Minqian Liu
Vivian Lo
Jiayi Zhu
Group Name: JVM

CS 170- Design Doc

**1.** We can just generate an MST on G, denoted as T. Then we do the remote operation following a DFS traversal on T. That is, we execute a remote on a node when we confirm that there are no more unvisited children from that node. In other words, say there is an edge u -> v; after all of v's children are visited, we remote the "back edge" v->u. We do this until all bots are remoted back to home vertex h.

**2.** We first construct a new graph G'. At first, it only includes the home vertex h. Then, we repeatedly find the shortest path p starting at a vertex in G' and ending at a vertex with bots and add all vertices and edges in p until all vertices that have bots are in G' (i.e. G' is updated after each iteration). Then, after we add all the bots in G' and finish updating, we can do the same remote operation like Q1 (after generating the MST).

**3.**

**Idea 1:** We divide all the vertices into groups of at most size k/2. For each vertex in a group, we send two different students to scout. If a vertex has conflicting reports, then we know for sure that one scout is definitely telling the truth. We can then remote from said vertex to an adjacent vertex and see if a bot was moved or not. We keep repeating this process until we find all bots.

**Idea 2:** We let each student scout all the vertices one time. After k iterations, we will receive a distribution of "YES" and "NO" for all vertices. We then greedily pick the vertex with highest number of "YES" (we implement it with a priority queue Q) and remote the edge with minimal weight incident to it. During remoting procedure, we use a list for each student to keep track of the vertices that they report wrong. When there is one student who reports wrongly for |V|/2 times, we know that he's now a perfect reporter for the rest |V|/2 vertices. As long as the perfect student comes out, we immediately send him out to scout the rest |V|/2 vertices. For the vertices he reports as "YES", we directly remote them. For those he predicts wrong, we just remove them from Q. After he finishes scouting, we just continue the procedure.

**4.**

**Idea 1:** We generate a MST on G. Then do the same operation as Q1.

Advantages: Since we do remote operation on all vertices, we can send all bots back to home.

Disadvantages: Since |L| is much smaller than |V|, sometimes no bot is moved during the remote operation. So, the time t may be very large.

**Idea 2:** We find the bots location by using the idea 2 in Q3. We now have all bots' new location. Then, we do the method in Q2 to move all bots to home.

Advantages: The new graph we construct cost less time. If there is perfect expert we can accelerate our algorithm and picking the vertices based on votes will increase the probability.

Disadvantages: Finding the location is not optimal, so it might cost a lot of time.

**5.**

**Idea 1:** The smaller the difference between the number of the vertices with bots and |V| (i.e. more uniformly distributed), the better the solution will be.

**Idea 2:** The smaller the size of each $S_i$ for each student, the better the solution will be. Also, in input where will come out a perfect reporter or every student reports correctly, the solution will be good. If everyone reports wrong or it's impossible to have a perfect reporter, it does badly.