

Medición de tiempos de ejecución en programas

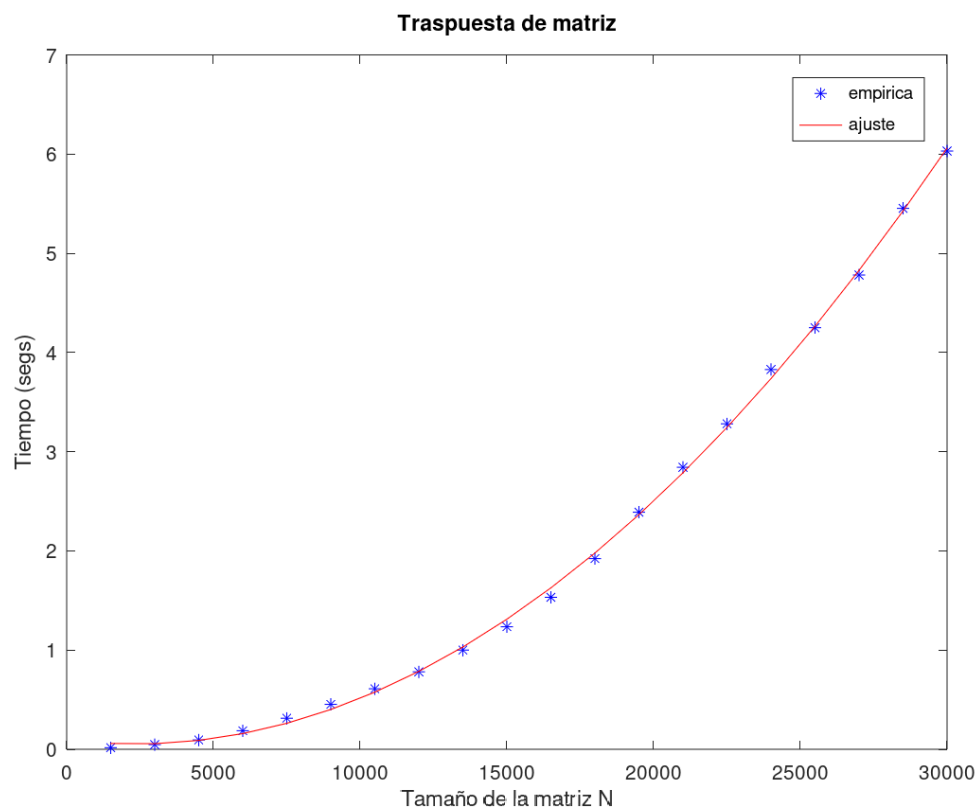
Pedro Durán Altés

pedro.duran@rai.usc.es

En este informe se recoge el estudio de la medida del tiempo ejecución de dos funciones, la primera referida a la trasposición de matrices y la segunda al cálculo de multiplicaciones de matrices. Para dicho estudio se representarán dos gráficas con unos rangos de tamaño que se especificarán a continuación.

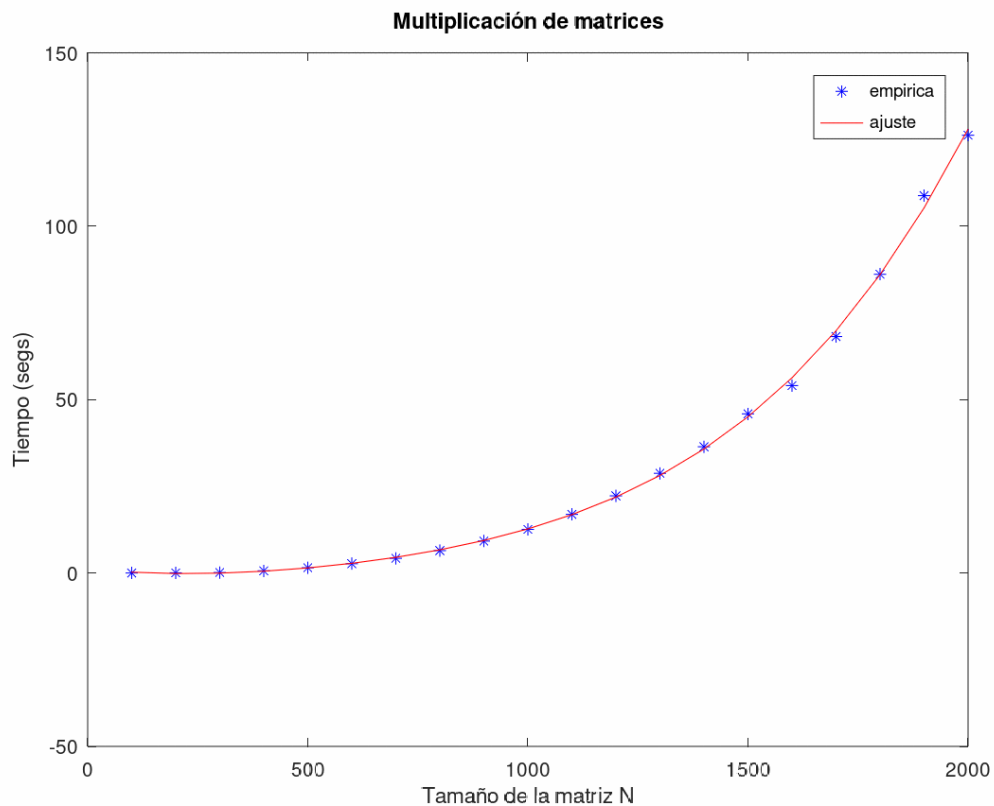
Debido a que el tamaño máximo para estos programas serán matrices 30000 x 30000 habría que hacer una modificación del TAD implementado para la Práctica 1 de la asignatura. La función `crear` cuenta con: `for(i=1; i <= tamf*tamc; i++)` y la variable `i` está declarada como una variable tipo `short`, por lo que se ha de reemplazar por una variable de tipo `long`, ya que el bucle puede llegar hasta `tamf*tamc`, por lo que habrá muchos casos en los que se pase del rango de una variable `short` (<32767).

1. Trasposición de Matrices



Para la representación de esta función se ha empleado un tamaño inicial de 1500, un tamaño final de 30000 y un paso de 1500, ya que estos rangos tienen un tiempo de ejecución razonablemente pequeño. Se estaría hablando de una **función cuadrática** debido a los **dos bucles for** implementados en la función.

2. Producto de Matrices



Para la representación de esta función se ha empleado un tamaño inicial de 100, un tamaño final de 2000 y un paso de 200, debido a que el tiempo de ejecución no es tan grande como si se usaran rangos más grandes, y con 20 iteraciones, como en el caso anterior. En este caso se estaría hablando de una **función cúbica**, debido a los **tres bucles for** implementados en la función.

En la función de Trasposición se puede observar como utilizando un tamaño máximo de 30000 el tiempo de ejecución será de unos 6 segundos, que ya es un tiempo considerado debido a las grandes dimensiones que se están empleando. Pero a la hora de analizar la función de multiplicación, se puede apreciar en la gráfica como utilizando un tamaño máximo de 2000 los tiempos de ejecución se disparan en comparación con el anterior algoritmo (debido a la complejidad del 2º), y aplicar este algoritmo con el tamaño máximo citado anteriormente (2000) implicaría un tiempo de ejecución de unos 2 minutos aproximadamente, que iría ascendiendo si el tamaño final siguiese aumentando, por lo que ya se estarían tratando tiempos de ejecución muy grandes.