Algorithmic Implementations for Solving BCRP-MNCC and BCRP-MLCC Problems

TEAM 19
Akshit Reddy Gudoor, Sai Hemanth Gantasala, **Sandeep Nadella**, Saquib Siddiqui

Problem Statement

The main goal of the paper [1] is to place the fewest number of relay nodes within a certain budget in the deployment area so that the network formed by the sensor nodes and the relay nodes is connected to a maximum extent. Every placement of the relay nodes is associated with a cost and placing all the relay nodes to make all the sensor nodes connected would result in a huge cost sometimes which is not within the budget. This paper tries to come up with a solution of placing the relay nodes within certain budget **B** and would still be able to achieve a network with a high level of connectedness. The paper defines the term connectedness for a disconnected graph and provides two metrics to measure it. The first metric to measure the connectedness of a disconnected graph involves having a lower number of connected components in a disconnected graph as an indicator of a higher degree of connectedness of the graph. The second metric to measure the connectedness of a disconnected graph is the size of the largest connected components of the graph. A larger size of the largest connected component in a disconnected graph is an indicator of a higher degree of connectedness of the graph.

Problem Formulation and Solution

We construct a graph **G = (V, E)** where **V** are the sensor nodes and **E** is an edge between the two nodes. We then have a communication range **R** which refers to the transmission range of the relay nodes. If the distance between the nodes is greater than **R**, then we can say that the graph constructed may be disconnected. We are also given a budget constraint **B** on the number of relay nodes that can be deployed in the development area. The goal is to create a new graph G' = (V', E') with as much connectedness as possible which can be achieved by either deploying the relay nodes in a fashion that minimizes the number of connected components (**BCRP-MNCC**) or deploying the relay nodes in a fashion that maximizes the size of the largest connected component (**BRCP-MLCC**).

We will follow the heuristic solution for both **BRCP-MNCC** and **BRCP-MLCC** with an arbitrary number of sensor nodes. We will implement algorithm 4 for solving the **BRCP-MNCC** problem and use algorithm 5 for solving the **BRCP-MLCC** problem, both are explained in the implementation section below.

Implementation Details

In the paper referred, the algorithms 4 and 5 propose solutions for Budget Constrained Relay node Placement with Minimum Number of Connected Components (**BCRP-MNCC**) and Budget Constrained Relay node Placement with Maximum size of Largest Connected Component (**BCRP-MLCC**) problems respectively based on Minimum Spanning Tree (**MST**) on sensor nodes. We will be constructing datasets to test our results with the experimental results mentioned in the source paper [1].

Algorithm 4 can be summarized as follows

1. Construct a graph with all sensor nodes as vertices using Graph class [5] and Graph generators [4] of **NetworkX** network simulator.
2. Assign weights to each edge connecting two sensor nodes with value
   w(e) = (length of edge 'e'/R) - 1
   where
   R is the Range of Communication
   w(e) represents the number of relay nodes needed for communication between the two sensor nodes at the end of the edge.
   Refer to *Attributes* section of [5] on how weights can be added in NetworkX.

3. Compute MST on this graph. Refer to [2] and [3] on how the inbuilt function can be used for getting MST which internally uses Kruskal's algorithm.
4. Observe that if
   Length of edge 'e' ≤ R then NO relay node is needed
   Length of edge 'e' > R then we need **w(e)** number of relay nodes for communication
5. Also observe that
   When B_1 represents the Budget for BCRP-MNCC problem
   Number of connected components = 1 if $\sum_{over\ all\ edges\ in\ MST} w(e) \leq B_1$

Number of connected components += 1 for every edge being removed when $\sum_{over\ all\ edges\ in\ MST} w(e) > B_1$

6. When $\sum_{\boldsymbol{over\ all\ edges\ in\ MST}} \boldsymbol{w(e)} > \boldsymbol{B_1}$ we remove an edge from MST having the maximum weight; breaking ties arbitrarily until the sum of edge weights is less than or equal to B_1.
7. The resulting forest is returned.

The heuristic for **BCRP-MLCC** is based on **k-MST** (Minimum Spanning Tree). The idea of **k-MST** is that, given an undirected graph G with non-negative edge cost **C(e)** for edge **e ∈ E(G)** and an integer **k**, the problem is to find the minimum spanning tree in **G** that includes at least **k** vertices. Since **MLCC** deals with maximum size, the heuristic function starts with **k = n** (given number of sensor nodes) and decrements by 1 in each iteration.

Algorithm 5 can be summarized as follows

1. Loop over the number of relay nodes starting with all the nodes and then decrement by 1 until the number of nodes becomes 2.
2. Compute the approximate k-MST using k from the for loop.
3. Assign weights to each edge connecting two sensor nodes with value **w(e) = (length of edge 'e'/R) - 1**
   Where
   R is the Range of Communication
   w(e) represents the number of relay nodes needed for communication between the two sensor nodes at the end of the edge

4. Observe that if
   Length of edge 'e' ≤ R then NO relay node is needed
   Length of edge 'e' > R then we need w(e) number of relay nodes for communication
5. When B_2 represents the Budget for BCRP-MLCC problem
   if $\sum_{\boldsymbol{over\ all\ edges\ in\ k-MST}} \boldsymbol{w(e)} \le \boldsymbol{B_2}$
   We found an **MST** that connects all the sensor nodes in the graph using the relay nodes within the budget **B**, hence, return the resulting forest involving the k-nodes.

6. If not, we decrement the **k** value and compute from step 2 again.
7. The returned result the solution of BCRP-MLCC solution.
8. If we don't find any such **k-MST**, we return arbitrary terminal point as solution.

References

1. *Mazumder, A., Zhou, C., Das, A., & Sen, A. (2016). Budget constrained relay node placement problem for maximal 'connectedness*'. In MILCOM 2016 - 2016 IEEE Military Communications Conference (pp. 849-854). [7795435] Institute of Electrical and Electronics Engineers Inc.. https://doi.org/10.1109/MILCOM.2016.7795435

2. https://networkx.github.io/documentation/networkx-1.10/reference/generated/networkx.algorithms.mst.minimum_spanning_edges.html#networkx.algorithms.mst.minimum_spanning_edges

3. https://networkx.github.io/documentation/networkx-1.10/reference/generated/networkx.algorithms.mst.minimum_spanning_tree.html

4. https://networkx.github.io/documentation/stable/reference/generators.html

5. https://networkx.github.io/documentation/stable/reference/classes/graph.html