1$, consideraremos la base local $\{\Phi_0, \Phi_1, \dots, \Phi_p\}$ determinada por la expresión:

$$\Phi_j(x) := \frac{\prod_{k=0,k\neq j}^{p}(x - x_k)}{\prod_{k=0,k\neq j}^{p}(x_j - x_k)},\ \ 0 \leq j \leq p \tag{6}$$

donde $x_j = -1 + 2j/p$, $0 \leq j \leq p$. Con base en la ecuación (6), es posible apreciar que cada elemento base de referencia Φ_j está en correspondencia con el elemento x_j de la partición uniforme correspondiente de [-1,1], de tamaño $p + 1$.

El cálculo de las bases de referencia puede realizarse utilizando el siguiente código Octave:

```
function P = interpol(n)
  P = zeros(n+1);
  r = -1:2/n:1;
```

```
  for k = 1:(n+1)
    P(k,:) = poly(r(r~=r(k)));
    P(k,:) = P(k,:)/polyval(P(k,:),r(k));
  endfor
```

Ahora la matriz de Gram S_h correspondiente a esta base, y al producto interior $\langle \cdot, \cdot \rangle_{L^2([-1,1])}$, cuyas entradas están definidas por la expresión:

$$S_{h,i,j} = \langle \Phi_{i+1}, \Phi_{j+1} \rangle_{L^2([-1,1])}$$

puede calcularse utilizando el siguiente programa.

```
function S = S_mat(P)
  n = size(P,1);
  E = eye(n);
  S = zeros(n);
  for k = 1:n
    p = @(x)polyval(P(k,:),x).^2;
    S(k,k) = integral(p,-1,1);
    for j = (k+1):n
      p = @(x)polyval(P(k,:),x).*polyval(P(j,:),x);
      S(k,j) = integral(p,-1,1);
      S(j,k) = S(k,j);
    endfor
  endfor
```

Finalmente, si se denota por A_h la discretización correspondiente a la forma bilineal $a(\cdot, \cdot)$ con respecto a la base de referencia, cuya representación matricial está determinada por la expresión:

$$A_h := D_h^\top S_h D_h$$

donde D_h representa la matriz de diferenciación corrspondiente a la partición de referencia de tamaño p de $[-1, 1]$. Puede calcularse utilizando el siguiente secuencia de comandos:

```
>> P = interpol(p);
>> Sh = S_mat(P);
>> Ah = Dh.'*Sh*Dh;
```

para $p > 0$ fijo pero arbitrario, y donde la matriz `Dh` puede generarse utilizando el siguiente programa.

```
function [D,x]=Dmat(x)
  x=x(:);
  N=length(x)-1;
  D=repmat(x,1,N+1);
  E=eye(N+1);
  D=D.'-D+E;
  p=diag(prod(D));
  s=diag(sum(1./D)-1);
  D=p*(1./D.')/p-E+s;
  endfunction
```

2.1.2 Cómputo de formas matriciales locales y globales

Utilizando las transformaciones:

$$\tau_j(x) := \frac{2x - (a_j + b_j)}{b_j - a_j} \tag{7}$$

las bases de referencia inducen bases locales en cada elemento correspondiente al intervalo I_j, las cuales estarán determinadas para cada $0 \leq k \leq N-1$, por las expresiones:

$$\phi_{k,j}(x) := \Phi_j \circ \tau_k(x), \ \ 0 \leq j \leq p \tag{8}$$

Es posible apreciar que para $0 \leq k \leq N-2$, $\phi_{k,p}$ y $\phi_{k+1,0}$ definen dos instancias de una misma función en la base $\{\phi_{h,j}\}$ del cerrado $V_h([a,b])$. Consecuentemente, con excepción de estos casos, el resto de las funciones en la base $\{\phi_{h,j}\}$ del cerrado $V_h([a,b])$, están reguladas por las condiciones de A-ortogonalidad generalizada por bloques:

$$a(\phi_{k,j}, \phi_{k',j'}) = 0, \tag{9}$$

siempre que no se cumple que: $k' = k+1$, $j = p$ y $j' = 0$, para $0 \leq k \leq N-2$. Las condiciones previamente expuestas a su vez implican que, si $\mathcal{J} = \{1, 2, \ldots, q\}$ denota el conjunto de índices de la base $\{\phi_{h,j}\}_{j \in \mathcal{J}}$, ordenados con el orden del diccionario con respecto a los índices locales k, j de las bases locales $\{\phi_{k,j}\}$, existen, un función sobre

$$J : \{0, \ldots, N-1\} \times \{0, \ldots, p\} \to \mathcal{J}$$

tal que:

$$J(k,p) = J(k+1,0),$$

$0 \leq k \leq N-2$. Al igual que una inyección:

$$\imath_J : \mathcal{J} \to \{0, \ldots, N-1\} \times \{0, \ldots, p\}$$

tal que:

$$\imath_J(J(k,p)) = (k,p) = \imath_J(J(k+1,0)),$$

$0 \leq k \leq N-2$. De manera que para cada $0 \leq k \leq N-2$:

$$\mathcal{A}_{h,J(k,j),J(k',j')} := \begin{cases} a(\phi_{h,J(k,p)}, \phi_{h,J(k,p)}) + a(\phi_{h,J(k+1,0)}, \phi_{h,J(k+1,0)}), & k' = k+1, j = p, j' = 0, \\ a(\phi_{h,J(k,j)}, \phi_{h,J(k',j')}), & \text{en otro caso.} \end{cases}$$

$$\hat{f}_{h,J(k,j)} := \begin{cases} \hat{f}(\phi_{h,J(k,p)}) + \hat{f}(\phi_{h,J(k+1,0)}), & k' = k+1, j = p, j' = 0, \\ \hat{f}(\phi_{h,J(k,j)}), & \text{en otro caso.} \end{cases}$$

Los elementos previamente expuestos permiten calcular las formas matriciales $\mathcal{A}_h, \hat{f}_h$ utilizando el programa Octave:

```
function [A,Sb,E,bf,x] = MEF1D(a,b,f,N,n)
  x = -1:2/n:1;
  P = interpol(n);
```

```
S = S_mat(P);
A = Dmat(x);
A = A.'*S*A;
h = (b-a)/N;
i = 1:N*(n+1);
j = [];for k = 1:N, j = [j,(1:(n+1))+n*(k-1)];endfor
E = sparse(j,i,1);
A = E*(2/h)*kron(speye(N),A)*E.';
Sb = (h/2)*E*kron(speye(N),S);
x = a:(b-a)/(n*N):b;
bf = f(x);
```

En este programa, el efecto de la sobreyección J es implementada a través de la matriz E_h representada en el programa como `E`.

La matriz global $\mathcal{A}_h$ se calcula con base en la matriz de referencia A_h, utilizando la matriz E_h y el siguiente procedimiento:

$$\mathcal{A}_h := \frac{2}{h} E_h (I_N \otimes A_h) E_h^\top$$

La matriz global de Gram $\mathcal{S}_h$ representada en el programa por `Sb`, puede calcularse a partir de la matriz de Gram de referencia, de acuerdo a la siguiente expresión.

$$\mathcal{S}_h := \frac{h}{2} E_h (I_N \otimes S_h)$$

El programa `MEF1D.m` puede utilizarse para resolver problemas de la forma (5) a través de la siguiente secuencia de comandos:

```
>> [A,Sb,E,bf,x] = MEF1D(0,1,@(x)pi^2*sin(pi*x),100,3);
>> u = A(2:(end-1),2:(end-1))\(Sb*E.'*bf.')(2:(end-1));
>> u = [0;u;0];
>> ue = @(x)sin(pi*x);
>> subplot(211),plot(x,ue(x),'k',x,u,'r--'),grid on,axis tight
>> subplot(212),semilogy(x,abs(ue(x)-u.')+eps,'r--'),grid on,axis tight
```

2.2 Ecuaciones dieferenciales en 2D

Consideremos PVFs de la forma:

$$\begin{cases} -\Delta\ u = f, \\ u(a_x, y) = u(b_x, y) = 0, \\ u(x, a_y) = u(x, b_y) = 0 \end{cases} \tag{10}$$

Por propiedades elementales de la teoría de integración en $\mathbb{R}^2$, es posible identificar la formulación débil del PVF (4), la cual estará determinada por la expresión:

$$\int_a^b \nabla u\ \cdot\ \nabla v\ dS = \int_a^b f\ v\ dS \tag{11}$$

2.2.1 Cómputo de formas matriciales locales y globales

Considerando la forma regular del dominio $\Omega = [a_x, b_x] \times [a_y, b_y]$, es posible considerar una representación de la base global $\{\phi_{h,j}\}$ del cerrado $V_h(\Omega)$, para la cual $\phi_{h,j}(x,y) = \phi^x_{h,j}(x)\phi^y_{h,j}(y)$, donde los elementos $\phi^x_{h,j}(x), \phi^y_{h,j}(y)$ pueden calcularse utilizando las técnicas en 1D implementadas en la sección anterior. Con estas simples consideraciones, la implementación computacional puede simplificarse elegantemente, sacando provecho de las propiedades elementales de la teoría de integración en varias variables, y para un dominio cuadrado $\Omega = [a.b]^2$, la solución débil aproximada del PVF correspondiente de la forma (10) puede calcularse utilizando el siguiente programa Octave.

```
function [A,Sb,bf,x,y] = MEF2D(a,b,f,N,n)
% Example:
% f = @(x,y)2*pi^2*sin(pi*x).*sin(pi*y);
% [A,Sb,bf,x,y]=MEF2D(0,1,f,20,3);
% u = A\bf;
% N = size(x);
% U =zeros(N);
% U(2:(N(1)-1),2:(N(2)-1)) = reshape (u,N(1)-2,N(2)-2);
% surf(x,y,abs(U-sin(pi*x).*sin(pi*y)))
    [A1,Sb1,E1,bf,x] = MEF1D(a,b,@(x)x,N,n);
    A1 = A1(2:(end-1),2:(end-1));
    [x,y] = meshgrid(x);
    f2 = f(x,y);
    f2 = Sb1*E1.'*f2*E1*Sb1.';
    bf = f2(2:(end-1),2:(end-1))(:);
    h = (b-a)/N;
    Sb = (Sb1*E1.')(2:(end-1),2:(end-1));
    A = kron(A1,Sb)+kron(Sb,A1);
```

3 Ejercicios

A. De dominios cuadrados a dominios rectangulares en 2D

Considere el PVF elíptico

$$\begin{cases} -\Delta u = f & \text{en } \Omega = [a_x, b_x] \times [a_y, b_y], \\ \quad u = 0 & \text{en } \partial\Omega, \end{cases}$$

con formulación débil

$$\int_\Omega \nabla u \cdot \nabla v \, dS = \int_\Omega f \, v \, dS, \quad \forall v \in H^1_0(\Omega).$$

1. **Factorización tensorial para rectángulos.** Muestre que, usando bases producto $\phi_i(x,y) = \phi^x_{i_x}(x)\, \phi^y_{i_y}(y)$, las matrices globales adoptan la forma

$$\mathcal{A}_h = K_x \otimes M_y + M_x \otimes K_y, \qquad \mathcal{M}_h = M_x \otimes M_y,$$

donde $K_\star$ y $M_\star$ son, respectivamente, las matrices de rigidez y masa 1D en la dirección $\star \in \{x, y\}$, ensambladas con las mallas (potencialmente distintas) N_x y N_y y grados locales n_x, n_y.

2. **Ensamble con rutinas 1D.** Usando `MEF1D` de la Sección 1D, construya

$$K_x = \mathcal{A}_h^{(x)}(\text{2:end-1, 2:end-1}), \quad M_x = \mathcal{S}_h^{(x)}(\text{2:end-1, 2:end-1}),$$

y análogamente K_y, M_y a partir de una llamada independiente a `MEF1D` sobre $[a_y, b_y]$. *Indicación:* reutilice las variables `A` y `Sb` devueltas por `MEF1D` tras el recorte de nodos de borde.

3. **Implementación.** Implemente una función `MEF2Drect.m` que reciba `(ax,bx,ay,by,f,Nx,Ny,nx,ny` y retorne `(A,M,b,x,y)`, donde:

$$A = K_x \otimes M_y + M_x \otimes K_y, \quad M = M_x \otimes M_y, \quad b = \text{vec}(F_{xy}),$$

con F_{xy} la quadratura tensorial usando las matrices de masa 1D:

$$F_{xy} \approx M_x \, F \, M_y^\top, \qquad F_{i_x, i_y} \approx f(x_{i_x}, y_{i_y}).$$

Resuelva $A\,u = b$ y reconstituya U en la retícula interior mediante `reshape`.

4. **Prueba numérica.** Use $\Omega = [0, 2] \times [0, 1]$ y

$$u(x, y) = \sin(\pi x) \sin(2\pi y), \qquad f(x, y) = (\pi^2 + 4\pi^2)\, u(x, y) = 5\pi^2\, u(x, y).$$

Verifique convergencia h-orden al refinar (N_x, N_y) con $n_x = n_y = 1, 2, 3$. Grafique el error $\|U_h - U_{\text{exacta}}\|_\infty$ y $\|U_h - U_{\text{exacta}}\|_{L^2}$ versus $h = \max\{h_x, h_y\}$ en escala log–log e identifique la pendiente observada.

B. Ecuación del calor en 1D con Dirichlet homogéneo

Considere el problema en $I = [a, b]$

$$\begin{cases} \partial_t u - \partial_x^2 u = g(x, t) & \text{en } I \times (0, T], \\ u(a, t) = u(b, t) = 0 & \text{en } (0, T], \\ u(x, 0) = u_0(x) & \text{en } I, \end{cases}$$

cuya forma débil (en cada t) es

$$(\partial_t u, v) + (\partial_x u, \partial_x v) = (g, v), \quad \forall v \in H_0^1(I),$$

donde $(\cdot, \cdot)$ denota el producto $L^2(I)$.

1. **Semidiscretización en el espacio.** Con la base FE 1D de la Sección correspondiente, derive el sistema ODE matricial

$$M\, \dot{\mathbf{u}}(t) + K\, \mathbf{u}(t) = \mathbf{f}(t), \qquad \mathbf{u}(0) = \mathbf{u}_0,$$

con $M = \mathcal{S}_h(\text{2:end-1, 2:end-1})$ y $K = \mathcal{A}_h(\text{2:end-1, 2:end-1})$.

2. **Integración temporal θ-método.** Para $\theta \in [0,1]$ (Euler hacia atrás θ=1, Crank–Nicolson θ=1/2) derive el esquema

$$\big(M + \theta\Delta t\, K\big)\, \mathbf{u}^{n+1} \;=\; \big(M - (1-\theta)\Delta t\, K\big)\, \mathbf{u}^{n} \;+\; \Delta t\big(\theta\, \mathbf{f}^{n+1} + (1-\theta)\, \mathbf{f}^{n}\big).$$

   Discuta estabilidad y orden temporal.

3. **Implementación.** Escriba una rutina `Calor1D_FE.m` que:
   (a) Ensamble M, K con `MEF1D`.
   (b) Aplique un lazo temporal con paso Δt hasta T, resolviendo el sistema lineal en cada paso.
   (c) Devuelva la solución en todos los tiempos en los nodos interiores y un vector `t`.

4. **Caso de prueba exacto.** En $I = [0,1]$, tome

$$u(x,t) = e^{-\pi^2 t}\sin(\pi x), \quad g \equiv 0, \quad u_0(x) = \sin(\pi x).$$

   Verifique convergencia en malla espacial (refinando N con $n = 1,2,3$) y en el paso de tiempo (refinando Δt) para $\theta = 1$ y $\theta = 1/2$. Reporte tasas empíricas de convergencia.

5. **Fuente dependiente del tiempo.** Repita con $g(x,t) = e^{-t}\sin(\pi x)$ y $u_0 \equiv 0$. Construya $\mathbf{f}^n \approx M\, g(\cdot, t_n)$ mediante evaluación nodal y producto con M.

6. **Conservación y disipación.** Muestre numéricamente que, con $g \equiv 0$, la norma de energía $E^n = \frac{1}{2}(\mathbf{u}^n)^\top M \mathbf{u}^n$ es monótonamente decreciente para $\theta \geq \frac{1}{2}$.

D. Entregables sugeridos

- Gráficas de solución y mapas de error para el caso 2D en rectángulos.
- Curvas log–log de error vs. h y vs. Δt (para $\theta = 1$ y $\theta = 1/2$).
- Tabla con tasas de convergencia empíricas en norma L^∞ y L^2.
- Discusión breve sobre estabilidad y disipación numérica en el calor 1D.

Referencias

- Weimin Han, Kendall E. Atkinson, *Theoretical Numerical Analysis: A Functional Analysis Framework*, Springer, 2009.
- Fredy Vides, *MNMC – Métodos Numéricos y Modelación Computacional*, lecturas de clase, 2021.