

Mathematical Estimation for Maximum Flow in Electricity Distribution Network by Ford-Fulkerson Iteration Algorithm

Myint Than Kyi*, San San Maw#, Lin Lin Naing**

* Faculty of Computing, University of Computer Studies, Pinlon, Southern Shan State, Myanmar

#Department of Engineering Mathematics, Mandalay Technological University, Mandalay Division, Myanmar.

**Faculty of Computing, University of Computer Studies, Hinthada, Ayeyarwady Division, Myanmar

DOI: 10.29322/IJSRP.9.08.2019.p9229

<http://dx.doi.org/10.29322/IJSRP.9.08.2019.p9229>

Abstract- In this paper, application of the graph theory to handle the function of power flow management will be introduced. Well known Ford-Fulkerson algorithm in graph theory is used to calculate the maximum flow in electricity distribution network. The maximum flow problem is one of the most fundamental problems in network flow theory and has been investigated extensively. The Ford-Fulkerson algorithm is a simple algorithm to solve the maximum flow problem and based on the idea of searching augmenting path from a started source node to a target sink node. It is one of the most widely used algorithms in optimization of flow networks and various computer applications. The implementations for the detail steps of algorithm will be illustrated by considering the maximum flow of proposed electricity distribution network in Kyaukse District, Mandalay Division, Myanmar as a case study. The goal of this paper is to find the maximum possible flow from the source node s to the target node t through a given proposed network.

Index Terms: Flow network, Ford-Fulkerson algorithm, Graph Theory, Maximum flow, Electricity distribution network.

1. INTRODUCTION

The maximum flow problem is a well-studied and widely applicable problem in algorithmic graph theory and optimization [1]. In many fields of applications, graphs theory [2] plays vital role for various modelling problems in real world such as travelling, transportation, traffic control, communications, and various computer applications and so on [3]. Graph is a mathematical representation of a network and it describes the relationship among a finite number of points connected by lines. The maximum flow problem is a type of network optimization problem in the flow graph theory [4]. Electric power distribution is the final stage in the delivery of electric power; it carries electricity from the transmission system to individual consumers. Distribution substations connect to the transmission system and lower the transmission voltage to medium voltage ranging between 2 kV and 35 kV with the use of transformers. A flow network is a directed graph where each edge has a capacity and receives a flow as weighted values [5]. In practice, the flow problems can be seen in finding the maximum flows of currents in an electrical

circuit, water in pipes, cars on roads, people in a public transportation system, goods from a producer to consumers, and traffic in a computer network, and so on.

II. RESOURCES AND METHOD

Network flow problems are central problems in operations research, computer science, and engineering and they arise in many real world applications. A simple, connected, weighted, digraph G is called a transport network if the weighted value associated with every directed edge in G is a non-negative number. This number represents the capacity of the edge and it is denoted by c_{ij} for the directed edge from node i to node j in G [6]. We consider a network of electricity distribution system and the case of network of cables having values allowing flows only in one direction. This type of network is represented by weighted connected digraph in which transformers are represented by vertices and lines which given electricity flows through by edges and capacities by weights. To provide the maximum flow from source vertex to sink vertex is one of the most important things in all transmission network. Source vertex is produced flows along the edges of the digraph G and sink vertex is received [2]. The residual graph can be defined by providing a systematic way in order to find the maximum flow. The residual capacity of an edge can be obtained by using the formula $cf_{ij} = c_{ij} - f_{ij}$ and a residual network can be defined by giving the amount of available capacity.

A. Problem Definition

The problem is given by a digraph G and an associated set of flow constraints on the edges of the graph, to find the maximum amount of flow that may travel between a source node s and a sink node t . The maximum flow problems involve finding a possible maximum flow through a single-source to single-sink flow network. The maximum flow problem can be seen as a special case of more complex network flow problems, such as the circulation problem [7]. A flow in a network is a collection of chain flows which has the property that the sum of the number of all chain flows that contain any edge is no greater than the capacity of that edge [8]. The goal is to provide as much flow as possible from s to t in the graph. The flow in any edge is never exceeded its capacity, and for any vertex except for s and t , the flow in to the vertex must equal the flow out from the

vertex. On any edge we have $0 \leq f_{ij} \leq c_{ij}$. This is called capacity constraint or edge condition.

B. Ford-Fulkerson Algorithm for Maximum Flow

In 1955, Ford, L. R. Jr. and Fulkerson, D. R. created the Ford-Fulkerson Algorithm [8]. This algorithm starts from the initial flow and recursively constructs a sequence of flow of increasing value and terminates with a maximum flow [9]. The idea behind the algorithm is simple. We send flow along one of the paths from source to sink in a graph with available capacity on all edges in the path that is called an augmenting path and then we find another path, and so on [7].

1). Steps of the Algorithm

The Ford-Fulkerson algorithm has two main steps. The first step is a labelling process that searches for a flow augmenting path and the second step is to change the flow accordingly. Otherwise, no augmenting path exists, and then we get the maximum flow [5]. The detail steps of the algorithm are stated as follows [2]:

- Step 1. Assign an initial flow f_{ij} , compute f .
- Step 2. Label s by \emptyset . Mark the other vertices “unlabelled.”

Step 3. Find a labelled vertex i that has not yet been scanned. Scan i as follows. For every unlabelled adjacent vertex j , if $c_{ij} > f_{ij}$ compute

$$\Delta_{ij} = c_{ij} - f_{ij}$$

$$\text{and } \Delta_j = \begin{cases} \Delta_{ij} & \text{if } i = 1 \\ \min(\Delta_i, \Delta_{ij}) & \text{if } i > 1 \end{cases}$$

and label j with a “forward label” (i^+, Δ_j) or if $f_{ji} > 0$, compute $\Delta_j = \min(\Delta_i, f_{ji})$ and label j by a “backward label” (i^-, Δ_j) .

If no such j exists then OUTPUT f . Stop.
 Else continue (i.e., go to step 4).

- Step 4. Repeat Step 3 until t is reached.
 [This gives a flow augmenting path $P: s \rightarrow t$.]
 If it is impossible to reach t , then OUTPUT f . Stop.
 Else continue (i.e., go to Step 5).

- Step 5. Backtrack the path P , using the labels.
- Step 6. Using P , augment the existing flow by Δ_t . Set $f = f + \Delta_t$.
- Step 7. Remove all labels from vertices $2, \dots, n$. Go to Step 3.

This is the end of the algorithm. The flowchart for the maximum flow of electricity distribution network is as follow:

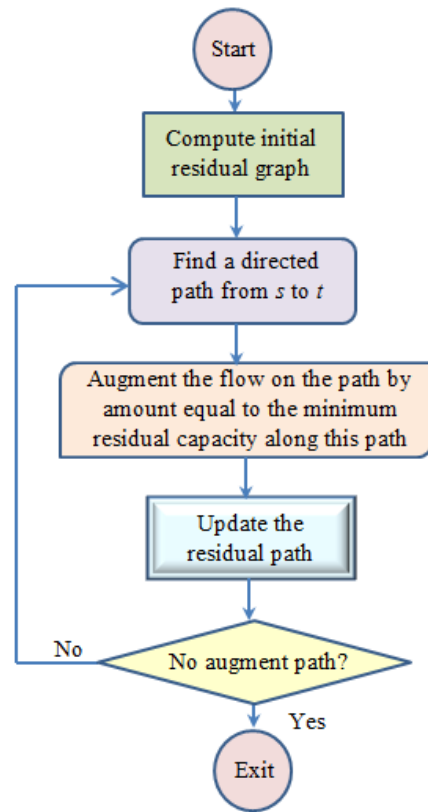


Fig. 1 Flowchart for maximum flow of electricity distribution network

III. RESULTS AND DISCUSSION

In this paper, the application of graph theory to find the maximum flow of electricity distribution network has been illustrated by using the Ford-Fulkerson algorithm [2] to the proposed electricity distribution network of Kyaukse District in Mandalay Division, Myanmar. The electrical flow will not necessarily preserve the capacity constraints that must be preserved in a maximum flow [1]. The weights on the links are referred as capacities and current flows for corresponding edges.

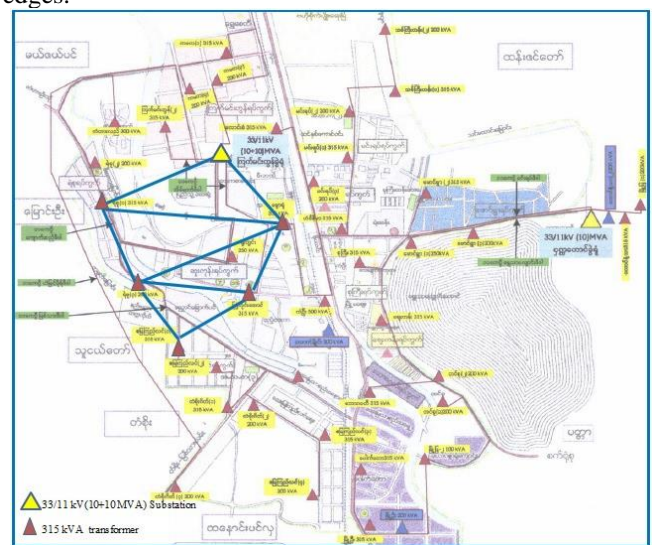


Fig. 2 Proposed network of electricity distribution network (blue) in Kyaukse District, Mandalay Division, Myanmar [10]

The flow network is a directed graph with two distinguished nodes; source and sink. Since the sizes of power cables used

in this network may not be the same, the capacity for each cable may also be different. So it can only maintain a flow of a certain amount of electricity. Anywhere those cables meet; we assumed that the total amount of electricity coming into that junction must be equal to the amount going out. Each edge between two nodes has a non-negative capacity c and receives a flow f where amount of flow on an edge cannot exceed its capacity [4]. The first number on each edge represents capacity and the second number represents current flow of electricity (kV). In this electricity distribution network, we denote the source node 1 as “ s ” and the sink (target) node 6 as “ t ” as shown in Fig. 3.

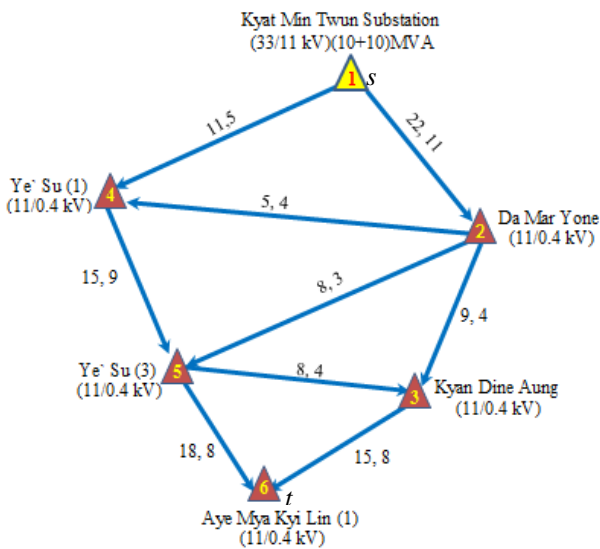


Fig. 3 Representation of digraph for proposed electricity distribution network in Kyaukse District, Mandalay Division, Myanmar

A. Implementation of the Algorithm

The implementation of the Ford-Fulkerson algorithm is illustrated to find the maximum flow for proposed electricity distribution network in Kyaukse District, Mandalay Division from source node s (Kyat Min Twun Substation) to sink node t (Aye Mya Kyi Lin (1) transformer). We can choose any path from source to sink for each iteration step as an augmenting path by using the edge with the non-zero residual capacity in previous step. This means that the edge with maximum flow cannot be used as a segment of augmenting path.

The detail implementation of algorithm can be seen as follows:

Step 1. An initial flow $f = 11 + 5 = 16$ (Given).

Step 2. Label ($s = 1$) by \emptyset . Mark 2, 3, 4, 5, 6 “unlabelled”

Step 3. Scan 1

Compute $\Delta_{12} = 22 - 11 = 11 = \Delta_2$. Label 2 by ($1^+, 11$).

Compute $\Delta_{14} = 11 - 5 = 6 = \Delta_4$. Label 4 by ($1^+, 6$).

Step 4. Scan 2.

Compute $\Delta_{23} = 9 - 4 = 5, \Delta_3 = \min(\Delta_2, \Delta_{23})$

$\Delta_3 = \min(11, 5) = 5$. Label 3 by ($2^+, 5$).

Scan 3.

Compute $\Delta_{36} = 15 - 8 = 7, \Delta_6 = \min(\Delta_3, \Delta_{36})$

$\Delta_6 = \min(5, 7) = 5$. Label 6 by ($3^+, 5$).

Step 5. $P: s = 1 \rightarrow 2 \rightarrow 3 \rightarrow 6 = t$ is a flow augmenting path.

Step 6. $\Delta_t = 5$. Augmentation gives $f_{12} = 16,$

$f_{23} = 9, f_{36} = 13$.

Augmented flow $f = 16 + 5 = 21$.

Step 7. Remove labels on vertices 3, ..., 6. Go to Step 3.

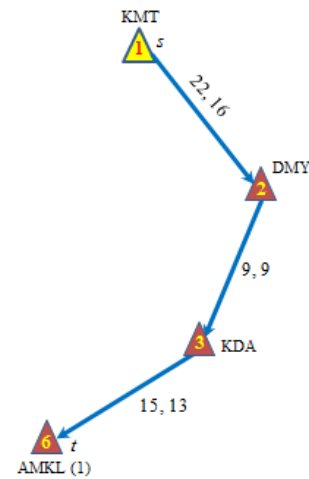


Fig. 4 Flow augmenting path 1-2-3-6

Step 3. Scan 1

Compute $\Delta_{12} = 22 - 16 = 6 = \Delta_2$. Label 2 by ($1^+, 6$)

Compute $\Delta_{14} = 11 - 5 = 6 = \Delta_4$. Label 4 by ($1^+, 6$)

Step 4. Scan 2.

Compute $\Delta_{25} = 8 - 3 = 5, \Delta_5 = \min(\Delta_2, \Delta_{25})$.

$\Delta_5 = \min(6, 5) = 5$. Label 5 by ($2^+, 5$).

Scan 5.

Compute $\Delta_{53} = 8 - 4 = 4, \Delta_3 = \min(\Delta_5, \Delta_{53})$

$\Delta_3 = \min(5, 4) = 4$. Label 3 by ($5^+, 4$).

Scan 3.

Compute $\Delta_{36} = 15 - 13 = 2, \Delta_6 = \min(\Delta_3, \Delta_{36})$.

$\Delta_6 = \min(4, 2) = 2$. Label 6 by ($3^+, 2$).

Step 5. $P: s = 1 \rightarrow 2 \rightarrow 5 \rightarrow 3 \rightarrow 6 = t$ is a flow augmenting path.

Step 6. $\Delta_t = 2$. Augmentation gives $f_{12} = 18, f_{25} = 5,$

$f_{53} = 6, f_{36} = 15$.

Augmented flow $f = 21 + 2 = 23$.

Step 7. Remove labels on vertices 2, ..., 6. Go to Step 3.

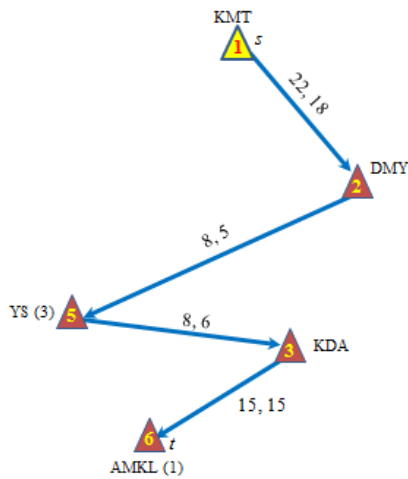


Fig. 5 Flow augmenting path 1-2-5-3-6

Step 3. Scan 1

Compute $\Delta_{12} = 22 - 18 = 4 = \Delta_2$. Label 2 by $(1^+, 4)$.

Compute $\Delta_{14} = 11 - 5 = 6 = \Delta_4$. Label 4 by $(1^+, 6)$.

Step 4. Scan 2.

Compute $\Delta_{25} = 8 - 5 = 3$, $\Delta_5 = \min(\Delta_2, \Delta_{25})$.

$\Delta_5 = \min(4, 3) = 3$. Label 5 by $(2^+, 3)$.

Scan 5.

Compute $\Delta_{56} = 18 - 8 = 10$, $\Delta_6 = \min(\Delta_5, \Delta_{56})$.

$\Delta_6 = \min(3, 10) = 3$. Label 6 by $(5^+, 3)$.

Step 5. $P: s = 1 \rightarrow 2 \rightarrow 5 \rightarrow 6 = t$ is a flow augmenting path.

Step 6. $\Delta_t = 3$. Augmentation gives $f_{12} = 21, f_{25} = 8,$

$f_{56} = 11$.

Augmented flow $f = 23 + 3 = 26$.

Step 7. Remove labels on vertices 2, ..., 6. Go to Step 3.

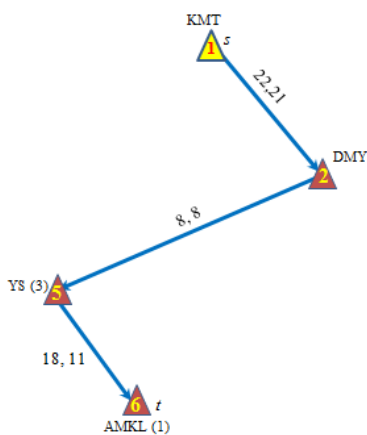


Fig. 6 Flow augmenting path 1-2-5-6

Step 3. Scan 1

Compute $\Delta_{14} = 11 - 5 = 6 = \Delta_4$. Label 4 by $(1^+, 6)$.

Step 4. Scan 4.

Compute $\Delta_{45} = 15 - 9 = 6$, $\Delta_5 = \min(\Delta_4, \Delta_{45})$.

$\Delta_5 = \min(6, 6) = 6$. Label 5 by $(4^+, 6)$.

Scan 5.

Compute $\Delta_{56} = 18 - 11 = 7$, $\Delta_6 = \min(\Delta_5, \Delta_{56})$.

$\Delta_6 = \min(6, 7) = 6$. Label 6 by $(5^+, 6)$.

Step 5. $P: s = 1 \rightarrow 4 \rightarrow 5 \rightarrow 6 = t$ is a flow augmenting path.

Step 6. $\Delta_t = 6$. Augmentation gives $f_{14} = 11, f_{45} = 8,$

$f_{56} = 17$.

Augmented flow $f = 26 + 6 = 32$.

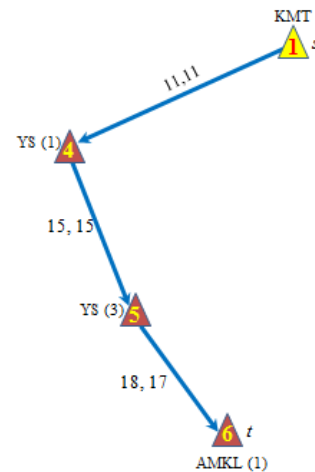


Fig. 7 Flow augmenting path 1-4-5-6

There are no augmenting paths possible from s to t , and then the flow is maximum flow. The maximum flows for all electricity in network have been calculated by iteration process of Ford-Fulkerson algorithm and can be seen in Fig. 8.

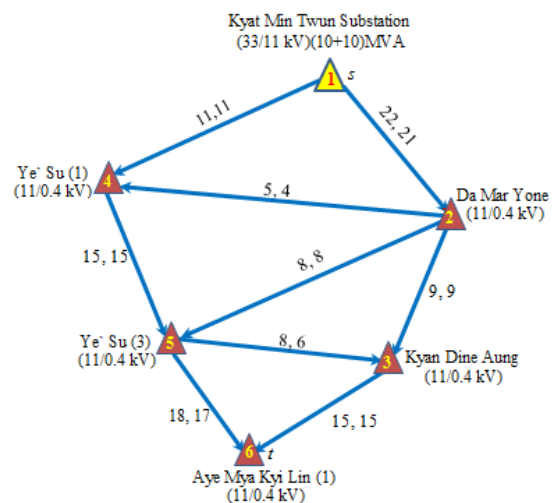


Fig. 8 Maximum flows in electricity distribution network from source node s to sink node t

The maximum flow will be the total flow out of source node which is also equal to total flow in to the sink node. In this case, total flow (maximum flow) out from source node s is $21 + 11 = 32$ and total flow in to the sink node t is

$17+15 = 32$ and hence maximum possible flow for this network is 32 kV.

IV. CONCLUSION

Nowadays, electricity is an important part of homes and industries. Almost whole the devices at homes, businesses and industries are running because of electricity. In this paper, Ford-Fulkerson algorithm has been implemented to find the maximum possible electrical flow in electricity distribution network of Kyaukse District, Mandalay Division, Myanmar. The main idea of the algorithm is to find a path through the graph from the source to the sink, in order to send a flow through this path without exceeding its capacity. Then we find another path, and so on. Maximum flow for proposed electricity distribution network has been developed to implement the algorithm by considering the capacity and flow rate of electricity in the network. The Ford-Fulkerson algorithm is guaranteed to terminate if the edge capacities are nonnegative real values. For this reason, we strongly recommend that Ford-Fulkerson algorithm is an algorithm to get the solution for finding maximum flow with simple procedure [12].

ACKNOWLEDGMENT

The author would like to acknowledge her thank to Dr. Myint Myint Khaing, Principal, University of Computer Studies, Pinlon for her permission to submit this paper in this journals. And Dr. Su Su Win, Associate Professor and Head, Faculty of Computing, University of Computer Studies, Pinlon, for her proofreading and recommendation to submit this research paper. The author would like to express her deeply gratitude to Dr. San San Maw, Professor, Department of Engineering Mathematics, Mandalay Technological University for her active encouragement, and perfect supervision throughout her research work and Dr. Lin Lin Naing, Professor and Head, Faculty of Computing, University of Computer Studies, Hinthada, Ayeyarwady Division, Myanmar, for his valuable ideas, advices and comments. Her special thanks are due to U Thein Hlaing Oo and U Win Naung Lin, Electricity Managers, Kyaukse District Electricity Manager Office under the Ministry of Electricity and Energy, for their help to collect the required data and information to her research. Finally, the author's thanks are to her beloved parents, brothers and sister because her research cannot be done successfully unless they support it.

REFERENCES

- [1] Cameron Musco, "Fast Approximation of Maximum Flow using Electrical Flows", CNMSpringThesis, April 18, 2011.
- [2] Erwin Kreyszig, *Advanced Engineering Mathematics*, 10th edition. Copyright © 2011 John Wiley & Sons, Inc. All rights reserved.
- [3] Meysam Effati, "Analysis of Uncertainty Considerations in Path Finding Applications". 5thSASTech 2011, Khavaran Higher-education Institute, Mashhad, Iran. May 12-14. [Online] Available: <http://5thsastech.khi.ac.ir/data1/civil/1%20%2812%29.pdf>.
- [4] Ola M. Surakhi, Mohammad Qatawneh, Hussein A. al Ofeishat, "A Parallel Genetic Algorithm for Maximum Flow Problem," (*IJACSA International Journal of Advanced Computer Science and Applications*, Vol. 8, No. 6, 2017. [Online] Available: http://thesai.org/Downloads/Volume8No6/Paper_20-A_Parallel_Genetic_Algorithm_for_Maximum.pdf.
- [5] "Electric Power Distribution". [Online] Available, https://en.wikipedia.org/wiki/Electric_power_distribution.
- [6] Santanu Saha Ray, *Graph Theory with Algorithms and its Applications In Applied Science and Technology*, Copyright Springer India. 2013. ISBN 978-81-322-0750-4.
- [7] Zhipeng Jiang¹, Xiaodong Hu¹, and Suixiang Gao. "A Parallel Ford-Fulkerson Algorithm For Maximum Flow Problem". Chinese Academy of Sciences, Beijing, China.
- [8] L. R. Ford, JR. and D.R. Fulkerson, *Maximal Flow Through a Network*. Rand Corporation, Santa Monica, California. September 20, 1955.
- [9] M. Shokry, "New Operators on Ford-Fulkerson Algorithm". *IOSR Journal of Mathematics (IOSR-JM)*. Vol. 11, Issue 2. Mar-Apr. 2015. pp. 58-67.
- [10] Thein Hlaing Oo and Win Naung Lin, "Kyaukse Lines and Location Maps" Kyaukse District Electricity Manager Office under the Ministry of Electricity and Energy, 2018.
- [11] Markus Josefsson, Martin Mützell, "Max Flow Algorithms, Comparison in regards to practical running time on different types of randomized flow networks". Degree Project in Computer Science, DD143X. Sweden, Stockholm 2015.
- [12] Myint Than Kyi, Lin Lin Naing, "Application of Ford-Fulkerson Algorithm to Maximum Flow in Water Distribution Pipeline Network", *International Journal of Scientific and Research Publications*, Volume 8, Issue 12, December 2018. [Online]. Available: <http://www.ijsrp.org/research-paper-1218/ijsrp-p8441.pdf>.

AUTHORS

First Author – Myint Than Kyi, Lecturer, Faculty of Computing, University of Computer Studies, Pinlon. Southern Shan State, Myanmar.

Email address: myinthankyi.tum@gmail.com

Second Author – Dr. San San Maw, Professor, Department of Engineering Mathematics, Mandalay Technological University, Mandalay, Myanmar.

Email address: dr.sansanmaw2017@gmail.com

Third Author – Dr. Lin Lin Naing, Professor and Head, Faculty of Computing, University of Computer Studies, Hinthada, Ayeyarwady Division, Myanmar.

Email Address: linlinnaing@ucsh.edu.mm