Análisis Numérico de Algoritmos basados en tensores con aplicaciones a problemas de grandes dimensiones

Doctorado en Matemáticas por: María Mora Jiménez[1]

Directores: J. Alberto Conejero[1], Antonio Falcó[2]

[1] Instituto Universitario de Matemática Pura y Aplicada – Universitat Politècnica de València,
[2] ESI International Chair@CEU-UCH Departamento de Ciencias Físicas, Matemáticas y de la Computación – CEU UCH

Resumen

Muchos de los actuales problemas científicos pueden plantearse como un sistema de ecuaciones de la forma $A\mathbf{x} = \mathbf{b}$. Cuando A tiene la forma

$$A = \sum_{i=1}^{d} \mathrm{id}_{n_1} \otimes \cdots \otimes \mathrm{id}_{n_{i-1}} \otimes A_i \otimes \mathrm{id}_{n_{i+1}} \otimes \cdots \otimes \mathrm{id}_{n_d},$$

decimos que A es una matriz Laplaciana. Estas matrices aparecen, por ejemplo, cuando discretizamos algunas EDPs, como la ecuación de Poisson, y son un objeto de estudio interesante, ya que el algoritmos tensoriales, como la Descomposición Propia Generalizada (PGD), convergen rápidamente a la solución del sistema lineal asocidado.

Este hecho nos hizo plantearnos si cualquier matriz cuadrada $M \in \mathrm{GL}(\mathbb{R}^N)$ podría descomponerse de alguna forma de manera que el estudio del problema lineal asociado $M\mathbf{x} = \mathbf{b}$ fuese más sencillo. Para ello diseñamos el Algoritmo de Descomposición Laplaciana, principal resultado de la presente tesis.

Actualmente estamos analizando las mejoras que supone la descomposición laplaciana de una matriz, por ejemplo, en el estudio de algunas EDPs o en ciertos problemas de grafos.

Introducción & Motivación

Los sistemas lineales son ampliamente utilizados para abordar modelos computacionales en ciencias aplicadas. Existen numerosos mecanismos para hacer frente a este tipo de problema. Sin embargo, la mayoría de ellos pierden eficiencia a medida que aumenta el tamaño de las matrices o vectores involucrados. Este efecto se conoce como *el problema de la maldición de la dimensionalidad*. Para intentar solucionar este problema, proponemos utilizar algoritmos basados en tensores [5], ya que su uso reduce significativamente el número de operaciones que debemos realizar.

Una de las técnicas más populares entre los algoritmos basados en productos tensoriales [3] es la familia de Descomposición Propia Generalidada (PGD), basada en el llamado algoritmo Greedy de Rango Uno (GROU) [4]. Del estudio de este procedimiento para resolver sistemas lineales de alta dimensión, observamos que hay un tipo de matrices para las que el algoritmo funciona especialmente bien: las llamadas matrices de tipo Laplaciano, que tienen la forma

$$A = \sum_{i=1}^{d} \mathrm{id}_{n_1} \otimes \cdots \otimes \mathrm{id}_{n_{i-1}} \otimes A_i \otimes \mathrm{id}_{n_{i+1}} \otimes \cdots \otimes \mathrm{id}_{n_d},$$

y que pueden relacionarse fácilmente con el operador laplaciano clásico.

Este es el caso de la famosa ecuación de Poisson $-\Delta\varphi = \mathbf{f}$. Mediante el uso de aproximaciones derivadas y métodos de diferencias finitas, podemos escribir la ecuación de Poisson en forma discreta como un sistema lineal $A \cdot \varphi_{ijk} = -\mathbf{f}_{ijk}$, donde A es una matriz por bloques (detalles en [4]).

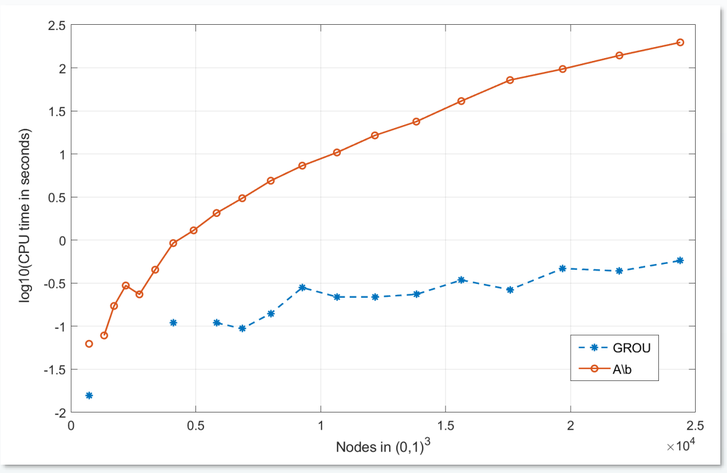


Fig 1: CPU-Time comparative

En la Fig. 1, comparamos el tiempo de CPU empleado para resolver este problema discreto de Poisson, con A escrita en forma laplaciana, cuando resolvemos el problema usando el algoritmo Greedy y el operador de Matlab $\mathbf{x} = A\backslash\mathbf{b}$, para diferente número de nodos en $(0,1)^3$.

La Descomposición Laplaciana

Supongamos que tenemos un sistema lineal de la forma $A\mathbf{x} = \mathbf{b}$, con A una matriz cuadrada en $\mathbb{R}^{N\times N}$. Queremos aproximar la matriz A mediante una matriz en forma laplaciana $L_A \in \mathbb{R}^{N\times N}$, y resolver el sistema lineal laplaciano asociado $L_A\mathbf{x} = \mathbf{b}$, para aproximar la solución del original sistema $\mathbf{x}^*$ por la solución obtenida del sistema laplaciano $\mathbf{x}_L^*$. Para ello, presentamos el siguiente **Teorema**.

Teorema: Sea $A \in \mathbb{R}^{N\times N}$, con $N = n_1 \cdots n_d$ tal que $\mathrm{tr}(A) = 0$, y sea

$$\Delta = \left\{ A \in \mathbb{R}^{N\times N} : A = \sum_{i=1}^{d} \mathrm{id}_{n_1} \otimes \cdots \otimes \mathrm{id}_{n_{i-1}} \otimes A_i \otimes \mathrm{id}_{n_{i+1}} \otimes \cdots \otimes \mathrm{id}_{n_d}, \mathrm{tr}(A_i) = 0 \ \forall i \right\}.$$

La proyección de A en Δ es

$$P_\Delta(A) = \sum_{i=1}^{d} \mathrm{id}_{[n_i]} \otimes X_i \doteq \sum_{i=1}^{d} \mathrm{id}_{n_1} \otimes \cdots \otimes \mathrm{id}_{n_{i-1}} \otimes X_i \otimes \mathrm{id}_{n_{i+1}} \otimes \cdots \otimes \mathrm{id}_{n_d},$$

donde las X_i se obtienen resolviendo los sucesivos problemas de minimización

$$\min_{X_i \in \mathbb{R}^{n_i \times n_i}} \left\| A - \sum_{k=1}^{i} \mathrm{id}_{[n_k]} \otimes X_k \right\|, \qquad i = 1, \ldots, d.$$

Además, si repetimos el esquema anterior, tenemos que

$$L_A = \lim_{k\to\infty} P_\Delta^{(k)}(A)$$

es la matriz laplaciana que mejor aproxima a la matriz A en Δ

La demostración de este resultado plantea un argumento iterativo similar al procedimiento del algoritmo Alternating Least Square (ALS), donde se reduce el error dado por la norma del residuo al actualizar los términos que forman parte de la descomposición laplaciana. Este procedimiento es descrito por el **Algoritmo 1 .**

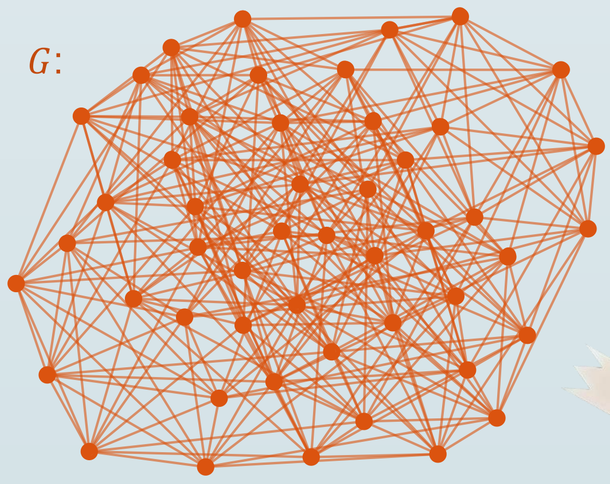
Algoritmo Principal

Algorithm 1 Laplacian Decomposition Algorithm

procedure Lap(A, iter_max, tol)
  iter $= 1$, Lap $= 0$
  while iter $<$ iter_max **do**
    $A \leftarrow A -$ Lap
    for $k = 1, \ldots, d$ **do**
      $P_k(A) = \mathrm{id}_{n_1} \otimes \cdots \otimes \mathrm{id}_{n_{k-1}} \otimes X_k \otimes \mathrm{id}_{n_{k+1}} \otimes \cdots \otimes \mathrm{id}_{n_d}$
      $X_k \leftarrow \min_{X_k} \left\| A - \sum_{i=1}^{k} P_i(A) \right\|$ **s.t.** $\mathrm{tr}(X_k) = 0$
      Lap $=$ Lap $+ P_k(A)$
    end for
    if $\|A -$ Lap$\| <$ tol **then goto** 14
    end if
    iter $=$ iter $+ 1$
  end while
  return Lap
end procedure

Ejemplo

La matriz de adyacencia del grafo G es una matriz de tipo Laplaciano en $\mathbb{R}^{49\times 49}$.



$$A = X_1 \otimes \mathrm{id}_{n_2} + \mathrm{id}_{n_1} \otimes X_2 \in \mathbb{R}^{49\times 49}$$

$$X_1, X_2 \in \mathbb{R}^{7\times 7}$$

Algoritmo:

X_1 -> $\min_{X_1} \|A - X_1 \otimes I_2\|_F$

X_2 -> $\min_{X_2} \|A - X_1 \otimes I_2 - I_1 \otimes X_2\|_F$

Conclusiones

En la tesis hemos enunciado un resultado que muestra cómo descomponer una matriz cuadrada como la suma de una matriz laplaciana y otra matriz linealmente independiente a ella. Además, hemos desarrollado el procedimiento para realizar esta descomposición en forma de algoritmo.

Actualmente estamos profundizando en el estudio de las mejoras que supone el empleo de estas matrices de tipo laplaciano en problemas dispersos, EDPs tras su discretización y ciertos problemas de grafos, como las redes de mundo pequeño.

Aunque la descomposición laplaciana de una matriz no tiene por qué ser única, la idea del trabajo presentado es poder cambiar el enfoque tradicional para resolver un sistema lineal $A\mathbf{x}=\mathbf{b}$, ayudándonos de técnicas que emplean descomposiciones tensoriales, ya que son más eficientes computacionalmente.

Referencias

- J. A. Conejero, A. Falcó and M. Mora, *Structure and approximation properties of Laplacian-Like matrices.* Preprint (2022).
- J. A. Conejero, A. Falcó and M. Mora, *On the tensor approximation of Watts-Strogatz networks.* Preprint (2023).
- V. Simoncini, *Numerical solution of a class of third order tensor linear equations.* Bollettino dell'Unione Matematica Italiana, 13:429–439, 2020.
- A. Ammar, F. Chinesta and A. Falcó, *On the convergence of a Greedy Rank-One Update Algorithm for a class of Linear Systems.* Archives of Computational Methods in Engineering, 17(4):473–486, 2010.
- A. Nouy, *Low-Rank Methods for High-Dimensional Approximation and Model Order Reduction.* Model Reduction and Approximation, 171–226, 2017.

Grant number ACIF/2020/269, supported by GENERALITAT VALENCIANA

UNIÓN EUROPEA
Fondo Social Europeo
El FSE invierte en tu futuro