Improving the Efficiency and Obtaining Optimality in Link State Routing

Kushal Hittanalli[1] Anitha R[2]
[1]M.Tech. Student [2]Associate Professor
[1]Department of Computer Engineering and Applications [2]Department of Computer Science & Engineering
[1,2]The National Institute of Engineering, Mysore, Karnataka, India

Abstract— Efforts are being made since decades to improve the efficiency of routing, particularly link state routing. Our paper aims in making the routing optimal by reducing the cost of carrying traffic through packet switched networks. Link state routing protocols like OSPF and IS-IS converge more quickly than the distance vector routing protocols such as RIPv1, RIPv2, EIGRP and others. This is achieved using flooding and triggered update techniques. In link state protocols the updates are flooded immediately and computed in parallel. Triggered updates improve convergence time by making routers send update messages immediately upon learning of a route change. These updates are triggered by events such as new link becoming available for the failure of existing link. Main drawbacks are CPU overhead and memory requirements. Our technique iteratively and independently updates the fraction of traffic destined to the node that leaves the source node on each of its outgoing links. For every iteration the updates are calculated based on shortest on each destination as determined by the marginal cost of the networks links. These costs are obtained by link state updates which are flooded through the network after each iteration. The technique is adaptive which automatically converges to the new optimal routing assignment for apparently static network changes. Main aim is to eliminate the tradeoff between optimality and ease of implementation and routing.
Key words: IP networks, Hop by hop, optimal routing, adaptive routing, link state

I. INTRODUCTION

Routing is the process of selecting best paths in a network. In the past, the term routing was also used to mean forwarding network traffic among networks. However this latter function is much better described as simply forwarding. Routing is performed for many kinds of networks, including telephone network (circuit switching), electronic data networks (such as the Internet), and transportation networks. This work is concerned primarily with routing in electronic data networks using packet switching technology.

The hop count refers to the number of intermediate devices (like routers) through which data must pass between source and destination, rather than flowing directly over a single wire. Each router along the data path constitutes a hop, as the data is moved from one Layer to another. Hop count is therefore a basic measurement of distance in a network. Hop count is a rough measure of distance between two hosts. A hop count of *n* means that *n* gateways separate the source host from the destination host.

There are several protocols used in data networking to describe the capability of a network to 'route around' damage, such as loss of a node or a connection between nodes:

- RIP
- OSPF
- IS-IS
- IGRP/EIGRP

Systems that do not implement adaptive routing are described as using static routing, where routes through a network are described by fixed paths (statically). A change, such as the loss of a node, or loss of a connection between nodes, is not compensated for. This means that anything that wishes to take an affected path will either have to wait for the failure to be repaired before restarting its journey, or will have to fail to reach its destination and give up the journey.

II. IMPORTANT DEFINITIONS

A. *Optimal Routing:*

It can be defined as the process of finding the routes that minimize the cost of carrying traffic through packet switched networks. Since the early 1970s, since the advent of ARPANET this has been the fundamental area of research and practical interest.

B. *Link State Routing:*

A link-state routing protocol is one of the two main classes of routing protocols used in packet switching networks for computer communications (the other is the distance-vector routing protocol). Examples of link-state routing protocols include open shortest path first (OSPF) and intermediate system to intermediate system (IS-IS).

C. *Dynamic Routing:*

Dynamic Routing, also called adaptive routing, describes the capability of a system, through which routes are characterized by their destination, to alter the path that the route takes through the system in response to a change in conditions. The adaptation is intended to allow as many routes as possible to remain valid (that is, have destinations that can be reached) in response to the change. The traffic demand matrix is not required by the algorithm as an explicit input to compute link weights. The algorithm recognizes and adapts to the changes in the network topologically and according to the traffic variations.

D. *Link state:*

The periodically flooded link state updates helps the routers to have the knowledge of state of all the network links. Routing decisions are made based on the states of the link states.

III. OSPF

OSPF which stands for Open Shortest Path First is a routing protocol which was developed for Internet Protocol (IP) networks by the Interior Gateway Protocol (IGP) working group of the Internet Engineering Task Force (IETF). OSPF can operate within a hierarchy, unlike RIP. The largest entity within the hierarchy is the autonomous system (AS), the autonomous system is a collection of networks under a common administration which share a common routing strategy. OSPF is an intra-AS routing protocol, even though it is capable of receiving routes from and sending routes to other ASs.

An autonomous system can be divided into different areas, which are groups of different networks and their attached hosts. Routers which have multiple interfaces can participate in multiple areas. These routers, which are called Area Border Routers, help in maintaining separate topological databases for each area. An area's topology is invisible to entities outside the area. By keeping the area topologies separate, OSPF helps in passing less routing traffic than it would if the autonomous system were not partitioned.

The technique of area partitioning creates two different types of OSPF routing, depending on whether the source and the destination are in the same area or different areas. Intra-area routing occurs when the source and destination are in the same area; inter-area routing occurs in the cases where they are in different areas.

An OSPF backbone is responsible for distributing routing information between areas. It comprises of all Area Border Routers, networks not wholly contained in any area, and their attached routers.

IV. PEFT

This routing algorithm gives a solution to an open question with a positive answer: Optimal traffic engineering or optimal multi-commodity flow can be obtained using link-state routing protocols along with hop-by-hop forwarding technique. Present typical versions of these protocols, Open Shortest Path First (OSPF) and Intermediate System-Intermediate System (IS-IS), split traffic evenly over shortest paths based on link weights also known as marginal weights of the links. However, the process of optimizing link weights for Open Shortest Path First (OSPF) and Intermediate System-Intermediate System (IS-IS) to the offered traffic is a known to be a NP-hard problem, and even the best setting or adjusting of the weights can deviate quite significantly from an expected optimal distribution of the traffic. In PEFT, the process includes splitting traffic over multiple paths with an exponential penalty on longer paths. Unlike the previous forms, DEFT, PEFT protocol provably achieves optimal traffic engineering also retaining the simplicity of the hop-by-hop forwarding. This protocol also leads to a significant reduction in the time needed to compute the best link weights. Both the protocol and the computational methods are developed in a conceptual framework, called Network Entropy Maximization that is used to identify the traffic distributions that are both optimal and realizable by link-state routing.

V. GALLAGER'S ROUTING

One of the most famous and adaptable routing algorithm is Gallager's algorithm for minimum delay routing. The merit of Gallager's routing is its rigorous mathematical approach to a problem, which is more often taken care of using heuristics. The approach is founded on a very well designed mathematical network model, which is customized to describe the minimum total delay routing problem. Mathematical observations on the model lead to two conditions for achieving global optimization, which are based on the marginal delay of links and neighbors. From these observations and conditions an iterative and distributed routing algorithm is naturally derived. Gallager finished his analysis by proving in detail that the algorithm achieves total minimum delay routing.

VI. THE EXISTING SYSTEM AND THE PROBLEMS

The information routers require to build their databases is provided in the form of Link State advertisement packets (LSAP). Routers do not advertise their entire routing tables, instead each router advertises only its information regarding immediately adjacent routers. The existing algorithms are good but not optimal like what is expected. The main area of interest in networking presently is improving the efficiency of the network by making best use of the available resources. Due to poor resource utilization from OSPF, network administrators are forced to overprovision their network in order to handle peak traffic. Most network links run at just 30%-40% utilization.

Link State protocols in comparison to Distance Vector protocols have:

- Bigger memory requirements.
- Shortest path computations require many CPU circles.
- If network is stable little bandwidth is used; react quickly to topology changes.
- Announcements cannot be "filtered". All items in the database must be sent to neighbors.
- All neighbors must be trusted.
- Authentication mechanisms can be used to avoid undesired adjacencies.
- No split horizon techniques are possible.

VII. IMPROVING THE EFFICIENCY

Improving the efficiency includes reducing the cost of carrying traffic from source to destination, reducing the memory requirements, time required to calculate different routes and the increasing the speed of transmission. Our goal in this paper is to eliminate the tradeoff between optimality and ease of implementation in routing. The result is a routing process which includes: Hop-by-hop, Adaptive, Link-state, Optimality, a routing solution that retains the simplicity of link-state, hop-by-hop protocols and also iteratively converging to the optimal routing assignment. This is a optimal link-state hop-by-hop routing solution. Not surprisingly, there are multiple challenges to overcome when designing such a solution. This gives us many advantages like using hop by hop method reduces the cost of carrying traffic from source to destination. The adaptive routing makes the data or traffic adapt to the changes in the network as and when the failure of the links or nodes occur, new routes are selected when the failure of present route occurs due to problems in the network.

In packet switching networks, routing directs packet forwarding (the transit of logically addressed network packets from their source toward their ultimate destination) through intermediate nodes.

Intermediate nodes are typically network hardware devices such as routers, bridges, gateways, firewalls,or switches. General-purpose computers can also forward packets and perform routing, though they are not specialized hardware and may suffer from limited performance. The routing process usually directs forwarding on the basis of routing tables which maintain a record of the routes to various network destinations. Thus, constructing routing tables, which are held in the router's memory, is very important for efficient routing. Most routing algorithms use only one network path at a time. Multipath routing techniques enable the use of multiple alternative paths.

As hop count is a rough measure of distance between two hosts. A hop count of *n* means that *n* gateways separate the source host from the destination host. By itself, this metric is, however, not useful for determining the optimum network path, as it does not take into consideration the speed, load, reliability, or latency of any particular hop, but merely the total count. Nevertheless, some routing protocols such as RIP use hop count as their sole metric.

Each time a capable device receives these packets, that device modifies the packet, incrementing the hop count by one. In addition, the device compares the hop count against atime to live limit and discards the packet if its hop count is too high. This prevents packets from endlessly bouncing around the network in the event of routing errors. Routers are capable of managing hop counts, but other types of intermediate devices (e.g. hubs and bridges) are not.

As discussed earlier dynamic routing, also called adaptive routing, is the capability of a system, through which routes are characterized by their destination, in order to alter the path that the route takes through the system in response to the changes in conditions. The adaptation is intended to allow as many routes as possible to remain valid (that is, have destinations that can be reached) in response to the change.

People using a transport system can display dynamic routing. For example, if a local railway station is closed, people can alight from a train at a different station and use another method, such as a bus, to reach their destination. Another example of adaptive routing can be seen within financial markets. For example, ASOR or Adaptive Smart Order Router (developed by Quod Financial), takes routing decisions dynamically and based on real-time market events.

A link-state routing protocol is one of the two main classes of routing protocols used in packet switching networks for computer communications (the other is the distance-vector routing protocol). Examples of link-state routing protocols include open shortest path first (OSPF) and intermediate system to intermediate system (IS-IS).

The link-state protocol is performed by every switching node in the network (i.e., nodes that are prepared to forward packets; in the Internet, these are called routers). The basic concept of link-state routing is that every node constructs a map of the connectivity to the network, in the form of a graph, showing which nodes are connected to which other nodes. Each node then independently calculates the next best logical path from it to every possible destination in the network. The collection of best paths will then form the node's routing table.

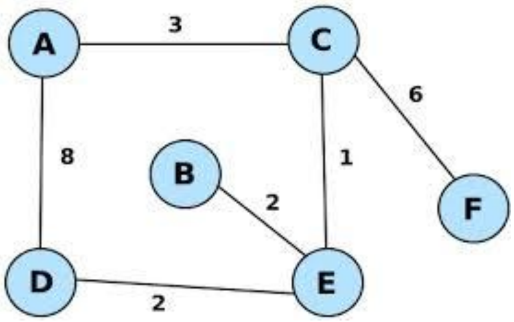

Fig. 1: A simple network consisting of nodes and paths

Above figure shows a simple network which consists of six nodes and six paths or links connecting them as shown in the figure. Each path has a value assigned to it. This may be considered as the path value or the cost. This cost helps to decide the shortest path among the available alternative paths to the destination. The data packets may travel through one or more paths to reach the destination.

Over the years, due to its importance, traffic engineering has attracted a lot of research attention. We provide a brief overview of major related results from different communities such as control, optimization, and networking. Broadly, the existing work can be divided into OSPF-TE, MPLS-TE, traffic demand agnostic/oblivious routing protocol design, and optimal routing algorithms.

The work on OSPF has concentrated on using good heuristics to improve the centralized link weight calculations. Although these techniques have been shown to improve the algorithm’s performance significantly by finding better weight settings, the results are still far from optimal.

Typically, these and other centralized traffic engineering techniques also require reliable estimates or measurements of the input traffic statistics in the form of a traffic matrix. While excellent work has been done in traffic matrix estimation from link loads, even the best results have errors on the order of 20%, which can lead to bad traffic engineering. Another approach is to directly measure the traffic to every destination at every router. While it is possible to globally aggregate the measurements into a traffic matrix that can be fed to a traffic engineering algorithm, it is more straightforward to use local measurements locally.

Also, usually it is smoother and quicker to respond to changes locally when they do occur. Thus, we are advocating a shift to relying directly on link loads and local traffic measurements instead of computing a traffic matrix for traffic engineering. A good way to avoid traffic matrices and a popular way to implement traffic engineering today is MPLS-TE.

The idea is to compute end-to-end tunnels for traffic demands with the available network bandwidth being assigned to new traffic demands using techniques like Constrained Shortest Path First. However, here, the performance gained over OSPF comes at the cost of establishing multiple end-to-end virtual circuits. Moreover, as the traffic changes, the end-to-end virtual circuits that were established for a particular traffic pattern become less useful, and performance degrades. Oblivious routing has also been proposed as a way around using traffic matrices for traffic engineering. The idea is to come up with a routing assignment that performs well irrespective of the traffic demand by comparing the "oblivious performance ratio" of the routing, i.e., the worst-case performance of the routing for a given network over all possible demands.

Breakthrough work in this area includes papers by Applegate and Cohen that developed a linear programming method to determine the best oblivious routing solution for the special case of minimizing maximum channel utilization and Kodialamet al that focused on maximizing throughput for the special case of two-phase routing. However, oblivious routing solutions do not adapt well to changes in the network topology and, by not tailoring the routing to the traffic demand, still incur significant performance losses.

On the other hand, we have the optimal distance-vector and source routing protocols discussed earlier. Despite their implementation problems, these iterative algorithms deliver on performance without a traffic matrix by relying on knowledge of the link flow rates. From an optimization standpoint, the algorithms by Gallager, Stern and Agnew are natural and mathematically elegant algorithms since the main ideas follow directly from the decomposition of the dual of the traffic engineering optimization problem. Decompositions like this, which have been very successful for problems of this type , can be used to yield updating rules for primal and dual variables (split ratios and node prices) that can be shown to converge to optimal solutions. Similar node-based ideas have also been applied to the cross-layer optimization of networks.

The source routing protocols involve algorithms that are based on iteratively calculating a shortest path at the source for each destination and transferring varying amounts of flow from the non-shortest paths to the shortest path till the optimal routing assignment is reached. It is worth noting here that although MPLS allows the source to control the path a packet takes through the network, it may not be practical for these protocols. Their iterative nature means that a large number of label switched paths might need to be computed and set up while they converge even though several of these paths will become redundant when the traffic changes.

There are different algorithms used to achieve efficient routing. Some of them are OSPF, Gallager's, PEFT, projected gradient, and HALO which is Hop by hop Adaptive Link state optimal routing. These algorithms have different characters like hop by hop, adaptive quality, link state, and optimal. The HALO has all of them, while the other algorithms miss one or more of the four qualities mentioned earlier. The comparison of different routing algorithms is as shown in the table below:

Algorithm	**Link-state**	**Hop by hop**	**Optimal**	**Adaptive**
OSPF	Yes	Yes	No	No
Gallager's	No	Yes	Yes	Yes
Projected Gradient	Yes	No	Yes	Yes
PEFT	Yes	Yes	No	No
HALO	Yes	Yes	Yes	Yes

Table 1: Comparison of routing algorithms

As seen in the above table the OSPF which is the most widely used routing technique adopts both link state and hop by hop. The Gallager's routing is adaptive, hop by hop and optimal but not link state. The projected gradient adopts all three except hop by hop. And PEFT adopts link state and hop by hop and is not adaptive and optimal. And the HALO is all four wiz link state, hop by hop, optimal and adaptive. So it can be shown that adopting all the above mentioned qualities in the algorithms can make the routing near optimal and hence improves the efficiency of the routing algorithm.

Advantages of the proposed system system are many, few are listed below**:**

– Dampen update frequency
– Target link state updates to multicast.
– Use link state area hierarchy for topology.
– Exchange route summaries at area borders.
– Use time-stamps update numbering and counters.
– Manage partitions using a area hierarchy.

References

[1] Nithin Michael, Student Member, IEEE,andAoTang, Senior Member, IEEE. 'HALO: Hop by hop adaptive link state optimal routing'. IEEE transactions on networking, vol. Pp, 10 September, 2014.
[2] N.Michael, A.Tang, and D.Xu, "Optimal link-state hop-by-hop routing," in Proc. IEEE ICNP, 2013, pp. 1–10.
[3] R. Gallager, "A minimum delay routing algorithm using distributed computation, "IEEE Trans. Commun., vol. COM-25, no. 1, pp. 73–85,Jan. 1977.
[4] L.Fratta, M.Gerla, and L.Kleinrock, "The flow deviation method: An approach to store-and-forward communication network design,"Networks, vol. 3, no. 2, pp. 97–133, 1973.
[5] B. Fortz and M. Thorup, "Increasing internet capacity using local nuity and invariance in hybrid automata," in Proc. 40th IEEE Decision search," Comput. Optim. Appl., vol. 29, no. 1, pp. 1348, Oct. 2004. Control, 2001, vol. 1, pp. 340345
[6] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, 5/E. New York, NY, USA: Addison-Wesley, 2010.
[7] D. Bertsekas and E. Gafni, "Projected newton methods and optimization of multicommodity flows," IEEE Trans. Autom. Control, vol. AC-28, no. 12, pp. 1090–1096, Dec. 1983.
[8] D. Xu, M. Chiang, and J. Rexford, "Link-state routing with hop-by-hop forwarding can achieve optimal traffic engineering," IEEE/ACM Trans. Netw., vol. 19, no. 6, pp. 1717–1730, Dec. 2011.
[9] A.Sridharan, R.Guerin, and C.Diot, "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks," IEEE/ACM Trans. Netw., vol. 13, no. 2, pp. 234–247, Apr. 2005.
[10] S. Srivastava, G. Agrawal, M. Pioro, and D. Medhi, "Determining link weight system under various objectives for OSPF networks using a lagrangian relaxation-based approach," IEEE Trans. Netw. Service Manag., vol. 2, no. 1, pp. 9–18, Nov. 2005.
[11] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," in Proc. ACM SIGMETRICS, New York, NY, USA, 2003, pp. 206–217.

[12] D. Awduche, “MPLS and traffic engineering in IP networks,”IEEE Commun. Mag., vol. 37, no. 12, pp. 42–47, Dec. 1999.
[13] A.Elwalid, C.Jin,S.Low, and I.Widjaja, “MATE: MPLSadaptive traffic engineering,” in Proc. 20th Annu. IEEE INFOCOM, 2001
[14] D. Applegate and E. Cohen, "Making routing robust to changing traffic demands: Algorithms and evaluation,” IEEE/ACM Trans. Netw., vol. 14, no. 6, pp. 1193–1206, Dec. 2006.
[15] M. Kodialam, T. V. Lakshman, J. Orlin, and S. Sengupta, “Oblivious routing of highly variable traffic in service overlays and IP backbones,” IEEE/ACM Trans. Netw., vol. 17, no. 2, pp. 459–472, Apr. 2009.
[16] T. Stern, “A class of decentralized routing algorithms using relaxation,” IEEE Trans. Commun., vol. COM-25, no. 10, pp. 1092–1102, Oct. 1977.
[17] C. E. Agnew, “On quadratic adaptive routing algorithms,” Commun. ACM, vol. 19, no. 1, pp. 18–22, Jan. 1976.
[18] D. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” IEEE J. Sel. Areas Commun., vol.24, no. 8, pp. 1439–1451, Aug. 2006.
[19] F. Paganini and E. Mallada, “A unified approach to congestion control and node-based multipath routing,” IEEE/ACM Trans. Netw., vol. 17, no. 5, pp. 1413–1426, Oct. 2009.
[20] Y. Xi and E. Yeh, “Node-based optimal power control, routing, and congestion control in wireless networks,” IEEE Trans. Inf. Theory, vol. 54, no. 9, pp. 4081–4106, Sep. 2008.