



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Métodos Numéricos

Trabajo Práctico 1

No creo que a él le gustará eso

Resumen

En este trabajo estudiaremos algoritmos para resolver sistemas de ecuaciones para problemas reales mediante una discretización. Se expondrán técnicas para obtener la temperatura en un punto crítico de un parabrisa usando eliminación Gaussiana y Factorización LU, aprovechando las matrices Banda, y la formula Sherman-Morrison Al final del trabajo, se llegarán a conclusiones sobre lo descubierto.

Integrante	LU	Correo electrónico
Armagno, Julián Adrián	377/12	julian.armagno@gmail.com
Balbachan, Alexis		
More, Ángel		
Pinzón, Germán		

Palabras claves:

Eliminación Gaussiana. Factorización LU. Punto Crítico. Matrices Banda. Sistemas de ecuaciones. Sherman-Morrison.

Índice

1. Introducción Teórica	3
2. Desarrollo	4
2.1. Matrices Banda	4
2.1.1. Por que queda una matriz Banda	4
2.1.2. Eliminacion Gaussiana	5
2.1.3. Factorización LU	6
2.1.4. BackWard Substitution	6
2.1.5. Eliminar sanguijuela	6
3. Resultados	7
3.1. Eliminar sanguijuela	7
3.1.1. Experimento para Hipótesis 1	7
3.1.2. Experimento para Hipótesis 2	8
4. Discusión	10
4.1. Eliminar sanguijuela	10
5. Conclusiones	11
6. Apéndices	11

1. Introducción Teórica

2. Desarrollo

2.1. Matrices Banda

2.1.1. Por que queda una matriz Banda

La obtención de la matriz Banda esta ligada directamente con nuestro sistema de ecuaciones, con la ecuación de calor discretizada y con el orden de numeración de las variables. Por ejemplo, la siguiente figura muestra un ejemplo de instancia de un parabrisa:

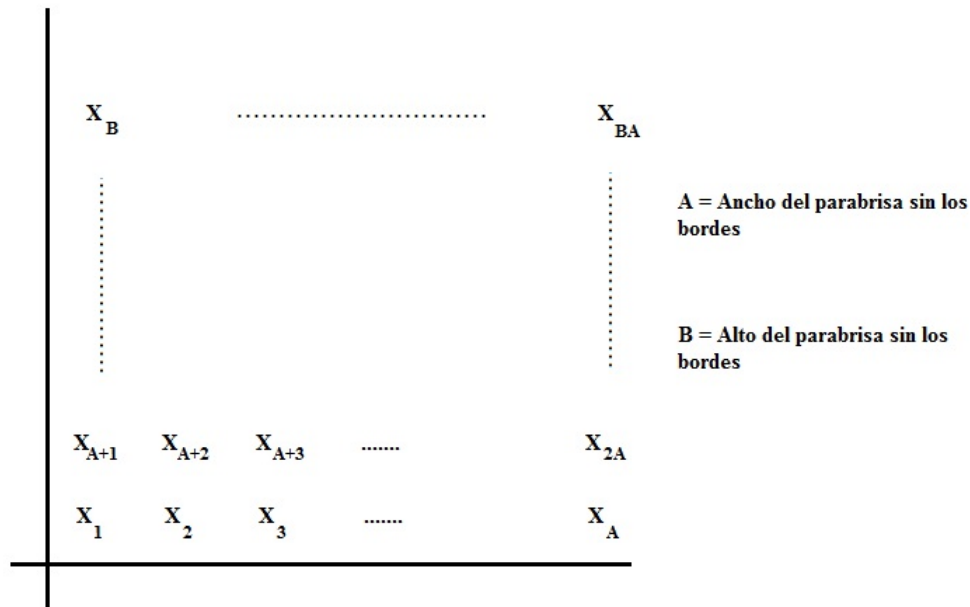


Figura 1: Cada X_i representa un punto de la discretización de este parabrisa

Empezemos a despejar las variables y armar nuestro sistema de ecuaciones:

- $X_1 = (X_2 + X_{a+1} - 100 - 100)/4$
 $4 \cdot X_1 - X_2 - X_{a+1} = -200$
- $X_2 = (X_1 + X_{a+2} + X_3 - 100)/4$
 $4 \cdot X_2 - X_1 - X_{a+2} - X_3 = -100$

Así, sucesivamente hasta despejar todas las incógnitas y completar el sistema. Como consecuencia del uso de la ecuación de calor, cada incógnita va a depender de una incógnita que este a la izquierda, derecha, arriba, y abajo de la misma. Esto se interpreta como que cada una de las variables, se relaciona con a lo sumo dos variable pegadas y con dos variable que están a A de distancia de la variable en estudio (Decimos a lo sumo, para especificar el caso de las variables que estás pegadas a los bordes, o sanguijuelas). Esto, y que empezamos despejando desde el extremo inferior izquierdo del parabrisa, son las causas de que la matriz quede establecida como una matriz banda.

Armando la matriz asociada al sistema de ecuaciones, nos da como resultado la siguiente figura, que muestra la relación de las variables con la distancia A .

$$A = \begin{pmatrix} 4 & -1 & \dots & \dots & -1 & 0 & \dots & \dots & 0 & -200 \\ -1 & 4 & -1 & \dots & \dots & -1 & 0 & \dots & 0 & -100 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & -1 & \dots & -1 & 4 & -1 & \dots & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & 0 & -1 & \dots & -1 & 4 & -1 & -100 \\ 0 & \dots & \dots & \dots & 0 & -1 & \dots & -1 & 4 & -200 \end{pmatrix}$$

A de distancia

2.1.2. Eliminacion Gaussiana

Algorítmicamente, la idea es similar a la Eliminacion Gaussiana clásica. En este caso, como consecuencia de nuestro sistema de ecuaciones, la matriz resultante del sistema asociado se compone por una diagonal integrada:

- Por números 4, si se trata de una posición sin sanguijuela.
- Por números 1, si se trata de una posición con sanguijuela.

Las demás posiciones de la matriz se llenan con 1, -1 o 0, depende el caso. Aprovechando las características de la diagonal y los posibles números, no es necesario intercambiar filas debido a que, al triangular, la diagonal nunca va a tener números 0. Además, ya que estas matrices tienen la cualidad de tener diagonales de 0 en los extremos superior derecho e inferior izquierdo nos simplifica las siguiente operación en relación al algoritmo clásico:

- No es necesario restar la fila completa, por la presencia de números 0 en el extremo superior derecho.
- Por cada paso que avanza en la diagonal, no es necesario realizar la resta de filas hasta abajo, por la presencia de 0 en el extremo inferior izquierdo

Además, en el momento de ir consiguiendo los número 0 de la triangulación, se los asigna directamente para no tener problemas de precisión. Resumiendo, el código quedaría así:

Algorithm 1 EliminacionGaussiana0()

```

0: for  $j = 0; j < \text{filas}; j++$  do
0:   for  $(i = j + 1; (i \leq j + \text{ancho})(i < \text{filas}); i++)$  do
0:     coeficiente = (Obtener(i,j)/Obtener(j,j));
0:     RestarFila0(coeficiente, j, i, j+1);
0:   end for
0:   cerosAlzquierda(j,j);
0: end for
=0

```

Siendo ancho, el ancho del parabrisa, es decir el ancho de la banda, filas la cantidad de filas de la matriz y restarFila0 es la función para restar filas con la particularidad de lo expresado anteriormente

2.1.3. Factorización LU**2.1.4. BackWard Substitution**

Aprovechando la estructura de las matrices bandas, al aplicar Eliminación Gaussiana, la matriz triangular superior resultante hereda las bandas superiores de 0. Ante esto el algoritmo clásico de BackWard Substitution se puede modificar para aprovechando que algunos valores con 0. El caso de ForWard Substitution es análogo

2.1.5. Eliminar sanguijuela

Ante lo observado y desarrollado, se llevará a cabo una cierta cantidad de experimentaciones para poder contrastar las siguiente hipótesis:

1. Al utilizar factorización LU y la formula de Sherman-Morrison, el algoritmo de EliminarSanguijuela logra una mejor performance en los casos de que existieran sanguijuelas unitarias.
2. Por otra parte quisimos ver que pasaba al modificar la granularidad. En otras palabras, verificar si al disminuir la misma, la posibilidad de encontrar sanguijuelas unitarias también disminuía, y en consecuencia el algoritmos modificado con Sherman-Morrison bajaba en su rendimiento, asemejándose en complejidad al algoritmo original

Para verificar la hipótesis (1), se llevará a cabo una experimentación tomando 3 instancias, cada una describiendo el caso mejor, caso promedio y caso peor de la cantidad de sanguijuelas unitarias. Para concluir se tomarán los tiempos de ejecución del algoritmo original y el algoritmo modificado por Sherman-Morrison por cada instancia y se los comparará.

Para verificar la hipótesis (2), se llevará a cabo otra experimentación tomando 3 instancias exactamente iguales, a excepción de la granularidad, que se la irá cambiando.

3. Resultados

3.1. Eliminar sanguijuela

3.1.1. Experimento para Hipótesis 1

Se modelaron 3 instancias, todas con los mismos valores de altura, ancho, granularidad, y cantidad de sanguijuelas. Lo único que se modificó en cada instancia fue el radio de estas últimas, principalmente con la idea de que:

- 1 Instancia = Todas sanguijuelas unitarias.
- 2 Instancia = Mitad sanguijuelas unitarias, mitad no unitarias.
- 3 Instancia = Todas sanguijuelas no unitarias.

La temperatura de las sanguijuelas no es relevante al experimento, ya que no nos agrega nada saber la temperatura del punto crítico al mismo, y por simplicidad las mismas fueron ubicadas en posiciones múltiplos de la granularidad

Los resultados obtenidos fueron los siguientes:

Algoritmo original			
Datos	Instancia 1	Instancia 2	Instancia 3
Entrada	"exp1.in"	"exp2.in"	"exp3.in"
N° de ejecuciones	10	10	10
Tiempo de ejecuciones	"MedicionesMetodo2.txt"	"MedicionesMetodo2b.txt"	"MedicionesMetodo2c.txt"
Media (en segundos)	38.78939	29.31231	26.0657
Desvío Standard	0.051020926100612	0.066167974881499	0.0047303277050597

Algoritmo modificado con Sherman-Morrison			
Datos	Instancia 1	Instancia 2	Instancia 3
Entrada	"exp1.in"	"exp2.in"	"exp3.in"
N° de ejecuciones	10	10	10
Tiempo de ejecuciones	"Mediciones Metodo3.txt"	"Mediciones Metodo3b.txt"	"Mediciones Metodo3c.txt"
Media (en segundos)	1.85237	15.23841	27.12823
Desvío Standard	0.003818334 7157167	0.0268594657 44926	0.0348354431 51916

(Los archivos de entrada y los tiempos de corridas se puede encontrar en Experimentos/Experimento3).

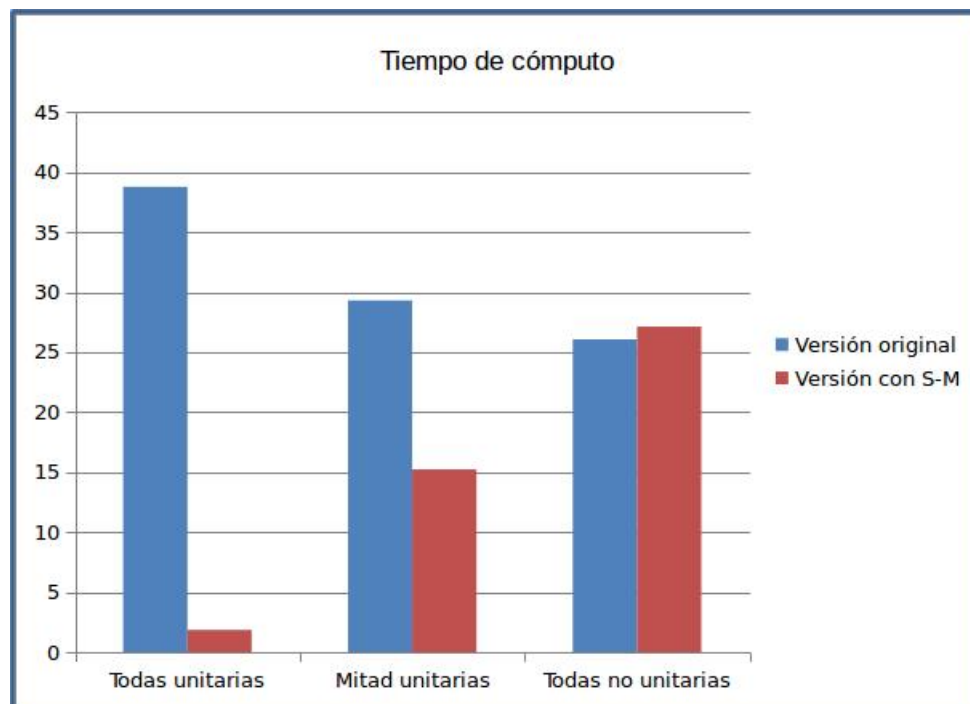


Figura 2: Gráfico obtenido de los valores de la media del tiempo de cada ejecución

3.1.2. Experimento para Hipótesis 2

Se modelaron 3 instancias, todas con los mismos valores de altura, ancho, cantidad de sanguijuelas, modificando la granularidad en cada una. El radio de las sanguijuelas se generó pseudo-

aleatoriamente con MATLAB en un rango posible de 0.1 a 16. Se eligió este rango para que el experimento sea más representativo con las granularidades elegidas para experimentar. Por simplicidad, las mismas fueron ubicadas en posiciones múltiplos de 16. Las instancias son las siguientes:

- 1 Instancia = Granularidad 8
- 2 Instancia = Granularidad 4
- 3 Instancia = Granularidad 2

La temperatura de las sanguijuelas no es relevante al experimento, ya que no nos agrega nada saber la temperatura del punto crítico al mismo.

Los resultados obtenidos fueron los siguientes:

Datos	Instancia 1	Instancia 2	Instancia 3
Entrada	"exp4.in"	"exp5.in"	"exp6.in"
Nº de ejecuciones	10	10	10
Media Algoritmo original (en segundos)	0.0593982	0.9738903	17.79381
Media Algoritmo modificado con Sherman-Morrison (en segundos)	0.0409079	0.7462364	15.92311
Porcentaje de eficiencia de Sherman-Morrison	45,19%	30,50%	11,74%

(Los archivos de entrada y los tiempos de corridas se puede encontrar en Experimentos/Experimento3).

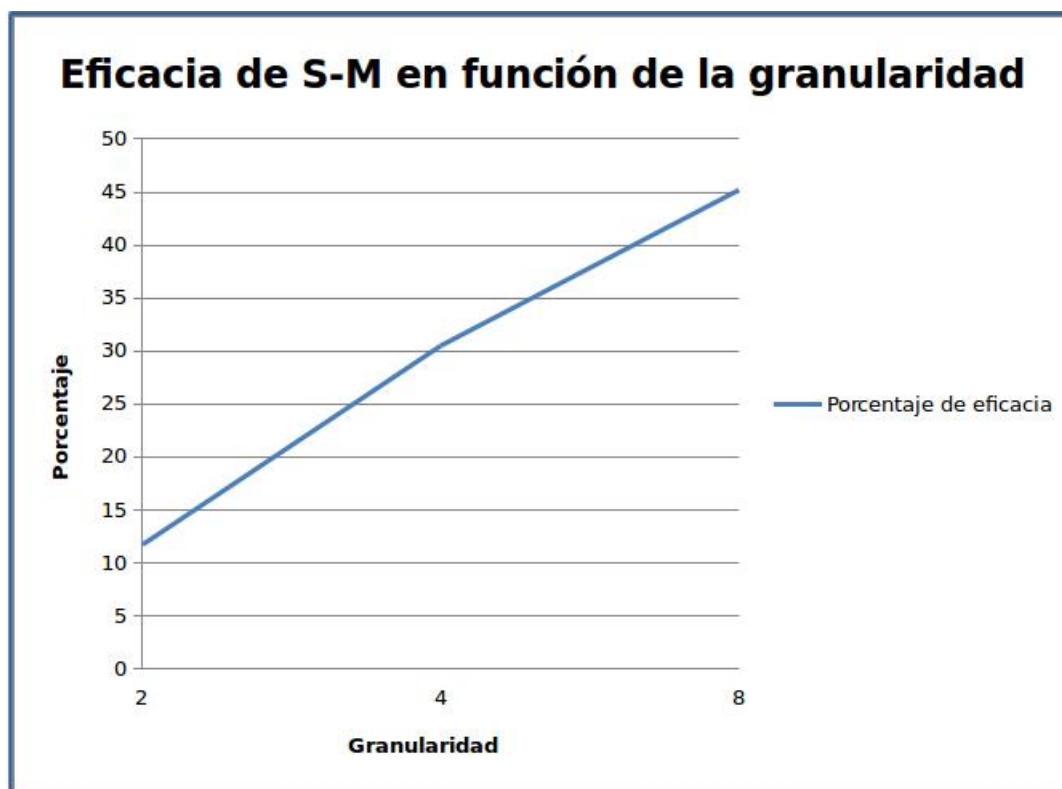


Figura 3: Gráfico obtenido de los valores del porcentaje de eficiencia del algoritmo con Sherman-Morrison comparándolo con el algoritmo original en función de la granularidad.

4. Discusión

4.1. Eliminar sanguijuela

Mediante la experimentación realizada, se logró verificar la hipótesis (1). Realmente, el tiempo de cómputo disminuye notablemente al usar el algoritmo modificado con Sherman-Morrison en casos donde haya sanguijuelas unitarias. Los resultados mostraron que cuanto más sanguijuelas unitaria haya, menor va a ser el tiempo de cómputo del mismo, comparado con el algoritmo original.

Esto significó un acercamiento práctico a lo expuesto teóricamente en la sección del desarrollo, es decir, que se pudo constatar la justificación teórica por la cual es mejor aplicar Sherman-Morrison. Un caso para resaltar, es cuando no hay ninguna sanguijuela unitaria, los 2 algoritmos tienen un tiempo de cómputo similar, hasta en algunos casos el algoritmo modificado con Sherman-Morrison tarda unos segundos más que el original, debido a que tiene más saltos condicionales y evalúa más expresiones.

En cuanto a la hipótesis (2), aunque asumimos predicados muy fuertes para que la experimentación sea representativa y no librada al azar (como que las sanguijuelas estén ubicadas en puntos múltiplos de la granularidad por ejemplo), se logró verificar lo que intuitivamente pensamos. Al disminuir la granularidad, la distancia entre 2 puntos cualesquiera se achica y la cantidad de puntos de la discretización aumenta. Esto implica que la probabilidad de encontrar sanguijuelas unitarias también disminuye, ya que es más probable, que al haber más puntos, cada sanguijuela toque más de un punto del parabrisa.

5. Conclusiones

6. Apéndices

Mayúsculas