Jitter Consideration for Reactive Protocols in Mobile Ad Hoc Networks (MANETs)

draft-yi-manet-reactive-jitter-01

Juan Antonio Cordero Fuertes ⭐ Jiazi Yi ⭐ Thomas Heide Clausen
Jitter - RFC 5148

Abstract

This document provides recommendations for jittering (randomly modifying timing) of control traffic transmissions in Mobile Ad hoc NETwork (MANET) routing protocols to reduce the probability of transmission collisions.

The mechanisms described in this document are applicable to the control messages of any MANET protocol in which simultaneous transmissions by different nodes are undesirable, and that contains mechanisms, such as periodic control message transmission, triggered control message transmission, or control message forwarding, which either make a simultaneous transmission more likely, or cause one to be repeated when it occurs. This particularly applies to protocols using broadcast transmissions in wireless networks, where proactive MANET routing protocols such as [6] employ scheduled messages, where reactive MANET routing protocols such as [6] employ event-triggered message propagation over message forwarding.
Jitter - RFC5148

The mechanisms described in this document are applicable to the control messages of any MANET protocol in which simultaneous transmissions by different nodes are undesirable, and that contains mechanisms, such as periodic control message transmission, triggered control message transmission, or control message forwarding, which either make a simultaneous transmission more likely, or cause one to be repeated when it occurs. This particularly applies to protocols using broadcast transmissions in wireless networks, where proactive MANET routing protocols such as [5] employ scheduled messages, where reactive MANET routing protocols such as [6] employ event-triggered messages.

Proactive protocol:
TCs/LSAs etc. carry adjacency information only
Flooding path independence

Reactive protocols:
RREQs carry implicit path information
Reactive Routing

w/o Jitter

Dest  Next  Dist
E    E     1
A    E     2
Reactive Routing

w. Jitter

Broadcast

A

B

C

D

RREQ(A, D, 0)

+ jitter

RREQ(A, D, 0)

+ jitter

RREQ(A, D, 0)

+ jitter

RREQ(A, D, 0)

+ jitter

RREQ(A, D, 0)

+ jitter

RREP(A, D, 0)

RREP(A, D, 1)

RREP(A, D, 2)

RREP(A, D, 3)

Unicast

Dest   Next   Dist
C      C      1
A      C      3
Reactive Routing

w. Jitter

Broadcast

A

RREQ(A, D, 0)

RREQ(A, D, 1)

RREQ(A, D, 2)

B

RREP(A, D, 0)

RREP(A, D, 1)

RREP(A, D, 2)

C

D

RREP(A, D, 0)

RREP(A, D, 1)

RREP(A, D, 2)

Unicast

Dest  Next  Dist

E  E  1

A  E  2

+ jitter

+ jitter

+ jitter
Reactive Routing

“Delay Inversion”

“Turning a longer (worse) path into a path, which is traversed faster”

Path sub-optimality

(and/or)

Increased control traffic
What does RFC5148 say....

5.2. Externally Triggered Message Generation

An internal or external condition or event generation by a node. Depending upon this may trigger generation of a single message limited to, an acknowledgement message) message schedule, or rescheduling of existing messages. Collision between externally triggered messages: if more than one node is likely to respond reduce this likelihood, an externally triggered message SHOULD be jittered by delaying it by a random duration; an internally triggered message MAY also be so jittered if appropriate. This delay SHOULD be generated uniformly in an interval between zero and MAXJITTER.
Important Points...

These behaviors not just of academic interest ... **observed** in LOADng-routed networks (AMI, sensor networks, ...)

Analysis has revealed that “delay inversion” occurs with **significant probability** when using jitter according to RFC5148, and this independently from the jitter interval length, and proportional to absolute path lengths.

**LOADng:**

**Jitter Analysis:**

Window Jitter

Uniform Jitter Window Jitter
(RFC5148)

- Reduces randomness, increases (deterministic) dependency of the total delay on the path length
- Increases the probability that the RREQ packet traverses faster through a "shorter" path
- Hop-count metric implicitly assumed.....

\[ T_{J_U} \sim \text{uniform}[0, J_m] \]

\[ T_{J_W} \sim \text{uniform}[\alpha \cdot J_m, J_m] \]
Generalization of Window Jitter for non-trivial metrics

\[ LQ \in (0, 1) \; ; \; \left[ \left( 1 - LQ \right) \cdot J_m, J_m \right] \]

Window Jitter

Uniform Jitter (RFC5148)

Window Jitter

Adaptive Jitter

Generalization of Window Jitter for **non-trivial** metrics

\[ T_{J_{W}} \sim \text{uniform}[\alpha \cdot J, J] \]
- 802.11 MAC, 1km x 1km field, 2-way ground, 100 s.
- 30 concurrent route discoveries launched every 2s
- Reactive routing protocol of choice: LOADng

- No jitter:
- Standard RFC 5148 jitter, $J_m = 100$ms. Jitter is selected within $[0, 100]$ ms (mean, 50ms).
- Standard RFC 5148 jitter, $J_m = 200$ms. Jitter is selected within $[0, 200]$ ms (mean, 100ms).
- Window jitter, $\alpha = 1/2$, $J_m = 100$ms. Jitter is selected within $[50, 100]$ ms (mean, 75ms).
- Window jitter, $\alpha = 2/3$, $J_m = 150$ms. Jitter is selected within $[100, 150]$ ms (mean, 125ms)
Concluding Remarks (1)

- Jitter is a Very Good Idea for reducing collisions in flooding operations in wireless networks.

- RFC5148 based on observations and operational experiences with proactive OLSR and OLSRv2.

- RFC5148, applied to the flooding operation of RREQs in reactive routing protocols, causes delay inversion - which leads to longer paths and higher control traffic overhead.
Concluding Remarks (2)

- Window-Jitter and Adaptive Jitter are simple modulations of RFC5148-Jitter, perhaps better adapted for reactive routing protocol RREQ flooding. Seems to, substantially, reduce the probability of experiencing delay inversion, and therefore yields:

  - Shorter paths
  - Lower control traffic load
  - Fewer collisions

- Not without a cost, alas:
  - Longer route discovery delays

- Generally applicable to reactive routing protocols