Layer-2 Routing Analytic Model by Linear Programming

Changjin Suh¹, Shin-woong Jung²

School Of Computing, Soongsil University¹
1-1 Sangdo-3-Dong, Dongjak-Gu, Seoul, Korea
¹cjsuh@ssu.ac.kr, ²jungsw@networking.com

Abstract— Carrier Ethernet integrates distributed layer-2 bridge networks to a backbone network. Since carrier Ethernet uses simple layer-2 routing, its performance is lower than that of layer-3 routing methods.

We introduce an analytic model that evaluates the routing methods in backbone networks. The model solves the multi-input and multi-output maximum flow problem in backbone networks using linear programming. Our method can be used as follows. Firstly, the model suggests appropriate link capacity if network topology and traffic pattern is given. Secondly, the model shows and compares the performance of layer-2 routing methods including perfect routing and our proposed routing called ENDIST (Edge Node Divided Spanning Tree). In this study, the performance is the maximum throughput that a given routing method can reach.

We additionally found many good results about ENDIST. Our analytic model proved that ENDIST outperforms the existing layer-2 standard protocols such as STP, MSTP for any offered load and performs perfectly under medium and light traffic load in the given network. Thus, it is clear that ENDIST is good as layer 3 routing from the point of throughput.

Keywords — Linear Programming, Carrier Ethernet, Layer-2 routing, maximum flow, STP, MSTP, SPB

1 Introduction

IEEE 802.1aq Provider Backbone Networks (PBBN), called carrier Ethernet networks, connecting several geographically distributed layer-2 bridge networks [1]. There are two types of nodes in PBBN. A Backbone Edge Bridge (BEB) manages inputting and outputting packets at the boundary of PBBN, and a Backbone Core Bridges (BCB) forwards fast frame from BEBs. BEBs interconnect a lot of Layer-2 bridged networks and BCBs interconnect BEBs. A BEB forwards packets to another BEB through core network of PBBN. For simplicity we will use “core node”, “edge node” and “backbone network” from now on. Figure 1 shows an example of a PBBN. In Figure 1 five circles indicate five core nodes and seven squares show seven edge nodes. An edge node u is a gateway bridge to upper level networks and all the other edge nodes connecting lower level networks.

Layer-2 networks currently use Spanning Tree Protocol (STP), Multiple STP (MSTP) for routing. And Shortest Path Bridging (SPB) is recently emerging. IEEE 802.1aq (STP) constructs a single spanning tree for routing [2]. STP shows very poor link utilization ratio because links excluded in the spanning tree keeps inactive at normal state. IEEE 802.1s Multiple Spanning Tree Protocol (MSTP) improves STP by allowing multiple spanning trees [3]. IEEE 802.1aq SPB heavily bases on MSTP and constructs n shortest path spanning trees rooted at each edge nodes in an n-edge node backbone network [4].

Lastly, we introduce the Edge-Node Divided Spanning Tree (ENDIST) proposed by ourselves [5]. ENDIST fundamentally uses SPB. ENDIST generates spanning trees as many as number of all sub-nodes, while SPB generates them as many as number of nodes. In a backbone network edge nodes are always connected to the core node with two or more links for redundancy. Utilizing this property, each edge node in ENDIST has sub-nodes as many as connecting links. Use of multiple routing paths contributes to considerable throughput enhancement. The adoption of sub-node is easily implemented by writing source and destination sub-node addresses in the mac-address field.

Let us find out the routing path from the edge-nodes A to F in Figure 1. As A and F have two connecting links, there are 4 possible (source, destination) routing paths such as (A₁,F₁), (A₂,F₂), (A₃,F₃) and (A₄,F₄). Based on the first link in the path, we divide four routing paths into two path groups. After calculating the best path at each path group, the best path is assigned to a sub-node. So choosing the routing path of (A,F) is simplifies as (A,F).

Figure 1. Example of a small backbone network

---

¹ This work was supported by the Soongsil University Research Fund
We choose a sub-node A, considering current status. For example if output queue is shorter in A than in B, we choose A. Figure 2(c) shows two best paths describing the routing path from node A in the backbone network in Figure 1 represented as bold lines. When we fix the destination node F, they are A-G-J-F in the left spanning tree and A-H-K-F in the right one in Figure 2(c). Suppose the lower parts of backbone network is congested. Then edge node A prefers the path A-G-J-F to the path A-H-K-F.

While keeping spanning tree as many as sub-nodes, sub-node selection is done in flow-basis. The sub-node is decided when the first packet in the flow arrives. The later packets use the same path that the first packet uses. We do not specify how to choose a sub-node in this paper.

![Figure 2. Spanning trees generated by layer-2 routing standards and ENDIST](image)

We consider STP, SPB and ENDIST and perfect routing in this paper. Figure 2 shows the spanning trees generated by STP, SPB and ENDIST respectively in a backbone network mentioned in Figure 1. The number on each link in Figure 1 indicates the link capacity. We assume that all links are full duplex and uses 10Gbps transmission medium only. For example, a link of capacity 50 has five 10Gbps channels. In Figure 2, solid lines indicate active links that constitute the spanning tree, and dotted lines mean inactive ones. The root node is colored in black.

STP establishes a single spanning tree as shown in Figure 2(a). SPB generates spanning trees as many as edge-nodes. Figure 2(b) shows two spanning trees rooted at the node A and B among seven spanning trees generated by SPB. The each edge U has four sub-nodes and all other 6 edge nodes have two sub-nodes respectively. As total there are 16 spanning trees in ENDIST. Figure 2(c) shows two routing paths rooted at node A generated by ENDIST. Lastly, we define the 'perfect' routing method that uses the entire links in network. The output result of perfect routing is the ideal upper bound in a network, and we this result compare with that of Layer-2 routing methods.

In this paper, we introduce an analytic model that evaluates the routing methods in backbone networks using linear programming (LP). The model solves the multi-input and multi-output maximum flow problem in backbone networks, and calculates maximum throughput the each routing method can reach. Therefore, if we have implemented version of ENDIST, we can check how close the reality is to the target with this paper. If implemented routing method performs significantly lower than the limit, we recommend some modifications or addition about the routing method to improve performance. Besides, the model suggests appropriate values of link capacity if network topology and traffic pattern is given. If we construct network based on this values, we can balance the traffic and network construction cost would be reduced.

There were a few researches that improves layer-2 routing protocols. The SmartBridge [7] learns network topology and forwards traffic through shortest path with reduced delay. SmartBridges should set up solution for load balancing and for interworking procedure in non-Smart Bridges and Smart Bridges during reconfiguration. STAR (Spanning Tree Alternate Routing) [8] provides multiple routing paths - an original spanning tree and alternative routes. Some traffic uses alternative routes which are shorter than the original spanning tree with reduced transmission delay. STAR needs to define reorganization process and prevention of link overload. ENDIST outperforms these two works and interworking problem is easy because it heavily relies on SPB.

LP is a convenient tool for optimization problem. Many networking problems especially MPLS traffic engineering problems [9, 10] are solved by LP. We introduce a few works of LP optimization problem in Ethernet and L2 networks. In [11] LP model choose routing path that uses minimum links within the defined capacitance. A link cost, result of LP optimization, is assigned to each bridge port. Referring to this link cost, each bridge set up spanning tree on a network. As a result, this method ensure that the class of high priority obtain shortest path and bandwidth. [12] introduces the method for providing Traffic Engineering (TE) in ethernet Virtual Private Network (VPN). Proposed method refer to current link load to provide TE. Thus this method is perfect in dynamic network condition. [13] solve the problem to design LAN topology being the most suitable. In this paper, LP model find out LAN topology that minimize delay.

This paper is organized as follows. Section 2 defines maximum flow problem to be solved by LP, and section 3 discusses on how to decide the link capacity in a given backbone network. Finally, in section 4, we analyze the simulation results using the analytic model proposed in section 3, and conclusions follow.

2. Linear Programming Formulation

In this section, we define the multi-input and multi-output maximum flow problem to be solved by LP considering four routing methods of STP, SPB, ENDIST, and perfect routing. We first explain variables for describing backbone networks and defining routing paths.

- $N_C$ : a set of edge nodes ($|N_C| = n_c$)
- $N_S$ : a set of core nodes ($|N_S| = n_c$)
- $N_E$ : a set of entire nodes ($N = N_E \cup N_C$)
- $u$ : the only gateway node to a upper level network ($u \in N_S$)
- $N_B$ : a set of busy edge nodes that access more traffic than non-busy edge nodes ($|N_B| = n_b$, $N_B \subset (N_E - u)$)
All traffic the non-busy have three classes of buffer overflow. Similarly all destination column amount is the total flow of entire links. Therefore the output scheme considered

We list up LP variables.

- $p(i,j,z)$: sum of flow to $z$ on the link $l(i,j)$ ($z \in N_g$, $i,j \in N$).
- $g(a,z)$: offered traffic from $a$ to $z$ ($a,z \in N_g$, $a \neq z$).
- $K(a)$: total flow generated by $a$ ($K(a) = \sum_{z \in N_g \setminus \{a\}} g(a,z)$).

We have some assumptions about the offered load. First, we do not consider traffic loss due to transmission error or buffer overflow. If an output buffer is full, the node does not receive packets that should be sent to the buffer. We assume three classes of edge nodes. The node $u$ is special because it is the only gateway to the upper level network. Except $u$, we have a busy edge nodes and non-busy ones. Both busy and non-busy nodes generate the same amount of traffic. So,

$$K(i) = K(j) \quad \forall i,j \in (N_g - u), \quad i \neq j. \quad (1)$$

All edge nodes send to the node $u$ with the upward rate

$$g(a,u) = e \cdot K(a), \quad a \in (N_g - u). \quad (2)$$

All nodes send $1 - e$ portion of traffic evenly distributed to all edge nodes but a busy edge node attracts twice much traffic than a non-busy node as a destination node. The node $u$ generates traffic to the other edge node with the same amount from each of them. Thus

$$g(u,z) = \frac{K(u)}{n_e + n_z - 1}, \quad z \in (N_g - N_z - u)$$

Similarly after calculating all cases of offered traffic $g(a,z)$, the result is sorted in Table 1 including Eq (2) and (3). The column of Table 1 shows three cases that a source edge node $a$ belongs to, and the row indicates three cases for a destination edge node $z(a \neq z)$.

| Destination $z$ | source $a$ | $u$ | $N_S$ | $N_z-N_u$-
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$u$</td>
<td></td>
<td>2 $\cdot$ $K(u)$</td>
<td>$n_e + n_z - 1$</td>
<td>$K(u)$</td>
</tr>
<tr>
<td>$N_S$</td>
<td></td>
<td>$e \cdot K(a)$</td>
<td>$\frac{2 \cdot (1 - e) \cdot K(a)}{n_e + n_z - 3}$</td>
<td>$(1 - e) \cdot K(a)$</td>
</tr>
<tr>
<td>$N_z-N_u-u$</td>
<td>$e \cdot K(a)$</td>
<td>$\frac{2 \cdot (1 - e) \cdot K(a)}{n_e + n_z - 2}$</td>
<td>$(1 - e) \cdot K(a)$</td>
<td>$n_e + n_z - 2$</td>
</tr>
</tbody>
</table>

Using the offered load constraints at Table 1, we are going to define the multiple-source multiple-destination maximum flow LP model that calculates the maximum throughput of backbone networks. Due to flow conservation, sum of inflow is same as sum of outflow for each node. Depending the node to which this rule is applied, we obtain three equations to a source node $a$, to a destination node $z$, and to a core node $i$ respectively,

$$\forall a,z \in N_g, \quad a \neq z, \quad \forall i \in N_C, \quad (4)$$

$$\sum_{(i,j) \in B(a,z)} p(i,j;z) = g(a,z), \quad (5)$$

$$\sum_{(j,i) \in B(a,z)} p(j,i;z) = g(a,z), \quad (6)$$

Since sum of any kind traffic on a link cannot exceed its link capacity, we have

$$\sum_{(i,j) \in B(a,z)} p(i,j;z) \leq c(i,j), \quad (7)$$

The routing scheme under consideration determines the routing variable $B(a,z)$. In perfect routing, $B(a,z)$ is the entire link set $L$. In STP the routing variable is represented by $B^{*}(b,k)$ because there is a single spinning tree as shown in Figure 2(a) and all traffic use this spanning tree disregarding source and destination. Figure 2(b) shows $B^{*}(a)$ and $B^{*}(b)$ used in SPB. Figure 2(c) describes $B^{*}(a)$ and $B^{*}(b)$ used in ENDIST. All traffic to $A$ can use these two spanning trees in ENDIST.

The objective function of maximum flow is expressed as

$$\text{Maximize} \quad \sum_{(a \in N_g)} K(a). \quad (8)$$

The LP finds out the maximum value in (8) using Table 1 and equations (4) to (7).
3. Capacity Assignment

In two backbone networks of Figure 1 and 3, all links specify link capacities. This section introduces how to declare them using Figure 1.

We assigned the same capacity to the same level of links in a backbone network. Link level is defined as follows. All nodes in the backbone network are divided into several levels. So, in Figure 1 all nodes in the backbone network divided into four levels such as U, I, (G,J,K,H), (A,B,C,F,E,D). We named these levels as T1,T2,T3,T4. A topology in Figure 1 is composed of four types of links such as T1-T2, T2-T3, T3-T3, T3-T4. That is, {l(U,I),} { l(G,J), l(K,J) , l(I,J)}, { l(G,J), l(J,K), l(K,H), l(H,G)}, { l(A,H), l(A,G), l(B,G), l(B,J), ..., l(D,K), l(D,H)}.

First of all we assign 10 Gbps to T3-T4 link types, and decide link capacities of the reminder. We assumed that perfect routing is applied to a topology and entire edge nodes forward 100% load without loss. And find out minimum capacities of links. To solve this problem, abundant capacities are assigned initially, such as 100Gbps, to all links. Next, we will repeat to solve maximum flow problem in concurrency keeping reducing the assigned capacities. This method will determine link capacities differently about upward traffic ratio ", e". Concerning minimum capacities obtained in several values of e, we find out link capacities expressed in Figure 1.

4. Performance Evaluation

In this chapter we simulate the performance of each layer-2 routing method using ILOG CPLEX 9.0. And In this paper we consider symmetric and non-symmetric condition. Symmetric condition means there is no superior edge node (n_s = 0) in the network and asymmetric one means the existence of superior edge nodes (n_s > 0).

4.1. Simulation Environment

Since a topology in Figure 1 has simple structure, a performance difference of each routing method can not be distinguished. So we will study additional experiment in extended model. In this chapter, we make use of a topology expressed in Figure 3 in order to evaluate the performance of each routing method. For simplicity each topology in Figure 1 and Figure 3 are called 'small model' and 'big model' respectively. A big model has PBBN structure similar to a small model. And link capacities are assigned by the method in Figure 4.

We evaluate the performance of each routing method by two conditions. First the case 1 assume that there is no superior edge node (n_s = 0), and the case 2 means the existence of superior edge nodes (n_s > 0).

4.2. Results

In this chapter, we analyze the results obtained by simulating LP modeling in chapter 3. We compare proposed ENDIST routing method with others such as perfect routing, STP routing, SPB routing. And these routing methods are concisely expressed in perfect, STP, SPB, ENDIST in this chapter. First of all we evaluate performance in big model since the simulation results of small and big model have similar pattern. Lastly, we mention the comparison of maximum throughput in small and big model.

Figure 4-5 show maximum throughput for four routing methods and two cases of e that is the upward traffic ratio. In Figure 4-5, the axis of x is the relative load factor that is loaded to network. For example, 100% means 20Gbps. Because the bandwidth of each link connected to an edge node is 10Gbps, 100% of the axis of x means the full of two links that is connected to an edge node. The axis of y means the maximum throughput backbone network can process. Figure 4-5 show the performance of each routing method in e 0.2 and 0.5 respectively. As e increase, on the whole, the throughput of each routing method increase too. Especially, the increament of ENDIST is higher than other routing methods, and is similar to the performance of perfect that is ideal upper bound in the target network.

Figure 4(a) and 5(a) show the performance of each routing method under the condition that the upward traffic ratio e is 0.2, 0.5 respectively and the forwarding rule is n_s = 0. In the case of perfect, the output is expressed in a straight line crossing the origin point, because the entire traffic load is no loss under all the traffic load conditions. All the cases except the perfect have the curved lines that are expressed in the straight lines crossing the origin point under the light traffic load. Figure 4(b) and 5(b) display the performance of each routing method in e > 0 condition. In this case, the throughput of routing methods is decreased on all occasions than n_s = 0 condition in Figure 4(a) and 5(a). As the increament of ENDIST is lower, the performance gap with perfect grows smaller. Especially in the case of Figure 5 that
upward traffic ratio $e$ is 0.5 and if $n_z > 0$, forwarding in Figure 5(b) shows that ENDIST has the same performance as perfect.

Table 2. Comparison of layer-2 routing methods

<table>
<thead>
<tr>
<th></th>
<th>$e=0.2$ ($n_z = 0$)</th>
<th>$e=0.2$ ($n_z &gt; 0$)</th>
<th>$e=0.5$ ($n_z = 0$)</th>
<th>$e=0.5$ ($n_z &gt; 0$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENDIST</td>
<td>0.67</td>
<td>0.82</td>
<td>0.93</td>
<td>1.00</td>
</tr>
<tr>
<td>SPB</td>
<td>0.41</td>
<td>0.50</td>
<td>0.51</td>
<td>0.56</td>
</tr>
<tr>
<td>STP</td>
<td>0.19</td>
<td>0.25</td>
<td>0.16</td>
<td>0.19</td>
</tr>
</tbody>
</table>

In Table 2, we compare the performance of layer-2 routing techniques when output of perfect is the standard. Four column represent the output of the above graph in Figure 4 - 5 respectively. If the output of certain routing method close to 1, it is reached to the performance of perfect. We found that ENDIST is much superior to STP and SPB, and it shows almost perfect performance. Especially under highly upward traffic ratio $e$ and in $n_z > 0$, ENDIST shows more performance that close to perfect.

Figure 7 show the comparison maximum throughput in the small model and big model under upward traffic ratio $e$ is 0.5. Four bar graphs on the left are small model(S) and four bar graphs on the right are big model(B). The axis of $x$ is each routing method, and displays the output of $n_z = 0$ and

[[image of Figure 4(a). big model, e=0.2, ns = 0]]

[[image of Figure 4(b). big model, e=0.2, ns > 0]]

[[image of Figure 5(a). big model, e=0.5, ns = 0]]

[[image of Figure 5(b). big model, e=0.5, ns > 0]]

[[image of Figure 5(b). big model, e=0.5, ns > 0]]

[[image of Figure 7. Comparison of maximum throughput in the small model(S) and the big model(B)]]
$n_y > 0$ all together. The axis of $y$ means the maximum throughput backbone network can process. Generally, the throughput of small model is lower than that of big model, but in small model the performance of ENDIST equal to that of perfect. That is, we can know that ENDIST has more ideal performance especially in small sized network.

In this chapter, we obtain the following conclusion. ENDIST is superior routing technique than existing layer-2 routing technique in the aspect of throughput. Especially, the throughput of ENDIST can reach to that of perfect when upward traffic ratio is 0.5 and $n_y > 0$ condition is assumed.

In the actual backbone network, most traffic is likely to concentrate on specified nodes. And much traffic is being forwarded to external backbone network. Thus we can know that ENDIST is suitable routing method in backbone network.

Acknowledgements
We appreciate professor “Dong-ho Lee” at Hanyang University and his laboratory students for us to use LP packages.

6. Conclusions

This paper proposes a new analysis tool that evaluates the layer-2 routing schemes in a backbone network. We obtain this tool by solving the multi-input and multi-output maximum flow problem in backbone networks using linear programming (LP). We can use this LP routing model as follows.

Firstly, we use the LP model in network planning. This model can calculate benefits of investment over total investment or the utilization efficiency. Repeatedly applying this model with different link capacities, we can find the appropriate link capacity on the given topology. And we use the network with calculated link capacities as a target network in section II and IV.

Secondly, the LP model informs us of the theoretical upper bound of a given layer-2 routing method. Since this result is achieved without details of the routing operation, we can check how close reality is to the target. If implemented routing scheme performs significantly lower than the limit, we recommend some modifications or addition about the routing scheme to improve performance.

Thirdly, using the LP model, we can compare layer-2 routing methods in terms of maximum throughput. We select two famous layer-2 routing schemes - STP, SPB, and our proposed routing scheme - ENDIST. Additionally we include the perfect routing. As a result, we found that ENDIST is much superior to STP and SPB, and that it shows almost perfect performance under light and medium traffic load. This implies that ENDIST is as good as layer 3 routing from the point of maximum throughput.

REFERENCES