Baltic J. Modern Computing, Vol. 3 (2015), No. 2, pp. 73–91

Returning to Strong Components for Identification of Nodes Grouping in the LAN

Abdul HAMEED[1], Adnan N. MIAN[2]

[1] Department of Computer Science, National University of Computer & Emerging Sciences, Lahore, Pakistan
[2] Department of Computer Science, Information Technology University, Arfa Software Technology Park, Lahore, Pakistan

hameed.abdul@nu.edu.pk, adnan.noor@itu.edu.pk

Abstract. Traffic localization and broadcast containment is among the main objective of VLANs implementation. To maximize this objective, an efficient VLAN topology should be designed that maximizes traffic within and minimizes traffic flows between VLANs. In this paper we use strong component technique with statistically calculated traffic threshold to identify nodes grouping in the LAN. Such information helps network administrator in deciding an efficient VLAN topology for the LAN. Our experimental results show that only Mean, 5%Trimmed Mean and DWM as a measures for traffic threshold works well with strong component technique with the observation that no single measure is best suited for all types of network scenarios. A tradeoff between time and performance is achieved by applying the three measures simultaneously and using the best result found that costs slight more execution time of the order of milliseconds. The comparison results of the proposed approach with existing VLAN partitioning algorithms confirm its usefulness.

Keywords: VLANs, partitioning, strong components, central tendency, cloud energy conservation

1 Introduction

VLAN technology is used for years in LANs to logically segment the LAN and to divide a large network into more manageable units. These units called Virtual LANs (VLANs) are group of nodes identifiers like MAC addresses, IP address or user specific names (IEEEStandard, 2011). VLANs are used for separating hosts from others, for instance for security purposes, Quality of Service(QoS), creating small manageable segments for improving overall performance, profiling applications like VoIP call recording, etc. The explicit separation is defined by some organizational policy. This separation between devices is logical and provided by the participating network switches through maintaining different forwarding tables for each VLAN. This logical separation allow VLAN

members to resides in different physical locations of the network while still considered part of the same group.

Network switches are configured by the operators to maintain information about node's or user's mapping into VLANs. The VLAN topology is only known to the participating network switches. User devices are unaware of the VLAN implementation in the network. A user device transmit a normal Ethernet frame which is tagged with the VLAN identifier of the source device by the first network switch that receive the frame. This information (VLAN ID) is extracted from the mapping table stored in the switch which map MAC address to VLAN IDs. The forwarding logic of the switch then define to which port the frame should be forwarded. A broadcast frame is forwarded to all those ports where some of the VLAN members are residing . If a frame is addressed for a device in another VLAN, it is passed towards a router where it is routed to the destination VLAN in the same manner as a network layer packet is routed towards the destination network. Though there may be many applications of VLANs that need to consider many different objectives but in this study we restrict to only one objective, that is, minimizing inter-VLAN or maximizing intra-VLAN traffic. There are some situations that just require such an application of VLANs. For instance, a large crowded software games exhibition where visitors are allowed to play different network games, each game's group produce large amount of network traffic. In such a case, the only objective is to optimize the gaming experience by putting users of each game in the same VLAN. The result is that maximum network traffic of each game's group remains within the VLAN and there is minimum traffic exchanged between VLANs. An inefficient VLAN topology will results in large amount of traffic generated in one VLAN that are destined for devices in another VLAN. This causes increased overhead on the router. In case of large broadcast traffic, for example in broadcast storm (Gopalakrishnan et al., 2010), more than one VLANs are affected. For efficient broadcast containment, an efficient VLAN topology should be configured in the network. An efficient VLAN topology is one where nodes exchanging large amount of traffic are placed in the same VLAN. In other words, most of the network traffic is contained in the logical VLAN units and very small amount of traffic passes through the router. For an ideal case, the network nodes must be partitioned in a way where all traffic generated in a VLAN is contained in it causing zero inter-VLAN traffic.

Consider a directed graph $G = (V, E)$ with vertex set V and directed edge set E. A directed graph in which each vertex has a directed path to every other vertex is called strongly connected graph (Dehmer and Emmert-Streib, 2014; Bollobas, 2013). Strongly connected components of a graph G are its sub graphs that are strongly connected. In every sub graph or strong component, each vertex has a directed path to every other vertex of the sub graph. Let us build a graph of network nodes such that node i exchanging a certain amount of traffic with node j are connected with a directed edge. In such a graph, group of nodes exchanging much of their traffic together will appear in the same strong component. Such strong components are good candidates for separate VLANs when the main objective of VLANs implementation is traffic localization or broadcast containment. Strong component techniques has been used in the research for various problems. In fact strong component technique could be used for any problem that can be represented by a graph. A number of algorithms exists

in the literature to find strongly connected components of a graph including the Tarjan's algorithm (Tarjan, R., 1992) and the path based depth first searching techniques (Gabow, 2000). In this paper we used the Tarjan's algorithm for finding strong components of the graph because of its linear time complexity and the easy availability of its implementation code. The complexity of Tarjan's algorithm is $O(|V|, |E|)$ where V is the set of vertices and E is the set of edges of the graph. Tarjan's algorithm uses the depth first search to explore the set of nodes which are not visited yet and placed them on a stack. The search process form sub trees of the search tree each having its own root node. The root node is the first visited node of the sub tree during the depth first search. When the search return to the root node during the recursion, the root together with its children on the stack are pop out and reported as a strongly connected component.

For VLAN partitioning, strong component technique has been used (discussed in section 2) but with a poor measure for traffic threshold. Traffic threshold is the value on the basis of which it is decided whether to create a directed edge from node i to j in the graph representing node's association according to the traffic matrix. If traffic from i to j is greater than or equal to the traffic threshold value, a directed edge from i to j is created. A value of 1, as used in the literature, is not a good measure for traffic threshold. With a value of 1 for traffic threshold, we are often unable to decompose a given graph into multiple strong components aiming for efficient segmentation of the LAN. The reason is explained in section 2. In order to provide graph representation of association of network nodes, we need a measure that is a good estimate of correlation of nodes association. So we use statistical measures for calculating traffic threshold rather than using a static value for every traffic matrix. For finding a best measure for traffic threshold, we compare classical statistical measures of Mean, Mode, Median together with other three robust measures for central tendency i.e. 20% Trimmed Mean, 5% Trimmed Mean and Distance-weighted Mean (DWM). The latter three measures are preferred by a recent study (Dodonov, Y. S. and Dodonova, 2011) for finding central tendency in data in the presence of outliers. According to our findings only Mean, 5% Trimmed Mean and DWM as measures for traffic threshold works well with strong component technique for finding efficient VLAN topology. It is also found that no single measure is best suited for all types of network scenarios. So a tradeoff between time and performance is achieved by applying the three measures simultaneously and using the best result found.

The motivation for this research work is to know and exploit node's grouping for Virtual LAN design. The solution is intended to help network administrator in designing a good VLAN topology for the purpose of traffic localization and broadcast containment. Currently it is assumed that prior to VLAN implementation, the network administrator is able to collect traffic statistics. The administrator then apply the proposed technique on the offline traffic statistics on a stand alone computer. With the help of proposed solution, he/she get an idea of nodes grouping in the LAN. Such grouping can be exploited to design a good VLAN structure that provide better traffic localization and broadcast containment. The administrator has the choice of changing the proposed VLAN topology incase of any conflict with organizational policy and implement the topology manually. Parameters like IP addressing, trunk ports, VLAN tagging etc, is decided after the revised topology. Another important application lies in

datacenter virtualization and cloud computing environments. Some of these platforms implement isolation among tenants through layer-2 VLANs (Sherwood et al., 2010; Drutskoy et al., 2013; Mudigonda et al., 2011; Hao, F. et al., 2010; Wood et al., 2009). An important consideration of the datacenters is to minimize energy consumption (Orgerie et al., 2014). In datacenters, major (70%) energy consumption is attributed to servers and cooling requirements and minor (30%) to network infrastructure. Only the infrastructure energy consumption sum for 3 billion kWh in 2006 (Nunes et al., 2014). Elastictree (Heller, B. et al., 2010) and HoneyGuide (Shirayanagi et al., 2012) are two well known proposal for lowering energy consumption in datacenter and clouds. ElasticTree finds minimal network subset that can handle the current traffic load and shutdown the network devices that are not needed. HoneyGuide on the hand addresses servers energy utalization by moving virtual machines from underutalized servers and shutdown them. The proposed technique could also be used by such energy conservation mechanisms, for example, to identify VMs which communicate mostly with each others and that are spread across datacenter. Thus migrating these VMs groups into closer physical machines and turning off the free servers will enhance the energy conservation. With a slight modification to identify switches that are part of the traffic flows between a pair of VMs, the unnecessary switches can also be turned off.

The contribution of this paper are three fold. First we find that statistical measures should be used for selecting the traffic threshold instead of using a static value while generating adjacency matrix from traffic matrix. Second the best statistical measures for approximating correlation in traffic matrix are Mean, 5%Trimmed Mean and DWM. Third for strong component technique, no single measure for traffic threshold is best suited for each and every type of network scenarios and a tradeoff between time and performance should be used.

The rest of the paper is organized as follows. Section 2 provides a brief discussion of state of the art algorithms for VLAN partitioning. Section 3 present a mathematical model for reducing graph partitioning into VLAN partitioning. In Section 4 we compare different measures for traffic threshold selection. Section 5 discuss the comparison of proposed techniques with existing state of the art algorithms. In section 6 we conclude the paper.

2 Related Work

VLANs optimization has been studied as a part of the whole network optimization in (Sun et al., 2011, 2010; Sung et al., 2008). These are mathematical cost models built through the interaction with network operators. The cost models are optimized to get an optimized plan for VLANs deployment. The problem with these approaches is the interaction required to build the cost model which is time consuming and less efficient for traffic localization. Moreover these solutions are not specific for VLAN membership optimization. CSS-VM (Li, F. et al., 2013) also used a modified form of cost model given in (2010) to further divide a VLAN into sub VLANs. It requires the current VLAN configuration for further partitioning and also needs operator's interaction for building the initial cost model. Since we are finding a good VLAN membership for ordinary network nodes, to do better broadcast containment and traffic localization, we

need traffic statistics of each of such node. Without traffic information we would need sufficient operator assistance to identify pair of nodes that are expected to exchange large amount of information. Such a technique is not scalable for large networks and also inefficient for today's dynamic networks. So in this research work we are considering only those heuristics approaches where traffic information of network nodes are used to identify their VLAN membership.

A recent research work (Hameed and Adnan, 2012) presents an algorithm for the problem in consideration. The algorithm is called Set-based algorithm (called SSAlgo in this text) that uses set or array processing of traffic matrix to find VLANs membership for each node. The algorithm is exhaustive and is not based on well known optimization or searching techniques.

The authors in (Esposito et al., 2004) discuss a heuristic which is based on tree processing. The algorithm build a graph of the network nodes and then remove cycles to represent it in the form of a tree. The largest non-repeating leaves of the tree are selected and used as candidate VLANs. Problem with this approach is that a node may be present in more than one leaves or VLANs. When such a topology is implemented in the real network, traffic to and from such a node must be forwarded to all respective VLANs, which limit the benefit of VLAN traffic localization. We call this a group collision and should always be avoided to get a good VLAN topology.

Sean Rooney (Rooney et al., 1999) provide a greedy heuristic approach for solving the problem in consideration. His algorithm merge a node or pair of nodes with another pair if the grouping maximizes the total traffic of the group. Groups are extended by the merging process subjected to a size constraint. When no nodes or groups are left to be merged, each group is used as a separate VLAN. Sean Rooney algorithm also suffer from the group collision with another problem caused by the group size constraint. A node might not get into its optimum set because the group has reached its maximum size. Such non-optimum placement of nodes increases inter-VLAN traffic.

Peltseverger (Peltsverger and McKenney, 2008) also uses the strong component technique from graph literature to find a good VLAN topology for a set of network nodes. Each strong component of the graph is used as a candidate VLAN. The algorithm first create a boolean matrix called the adjacency matrix from traffic matrix using the threshold value equal to 1. With this threshold, any value greater than 0 in traffic matrix causes Peltserverger's technique to place a 1 in the adjacency matrix. A value of 1 of adjacency matrix means an edge in the graph created from the same matrix. With this practice, in most cases, each vertex will have a path to every other vertex of the graph. This turned the whole graph into a single strong component. With a single strong component in the graph, all nodes are part of the same VLAN just like in case of no VLAN configuration where all nodes are considered as members of the default VLAN and part of the single broadcast domain. So the algorithm often failed to identify a good VLAN membership for each node.

Our proposed technique for VLAN partitioning is similar to that of Peltseverger's in that it finds strongly connected components of the graph constructed from the traffic matrix. The advantage of our approach lies in the identification and use of a good value for traffic threshold while calculating the adjacency matrix. Rather than using a static value for traffic threshold, we identify and use a value that is a good estimate of

correlation in the traffic matrix. This correlation is identified through the use of statistical measures. With a statistically identified traffic threshold, adjacency matrix and the resultant graph shows a good association among nodes exchanging comparatively large amount of traffic. In order to cope with each and every type of network environment, we use best of the three results. These results are calculated by the application of three chosen statistical measures for the same traffic matrix. Additionally the proposed technique uses the well-known linear time Tarjan's algorithm for computing strongly connected component of the graph.

3 Turning graph partitioning into VLAN partitioning

In graph partitioning a graph is divided into sub components each having certain properties. The total possible number of partitions for n nodes and arbitrary k partition size is called the Bell number (Erickson, 2013; Mezo, I., 2011) and is calculated by the Bell formula in Eq 1.

$$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k \tag{1}$$

Graph partitioning has application in various fields like VLSI design, network design, routing, resource scheduling, data clustering etc. In this section we discuss how to use graph partitioning to partition a network of nodes into appropriate VLANs.

Let us have n nodes for which VLAN membership has to be found on basis of their traffic statistics gathered from their network activities. Traffic information is in the form of $n \times n$ matrix called the traffic matrix T. Each value $t_{i,j}$ of matrix T is the total amount of traffic from node i to node j. Lets we have a value called traffic threshold θ used to construct a $n \times n$ boolean matrix called the adjacency matrix A from matrix T. Each value $a_{i,j}$ of A is 1 if the corresponding $t_{i,j}$ value of T is equal to or greater than the θ otherwise $a_{i,j}$ is 0, i.e.

$$a_{i,j} = \begin{cases} 1, & if\ t_{i,j}\ \geq\ \theta \\ 0, & otherwise \end{cases} \tag{2}$$

To represent association between n network nodes, we have a graph $G = (V, E)$ with vertex set V containing nodes IDs of n nodes. The edge set E of G contains pairs of node IDs representing association between nodes. Each pair $e_{u,v}$ represents an edge from node u to node v. These pairs are calculated from matrix A such that, for each value $a_{i,j} = 1$ in A there exists a pair $e_{i,j}$ in E i.e.

$$e_{i,j}\ \in E\ iff\ a_{i,j} = 1 \tag{3}$$

Now the objective of is to find group of nodes exchanging much of their traffic together represented by strongly connected components of G. In other words, the objective is to partition the vertex set V of G into subset $(g_1, g_2....g_m \subset G)$ such that each vertex in subset g_i has a path to every other vertex of the subset. Finally we use each

Algorithm 1: SC VLAN partitioning Algorithm

Input: ***Traffic Matrix T***
Output: VLAN Topology

```
1  begin
2      Calculate traffic threshold θ from matrix T
3      Loop through the rows and columns of matrix T and A such that:
4      if T[row][column] ≥ θ then
5          A[row][column] = 1
6          else
7              A[row][column] = 0
8          end
9      end
10     Loop through the rows and columns of matrix A and do V.addVertex(row) on each new row occurrence only.
       On each column,
11     if A[row][column] == 1 then
12         E.addEdge(row,column)
13     end
14     Find strongly connected components of G (V,E)
15     Use each strong component as a separate VLAN.
16 end
```

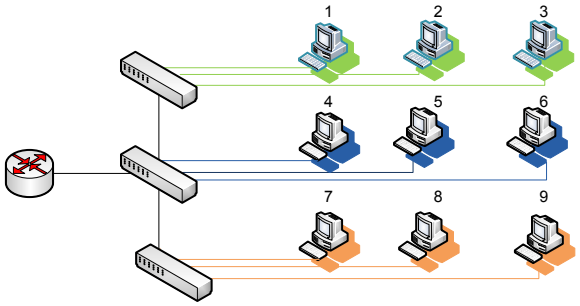


Fig. 1. Example Network

	1	2	3	4	5	6	7	8	9
1	0	0	448	1440	128	0	0	2469512	0
2	128	0	34294	6952	648	256	0	0	0
3	0	34230	0	2120	0	7320	0	2400	0
4	0	2400	120	0	0	0	2996280	1976	0
5	0	4200	0	0	0	8400	0	6326576	0
6	0	0	832	0	528	0	0	125480	393160
7	0	0	0	115280	0	0	0	0	0
8	95224	0	256	336	242832	4744	0	0	0
9	0	0	0	0	0	15040	0	0	0

(a) Traffic Matrix

	1	2	3	4	5	6	7	8	9
1	0	0	0	0	0	0	0	1	0
2	0	0	1	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	0	0	1	0
6	0	0	0	0	0	0	0	1	1
7	0	0	0	1	0	0	0	0	0
8	1	0	0	0	1	0	0	0	0
9	0	0	0	0	0	1	0	0	0

(b) Adjacency Matix

Fig. 2. Matrix conversion

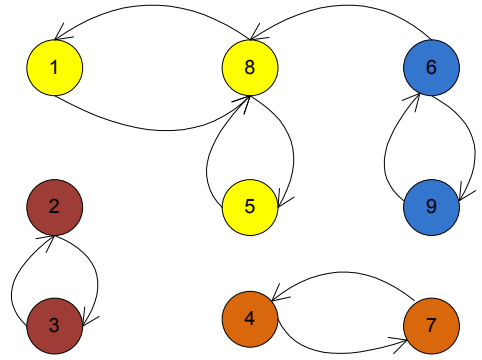


Fig. 3. Graph with strongly connected component

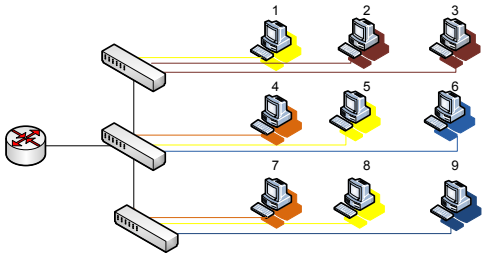


Fig. 4. Network with proposed VLAN topology

of the subset or strong component as a candidate for separate VLAN. This process is shown step by step in algorithm 1.

Algorithm 1 is a simple sequential steps of procedure for finding a good VLAN topology using strong component technique. In the first step (line 2) of the algorithm, we calculate the threshold value θ from traffic matrix T. Next (line 4-9) we calculate adjacency matrix A from T using the value θ according to equation 2. In step three (line 10-13) of the algorithm, we generate the graph G from boolean matrix A. In step four (line 14) we decompose the graph G into subgraphs such that each subgraph is a strongly connected component of G using the well-known Tarjan's algorithm. Finally we use each strong component founds in step four of the algorithm as a separate VLAN. As discussed earlier and also explained in section 4 we adopt best of the three strategy. We apply three different statistical measures for calculating traffic threshold to cope with different types of network scenarios. So in this way to get a final solution for the input traffic matrix, Algorithm 1 is executed thrice each time with a different threshold value calculated by one of the three measures. Finally, the best result (in terms of maximum intra-VLAN traffic or minimum inter-VLAN traffic) among the three is used.

The proposed algorithm is explained with the help of an example for finding VLAN membership (VLAN topology) for a 9 nodes network shown in figure 1. Similar color nodes are present in the same physical premises of the organization. Traffic matrix T for the network is shown in figure 2(a). For this example the threshold value θ is calculated as the mean of traffic matrix which is 14027 in this case. Adjacency matrix A on basis of the calculated θ value is shown in figure 2(b) where cells with value of $1's$ are highlighted. These $1's$ shows association among the network nodes. The graph G for the corresponding adjacency matrix A (adjacency matrix in figure 2(b)) is shown in figure 3. Nodes of G which are in the same strong component are colored with the same color. As shown by figure 3, node 6 has a directed edge to node 8 but they are not in the same strong component because all nodes of yellow strong component does not have a directed path to the blue strong component nodes. The resultant VLAN topology is implemented in the network shown by figure 4 where VLAN members are colored with the same color.

4 Selection of measure for traffic threshold

Data with graphs could be represented in a number of ways including the adjacency or boolean matrix method. Using adjacency matrix method for constructing a graph representation of a network for a given traffic matrix, we need a value used to represent association among nodes. This value is called traffic threshold. Any value greater than 1 is not a good measure as discussed in previous section. We need a measure that is a good estimate of nodes correlation for the current traffic matrix. If we draw all values of the traffic matrix on a plane then a good measure for nodes correlation is a point that is a good representation of all values on the plane. This is equivalent to finding central tendency in the traffic matrix (Weisburd and Britt, 2014; Sinova et al., 2014). A number of statistical measures are used for finding central tendency in a distribution including the Mean, Mode, Median Distance-Weighted Mean etc.

4.1 Commonly used measures

Mean, Median and Mode are three commonly used measures of central tendency. The mean value for a collection $x_1, x_2....x_n$ is given by Eq 4.

$$Mean = \frac{\sum_{i=1}^{n} xi}{n} \tag{4}$$

Mean is used more frequently for finding central tendency in the collection but is not best for unsymmetrical data. The reason is that, in mean larger values of the distribution have more influence on the mean value than the smaller ones. The best measure for our problem is one that selects a point with more neighboring points in its surrounding in the plane.

Median represents the middle item of a sorted collection of values i.e. it divide the collection into a lower half and higher half. When the number of values of the collection is odd then median is just the middle item i.e. $(n + 1)/2$th item. For an even number list, median is the average of the two middle values of the order list. Mode of a list represents the value the appears mostly in the list.

4.2 The Trimmed Mean

Trimmed mean is more robust than the simple mean and is less sensitive to outliers in the dataset (Wilcox, 2012). All trimming measures of central tendency are inspired from the idea that removing outliers from the dataset and further averaging the remaining values will provide more stable estimate of central tendency. To calculate $k\%$ trimmed mean for a collection, the data is sorted and the highest $k\%$ and the lowest $k\%$ values from the list are removed. Mean is then calculated on the remaining values. Different percent of trimming from 10% to 25% are proposed by researchers depending on the type of data analysis.

4.3 The Distance-Weighted Mean

Distance-Weighted Mean (DWM) or Distance-Weighted Estimator (DWE) is similar to simple mean but unlike the mean, each data point does not contribute equally to mean value. In this measure of central tendency, each point is assigned a weight such that points that are closer to other points carry higher weights than points which are away or isolated from others. Thus central point of the observation have more influence on the mean value. The weight w_i for data point x_i is calculated as the inverse of mean distance between point x_i and other data points of the distribution, i.e.

$$w_i = \frac{n-1}{\sum_{j=1}^{n} |x_i - x_j|} \tag{5}$$

The DWM is then calculated as

$$DWM = \frac{\sum_{i=1}^{n} (w_i \cdot x_i)}{\sum_{i=1}^{n} w_i} \tag{6}$$

DWM is a stable measure of central tendency in the presence of outlier data and is preferred for speedy task (Dodonov, Y. S. and Dodonova, 2011). Other advantage of DWM includes easy calculation and consideration of all data points. Additional advantage of DWM over trimming is that it does not require the deletion of certain data which is more important in analysis when no points could be identified as outliers.

4.4 Metrics for evaluation

To evaluate which of the central tendency measure is best suited for our problem, the proposed technique is tested with each of the measures for central tendency described in this section. As mentioned earlier that the objective of finding an efficient VLAN topology is to arrange nodes of the network into VLANs in such a way that maximizes traffic within the VLANs called intra-VLAN traffic. This also implies minimizing traffic going outside of the VLANs called the inter-VLAN traffic. A good measure of central tendency when used with strong component should result in a high intra-VLAN traffic or low inter-VLAN traffic. To compare the results, let assume the fraction of total traffic contained within the VLAN boundaries as the percent intra-VLAN traffic $\%\alpha$. This is calculated as $\frac{\alpha}{\Gamma} \times 100$ where the symbol Γ represent the total traffic. Similarly $\%\beta$ represents the fraction of traffic exchanged with other VLANs i.e., percent inter-VLAN traffic given by $\frac{\beta}{\Gamma} \times 100$. A theoretical optimal solution for this problem is a solution where all traffic generated in the LAN is contained in the source VLANs only leading to $\%\alpha = 100$. So the optimal bounds for $\%\alpha$ is 100. This means that zero traffic is exchanged between VLANs causing $\%\beta = 0$. So for $\%\beta$, the optimal bound is 0. Any of the two bounds i.e. $\%\alpha = 100$ or $\%\beta = 0$ could be used to favor some solutions over others. Given a traffic matrix, a topology is favored over another on the basis of how much its $\%\alpha$ or $\%\beta$ values are closer to their respective optimal bounds. For example topology 1 with $\%\alpha = 90$ is better than topology 2 with $\%\alpha = 80$.

It is important to note that the number of groups (strong components) produced by the proposed technique depends on the threshold value and also on the implicit grouping that already exists among nodes. The traffic threshold identifies which value in traffic matrix will result in an edge in the graph. A valid solution for a given traffic matrix and threshold, confirms to two constraints. The first constraint is that the total number of groups in the solution should be in the range $2...(n-1)$,where n is the number of vertices in the graph i.e.

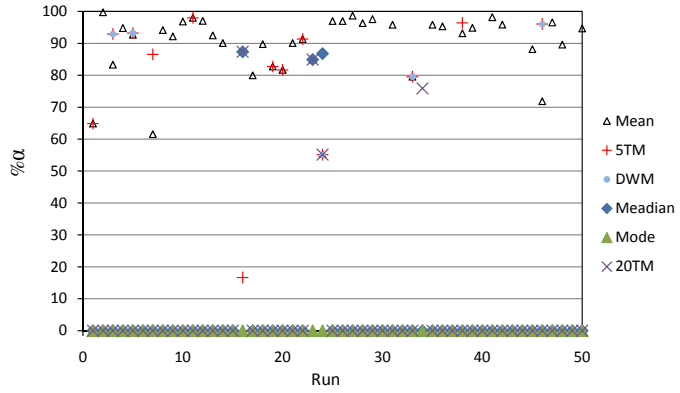
$$C1 : nGroups \in \{2...(n-1)\} \mid n = vertices\_in\_graph \tag{7}$$

With constraint $C1$, a solution is invalid if it has a single group or a separate group for each node. The second constraint is that the number of nodes per group is in the range $1...(n-1)$ i.e.

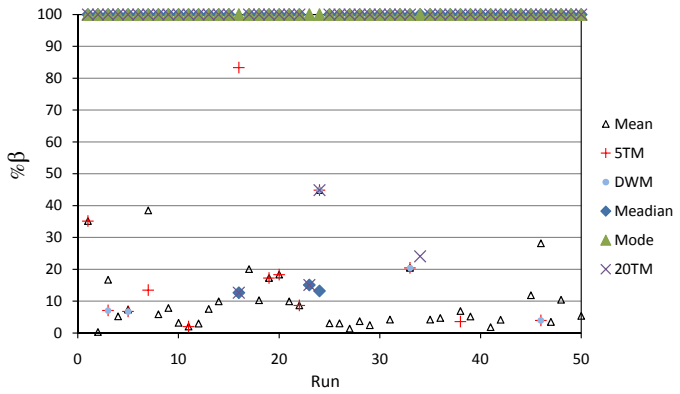
$$C2 : nodes\_per\_Group \in \{1...(n-1)\} \mid n = vertices\_in\_graph \tag{8}$$

$C2$ makes sure that a valid solution does not include any empty group or null strong component. An invalid solution, on the other hand, indicates that the algorithm failed to decompose the graph. For such solution worst values are assumed i.e. 0 and 100 for $\%\alpha$ and $\%\beta$ respectively. For instance considering the traffic matrix of Internet cafe

that have a single gateway, the algorithm will result in a single strong component and thus a single VLAN will be generated. Since the solution does not confirm to the two stated constraints i.e. C1 and C2, it is invalid. In this case the administrator can choose to either keep or discard the solution.



(a) IntraVlan traffic Percentage



(b) InterVlan traffic Percentage

Fig. 5. Central tendency measure comparison

4.5 Comparison of central tendency measures

The experimentation detail for comparison of the discussed measures of central tendency is as follows. All measures are tested 50 times each constitute a run. In each run of the simulation, a network with random number of nodes is created. To simulate a complete randomized network activity of nodes, a traffic matrix with all random values is created in each run. Thus each of the experimentation run simulate a different

network scenario. In each run, the strong component technique is used with various measures (mean, mode, median, 20%TM,5%TM and DWM) for central tendency and $\%\alpha$ and $\%\beta$ values are calculated for each of the measure. The result of the experimentation is plotted in figure 5 where $\%\alpha$ and $\%\beta$ are drawn on the scale between 0 and 100 on y-axis.

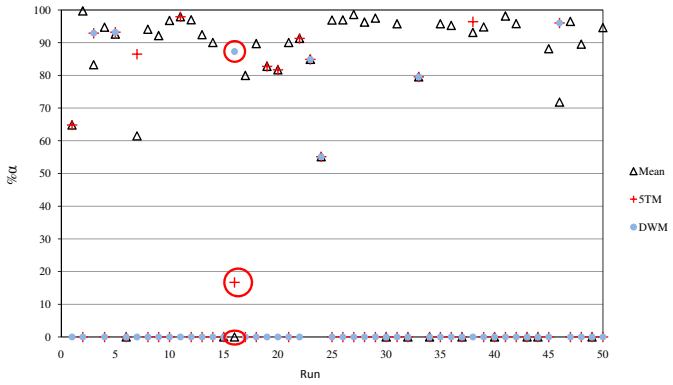
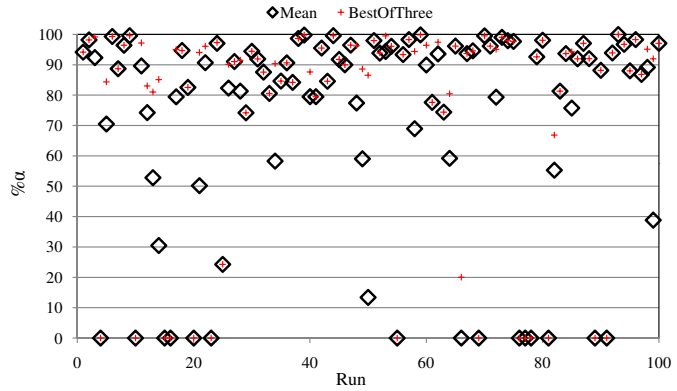


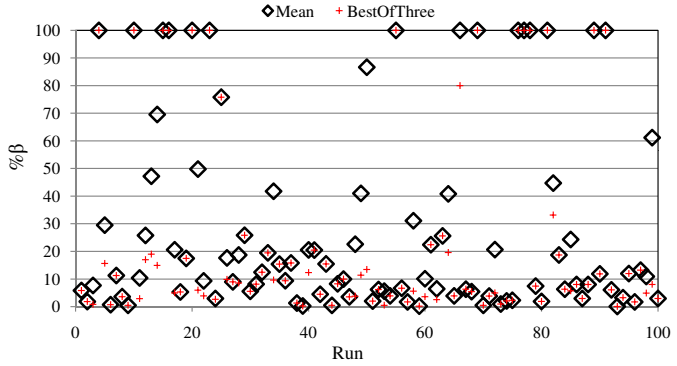
Fig. 6. Three selected measures comparison

In figure 5(a), x-axis shows run or simulation number and y-axis shows $\%\alpha$. Note that $\%\alpha$ values that are closer to the optimal bound (100 for $\%\alpha$) represent good solutions. Such a solution means that major portion of the total traffic is contained within the VLAN groups which is the objective of our VLAN partitioning. Higher values for $\%\alpha$ means lower values for $\%\beta$ which is shown in figure 5(b), where $\%\beta$ for good solutions are closer to zero (optimal bound for $\%\beta$). From both figures we observe that Median, Mode, and 20% TM as a measure for central tendency in traffic matrix perform worst so we drop these three measures from our analysis. The result for the remaining three measures i.e. Mean , 5%TM and DWM is further plotted in figure 6 to get a clear idea of their performance with respect to each other. Figure 6 shows that for our problem, Mean as a measure for central tendency in traffic matrix perform better than others. In figure 6 it can be clearly seen that most of the triangle shapes that represent $\%\alpha$ for the Mean measures lies higher than others in the graph. However we found that sometimes Mean resulted in a poor $\%\alpha$ values than the other two measures. An instance in the graph is highlighted with circles that shows DWM yields higher $\%\alpha$ than 5% TM and Mean. In the figure 6 we also found that 5% TM perform comparatively better than DWM that is shown by more plus signs laying higher in the graph than the small filled circles for DWM. To summaries the conclusion of this experimentation, Mean as a measure for central tendency in traffic matrix perform mostly better but some time gives equal or worst result than 5% TM and DWM. The same is true for 5% TM where it gets the second position in performance but in some cases it gives poor result than DWM. The fact suggest that given a different network scenario each time, no single measure for central tendency in traffic matrix could result a best solution all the times. A better way is to

apply all of the three measures and use best of the three results. With this approach the idea is to apply strong component technique with the three measures on the traffic matrix and save the three outputs. Each measure may result in a solution that will be equal to , better or worst in terms of $\%\alpha$ or $\%\beta$ than the others two solutions. Then select the solution that has the highest $\%\alpha$ or lowest $\%\beta$ as a final result. With this extension the time consumed for producing a final result is simply multiplied by three which is not too much if we get a better result.



(a) Percent IntraVlan Traffic

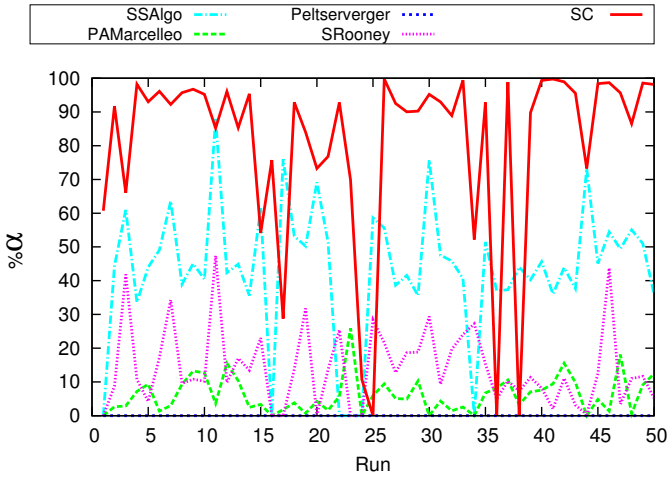


(b) Percent InterVlan Traffic

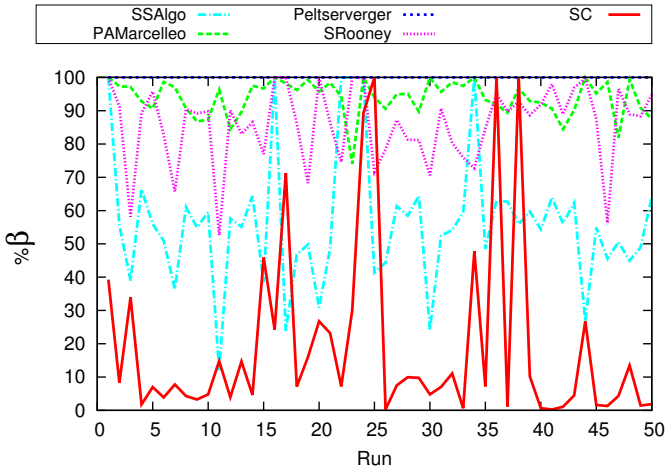
Fig. 7. Mean Vs. Best of three mechanism

To further explain the usefulness of the best of three mechanism, we add another experiment that compare the best of three approach to the single Mean as a measure. The experiment contains 100 simulation runs each time with a different network scenario. The result is plotted in figure 7. Again x-axis shows simulation run number and y-axis shows percent inter-VLAN traffic i.e. $\%\alpha$ for that run. In figure 7(a) a sign of plus over diamond shows that both Mean and BestOfthree mechanism produces the same result.

An empty diamond shows that Mean measure produce worst result than BestOfThree mechanism. A number of empty diamond shapes in figure 7(a) shows that the solution could be improved using the three measures i.e. Mean, 5%TM and DWM at the same time. Figure 7(b) shows $\%\beta$ results for the same experimentation that also depict the same fact.

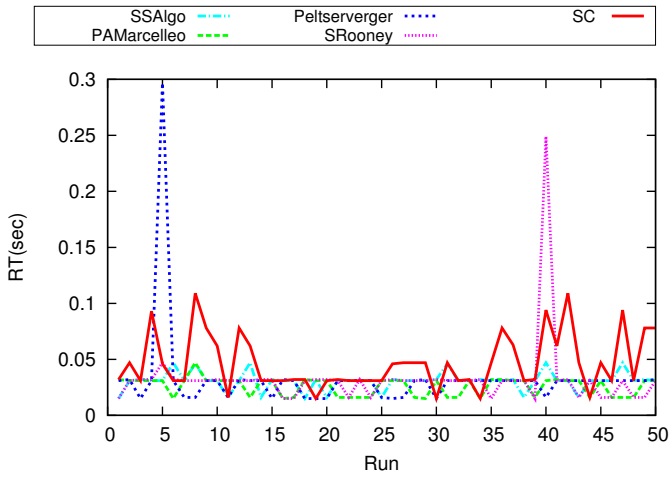


(a) Percent IntraVlan Traffic

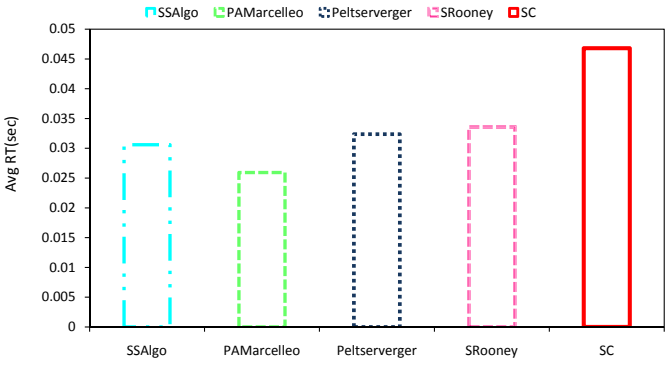


(b) Percent InterVlan Traffic

Fig. 8. Comparison with other heuristics



(a) Execution time in seconds



(b) Average execution time in seconds

Fig. 9. Execution time comparison

5 Experiments and Results discussion

This section discuss the comparison of the proposed technique for VLAN partitioning to the state of art heuristics discussed in section 2. As discussed earlier we are considering only those heuristics approaches where traffic information of network nodes are used to identify their VLAN membership. These heuristics are Simple Set-based algorithm (called SSAlgo in this paper), Esposito Partitioning Algorithm(PA) (called PAMarcelleo in this paper), Peltsverger Algorithm and Sean Rooney Algorithm (called SRooney in this paper). All of the comparative heuristics takes traffic matrix as input and identify grouping of nodes corresponding to the traffic matrix.

The comparison is carried out on a machine with specification Intel Core(TM) i3-2310M CPU @2.10GHz. The experiment is repeated 50 times, each time with a different traffic matrix and number of nodes in the network. Three parameters $\%\alpha$, $\%\beta$ and running time (RT) for each heuristics are calculated in every run. Figure 8(a) shows the result for percent inter-VLAN traffic i.e. $\%\alpha$. X-axis of the graph shows run number and Y-axis shows the $\%\alpha$ value calculated for the VLAN topology produced by the respective algorithm. For the input traffic matrix, our proposed technique compute three VLAN topologies using three measures for central tendency and select one with the highest $\%\alpha$ value. The solution with highest $\%\alpha$ indicate efficient grouping of network nodes into VLANs that localize much of the traffic produced. Figure 8(a) shows that the proposed technique outperform others as shown by the red solid line which remains almost high for all of the simulation runs.

Efficient traffic localization or higher values for $\%\alpha$ imply that a small portion of total traffic crosses the different VLANs boundaries. Figure 8(b) shows this portion of traffic for each heuristic. VLAN grouping proposed by our Strong Component (SC) based techniques with best of three mechanism minimize $\%\beta$ more than other heuristics shown by the lower RED solid line. The time consumed by each of the heuristics for producing a VLAN topology for the input traffic matrix is shown in figure 9. Figure 9(a) shows running time in seconds on y-axis for the simulation run number shown on x-axis. This figure describe the fact that the proposed technique consume slight more execution time than others heuristics. The reason is that strong component technique is used three times for producing a single result. But the figure also forward the fact that the difference in execution time is only in magnitude of millisecond which is also confirmed by the height of bar for average execution time in figure 9(b) for the 50 runs. So producing an efficient VLAN grouping that maximizes intra-VLAN traffic and minimizes inter-VLAN traffic with a slight more running time in magnitude of millisecond, confirm the usefulness of the proposed technique for the problem in consideration.

6 Conclusion

An efficient VLAN topology is always desirable when the basic purpose of VLANs implementation is traffic localization and broadcast containment. In this paper we present a strong component based technique for identifying nodes grouping in the LAN prior to VLANs implementation. Information about such node's grouping allows network administrator in deciding a good VLAN topology that maximizes intra-VLAN traffic and

minimizes inter-VLAN traffic. Such grouping information could also be used by clouds energy conservation mechanisms to identify group of VMs communicating mostly with each others and migrating them into closer physical premises. Strong component technique has already been used for identifying VLANs grouping on the basis of traffic information but with a poor measure for traffic threshold. This paper discusses the use of central tendency measures for finding a better value for traffic threshold to represent nodes association in terms of a graph. Six different measures i.e. Mean, Median, Mode, 20% TM, 5% TM and DWM are cross compared to find one best suited for the problem. The experimentation results show that only Mean, 5% TM and DWM produce notable results when used with strong component technique which are further explored to choose one as a best. Further experimentation suggest that no single measure of central tendency used with strong component technique is best suited for each and every network scenario. A tradeoff between time and performance is adopted instead of using a single measure. To calculate traffic threshold value, three measures i.e Mean, 5% TM and DWM are used with strong component technique for identifying nodes grouping according to the traffic information. The final solution selected is the best result among the three, get from the application of strong component technique with each of the three said measures. The proposed approach is compared with four other heuristics for the same problem where it out perform others in terms of maximizing intra-VLAN and minimizing inter-VLAN traffic with a slight more time consumption which is in magnitude of milliseconds. Comparison with state of the art suggest that strong component technique with the three mentioned measures for calculating traffic threshold value, can identify efficient VLANs grouping.

References

Bollobas, B. (2013). Modern Graph Theory*(Graduate Texts in Mathematics)*, Springer.

Dehmer, M. and Emmert-Streib, F. (2014). Quantitative Graph Theory: Mathematical Foundations and Applications *(Discrete Mathematics and Its Applications)*,CRC Press.

Dodonov, Y. S. and Dodonova, Y. A. (2011). Robust measures of central tendency: weighting as a possible alternative to trimming in response-time data analysis, *Psikhologicheskie Issledovaniya,* vol. 5(9).

Drutskoy, D., Keller, E. and Rexford, J. (2013). Scalable network virtualization in software-defined networks, *Internet Computing, IEEE* vol. 17(2), 20–27.

Erickson, M. (2013).Introduction to Combinatorics, *Wiley Series in Discrete Mathematics and Optimization*, Wiley.

Esposito, M., Pescape, A. and Ventre, G. (2004). An efficient approach to the network division problem, VLANs configuration and WLANs hosts grouping, *in* Local and Metropolitan Area Networks, 2004 (LANMAN 2004). The 13th IEEE Workshop on, pp. 241–246.

Gabow (2000). Path-based depth-first search for strong and biconnected components, *IPL: Information Processing Letters* vol.74.

Gopalakrishnan Nair, T. R., et al. (2010). A novel agent based approach for controlling network storms, *in* 'IEEE Third International Conference on Communications and Electronics(ICCE)'.

Hameed, A. and Mian, A. N. (2012). Finding efficient vlan topology for better broadcast containment, *in* Network of the Future (NOF), 2012 Third International Conference, pp. 108–113.

Hao, F. et al. (2010). Secure cloud computing with a virtualized network infrastructure, *in* Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, HotCloud'10, USENIX Association, Berkeley, CA, USA, pp. 16–16.

Heller, B. et al. (2010). Elastictree: Saving energy in data center networks, *in* Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation, NSDI'10, USENIX Association, Berkeley, CA, USA, pp. 17–17.

IEEEStandard (2011). 802.1Q-2011 IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks.

Li, F. et al. (2013). CSS-VM: A centralized and semi-automatic system for VLAN management, *in Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pp. 623–629, 27-31 May 2013.

Mezo, I. (2011). The r-bell numbers, *Journal of Integer Sequences* vol.14, pp. 2932 – 2936.

Mudigonda, J. et al. (2011). Netlord: A scalable multi-tenant network architecture for virtualized datacenters, *SIGCOMM Comput. Commun. Rev.* vol. 41(4), pp. 62–73.

Nunes, B. et al.(2014). A survey of software-defined networking: Past, present, and future of programmable networks, *Communications Surveys Tutorials, IEEE* vol. 16(3),pp. 1617–1634.

Orgerie, A.-C., Assuncao, M. D. d. and Lefevre, L. (2014). A survey on techniques for improving the energy efficiency of large-scale distributed systems, *ACM Comput. Surv.* vol. 46(4), pp. 47:1–47:31.

Rooney, S., Hörtnagl, C. and Krause, J. (1999). Automatic VLAN creation based on on-line measurement, *Computer Communication Review* vol. 29(3).

Sherwood, R. et al.(2010). Can the production network be the testbed?, *in* Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation, OSDI'10, USENIX Association, Berkeley, CA, USA, pp. 1–6.

Shirayanagi, H., Yamada, H., Kono, K.(2012). Honeyguide: A vm migration-aware network topology for saving energy consumption in data center networks, *in* Proceedings of the 2012 IEEE Symposium on Computers and Communications (ISCC), ISCC '12, IEEE Computer Society, Washington, DC, USA, pp. 460–467.

Sinova, B., Casals, M. R. and Gil, M. A. (2014). Central tendency for symmetric random fuzzy numbers, *Information Sciences* vol. 278,pp. 599–613.

Peltsverger S. and McKenney C. (2008). Students research in dynamic VLAN configuration, *in* InfoSecCD Conference'08', Kennesaw, GA.

Sun, X. and Rao, S. G.(2011). A cost-benefit framework for judicious enterprise network redesign, *INFOCOM, 2011 Proceedings* IEEE, pp. 221–225, 10-15 April 2011.

Sun, X. et al.(2010). A systematic approach for evolving VLAN designs, *in* INFOCOM, IEEE, pp. 1451–1459.

Sung, Y.-W. E. et al. (2008). Towards systematic design of enterprise networks, *In* Proceedings of the 2008 ACM CoNEXT Conference (CoNEXT '08). ACM, New York, NY, USA,

Tarjan, R. (1992). Depth-first search and linear graph algorithms, *SIAM Journal of Computing* vol. 1(2).

Weisburd, D. ,Britt, C. (2014). Describing the Typical Case: Measures of Central Tendency,*Statistics in Criminal Justice*,4th Ed., Springer US, p. 59–85.

Wilcox, R. R. (2012). Introduction to Robust Estimation and Hypothesis Testing, 3rd Ed., *Statistical modeling and decision science*, Academic Press.

Wood, T. et al. (2009). The case for enterprise-ready virtual private clouds, *in* Proceedings of the 2009 Conference on Hot Topics in Cloud Computing, HotCloud'09, USENIX Association, Berkeley, CA, USA.

Received January 7, 2015 , revised April 9, 2015, accepted April 20, 2015