Journal of Advanced Computing Research Vol. 1, Issue 1 (2016) 60-63

Understanding the Simple Development Process of Kruskal's Minimum Spanning Tree Algorithm

Nurul Intan Qodreah Sanusi #1, Mohamed Faidz Mohamed Said #2

Universiti Teknologi MARA
70300 Seremban Negeri Sembilan, MALAYSIA
1 n.intanqodreah@gmail.com
2 faidzms@ieee.org

***Abstract*—The basic idea of Kruskal's algorithm is to scan involved edge on increasing weight command. Since this algorithm is popular to solve real world problem, many students need to explore this minimum spanning tree (MST) to apply the suitable real-world problem regarding to their studies. The main purpose of this paper is to brief the basic idea of Kruskal's algorithm for fresh researcher or forecaster that are new to Kruskal's MST. In this paper, graph theory will be used to recognize this Kruskal's algorithm. With the basic idea, step to use Kruskal's algorithm is shown in the implementation of this algorithm by using real world popular situation, Chinese Postman Problem.**

Keywords: **Kruskal's algorithm, MST, Prim, HCPP**

I. INTRODUCTION

This algorithm was first introduced in 1956 and published by Joseph Kruskal. It is appeared in a monthly mathematics journal; Proceedings of the American Mathematic Society. Kruskal's algorithm is one of the minimum spanning trees (MST) that it created. The other name for MST is fast greedy algorithm. There are two common algorithms in MST which are Kruskal's Algorithm and Prim's Algorithm. It is standard practice among authors discussing the MST problem to the work of Kruskal's and Prim's as sources of the problem and it is first efficient solution [1].

Kruskal's algorithm is a technique which helps to find a MST from a related weighted graph. There are two best-known algorithms used in finding the minimum spanning tree in a weighted graph [2, 3]. It is a tree which has connected graph and there is no cycle during run this algorithm. In addition, in a graph it will include the spanning tree in which the tree touching all nodes in the graph. Related weighted graph means that every edge has its own value or cost named Weight. Usually, each of the Weight has its own logical meaning depending on the problem.

Many application and problem in real world are solved by using Kruskal's algorithm such as fast multiple histogram computation published by Raoul Berger, S´everine Dubuisson and Christophe Gonzales. The researchers proposed to speed up the computation of the histogram of multiple overlapping nonrotating of a single image [4].

Other application is a Hierarchical Chinese Postman Problem [5], to solve postman difficulty and aim to catch the shortest journey for sending letter. Alternate way to improve complexity of Hierarchical Chinese Postman Problem (HCPP) algorithm is by dropping the number of couples of edges in graph [6]. Therefore, since there is no cycle during run this algorithm, many researchers decide to apply Kruskal's algorithm so that the quantity of edges in the graph can be reduced.

The basic idea of Kruskal's algorithm is to scan involved edge on increasing weight command. Since this algorithm is popular to solve real world problem, many students need to explore this MST by applying the suitable real-world problem regarding to their studies. The main benefit of this Kruskal's algorithm is its flexibility. In theory, any minimum spanning tree problem with additional constraints can be solved using the proposed method [7].

II. OBJECTIVE

The main purpose of this paper is to brief the basic idea of Kruskal's algorithm for fresh researcher or forecaster that are new to Kruskal's MST. Since, there are several greedy algorithms for finding MST [7], fresh researcher or forecaster might be confused with the specific characteristic of Kruskal's algorithm. There are many ways to understand Kruskal's algorithm such as by using graph or pseudocode.

In this paper, graph theory will be used to recognize this Kruskal's algorithm. In addition, Murty's algorithm for ranking assignments in order of increasing cost has been used in a similar fashion to generate an optimal solution to the travelling salesman problem [8]. Therefore, since this algorithm is suitable for some cases in real world, let us illustrate any possible problem so that Kruskal's algorithm can be used. At the end of this paper, reader can understand the basic of this algorithm, can apply Kruskal's algorithm by using weight graph as well as apply this algorithm to the suitable real problem in any situation.

III. CHARACTERIZATION

Kruskal's algorithm is a graph concept to recognize MST for a related weighted graph. The algorithm will find a subset of the boundaries that is generated by a tree that involves each vertex, in which the sum of the weight for all the boundaries in the tree is minimized. If there is any case if the boundaries are not connected, then it is concluded as the graph is to try to find a minimum spanning forest. There are some case studies noted that greedy algorithms find the minimum spanning tree of a graph, which is a tree that connects all the vertices of a graph and the total cost of its edges is minimized [9].

Kruskal's MST algorithm uses the greedy method [10, 11]. The simple procedure of the Kruskal's algorithm is shown below:

step Kruskal:

This program create the MST T for a related *m*-vertex graph K(M,B)

begin

T:=∅;

while |T| < m-1 **do begin**

Select an branch (edge) *b'* of lowest cost from B;

Delete *b'* from B;

if TÈ *b'* does not involved a cycle **then** T:=TÈ*b'*;

end;

end.

The branches are noted in increasing order of the value and involved in the set of *T* in the selected branches if the branches in the *T* do not form a cycle after the probable insertion. The step of choosing a branch is repeated up to *m*-1 branches are included in *T*. Next *T* forms the needed MST of *K*.

IV. GRAPH THEORY

In query optimization, the optimal query execution plan can be described as a MST of the graph that represents the query [9]. In this type of greedy algorithm, using graph is the best way to see the characteristic and the step of Kruskal's algorithm. These are the description of the graph theory in using Kruskal's algorithm.

Let T= set of trees,

Step 1: Form a forest *T*. Each vertex in the graph is distinct tree.
Step 2: Form a set *B* involving the entire branch in the graph.
Step 3: While *B* is non-empty, and *T* is not yet spanning.
Step 4: Remove a branch that have a minimum weight from *B*.
Step 5: In any case, if the branch is relating with two different trees, then add it to the forest. In this case, the two trees that are involved need to be combined and make it into a single tree. Then, discard that edge.

At the end of the step, one component will be forms in the forest and that will create a MST in the graph.

V. IMPLEMENTATION

Since the last decade, MST have become one of the main streams in econophysics to filter the important information contained [12]. In addition, the implication of this is that either greedy algorithm for the MST problem produces an optimal solution [13]. This proves that Kruskal's MST algorithm is very flexible in solving some real-world problems. With the basic idea shown in content 3 and 4, a situation will be created regarding some real problem. Hence, content 3 and 4 will be used in this situation.

The most basic vehicle routing problems are the traveling salesman problem [5, 6, 8, 14-16]. The Chinese mathematician Kwan Mei-Ko first posed the Chinese Postman Problem (CPP) stated the problem as: 'A mailman has to cover his assigned segment before returning to the post office. The problem is to find the shortest walking distance for the postman' [6].

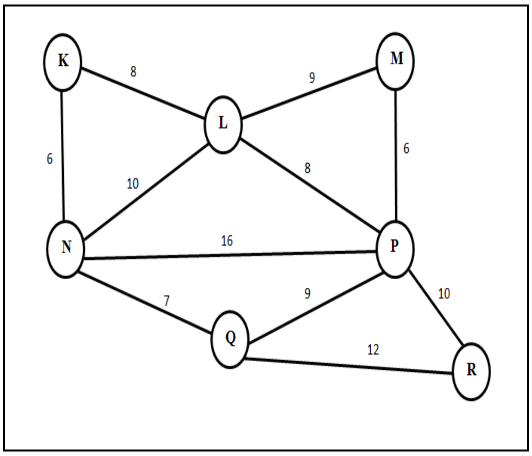

Figure 1. Illustration for the CPP

Figure 1 shows the illustration for the mailman traveling to sending letter. There are 11 branches are in the heap and every vertex is out-of-the-way. Then the minimum branches K to N and M to P.

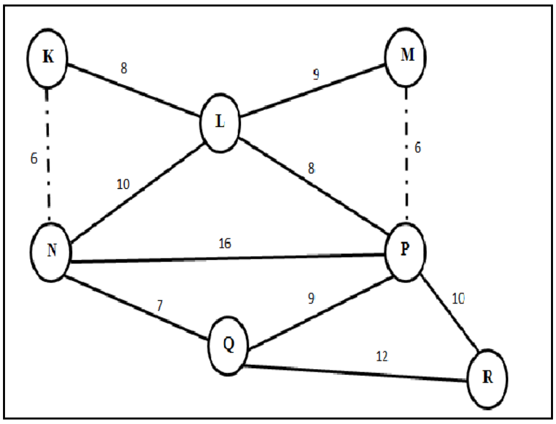

Figure 2. The minimum branches are chosen

Therefore, as in Figure 2 the MSTs are {{K}{L}{M}{N}{P}{Q}{R}}. Next, N to Q and L to P will be chosen.

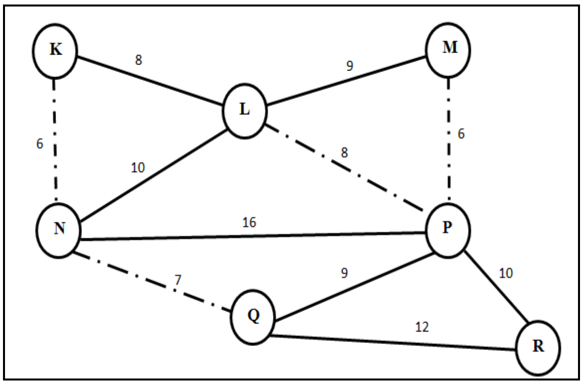

Figure 3. Next minimum branches are selected

According to Figure 3, for N to Q the MSTs are {{K N}{L}{M P}{Q}{R}}. Then for L to P the MST is {{K N Q}{L}{M P}{R}}. Next K to L, this branch is joining 2 trees with L to M.

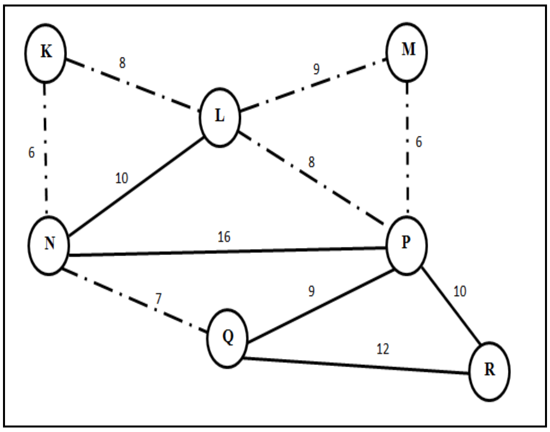

Figure 4. Next selected step

As shown in Figure 4, MST for K to L is {{K N Q}{M P L}{R}}. For L to M the MST is {{K N Q M P L}{R}}. The stage in deciding the MST must follow the lowest-cost-branches rules. Next steps, 3 branches chosen are L to M, P to Q and N to L. However, these ways will cause cycles in the travel. Hence, these steps are ignored.

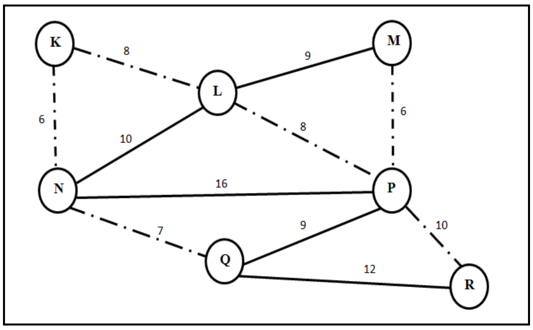

Figure 5. The last step for the situation

Figure 5 shows, the last procedure before stop. It is P to R branch, so the mailman completely goes to all vertices. Although, Q to R and N to P are not selected, the selected step will be ignored.

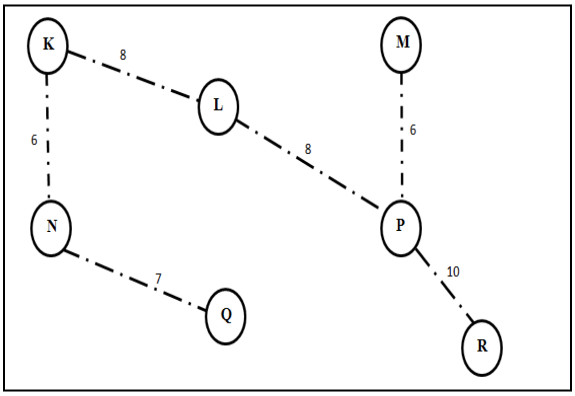

Figure 6. Shortest traveling

Figure 6 shows that the shortest way for the mailman in traveling while doing his work by using Kruskal's algorithm. It will reduce time and cost.

VI. Conclusion

Many application and problem in real world are solved by using Kruskal's algorithm such as fast multiple histogram computation published by Raoul Berger, S´everine Dubuisson and Christophe Gonzales. From this paper, the basic idea of MST by using Kruskal's algorithm is demonstrated. It is easy to new researcher to refer to this step of Kruskal's algorithm. In addition, it is suitable for educational purpose.With the simple implementation, the procedure in using Kruskal's algorithm is made easier. It is recommended for future work that more implemented situations are shown.

References

[1] RGraham, P., *On the history of minimum spanning tree problem.* IEEE Annals of the History of Computing, 1985. **7**(1): p. 43-57.

[2] Hsu, L.-H., et al., *Finding the most vital edge with respect to minimum spanning tree in weighted graphs.* Information Processing Letters, 1991. **39**(5): p. 277-281.

[3] Lorenzo, L. and S. Lorenzo-Freire, *A characterization of Kruskal sharing rules for minimum cost spanning tree problems.* International Journal of Game Theory, 2008. **38**(1): p. 107-126.

[4] Berger, R., S. Dubuisson, and C. Gonzales. *Fast multiple histogram computation using Kruskal's algorithm*. in *2012 19th IEEE International Conference on Image Processing*. 2012.

[5] Sayata, U.B. and N.P. Desai. *An algorithm for Hierarchical Chinese postman problem using minimum spanning tree approach based on Kruskal's algorithm*. in *Advance Computing Conference (IACC), 2015 IEEE International*. 2015.

[6] Korteweg, P., *Postman Problems, Priorities and the Concept of Servicing.* Salamanca, 2002.

[7] Sörensen, K. and G.K. Janssens, *An algorithm to generate all spanning trees of a graph in order of increasing cost.* Pesquisa Operacional, 2005. **25**: p. 219-229.

[8] Panayiotopoulos, J.-C., *Probabilistic analysis of solving the assignment problem for the traveling salesman problem.* European Journal of Operational Research, 1982. **9**(1): p. 77-82.

[9] Guttoski, P.B., M.S. Sunye, and F. Silva. *Kruskal's Algorithm for Query Tree Optimization*. in *Database Engineering and*

Applications Symposium, 2007. IDEAS 2007. 11th International. 2007.

[10] Katajainen, J. and O. Nevalainen, *An alternative for the implementation of Kruskal's minimal spanning tree algorithm.* Science of Computer Programming, 1983. **3**(2): p. 205-216.

[11] Horowitz, E. and S. Sahni, *Fundamentals of data structures*. 1983, Pitman.

[12] Djauhari, M.A. and S.L. Gan, *Minimal spanning tree problem in stock networks analysis: An efficient algorithm.* Physica A: Statistical Mechanics and its Applications, 2013. **392**(9): p. 2226-2234.

[13] Greenberg, H.J., *Greedy algorithms for minimum spanning tree.* University of Colorado at Denver, 1998.

[14] Greistorfer, P., *A Tabu Scatter Search Metaheuristic for the Arc Routing Problem.* Computers & Industrial Engineering, 2003. **44**(2): p. 249-266.

[15] Kruskal, J.B., *On the shortest spanning subtree of a graph and the traveling salesman problem.* Proceedings of the American Mathematical society, 1956. **7**(1): p. 48-50.

[16] Santos, L., J. Coutinho-Rodrigues, and J.R. Current, *Implementing a multi-vehicle multi-route spatial decision support system for efficient trash collection in Portugal.* Transportation Research Part A: Policy and Practice, 2008. **42**(6): p. 922-934.