Optimization of Jitter Configuration for Reactive Route Discovery in Wireless Mesh Networks

Juan Antonio Cordero
ICTEAM, Université catholique de Louvain
Belgium
juan.cordero@uclouvain.be

Jiazi YI
LIX, École Polytechnique
France
Jiazi@JiaziYi.com

Thomas Clausen
LIX, École Polytechnique
France
Thomas@ThomasClausen.org

***Abstract*—Jitter is a small, random variation of timing before message emission that is widely used in non-synchronized wireless communication. It is employed to avoid collisions caused by simultaneous transmissions by adjacent nodes over the same channel. In reactive (on-demand) routing protocols, such as AODV and LOADng, it is recommended to use jitter during the flooding of Route Request messages. This paper analyzes the cost of jitter mechanisms in route discovery of on-demand routing protocols, and examines the drawbacks of the standard and commonly used uniformly distributed jitter. The main studied drawback is denominated *delay inversion effect*. Two variations on the jitter mechanism –window jitter and adaptive jitter– are proposed to address this effect, which take the presence and the quality of traversed links into consideration to determine the per-hop forwarding delay. These variations allow to effectively reduce the routing overhead, and increase the quality of the computed paths with respect to the standard uniform jitter mechanism. Simulations are also performed to compare the performance of different jitter settings in various network scenarios.**

I. INTRODUCTION

Since the late nineties, the MANET (Mobile Ad-hoc Networks) working group of the Internet Engineering Task Force[1] has been investing substantial efforts in developing routing protocols for MANET and wireless mesh networks.

In this kind of self-organized and decentralized networks, channel collisions (*i.e.*, collisions due to the simultaneous transmission of adjacent nodes over the same channel) constitute an important source of packet losses.

A. Packet Collisions and Jittering Techniques

Different ways have been explored to address this issue and minimize its impact in wireless multi-hop networks. Classic MAC (Medium Access Control) collision avoidance mechanisms [1] [2] are not suited to current wireless mesh scenarios and are unable to solve all possible cases of collisions (*e.g.*, broadcast or multicast transmissions, collisions between non-neighboring nodes). Recent research efforts [3], [4] have focused on other alternatives, such as the use of multi-channel assignments in wireless sensor networks. These approaches are able to reduce the problem of collisions in potentially dense networking scenarios, at the cost of adding an additional complexity layer (or relying on previous knowledge of the network topology) and renoncing to the semibroadcast capability [5] of the wireless network.

According to IETF, the problem of packet collisions in a wireless multi-hop mesh network can be further alleviated by introducing jitter (a small, random delay on transmissions) in the network layer. In RFC 5148 [6], the use of jitter is recommended for MANETs and wireless mesh networks as a *simple* collision avoidance mechanism for routing protocol control traffic, such as periodically scheduled packets, or event-triggered packets.

For reactive protocols such as AODV (On-demand Ad hoc Distance Vector routing protocol [7]) and LOADng (Lightweight On-demand Ad hoc Distance Vector Routing protocol - Next Generation [8]), jitter is recommended during route discovery. Route Request messages are flooded in the network until they reach their destination. In these flooding processes, concurrent retransmissions of the same message by adjacent nodes cause collisions.

B. Related Work

After the standardization of jittering techniques by the IETF [6], and their implementation in different routing protocols, there has been some research to evaluate and discuss the impact of these techniques in the performance of the protocols making use of them. [9] introduced an analytical model for investigating the impact of the standardized jitter mechanism on network-wide packet dissemination, and studied and quantified the additional delay incurred, the reduction in number of transmissions, and the effect of jitter in packet size. [10] presented the relationship between the maximal jitter duration and the probability of successful transmission, and provided a comparison between different strategies of implementing jitter mechanisms. [10] concluded that implementing jitter at any layer above IP (*e.g.* at the transport or application layer) brings virtually no benefits. Finally, [11] began to explore variations in the jitter distribution, in a research workline that is continued in this paper.

C. Statement of Purpose

This paper studies the optimization of jitter mechanisms for route discovery of reactive protocols. During route discovery, Route Request messages are flooded through the network in order to discover available routes from (requesting) source to (requested) destinations. A Route Request message is rebroadcast immediately after first received by an intermediate node.

---

[1] IETF MANET working group: http://datatracker.ietf.org/wg/manet/.

Jitter is thus used to reduce the probability that neighboring nodes will transmit at the same time.

With the method introduced in [6], jitter values are distributed uniformly between 0 and a maximum value J_m. While this can reduce the collisions by randomizing the adjacent transmission, the uniform distributed jitter also brings side effects, in particular more routing overhead and sub-optimal paths.

In this paper, the jitter behavior is first analyzed to investigate its impact on network performance. Two variations on the mechanism described in RFC 5148 [6] –window jitter and adaptive jitter– are then proposed to reduce the routing overhead and discover better quality (closer to optimal) paths. Different settings are implemented and compared in simulations.

D. Paper Outline

The remainder of this paper is organized as follows: section II introduces the background of reactive protocols for mesh networks and the jitter technique used for message flooding. In section III, the drawback of uniform distributed jitter, named *delay inversion effect*, is analyzed, and followed by the proposal of adaptive jitter mechanism. A performance study and a comparison of different jitter mechanisms are presented in section IV. The paper is concluded in section V.

II. REACTIVE PROTOCOLS AND APPLICATION OF JITTER

This section describes the basic operations of the reactive protocol in wireless ad hoc and mesh networks. Then the jitter mechanism and its impact on flooding performance are briefly introduced and discussed.

A. Basic Operations of On-demand Routing Protocols

In reactive protocols, routes are computed on demand, *i.e.*, only when a data transmission to an unknown destination is expected. Acquisition and maintenance of routes are based on two mechanisms: *route discovery* and *route maintenance*.

1) Route Discovery: Route REQuest (RREQ) messages are flooded through the network until they reach the sought destination – at which point that destination generates an RREP (Route REPly), which is unicast along the reverse path to the RREQ source. RREQ and RREP messages carry a monotonically-increasing sequence number, permitting both duplicate detection and detecting which of two messages contains the most "fresh" information. Two flooding modes are possible: the *shortest-delay* mode and the *shortest-path* mode. Depending on the flooding mode, RREQ forwarding and RREP generating rules may be slightly different.

Under the *shortest-delay* mode, routers in the network only forward the first RREQ message received from a given source to a given destination – forthcoming RREQs with the same pair $(src, sequence\_number)$ will be dropped, even if they advertise better paths than the first one. The requested destination behaves similarly: it only sends back one RREP upon the first reception of an RREQ from a given source. Routes discovered in this mode may be thus suboptimal, but they are acquired with minimal delay.

In contrast, under the *shortest-path* mode, routers may forward or reply to an RREQ message several times, if the traversed route is better than the one traversed by previously forwarded/replied RREQs. This improves the quality of the acquired routes, at the cost of increasing considerably the overhead associated to route discovery processes.

2) Route Maintenance: It is performed when an actively used route fails, *i.e.*, when a data packet cannot be delivered to the next hop towards the intended destination. On detecting that a route has failed, a Route Error (RERR) message is generated. On receiving such an RERR message, the source of the failed data packet can initiate a new *Route Discovery* procedure to re-establish connectivity.

B. Jitter Technique for Route Request (RREQ) Flooding

Simultaneous packet transmissions –as those performed in reactive protocols during *Route Discovery* processes– are likely to cause packet losses in wireless mesh networks, due to collisions between concurrent transmissions of routers having (at least) a common neighbor. In order to prevent or minimize these collisions, RFC 5148 [6] recommends the use of jitter for different cases in which packets may be expected to be sent concurrently. Several well-known reactive protocols (*e.g.*, AODV [7], LOAD [12], LOADng [8]) use or provide support to jitter when flooding RREQ packets over a wireless mesh network.

Without jitter, a router receiving an RREQ packet to be forwarded retransmits it immediately after processing. As retransmissions in neighboring routers are triggered by this single event (the reception of the RREQ packet), there is a high probability of collision. Instead, when using jitter, every receiving node adds a small, random delay before rebroadcasting the RREQ packet. RFC 5148 [6] recommends that delays are selected following an uniform distribution between 0 and a maximum jitter value, J_m. Note that this is the maximum entropy distribution among those assigning continuous jitter values between 0 and J_m [13]: the use of this distribution thus maximizes the randomness of the total delay suffered by an RREQ packet sent along a certain path.

Other than prevention of packet collisions from simultaneous transmissions, the use of jitter in flooding has two immediate additional effects:

(i) the RREQ flooding, and therefore the route discovery, is slowed, and
(ii) nodes need larger buffers to store packets that have been received, but not yet forwarded.

The trade-off between these drawbacks and the reduction in the probability of collisions can be controlled by way of the length of the jitter interval, J_m [9].

III. JITTER: UNIFORM, WINDOW AND ADAPTIVE DISTRIBUTIONS

This section analyzes the *delay inversion effect*, a side effect of uniform distributed jitter observed when performing Route

Request (RREQ) flooding. Two variations on jitter distribution –window jitter and adaptive jitter– are then proposed and examined to alleviate this side effect.

A. The Delay Inversion Effect

The fact that RREQ messages reach their destination with a uniformly-distributed delay at each intermediate hop presents some drawbacks, in terms of path sub-optimality and/or control traffic inefficiency.

Consider the topology shown in Figure 1, and assume that node A floods (broadcasts) an RREQ to identify a route towards D. Under normal operation of a reactive routing protocol (without jitter), the RREQ would reach D through the path $p_2 = \{A, E, D\}$ faster than through the path $p_1 = \{A, B, C, D\}$, assuming that processing times at each intermediate node, before retransmission, are similar.

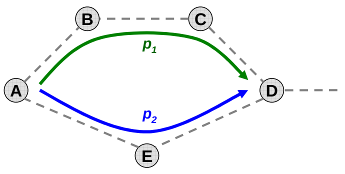


Figure 1. Topology example. Node A tries to broadcast an RREQ message through the network, through paths p_1 and p_2.

If a uniform random distribution $[0, J_m]$ is used at each hop to determine an additional delay before retransmission, the message copy sent through the longer path (in number of hops), p_1, may reach the destination faster than the message copy over p_2 with a non-negligible probability. The example of Figure 2 illustrates this case.

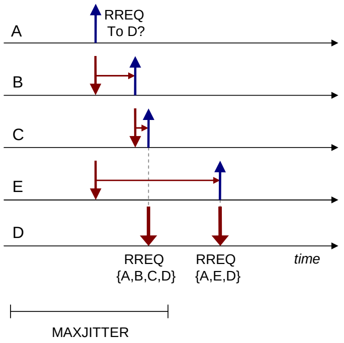


Figure 2. Example of jitter values assignement for an RREQ from A towards D, in the topology of Figure 1. RREQ through longer path $\{A, B, C, D\}$ travels faster than the one through shorter path $\{A, E, D\}$

Consider the transmission of an RREQ packet from A, received simultaneously at B and E. Although the RREQ needs to traverse two hops (B and C) to reach D via p_1, and only one (E) via p_2, the RREQ sent across p_1 would be received first at D if $j_E > j_B + j_C$, as shown in Figure 2.

Router D would reply to the Route Request from A with an RREP that advertises path p_1, which is suboptimal. When the RREQ traversing p_2 reaches D, D would reply again to A's Route Request with the (shorter) path p_2. This implies that A would get, and possibly use for a certain amount of time, a suboptimal path towards D (p_1), and it would need two RREP from D in order to learn the optimal path from A to D. If D was not the destination of the requested route, but only an intermediate router towards that destination, then D would retransmit the RREQ twice as it is received from p_1 and then p_2.

This example illustrates that the use of uniform random distribution for jitter values when forwarding RREQ packets during route discovery in a reactive routing protocol may lead to cases in which "*transmissions over longer paths get first*". This effect is hereafter denominated *delay inversion* caused by jitter, and is more frequent in long paths (in number of hops), due to the fact that the range in which total jitter values are possible (adding all per-hop jitter values) has a linearly growing upper bound (nJ_m, where n is the path hop length) and a fixed lower bound set to 0 [11]. Delay inversions are harmful due to at least three undesirable effects:

(i) increase of sub-optimality of reported routes,
(ii) growth of unnecessary RREQ broadcast traffic, and
(iii) growth of unnecessary RREP (unicast) traffic.

B. The Window Jitter

The window jitter distribution modifies the uniform distribution of RFC 5148 by introducing a minimum jitter interval in each hop. Jitter values are then instances of a random variable $TJ_W \sim Uniform[\alpha J_m, J_m]$, where $\alpha \in (0, 1)$ and αJ_m is a minimum jitter value. Note that $\alpha = 0$ corresponds to the uniform jitter distribution specified in RFC 5148, $\alpha = 1$ would imply a deterministic delay (of length J_m). The fact that $\alpha \neq 0$ entails that the lower bound for the RREQ delay grows linearly with the length of the traversed path.

Figure 3 shows the probability density functions (PDFs) for the jitter value as specified in RFC 5148 (TJ_U) and the modified jitter random variable (TJ_W).

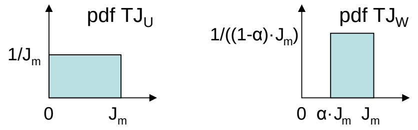


Figure 3. PDF of random variables TJ_U (RFC 5148), TJ_W (window jitter).

The window jitter reduces the randomness and increases the (deterministic) dependency of the total RREQ delay to the length n of the traversed path. When assigning jitter values according to the distribution of random variable J_w, the total delay caused by jitter in a path of n hops belongs to the interval $[n\alpha J_m, nJ_m]$ ($\alpha \neq 0$). The trade-off between randomness and path length deterministic dependence can be controlled by way of parameter $\alpha \in (0, 1)$: the closer α is to 1, the more deterministic becomes the total delay of an RREQ packet with respect to the path length.

Under the window jitter distribution, each additional hop in the path traversed by an RREQ packet causes at least an additional delay of αJ_m. As shown analytically in [11], this increases the probability that the RREQ packet traverses faster through a ”shorter” path, in number of hops, rather that through a ”longer” path, which is considered worse for routing. This model thus assumes that longer paths are preferable to shorter paths, that is, a hop count metric is implicitly assumed.

C. The Adaptive Jitter for Non-Trivial Metrics

The window jitter principle can be naturally extended to non-trivial link metrics, for instance based on probability of successful transmission (ETX [14]) or available bandwidth in the link. This extension of window jitter to link metrics other than hop count is denominated *adaptive jitter*.

Given a link quality indicator $LQ \in (0, 1)$ ($LQ \longrightarrow 1$ for high quality links), jitter values are selected uniformly within the interval $[(1-LQ)J_m, J_m]$. This reduces the probability of delay inversion or, equivalently, increases the probability that an RREQ packet is forwarded faster by routers receiving it on better links.

Note that the window jitter distribution presented in section III-B corresponds to the particular case of $LQ = 1-\alpha$ for all available links.

IV. SIMULATIONS AND RESULTS

To evaluate the performance of different jitter mechanisms, simulations are performed. In this section, the simulation results are presented with further discussion.

A. Simulation Setup

The performance of the three different jitter configurations (standard, window and adaptive) is evaluated in *shortest-delay mode* and *shortest-path mode* of RREQ flooding (see section II-A1) for different network scenarios. Network scenarios are characterized by triplets $(N, \rho, metric)$, where:

- N stands for the network population (number of nodes),
- ρ stands for the network node density (number of nodes per km^2), and
- $metric$ identifies the link metric model – uniform (hop count, in which all available links have cost 1) or random (links have a random integer cost from 1 to 10).

Values for each network profile are averaged over 20 samples, each sample corresponding to a random distribution of nodes over the network grid, in which RREQs are sent from a fixed random source to a fixed random destination. Each value related to a distribution corresponds to the average of 10 RREQ floodings simulated between source and destination.

The following aspects are used to evaluate the performance of different jitter mechanisms:

- Number of collisions. A collision is counted when a router receives two transmissions simultaneously.
- Optimality index. It measures the quality of discovered paths. Given a source s and a destination d, the optimality index for a path between s and d is the quotient of the cost of this path and the cost of the shortest (minimal) path between s and d.
- Routing overhead. The number of RREQ or RREP retransmissions.
- Route discovery delay. In *shortest-delay mode*, it is the time required to obtain the first path. In *shortest-path mode*, it is the time to discover the best path.

B. Impact of Parameters and Considerations

Jitter distribution is characterized by way of two parameters: the maximum jitter value, J_m (used in all three configurations); and the α parameter ($\alpha \in (0, 1)$, used only in window and adaptive jitter variations), such that αJ_m is the non-zero minimum jitter value. It is assumed that J_m has the same value for all routers in the network, and satisfies the recommendations of RFC 5148 [6]. In the window jitter, α is a fixed value, the same for all routers in the network; in the adaptive jitter, $\alpha = 1 - LQ$ depends on the link quality value and therefore may change during the network operation.

The effect of these two parameters in the performance of each configuration is relatively straightforward. In [11] it was proven that changes in the absolute value of J_m have no impact in the delay inversion effect. For simplicity, simulations in this section assume $J_m = 1sec$.

As indicated in section III-B, the value of α traduces the trade-off between randomness and (deterministic) influence of the path quality. Consequently, as α approaches 1, forwarding delays become more deterministic (and longer), the delay inversion effect becomes more rare and collisions are more likely (and inversely when $\alpha \longrightarrow 0$). As this behavior is immediate from the parameter definition, this section focuses on the comparison between the different configurations, rather than on the impact of the variation of these parameters. For simplicity and clarity in the evaluation, the figures are shown for $J_m = 1sec$ (for all configurations) and $\alpha = 0.5$ (for window jitter).

C. Simulation Results and Discussion

1) Uniform link metrics: The simulation of the *shortest-path* mode of route discovery in networks with uniform link cost (hop count) shows that window jitter configuration is able to reduce significantly the number of collisions caused by RREQ flooding, when compared to the standard jitter configuration. Figure 4 shows that the collision reduction becomes more relevant as the network density grows.

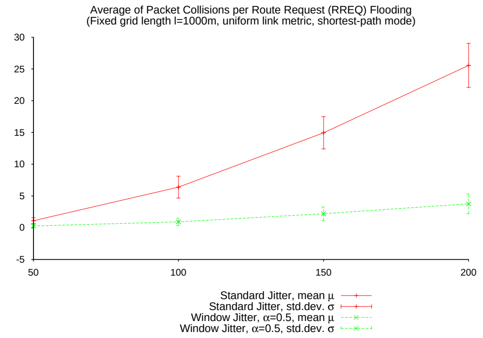


Figure 4. Number of collisions, shortest-path RREQ forwarding

This reduction is due to the fact that the use of window jitter, when compared with standard jitter, increases the probability that the first RREQ received by an intermediate router (or a destination) has traversed the *shortest* path (according to the metric in use) available, and therefore no additional RREQ retransmissions need to be performed (and no additional Route Replies need to be sent after the first one) over a path with *better* quality than the one previously advertised. The better the quality of the first advertised path, the fewer control packets (RREQ and RREP) involved in a single Route Discovery process, and the less likely packet collisions.

Improvement of discovered route quality can be observed through of the optimality index in *shortest-delay mode*. Figure 5 illustrates the optimality index of window jitter and uniform jitter depending on the network density. When routers are only allowed to forward the first RREQ received from a given source towards a given destination, the use of window jitter improves significantly the quality of the routes identified through RREQ flooding. This confirms the results from the theoretical analysis of [11] about the probability of delay inversion in standard jitter and window jitter.

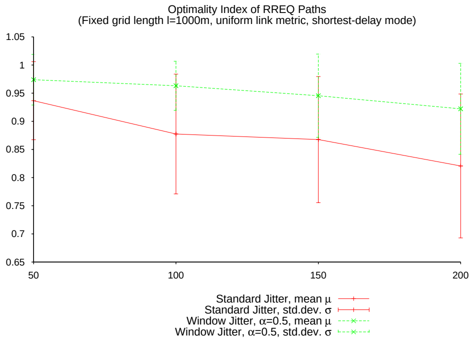
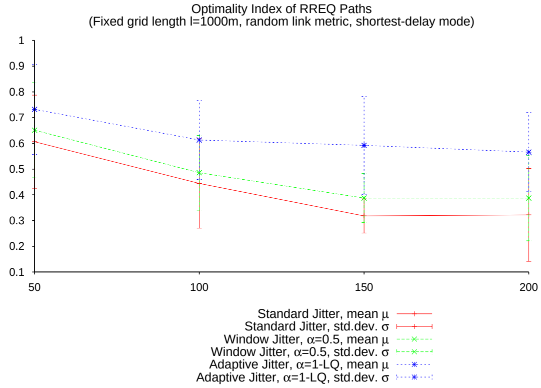


Figure 5. Optimality index, shortest-delay RREQ forwarding

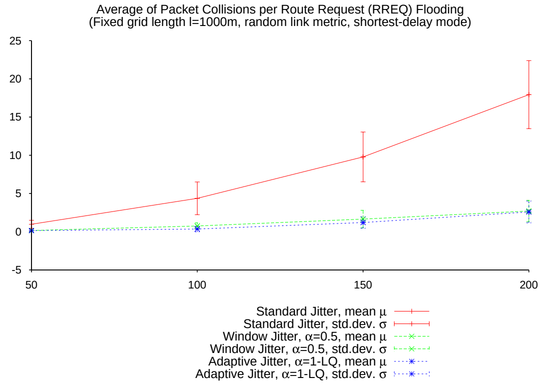
As mentioned in section III-B, window jitter relies on the principle that RREQs traversing *less* hops are preferable (better) to RREQs traversing *more* hops, and therefore the later should be delayed with respect to the former – it implicitly assumes a constant link metric, and it is able to provide a significant improvement in the route discovery performance when no more information about link quality is available.

2) Shortest-delay mode over non-trivial link metrics: The advantages of window jitter with respect to standard jitter, however, become less significant when link metrics are not uniform: the ability to identify better paths by introducing fixed minimum delays (αJ_m) per hop degrades, as Figura 6 indicates. For these cases, the use of the adaptive jitter presented in section III-C reveals more adequate, according to simulation results. This is because routers using adaptive jitter can take the actual link metric (e.g., ETX, bandwidth, etc.) into consideration, rather than the single presence of these links in the path.

Figure 6(a) shows that adaptive jitter clearly outperforms window jitter and standard jitter in terms of optimality index. As shown in Figure 6(b) for random link quality values, this benefit from the adaptive jitter is compatible with a low level of packet collisions (similar to the level achieved with window jitter, and significantly lower than the level achieved with standard jitter) in networks with heterogeneous link qualities (*i.e.*, non-uniform metrics).



(a) Optimality index

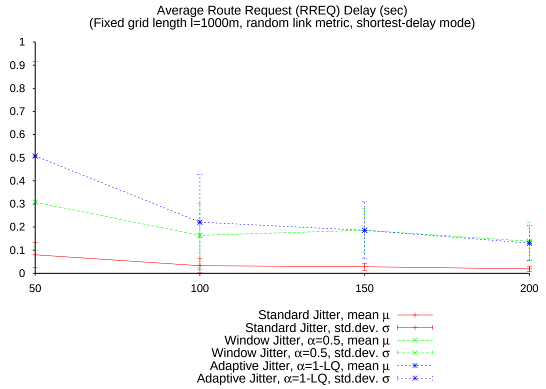


(b) RREQ packet collisions

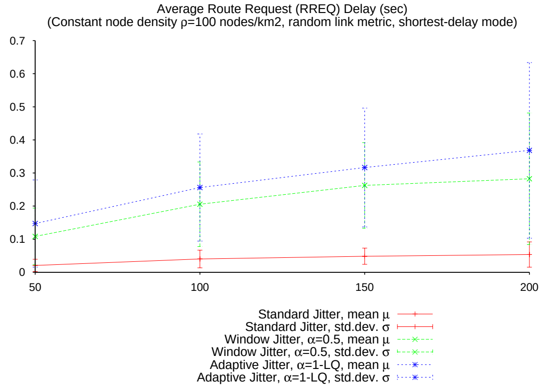
Figure 6. Shortest-delay RREQ flooding

Discrimination of RREQs based on quality of traversed links is performed by introducing pre-forwarding delays. This entails a trade-off between RREQ path optimality and RREQ forwarding delay, as it can be observed in Figure 7 for the three considered jitter configurations: in general, the better the path indicated in the first RREQ received by the intended destination, the more delay between the RREQ transmission by the source and its reception in the destination. This can be observed, in particular, for networks of constant node density (Figure 7(b)). Results from Figure 7(a) indicate, in addition, that additional delay caused by adaptive jitter with respect to window jitter strongly depends on the network density: as more paths are available to reach the destination (because the network is denser), heterogeneity of the quality of the involved links in flooding is also higher and the adaptive jitter configuration allows to deliver Route Requests (RREQs) faster, while window jitter configuration cannot reduce the *per*-hop delay beyond a minimum value αJ_m.

3) Shortest-path mode over non-trivial link metrics: The use of adaptive jitter in the *shortest-path* mode of route discovery is also beneficial, although not due to the same reasons (RREQ path quality improvement, mainly) as in the *shortest-delay* mode. The fact that routers are able to forward

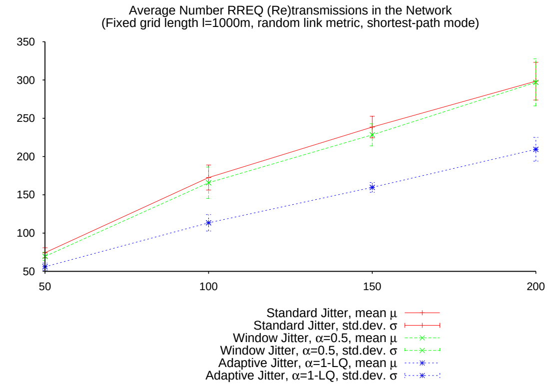


(a) Fixed grid

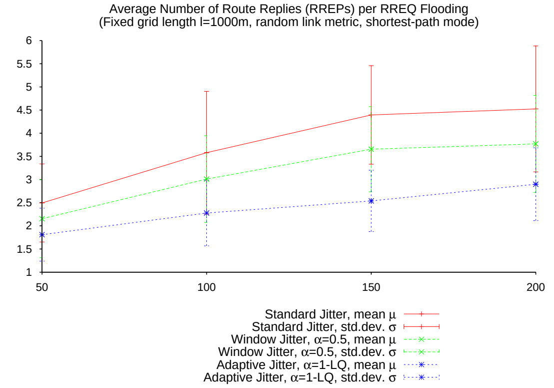


(b) Constant node density

Figure 7. Route discovery delay, shortest-delay RREQ forwarding



(a) Number of RREQ (re)transmissions



(b) Number of RREP (re)transmissions

Figure 8. Route overhead per route discovery, shortest-path RREQ forwarding

RREQs indefinitely, any time that they receive a RREQ with a better route than the last forwarded RREQ, entails that RREQ flooding ideally provides the optimal route between source and destination, if it terminates successfully (without packet losses, collisions or inaccuracies in link quality estimation). However, the *shortest-path* mode with static jitter configurations (standard jitter, window jitter) presents a relevant drawback: as every packet may forward each RREQ several times, and the source may send several RREP to the same destination, probability of packet collisions and route discovery failure also increases – more significantly for dense networks. Figures 8(a) and 8(b) show the evolution of RREQ retransmissions and RREP transmissions per route discovery, when the network density increases. It can be observed that the use of adaptive jitter, by increasing the quality of the firstly-discovered paths, entails a reduction in the number of control packets *per* route discovery (RREQ retransmissions and Route Replies) up to 30%, with respect to the static configurations.

Figure 9 shows the average RREQ delays for the different jitter configurations when using *shortest-path* (sh-p) and *shortest-delay* (sh-d) modes. For any given configuration, delay for the *shortest-path* mode is always longer or equal to the delay for the *shortest-delay* mode: in the later, the flooding terminates when the destination receives the first RREQ; in the former, the flooding terminates when the destination receives the RREQ through the best path, which can correspond to the first or to a posterior reception. More interestingly, two observations can be drawn from Figure 9. In first term, RREQ delay caused by adaptive jitter decreases with the network density (a result consistent with what was shown in Figure 7), while, in contrast, standard and window jitter present in the *shortest-path* mode a roughly constant delay with respect to network density. In second term, the gap between RREQ delays in *shortest-path* and *shortest-delay* modes, *i.e.*, the additional delay caused by reception in the destination of better RREQ packets later to the first, is different for each configuration. The adaptive jitter configuration has the smallest gap between modes, which is consistent with the previous observation about the quality of first-received RREQs at the destination. The significant difference between modes when using window jitter is another indication, in turn, of the poor performance achieved by this configuration in networks with diverse link qualities – as the non-trivial link metrics scenarios considered in this section.

V. CONCLUSION

The use of jitter for packet flooding has been proved beneficial in wireless mesh networks. The addition of a random delay before retransmission of a flooded packet helps reducing the number of collisions due to concurrent wireless transmissions from neighboring nodes. Jitter techniques for

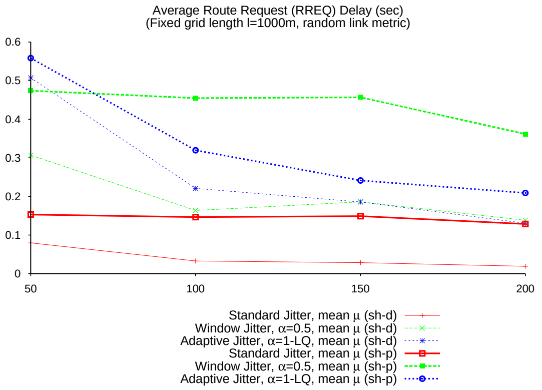


Figure 9. Route discovery delay, shortest-path and shortest-delay RREQ forwarding

flooding, as specified in RFC 5148, present however significant side-effects when employed in route discovery processes of reactive routing protocols, that need to be taken into account: in previous work [11], it was identified the *delay inversion effect*, by which the use of jitter may lead the network to select suboptimal routes between sources and destinations.

This paper explores and evaluates different settings in the statistical distribution of jitter values in order to minimize this effect, following the work started in [11]. Two modifications over the distribution of RFC 5148 are studied via network graph simulations: the window jitter distribution and the adaptive jitter distribution. For both configurations, the impact in the RREQ flooding performance is measured for different link metrics, showing that they are able to improve substantially the quality of the discovered routes (and therefore, reducing the amount of involved control packets and the probability of collisions), at the cost of increasing the delay of RREQ flooding. Depending on the intended application (*e.g.*, data collection and transmission with long sampling periods in wireless sensor networks), this additional delay may be a reasonable price for acquiring and using better routing paths.

Window jitter outperforms standard jitter in hop-count networks, but performs poorly when link quality values are heterogeneous. In these scenarios, the adaptive jitter presents clear advantages with respect to the two static configurations (standard and window jitter). When route discovery is performed in shortest-delay mode, adaptive jitter enables routers to discover significantly better paths than those obtained otherwise. Under the shortest-path mode, route quality is similar for all settings, but adaptive jitter reduces the number of RREQs and RREPs up to 30%. There is still a trade-off between route quality and flooding delay, meaning that the use of adaptive jitter entails slower flooding processes than standard or window jitter, but this additional delay becomes less significant as the network density grows: in dense networks, adaptive jitter causes flooding delays comparable to window jitter, with a substantially less overhead and optimal paths. This property makes it more interesting in resource constrained networks, such as battery charged mesh networks and sensor networks.

REFERENCES

[1] L. Kleinrock and F. A. Tobagi, “Packet switching in radio channels: Part i - carrier sense multiple-access modes and their throughput-delay characteristics,” *IEEE Transactions on Communications, vol. 23, no. 12*, 1975.

[2] P. Karn, “Maca, a new channel access method for packet radio.” Proceedings of the 5th ARRL/CRRL Computer Networking Conference, London, Ontario, Canada, September 1990.

[3] Q. Yu, J. Chen, Y. Fan, X. S. Shen, and Y. Sun, “Multi-channel assignment in wireless sensor networks: A game theoretic approach.” Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM 2010), San Diego, USA, March 2010.

[4] Y. Wu, J. A. Stankovic, T. He, and S. Lin, “Realistic and efficient multi-channel communications in wireless sensor networks.” Proceedings of the 27th IEEE International Conference on Computer Communications (INFOCOM 2008), Phoenix, USA, April 2008.

[5] I. Chakeres, J. Macker, and T. Clausen, “Mobile Ad hoc Network Architecture,” The Internet Engineering Task Force, November 2007, internet Draft, work in progress, draft-ietf-autoconf-manetarch-07.

[6] T. Clausen, C. Dearlove, and B. Adamson, “Jitter Considerations in MANETs,” IETF Inf. RFC 5148, February 2008.

[7] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” Experimental RFC 3561, July 2003.

[8] T. Clausen, A. C. de Verdiere, J. Yi, A. Niktash, Y. Igarashi, H. Satoh, and U. Herberg, “The lln on-demand ad hoc distance-vector routing protocol - next generation,” The Internet Engineering Task Force, October 2011, internet Draft, work in progress, draft-clausen-lln-loadng.

[9] J. A. Cordero, P. Jacquet, and E. Baccelli, “Impact of jitter-based techniques on flooding over wireless ad hoc networks: Model and analysis.” Proceedings of the 31st IEEE International Conference on Computer Communications (INFOCOM 2012), Orlando, USA, 2012.

[10] R. Friedman, D. Hay, and G. Kliot, “Jittering Broadcast Transmissions in MANETs: Quantification and Implementation Strategies,” Department of Computer Science, Technion, Tech. Rep., 2009. [Online]. Available: http://www.cs.technion.ac.il/ gabik/publications/Jitter-submit2009.pdf

[11] J. Yi, J. A. Cordero, and T. Clausen, “Jitter Considerations in On-Demand Route Discovery for Mobile Ad Hoc Networks,” Université catholique de Louvain / École polytechnique, Tech. Rep., 2012.

[12] K. Kim, S. D. Park, G. Montenegro, S. Yoo, and N. Kushalnagar, “6LoWPAN Ad Hoc On-Demand Distance Vector Routing,” June 2007, Internet Draft, work in progress, draft-daniel-6lowpan-load-adhoc-routing-03.

[13] K. (ed.), *Entropy Measures, Maximum Entropy Principle and Emerging Applications*, 1st ed. Springer, 2003.

[14] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom)*, San Diego, California, USA, Sep 2003, p. 8.