Intelligent Load Balancing for Shortest Path Bridging

David Allan, János Farkas, Scott Mansfield

***Abstract*— This article provides an overview of a method to enhance the load spreading algorithms used by IEEE 802.1aq Shortest Path Bridging. The technique has a number of interesting properties with respect to distributing load on the basis of a measure of link utilization. What this provides is balanced network utilization and the ability to apply subtle modifications to the distribution of the traffic matrix previously unachievable with shortest path forwarding alone. This is achieved within the framework of the existing Ethernet technology base.**

I. Introduction

IEEE 802.1aq Shortest Path Bridging (SPB) [1] is a singularly interesting networking technology. It is the application of contemporary compute power and a state of the art control plane to replace spanning tree protocols. In essence it is a combination of standard and proven components that have been specified by the IEEE with algorithms that construct connectivity based on the shortest paths. This is achieved while preserving key aspects of the Ethernet architecture. The basics of the technology have previously been discussed in this journal in [4] and [5].

SPB takes advantage of key attributes of the design of Ethernet to permit the consolidation of a number of control protocols into the combination of a single link state routing protocol (IS-IS) [2] and enhanced algorithms for the computation of both unicast and multicast forwarding. One consequence of this consolidation is a radically simplified control plane with a significantly diminished messaging load. This becomes a virtuous circle as reducing the number of messages exchanged as an artifact of normal network operation diminishes the number of interruptions the control plane is required to handle when computing.

Another key attribute of the Ethernet technology base is the concept of Virtual LAN (VLAN) as a vertical partition of the network. That is, each VLAN may comprise a different subset of the physical topology and SPB ensures full connectivity on a shortest path within each VLAN. SPB does enable Ethernet to overcome the limitations of spanning trees and fully utilize much richer topologies. The combination of path computation, shortest path forwarding and the use of VLANs opens up new possibilities in the efficient use of the breadth of connectivity within the bounds of the existing Ethernet technology base.

II. The Problem Space

SPB is expected to make efficient use of a variety of arbitrary networking topologies with an absolute minimum of operator intervention. These can range from sparse mesh or dual homed hub and spoke in the metro up to Clos or “fat tree” [6] architectures employed in the datacenter. Frequently, the offered load in the network will not match the topology or routing system resulting in “hot spots”. Operators need the ability to shift load away from a hot spot.

It is well understood that a single shortest path solution in the network will not make good utilization of a network, and is difficult to manipulate due to the sensitivity of shortest path forwarding to metric manipulation. This has led the industry to implement multi-path solutions.

Most frequently deployed today is per hop Equal Cost Multipath (ECMP). In most implementations, when the presence of multiple equal cost next hops is encountered, the packet is inspected for a source of entropy (typically an IP header) and a hash of header information modulo the number of next hops is used to select the next hop for the particular packet. It will distribute load well in very regular topologies, but for asymmetric topologies or failure scenarios the distribution of load will be perturbed across the network as the load spreading view is based only on a view of the next hop and not the network beyond the next hop. Reactive measures to address hot spots are difficult as a multi-path design tends to “lock in” the metrics used; hence shifting load off a specific link or node is not easily addressed. Further, ECMP frequently requires additional packet processing inherent to the next hop selection process. Finally as far as Ethernet is concerned this would be a significant change to existing implementations and a major architectural change as congruence of forward-backward and unicast-multicast forwarding would be lost, and data plane OAM [3] would require a significant re-design. The reason for this is that per-hop ECMP cannot be applied to multicast. So while ECMP preserves flow ordering for unicast, virtualized bridging utilizing a combination of unicast and multicast (for flooding of frames with unknown destination) could not make that claim.

As noted Ethernet today does not have per hop multipath support. What is in 802.1aq is the ability to generate multiple equal cost tree (ECT) solutions (known as ECT sets) instantiating each as a distinct VLAN and permitting edge assignment of load to these VLANs. The path permutations are generated via algorithmic manipulation of the node IDs used in tie breaking for multiple equal cost paths and a framework (rather wisely provided) for the definition and coordination of deployment of future algorithms.

The algorithmic manipulation of node IDs will perform path selection on a pseudo random basis and in a way that does not actually guarantee path diversity, hence a dilation factor is required (creating more paths in the routing system and forwarding database than exist in reality) to ensure good link utilization. It has the other consequence of driving up the amount of state in the Filtering Databases (FDB) without guaranteeing true load diversity.

Ethernet does allow each VLAN to be manipulated as if it was a distinct topology. Multi-topology extensions to IS-IS could be employed to manage the independent VLAN topologies. However, this requires extensive administration to work (the modification of metrics from the defaults they would normally assume) and would be specialized to a specific topology. This is contrary to the simplicity we are trying to achieve.

A further observation is that the techniques for engineering routed networks today have largely centered on artificially increasing the mesh density via virtual links (e.g. MPLS Forwarding Adjacencies), a technique again not well suited to, and requiring modifications to the existing Ethernet technology base.

So our design objective is to provide a load spreading algorithm that while leveraging the existing Ethernet technology base would find useful path diversity. What we found is the proper way to do this is to consider summing potential load offered to the network and seeking to place the paths to minimize the per-link deviation in load across the network.

III. The Algorithm

If the physical topology provides multiple equal cost paths between a pair of nodes, then the Dijkstra algorithm applied for shortest path computation has to be extended with tie breaking. The existing tie breaking algorithm specified for 802.1aq has a number of desirable properties that are useful to replicate in any new solution going forward. These are primarily focused on ensuring consistency and symmetry for forwarding paths across the network. A consequence of this algorithm is that any portion of the shortest path is also the shortest path, a property leveraged further by SPB.

What we have done is modified the set of information used by the algorithm while preserving its key properties. What we are doing is leveraging the ability to instantiate multiple virtual topologies. We apply a model that incorporates the link load computed during previous iterations of path generation into the tie breaking of subsequent iterations in order to even out the anticipated loading of links in the network. The operation of the algorithm will inherently favor choosing links transited by fewer shortest paths for subsequent iterations of the algorithm.

A simplified view of the algorithm is as follows:

The initial set of shortest paths is determined as the intersection of the set of paths between any two points with the minimum metric cost and fewest hops. Tie breaking is performed via lexicographically ranking the path IDs of the set of shortest cost paths, where the path ID is the lexicographic sorted list of node IDs. The tie resolves to selecting the path with the lowest path ID.

During the course of generating the initial path set, all node pairs are considered, and the number of times each link appears on a selected pairwise shortest path is recorded. This number is referred to as the Ethernet Switched Path (ESP) count.

The second and subsequent rounds of tie breaking as further ECT sets are generated is performed based on a two stage ranking of the equal cost paths. The initial ranking is generated from the sum of the ESP counts for each path. If there are multiple paths with the same lowest sum, then the lexicographic order is used as the tie-breaker.

The path selection of the second and subsequent passes through the database will ensure the paths with the least anticipated load tend to be selected. Repeated passes through the database utilize the cumulative ESP counts of the previous iterations, hence the paths with least cumulative anticipated load will consistently be selected.

For example, consider the network fragment below in Figure 1 with 6 nodes: 1, 2, 3, 4, 5 and 6:

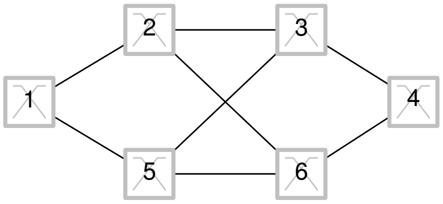

Figure 1 Example topology with equal cost shortest paths

Initially no path pairs have been determined and the ESP count reflecting the link load is set to zero for each link. Examining the set of paths of equal cost between 1 and 4 will generate the following ranked set of four path IDs:

Unsorted	Sorted & Ranked
1-2-3-4	1-2-3-4
1-2-6-4	1-2-4-6
1-5-3-4	1-3-4-5
1-5-6-4	1-4-5-6

The initial pass selects 1-2-3-4 as the low ranked path. To keep this example simple only the paths between nodes 1 and 4 were considered in determining the ESP count transiting each link. So in this example the links in the selected path would be prefixed with a ESP count of 1. For the next pass through the database this would yield the following ranked set of paths based on the sum of the ESP counts:

Path ID	Sum of ESP counts
1-4-5-6	0
1-3-4-5	1
1-2-4-6	1
1-2-3-4	3

One path uniquely has the lowest sum of ESP counts which is 1-4-5-6 and hence is the one selected. Updating the ESP counts after selecting 1-4-5-6 provides a new ranking:

Path ID	Sum of ESP counts
1-2-4-6	2
1-3-4-5	2
1-2-3-4	3
1-4-5-6	3

At this point there is a tie for lowest sum of ESP counts, hence the link IDs are used to resolve this tie further. For the two paths tied for the lowest sum of ESP counts the lowest ranked on the basis of path IDs is 1-2-4-6.

Whereas the example only considered one path pair per iteration of the database in assigning link load, it is easy to envision that after a pass of the database and considering all node pairs, a comprehensive view of the potential traffic distribution exists, and as the tie breaking of subsequent passes will inherently favor the minima, therefore, how the load is distributed across the network will tend to be evened out. As one would expect, the degree of modification proportionately diminishes with each new set of paths considered as the effect is cumulative.

IV. Results

To fully model the network, each node has to compute the shortest path solution for all nodes, which is referred to as all pairs shortest path. Depending on the implementation, an "all pairs" pass of the database can be an expensive proposition in terms of computational complexity. So if this technique required a significant number of passes through the database in order to yield measurable benefits, the technique would be of little value in a system needing a real time response.

For highly regular "fat tree" architectures, if the number of ECT sets generated corresponded to the number of top tier nodes and uplinks from each node, the load distribution would also be very regular; one path between each node pair would traverse each link. That is easily demonstrated. What was more interesting was how the technique would work on arbitrary topologies. So to evaluate the technique we utilized the Erdős-Rényi (E-R) model [7] in order to evaluate the technique against random graphs and gain some insight into the capabilities of the technique. We used the *G(N,P)* model, where in an *N*-node topology each link is included with probability *P* independently from the inclusion of any other link. The E-R model we used had equal metrics for all links and therefore the more lightly meshed the network, the larger the number of multi-hop paths, which tended to diffuse the effectiveness of the load spreading.

We set out to measure how successfully the technique evened the anticipated load on all links in the network. Or to express it formally, to how successfully we minimized the coefficient of variation (the ratio of standard deviation to the mean value) in the number pairwise ESPs that transited each of the links in the network once all ECT sets had been generated.

The results in Figure 2 show the Average Number of Shortest Paths between all node pairs vs. the Coefficient of Variation (CV) of the number of shortest paths that transited each link for a 150-node network, where the ESP count is zero for the first pass and the subsequent passes tie break on the cumulative ESP counts of the preceding passes. The Average Number of Shortest Paths is determined by averaging the number of equal cost shortest paths between each node pairs in the E-R topology generated for a given *P* value.

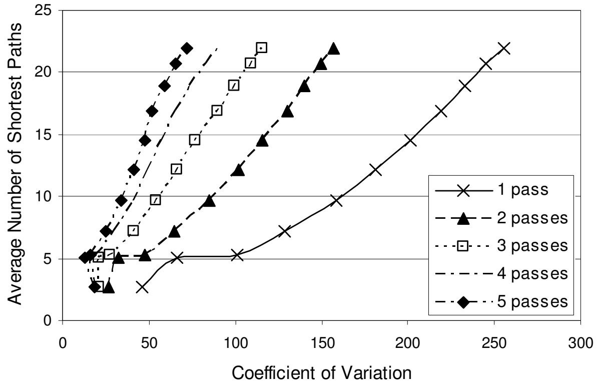

Figure 2: Actual Coefficient of Variation for randomly generated 150-node networks

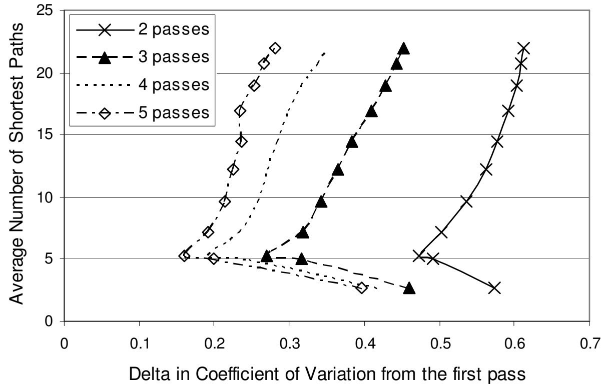

Figure 3: Improvement in Coefficient of Variation from Pass to Pass for the same randomly generated 150 node networks (baseline is a single pass)

Figure 3 illustrates the relative improvement in network load leveling in proportion to the mesh density and the number of passes through the database performed. As the figure shows, increasing the mesh density by increasing *P*, thus increasing the number of equal cost paths, but having the same number of passes on the database increases diversity in link utilization. The curve for the single pass is the most illustrative for this because the tie breaking is in fact only based on node IDs and no load value is taken into account thus hot spots appear. On the other hand the figures clearly show that increasing the number of passes on the database for a given *P* improves load spreading as the coefficient of variation decreases.

There are some artifacts to note. The first is the rise in CV as the average number of shortest paths becomes larger than the number of ECT sets. Similarly the CV rises when the average number of shortest paths is small and diminishing, as the technique has little to work with. These artifacts combined with the random graph nature of the E-R model producing the obvious "sweet spot" where the results of the technique are maximized. This occurs at around an average of 5 shortest paths between any two points in the network in the examples modeled.

The graph does not illustrate the effect above an average number of shortest paths of 22. This was the maximum in the average number of shortest paths the model generated. This occurred at about *P*=0.6 beyond which the number of direct adjacencies began to dwarf the number of multi-hop paths and both the average number of shortest path paths and CV diminished correspondingly as again the technique depends on breadth of connectivity, and that was diminishing.

The most interesting and important result is that the majority of the benefit accrues in the first two passes, i.e. when the first two sets of shortest path trees generated from the database. What we typically see from the first to second pass in any reasonably meshed network is a 40-45% reduction in the Coefficient of Variation of link load across the network. This in itself is not surprising as we have doubled the number of paths through the system explicitly placing the second set to avoid the loading of the first. Further, the rate of improvement of the coefficient of variation drops off from pass to pass along the lines that the 1/2, 1/3, 1/4 series that the cumulative path count would suggest.

What this means is that significant results can be achieved while keeping the technique tractable in terms of both computation and forwarding entries as the set of ECTs resulting from each pass through the database is instantiated in a unique VLAN.

V. ADDITIONAL DISCUSSION

For simplicity of interpretation the results presented are based on generating a single ECT set from each pass of the database. This also produces the lowest CV for the lowest amount of state in the FDBs but at the expense of computational complexity and increasing the impact of a topology change. This is because all path selection is dependent on the selection of the first ECT set.

The technique of selecting low/high ranking described in the current 802.1aq specification permits selection of two or more paths per pass of the database. This would typically result in a lower CV per pass of the database and as not all path selections chained off a single path, and greater stability across topology changes. However the amount of state required for a given improvement in load spreading would increase and if several paths were selected per pass, could increase dramatically in proportion to the improvement in overall network performance due to the "blind" nature of how more than two paths per pass are selected.

With respect to network stability, one of the more interesting properties of the technique proposed here is that because it is effectively connection oriented, and seeks out the least loaded links, any perturbation of the traffic matrix caused by a failure tends to be isolated and local in nature. The mechanism will tend to steer the traffic back into the original distribution once a constriction in the network has been bypassed, and the original lexicographic tie breaker will provide a significant degree of path stability in the unperturbed parts of the network.

Another property to consider is that any individual pass of the database permits the Dijksta to be striped across multiple processing elements. Further a complete "all pairs" computation of the final pass is not required and algorithmic improvements such as those proposed in [8] can be exploited.

The most important property of the technique is that it works with the existing Ethernet technology base unmodified, hence preserves the architecture and service guarantees. Further OAM as specified in [3] just works.

VI. More Precisely Modeling the Traffic Matrix

The example above simply considered the count of node pairs for which the shortest path transited a link and a network wide common link metric. Sufficient information exists in the ISIS-SPB control plane to model either the physical topology or the offered load to a much finer level of granularity. Each node registers interest in specific service instances via IS-IS, and with the simple augmentation of such registrations with a traffic descriptor, a near exact model of the traffic matrix would exist and could be taken into account when determining the load for tie breaking.

The question becomes what level of detail is useful? Constantly tweaking the network in response to any service change will be constantly perturbing the distribution of the matrix, likely to little benefit. The goal should be to get it approximately right and then leave it alone as much as possible.

On that basis what does make sense is to consider the load that can be exchanged between a node pair when weighting the link IDs for tie breaking. Hence taking the lower metric of the uplinks from each node as the factor to consider for link weighting would appear to provide a reasonable representation of the matrix that would have a high degree of stability. Procedures can then be designed to map services to the topologies on the basis of anticipated matrix to further enhance the stability of the network.

VII. Manipulating The Distribution of The Traffic Matrix

Given we are looking to a relatively generalized modeling of load for network operation, it is then reasonable to consider that we will see “hot spots” where the combination of routing and physical topology is not aligned with the current network loading.

An interesting attribute of the technique is that a new tool becomes available that can help to address the hot spot problem too. It becomes possible to “pre-bias” a link with a load factor which will have the effect of shifting some load away from the particular link when considered in the tie breaking process. This permits significantly subtler gradations of manipulation of routing behavior to be achieved when compared with link metric modification, much simpler administration than multi-topology routing, and obviates the need for link virtualization to drive up the mesh density. Furthermore, shortest path forwarding is always maintained with this technique, and the freedom of using link metric for other purposes remains.

Further, the technique still works within the confines of the techniques minimization of the coefficient of variation of link use, so whatever is displaced will be rationalized with the distribution of the rest of the traffic matrix. Another interesting property specific to the operation of shortest path bridging is that as no distance to a root will change as a consequence of the technique, all existing loop avoidance agreements will remain in affect, accelerating convergence after any operational changes.

VIII. Summary and Future Work

Whereas SPB to date has been about the substitution of computation for control plane complexity, this technique is a practical example of how computation can also displace the need for data plane complexity. Occasional brief periods of a bit more complex computation can replace the need for deep packet inspection, hashing and per hop steering in order to achieve efficient load spreading. Hence implementation complexity, additional power consumption on a per-packet per-hop based processing and a whole raft of intractable OAM problems are completely circumvented while delivering superior utilization of network resources.

Future work in this area will explore further algorithm optimizations as we continue to seek to optimize the performance of SPB in all dimensions.

References

[1] IEEE Std. 802.1aq D3.6, “IEEE Draft Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks – Amendment 9: Shortest Path Bridging,” February 2011.

[2] ISO/IEC 10589, Information Technology – Telecommunications and Information Exchange Between Systems – Intermediate System to Intermediate System Intra-Domain Routing Information Exchange Protocol for Use in Conjunction with the Protocol for Providing the Connectionless- Mode Network Service (ISO 8473),” 2nd ed., 2002.

[3] IEEE Std. 802.1ag, “IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks – Amendment 5: Connectivity Fault Management,” December 2007.

[4] D. Allan, P. Ashwood-Smith, N. Bragg and D. Fedyk, "Provider Link State Bridging," IEEE Communications Magazine, September 2008.

[5] D. Allan, P. Ashwood-Smith, N. Bragg , J. Farkas, D. Fedyk, M.Ouellete, M. Seaman, P. Unbehagen "Shortest Path Bridging: Efficient Control of Larger Ethernet Networks," IEEE Communications Magazine, October 2010.

[6] M. Al-Fares, A/ Loukissas and A. Vahdat, “A scalable, commodity data center network architecture,” In Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, August 2008.

[7] P. Erdős and A. Rényi, “On Random Graphs. I.” Publicationes Mathematicae 6: p290-297, 1959.

[8] J,. Chiabaut and N Bragg, “Speeding up the SPB Computation,” IEEE contribution: http://ieee802.org/1/files/public/docs2009/aq-nbragg-fast-spf-for-SPB-1109-v01.pdf

David Allan (david.i.allan@ericsson.com) is a Distinguished Engineer at Ericsson, and a former distinguished member of technical staff at Nortel. He has been active in data telecommunications standards for the past 15 years.. He has been active for over 25 years as an architect, design engineer, and developer of real-time systems in diverse areas of technology ranging from process control and avionics to financial transaction processing. His current role at Ericsson is focused on carrier infrastructure based on MPLS and Ethernet. He has a B.Eng. (1978) from Carleton University in Ottawa.

János Farkas (janos.farkas@ericsson.com) is a research engineer at Ericsson Research since 1999. He is an active contributor to the IEEE 802.1aq Shortest Path Bridging project. He has focused on carrier transport networks based on Ethernet in the past few years. His former research activities include IP QoS solutions for radio access networks and network traffic management. He has a Master’s degree in electrical engineering from the Budapest University of Technology and Economics.

Scott Mansfield (scott.mansfield@ericsson.com) Mr. Mansfield is Lead Architect in Ericsson’s DUIB Technology, Network Architecture group. He is responsible for Metro Ethernet Forum (MEF) coordination for Ericsson and is the editor of the MEF SOAM PM Document. He is currently focused on Carrier Ethernet and MPLS-TP Network Management standards and architectures. Previously, he has focused on usage mediation and performance-monitoring requirements for ATM, IP and MPLS based networks. He has a B.S. (1985) from The Pennsylvania State University.