

Discretización matricial conservativa para la Ecuación de Poisson en mallas bloco estructuradas

Hector D. Ceniceros¹, Alexandre M. Roma², Catalina M. Rua^{3,*}



1. Department of Mathematics, University of California - Campus Santa Barbara.

2. Instituto de Matemática e Estatística, Universidade de São Paulo.

3. Departamento de Matemáticas y Estadística, Universidad de Nariño.

*Contacto: cmra03@gmail.com



INTRODUCCIÓN

Diversos problemas modelados matemáticamente por ecuaciones en derivadas parciales necesitan de mallas particularmente finas para capturar fenómenos importantes, muchas veces, restringidos a regiones relativamente pequeñas del dominio computacional. La discretización en mallas uniformes propaga esta necesidad de mallas finas para todo el dominio, haciendo la resolución numérica ineficiente. Como una alternativa, el uso de mallas con refinamiento localizado concentra el esfuerzo computacional en las regiones de mayor interés [2,4]. Construir una matriz que represente el conjunto de ecuaciones lineales resultantes de la discretización espacial del problema en estas mallas es muy útil, ya que al tenerse esta matriz disponible se tiene acceso a librerías gratuitas de álgebra lineal numérica que contienen métodos numéricos variados tanto seriales como paralelos para su solución, como por ejemplo la librería PETSc. Aunque usualmente se ha trabajado en discretizaciones sobre esta clase de mallas, usando métodos sin matrices (free-matrices), en los últimos años se está explorando con esta nueva opción matricial [6,7]. Al discretizar ecuaciones diferenciales usando diferencias finitas en mallas adaptativas estructuradas localmente, (AMR), para no cambiar el stencil de la discretización entre la intersección de mallas con diferente espaciamento, en las fronteras de cada sub-malla se usan células fantasmas con valores aproximados con interpolaciones de un orden que no afecte el de la discretización.

OBJETIVO

El objetivo de este trabajo es mostrar como obtener y solucionar un sistema lineal asociado con una discretización conservativa de la ecuación de Poisson

$$\nabla \cdot (\omega(x, y) \nabla \varphi(x, y)) = f(x, y) \quad (x, y) \in \Omega, \quad (1)$$

en una malla bloco-estructurada con refinamiento adaptativo usando condiciones de contorno de Dirichlet, de Neumann o periódicas [5], en donde $f(x, y)$ es un término forzante a la solución exacta, $\omega(x, y)$ es una función definida según la aplicación, Ω es el dominio y $\varphi(x, y)$ define la variable incógnita. Para obtener segundo orden en la discretización de (1) en una malla AMR, el operador divergente debe cumplir que para $\mathbf{f} = (f_x, f_y)$ que representa un flujo, donde $\mathbf{f} = \nabla \varphi$ y así $\nabla^2 \varphi = \nabla \cdot \mathbf{f}$, que

$$\int_{\Omega} \nabla \cdot \mathbf{f} = \int_{\partial \Omega} \mathbf{f} \cdot \mathbf{n}. \quad (2)$$

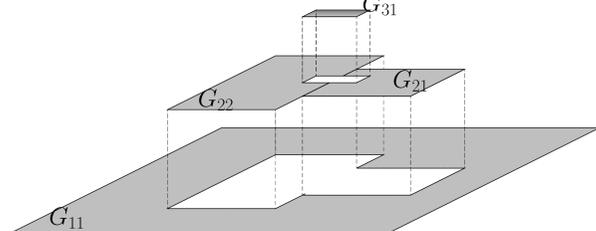


Figura 1: Malla estructurada por bloques.

METODOLOGÍA

- Malla adaptativa bloco estructurada y refinada localmente:** La malla está formada por la unión de bloques rectangulares de igual o diferente espaciamento. Los bloques en el mismo nivel deben tener intersección vacía y la unión de ellos forman un nivel de refinamiento. Bloques en niveles diferentes consecutivos deben respetar ciertas condiciones, ver Figura 1.
- Células fantasmas:** Ayudan en la comunicación entre los diferentes niveles de refinamiento y en las discretizaciones sin cambiar el stencil.
- Representación matricial:** A partir de la enumeración de las células visibles de la malla y con las interpolaciones de las células fantasmas se forma la matriz esparsa de la discretización.

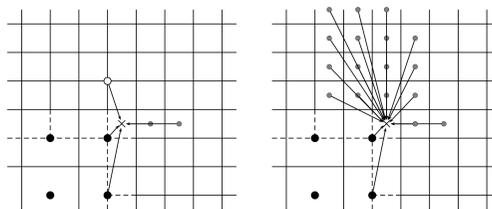


Figura 2: Estencil de célula fantasma T.

- Discretización:** Definimos las variables escalares y las vectoriales respectivamente en el centro de célula y en las faces, [4,5,9].

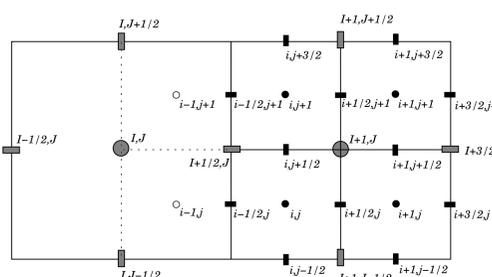


Figura 3: Interfaz Fina-Gruesa.

Para una célula interna (i, j) de la malla tenemos

$$(\nabla^2 \varphi)_{i,j} = (\nabla \cdot \mathbf{f})_{i,j}, \quad (3)$$

$$= \frac{f_{x_{i+\frac{1}{2},j}} - f_{x_{i-\frac{1}{2},j}} + f_{y_{i,j+\frac{1}{2}}} - f_{y_{i,j-\frac{1}{2}}}}{h}$$

$$f_{x_{i+\frac{1}{2},j}} = \frac{1}{h}(\varphi_{i+1,j} - \varphi_{i,j}), \quad f_{x_{i-\frac{1}{2},j}} = \frac{1}{h}(\varphi_{i,j} - \varphi_{i-1,j}),$$

$$f_{y_{i,j+\frac{1}{2}}} = \frac{1}{h}(\varphi_{i,j+1} - \varphi_{i,j}), \quad f_{y_{i,j-\frac{1}{2}}} = \frac{1}{h}(\varphi_{i,j} - \varphi_{i,j-1}),$$

pero en una interfaz fina-gruesa, tenemos

$$\nabla^2 \varphi_{I,J} = \frac{f_{x_{I+\frac{1}{2},J}}^{*l} - f_{x_{I-\frac{1}{2},J}}^l + f_{y_{I,J+\frac{1}{2}}}^l - f_{y_{I,J-\frac{1}{2}}}^l}{h_I}, \quad (4)$$

$$f_{x_{I+\frac{1}{2},J}}^{*l} = \frac{1}{2}(f_{x_{I-\frac{1}{2},J+1}}^{l+1} + f_{x_{I-\frac{1}{2},J}}^{l+1})$$

$$f_{x_{I-\frac{1}{2},J+1}}^{l+1} = \frac{\varphi_{I-1,J+1}^{l+1} - \varphi_{I,J+1}^{l+1}}{h_{I+1}}, \quad f_{x_{I-\frac{1}{2},J}}^{l+1} = \frac{\varphi_{I-1,J}^{l+1} - \varphi_{I,J}^{l+1}}{h_{I+1}}$$

- Solución del sistema lineal:** Usamos la biblioteca PETSc, "Portable, Extensible Toolkit for Scientific Computation", que incluye estructuras de datos para almacenar matrices y métodos numéricos para la solución de sistemas con preconditionadores en serial y en paralelo, [1,8].

RESULTADOS

Teste: $\varphi(x, y) = \sin(2\pi x) * \sin(2\pi y)$
 $\omega(x, y) = (1 + h) + h * \tanh(2\pi d(x - 0.5))$,
 C.C. : Neumann. $\Omega = [0, 1] \times [0, 1]$

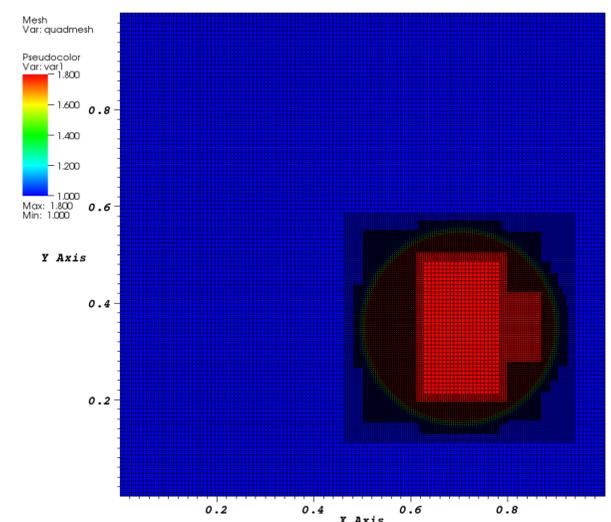


Figura 4: Malla compuesta G7L3, $h=0.4$. Condicionamiento da Matriz $K_2 = 1.0005$.

	L1	L2	L3	L4
Erro(Max)	3.465E-3	8.872E-4	2.235E-4	5.605E-5
iter	134	148	110	248

Tabla 1: Error y número de iter. PETSC: BICG-Stab, pre-condicionador ILU(0), $tol = 10^{-12}$.

La verificación numérica se realizó imponiendo la solución y en la Tabla 2 verificamos el orden 2 de convergencia.

Malla	Erro(Max)	R	Iter
$64 \times 64L3$	5.629E-05		284
$128 \times 128L3$	1.404E-05	4.01	664
$256 \times 256L3$	3.5074E-06	4.00	1836
$512 \times 512L3$	8.829E-07	3.97	5685

Tabla 2: Error, razón de convergencia y número de iter. con BICG-Stab-ILU(0). $h = 100$, $d = 2$ y $L2$.

Aunque el código que usamos para controlar la malla es serial, es posible usar paralelismo en la solución del sistema.

BIBLIOGRAFÍA

- [1] S. Balay; W. D. Gropp; L. C. McInnes and B. F. Smith, *Efficient Management of Parallelism in Object Oriented Numerical Software Libraries*. Birkhäuser Press, 163-202, 1997.
- [2] M. J. Berger and P. Colella, *Local adaptive mesh refinement for shock hydrodynamics*, J. Comput. Phys. **82**, 64-84, 1989.
- [3] M. J. Berger; S. M. Murman and M. J. Aftosmis, *Applications of space-filling curves to cartesian methods for CFD*, 2004.
- [4] H. D. Ceniceros; A. M. Roma; A. M. da Silveira-Neto and M. Villar; *A robust, fully adaptive hybrid level-set/front-tracking method for two-phase flows with an accurate surface tension computation*. Commun. Comput. Phys., **8**(1), 51-94, 2010.
- [5] D. Martin and K. Cartwright, *Solving poisson's equation using adaptive mesh refinement*, 1996.
- [6] A. Pletzer, B. Jamroz, R. Crockett and S. Sides. *Compact cell-centered discretization stencils at fine-coarse block structured grid interfaces*, Journal of Computational Physics, 2014.
- [7] C. Rua, *Simulação computacional adaptativa de escoamentos bifásicos viscoelásticos*, Tese de Doutorado, IME-USP, 2013.
- [8] Y. Saad, *Iterative Methods for Sparse Linear Systems, Second Edition*, 2 ed., Society for Industrial and Applied Mathematics, April 2003.
- [9] J. C. Strikwerda, *Finite difference schemes and partial differential equations*, second ed., Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2004.