# COSC 262 Algorithms

# Assignment: Convex Hull


## Student Name : Xiao Meng

## Student ID : 88211337


## 1. Description

In this assignment, I implemented three algorithms for computing convex hull of all points in set A and set B with Python. The Gift Wrap algorithm, Graham-Scan algorithm and Monotone Chain algorithm. Then analysis the amount and time for each results.

## 2. Algorithm implementation

- **Gift Wrap Algorithm**

The input data structure used in Gift Wrap algorithm is list which is alterable and easy to extract. By packing each point as an element of this list in a form of (x, y) to represent the coordinate of vertex. To implemented this, an extra function ReadDataPts(filename, N) is used. This function reads N lines of each file, changed data type to float and packed datas as a form of point into a list.

As the method mentioned in lecture, there are two extra functions used for implementing this algorithm. One is introduced in paragraph one, another function is theta(ptA, ptB). This function computes an approximation of the angle between the line AB and a horizontal line through A. It returned a value of angle which need to be compared to find the vertex in convex hull.

Implementing this algorithm, the first step is to find the right most point with minimum y-value. Then swap this point to the last element of list. Recurisively compare this point to each point to choose one generaed the nimimum angle which is greater than the previous minimum angle. Swap it with the next position of previous chosen one until the chosen point is the start one. The part before the vertex 0 is the result of convex hull vertices. In a special case, the chosen point is in a tie with the start point which would generate the ange of 0 degree with the start point. In this case, he theta() function would set the angle to 360 degree when angle is equal to 0 degree.

- **Graham-Scan Algorithm**

The input data structure is the same as above. Except for function readDataPts, there are two extra functions used in this algorithm.

lineFn(ptA, ptB, ptC) returns the true if these three points are in a line. If the last point is in the leftside of line made up of previous two point, these three poins is make a counter-clockwise turn. These parts implements in a function isCCW(ptA, ptB, ptC).

First to implement in this algorithm is to find the rightmost lowest point p0 as the same as above. Then sorted the points by angle generated by a horizontal line through p0 and the line from p0 to each point which also needs to use function theta(ptA, ptB). Use the methods in lecture, remove the points which is not counter-clockwise. All the left points in list is vertices of convex hull.

- **Monotone Chain Algorithm**

After searching many introdutions on websites, I choose a Monoton Chain Algoritym to find convex hull vertices. It is also used function isCCW(ptA, ptB, ptC) to find upper and lower vertices of convex hull.

In this implemetion, firstly need to sort all points as x-coordinate and then y-coordinate. Then to find the lower bound from leftmost point and add points to list if it is counter-clockwise with previous points until to the rightmost point. Start from the rightmost point as the same to find upper bound of convex hull.

## 3. Algorithm Analysis

- **Gift Wrap Algorithm**

The procedure time of Gift Wrap is based on the number of both input and output, but more obviously by number of output.

| Time Computation of Giftwrap | | | | |
|---|---|---|---|---|
| | Set A | | Set B | |
| Number of input | Time(s) | Number of output | Time(s) | Number of output |
| 3000 | 0.008 | 3 | 0.058 | 30 |
| 6000 | 0.012 | 3 | 0.023 | 60 |
| 9000 | 0.019 | 3 | 0.512 | 90 |
| 12000 | 0.032 | 4 | 0.916 | 120 |
| 15000 | 0.04 | 4 | 1.423 | 150 |
| 18000 | 0.049 | 4 | 2.071 | 180 |
| 21000 | 0.056 | 4 | 2.783 | 210 |
| 24000 | 0.081 | 5 | 3.653 | 240 |
| 27000 | 0.091 | 5 | 4.619 | 270 |
| 30000 | 0.105 | 5 | 5.668 | 300 |

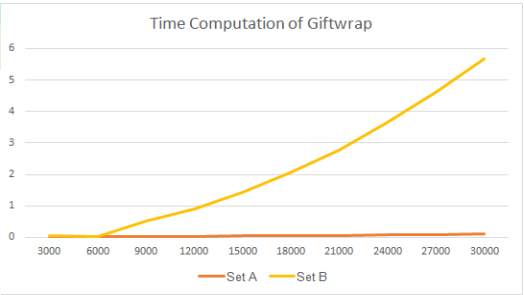


Table and Graph of Time Computation of Giftwrap

As table and graph shown, with longitudinal comparison for number of input with the same number of output points, I discovers number of input increased, the time of Giftwrap Algorithm has a slight growth. Then with horizontal comparison for same number of input vertices, when number of output increases, time of procedure for Gift Wrap Algorithm has a huge growth. Totally it proofs this algorithm is an output

sensitive algorithms, the Time Complexity is O(mn) which m is the number of output points.

- **Graham-Scan Algorithm**

| Time Computation of Graham-scan | | | | |
|---|---|---|---|---|
| | Set A | | Set B | |
| Number of input | Time(s) | Number of output | Time(s) | Number of output |
| 3000 | 0.006 | 3 | 0.007 | 30 |
| 6000 | 0.013 | 3 | 0.013 | 60 |
| 9000 | 0.02 | 3 | 0.02 | 90 |
| 12000 | 0.027 | 4 | 0.027 | 120 |
| 15000 | 0.035 | 4 | 0.034 | 150 |
| 18000 | 0.042 | 4 | 0.041 | 180 |
| 21000 | 0.048 | 4 | 0.049 | 210 |
| 24000 | 0.055 | 5 | 0.055 | 240 |
| 27000 | 0.064 | 5 | 0.062 | 270 |
| 30000 | 0.069 | 5 | 0.069 | 300 |

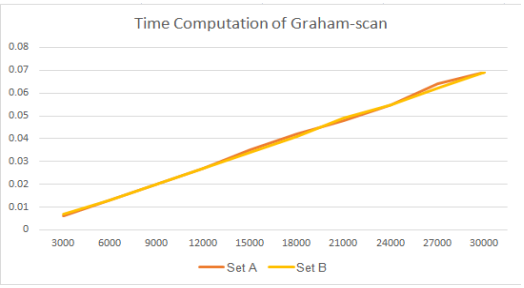


Table and Graph of Time Computation of Giftwrap

As the result in table, I use the same comparision methods with Gift Wrap. There is no obvious influences on number of output points as Gift Wrap. Graham-Scan always performs a stable result when output quantity increses. It is only effected by the number of input points. The complexity of Graham-Scan are twice of O(n) for finding the start point and convex hull vertices, the sorted time is O(nlogn). So the total complexity of this algorithm is O(nlogn) which mentioned in lecture.

- **Monotone Chain Algorithm**

| Time Computation of Graham-scan | | | | |
|---|---|---|---|---|
| | Set A | | Set B | |
| Number of input | Time(s) | Number of output | Time(s) | Number of output |
| 3000 | 0.009 | 3 | 0.009 | 30 |
| 6000 | 0.019 | 3 | 0.018 | 60 |
| 9000 | 0.028 | 3 | 0.027 | 90 |
| 12000 | 0.037 | 4 | 0.037 | 120 |
| 15000 | 0.048 | 4 | 0.048 | 150 |
| 18000 | 0.058 | 4 | 0.057 | 180 |
| 21000 | 0.069 | 4 | 0.067 | 210 |
| 24000 | 0.077 | 5 | 0.076 | 240 |
| 27000 | 0.089 | 5 | 0.086 | 270 |
| 30000 | 0.097 | 5 | 0.1 | 300 |

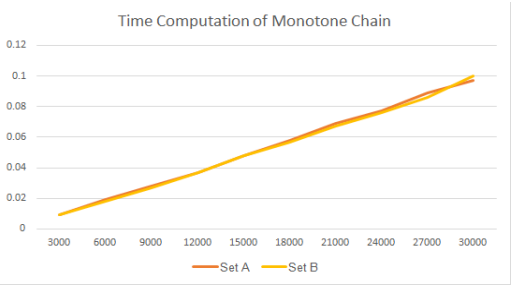


Table and Graph of Time Computation of Giftwrap

As the table shown, I get the same conclusion as Graham-Scan that Monotone Chain is also only effected by number of input points, the complexity of sorted are twice O(nlogn) and O(n) for finding convex hull vertices. Totally the complexity of Monoton Chain is also O(nlogn) totally as the reference mentioned.

- **Cross Contrast of these three algorithms**

| Time Computation of methods for Set A | | | | |
|---|---|---|---|---|
| Number of Vertices | Giftwrap | Graham-scan | Monotone Chain | Number of output |
| 3000 | 0.008 | 0.006 | 0.009 | 3 |
| 6000 | 0.012 | 0.013 | 0.019 | 3 |
| 9000 | 0.019 | 0.02 | 0.028 | 3 |
| 12000 | 0.032 | 0.027 | 0.037 | 4 |
| 15000 | 0.04 | 0.035 | 0.048 | 4 |
| 18000 | 0.049 | 0.042 | 0.058 | 4 |
| 21000 | 0.056 | 0.048 | 0.069 | 4 |
| 24000 | 0.081 | 0.055 | 0.077 | 5 |
| 27000 | 0.091 | 0.064 | 0.089 | 5 |
| 30000 | 0.105 | 0.069 | 0.097 | 5 |

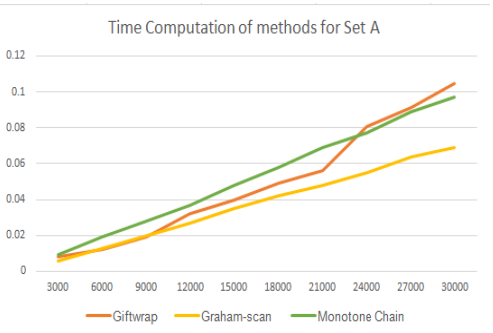

Table and Graph of Time Computation of Set A

From the result of set A shows in tables and graph, the Graham-Scan Algorithm has the best performance amont these three algorithms while the Monotone Chain does the worst. With only a few output points, Gift Wrap performs well as the other two.

| Time Computation of methods for Set B | | | | |
|---|---|---|---|---|
| Number of Vertices | Giftwrap | Graham-scan | Monotone Chain | Number of output |
| 3000 | 0.058 | 0.007 | 0.009 | 30 |
| 6000 | 0.023 | 0.013 | 0.018 | 60 |
| 9000 | 0.512 | 0.02 | 0.027 | 90 |
| 12000 | 0.916 | 0.027 | 0.037 | 120 |
| 15000 | 1.423 | 0.034 | 0.048 | 150 |
| 18000 | 2.071 | 0.041 | 0.057 | 180 |
| 21000 | 2.783 | 0.049 | 0.067 | 210 |
| 24000 | 3.653 | 0.055 | 0.076 | 240 |
| 27000 | 4.619 | 0.062 | 0.086 | 270 |
| 30000 | 5.668 | 0.069 | 0.1 | 300 |

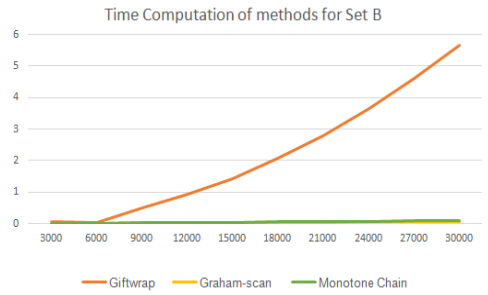


Table and Graph of Time Computation of Set B

The time performance for set B shows that Gift Wrap becomes so much slower than the other two algorithms as the number of output increasing greatly. The differences between Graham-scan and Monotone Chain are quite small as they both have the complexity of O(nlogn). These differences might because Monotone Chain did twice of sorting that might cost a litter longer time than Graham-scan.

## 4. Conclusion and more

To conclude, the Gift Wrap is a sensitive output algorithm which will have a greater affection by the increasement number of output than the number of input. Graham-Scan and Monotone Chain algorithms will only affected by the number of input. That makes them performed better in real problems.

The assignment I implement did not show other algorithms which have different time complexity from each other. Further more I would do outside the assignment is to try the Quick Hull Algorithms which has a complexity of $O(n^2)$.

## 5. Conclusion and more

Lecture Convex Hull

https://en.wikibooks.org/wiki/Algorithm_Implementation/Geometry/Convex_hull/Monotone_chain

http://www.csie.ntnu.edu.tw/~u91029/ConvexHull.html#4