

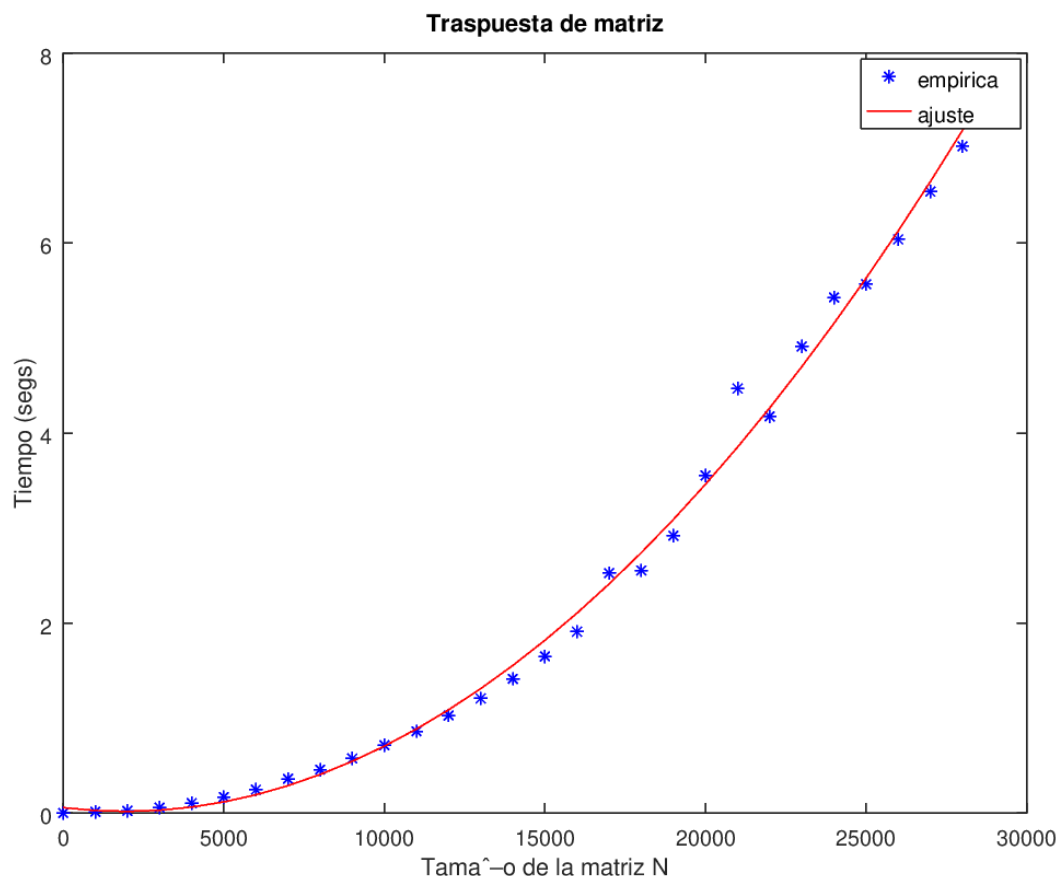
PRÁCTICA3

MANUEL FERNÁNDEZ MÍGUEZ

Introducción:

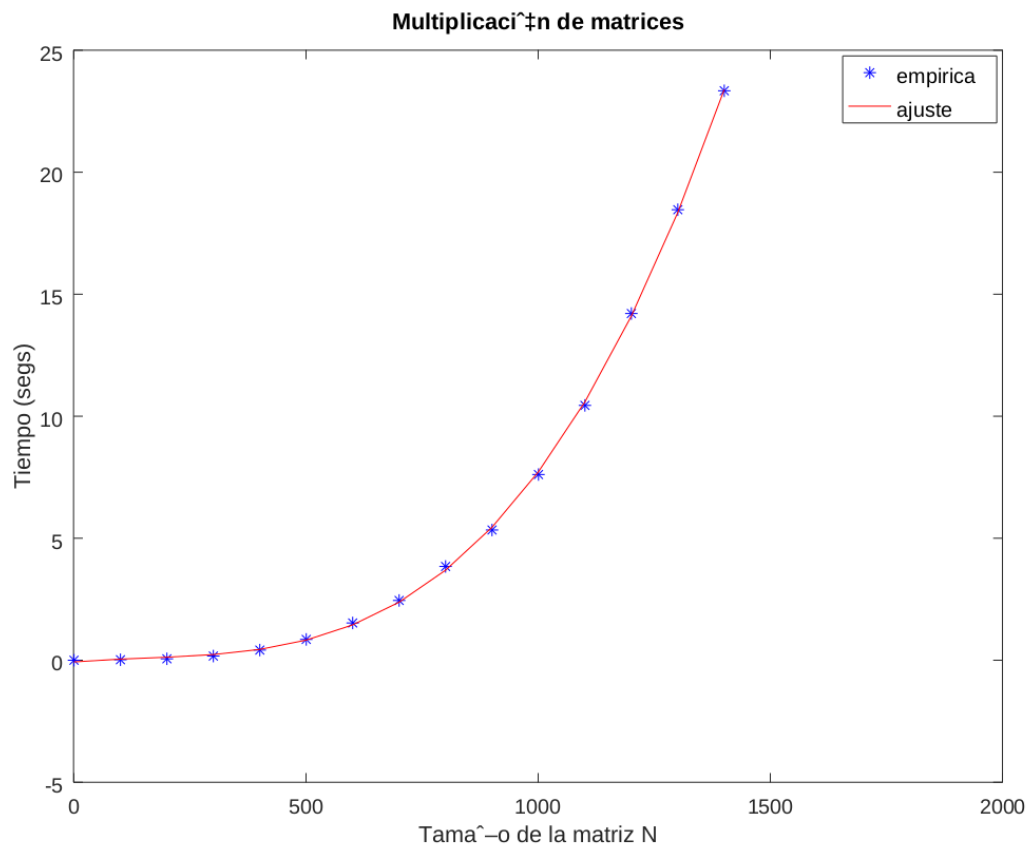
En esta práctica debemos comprobar la diferencia en el tiempo de ejecución entre dos operaciones (multiplicación de matrices y trasposición de matrices) que tienen diferente complejidad temporal, para ello programamos un código que nos devuelva dependiendo del input introducido un fichero y unos datos que luego se utilizarán empleando Octave para generar unas gráficas que a su vez representarán y nos servirán para explicar la complejidad temporal antes mencionada de cada operación, la cuál justificaremos según los datos aportados, para la medición temporal de las operaciones. Aclarar que hemos utilizado la biblioteca `time.h` como se aclaraba que se debía usar, además se debe dar respuesta a una pregunta planteada dentro del pdf de la práctica sobre dónde es preciso en nuestro programa cambiar las variables de `short` a `long`, en la `matrizdinamica.c` es preciso este cambio en la función `crear`, en su variable `i` debido a que se emplea en un bucle que hace `tamc * tamf` iteraciones y se nos dice que el máximo tamaño de la matriz podría ser `30000x30000` y eso superaría el máximo de la capacidad de una variable `short`.

TRASPOSICIÓN DE MATRICES:



Para la traspuesta de matrices usamos como tamaño inicial de la matriz el número 2 y tomamos un tamaño final de 30000 (debido a que en el planteamiento del problema se nos había dicho que el tamaño máximo de las filas de la matriz era 30000 y que observamos que podíamos alcanzar bien ese tamaño sin que el coste temporal se volviese inasumible), además usamos un tamaño de paso de 1000 para que tomemos suficientes datos para hacer que la gráfica tenga suficiente información y por lo tanto se ajuste más al coste temporal que tiene el programa. Por último al observar la gráfica se puede apreciar un orden temporal de $O(n^2)$, es decir, cuadrático, esto es debido a que según va avanzando n (el tamaño de la matriz) el coste temporal al principio aumenta poco a poco y al llegar al final el crecimiento se va haciendo cada vez mayor pero sin curvarse tanto como sería el caso de una gráfica que representase un crecimiento temporal cúbico, además esta operación fue implementada mediante el uso de dos bucles de n y $n/2$ iteraciones lo que produce un orden temporal de $O(n^2)$, por lo tanto se puede concluir que esta operación es cuadrática.

MULTIPLICACIÓN DE MATRICES:



Para la multiplicación de matrices utilizamos como tamaño inicial 2, como tamaño final 1500 y como tamaño de paso 100, esta elección diferente a la anterior se debe a que a partir de 1500/2000 de n (tamaño de la matriz) el coste temporal de esta operación se vuelve muy costoso y por lo tanto lleva demasiado tiempo, además con este rango de valores (y para tener suficiente cantidad de datos recogimos uno cada 100 n) se aprecia suficientemente bien que el orden temporal de la operación de multiplicación es $O(n^3)$ debido a que el crecimiento de la gráfica es mayor que la anterior gráfica (se curva más hacia arriba cuando crece) y que se puede observar como alcanza mucho antes valores que la gráfica de transposición alcanza al final de su recorrido, a parte de toda esta explicación la forma que toma la gráfica recuerda a otras gráficas cúbicas y esta operación fue implementada con el uso de tres bucles for, dos de n iteraciones y uno de $n/2$, lo que los hace $O(n^3)/2$ que es $O(n^3)$, por lo tanto el orden temporal de esta gráfica es cuadrático.

