Report for Assignment 3

Deep Kiran Shroti
Roll Number- 12EC35013

March 9, 2016

Skylines

0.1 Theory

Though the divide-and-conquer algorithm is a powerful technology, but it is difficult to be used in practice. The problems which can be solved by the divide-and-conquer algorithm generally have following characters.

1. When the scale of the problem is reduced to certain extent, it can be easily solved.
2. The problem can be divided into many same problems with small scale, i.e. the problem has the character of the optimal sub-structure.
3. The solutions of the sub-problems of the problem can be combined as the solution of the problem.
4. Various sub-problems divided by this problem are independent each other, i.e. the public problem is not contained among sub-problems.

1 Skyline of Flat Rooftop building

Assignment Statement Given the exact location and shape information of n buildings which lie on a fixed horizontal line in a 2-dimensional city, design an algorithm that reports the skyline of these buildings, eliminating hidden lines.

1.1 Implementation Details

We can find Skyline in *(nLogn)* time using Divide and Conquer. The idea is similar to Merge Sort, divide the given set of buildings in two subsets. Recursively construct skyline for two halves and finally merge the two skylines.

1. **First Step:** The base case would be a single building with a strip of size two, i.e. (start, start_height) and (end ,end_height).
2. **Second Step:** Recursively divide the buildings in two halves.i.e *result1* and *result2*
3. **Third Step:** Height of new Strip is always obtained by taking maximum of following.

    - Current height from *result1* skyline1, say 'h1'.
    - Current height from *result* skyline2, say 'h2'.

h1 and h2 are initialized as 0. h1 is updated when a strip from *result1* i.e skyline1 is added to *result* and h2 is updated when a strip from *result* i.e. SkyLine2 is added.

1.2 Conclusion

In the algorithm of the flat rooftop skyline problem. Time complexity of above recursive implementation is same as Merge Sort.

$$T(n) = T(n/2) + \theta(n)$$

Now doing merge sort in the prepossessing step in the algorithm gives

$$T(n) = 2T(n/2) + \theta(n)$$

Therefore the complexity is

$$\theta(nLogn)$$

2 Skyline of Flat Rooftop building

Assignment Statement Given the exact location and shape information of n buildings which lie on a fixed horizontal line in a 2-dimensional city, design an algorithm that reports the skyline of these buildings, eliminating hidden lines

2.1 Implementation Details

We can find Skyline in *(nLogn)* time using Divide and Conquer. The idea is similar to Merge Sort, divide the given set of buildings in two subsets. Recursively construct skyline for two halves and finally merge the two skylines.

1. **First Step:** The base case would be a single building with a strip of four points, i.e. (start,0),(start, start_height) and (end ,end_height), (end , 0).

2. **Second Step:** Recursively divide the buildings in two halves.i.e *result1* and *result2*

3. **Third Step:** Height of new Strip is always obtained by taking maximum of following.

    - Current calculated height from *result1* skyline1, say 'h1'.
    - Current calculated height from *result* skyline2, say 'h2'.

the lines corresponding to the points l1 and l2 are initialized to 0. A point is updated to the skyline *result*, if it is above the predicted height of skyline2 *result2*.

2.2 Conclusion

In the algorithm of the slant rooftop skyline problem. Time complexity of above recursive implementation is same as Merge Sort.

$$T(n) = T(n/2) + \theta(n)$$

Now doing merge sort in the prepossessing step in the algorithm gives

$$T(n) = 2T(n/2) + \theta(n)$$

Therefore the complexity is

$$\theta(nLogn)$$