**Introducción**

Cuando se tiene un algoritmo que permite resolver problemas de cálculo, hay que tener conocimiento de que tan alto es su costo de cálculo, es decir, cuantas operaciones deberá realizar el programa para obtener los que resultados deseados. Esto permite calificar la eficiencia del algoritmo respecto a otros que resuelven el mismo problema. Para determinar la calidad del algoritmo existe una variedad de formas, pero el motivo de la existencia de estos métodos en el ámbito de la informática es saber si un determinado algoritmo es capaz de resolver un problema, es decir saber si el algoritmo será capaz en algún momento de llegar a una solución.

El problema será la creación de un algoritmo que logre ser los más eficiente posible usando la menor cantidad de instrucciones para determinar que tanto esfuerzo fue necesario para conseguir la solución.

El objetivo de este estudio será evidenciar el esfuerzo que necesita el procesador para resolver sistemas matriciales de orden cuadrático, es decir determinaremos si una matriz de orden n es un sistema de ecuaciones estables, para esto se medirá el tiempo que fue necesario para determinar que ese sistema de ecuaciones era estable, y como el esfuerzo y tiempo para determinar la estabilidad de la matriz aumenta a medida que el orden crece.

**Problema**

El esfuerzo computacional en este caso de estudio esta directamente relacionado con tiempo necesario de parte del procesador para terminar de ejecutar el conjunto de instrucciones necesarias para determinar la estabilidad de las matrices cuadráticas. La estabilidad de una matriz cuadrática existe si y solo si se cumplen las siguientes dos condiciones.

Estas condiciones implican que existe una matriz A ubicada al lado negativo de un sub espacio y además un valor determinado de la matriz debe ser un numero positivo. Cuando se cumplen están dos condiciones implica que el procesador a logrado encontrar una cantidad determinada de líneas y variables dependiendo del orden de la matriz que satisfacen las condiciones.

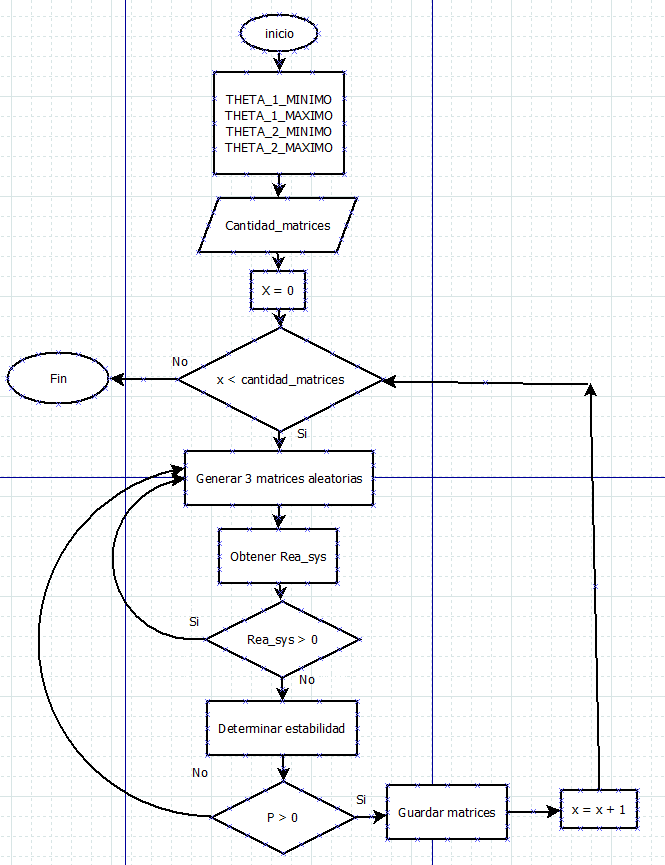
Lo primero será entonces la creación de este sub espacio, para esto será necesario tener dos valores mínimos y dos valores máximos. Siendo así posible evaluar una de las dos condiciones necesarias para las matrices estables.

La matrices o sistemas de ecuaciones del estudio fueron creadas de forma aleatoria con la función rand de Matlab, la cual genera una matriz cuadrática de orden n.



A continuación, determinaremos en qué lado del sub espacio están las matrices generadas, en el caso de estar al lado negativo o lo más cerca del cero, simplemente necesitaremos obtener el valor de p para conocer si es o no estable. Por el contrario de estar al lado positivo tenemos la opción de restar a cada matriz su simétrica con la finalidad de desplazar las matrices más cercanas al lado negativo.

Aquí se deja un diagrama de flujo representado el proceso básico descrito en los párrafos anteriores.



Debido a que los resultados pueden variar dependiendo de qué tan rápido el procesador es capaz de realizar una instrucción se dejaran detalles de que procesador, velocidad de instrucciones por segundo y cuanta memoria fue usada para obtener los resultados.

CPU: 7-4790

GHZ: 3.60

RAM: 8G

**Resultados**

|  |  |
| --- | --- |
| Orden | Tiempo promedio |
| 3 | 0.02266 |
| 4 | 0.01189 |
| 5 | 0.01281 |
| 6 | 0.01313 |
| 7 | 0.0157 |
| 8 | 0.01876 |
| 9 | 0.02135 |
| 10 | 0.02858 |

|  |  |
| --- | --- |
| Orden | Desviación estándar |
| 3 | 0.0080987 |
| 4 | 1.6746e-06 |
| 5 | 2.1353e-06 |
| 6 | 2.0334e-06 |
| 7 | 2.8788e-06 |
| 8 | 7.499e-07 |
| 9 | 7.7525e-07 |
| 10 | 1.1256e-05 |

|  |  |  |
| --- | --- | --- |
| Orden | Tiempo mínimo | Tiempo máximo |
| 3 | 0.014561 | 0.030759 |
| 4 | 0.011888 | 0.011892 |
| 5 | 0.012808 | 0.012812 |
| 6 | 0.013128 | 0.013132 |
| 7 | 0.015697 | 0.015703 |
| 8 | 0.018759 | 0.018761 |
| 9 | 0.021349 | 0.021351 |
| 10 | 0.028569 | 0.028591 |

**Conclusiones**

El esfuerzo de parte del procesador al momento de determinar la estabilidad de una matriz de orden n crece de forma significativa, el tiempo necesario parece crecer de forma exponencial hasta el punto que incluso resolver determinados sistemas pueden llegar requerir horas o incluso semanas para matrices con un orden de gran tamaño.

Las aplicaciones de este algoritmo pueden ser usadas en distintas áreas, para comprobar si ciertas combinaciones de variables pueden ser efectivas. Como las combinaciones de ciertas características de componentes al momento crear un objeto más grande como los son aviones, autos, etc. Simular efectos de los químicos encapsulados en medicamentos y observan el cómo se desempeñan antes de ser aprobados. Unas sugerencias podrían ser buscar o crear formas de que el sistema o combinaciones de variables tiendan a ser estables con reconocimiento de parationes, para determinar que variables son las más significativas.

**Bibliografía**

Costo computacional de un algoritmo

Nombre: Ricardo Vergara Toloza

Rut: 20.199.732-1