CS 381

HW 7

Tian Qiu

1.

(a)

1. Construct current residual network // O (E)
2. Use Edmonds Karp algorithm to construct max flow and final residual network // O(V)
3. Do DFS on residual network, make reachable vertices from source // O(V+E)
4. A minimum cut consists of all edges from reachable vertex to non-reachable vertex.

// O (E)

Time: Edmonds Karp algorithm dominates the time.

O(V)

(b)

1. Construct current residual network // O (E)
2. Use Edmonds Karp algorithm to construct max flow and final residual network // O(V)
3. Do DFS on residual network, make reachable vertices from source // O(V+E)
4. A minimum cut C consists of all edges from reachable vertex to non-reachable vertex.

// O (E)

1. For all edges () in G // O (E)

If () does not exist in residual network && minimum cut C

// New saturated path from source to sink

Print first minimum cut which includes (), second minimum cut C

Print the only minimum cut C // O(1)

Time: Edmonds Karp algorithm dominates the time.

O(V)

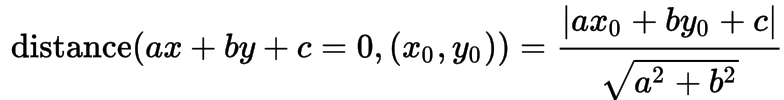
2.

Given:

S: set of n points

Input line: y = mx + b

Distance from point to line:



Pseudocode:

Initial array Distance []

// determine the point is in left or right side of y = mx + b

For each point ( // O(n)

If >

Distance[i] = -1 // left side

If =

Distance[i] = 0

Else

Distance[i] = 1 // right side

For each point (

Distance[i] = Distance[i] \* Distance from point to line ((, y = mx + b)

Find the median k of Distance array // O (n), use the method that introduced in class to // separate array in group of five entries to find

// median of an array in O(n) time

Take line y = mx + n with ( to get the number of n // O (1)

Return y = mx + n

Time: The algorithm to find the median of Distance array takes O (n) time dominating the whole algorithm.

So, time is O(n)