Workgroup: Network Working Group
Internet-Draft:
draft-white-lsr-distoptflood-02
Published: 25 April 2022
Intended Status: Informational
Expires: 27 October 2022

Authors: R. White S. Hegde T. Przygienda
Juniper Networks Juniper Networks Juniper Networks

# IS-IS Optimal Distributed Flooding for Dense Topologies

## Abstract

In dense topologies (such as data center fabrics based on the Clos and butterfly topologies, though not limited to these), IGP flooding mechanisms designed for sparse topologies can "overflood," or carry too many copies of topology and reachability information to fabric devices. This results in slower convergence times and higher resource utilization. The modifications to the flooding mechanism in the Intermediate System to Intermediate System (IS-IS) link state protocol described in this document reduce resource utilization significantly, while increasing convergence performance in dense topologies.

Note that a Clos fabric is used as the primary example of a dense flooding topology throughout this document. However, the flooding optimizations described in this document apply to any topology.

## Status of This Memo

## Copyright Notice

## Table of Contents

## 1. Introduction

### 1.1. Goals

The goal of this draft is to solve one specific set of problems involved in operating a link state protocol in a densely meshed topology. The problem with such topologies is the connectivity density, which causes too many copies of identical information to be flooded. Analysis and experiment show, for instance, that in a butterfly fabric of around 2500 intermediate systems, each intermediate system will receive more than 40 copies of any changed LSP fragment. This not only wastes bandwidth and processor time, this dramatically slows convergence speed.

This document describes a set of modifications to existing IS-IS flooding mechanisms which minimize the number of LSP fragments received by individual intermediate systems, in its extreme version to one copy per intermediate system. The mechanisms described in this document are similar to those implemented in OSPF to support mobile ad-hoc networks, as described in [RFC5449], [RFC5614], and

[RFC7182]. These mechanisms have been widely implemented and deployed.

## 1.2. Contributors

The following people have contributed to this draft: Abhishek Kumar, Nikos Triantafillis, Ivan Pepelnjak, Christian Franke, Hannes Gredler, Les Ginsberg, Naiming Shen, Uma Chunduri, Nick Russo, Shawn Zandi, and Rodny Molina.

## 1.3. Experience

Laboratory tests show modifications similar to these reduce flooding in a large scale emulated butterfly network topology; without these modifications, intermediate systems receive, on average, 40 copies of any changed LSP fragment. With the modifications described in this document intermediate systems recieve, on average, two copies of any changed LSP fragment. In many cases, each intermediate system receives only a single copy of each changed LSP. In terms of performance, the modifications described here cut convergence times in half. Processor load times were not checked, as this was an emulated environment.

A mechanism similar to the one described in this document has been implemented in the FR Routing open source routing stack as part of fabricd.

## 1.4. Sample Network

The following spine and leaf fabric will be used to describe these modifications.

```
+----+ +----+ +----+ +----+ +----+ +----+
| 1A | | 1B | | 1C | | 1D | | 1E | | 1F | (T0)
+----+ +----+ +----+ +----+ +----+ +----+

+----+ +----+ +----+ +----+ +----+ +----+
| 2A | | 2B | | 2C | | 2D | | 2E | | 2F | (T1)
+----+ +----+ +----+ +----+ +----+ +----+

+----+ +----+ +----+ +----+ +----+ +----+
| 3A | | 3B | | 3C | | 3D | | 3E | | 3F | (T2)
+----+ +----+ +----+ +----+ +----+ +----+

+----+ +----+ +----+ +----+ +----+ +----+
| 4A | | 4B | | 4C | | 4D | | 4E | | 4F | (T1)
+----+ +----+ +----+ +----+ +----+ +----+

+----+ +----+ +----+ +----+ +----+ +----+
| 5A | | 5B | | 5C | | 5D | | 5E | | 5F | (T0)
+----+ +----+ +----+ +----+ +----+ +----+
```

Figure 1

To reduce confusion (spine and leaf fabrics are difficult to draw in plain text art), this diagram does not contain the connections between devices. The reader should assume that each device in a given layer is connected to every device in the layer above it. For instance:

*5A is connected to 4A, 4B, 4C, 4D, 4E, and 4F

*5B is connected to 4A, 4B, 4C, 4D, 4E, and 4F

*4A is connected to 3A, 3B, 3C, 3D, 3E, 3F, 5A, 5B, 5C, 5D, 5E, and 5F

*4B is connected to 3A, 3B, 3C, 3D, 3E, 3F, 5A, 5B, 5C, 5D, 5E, and 5F

*etc.

The tiers or stages of the fabric are also marked for easier reference. T0 is assumed to be connected to application servers, or rather they are Top of Rack (ToR) intermediate systems. The remaining tiers, T1 and T2, are connected only to other devices in the fabric itself. A common alternate representation of this topology is drawn "folded" with T2, the "top of fabric," shown on top, while T1 is shown below, and T0 below T1.

## 2. Flooding Modifications

Flooding is perhaps the most challenging scaling issue for a link state protocol running on a dense, large scale topology. This section describes detailed modifications to the IS-IS flooding process to reduce flooding load in a densely meshed topology.

### 2.1. Optimizing Flooding

The simplest way to conceive of the solution presented here is in two stages:

*Stage 1: Forward Optimization

*  -Find the group of intermediate systems that will all flood to the same set of neighbors as the local IS

   -Decide (deterministically) which subset of the intermediate systems within this group should re-flood any received LSPs

*Stage 2: Reverse Optimization

*  -Find neighbors on the shortest path towards the origin of the change

   -Do not flood towards these neighbors

The first stage is best explained through an illustration. In the network above, if 5A transmits a modified Link State Protocol Data Unit (LSP) to 4A-4F, each of 4A-4F will, in turn, flood this modified LSP to 3A (for instance). 3A will receive 6 copies of the modified LSP, while only one copy is necessary for the intermediate systems shown to converge on a single view of the topology. If 4A-4F could determine they will all flood identical copies of the modified LSP to 3A, it is possible for all of them except one to decide not to flood the changed LSP to 3A.

The technique used in this draft to determine the flooding group is for each intermediate system to calculate a special Shortest-path Spanning Tree (SPT) from the point of view of the transmitting neighbor. By setting the metric of all links to 1 and truncating the SPT to two hops, the local IS can find the group of neighbors it will flood any changed LSP towards and the set of intermediate systems (not necessarily neighbors) which will also flood to this same set of neighbors. If every intermediate system in the flooding set performs this same calculation, they will all obtain the same flooding group.

Once this flooding group is determined, the members of the flooding group will each (independently) choose which of the members should re-flood the received information. Each member of the flooding group calculates this independently of all the other members, but a common hash MUST be used across a set of shared variables so each member of the group comes to the same conclusion. The group member which is selected to flood the changed LSP does so normally; the remaining group members do not flood the LSP.

Note there is no signaling between the intermediate systems running this flooding reduction mechanism. Each IS calculates the special, truncated SPT separately, and determines which IS should flood any changed LSPs independently based on a common hash function. Because these calculations are performed using a shared view of the network, however (based on the common link state database) and a shared hash function, each member of the flooding group will make the same decision.

The second stage is simpler, consisting of a single rule: do not flood modified LSPs along the shortest path towards the origin of the modified LSP. This rule relies on the observation that any IS between the origin of the modified LSP and the local IS should receive the modified LSP from some other IS closer to the source of the modified LSP.

## 2.2. Optimization Process

Each intermediate system will determine whether it should re-flood LSPs as described below. When a modified LSP arrives from a Transmitting Neighbor (TN), the result of the following algorithm obtains the necessary decision:

Step 1: Build the Two-Hop List (THL) and Remote Neighbor's List (RNL) by:

- Set all link metrics to 1
- Calculate an SPT truncated to 2 hops from the perspective of TN
- For each IS that is two hops (has a metric of two in the truncated SPT) from TN:
  - -If the IS is on the shortest path towards the originator of the modified LSP, skip
  - -If the IS is not on the shortest path towards the originator of the modified LSP, add it to THL
- Add each IS that is one hop away from TN to the RNL

Step 2: Sort RNL by system IDs, from the least to the greatest.

Step 3: Calculate a number, N, by adding each byte in LSP-ID (without the fragment ID) and fragment ID MOD 2 (allowing for some balancing of LSPs coming from same system ID without introducing excessive amount of state in an implementation) and then taking MOD on the number of neighbors. N MUST be less than the number of members of RNL.

Step 4: Starting with the Nth member of RNL:

- *If THL is empty, exit

- *If this member of RNL is the local calculating IS, this IS MUST reflood the modified LSP; exit

- *Remove all members of THL connected to (adjacent to) this member of RNL

- *Move to the next member of RNL, wrapping to the beginning of RNL if necessary

Note: This description is geared to clarity rather than optimal performance.

### 2.3. Flooding Failures

It is possible in some failure modes for flooding to be incomplete because of the flooding optimizations outlined. Specifically, if a reflooder fails, or is somehow disconnected from all the links across which it should be reflooding, it is possible an LSP is only partially flooded through the fabric. To prevent such partition failures, an intermediate system which does not reflood an LSP (or fragment) should:

- *Set a short timer; the default should be one second

- *When the timer expires, send Partial Sequence Number Packet (PSNP) of all LSPs that have not been reflooded during the timer runtime to all neighbors unless an up-to-date PSNP or CSNP has been already received from the neighbor

- *Process any Partial Sequence Number Packets (PSNPs) received that indicate that neighbors still have older versions of the LSP per normal protocol procedures to resynchronize

- *If resynchronization above a configurable threshold is required, an implementation SHOULD notify the network operator

### 2.4. Flooding Example

Assume, in the network above, 5A floods some modified LSP towards 4A-4F. To determine whether 4A should flood this LSP to 3A-3F:

- 5A is TN; 4A calculates a truncated SPT from 5A's perspective with all link metrics set to 1
- 4A builds THL, which contains 3A, 3B, 3C, 3D, 3E, 3F, 5B, 5C, 5D, 5E and 5F
- 4A builds RNL, which contains 4A,4B,4C,4D,4E and 4F, sorting it by the system ID
- 4A computes hash on the received LSP-ID to get N; assume N is 1 in this case
- Since 4A is the Nth member of R-NL and there are members in N-NL, 4A must reflood; the loop exits

### 2.5. A Note on Performance

The calculations described here are complex, which might lead the reader to conclude that the cost of calculation is so much higher than the cost of flooding that this optimization is counter-productive. The description provided here is designed for clarity rather than optimal calculation, however. Many of the calculations can be performed in advance and stored, rather than being performed for each LSP and each neighbor. Optimized versions of the process described here have been implemented, and do result in strong convergence speed gains.

## 3. Security Considerations

This document outlines modifications to the IS-IS protocol for operation on high density network topologies. Implementations SHOULD implement IS-IS cryptographic authentication, as described in [RFC5304], and should enable other security measures in accordance with best common practices for the IS-IS protocol.

## 4. References

### 4.1. Normative References

**[I-D.ietf-lsr-dynamic-flooding]**
Li, T., Przygienda, T., Psenak, P., Ginsberg, L., Chen, H., Cooper, D., Jalil, L., Dontula, S., and G. S. Mishra, "Dynamic Flooding on Dense Graphs", Work in Progress, Internet-Draft, draft-ietf-lsr-dynamic-flooding-10, 7

December 2021, <https://www.ietf.org/archive/id/draft-ietf-lsr-dynamic-flooding-10.txt>.

**[ISO10589]** ISO, "Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Second Edition, November 2002.

**[RFC2119]** Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

**[RFC2629]** Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, DOI 10.17487/RFC2629, June 1999, <https://www.rfc-editor.org/info/rfc2629>.

**[RFC5120]** Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <https://www.rfc-editor.org/info/rfc5120>.

**[RFC5301]** McPherson, D. and N. Shen, "Dynamic Hostname Exchange Mechanism for IS-IS", RFC 5301, DOI 10.17487/RFC5301, October 2008, <https://www.rfc-editor.org/info/rfc5301>.

**[RFC5303]** Katz, D., Saluja, R., and D. Eastlake 3rd, "Three-Way Handshake for IS-IS Point-to-Point Adjacencies", RFC 5303, DOI 10.17487/RFC5303, October 2008, <https://www.rfc-editor.org/info/rfc5303>.

**[RFC5305]** Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

**[RFC5308]** Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, DOI 10.17487/RFC5308, October 2008, <https://www.rfc-editor.org/info/rfc5308>.

**[RFC5309]** Shen, N., Ed. and A. Zinin, Ed., "Point-to-Point Operation over LAN in Link State Routing Protocols", RFC 5309, DOI 10.17487/RFC5309, October 2008, <https://www.rfc-editor.org/info/rfc5309>.

**[RFC5311]** McPherson, D., Ed., Ginsberg, L., Previdi, S., and M. Shand, "Simplified Extension of Link State PDU (LSP) Space for IS-IS", RFC 5311, DOI 10.17487/RFC5311, February 2009, <https://www.rfc-editor.org/info/rfc5311>.

**[RFC5316]** Chen, M., Zhang, R., and X. Duan, "ISIS Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS Traffic Engineering", RFC 5316, DOI 10.17487/RFC5316, December 2008, <https://www.rfc-editor.org/info/rfc5316>.

**[RFC7356]** Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, DOI 10.17487/RFC7356, September 2014, <https://www.rfc-editor.org/info/rfc7356>.

**[RFC7981]** Ginsberg, L., Previdi, S., and M. Chen, "IS-IS Extensions for Advertising Router Information", RFC 7981, DOI 10.17487/RFC7981, October 2016, <https://www.rfc-editor.org/info/rfc7981>.

**[RFC8174]** Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

### 4.2. Informative References

**[I-D.ietf-isis-segment-routing-extensions]** Previdi, S., Ginsberg, L., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", Work in Progress, Internet-Draft, draft-ietf-isis-segment-routing-extensions-25, 19 May 2019, <https://www.ietf.org/archive/id/draft-ietf-isis-segment-routing-extensions-25.txt>.

**[RFC3277]** McPherson, D., "Intermediate System to Intermediate System (IS-IS) Transient Blackhole Avoidance", RFC 3277, DOI 10.17487/RFC3277, April 2002, <https://www.rfc-editor.org/info/rfc3277>.

**[RFC3719]** Parker, J., Ed., "Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)", RFC 3719, DOI 10.17487/RFC3719, February 2004, <https://www.rfc-editor.org/info/rfc3719>.

**[RFC4271]** Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI

10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

**[RFC5304]** Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

**[RFC5440]** Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

**[RFC5449]** Baccelli, E., Jacquet, P., Nguyen, D., and T. Clausen, "OSPF Multipoint Relay (MPR) Extension for Ad Hoc Networks", RFC 5449, DOI 10.17487/RFC5449, February 2009, <https://www.rfc-editor.org/info/rfc5449>.

**[RFC5614]** Ogier, R. and P. Spagnolo, "Mobile Ad Hoc Network (MANET) Extension of OSPF Using Connected Dominating Set (CDS) Flooding", RFC 5614, DOI 10.17487/RFC5614, August 2009, <https://www.rfc-editor.org/info/rfc5614>.

**[RFC5820]** Roy, A., Ed. and M. Chandra, Ed., "Extensions to OSPF to Support Mobile Ad Hoc Networking", RFC 5820, DOI 10.17487/RFC5820, March 2010, <https://www.rfc-editor.org/info/rfc5820>.

**[RFC5837]** Atlas, A., Ed., Bonica, R., Ed., Pignataro, C., Ed., Shen, N., and JR. Rivers, "Extending ICMP for Interface and Next-Hop Identification", RFC 5837, DOI 10.17487/RFC5837, April 2010, <https://www.rfc-editor.org/info/rfc5837>.

**[RFC6232]** Wei, F., Qin, Y., Li, Z., Li, T., and J. Dong, "Purge Originator Identification TLV for IS-IS", RFC 6232, DOI 10.17487/RFC6232, May 2011, <https://www.rfc-editor.org/info/rfc6232>.

**[RFC7182]** Herberg, U., Clausen, T., and C. Dearlove, "Integrity Check Value and Timestamp TLV Definitions for Mobile Ad Hoc Networks (MANETs)", RFC 7182, DOI 10.17487/RFC7182, April 2014, <https://www.rfc-editor.org/info/rfc7182>.

**[RFC7921]** Atlas, A., Halpern, J., Hares, S., Ward, D., and T. Nadeau, "An Architecture for the Interface to the Routing System", RFC 7921, DOI 10.17487/RFC7921, June 2016, <https://www.rfc-editor.org/info/rfc7921>.

## Authors' Addresses

Russ White
Juniper Networks

Email: russ@riw.us

Shraddha Hegde
Juniper Networks

Email: shraddha@juniper.net

Tony Przygienda
Juniper Networks

Email: prz@juniper.net