



Análisis

Tarea 2 - Cerradura Convexa

Geometría Computacional

Integrante: Lucas Llort Bonnefont

Profesores: Nancy Hitschfeld K.

Auxiliar: Sergio P. Salinas

Ayudante: Andrés Abarca M.

Fecha de entrega: 3 de Julio de 2024
Santiago, Chile

Resumen

En este informe se detallan los resultados obtenidos en la experimentación con dos algoritmos que calculan la cerradura convexa de un conjunto de puntos, en particular, los algoritmos *Gift Wrapping* e *Incremental* (también conocido como *Monotone Chain*).

El objetivo del trabajo es estudiar como se comportan ambos algoritmos dados inputs de varios tamaños y con distintas proporciones entre los puntos que se encuentran en la cerradura y los que no, también se busca observar la correctitud de las predicciones teóricas sobre la complejidad temporal de ambos algoritmos.

Índice

1. Metodología	1
2. Resultados	1
3. Conclusión	5

1. Metodología

Ambos algoritmos fueron implementados utilizando las clases creadas en la tarea 1 del ramo, Punto, Vector y Polígono, ambos recibiendo un arreglo de puntos y retornando un polígono, el algoritmo *Incremental* recibe un parámetro adicional con el que indica si el arreglo que se le está entregando está ordenado, esto con el objetivo de investigar el impacto que tiene el paso de ordenamiento en el tiempo de ejecución del algoritmo.

Se realizaron dos experimentos llamados *Radial* y *Random*, el primero consiste en crear un conjunto de entrada en el cual todos los puntos de la cerradura convexa se encuentran a una distancia fija del origen, ordenados de manera que formen un polígono regular, y el resto de los puntos se distribuyen aleatoriamente dentro de la figura de manera que no formen parte de la cerradura, el conjunto luego se utiliza como input para ambas funciones para obtener sus tiempos de ejecución, este experimento permite fijar la proporción entre los puntos que pertenecen a la cerradura y los que no, por lo que se utiliza para probar como esta proporción afecta a los tiempos de ejecución.

En el segundo experimento, *Random*, se genera un conjunto de prueba con puntos distribuidos aleatoriamente, este experimento no entrega control sobre la cantidad de puntos que forman parte de la cerradura convexa, por lo que se utiliza para medir los tiempos que toman los algoritmos en un caso general.

Además del algoritmo incremental y Gift Wrapping, se realizaron los experimentos con la función Gift Wrapping con los conjuntos de puntos pre-ordenados, con el objetivo de saber cuanto afecta el paso de ordenar los puntos al tiempo total que toma el algoritmo.

2. Resultados

En la siguiente tabla se muestran gráficos en los que se tomaron cantidades fijas de puntos en la cúpula convexa y se le tomo el tiempo a los algoritmos con cantidades crecientes de puntos al interior de la cúpula, se puede observar que el el algoritmo de Gift Wrapping crece rápidamente con respecto a la cantidad de puntos, mientras que el tiempo de ejecución del algoritmo incremental se mantiene casi despreciable.

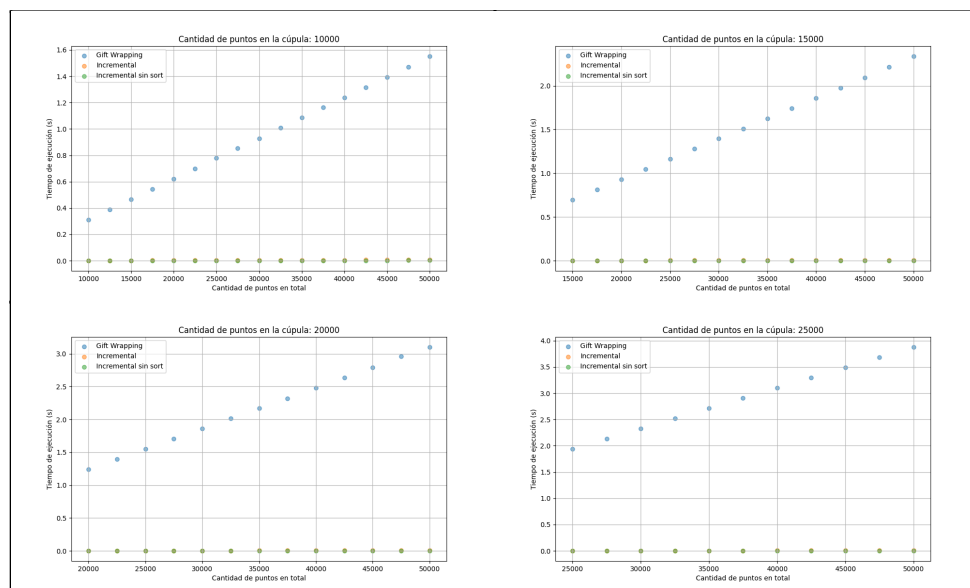


Tabla 1: Gráficos mostrando como evoluciona el tiempo de ejecución con respecto a la cantidad de puntos totales, fijando cantidades de puntos en la cúpula.

En la Tabla 2 se muestran gráficos con una medición similar a la dela tabla anterior pero con cantidades de puntos mucho menores, se puede apreciar que para cúpulas muy pequeñas Gift Wrapping resulta

ser mejor que el algoritmo incremental, pero a medida que aumenta el tamaño de las cúpulas la curva de Gift Wrapping se pone por encima.

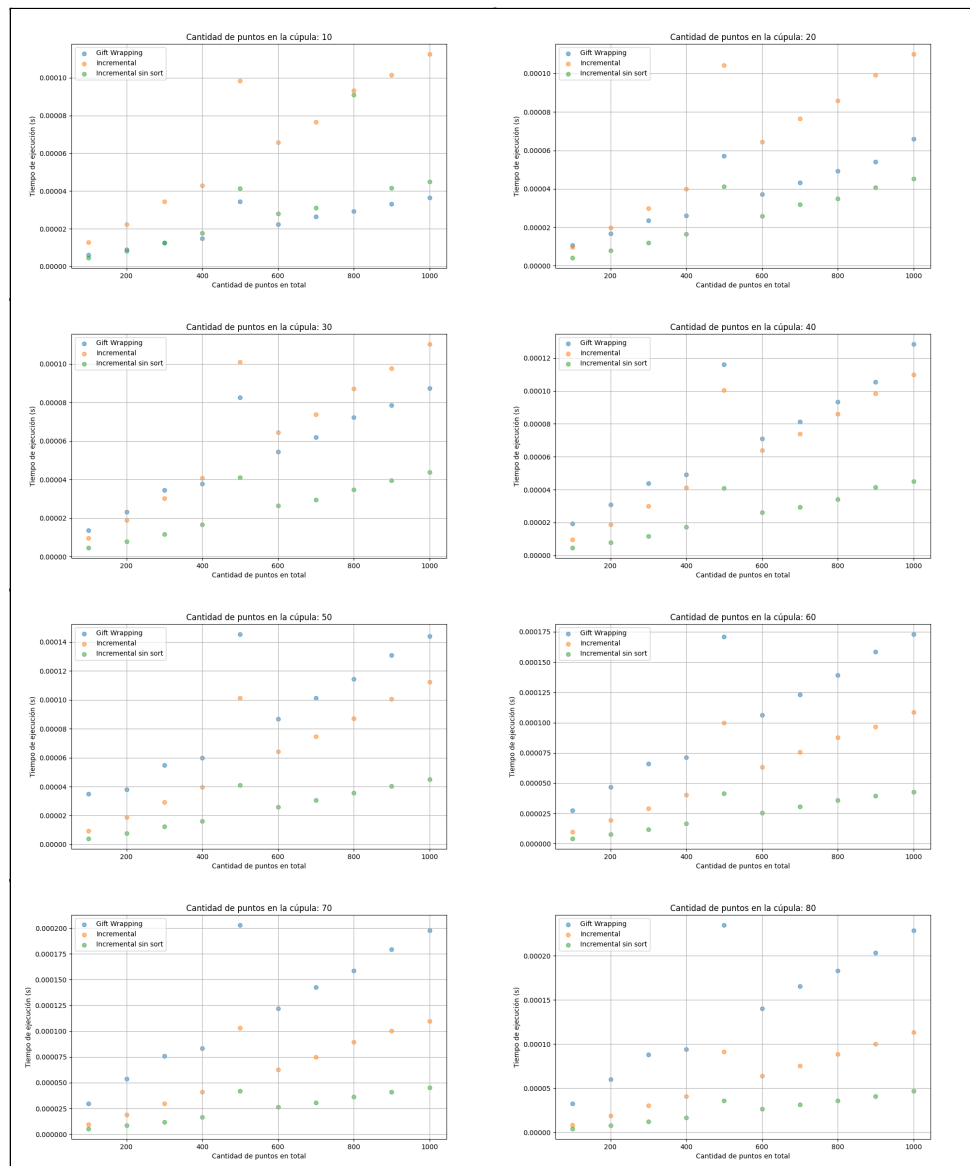


Tabla 2: Gráficos mostrando como evoluciona el tiempo de ejecución con respecto a la cantidad de puntos totales, fijando cantidades de puntos en la cúpula.

En la Tabla 3 se muestra un resultado similar al de la anterior, tomando cantidades de puntos grandes pero manteniendo las cúpulas pequeñas, se puede observar que la diferencia entre los dos algoritmos es menos notoria cuando las cúpulas tienen menor tamaño.

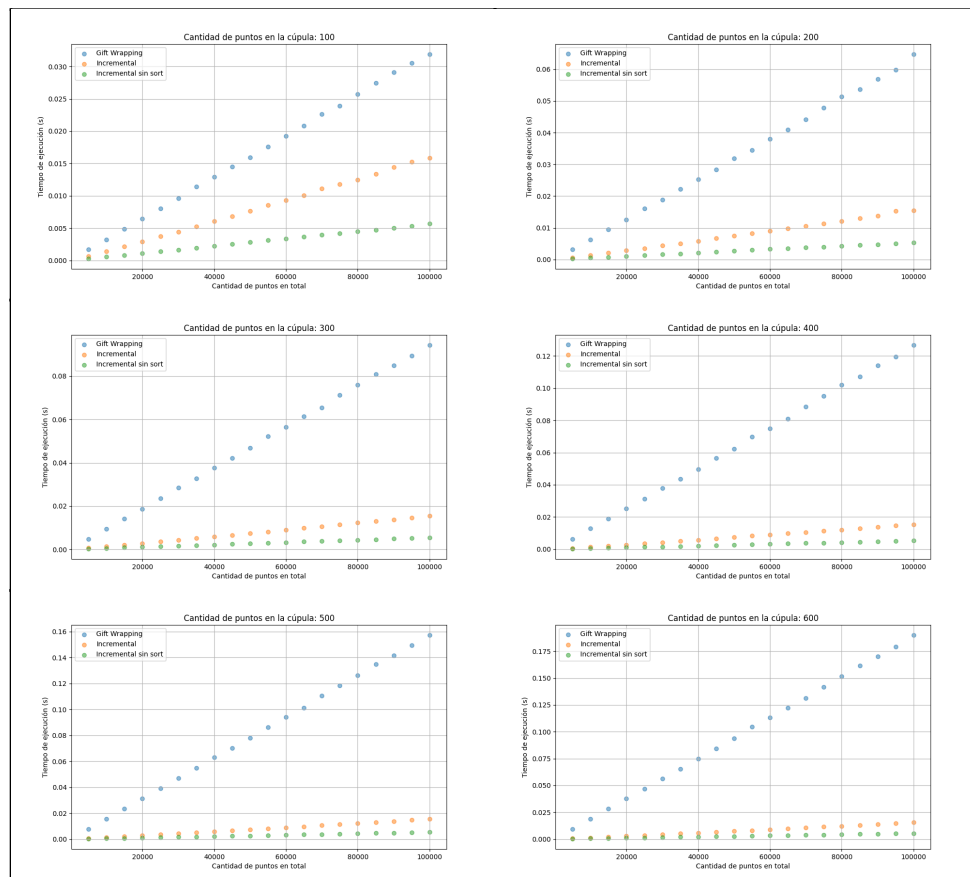


Tabla 3: Gráficos mostrando como evoluciona el tiempo de ejecución con respecto a la cantidad de puntos totales, fijando cantidades de puntos en la cúpula.

En los siguientes gráficos se puede apreciar como aumenta el tiempo de ejecución del algoritmo con respecto a la proporción entre puntos que pertenecen a la cúpula y los puntos totales, se puede observar que para las distintas cantidades de puntos fijadas los resultados son bastante similares, el tiempo de ejecución de Gift Wrapping crece rápidamente con respecto a la proporción mientras que el algoritmo incremental se mantiene constante.

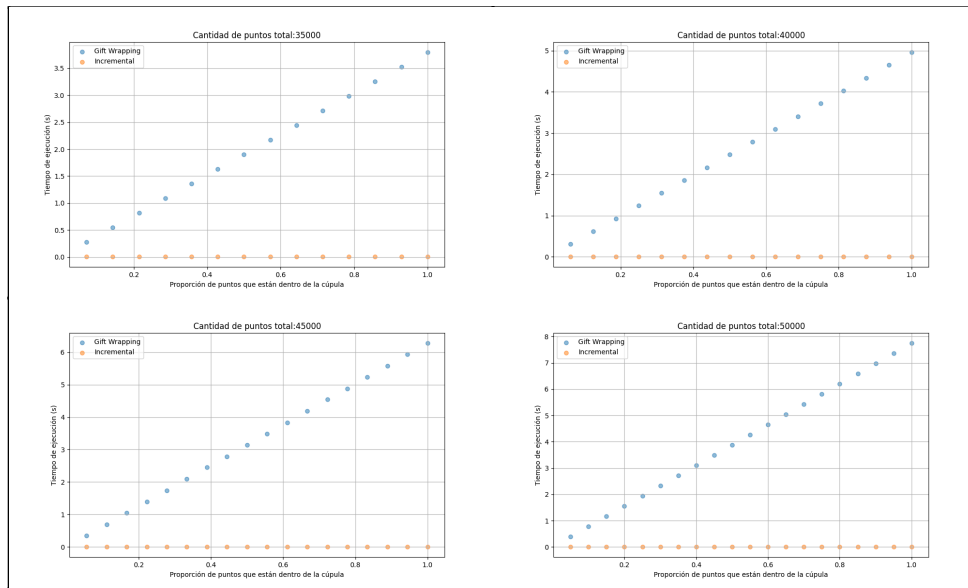


Tabla 4: Gráficos mostrando la evolución del tiempo que toman los dos algoritmos con respecto a la proporción de puntos pertenecientes a la cúpula convexa.

El mismo resultado se puede observar cuando se toman cantidades menores de puntos, como se puede apreciar en la Tabla 5, además se puede apreciar el punto en el que el algoritmo Gift Wrapping se vuelve peor que el algoritmo incremental en tiempo, además se puede observar un como el tiempo del algoritmo incremental decrece ligeramente a medida que aumenta la proporción de puntos pertenecientes a la cúpula.

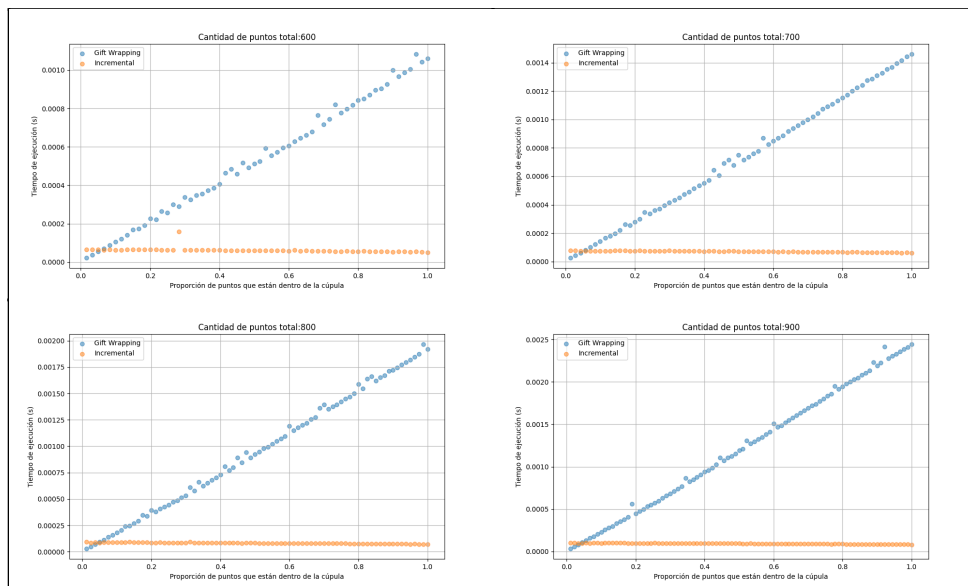


Tabla 5: Gráficos mostrando la evolución del tiempo que toman los dos algoritmos con respecto a la proporción de puntos pertenecientes a la cúpula convexa.

La Figura 1 muestra el tiempo que toman los algoritmos para una cierta cantidad de puntos, donde todos los puntos están distribuidos de manera aleatoria, se puede observar que el algoritmo Gift Wrapping tiene un comportamiento más errático, esto se puede deber a que el tiempo de ejecución del algoritmo es fuertemente influenciado por la la cantidad de puntos pertenecientes a la cúpula convexa, y esta cantidad varía bastante, por otro lado, el algoritmo incremental tiene un comportamiento mucho más predecible.

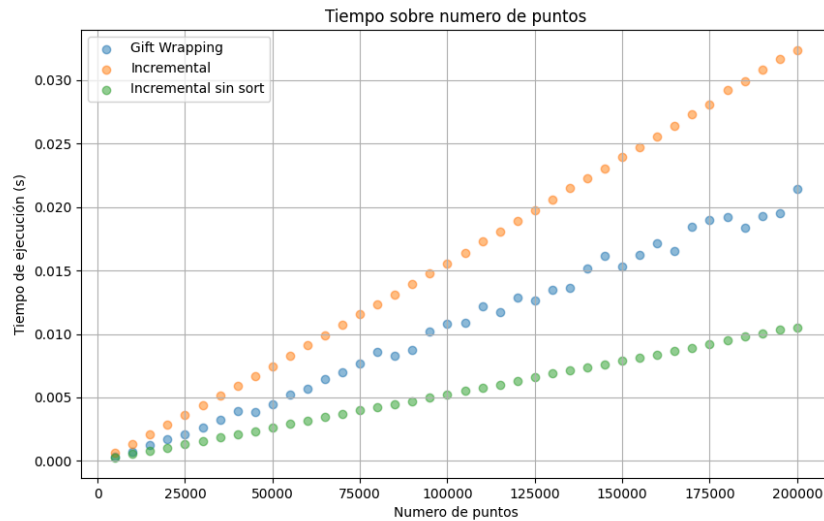


Figura 1: Diagrama que muestra como varía el tiempo de ejecución de los algoritmos en relación a la cantidad de puntos, dada una distribución de puntos aleatoria.

3. Conclusión

Los resultados parecen indicar que el algoritmo de Gift Wrapping resulta ser mejor que el algoritmo incremental cuando el tamaño de la cúpula convexa es pequeño, este hecho calza con el hecho de que por cada punto que pertenezca a la cúpula, Gift Wrapping tiene que iterar por todos los puntos del conjunto de entrada.

Por otro lado, el algoritmo incremental es mucho mejor que Gift Wrapping cuando el tamaño de la cúpula es grande, y de hecho, el tiempo de ejecución de este decrece ligeramente a medida que aumenta el porcentaje de puntos pertenecientes a la cúpula, esto calza con el hecho de que el algoritmo se basa en backtracking y mientras más puntos pertenezcan a la cúpula, menor debe ser la cantidad de backtracking.