Application of Greedy Algorithm to Design Power Cable Network

Christopher Clement Andreas / 13515105
Major of Informatics
School of Electrical Engineering and Informatics
Bandung Institute of Technology, 10 Ganesha, Bandung, 40132, Indonesia
clement.andreas97@gmail.com

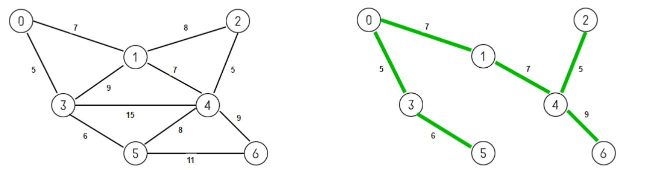
***Abstract*— Electricity is one of the most important innovation that science has given to mankind. It has become a part of modern life and one cannot think of a world without it. Electricity has many uses in our daily life, it is used for lighting rooms, domestic appliances, computer and more. But ironically, most village or region in Indonesia does not have electricity as a part of their daily life. By using Kruskal's algorithm, one can design an efficient cable network design, whether it is "distance"-wise or cost-wise. Even though greedy algorithm usually does not give optimal solution, but in this case Kruskal's algorithm guaranteed an optimal solution to minimum spanning tree.**

Keywords—optimal, Kruskal's algorithm, minimum spanning tree, greedy algorithm, electricity, cable network design.

I. INTRODUCTION

A. Minimum Spanning Tree Problem

A minimum spanning tree is a spanning tree of a connected, undirected graph. It connects all the vertices together with the minimal total weighting for its edges. The minimum spanning forest is a generalization of the minimum spanning tree for unconnected graphs. A minimum spanning forests consists of minimum spanning trees on each of the connected components of the graph.



Picture 1 Example of Graph (Left) and Minimum Spanning Tree (Right).
Source: techiedelight.com

Minimum spanning trees find applications in a range of fields. For example are;

They are considered as an important part of approximation algorithm for NP-hard and NP-complete problems. For example, in most of approximation algorithms, the first step for the Steiner tree problem requires computing the minimum spanning tree. It is also the first step in the Christofede's algorithm for the travelling salesperson problem.

Other example are the application of minimum spanning tree in image processing. For example, image of cells on a slide, then computing the minimum spanning tree of the graph formed by the nuclei can be used to describe the arrangement of the cells.

The next application is minimum spanning tree acts as the basis for single-linkage clustering. These single-linkage clustering is a hierarchical clustering method, in which each element is in its own cluster at the beginning. The clusters then sequentially mix or combined as large clusters. For each step, the clusters separated by the shortest distance combined.

The obvious application of minimum spanning tree is in the construction of power cable, road or telephone networks. This application are the one that being focused in this paper.

Minimum spanning tree have many properties, such as cut property, cycle property, minimum cost subgraph, uniqueness and possible multiplicity.

Possible multiplicity property state that *If there are v vertices in the graph, then each spanning tree has v-1 edges.* There may be several minimum spanning trees of the same weight, in particular if all the edge weights of a given graph are the same, then every minimum spanning tree of the graph is minimum.

The cut property state that *For any cut C in the graph, if the weight of edge e of C is strictly smaller than the weights of all other edges of C, then the edge belongs to all Minimum Spanning Tree of the graph.*

The cycle property state that *For any cycle C in the graph, if the weight of edge e of C is larger than the individual weights of all other edges of C, this the edge don't belong to a Minimum Spanning Tree of the graph.* This being said, for any edge from the Minimum spanning tree being deleted, will break the minimum spanning tree into 2 subtrees. While adding an edge to the minimum spanning trees result the tree to have a cycle.

Uniqueness property state that *If each edge has a distinct weight then there will be only one unique minimum spanning tree*. This is true in many situations, since it's unlikely that any two paths have exactly the same cost.

Last property is the minimum cost subgraph that state *if the weights are positive, then a minimum spanning tree is in fact a minimum cost subgraph connecting all vertices, since* *subgraphs containing cycles necessarily have more total weight.*

As for solving the minimum spanning tree problem, there are many different algorithms to approach the problem depending on the assumption one make:

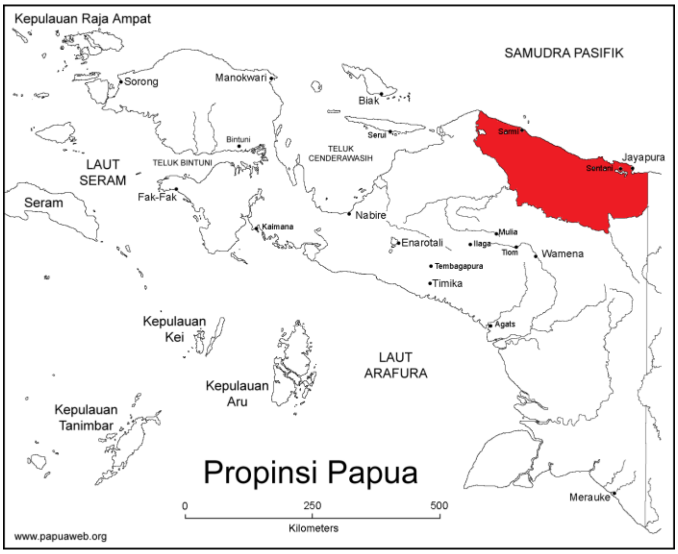
A randomized algorithm can solve it in linear expected time. It can be solved in linear worst case time if the weights are small integers. Otherwise, the best solution is close to linear, but not exactly linear.

B. Power Cable Network Design

Electricity is one of the most basic needs in the current era. It has become a part of modern life that one cannot think of a world without it. Electricity itself has many applications in our daily life, such as lighting the rooms, domestic appliances, computer and more.

According to data World Bank, electricity is available for everyone throughout in most country, yet still most third world country have less than 50% of electricity availability. Ironically, Indonesia starts off at 60.3%, while most of our neighboring country has 100% availability for electricity, such as Singapore, Malaysia and Australia.

Based on viva, in early 2017, there are 2519 area in Indonesia still covered in darkness without electricity. These areas span mostly in eastern region of Indonesia, such as NTT, NTB and Papua. Therefore, it is fundamental to design an efficient power cable network system to accommodate people that still live in darkness so that Indonesia itself can strife for a better future.



Picture 1 District Area of Papua.
Source: papuaweb.org

This paper intents is to virtualize or demonstrate the usage of greedy algorithm through Kruskal's algorithm to solve the current electricity availability in Indonesia. Through this research, the writer hope for the possibilities of a better future for Indonesia.

II. BASIC THEORIES

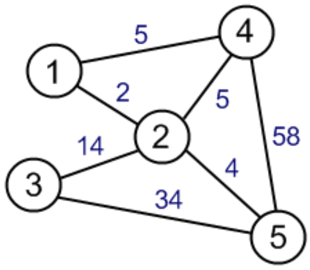
A. Graph

Graph is one form of data structures type that represents object as nodes or vertices and edges that connects 2 vertices together. Graph is defined as G = (V,E), with V is a non-empty set of vertices as { V_1, V_2, V_3, … , V_n}. E is the set of edges as {E_1, E_2, E_3,… , E_n}.

There exists variations of graph, each of them is able to represent certain condition or data.

1. Weighted Graph

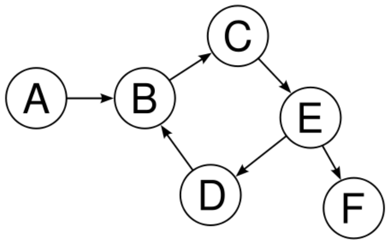
A weighted graph is a variation of graph with a number assigned to every edge called weight. This number may represent distance, cost or any other measurable value.



Picture 2 Example of Weighted Graph.
Source: web.cecs.pdx.edu/Graphs.html

2. Directed Graph

A directed graph is a variation of graph in which every edge has a direction assigned to it. In application, most graphs usually are directed and weighted graph, as this type of graph includes most information needed.



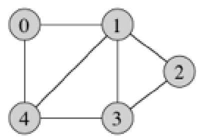
Picture 3 Example of Directed Graph.
Source: mrgeek.me

B. Graph Representation

There are plenty ways to represents a graph, which are adjacency matrix, incidence matrix and adjacency list.

Adjacency matrix can be defined as:

$$A = [a_{ij}],$$
$$a_{ij} = \text{set of 1 and 0.}$$
1 if node i and j adjacent
0 if node i and j not adjacent

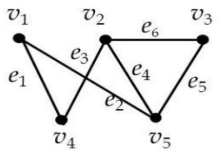


	0	1	2	3	4
0	0	1	0	0	1
1	1	0	1	1	1
2	0	1	0	1	0
3	0	1	1	0	1
4	1	1	0	1	0

Picture 4 Example of Adjacency Matrix.
Source: cs.dartmouth.edu

Incidence matrix can be defined as:

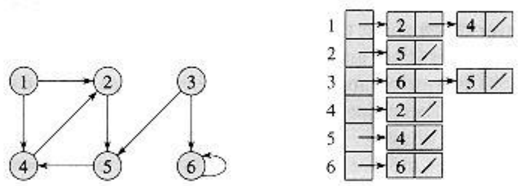
A = [a_{ij}],
a_{ij} = set of 1 and 0.
1 if node i incidence with edge j
0 if node i not incidence edge j



	e_1	e_2	e_3	e_4	e_5	e_6
v_1	1	1	0	0	0	0
v_2	0	0	1	1	0	1
v_3	0	0	0	0	1	1
v_4	1	0	1	0	0	0
v_5	0	1	0	1	1	0

Picture 5 Example of Incidence Matrix.
Source: cse.edu

Adjacency list can be defined as:



Picture 6 Example of Adjacency List.
Source: staff.ustc.edu.cn

C. Greedy Minimum Spanning Tree Algorithm (Kruskal's)

Basically greedy algorithm is an algorithmic paradigm that follows the problem solving heuristic of making the locally optimal choice at each stage with the hope of finding a global optimum.

Greedy algorithm is designed to achieve optimum solution for a given problem. In greedy approach, decisions are made from the given solution domain. As being greedy, the solution closest to optimum is chosen.

Greedy algorithms try to find a localized optimum solution, hoping the solution may eventually lead to globally optimized solutions. However, generally greedy algorithms do not provide globally optimized solutions.

Basically greedy algorithm, as the name suggests, pick the subset of the solution with the target, either minimum or maximum, which probably is the local optimum solution hoping that the solutions may lead to globally optimal solution.

Greedy algorithm have 5 main components or elements, which is set of candidate, set of solution, selection function, feasible function and objective function. Candidate set is the set of candidate solution of the given problem. Solution set is the set of solutions of the given problem. Selection function is the function to select the optimization of candidate. Feasible function is the function to check the feasibility of solution, while objective solution is the optimization function.

In this case, the feasibility function are checking whether the edges that is selected create a cycle or not. While selection function is the function to pick the minimum edges from the set of edges of the graph G, which is automatically selected in the Kruskal's algorithm, because the edges are sorted in an increasingly order at first before the selection begins. Objective function, as the name suggest, ensure that the objective of the algorithm is achieved, which is minimum cost of spanning tree with V-1 amount of edges.

Greedy algorithm can be applied to find a minimum spanning tree for a connected weighted graph. This greedy algorithm which is Kruskal's algorithm, is one of a special form of greedy algorithm, in which Kruskal's algorithm guarantees the optimality of the solution. The algorithm works by finding a subset of edges from the given graph covering every vertex present in the graph, such that they forms a tree with the minimum sum of weights of edges.

The general idea of the algorithm is:

1) Sort the edges in graph G in increasing order of weights.
2) Select the next edge with minimum weight from graph G.
3) Check whether there is a cyle by picking the edge from step 2, if there is no cycle, add the edge to the tree.
4) Repeat step 2 V-1 times. (V = amount of vertices)

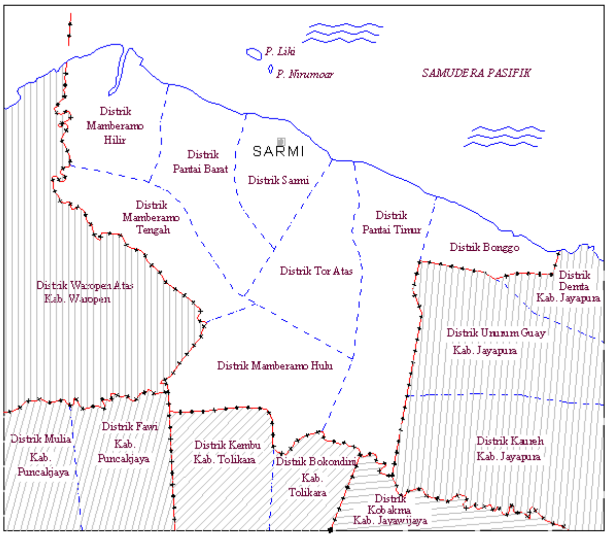
The time complexity provided by the Kruskal's algorithm is O (E log E) with E being the number of edges in the graph. In which the first step of the algorithm, sorting the edges have time complexity of O (E log E). While the rest of the step of the algorithm have the time complexity of O (V log V). From both of the complexity, it is safe to conclude that the time complexity is O (E log E) or O (V log V), which does not make any difference since both E and V are constant numbers so it does not make much of a difference.

III. CASE STUDY

A. *Problem Identification*

In Papua, one of the most eastern region of Indonesia, have almost half of the area covered in darkness without electricity. According to viva, in early 2017, there's still 2519 area in Indonesia without electricity, and almost half of them, 1167 area were in Papua. This lack of facility in some parts or area in Indonesia, is definitely hindering the advancement of Indonesia itself as a country.

These area and village includes Mamberamo Downstream district, West Beach district, Central Mamberamo district, Sarmi district, Mamberamo Upstream district, Tor Atas district, East Beach district and Bonggo District of the Northeast part of Papua, near the border with Papua New Guinea.

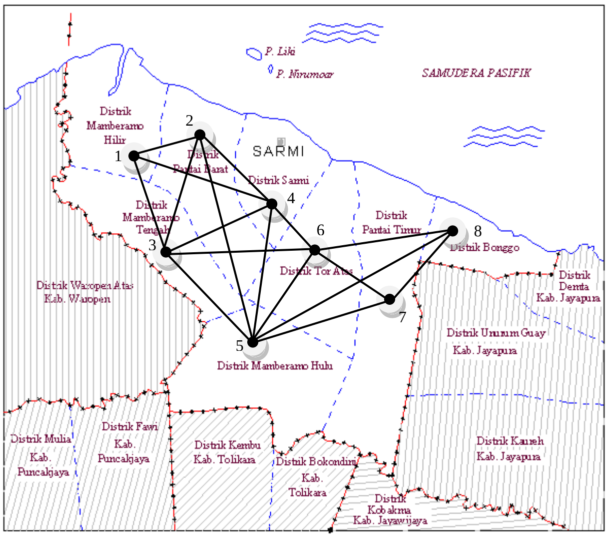


Picture 7 District Area of Northeast Papua.
Source: papuaweb.org

In this paper, the writer use districts in the northeast of Papua as a case study of designing power cable network for electricity.

B. Converting Map to Graph

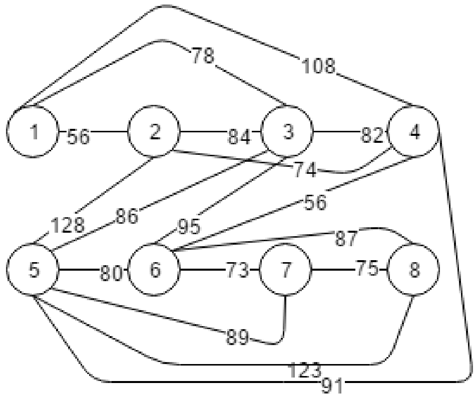
After inspecting the area from certain point of view, even with the lack of field inspection of the districts, an illustration of the map was able to be made. The map below illustrates feasible connections of the districts or area, in which the nodes represents the center of the network in a district.



Picture 8 Edited District Area of Northeast Papua.
Source: papuaweb.org

The nodes in picture 8 represents an estimated position of crowded area of the respective district, with the assumption which each of the respective district needing only a pinpoint area for the power network. The edges that defined in the graph were the network connection in which there is at least a direct path between respective districts.

From the illustration, a weighted graph can be drawn based on the depicted graph.



Picture 9 Weighted Graph of Power Cable Network Design.
Source: writer

From picture 9, each of the edge is bidirected, in which there is no specific direction or orientation. This represents the real situation in which the network is independent of direction.

In this research, the weight of each edge is the closest distance between two vertices, estimated position of the crowded area of a district. The distance or the weight of the edge might not be accurate, since it is measured by kilometers rounded to the nearest 10 or integer, by using the scale on the map.

Other than the graph itself, it can be implemented also as adjacency matrix, where it can be used directly of the algorithm.

District	1	2	3	4	5	6	7	8
1	∞	56	78	108	∞	∞	∞	∞
2	56	∞	84	74	128	∞	∞	∞
3	78	84	∞	82	86	95	∞	∞
4	108	74	82	∞	91	56	∞	∞
5	∞	128	86	91	∞	80	89	123
6	∞	∞	95	56	80	∞	73	87
7	∞	∞	∞	∞	89	73	∞	75
8	∞	∞	∞	∞	123	87	75	∞

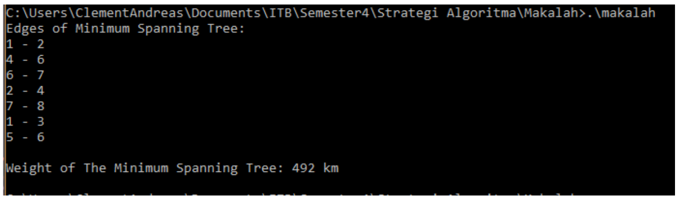
Table 1 Adjacency Matrix of Power Cable Network Design.
Source: writer

IV. SOLVING WITH GREEDY ALGORITM

A. Applying Greedy Algorihm

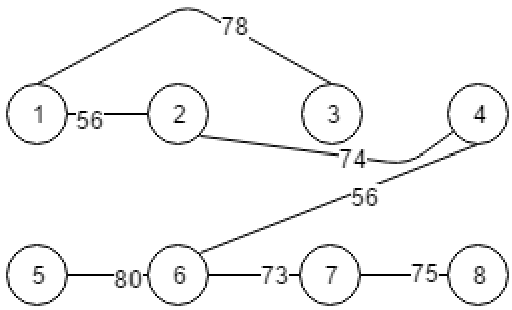
In this research, the matrix is solved by using the Kruskal's algorithm. The general idea of the algorithm is not really that much of a difference from the theory stated above, which starts by sorting the edges based on their weights in an increasing order. After all of the edges are sorted, it starts to select an edge from the sorted edges, and checks whether the selected edge create a cycle in the current tree, if it does not create a cycle, then the selected edge is inserted to the tree, thus ignored if it does create cycle. The process repeated until the tree have V-1 edges. The application that is used to solve the Power Cable Network Design is Kruskal's algoritm approach with object oriented paradigm with C++, which is the modified version of the third task of the IF2211 Algorithm Strategy, the Travelling Salesperson Problem solution.

The application will run greedy algorithm through Kruskal's algorithm as written in Section II.C. User may choose the input, whether it is by inputting the graph with the node itself, or the matrix by reading from a text file. After the application reads user inputs, it will run through the algorithm, then write the minimum spanning tree, by writing the edges with the minimum cost.



Picture 10 Execution Results of The Greedy Algorithm.
Source: writer

Based on the computation of the application, the minimum spanning tree of the power cable network design cost 492 kilometers with altered graph:



Picture 11 Minimum Spanning Tree of Power Cable Network Design.
Source: writer

B. Result Analysis

According to the solution computed by the application, the minimum spanning tree costs 492 km. Even though the distance it spans is the minimum, but there might be many other factor that have to be considered other than the distance itself.

The power cable network design itself only considered the distance covered or spanned by the power cable, it have not considered the terrain the power cable network will be build.

For example, the terrain or the area might not be easily used for building these kind of technology so that the idea of building through that path, might end up with extra costs to deal with the hindrance.

Other than that, alternative solution might include building a path that were not defined before, which may lead to shorter path and lesser cost. Other factors that might lead a better solution is the placement of the vertices or the pinpoint area respective of the district. In this research, the pinpoint area itself are based crowded area speculation, and not from the ideal research from the field directly for example.

Other consideration that might have to be taken to account is the idea of a district might not only need one of those pinpoint area which is assumed in this research where each district only need one pinpoint area for the power network. By doing this, each district might have more than 1 pinpoint area, which adds the amount of vertices in graph thus resulting a different minimum spanning tree, which also resulted in the differences of the power cable network design generated.

The main purpose of this research is basically to produce a power cable network design with minimum "distance" thus cost-wise. But, the cost calculation is only based on the distance itself, so technical part of the design is not taken to account.

V. CONCLUSION

Greedy algorithm is able to compute and produce the best and efficient power cable network design "distance"-wise and hopefully cost-wise. However, with many of other factors that can't be calculated through the algorithm such as road alternatives and terrain that are not quantitative factors, the solution generated is not applicable directly without further field research.

Despite its drawback, greedy or Kruskal's algorithm approach of the problem, might be applicable when the qualitative factors are quantified, and the idea of power cable network building is distance dependent only. By then, a better power cable network design can be generated in "distance"-wise and hopefully cost-wise so that people that have been living in darkness might actually translate to current era and Indonesia might be able to strife for a better future.

ACKNOWLEDGMENT

I would like to thank the Lord Jesus Christ, for His blessing and grace in every step of my life. Praise God for His mercy, that I can finish this paper in the appropriate time as it has been established. I would also like to express my deepest gratitude and appreciation to Dr. Ir. Rinaldi Munir, M.T. as the lecturer of IF2211 Algorithm Strategy for the paper task to the students to explore far more, and also the knowledge and support given up until the finishing of this paper. Lastly, I would also like to express my gratitude to my parents and family for being faithful to support me through their prayers.

REFERENCES

[1] Munir, Rinaldi. *IF2211: Algorithm Strategy* Lecture. Bandung: Major of Informatics School of Electrical Engineering and Informatics Bandung Institute of Technology, 2006.

[2] Robin J. Wilson, Introduction to Graph Theory. Fourth Edition. Harlow: Longman Limited, 1996.

[3] https://www.cs.princeton.edu/13mst/. Access time: 13 May 2017 17.23 GMT+7.

[4] https://www.cs.princeton.edu/courses/archive/fall06/cos341/handouts/graph1.pdf. Access time: 13 May 2017, 17.47 GMT+7.

[5] https://data.worldbank.org/indicator/EG.ELC.ACCS.ZZ. Access time: 13 May 2017, 16.59 GMT+7.

[6] https://cis.upenn.edu/~matuszek/cit594-2014/Lectures/27-greedy.ppt. Access time: 13 May, 18.33 GMT+7.

[7] https://lcm.cas.iisc.ernet.in/dsa/node184.html. Access time: 13 May 2017, 18.57 GMT+7.

[8] https://personal.kent.edu/~rmuhamma/Algorithms/MyAlgorithms/MyAlgorithms/GraphAlgorithm/kruskalAlgorithm.htm. Access time: 13 May 2017, 19.03 GMT+7.

[9] https://cs.dartmouth.edu. Access time: 13 May 2017, 19.21 GMT+7.

[10] https://papuaweb.org. Access time: 13 May 2017, 15.03 GMT+7.

[11] https://cse.edu. Access time: 13 May 2017, 19.34 GMT+7.

[12] https://staff.ustc.edu.cn. Access time: 13 May 2017, 19.35 GMT+7.

[13] https://web.cecs.pdx.edu/Graphs.html. Access time: 13 May 2017, 19.37 GMT+7.

[14] https://mrgeek.me. Access time: 13 May 2017, 19.42 GMT+7.

STATEMENT

I hereby declare that the paper I am writing is my own, not an adaptation, or translation of someone else's paper and not plagiarism.

Bandung, 13 May 2017

Christopher Clement Andreas / 13515105