Tarea 1
Algoritmos y complejidad

Harold Caballero; 201773602-k
Maximiliano Ojeda; 201773576-7
Katherine Salgado; 201610515-8

Junio 2020

1 Problema 1

El problema de multiplicar 2 matrices $(knxn) \times (nxkn)$ se puede solucionar multiplicando cada cuadrante de nxn en cada matriz, es decir, la primera matriz (matriz vertical), se puede separar en k sub-matrices de nxn, esta división se realiza comenzando en la fila 0 y cada n filas se tiene una nueva sub-matriz, lo mismo se hace con la otra matriz (matriz horizontal), pero en vez de avanzar por filas, se avanza por columnas, luego la matriz resultado de tamaño $knxkn$ se puede separar en k^2 sub-matrices partiendo en el punto $(0, 0)$ y avanzando n por cada fila, obteniendo matrices $nxkn$, para luego avanzar por las columnas de cada una de esas sub-matrices separándolas en k sub matrices de nxn.

Finalmente se puede decir que la sub-matriz de nxn en la posición (i, j), está dada por la multiplicación entre las sub-matriz i de la matriz vertical por la matriz j en la matriz horizontal.

2 Problema 2

El problema de multiplicar 2 matrices $(nxkn) \times (knxn)$ se puede resolver realizando la misma separación de sub-matrices del problema anterior en las matrices a multiplicar, pero la matriz resultante se obtiene de sumar todas las matrices obtenidas del producto entre la matriz i de la horizontal y la j de la vertical. Asumiendo que las filas crecen hacia abajo y las columnas hacia la derecha.

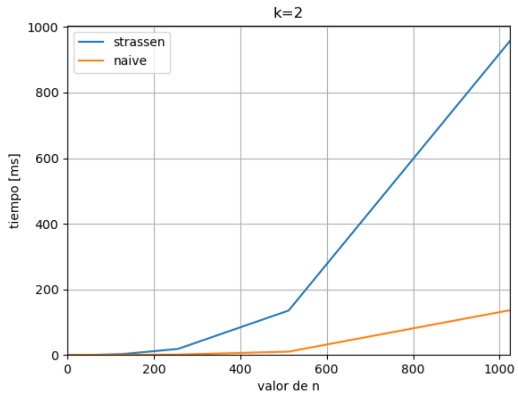
3 Observaciones

Sólo se tomo el tiempo que le tomaba a cada algoritmo resolver el problema 2 ya que requería de menos multiplicaciones, los algoritmos de multiplicación se utilizaron como sub rutina multiplicando solamente matrices de nxn.

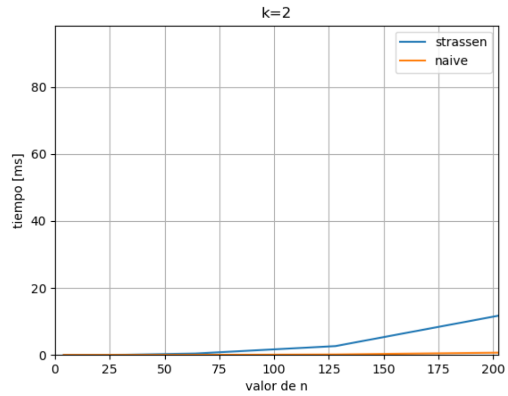
4 Conclusiones

En todos los gráficos se puede observar que el algoritmo naive siempre es más rápido, esto debido a que la complejidad de Strassen calculada en clases se utilizó tomando las multiplicaciones como el factor a castigar, sin embargo en procesadores modernos, la multiplicación de enteros está implementada de forma que no toma tanto tiempo como en la época en que se hizo el algoritmo, esto sumado a que Strassen tiene mucho movimiento de memoria aparte de la recursión, lleva a que la implementación sea más lenta que el algoritmo naive de multiplicación de matrices.

5 Resultados

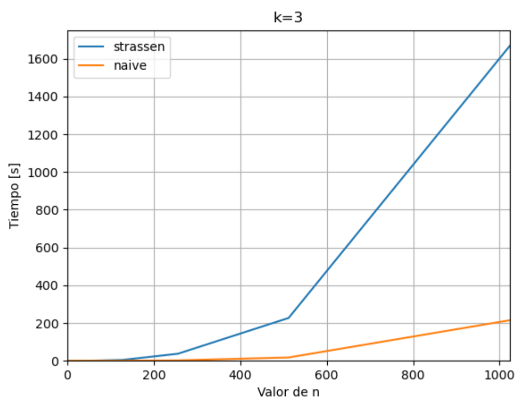


(a) Comparación Strassen y Naive.

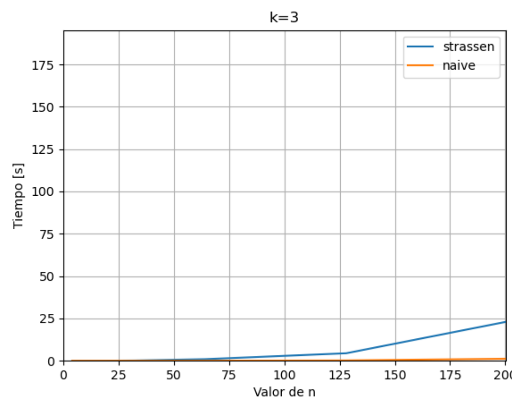


(b) zoom entre 0 y 200 del gráfico de la izquierda

Figure 1: Comparación para k=2.

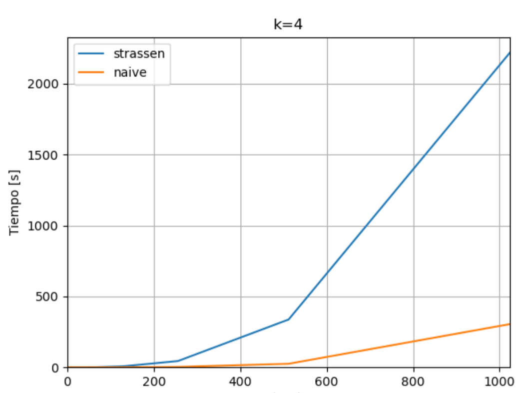


(a) Comparación Strassen y Naive.

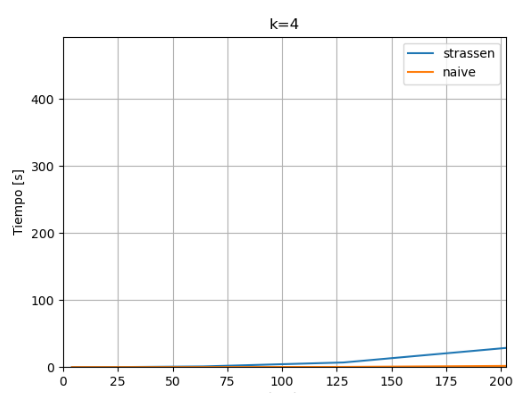


(b) zoom entre 0 y 200 del gráfico de la izquierda

Figure 2: Comparación para k=3.



(a) Comparación Strassen y Naive.



(b) zoom entre 0 y 200 del gráfico de la izquierda

Figure 3: Comparación para k=4.