Framework for Loop-Free Convergence
Using the Ordered Forwarding Information Base (oFIB) Approach

Abstract

   This document describes an illustrative framework of a mechanism for
   use in conjunction with link-state routing protocols that prevents
   the transient loops that would otherwise occur during topology
   changes. It does this by correctly sequencing the forwarding
   information base (FIB) updates on the routers.

   This mechanism can be used in the case of non-urgent (management
   action) link or node shutdowns and restarts or link metric changes.
   It can also be used in conjunction with a fast reroute mechanism that
   converts a sudden link or node failure into a non-urgent topology
   change. This is possible where a complete repair path is provided
   for all affected destinations.

   After a non-urgent topology change, each router computes a rank that
   defines the time at which it can safely update its FIB. A method for
   accelerating this loop-free convergence process by the use of
   completion messages is also described.

   The technology described in this document has been subject to
   extensive simulation using pathological convergence behavior and real
   network topologies and costs. However, the mechanisms described in
   this document are purely illustrative of the general approach and do
   not constitute a protocol specification. This document represents a
   snapshot of the work of the Routing Area Working Group at the time of
   publication and is published as a document of record. Further work
   is needed before implementation or deployment.

Status of This Memo

   This document is not an Internet Standards Track specification; it is
   published for informational purposes.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Not all documents
   approved by the IESG are a candidate for any level of Internet
   Standard; see Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6976.

Copyright Notice

Table of Contents

1. Introduction

1.1. The Purpose of This Document

   This document describes an illustrative framework of a mechanism for
   use in conjunction with link-state routing protocols that prevents
   the transient loops that would otherwise occur during topology
   changes. It does this by correctly sequencing the forwarding
   information base (FIB) updates on the routers.

   At the time of publication there is no demand to deploy this
   technology; however, in view of the subtleties involved in the design
   of extensions for loop-free convergence routing protocols, the
   Routing Area Working Group considered it desirable to publish this
   document to place on record the design consideration of the ordered
   FIB (oFIB) approach.

   The mechanisms presented in this document are purely illustrative of
   the general approach and do not constitute a protocol specification.
   This document represents a snapshot of the work of the working group
   at the time of publication and is published as a document of record.
   Additional work is needed to specify the necessary routing protocol
   extensions necessary to support this IP fast reroute (FRR) method
   before implementation or deployment.

1.2. Overview

   With link-state protocols, such as IS-IS [ISO10589] and OSPF
   [RFC2328], each time the network topology changes, some routers need
   to modify their forwarding information bases (FIBs) to take into
   account the new topology. Each topology change causes a convergence
   phase. During this phase, routers may transiently have inconsistent
   FIBs, which may lead to packet loops and losses, even if the
   reachability of the destinations is not compromised after the
   topology change. Packet losses and transient loops can also occur in
   the case of a link down event implied by a maintenance operation,
   even if this operation is predictable and not urgent. When the link-
   state change is a metric update and when a new link is brought up in
   the network, there is no direct loss of connectivity, but transient
   packet loops and loss can still occur.

   In this document, a distinction is made between urgent and non-urgent
   network events. Urgent events are those that arise from
   unpredictable network outages (such as node or link failures) that
   are traditionally resolved through the convergence of routing
   protocols or by protection mechanisms reliant on fault detection and
   reporting (such as through Operations, Administration, and
   Maintenance). Non-urgent events are those that arise from

predictable events such as the controlled shutdown of network
resources by a management system, or the modification of network
parameters (such as routing metrics). Typically, non-urgent events
can be planned around, while urgent events must be handled by dynamic
systems. All network events, both urgent and non-urgent, may lead to
transient packet loops and loss.

For example, in Figure 1, if the link between X and Y is shut down by
an operator, packets destined to X can loop between R and Y when Y
has updated its FIB while R has not yet updated its FIB, and packets
destined to Y can loop between X and S if X updates its FIB before S.
According to the current behavior of IS-IS and OSPF, this scenario
will happen most of the time because X and Y are the first routers to
be aware of the failure, so that they will update their FIBs first.

```
                                   1
                 X-------------/-------------Y
                 |                           |
                 |                           |
                 |                           |
                 |                           |
              1  |                           |  1
                 |                           |
                 |                           |
                 |                           |
                 |                           |
                 S---------------------------R
                              2
```
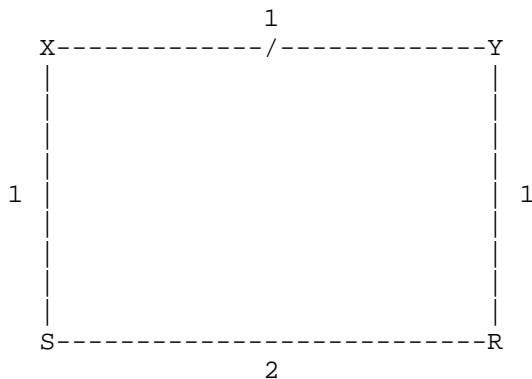
Figure 1: A Simple Topology

It should be noted that the loops can occur remotely from the
failure, not just adjacent to it.

[RFC5715] provides an introduction to a number of loop-free
convergence methods, and readers unfamiliar with this technology are
recommended to read it before studying this document in detail. Note
that in common with other loop-free convergence methods, oFIB is only
capable of providing loop-free convergence in the presence of a
single failure.

The goal of this document is to describe a mechanism that sequences
the router FIB updates to maintain consistency throughout the
network. By correctly setting the FIB change order, no looping or
packet loss can occur. This mechanism may be applied to the case of
managed link-state changes, i.e., link metric change, manual link
down/up, manual router down/up, and managed state changes of a set of
links attached to one router. It may also be applied to the case

where one or more network elements are protected by a fast reroute
mechanism (FRR) [RFC5714] [RFC4090]. The mechanisms that are used in
the failure case are exactly the same as those used for managed
changes. For simplicity, this document makes no further distinction
between managed and unplanned changes.

It is assumed in the description that follows that all routers in the
routing domain are oFIB capable. This can be verified in an
operational network by having the routers report oFIB capability
using the IGP. Where non-oFIB-capable routers exist in the network,
normal convergence would be used by all routers. The operation of
mixed-mode networks is for further study.

The technology described in this document has been subject to
extensive simulation using pathological convergence behavior and real
network topologies and costs. A variant of the technology described
here has been experimentally deployed in a production network.

2. The Required FIB Update Order

   This section provides an overview of the required ordering of the FIB
   updates. A more detailed analysis of the rerouting dynamics and
   correctness proofs of the mechanism can be found in [refs.PFOB07].

2.1. Single Link Events

   For simplicity, the correct ordering for single link changes are
   described first. The document then builds on this to demonstrate
   that the same principles can be applied to more complex scenarios
   such as line-card or node changes.

2.1.1. Link Down / Metric Increase

   First, consider the non-urgent failure of a link (i.e., where an
   operator or a network management system (NMS) shuts down a link,
   thereby removing it from the currently active topology) or the
   increase of a link metric by the operator or NMS. In this case, a
   router R must not update its FIB until all other routers that send
   traffic via R and the affected link have first updated their FIBs.

   The following argument shows that this rule ensures the correct order
   of FIB changes when the link X->Y is shut down or its metric is
   increased.

   An "outdated" FIB entry for a destination is defined as being a FIB
   entry that still reflects the shortest path(s) in use before the
   topology change. Once a packet reaches a router R that has an
   outdated FIB entry for the packet destination, then, provided the

oFIB ordering is respected, the packet will continue to X only
traversing routers that also have an outdated FIB entry for the
destination. The packet thus reaches X without looping and will be
forwarded to Y via X->Y (or in the case of FRR, the X->Y repair path)
and will reach its destination.

Since it can be assumed that the original topology was loop-free, Y
will never use the link Y->X to reach the destination, and hence the
path(s) between Y and the destination are guaranteed to be unaffected
by the topology change. It therefore follows that the packet
arriving at Y will reach its destination without looping.

Since it can also be assumed that the new topology is loop-free, by
definition a packet cannot loop while being forwarded exclusively by
routers with an updated FIB entry.

In other words, when the oFIB ordering is respected, if a packet
reaches an outdated router, it can never subsequently reach an
updated router, and it cannot loop because from this point on it will
only be forwarded on the consistent path that was used before the
event. If it does not reach an outdated router, it will only be
forwarded on the loop-free path that will be used after the
convergence.

According to the proposed ordering, X will be the last router to
update its FIB. Once it has updated its FIB, the link X->Y can
actually be shut down (or the repair removed).

If the link X-Y is bidirectional, a similar process must be run to
order the FIB update for destinations using the link in the direction
Y->X. As has already been shown, no packet ever traverses the X-Y
link in both directions, and hence the operation of the two ordering
processes is orthogonal.

2.1.2. Link Up / Metric Decrease

In the case of link up events or metric decreases, a router R must
update its FIB before all other routers that will use R to reach the
affected link.

The following argument shows that this rule ensures the correct order
of FIB changes when the link X->Y is brought into service or its
metric is decreased.

Firstly, when a packet reaches a router R that has already updated
its FIB, all the routers on the path from R to X will also have
updated their FIB, so that the packet will reach X and be forwarded
along X->Y, ultimately reaching its destination.

Secondly, a packet cannot loop between routers that have not yet
updated their FIB. This proves that no packet can loop.

2.2. Multi-Link Events

The following sections describe the required ordering for single
events that may manifest as multiple link events. For example, the
failure of a router may be notified to the rest of the network as the
individual failure of all its attached links. The means of
identifying the event type from the collection of received link
events is described in Section 3.1.

2.2.1. Router Down Events

In the case of the non-urgent shutdown of a router, a router R must
not update its FIB until all other routers that send traffic via R
and the affected router have first updated their FIBs.

Using a proof similar to that for link failure, it can be shown that
no loops will occur if this ordering is respected [refs.PFOB07].

2.2.2. Router Up Events

In the case of a router being brought into service, a router R must
update its FIB BEFORE all other routers that WILL use R to reach the
affected router.

A proof similar to that for link up shows that no loops will occur if
this ordering is respected [refs.PFOB07].

2.2.3. Line-Card Failure/Restoration Events

The failure of a line card involves the failure of a set of links,
all of which have a single node in common, i.e., the parent router.
The ordering to be applied is the same as if it were the failure of
the parent router.

In a similar way, the restoration of an entire line card to service
as a single event can be treated as if the parent router were
returning to service.

3. Applying Ordered FIB Updates

3.1. Deducing the Topology Change

   As has been described, a single event such as the failure or
   restoration of a single link, single router, or line card may be
   notified to the rest of the network as a set of individual link
   change events. It is necessary to deduce from this collection of
   link-state notifications the type of event that has occurred in the
   network and hence the required ordering.

   When a link change event is received that impacts the receiving
   router's FIB, the routers at the near and far end of the link are
   noted.

   If all events received within some hold-down period (the time that a
   router waits to acquire a set of Link State Packets (LSPs) that
   should be processed together) have a single router in common, then it
   is assumed that the change reflects an event (line-card or router
   change) concerning that router.

   In the case of a link change event, the router at the far end of the
   link is deemed to be the common router.

   All ordering computations are based on treating the common router as
   the root for both link and node events.

3.2. Deciding If Ordered FIB Updates Apply

   There are some events (for example, a subsequent failure with
   conflicting repair requirements occurring before the ordered FIB
   process has completed) that cannot be correctly processed by this
   mechanism. In these cases, it is necessary to ensure that
   convergence falls back to the conventional mode of operation (see
   Section 6).

   In all cases, it is necessary to wait some hold-down period after
   receiving the first notification to ensure that all routers have
   received the complete set of link-state notifications associated with
   the single event.

   At any time, if a link change notification is received that would
   have no effect on the receiving router's FIB, then it may be ignored.

   If no other event is received during the hold-down time, the event is
   treated as a link event. Note that the IGP reverse connectivity
   check means that only the first failure event or second up event has
   an effect on the FIB.

   If an event that is received within the hold-down period does NOT
   reference the common router (R), then, in this version of the
   specification, normal convergence is invoked immediately (see
   Section 6).

   Network reconvergence using the ordered FIB approach takes longer
   than the normal reconvergence process. Where the failure is
   protected by an FRR mechanism, this additional delay in convergence
   causes no packet loss. When the sudden failure of a link or a set of
   links that are not protected using an FRR mechanism occurs, the
   failure must be processed using the conventional (faster) mode of
   operation to minimize packet loss during reconvergence.

   In summary, an ordered FIB process is applicable if the set of link
   state notifications received between the first event and the hold-
   down period reference a common router R, and one of the following
   assertions is verified:

   o The set of notifications refers to link down events concerning
      protected links and metric increase events.

   o The set of notifications refers to link up events and metric
      decrease events.

4. Computation of the Ordering

   This section describes how the required ordering is computed.

   This computation required the introduction of the concept of a
   reverse Shortest Path Tree (rSPT). The rSPT uses the cost towards
   the root (rather than from it) and yields the best paths towards the
   root from other nodes in the network [IPFRR-TUNNELS].

4.1. Link Down, Router Down, or Metric Increase

   To respect the proposed ordering, routers compute a rank that will be
   used to determine the time at which they are permitted to perform
   their FIB update. In the case of a failure event rooted at router Y
   or an increase of the metric of link X->Y, router R computes the rSPT
   in the topology before the failure (rSPT_old) rooted at Y. This rSPT
   gives the shortest paths to reach Y before the failure. The branch
   of the rSPT that is below R corresponds to the set of shortest paths
   to R that are used by the routers that reach Y via R.

   The rank of router R is defined as the depth (in number of hops) of
   this branch. In the case of Equal Cost Multipath (ECMP), the maximum
   depth of the ECMP path set is used.

Router R is required to update its FIB at time

    T0 + H + (rank * MAX_FIB)

where T0 is the arrival time of the Link State Packet containing the
topology change, H is the hold-down time, and MAX_FIB is a network-
wide constant that reflects the maximum time required to update a FIB
irrespective of the change required. The value of MAX_FIB is network
specific, and its determination is out of the scope of this document.
This value must be agreed to by all the routers in the network. This
agreement can be performed by using a capability TLV as defined in
Appendix B.

All the routers that use R to reach Y will compute a lower rank than
R, and hence the correct order will be respected. It should be noted
that only the routers that used Y before the event need to compute
their rank.

4.2. Link Up, Router Up, or Metric Decrease

In the case of a link or router up event rooted at Y or a link metric
decrease affecting link Y->W, a router R must have a rank that is
higher than the rank of the routers that it will use to reach Y,
according to the rule described in Section 2. Thus, the rank of R is
the number of hops between R and Y in its renewed Shortest Path Tree.
When R has multiple equal-cost paths to Y, the rank is the length in
hops of the longest ECMP path to Y.

Router R is required to update its FIB at time

    T0 + H + (rank * MAX_FIB)

It should be noted that only the routers that use Y after the event
have to compute a rank, i.e., only the routers that have Y in their
SPT after the link-state change.

5. Acceleration of Ordered Convergence

The mechanism described above is conservative and hence may be
relatively slow. The purpose of this section is to describe a method
of accelerating the controlled convergence in such a way that ordered
loop-free convergence is still guaranteed.

In many cases, a router will complete its required FIB changes in a
time much shorter than MAX_FIB, and in many other cases, a router
will not have to perform any FIB change at all.

This section describes the use of completion messages to speed up the
convergence by providing a means for a router to inform those routers
waiting for it that it has completed any required FIB changes. When
a router has been advised of completion by all the routers for which
it is waiting, it can safely update its own FIB without further
delay. In most cases, this can result in a sub-second reconvergence
time, which is comparable with a normal convergence time.

Routers maintain a waiting list of the neighbors from which a
completion message must be received. Upon reception of a completion
message from a neighbor, a router removes this neighbor from its
waiting list. Once its waiting list becomes empty, the router is
allowed to update its FIB immediately even if its ranking timer has
not yet expired. Once this is done, the router sends a completion
message to the neighbors that are waiting for it to complete. Those
routers are listed in a list called the Notification List.
Completion messages contain an identification of the event to which
they refer.

Note that, since this is only an optimization, any loss of completion
messages will result in the routers waiting their defined ranking
time, and hence the loop-free properties will be preserved.

5.1. Construction of the Waiting List and Notification List

5.1.1. Down Events

Consider a link or node down event rooted at router Y or the cost
increase of the link X->Y. A router R will compute rSPT_old(Y) to
determine its rank. When doing this, R also computes the set of
neighbors that R uses to reach the failing node or link, and the set
of neighbors that are using R to reach the failing node or link. The
notification list of R is equal to the former set, and the waiting
list of R is equal to the latter.

Note that R could include all its neighbors in the notification list
except those in the waiting list; this would have no impact on the
correctness of the protocol but would be unnecessarily inefficient.

5.1.2. Up Events

Consider a link or node up event rooted at router Y or the cost
decrease of the link Y->X. A router R will compute its new SPT
(SPT_new(R)). The waiting list is the set of next-hop routers that R
uses to reach Y in SPT_new(R).

In a simple implementation, the notification list of R is all the
neighbors of R excluding those in the waiting list. This may be
further optimized by computing rSPT_new(Y) to determine those routers
that are waiting for R to complete.

5.2. Format of Completion Messages

The format of completion messages and means of their delivery is
routing protocol dependent and is outside the scope of this document.

The following information is required:

o Identity of the sender.

o List of routing notifications being considered in the associated
   FIB change. Each notification is defined as:

      Node ID of the near end of the link

      Node ID of the far end of the link

      Inclusion or removal of link

      Old metric

      New metric

6. Fallback to Conventional Convergence

In circumstances where a router detects that it is dealing with
incomplete or inconsistent link-state information, or when a further
topology event is received before completion of the current ordered
FIB update process, it may be expedient to abandon the controlled
convergence process. A number of possible fallback mechanisms are
described in Appendix A. This mechanism is referred to as
"Abandoning All Hope" (AAH). The state machine defined in the body
of this document does not make any assumption about which fallback
mechanism will be used.

7. oFIB State Machine

An implementation must be capable of interworking with the model of
an oFIB state machine described in this section.

An oFIB-capable router maintains an oFIB state value, which is one
of: OFIB_STABLE, OFIB_HOLDING_DOWN, OFIB_HOLDING_UP, OFIB_ABANDONED,
or OFIB_ONGOING.

An oFIB-capable router maintains a timer, Hold_down_timer. An oFIB-
capable router is configured with a value referred to as
HOLD_DOWN_DURATION. This configuration can be performed manually or
using Appendix B.

An oFIB-capable router maintains a timer, rank_timer.

7.1. OFIB_STABLE

OFIB_STABLE is the state of a router that is not currently involved
in any convergence process. This router is ready to process an event
by applying oFIB.

EVENT: Reception of a Link State Packet that describes an event of
the type link X--Y down or metric increase and is to be processed
using oFIB.

ACTION:

   Set state to OFIB_HOLDING_DOWN.

   Start Hold_down_timer.

   ofib_current_common_set = {X,Y}.

   Compute rank with respect to the event, as defined in Section 4.

   Store the waiting list and notification list for X--Y obtained
   from the rank computation.

EVENT: Reception of a Link State Packet that describes an event of
the type link X--Y up or metric decrease and is to be processed using
oFIB.

ACTION:

   Set state to OFIB_HOLDING_UP.

   Start Hold_down_timer.

   ofib_current_common_set = {X,Y}.

   Compute rank with respect to the event, as defined in Section 4.

   Store the waiting list and notification list for X--Y obtained
   from the rank computation.

7.2. OFIB_HOLDING_DOWN

   OFIB_HOLDING_DOWN is the state of a router that is collecting a set
   of link down or metric increase Link State Packets to be processed
   together using controlled convergence.

   EVENT: Reception of a Link State Packet that describes an event of
   the type link up or metric decrease and can be processed using oFIB.

   ACTION:

      Set state to OFIB_ABANDONED.

      Reset Hold_down_timer.

      Trigger AAH mechanism.

   EVENT: Reception of a Link State Packet that describes an event of
   the type link A--B down or metric increase and can be processed using
   oFIB.

   ACTION:

      ofib_current_common_set =
      intersection(ofib_current_common_set,{A,B}).

      If ofib_current_common_set is empty, then there is no longer a
      node in common in all the pending link-state changes.

         Set state to OFIB_ABANDONED.

         Reset Hold_down_timer.

         Trigger AAH mechanism.

      If ofib_current_common set is not empty, update the waiting list
      and notification list as defined in Section 4. Note that in the
      case of a single link event, the Link State Packet received when
      the router is in this state describes the state change of the
      other direction of the link; hence, no changes will be made to the
      waiting and notification lists.

EVENT: Hold_down_timer expires.

ACTION:

   Set state to OFIB_ONGOING.

   Start rank_timer with computed rank.

EVENT: Reception of a completion message.

ACTION: Remove the sender from the waiting list associated with the
event identified in the completion message.

7.3. OFIB_HOLDING_UP

   OFIB_HOLDING_UP is the state of a router that is collecting a set of
   link up or metric decrease Link State Packets to be processed
   together using controlled convergence.

   EVENT: Reception of a Link State Packet that describes an event of
   the type link down or metric increase and is to be processed using
   oFIB.

   ACTION:

      Set state to OFIB_ABANDONED.

      Reset Hold_down_timer.

      Trigger AAH mechanism.

   EVENT: Reception of a Link State Packet that describes an event of
   the type link A--B up or metric decrease and is to be processed using
   oFIB.

   ACTION:

      ofib_current_common_set =
      intersection(ofib_current_common_set,{A,B}).

      If ofib_current_common_set is empty, then there is no longer a
      common node in the set of pending link-state changes.

         Set state to OFIB_ABANDONED.

         Reset Hold_down_timer.

         Trigger AAH mechanism.

If ofib_current_common set is not empty, update the waiting list and notification list as defined in Section 4. Note that in the case of a single link event, the Link State Packet received when the router is in this state describes the state change of the other direction of the link; hence, no changes will be made to the waiting and notification lists.

EVENT: Reception of a completion message.

ACTION: Remove the sender from the waiting list associated with the event identified in the completion message.

EVENT: Hold_down_timer expires.

ACTION:

Set state to OFIB_ONGOING.

Start rank_timer with computed rank.

7.4. OFIB_ONGOING

OFIB_ONGOING is the state of a router that is applying the ordering mechanism with respect to the set of Link State Packets collected when in OFIB_HOLDING_DOWN or OFIB_HOLDING_UP state.

EVENT: rank_timer expires or waiting list becomes empty.

ACTION:

Perform FIB updates according to the change.

Send completion message to each member of the notification list.

Set state to OFIB_STABLE.

EVENT: Reception of a completion message.

ACTION: Remove the sender from the waiting list.

EVENT: Reception of a Link State Packet describing a link-state change event.

   ACTION:

      Set state to OFIB_ABANDONED.

      Trigger AAH.

      Start Hold_down_timer.

7.5. OFIB_ABANDONED

   OFIB_ABANDONED is the state of a router that has fallen back to fast
   convergence due to the reception of Link State Packets that cannot be
   dealt with together using oFIB.

   EVENT: Reception of a Link State Packet describing a link-state
   change event.

   ACTION: Trigger AAH, reset AAH_Hold_down_timer.

   EVENT: AAH_Hold_down_timer expires.

   ACTION: Set state to OFIB_STABLE.

8. Management Considerations

   A system for recording the dynamics of the convergence process needs
   to be deployed in order to make a post hoc diagnosis of the
   reconvergence. The sensitivity of applications to any packet
   reordering introduced by the delayed convergence process will need to
   be studied. However, these needs apply to any loop-free convergence
   method and are not specific to the ordered FIB method described in
   this document.

9. Security Considerations

   This document requires only minor modifications to existing routing
   protocols and therefore does not add significant additional security
   risks. However, a full security analysis would need to be provided
   within the protocol-specific specifications proposed for deployment.

   Security considerations related to timer values set by routers are
   noted in Appendix B.4.

10. Acknowledgments

   We would like to thank Jean-Philippe Vasseur and Les Ginsberg for
   their useful suggestions and comments.

11. Informative References

   [IPFRR-TUNNELS]
             Bryant, S., Filsfils, C., Previdi, S., and M. Shand, "IP
             Fast Reroute using tunnels", Work in Progress, November
             2007.

   [ISO10589] International Organization for Standardization,
             "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)", ISO/IEC
             10589:2002, Second Edition, November 2002.

   [LF-TIMERS]
             Atlas, A., Bryant, S., and M. Shand, "Synchronisation of
             Loop Free Timer Values", Work in Progress, February 2008.

   [RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

   [RFC4090] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute
             Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May
             2005.

   [RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC
             5714, January 2010.

   [RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free
             Convergence", RFC 5715, January 2010.

   [refs.PFOB07]
             Francois, P. and O. Bonaventure, "Avoiding transient loops
             during the convergence of link-state routing protocols",
             IEEE/ACM Transactions on Networking, Vol. 15, No. 6, pp.
             1280-1292, December 2007,
             <http://dx.doi.org/10.1109/TNET.2007.902686>.

Appendix A. Candidate Methods of Safely Abandoning Loop-Free
             Convergence (AAH)

   IP Fast Reroute [RFC5714] and loop-free convergence techniques
   [RFC5715] can deal with single topology change events, multiple
   correlated change events, and in some cases even certain uncorrelated
   events. However, in all cases, there are events that cannot be dealt
   with, and the mechanism needs to quickly revert to normal
   convergence. This is known as "Abandoning All Hope" (AAH).

   This appendix describes the outcome of a design study into the AAH
   problem and is included here to trigger discussion on the trade-offs
   between complexity and robustness in the AAH solution space.

A.1. Possible Solutions

   Two approaches to this problem have been proposed:

   1. Hold-down timer only.

   2. Synchronization of AAH state using AAH messages.

   They are described below.

A.2. Hold-Down Timer Only

   The "hold-down timer only" AAH method uses a hold-down to acquire a
   set of LSPs that should be processed together. On expiry of the
   local hold-down timer, the router begins processing the batch of LSPs
   according to the loop-free prevention algorithm.

   There are a number of problems with this simple approach. In some
   cases, the timer value will be too short to ensure that all the
   related events have arrived at all routers (perhaps because there was
   some unexpected propagation delay, or one or more of the events are
   slow in being detected). In other cases, a completely unrelated
   event may occur after the timer has expired but before the processing
   is complete. In addition, since the timer is started at each router
   on reception of the first LSP announcing a topology change, the
   actual starting time is dependent upon the propagation time of the
   first LSP. So, for a subsequent event occurring around the time of
   the timer expiry, because of variations in propagation delay, it may
   reach some routers before the timer expires and others after it has
   expired. In the former case, this LSP will be included in the set of
   changes to be considered; while in the latter, it will be excluded
   leading to serious routing inconsistency. In such cases, continuing
   to operate the loop-free convergence protocol may exacerbate the
   situation.

   The simple approach to this would be to revert to normal convergence
   (AAH) whenever an LSP is received after the timer has expired.
   However, this also has problems for the reasons above and therefore
   AAH must be a synchronous operation, i.e., it is necessary to arrange
   that an AAH invoked anywhere in the network causes ALL routers to
   invoke AAH.

   It is also necessary to consider the means of exiting the AAH state.
   Again, the simplest method is to use a timer. However, while in AAH
   state, any topology changes that are previously received or
   subsequently received should be processed immediately using the
   traditional convergence algorithms, i.e., without invoking controlled
   convergence. If the exit from the AAH state is not correctly
   synchronized, a new event may be processed by some routers
   immediately (as AAH), while those that have already left AAH state
   will treat it as the first of a new batch of changes and attempt
   controlled convergence. Thus, both entry and exit from the AAH state
   need to be synchronized. A method of achieving this is described in
   Appendix A.3.

A.3. AAH Messages

   Like the simple timer AAH method, the "AAH messages" method uses a
   hold-down to acquire a set of LSPs that should be processed together.
   On expiry of the local hold-down timer, the router begins processing
   the batch of LSPs according to the loop-free prevention algorithm.
   This is the same behavior as the hold-down timer only method.
   However, if any router, having started the loop-free convergence
   process receives an LSP that would trigger a topology change, it
   locally abandons the controlled convergence process and sends an AAH
   message to all its neighbors. This eventually triggers all routers
   to abandon the controlled convergence. The routers remain in AAH
   state (i.e., processing topology changes using normal "fast"
   convergence), until a period of quiescence has elapsed. The exit
   from AAH state is synchronized by using a two-step process. To
   achieve the required synchronization, two additional messages are
   required, AAH and AAH ACK. The AAH message is reliably exchanged
   between neighbors using the AAH ACK message. These could be
   implemented as a new message within the routing protocol or carried
   in existing routing hello messages. Two types of state machines are
   needed -- a per-router AAH state machine and a per-neighbor AAH state
   machine (PNSM). These are described below.

A.3.1. Per-Router State Machine

EVENT	Q	Hold	CC	AAH	AAH-hold
=======	===	======	====	=====	==========
RX LSP triggering change	Start hold-down timer. [Hold]	-	TX-AAH Start AAH timer. [AAH]	Restart AAH timer. [AAH]	TX-AAH Start AAH timer. [AAH]
RX AAH (Neighbor's PNSM processes RX AAH.)	TX-AAH Start AAH timer. [AAH]	TX-AAH Start AAH timer. [AAH]	TX-AAH Start AAH timer. [AAH]	[AAH]	TX-AAH Start AAH timer. [AAH]
Timer expiry	-	Trigger CC. [CC]	-	Start AAH-hold timer. [AAH-hold]	[Q]
Controlled convergence completed	-	-	[Q]	-	-

```
RX = Reception
TX = Transmission
TX-AAH = Send "go to TX-AAH" to all other PNSMs.
```

Per-Router State Table

Operation of the per-router state machine is as follows:

Operation of this state machine under normal topology change involves
only states: Quiescent (Q), Hold-down (Hold) and Controlled
Convergence (CC). The remaining states are associated with an AAH
event.

The resting state is Quiescent. When the router in the Quiescent
state receives an LSP indicating a topology change, which would
normally trigger an SPF, it starts the hold-down timer and changes
state to Hold-down. It normally remains in this state, collecting
additional LSPs until the hold-down timer expires. Note that all
routers must use a common value for the hold-down timer. When the
hold-down timer expires, the router then enters Controlled

Convergence (CC) state and executes the CC mechanism to reconverge
the topology. When the CC process has completed on the router, the
router re-enters the Quiescent state.

If this router receives a topology-changing LSP whilst it is in the
CC state, it enters AAH state and sends a "go to TX-AAH" command to
all per-neighbor state machines; this causes each per-neighbor state
machine to signal this state change to its neighbor. Alternatively,
if this router receives an AAH message from any of its neighbors
whilst in any state except AAH, it starts the AAH timer and enters
the AAH state. The per-neighbor state machine corresponding to the
neighbor from which the AAH was received executes the RX AAH action
(which causes it to send an AAH ACK), while the remainder of
neighbors are sent the "go to TX-AAH" command. The result is that
the AAH is acknowledged to the neighbor from which it was received
and propagated to all other neighbors. On entering AAH state, all CC
timers are expired, and normal convergence takes place.

Whilst in the AAH state, LSPs are processed in the traditional
manner. Each time an LSP is received, the AAH timer is restarted.
In an unstable network, ALL routers will remain in this state for
some time, and the network will behave in the traditional
uncontrolled convergence manner.

When the AAH timer expires, the router enters AAH-hold state and
starts the AAH-hold timer. The purpose of the AAH-hold state is to
synchronize the transition of the network from AAH to Quiescent. The
additional state ensures that the network cannot contain a mixture of
routers in both AAH and Quiescent states. If, whilst in AAH-hold
state the router receives a topology changing LSP, it re-enters AAH
state and commands all per-neighbor state machines to "go to TX-AAH".
If, whilst in AAH-hold state, the router receives an AAH message from
one of its neighbors, it re-enters the AAH state and commands all
other per-neighbor state machines to "go to TX-AAH". Note that the
per-neighbor state machine receiving the AAH message will
autonomously acknowledge receipt of the AAH message. Commanding the
per-neighbor state machine to "go to TX-AAH" is necessary, because
routers may be in a mixture of Quiescent, Hold-down, and AAH-hold
states, and it is necessary to rendezvous the entire network back to
AAH state.

When the AAH-hold timer expires, the router changes to Quiescent and
is ready for loop-free convergence.

A.3.2. Per-Neighbor State Machine

```
+---------------------------+-------------+----------------------+
| EVENT                     | IDLE        | TX-AAH               |
+===========================+=============+======================+
| RX AAH                    | Send ACK.   | Send ACK.            |
|                           | [IDLE]      | Cancel timer.        |
|                           |             | [IDLE]               |
+---------------------------+-------------+----------------------+
| RX ACK                    | ignore      | Cancel timer.        |
|                           |             | [IDLE]               |
+---------------------------+-------------+----------------------+
| RX "go to TX-AAH" from    | Send AAH    | ignore               |
| Router State Machine      | [TX-AAH]    |                      |
+---------------------------+-------------+----------------------+
| Timer expires             | impossible  | Send AAH             |
|                           |             | Restart timer.       |
|                           |             | [TX-AAH]             |
+---------------------------+-------------+----------------------+
```

                    Per-Neighbor State Table

   There is one instance of the per-neighbor state machine (PNSM) for
   each neighbor within the convergence control domain.

   The normal state is IDLE.

   On command ("go to TX-AAH") from the router state machine, the state
   machine enters TX-AAH state, transmits an AAH message to its
   neighbor, and starts a timer.

   On receipt of an AAH ACK in state TX-AAH, the state machine cancels
   the timer and enters IDLE state.

   In state IDLE, any AAH ACK message received is ignored.

   On expiry of the timer in state TX-AAH, the state machine transmits
   an AAH message to the neighbor and restarts the timer. (The timer
   cannot expire in any other state.)

   In any state, receipt of an AAH causes the state machine to transmit
   an AAH ACK and enter the IDLE state.

   Note that for correct operation the state machine must remain in
   state TX-AAH until an AAH ACK or an AAH is received or until the
   state machine is deleted. Deletion of the per-neighbor state machine
   occurs when routing determines that the neighbor has gone away or
   when the interface goes away.

When routing detects a new neighbor, it creates a new instance of the
per-neighbor state machine in state IDLE. The consequent generation
of the router's own LSP will then cause the router state machine to
execute the LSP receipt actions that, if necessary, will result in
the new per-neighbor state machine receiving a "go to TX-AAH" command
and transitioning to TX-AAH state.

Appendix B. Synchronization of Loop-Free Timer Values

   This appendix provides the reader with access to the design
   considerations originally described in [LF-TIMERS].

B.1. Introduction

   Most of the loop-free convergence mechanisms [RFC5715] require one or
   more convergence delay timers that must have a duration that is
   consistent throughout the routing domain. This time is the worst-
   case time that any router will take to calculate the new topology and
   to make the necessary changes to the FIB. The timer is used by the
   routers to know when it is safe to transition between the loop-free
   convergence states. The time taken by a router to complete each
   phase of the loop-free transition will be dependent on the size of
   the network and the design and implementation of the router.
   Therefore, it can be expected that the optimum delay will need to be
   tuned from time to time as the network evolves. Manual configuration
   of the timer is fraught for two reasons. Firstly, it is always
   difficult to ensure that the correct value is installed in all of the
   routers. Secondly, if any change is introduced into the network that
   results in a need to change the timer (for example, due to a change
   in hardware or software version), then all of the routers need to be
   reconfigured to use the new timer value. Therefore, it is desirable
   that a means be provided by which the convergence delay timer can be
   automatically synchronized throughout the network.

B.2. Required Properties

   The timer synchronization mechanism must have the following
   properties:

   o The convergence delay time must be consistent amongst all routers
      that are converging on the new topology.

   o The convergence delay time must be the highest delay required by
      any router in the new topology.

   o The mechanism must increase the delay when a new router that
      requires a higher delay than is currently in use is introduced to
      the network.

    o When the router that had the longest delay requirements is removed
       from the topology, the convergence delay timer value must, within
       some reasonable time, be reduced to the longest delay required by
       the remaining routers.

    o It must be possible for a router to change the convergence delay
       timer value that it requires.

    o A router that is in multiple routing areas or is running multiple
       routing protocols may signal a different loop-free convergence
       delay for each area and for each protocol.

    How a router determines the time that it needs to execute each
    convergence phase is an implementation issue and outside the scope of
    this specification. However, a router that dynamically determines
    its proposed timer value must do so in such a way that it does not
    cause the synchronized value to continually fluctuate.

B.3. Mechanism

    The following mechanism is proposed.

    A new information element is introduced into the routing protocol
    that specifies the maximum time (in milliseconds) that the router
    will take to calculate the new topology and to update its FIB as a
    result of any topology change.

    When a topology change occurs, the longest convergence delay time
    required by any router in the new topology is used by the loop-free
    convergence mechanism.

    If a routing protocol message is issued that changes the convergence
    delay timer value but does not change the topology, the new timer
    value must be taken into consideration during the next loop-free
    transition but must not instigate a loop-free transition.

    If a routing protocol message is issued that changes the convergence
    timer value and changes the topology, a loop-free transition is
    instigated, and the new timer value is taken into consideration.

    The loop-free convergence mechanism should specify the action to be
    taken if a timer change (only) message and a topology change message
    are independently generated during the hold-off time. A suitable
    action would be to take the same action that would be taken if two
    uncorrelated topology changes occurred in the network.

All routers that support loop-free convergence must advertise a loop-
free convergence delay time. The loop-free convergence mechanism
must specify the action to be taken if a router does not advertise a
convergence delay time.

B.4. Security Considerations Related to Router Timer Values

If an abnormally large timer value is proposed by a router, then
there is a danger that the loop-free convergence process will take an
excessive amount of time. If during that time the routing protocol
signals the need for another transition, the loop-free transition
will be abandoned and the default best-case (traditional) convergence
mechanism used.

It is still undesirable that the routers select a convergence delay
time that has an excessive value. The maximum value that can be
specified in the LSP or Link State Advertisement (LSA) is limited
(through the use of a 16-bit field) to about 65 seconds. When
sufficient implementation experience is gained, an architectural
constant will be specified as the upper limit of the convergence
delay timer.

Authors' Addresses

   Mike Shand
   Individual Contributor

   EMail: imc.shand@googlemail.com

   Stewart Bryant
   Cisco Systems
   10 New Square, Bedfont Lakes
   Feltham, Middlesex TW18 8HA
   United Kingdom

   EMail: stbryant@cisco.com

   Stefano Previdi
   Cisco Systems
   Via Del Serafico 200
   00142 Roma
   Italy

   EMail: sprevidi@cisco.com

Clarence Filsfils
Cisco Systems
Brussels
Belgium

EMail: cfilsfil@cisco.com

Pierre Francois
Institute IMDEA Networks
Avda. del Mar Mediterraneo, 22
Leganese 28918
Spain

EMail: pierre.francois@imdea.org

Olivier Bonaventure
Universite catholique de Louvain
Place Ste Barbe, 2
Louvain-la-Neuve 1348
Belgium

EMail: Olivier.Bonaventure@uclouvain.be
URI: http://inl.info.ucl.ac.be/