Breadth First Search in Graph and its Applications in Genome Assembly

GUMMADI PAVANI
Dept. of Computer Science and Engineering (AI)
Amrita Vishwa Vidyapeetham
Amritapuri, Kerela
amenu4aie20033@am.students.amrita.edu

KONIJETI SRI VYSHNAVI
Dept. of Computer Science and Engineering (AI)
Amrita Vishwa VidyaPeetham
Amritapuri, Kerela
amenu4aie20042@am.students.amrita.edu

METHUKU SAMHITHA
Dept. of Computer Science and Engineering (AI)
Amrita Vishwa VidyaPeetham
Amritapuri, Kerela
amenu4aie20049@am.students.amrita.edu

SREYA SUNIL KURUP
Dept. of Computer Science and Engineering (AI)
Amrita Vishwa Vidyapeetham
Amritapuri, Kerela
amenu4aie20068@am.students.amrita.edu

***Abstract*—Genome sequencing is the technique of determining the sequences of nucleotide bases in an organism's genome. BFS is a fundamental technique for exploring the edges and vertices of a graph and is used in a variety of real-world applications. In this project we will be understanding BFS and its various applications and then be discussing and implementing one of the application of Breath First Search in genome assembly that is Breath First Search-based path-finding algorithm.**

***Index Terms*—- BFS, Genome sequencing, path finding algorithm, nucleotide**

I. Introduction

A. Genome Sequencing

The amount of genomic data available as reads generated by various genome sequencers has exploded in recent years. The number of reads produced can be enormous, ranging from hundreds to billions of nucleotides, each of which is different in size. For both biomedical and data scientists, assembling such a vast amount of data is one of the most difficult computing issues.

Genome sequence information is currently crucial in a variety of fields, including medical diagnostics, molecular biology, forensic biology, biotechnology, and biological research. Genome sequences are obtained by several methodologies and technologies that are all called genome or DNA sequencing. Genome sequencing is the technique of determining the sequences of nucleotide bases (Adenine (A), Cytosine (C), Thymine (T), and Guanine (G)) in an organism's genome.

B. BFS

BFS (Breadth-First Search) is a fundamental technique for exploring the edges and vertices of a graph and is used in a variety of real-world applications. It has an O(V+E) complexity, where O, V, and E stand for Big O, vertices, and edges, respectively. This mechanism serves as a key component in various applications. As the name implies, the traversal in BFS visits the breadth before the depth. There are various applications of BFS like to find BFS-based path, to find all neighbour nodes in peer-to-peer networks like BitTorrent, to construct indexes by search engine crawlers (it finds all links in the original page to get new pages, starting with the source page), to find nearby locations using a GPS navigation system, to find maximum flow in a network in the Ford-Fulkerson algorithm and in networking for broadcasting packets.

C. Path finding

A pathfinding method, at its most basic level, searches a graph by starting at one vertex and exploring adjacent nodes until the destination node is reached, with the goal of finding the cheapest route. There are many methods and algorithms for the said pathfinding but in this project, we will be implementing using BFS. Although a breadth-first search would find a route if given enough time (that means it is time-consuming), other methods that "explore" the graph would arrive at the destination sooner.

In this project we will be implementing and discussing the path finding for genome assembly using BFS.

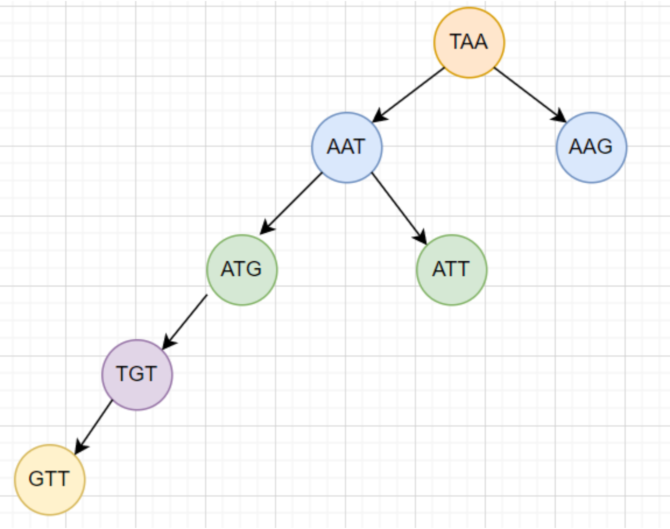
II. Literature Review

The amount of genomic data is rising at an exponential rate, yet representations are frequently unclear. Various ontologies for structurally and semantically defining data have been released to explicitly specify experimental details in order to give better comprehension and quality checking, as well as facilitate reuse, preproduction, and integration of the data. In modern IT, graphs have swiftly become one of the most essential data structures, such as in social media, where a large number of people are modelled as vertices and their social relationships as edges, and then evaluated collectively to implement numerous advanced services. Another example is modelling biophysical structures and phenomena, such as synaptic connections in the brain or protein-enzyme interaction networks, in order to diagnose diseases in the future. The huge scale and complexity of such current graph applications, with billions of edges and trillions of vertices, necessitates not just massive storage but also tremendous processing capacity to analyse them.

Breadth first search is a graph traversal algorithm that starts at the root node and traverses the graph to all of its neighbours. Then it chooses the closest node and explores all the nodes that have yet to be explored. The algorithm repeats the process for each closest node until it reaches the target.The Breadth First Search investigates each node once and places it in a queue, after which it removes nodes from the queue and investigates its neighbours. BFS has multiple advantages like; using GPS navigation system BFS is used to find neighboring places, in networking when we want to broadcast some packets we use the BFS algorithm and path finding algorithm is based on BFS or DFS. The vertex cover problem is a classical graph optimization problem which remains intractable even for cubic graphs and planar graphs with maximum degree at most three. In the paper A Breadth First Search Approach For Minimum Vertex Cover of Grid Graphs, the exact solution of the vertex cover problem is obtained for grid graphs. The paper introduced the breadth first search tree (bfs) technique to find the minimum vertex cover and the inverse vertex cover sets of grid graphs and prove that grids are invertible graphs. It also studies the relationship between strong cover and minimum vertex cover. Structure enumeration is a problem of generating all non-redundant chemical compounds based on a given constraint, such as a chemical formula. It is important in chemoinformatics since it appears as a subproblem of several critical problems, such as drug discovery and structure elucidation. Breadth-first Search Based Approach to Enumerating Chemical Compounds Containing Outerplanar Fused Benzene Ring Substructures was investigated by Jina Jindalertudomdee. In ”Efficient construction and its application for compressioninary” by Yuansheng Liu and Jinyan Li, Breadth First Search graphs have been widely adopted for the de novo assembly of genomic short reads. This work studies another important problem in the field: how graphs can be used for high-performance compression of the large-scale sequencing data.

III. DATA DESCRIPTION

INPUT : The graph which we implemented for the code is:



Input data

All the nodes in the data are 3-mers, our goal is to find out the correct path for genome assembly through breadth first search algorithm. We start with the root node ‘TAA’ which has 2 children. The children are arranged in such a manner that the k-1 string from the end of the parent node overlap with the k-1 string from the beginning of child node. We traverse through every node and find out the efficient path from the graph.

EXPECTED OUTPUT : TAA AAT AAG ATG ATT TGT GTT

IV. METHODS

The algorithm Breadth-first search (BFS) is for searching a tree data structure for a node that satisfies a given property. It starts at the tree root and explores all nodes at the present depth prior to moving on to the nodes at the next depth level. The Breadth-First Search (BFS) is another fundamental search algorithm used to explore nodes and edges of a graph. It runs with a time complexity of 0(V+E) and is often used as a building block in other algorithms. A BFS starts at some arbitrary node of a graph and explores the neighbors. Breadth-First Search (BFS) is used to traverse graphs or trees. Traversing a graph entail visiting every node. A recursive algorithm for searching all the vertices of a network or tree is called Breadth-First Search. BFS and its application in finding connected components of graphs were invented in 1945 by Konrad Zuse. Data structures such as a dictionary and lists can be used to build BFS in Python. The breadth-first search in a tree and a graph is nearly same. The only difference is that the graph might have cycles, allowing us to return to the same node.

V. ALGORITHM

The BFS algorithm starts at the root node and travels through every child node at the current level before moving to the next level. Let's look through the algorithm that Breadth-First uses before learning the python code for it and its output. Breadth-First Search uses a queue data structure to store the node and mark it as "visited" until it marks all the neighboring vertices directly related to it. The queue operates on the First In First Out (FIFO) principle, so the node's neighbors will be viewed in the order in which it inserts them in the node, starting with the node that was inserted first. Consider the Rubik's Cube as an example. The Rubik's Cube is said to be looking for a way to turn it from a jumble of hues into a single color. When we compare the Rubik's Cube to a graph, we may say that the cube's potential states correspond to the graph's nodes, and the cube's possible actions correspond to the graph's edges.

A. STEPS OF THE ALGORITHM :

1. Put any of the graph's vertices at the end of the queue to begin.

2. Take the first item in the queue and add it to the list of items that have been visited.

3. Make a list of the nodes that are next to that vertex. Toss individuals who aren't on the visited list to the back of the line.

4. Steps two and three should be repeated until the queue is empty.

Because a graph might sometimes have two separate unconnected portions, we can execute the BFS algorithm at each node to ensure that we've visited every vertex.

B. PSEUDO CODE:

The pseudocode for BFS in python goes as below:

```
create a queue Q
mark v as visited and put v into Q
while Q is non-empty:
        remove the head u of Q
        mark and enqueue all (unvisited) neighbors of u
```

VI. RESULTS

First, we'll design the graph for which we'll utilise the breadth-first search in the code above. After that, we'll make two lists: one to keep track of the graph's visited nodes, and another to keep track of the nodes in the queue.

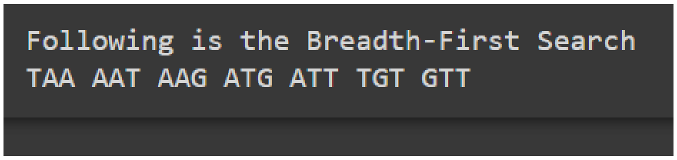
Following the preceding steps, we'll define a function with the arguments visited nodes, the graph itself, and the node. We'll keep adding the visited and queue lists throughout a function.

Then we'll execute the while loop for the queue of nodes to visit, and then we'll delete and print the same node as it's visited.

Finally, we'll use the for loop to check for unvisited nodes before appending them to the visited and queue lists.

We'll call the user to create the bfs function with the first node we want to visit as the driver code.

OBTAINED OUTPUT :



```
Following is the Breadth-First Search
TAA AAT AAG ATG ATT TGT GTT
```

VII. CONCLUSION

In this report, a Breath First Search based path finding algorithm was implemented. Various applications of BFS was discussed in this report, and one of these application was successfully implemented. BFS consumes a significant amount of memory. It has a greater level of time complexity. It has long pathways when all paths to a destination have about the same search depth.Although a breadth first search consumes more memory, it always finds the shortest path first.

REFERENCES

[1] Medvedev, Paul, and Michael Brudno. "Maximum likelihood genome assembly." Journal of computational biology : a journal of computational molecular cell biology vol. 16,8 (2009): 1101-16. doi:10.1089/cmb.2009.0047

[2] Wenyu Shi, Peifeng Ji, Fangqing Zhao, The combination of direct and paired link graphs can boost repetitive genome assembly, Nucleic Acids Research, Volume 45, Issue 6, 7 April 2017, Page e43,

[3] Arnab Chakraborty, Applications of DFS and BFS in Data Structures, Published on 27-Aug-2019 12:25:37

[4] Singh, Rina, Graves, Jeffrey A, Lee, Sangkeun, Sukumar, Sreenivas R, and Shankar, Mallikarjun. Enabling Graph Appliance for Genome Assembly. United States: N. p., 2015. Web.

[5] Kelvin Jose,Graph Theory — Breadth First Search, Towards Data Science, May 9, 2020

[6] DNA Sequencing: Definition, Methods, and Applications, CD Genomics Blog

[7] Mardis E R. DNA sequencing technologies: 2006–2016. Nature protocols, 2017, 12(2): 213

[8] Rina Singh , Jeffrey A. Graves , Sangkeun Lee† Sreenivas R. Sukumar† and Mallikarjun Shankar† , Enabling Graph Appliance for Genome Assembly, Knowledge Discovery Lab, Tennessee Technological University, TN, USA †Computational Sciences and Engineering Division, Oak Ridge National Laboratory, TN,USA

APPENDIX

```
graph =
     'TAA' : ['AAT', 'AAG'],
     'AAT' : ['ATG','ATT'],
      'ATG' : ['TGT'],
      'TGT' : ['GTT'],
      'GTT' : [],
      'ATT' : [],
      'AAG' : []
visited = []
queue = []
def bfs(visited, graph, node):
        visited.append(node)
```

```
    queue.append(node)

    while queue:
        m = queue.pop(0)
        print (m, end = ” ”)

        for neighbour in graph[m]:
            if neighbour not in visited:
                visited.append(neighbour)
                queue.append(neighbour)
print(”Following is the Breadth-First Search”)
bfs(visited, graph, ’TAA’)
```