

PEP N°1 - Algoritmos Numéricos

Israel Arias Panéz

Departamento de Ingeniería Informática

Universidad de Santiago de Chile, Santiago, Chile

israel.arias@usach.cl

Resumen—En el presente documento se presentan los resultados de la PEP N°1 de la asignatura Algoritmos Numéricos. Se muestra un análisis de distintos métodos numéricos para resolución de sistemas de ecuaciones implementados en Matlab, se presentaran los resultados y posteriormente se hará un análisis el que contempla una comparación de los errores, costos espaciales y costos temporales obtenidos por cada método.

I. INTRODUCCIÓN

En la presente PEP se efectuó la resolución de sistemas de ecuaciones mediante métodos numéricos. “Un método numérico es un procedimiento mediante el cual se obtiene, casi siempre de manera aproximada, la solución de ciertos problemas realizando cálculos puramente aritméticos y lógicos (operaciones aritméticas elementales, cálculo de funciones, consulta de una tabla de valores, cálculo proposicional, etc.)” [1]. Se efectuó una simulación la cual consistió de resolver un sistema de ecuaciones obtenido de la formulación variacional de la ecuación de aproximación de la difusión de fotones de luz en un medio turbio, descrita en la ecuación 1.

$$-\nabla \cdot k(r)\Phi(r) + u_a(r)\Phi(r) + \frac{1}{c} \frac{\partial \Phi(r)}{\partial t} = q_0(r) \quad (1)$$

Con el fin de resolver distintos sistemas de ecuaciones por cada iteración de la simulación, es que se realizó una rotación del vector q 100 veces, una rotación por cada iteración. Estos distintos 100 sistemas de ecuaciones obtenidos de rotar el vector q serán resueltos por 3 métodos de resolución de sistemas de ecuaciones: un método iterativo, un método directo y un método de ecuaciones normales(ortogonales).

El presente documento tiene como objetivo el mostrar los resultados obtenidos de la PEP N°1 de la asignatura Algoritmos Numéricos, en la cual se plantearon las siguientes actividades:

1. Programar en Matlab tres métodos numéricos de resolución de sistema de ecuaciones: Método iterativo, Método directo y Método de ecuaciones normales(ortogonales).
2. Usar los métodos para resolver los 100 distintos sistemas de ecuaciones obtenidos de rotar q en cada iteración de la simulación.
3. Registrar en cada iteración los errores mínimos, costos temporales y costos operacionales
4. Graficar los errores, costos temporales y operacionales para los 100 distintos sistemas de ecuaciones.
5. Realizar una análisis de la eficacia y eficiencia de los métodos de acuerdo a los resultados obtenidos.

II. METODOLOGÍA

En esta sección se presentara la metodología usada para la resolución de la PEP N°1.

II-A. Métodos Numéricos

Como se menciona en la sección de introducción, en esta experiencia de laboratorio se usaran tres métodos numéricos para resolver sistemas de ecuaciones, un método iterativo, un método directo y un método de ecuaciones normales(ortogonales). los métodos a utilizar son los siguientes:

- Método de Jacobi (Iterativo).
- Método de Doolittle (Directo).
- Método de Gram-Schmidt Modificado (Ortogonal).

II-B. Sistema de ecuaciones

El sistema de ecuaciones a resolver es el sistema de ecuaciones descrito en la ecuación 1 en la sección de Introducción del presente documento, con el objetivo de tener 100 distintos sistemas de ecuaciones, en la simulación se realiza una rotación de q una vez por iteración, por un total de 100 iteraciones, por lo que se resolverán 100 sistemas de ecuaciones.

II-C. Error

Los errores mínimos para los tres métodos fueron medidos usando la norma 2 (norma euclídea) de la forma $\|Ax - b\|_2$ donde x es la solución obtenida por cada método.

II-D. Tolerancia y Condición de parada

Para el caso del método iterativo de Jacobi, es necesario establecer una condición de parada ligada a la tolerancia, con el fin de conseguir resultados con alta precisión, se asigno una tolerancia de 1×10^{-10} . La condición de para para el método de Jacobi se estableció como: Si la norma de la diferencia entre el resultado de la iteración +1 y el resultado es actual es menor a la tolerancia fijada, se corta la ejecución del método y se retorna el resultado, o sea:

Si $|x_{i+1} - x_i| < tolerancia$

Retornar resultado

Además se fija la condición de parada por iteraciones, si el método sigue corriendo luego de 10.000 iteraciones, parara y retornara su resultado.

II-E. Costo temporal

El costo temporal es el tiempo que tarda en ejecutarse el método hasta entregar un resultado, con el fin de poder medir este tiempo es que en la implementación se usan las funciones *tic* y *toc* de Matlab, las cuales permiten obtener el tiempo que tarda la máquina en la ejecución de un algoritmo medido en segundos.

II-F. Costo espacial

El costo espacial es la cantidad de operaciones que realiza el algoritmo de un método, las operaciones que realiza un algoritmo son operaciones matemáticas como sumas y restas, comparaciones, asignaciones a variables, multiplicaciones, divisiones, llamadas a funciones, etc.

Con el fin de poder medir el costo espacial se estableció una heurística, la cual asigna un valor a cada operación mencionada anteriormente, con el fin de que el costo espacial para cada método estará determinado por la suma de todos estos valores al final de la ejecución del método. La heurística implementada establece los siguientes costos:

- Sumas y restas | Condiciones y lógica | Asignaciones = 1 de costo.
- Multiplicaciones y divisiones = 2 de costo.
- Llamados a funciones = 5 de costo.

II-G. Valores promedio

Con el propósito de realizar un posterior análisis de los resultados obtenidos por los tres distintos métodos es que se calculara y almacenaran los valores promedio de los vectores solución en las 100 simulaciones, este valor promedio es calculado con la función *mean()* de Matlab, la cual calcula y retorna el valor promedio de todos los elementos del vector solución.

II-H. Especificaciones técnicas del equipo

Uno de los factores que influye en los resultados en los métodos numéricos son las limitaciones que pueda presentar el equipo en el cual se efectúa el cálculo, por ejemplo el costo temporal puede ser mayor o menor dependiendo de la máquina en la cual se ejecuta, por ende no esta de más mencionar las especificaciones técnicas en la cual se desarrollo esta PEP:

- CPU: Intel Core i5-6600.
- 16 GB de memoria RAM.
- Sistema Operativo: Windows 10 Pro.

III. RESULTADOS

En la presente sección se presentaran los resultados obtenidos de la simulación

III-A. Errores Mínimos

En la Figura 1 se muestra el resumen en un gráfico de los errores mínimos obtenidos por los tres métodos en cada iteración al resolver el sistema de ecuaciones obtenido en esa iteración por la rotación del vector q .

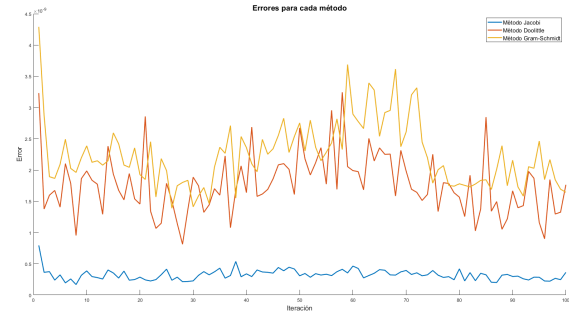


Figura 1: Errores mínimos de los tres métodos

III-B. Valores promedio

El resumen de los valores promedio de los vectores resultados obtenidos en la solución por los tres métodos en cada iteración son presentados en la Figura 2 en forma de una gráfica.

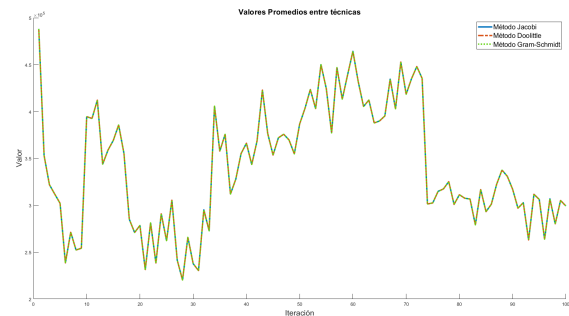


Figura 2: Valores promedio de los tres métodos

III-C. Costos operacionales

El resumen de los costos operacionales obtenidos por los tres métodos para cada iteración de la simulación es presentado en la Figura 3.

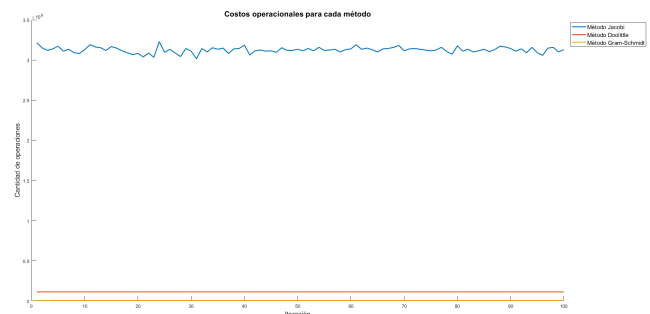


Figura 3: Costos operacionales de los tres métodos

III-D. Costos temporales

El resumen de los costos temporales obtenidos por los tres métodos para cada iteración de la simulación es presentado en la Figura 4.

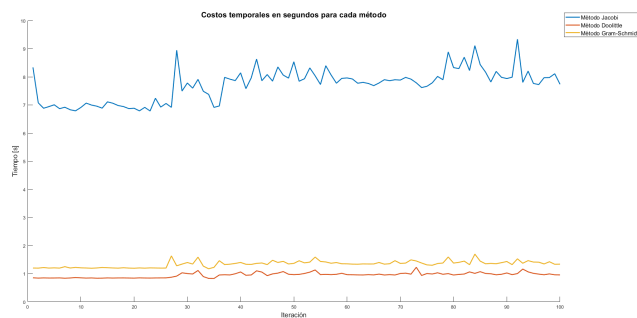


Figura 4: Costos temporales de los tres métodos

IV. ANÁLISIS

En esta sección se efectuara el análisis de los resultados presentados en la sección de resultados del presente documento

IV-A. Discusión de resultados

En primer lugar, al observar la Figura 1 disponible en la sección de resultados del presente documento es posible notar que el método que presento los errores mínimos más bajos a través de toda la simulación fue el método iterativo de Jacobi y el método que presento los errores más altos fue el método ortogonal de Gram-Schmidt.

Por otra parte, al observar las Figuras 3 y 4 es posible notar que el método con mayores costos operacionales y temporales es el Método iterativo de Jacobi, con costos bastante más elevados en comparación a los métodos de Doolittle y Gram-Schmidt. El costo temporal más bajo lo tiene el método de Doolittle con una baja diferencia con el método de Gram-Schmidt. El costo operacional más bajo lo tiene el método de Gram-Schmidt.

Al observar el gráfico de valores promedio en la Figura 2 es posible notar que todos los métodos consiguieron prácticamente los mismos valores promedio en toda la simulación para cada iteración, también es posible notar que los valores promedio fluctuaron bastante a través de toda la simulación, nunca estabilizándose o siguiendo una tendencia al contrario de todos los otros parámetros medidos, esto sucede debido a que el vector q es rotado en cada iteración, esta rotación es hecha por la función *circshift()*, sin embargo esta no es una simulación precisa por lo que el método de rotación por *circshift()* no es realmente como debería realizarse una rotación real para la ecuación de aproximación de la difusión de fotones de luz y por eso cada vez que se rota el vector q en cada iteración se obtienen valores medios muy distintos a la iteración anterior.

De los resultados es posible determinar que el método que posee la mayor eficacia es el método iterativo de Jacobi, ya que es el método que tiene el menor error mínimo de los tres métodos para los 100 sistemas de ecuaciones resueltos en la simulación.

A pesar que el método de Jacobi sea el más eficaz, es a la vez el más ineficiente, debido a que es el que más costo

temporal y operacional tiene de los tres métodos, el método más eficiente en termino de operaciones es el método ortogonal de Gram-Schmidt, ya que es el método que menos operaciones ejecuta para resolver un sistema de ecuaciones.

En términos de eficiencia de tiempo el más eficiente es el método directo de Doolittle, ya que es el método que menos tiempo toma en su ejecución para todos los sistemas de ecuaciones resueltos en la simulación.

El método de Doolittle es el que menos tiempo tarda en resolver y tampoco posee una diferencia considerable en costo operacional al método de Gram-Schmidt, es por esto que el método Directo de Doolittle es el más eficiente entre los métodos presentados para los sistemas resueltos en esta simulación.

V. CONCLUSIONES

En el presente documento se presento el desarrollo, implementación, resultados y análisis de la PEP N°1 comparando tres métodos numéricos enfocados a resolver sistemas de ecuaciones, como conclusiones significativas respecto a la simulación y análisis efectuados se puede destacar que para los métodos presentados el más eficaz es el método de Jacobi debido a que presenta siempre los menores errores mínimos, sin embargo es a su vez el método más ineficiente debido a sus altos costos operacionales y temporales. Por otro lado el método más eficiente para esta simulación en términos de tiempo es el método de Doolittle, por lo que en caso de tener que resolver un gran número de sistemas de ecuaciones con gran cantidad de variables el método de Doolittle podría ser el más eficiente dentro de los métodos presentados y lograr la tarea en el menor tiempo posible.

Queda como tarea el replicar la simulación pero resolviendo nuevos sistemas de ecuaciones con el fin de comprobar si los resultados de eficiencia y eficacia se mantienen a los conseguidos en esta simulación.

REFERENCIAS

- [1] R. S. Vasquez. Métodos numéricos para ingeniería. [Online]. Available: <https://disi.unal.edu.co/~lctorress/MetNum/LiMetNu2.pdf>