

Complex Networks: Quiz #8

Due on Jan 9, 2019

RUOPENG XU
18M38179

Problem 1

Make a program of breadth-first search (BFS).

Answer 1

The BFS made by myself:

```
import networkx as nx

# note the iterated nodes as 1
# push the current node into list2(FIFO)
# -> push its adjacent nodes into list2
# ->dequeue current node

def my_BFS(G,s):
    count = 0
    statement = []
    queue = []
    n = len(G)
    result = []

    # use one list to save the statement
    # initial all nodes as zero (haven't iterate)
    for node in range(0,n):
        statement.append(0)

    #visit the source node
    queue.append(s)
    count = count + 1
    statement[s] = 1

    while(queue!=[]):
        current = queue[0]
        result.append(current)

        #if haven't reach the node, push it into the queue
        if(count-1<n):
            for i in G[count-1]:
                if(statement[i] == 0):
                    queue.append(i)

        # dequeue current node
        queue.remove(current)
        count = count + 1

        #change the statement of nodes in queue
        for v in queue:
            statement[v] = 1
    return result

G = nx.karate_club_graph()
source = 0
result = my_BFS(G,source)
print(result)
```

and the result is:

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 17, 19, 21, 31, 30, 9, 27, 28, 32, 16, 33, 25,
29, 23, 24]
```

Problem 2

Show the final status of distance array and queue after BFS is done from vertex 0 of Karate club network.

Answer 2

After BFS has been done, the queue is empty.

This distance array is:

[0, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 3, 3, 2, 1, 3, 1, 3, 1, 3, 3, 2, 2, 3, 2, 2, 3, 2, 1, 2, 2]

Problem 3

Explain why BFS is not good for networks with varying edge lengths.

Answer 3

In BFS, it will search all of the nearest nodes from the source firstly, and after that expand the nodes connected to the current node. So the depth of searching is small and need a lot of memory to storage the array.

If the edge length is varying, which means the graph is deeper, BFS may not have a good performance because of the huge cost of space. In terms of mathematics, the space complexity of BFS is $O(B^M)$ (*from wiki*), in which B is the max number of branches and M is the max length of the tree. Therefore, with more branches and when the graph is deeper, the space cost will increase.