ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Minimizing the cost of carrying Traffic Using Link-state routing in Packet-switching Networks

Bhargavi Nagasuri
MTech Student
Department of CSE
Prakasam Engineering College, Kandukur

K.V. Srinivasa Rao (Ph.D)
Associate Professor
Department of CSE
Prakasam Engineering College, Kandukur

Abstract: ***Packet switching is a digital networking communications method that groups all transmitted data into suitably sized blocks, called packets, which are transmitted via a medium that may be shared by multiple simultaneous communication sessions. Packet switching increases network efficiency, robustness and enables technological convergence of many applications operating on the same network. Link-state routing protocols are one of the two main classes of routing protocols used in packet switching networks for computer communications, the other being distance-vector routing protocols. Link State routing protocols do not view networks in terms of adjacent routers and hop counts, but they build a comprehensive view of the overall network which fully describes the all possible routes along with their costs. Using the SPF (Shortest Path First) algorithm, the router creates a "topological database" which is a hierarchy reflecting the network routers it knows about. It then puts it's self on the top of this hierarchy, and has a complete picture from it's own perspective. Open Shortest Path First (OSPF) and Intermediate System-Intermediate System (IS-IS), split traffic evenly over shortest paths based on link weights. However, optimizing the link weights for OSPF/IS-IS to the offered traffic is a well-known NP-hard problem, and even the best setting of the weights can deviate significantly from an optimal distribution of the traffic. In this paper, we propose a new link-state routing protocol, PEFT, that splits traffic over multiple paths with an exponential penalty on longer paths.***

Keywords: ***Packet-Switching Networks, Routing, Shortest path, OSPF, HALO, Routers.***

Introduction:

Packet Switching is the routing and transferring of data by means of addressed packets so that a channel is occupied during the transmission of the packet only, and upon completion of the transmission the channel is made available for the transfer of other traffic. Packets are composed of a header and payload. Information in the header is used by networking hardware to direct the packet to its destination where the payload is extracted and used by application software. Packet switching features delivery of variable bit rate data streams, realized as sequences of packets, over a computer network which allocates transmission resources as needed using statistical multiplexing or dynamic bandwidth allocation techniques. When traversing network nodes, such as switches and routers, packets are buffered and queued, resulting in variable latency and throughput depending on the link capacity and the traffic load on the network.

Packet switching contrasts with another principal networking paradigm, circuit switching, a method which pre-allocates dedicated network bandwidth specifically for each communication session, each having a constant bit rate and latency between nodes. In cases of billable services, such as cellular communication services, circuit switching is characterized by a fee per unit of connection time, even when no data is transferred, while packet switching may be characterized by a fee per unit of information transmitted, such as characters, packets, or messages.

Packet mode communication may be implemented with or without intermediate forwarding nodes (packet switches or routers). Packets are normally forwarded

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

by intermediate network nodes asynchronously using first-in, first-out buffering, but may be forwarded according to some scheduling discipline for fair queuing, traffic shaping, or for differentiated or guaranteed quality of service, such as weighted fair queuing or leaky bucket.

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.

Path selection involves applying a routing metric to multiple routes to select (or predict) the best route. In computer networking, the metric is computed by a routing algorithm, and can cover information such as bandwidth, network delay, hop count, path cost, load, MTU (maximum transmission unit), reliability, and communication cost. The routing table stores only the best possible routes, while link-state or topological databases may store all other information as well.

Because a routing metric is specific to a given routing protocol, multi-protocol routers must use some external heuristic to select between routes learned from different routing protocols. Cisco routers, for example, attribute a value known as the administrative distance to each route, where smaller administrative distances indicate routes learned from a supposedly more reliable protocol.

Link-state routing protocol

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. Examples of link-state routing protocols include open shortest path first (OSPF) and intermediate system to intermediate system (IS-IS).

Calculating Shortest paths:

Each node independently runs an algorithm over the map to determine the shortest path from itself to every other node in the network; generally some variant of Dijkstra's algorithm is used. This is based around a link cost across each path which includes available bandwidth among other things.

A node maintains two data structures: a tree containing nodes which are "done", and a list of candidates. The algorithm starts with both structures empty; it then adds to the first one the node itself. The variant of a Greedy Algorithm then repetitively does the following:

- All neighbour nodes which are directly connected to the node are just added to the tree (excepting any nodes which are already in either the tree or the candidate list). The rest are added to the second (candidate) list.
- Each node in the candidate list is compared to each of the nodes already in the tree. The candidate node which is closest to any of the nodes already in the tree is itself moved into the tree and attached to the appropriate neighbor node. When a node is moved from the candidate list into the tree, it is removed from the candidate list and is not considered in subsequent iterations of the algorithm.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The above two steps are repeated as long as there are any nodes left in the candidate list. (When there are none, all the nodes in the network will have been added to the tree.) This procedure ends with the tree containing all the nodes in the network, with the node on which the algorithm is running as the root of the tree. The shortest path from that node to any other node is indicated by the list of nodes one traverses to get from the root of the tree, to the desired node in the tree.

Existing System

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.

Link State protocols in comparison to Distance Vector protocols have:

- Big 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

Proposed System

Our goal in this paper is to eliminate this tradeoff between optimality and ease of implementation in routing. The result is Hop-by-hop Adaptive Link-state Optimal (HALO), a routing solution that retains the simplicity of link-state, hop-by-hop protocols while iteratively converging to the optimal routing assignment. To the best of our knowledge, this is the first optimal link-state hop-by-hop routing solution. Not surprisingly, there are multiple challenges to overcome when designing such a solution. Before getting into them, we define the following important recurring terms for ease of exposition.

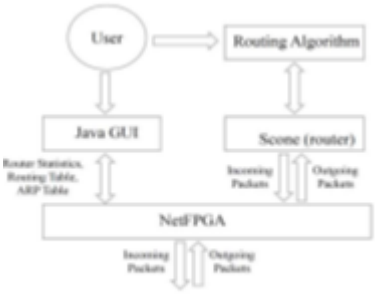
Hop-by-hop: Each router, based on the destination address, controls only the next hop that a packet takes.

Adaptive: The algorithm does not require the traffic demand matrix as an explicit input in order to compute link weights. Specifically, the algorithm seamlessly recognizes and adapts to changes in the network, both topology changes and traffic variations, as inferred from the network states like link flow rates.

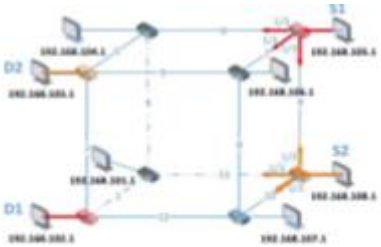
Link-state: Each router receives the state of all the network's links through periodically flooded link-state updates and makes routing decisions based on the link states.

Optimal: The routing algorithm minimizes some cost function (e.g., minimize total delay) determined by the network operator. The problem of guiding network traffic through routing to minimize a given global cost function is called traffic engineering (TE).

Architecture

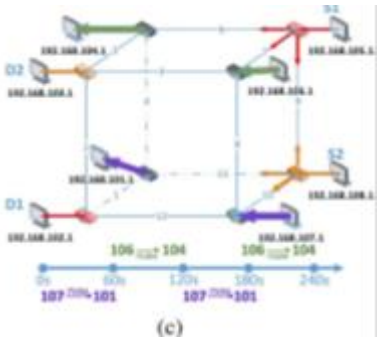


(a)



(b)

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal



Modules

Network Node Configuration

Iteratively adjust each router's split ratios and move traffic from one outgoing link to another. This only controls the next hop on a packet's path leading to hop-by-hop routing. If instead we controlled path rates, we would get source routing.

Path Design

Increase the split ratio to the link that is part of the shortest path at each iteration even though the average price via the next-hop router may not be the lowest. If instead we forwarded traffic via the next-hop router with the lowest average price, we get Gallager's approach, which is a distance vector solution.

Link Management

Adapt split ratios dynamically and incrementally by decreasing along links that belong to nonshortest paths while increasing along the link that is part of the shortest path at every router. If instead split ratios are set to be positive instantaneously only to the links leading to shortest paths, then we get OSPF with weights,

Dijkstra

For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city and all other cities. As a result, the shortest path algorithm is widely used in network routing protocols, most notably IS-IS and OSPF (Open Shortest Path First)

function Dijkstra(*Graph*, *source*):

dist[*source*] := 0 *// Distance from source to source*

for each vertex *v* in *Graph*: *// Initializations*

if *v* ≠ *source*

dist[*v*] := infinity *// Unknown distance function from source to v*

previous[*v*] := undefined *// Previous node in optimal path from source*

end if

add *v* to *Q* *// All nodes initially in Q (unvisited nodes)*

end for

while *Q* **is not** empty: *// The main loop*

u := vertex in *Q* with min dist[u] *// Source node in first case*

remove *u* from *Q*

for each neighbor *v* of *u*: *// where v has not yet been removed from Q.*

alt := dist[*u*] + length(*u*, *v*)

if *alt* < dist[*v*]: *// A shorter path to v has been found*

dist[*v*] := *alt*

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

```
            previous[v]  := u
        end if
    end for
end while
return dist[], previous[]
end function
```

Advantages of the Proposed System

1) Dampen update frequency

2) Target link-state updates to multicast

3) Use link-state area hierarchy for topology

4) Exchange route summaries at area borders

5) Use Time-stamps Update numbering & counters

6) Manage partitions using a area hierarchy

Conclusion:

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 qualities. 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.

References:

[1] Nithin Michael and Ao Tang, 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] Pradeep Raj Savarapu, Dynamic Traffic and Energy Aware Routing Algorithm for Multi-Sink Wireless Sensor Networks, IJMETMR, http://www.ijmetmr.com/olseptember2015/PradeepRaj Savarapu-61.pdf, Volume No: 2 (2015), Issue No: 9 (September)

[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] K.Nivedha, P.M.Sharmila & V.Shobana, ON-Demand Secure Routing Protocol For Authentication And Data Integrity in Manet, Yuva Engineers, http://www.yuvaengineers.com/on-demand-secure-routing-protocol-for-authentication-and-data-integrity-in-manet-k-nivedha-p-m-sharmila-v-shobana-mr-b-ram-kumar/

[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 nearoptimal 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

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

relaxation-based approach," IEEE Trans. Netw.Service Manag., vol. 2, no. 1, pp. 9–18, Nov. 2005.

[11] R. Gallager, "A minimum delay routing algorithm using distributed computation, "IEEE Trans. Commun., vol. COM-25, no. 1, pp. 73–85,Jan. 1977.

[12] 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