Performance Evaluation of Routing Protocols in WSNs Arranged in Linear Topologies

Luca Bergesio, Mirko Franceschinis, Maurizio Spirito, and Riccardo Tomasi

1 Introduction

The application of Wireless Sensor Network (WSN) technology to many heterogeneous fields, such as environmental monitoring, control and automation, logistics, assisted living, and e-health, has widened very much recently. Intelligent Transportation Systems (ITS) is another area where the use of low-consumption wireless devices has shown a promising interest due to cable replacement opportunity and reduced system maintenance costs.

Within the ITS context, monitoring vehicular traffic and detecting road accidents along a road section through WSN systems is one of the most attractive applications. The basic idea is to deploy a number of sensor nodes along the roadside, at regular distances from the predecessor and from the successor, thus forming a linear chain. In order to detect the transit of vehicles and car crash events, each node should be equipped with suitable sensors: e.g., accelerometers, pyrometers, and magnetic sensors. Traffic information locally achieved by a sensor node should be delivered to a sink node, positioned at one edge of the chain, which possibly performs data fusion operations on the pieces of traffic information coming from the many sensor nodes [1].

Each sensor node and the sink node are equipped with a radio transceiver making them able to communicate with each other. However, while low power consumption is a desired feature characterizing WSN technology, it also implies constrained node transmission power and consequently reduced radio coverage when compared with the typical length of the road section under monitoring. Definitively, multihop communication is necessary to let the sink node receive data packets from distributed sensor nodes.

The investigation of networks arranged in linear topologies, the just mentioned WSN as well as the metropolitan network standard DQDB based on a two-level grid, is very interesting from both practical and theoretical viewpoints. On the one hand,

L. Bergesio, M. Franceschinis, M. Spirito (✉), and R. Tomasi
Pervasive Radio Technologies Lab, Istituto Superiore Mario Boella (ISMB), Torino, Italy
e-mail: guglielmo@ismb.it

L. Bergesio
Dipartimento di Automatica e Informatica (DAUIN), Politecnico di Torino, Italy

D. Giusto et al. (eds.), *The Internet of Things: 20th Tyrrhenian Workshop on Digital Communications*, DOI 10.1007/978-1-4419-1674-7_3,

WSN applications inducing linear topologies are common in the real world, and the vehicular traffic monitoring mentioned so far is only an example. Other application fields could be the structural monitoring of infrastructures like railway bridges [2] and the industrial process monitoring of a production line. On the other hand, linear topologies are characterized by symmetries making possible the derivation of closed-form results through mathematical modeling analysis. Linear topologies play a central role in several works in the literature. The objective in [3] is to determine the best deployment of a WSN whose monitored area is known, the reference performance metric being the network lifetime. The goal of [4] is the simulation analysis and comparison of some routing protocols. [5] presents an analytical model to investigate the performance limit of a WSN as a function of the number of nodes, focusing on MAC protocols. The authors of [6] deal with the combination of directional antennas and suitable MAC protocols in linear WSNs and explicitly mention roadside or highway scenarios as reference applications. A cross-layer methodological study based on convex programming is presented in [7]. Finally, a linear topology is the scenario considered in [8], where an improved version of S-MAC protocol is introduced.

Inspired by the relevance of linear topologies and motivated by vehicular traffic monitoring through roadside WSN deployment as reference application (in particular, see [1] and the EU-funded SAFESPOT Integrated Project [9]), in this paper we propose three routing algorithms designed for WSN systems arranged in linear topologies and discuss their performance in terms of information delivery success rate and delay. The three algorithms, called *Single Hop (SH)*, *Limited Flooder (LF),* and *Hopefully Longest Jump First (HLJF)*, can be categorized as geographic routing: they are all based on packet forwarding rules exploiting the strict correlation between node address and relative position in the linear topology. The first two algorithms, *SH* and *LF*, are simplistic and are used as benchmarks. The performance evaluation is carried out experimentally. The experimental test-bed is based on an indoor WSN consisting of 11 Telos motes, on which the three routing strategies have been implemented.

The chapter is organized as follows. Section 2 introduces the system in all its aspects, including WSN deployment, test application, and performance metrics. A detailed description of the proposed routing protocols is provided in Sects. 3, 4, and 5, respectively. In Sect. 6, experimental results are presented and protocols performance commented as well. Finally, Sect. 7 briefly sketches future work and concludes the paper.

2 System Description

2.1 WSN Deployment

We consider a WSN composed of $N+1$ nodes: N sensor nodes, addressed from 1 to N, and 1 sink node whose id is 0. In light of the network application illustrated in Sect. 2.2, hereafter sensor nodes and the sink node are referred to respectively as Slave Nodes (SNi, where $i = 1, \ldots, N$) and the Master Node (MN). We assume

Fig. 1 Schematic representation of a WSN composed of $N+1$ nodes, 1 MN and N SNs, arranged in a linear topology

that the monitored area can be considered monodimensional, i.e., nodes are deployed along a line, in increasing order of address, as shown in Fig. 1.

2.2 Test Application

The performance of routing protocols is experimentally evaluated based on a test application implemented on WSN nodes. In order to estimate the end-to-end communication effectiveness between the MN and the many SNs, the following simple querying paradigm is utilized. The MN interrogates SNi by transmitting a query packet destined to the final recipient SNi. The query packet is forwarded by intermediate nodes until, hopefully, it is received by SNi. Depending on the specific multihop algorithm implemented on nodes, the query packet follows one or multiple paths joining MN to SNi. If SNi receives the query packet, it reacts by sending back a reply packet to the MN. Since packet losses can be experienced along paths, it is not guaranteed that a packet round trip concludes successfully. The MN progressively queries sensor nodes in sequence, from SN1 to SNN. Nonetheless, query cycles come one after another in order to collect large sets of data samples. The test application is run until a statistically sufficient number of samples is collected and the end-to-end packet delivery success rate can be estimated as the fraction of round trip successfully completed. A constant time interval, whose duration is longer than the maximum round trip time experienced, elapses between any two consecutive queries.

2.3 System Cares and Assumptions

A number of system cares and assumptions, discussed in the following, are taken and made in order to simplify the analysis of experimental results.

- The distance between any two adjacent WSN nodes, MN and SN1 or SNi and SN$i+1$, for all $i = 1, \dots, N-1$, is constant in the deployment of our WSN system inside ISMB Lab.
- Communication channel features are supposed not to vary along time and space, or, at most, the time dependence could be very slow. This appears reasonable in the indoor environment where we deployed our WSN, although long duration

test sessions inside ISMB Lab have exhibited appreciably different channel conditions when comparing day vs. night and working vs. holiday days.

- All nodes have equivalent hardware performance, in particular referring to transmission power, receiver sensitivity, and on-board antennas. Even if this sounds as an obvious assumption, it is worth noting that nominally identical hardware components could exhibit significantly different performance due to typical tolerances and functioning ranges.
- The Free Space Path Loss (FSPL) model can be adopted to characterize node-to-node signal propagation. This model neglects the impact of the specific environment and represents a very simplified model, particularly in indoor scenarios. On the other hand, the ISMB Lab in which tests have been conducted is overall space homogeneous.
- When running experiments in an indoor space-constrained scenario, Telos transmission power is set to the minimum value, i.e., $-25\,\text{dBm}$, in order to severely limit the radio coverage of a node and to balance the typically short distance among nodes.

2.4 Performance Evaluation Metrics

Experimental analysis is based on the joint evaluation of the two following performance metrics:

- $P_{e2e}(n)$: The end-to-end information delivery success rate is defined as the probability that, when querying SNn, the MN receives back a reply packet from that sensor node. This metric exists for all $n = 1, \ldots, N$. It is empirically calculated as the ratio between the number of reply packets the MN receives from SNn and the total number of queries destined to SNn, provided that the test session duration is long enough.
- $D_{e2e}(n)$: The end-to-end delay is defined as the time elapsed from the instant when the MN sends a query packet to the final recipient SNn and the instant when the MN receives a correspondent reply packet from SNn. Even in this case the definition must be extended to all $n = 1, \ldots, N$. We take many different samples of $D_{e2e}(\text{n})$ by time-stamping via software, at MN side, the two events in correspondence of each successful end-to-end communication. Thus, we are able to estimate the distribution of the random variable $D_{e2e}(\text{n})$.

3 Single Hop Algorithm

Single Hop (SH) is a simplified reference routing scheme and works as follows. When the MN originates a query packet specifying SNn as the final recipient, it also explicitly indicates SN1 as next-hop destination. If receiving the packet, only SN1 forwards it, by keeping SNn as the final destination and expressing SN2 as next-hop recipient. On the contrary, any other SN receiving the packet simply discards it. Generalizing the procedure, SNi forwards the packet to SN$i+1$, the only node authorized to iterate the process, while ..., SN$i-1$, SN$i+2$, ..., if receiving, discard the packet. The functioning is symmetrical for the reply packet originated by SNn and addressed to the MN: when forwarding the packet, SNi indicates SN$i-1$ as next-hop destination while other possibly receiving nodes would discard it. No packet retransmission procedure is supported to recover from packet losses.

4 Limited Flooder Algorithm

The *Limited Flooder* (LF) algorithm is a simplified one-dimensional version of the Directed Flood-Routing approach [10], based on the simple idea of flooding the network with multiple copies of the same packet in order to increase the probability that it could successfully reach the final recipient. However, it is well known that the basic packet flooding approach cannot be sustained in a large network being responsible for exponential growth of the number of packets overall transmitted.

The LF algorithm partially reduces the actual number of packet copies in two ways. First of all, when receiving a packet whose final destination is different from itself, a node forwards it in broadcast mode only if it is the first time that packet is received. This is made possible by reserving a field of the header packet for reporting a packet sequence number.

In addition, the correlation between node address and relative physical node location in the network is exploited by letting a node forward the packet only if its position is closer to the final destination than the last relay node. To this aim, for any p and q, the distance between SNp and SNq is calculated as $|p-q|$, the absolute value of the difference of nodes addresses. Note that backward jumps may occur: for instance, at least theoretically, the path followed by a query packet addressed to SN7 could result as ... –4–9–7 since SN9 is closer to SN7 than SN4 is. In fact, backward jumps have been regularly observed during real test sessions, even if limited to one-hop neighbors of the final destination. These events randomly happen and they depend on nodes hardware efficiency.

Definitively, when receiving from SNi a packet originated by the MN, addressed to SNn and identified by sequence number k, SNj forwards it in broadcast mode only if $|n–j| > |n–i|$ and SNj has never received the same packet before. In order to avoid the possibility of backward jumps, an alternative forwarding rule could be proposed as follows: SNj forwards a packet received by SNi and destined to SNn if $i < j < n$.

Nonetheless, the number of packet copies generated (and possibly received, due to the broadcast mode) can still remain very large and, moreover, it is likely that several nodes try to access the shared medium almost at the same time to transmit a packet. This means that collisions could occur more frequently and that every node, endowed with poor resources, could be overloaded.

To conclude, it is worth noting that, even if no packet retransmission procedure is foreseen, the final recipient could receive multiple copies of the same packet and that any duplicate is anyway discarded.

5 Hopefully Longest Jump First Algorithm

The *Hopefully Longest Jump First* (HLJF) is a novel algorithm that tries to take the most desirable features of SH and LF algorithms, while at the same time overcoming their expected weaknesses. In particular, it pursues two objectives. On the one hand, it aims to achieve more reliable communication and to support network scalability in terms of packet delivery success making use of acknowledgments on a link-basis. Clearly, this is obtained to the detriment of longer average end-to-end delays. On the other hand, HLJF algorithm aspires to minimize the number of intermediate forwarders along the source-destination path by forwarding a packet as close to the destination as possible. This forethought could have beneficial effect for reducing delays.

According to HLJF algorithm, when SNi receives a query packet addressed to SNn, $n > i$, SNi forwards the packet to its current *Farthest Reliable Neighbor Node* (FRNN). The FRNN can be any SN whose address is larger than the current forwarder (i.e., concerning SNi, its FRNN could be SN$i+1$, SN$i+2, \ldots$). Note that if the FRNN address exceeds n, the packet is directly transmitted to SNn.

The core of the protocol resides in the procedure for updating the node FRNN. Let SN$i+k$ be the current FRNN of SNi for some $k > 0$. Then, SNi forwards the query packet to SN$i + k$ which, in its turn, confirms the correct packet reception by sending back an acknowledgment (ack, from now on) to SNi. If the ack is received, no FRNN update is made and the forwarding procedure is shifted on SN$i + k$. On the contrary, if either the query packet is not received by SN$i + k$ or SNi does not receive back the ack, after a Time-Out SNi sets SN$i + k - 1$ as its current FRNN and retransmits (only once) the packet. This couple of actions, FRNN reduction and packet retransmission, is potentially protracted until SN$i + 1$ becomes the current FRNN of SNi. In case not even SN$i + 1$ turns out to be reliable, the path is interrupted and the packet is definitively lost.

So far, FRNN refreshments have only concerned downgrading operations, which always proceed with unitary decrements. Obviously, increments, which are not necessarily unitary, are concerned too. The rule is quite simple and exploits the shared nature of the radio channel. Each time SNi receives a packet (indifferently query packets, reply packets and acks), regardless of SNi being or not the real packet destination, it checks whether the packet sender (not the original source) address is larger than the current FRNN and, if that's the case, the FRNN is consequently updated.

Actually, SNi keeps the addresses of two distinct FRNNs: beside the one described above, a similar one is referred when forwarding reply packets. Since reply packets flow in the opposite direction, from a certain SN to the MN, the address of the related FRNN is always smaller than the current forwarder. Nonetheless, note that reply packets and related acks influence the updating of both the FRNNs, the one referred for reply packets forwarding as well the one concerning query packets. In the following paragraphs, we focus our attention on the FRNN regarding query packets, however extensions are immediate when considering the twin FRNN.

The behavior in two particular circumstances needs to be specified in order to accomplish the protocol description. The former situation concerns the initialization phase, when SNi completely ignores how reliable the direct communication with its neighbor nodes is. The latter happens when query packet forwarding to SN$i+1$, the current FRNN of SNi, fails. When indifferently one of these circumstances occurs, SNi conventionally sets itself as FRNN. Finally, broadcast mode is selected as temporary forwarding strategy when SNi downgraded its FRNN to itself. In other words, from the relay node perspective, HLJF and LF algorithms coincide in such circumstances.

Setting the Time-Out is tricky: it should be large enough to allow the reception of acks avoiding useless retransmissions; however, too large values would result in high end-to-end latencies and overall protocol inefficiency. We experimentally measured the typical distribution of a single-link RTT and observed that the average RTT was about 20 ms, but at least 40 ms was necessary to include around 95% of samples.

To conclude, note that, differently from SH algorithm, multiple paths joining source and destination could be built at the same time. In fact, this could happen because HLJF trivially reduces to LF under certain particular conditions.

6 Experimental Results

The three protocols detailed in Sects. 3, 4, 5, along with the test application described in Sect.2.2 have been implemented on Telos motes using TinyOS development tool. In order to evaluate the effectiveness of the three routing schemes, a linear network has been deployed inside our Lab. The WSN consists of 10 SNs and 1 MN, connected to a pc where data is stored.

The distance between each couple of adjacent nodes in the test-bed is homogeneous and in the order of about 30 cm. This value was chosen after performing several preliminary tests in order to empirically characterize the typical node radio coverage area when the node transmission power was set to the minimum available value, -25 dBm. The goal was to induce with good approximation a scenario where, as schematically depicted in Fig. 2, SNi is:

- (Almost) always able to directly communicate with SN$i-1$ and SN$i+1$
- Often able to directly communicate with SN$i-2$ and SN$i+2$
- Rarely able to directly communicate with SN$i-3$ and SN$i+3$
- Never able to directly communicate with farther nodes than SN$i-3$ and SN$i+3$

For each proposed routing algorithm, test sessions have run along multiple consecutive days, including both working and weekend days. In the case of HLJF algorithm,

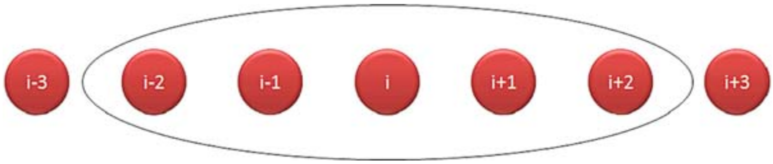

Fig. 2 Expected node radio coverage for the indoor WSN deployment

the Time-Out has been set equal to 50 ms. In the following section, we report and discuss a selected set of trials, concerning a "night" and a "day" scenarios, which have taken place during an ordinary working day. More precisely, the "day scenario" covers the 10-h span of time from 9 in the morning to 7 in the afternoon, i.e., that interval when most of the people are working at their desks and commercial radio devices as wireless networks are likely switched on. The "night scenario" is complementary to the previous one, lasts from 19 pm to 9 am of the day after thus having a duration of 14 h.

6.1 Packet Delivery Success Rate

Figure 3 shows the end-to-end information delivery success rate as a function of the target node address SNi, for the "night" and the "day" scenarios respectively.

The first observation regards a substantial difference of results obtained in the "night scenario" with respect to the "day scenario." This is likely due to a noisier channel available when people move around the test field and, above all, when other wireless technologies operating in the same 2.4 GHz ISM frequency band, such as IEEE 802.11 networks and Bluetooth, are active at the same time. In fact, other test sessions, not reported here, performed during weekend days in the absence of humans and with WiFi APs disabled, have not at all exhibited the remarkable performance heterogeneity from day to night.

Nonetheless, the performance degradation experienced by HLJF algorithm in the two scenarios is much more limited than in the case of both SH and LF schemes. This is achieved thanks to the (though single) packet retransmission procedure supported by the protocol in case of Time-Out expiration: it allows the success rate for the worse node, SN8, to be kept around 80%. The performance degradation suffered by SH is still more evident because, differently from LF, only one end-to-end path is admissible and, in addition, it always involves a larger number of node-to-node communications. The behavior of the SH protocol, mainly the deterministic end-to-end path, makes the interpretation of its results simpler and theoretically more easily predictable. In particular, an exponential decaying of delivery success rate as a function of the distance between MN and target SN is expected. Approximately, this is what really appears in Fig. 3. The LF algorithm exhibits a more irregular trend than

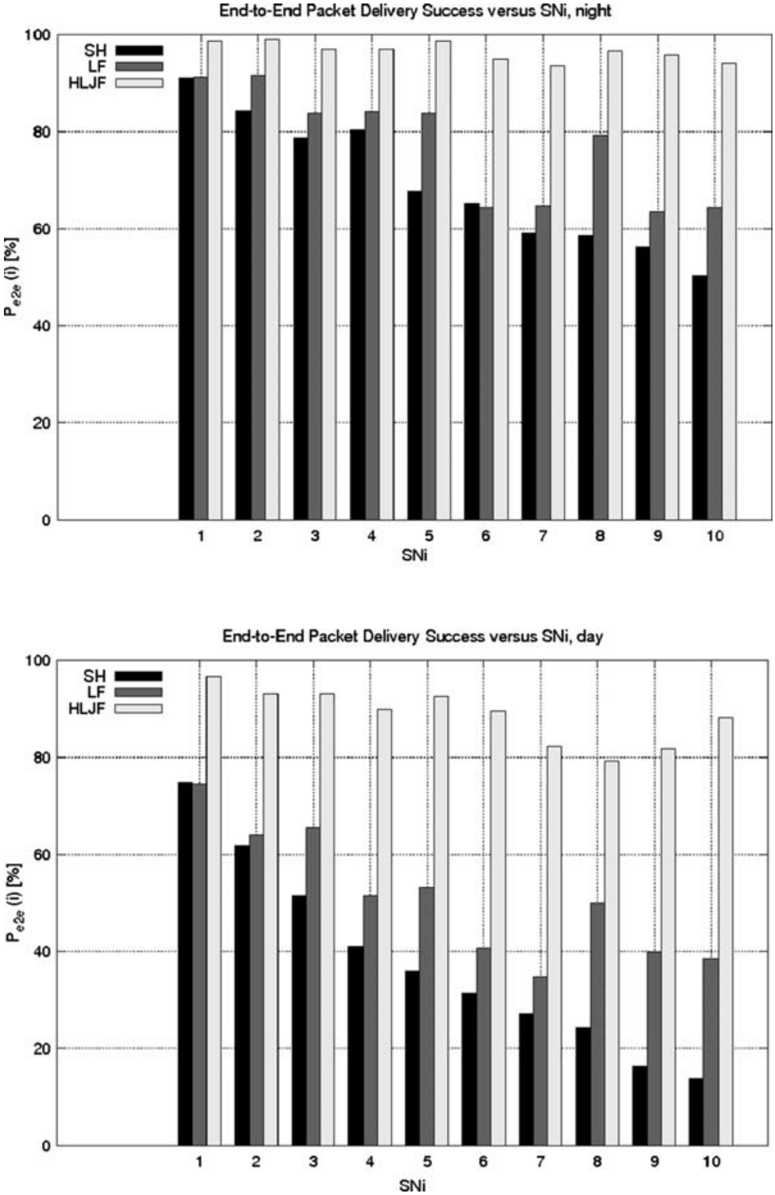

Fig. 3 Performance comparison of routing schemes from the viewpoint of the end-to-end communication success rate, according to "night scenario" (on the top) and "day scenario" (on the bottom)

the other two routing schemes in both the scenarios, and this is coherent with its functioning that leaves open several possible parallel paths and, as a consequence, many different destinies to an end-to-end communication packet. Definitively, the HLJF algorithm outperforms SH and LF in terms of communication reliability.

6.2 Delays

Average end-to-end delays are plotted in Fig. 4, which refers to the "day scenario." The analogous picture for the "night scenario" is not reported since they are qualitatively very similar. The only significant difference concerns the HLJF algorithm that shows a bit larger delays in the less favorable "day scenario." To provide further details about delays, Tables 1 and 2 respectively report minimum and maximum delays in the two scenarios.

The curve of SH average delays is perfectly linear as could be imagined considering how this protocol works. The slope of the curve is about 20 ms/node; thus,

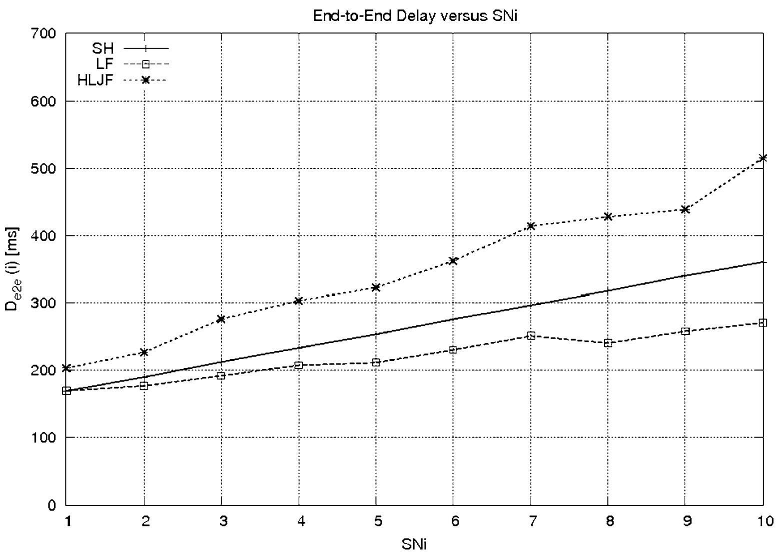

Fig. 4 Average end-to-end delays for "day scenario." Qualitatively similar results hold for the "night scenario"

Table 1 Minimum end-to-end delays, in ms

	SH		LF		HLJF	
	Day	Night	Day	Night	Day	Night
SN1	164.28	164.73	164.46	164.55	186.10	185.70
SN2	182.37	183.11	164.61	164.70	186.07	185.82
SN3	203.31	199.89	176.61	176.64	188.05	189.61
SN4	220.37	218.72	186.83	182.98	189.48	212.59
SN5	237.98	242.31	186.61	183.41	227.51	224.79
SN6	261.17	259.31	208.53	205.29	228.82	237.21
SN7	280.73	279.75	224.33	199.83	269.90	273.71
SN8	303.47	301.51	205.38	205.38	273.93	275.48
SN9	322.97	319.70	229.52	224.79	269.62	293.70
SN10	345.58	342.53	241.00	230.41	314.73	344.51

Table 2 Maximum end-to-end delays, in ms

	SH		LF		HLJF	
	Day	Night	Day	Night	Day	Night
SN1	186.98	178.62	188.11	188.57	338.68	289.92
SN2	202.67	205.51	201.93	195.98	411.35	333.01
SN3	229.86	228.67	240.05	219.06	422.06	452.24
SN4	262.63	252.11	231.69	226.87	669.34	525.48
SN5	287.02	271.42	245.42	238.83	590.85	463.62
SN6	297.67	295.75	272.86	262.05	653.23	551.97
SN7	320.37	316.86	298.43	284.12	686.86	577.39
SN8	347.93	345.83	292.82	275.09	786.41	555.15
SN9	359.34	369.75	311.61	307.46	702.39	701.63
SN10	376.07	409.52	306.82	312.81	970.46	751.46

the cost of each hop is in the order of 10 ms (recall that querying the node SNi according to SH algorithm requires $2 \cdot i$ hops). The conclusion is the same even when considering minimum or maximum delay samples.

Overall, the LF protocol is the one with the best performance in terms of delays: this can be explained by observing that this approach joins the twofold advantage of not managing acks and packet retransmissions while, at the same time often selecting the path with a number of hops as small as possible.

The performance of HLJF routing scheme is the worst on the average since the support for packet retransmissions after Time-Out expirations causes a large variance in experienced delays. Looking at Fig. 4, we can see that its trend in mean delays growth is higher when compared to SH, while the one of LF is the smallest. Coherently, the increase of HLJF maximum delays is the most emphasized too.

On the other hand, HLJF shows minimum delays that are smaller than the ones obtained with SH. This is not surprising because in case of no retransmissions (the actual source of delays), HLJF is able to reduce delays thanks to long jumps as LF.

Definitively, the LF algorithm is preferable in terms of end-to-end delays, but, depending on the application requirements, even a modified version of HLJF could be competitive and adapt to application constraints.

7 Conclusions

In this paper, we have proposed three different routing protocols designed for linear WSNs. Their performance in terms of end-to-end communication reliability and delays have been studied based on results coming from an indoor WSN composed on 11 nodes.

The performance evaluation of a larger experimental network, with up to 30 nodes, is the next step in order to investigate protocol scalability from an experimental perspective. In parallel, the development of some mathematical model would allow to reciprocally validate experimental and theoretical results.

Acknowledgments The research leading to the results presented in this paper has been carried out within the SAFESPOT Project [9], funded by the European Community's Sixth Framework Programme FP6.

References

1. Franceschinis M, Gioanola L, Messere M, Tomasi R, Spirito MA, Civera P (2009) Wireless sensor networks for intelligent transportation systems. 2009 IEEE 69th vehicular technology conference: VTC2009-Spring, Barcelona, Spain, 26–29 Apr 2009
2. Haridas H (2006) BriMon: design and implementation of railway bridge monitoring application. Master Thesis, Indian Institute of Technology, Kanpur
3. Barboni L, Valle M (2008) Wireless sensor networks power aware deployment. Sensor technologies and applications, 2008. SENSORCOMM '08. Second international conference on, 25–31 Aug 2008, pp 252–257
4. Hellman K, Colagrosso M (2006) Investigating a wireless sensor network optimal lifetime solution for linear topologies. J Interconn Netw 7:91–99
5. Gibson J, Xie GG, Xiao Y. (2007) Performance limits of fair-access in sensor networks with linear and selected grid topologies. Global telecommunications conference, 2007. IEEE GLOBECOM '07, 26–30 Nov 2007, pp 688–693
6. Karveli T, Voulgaris K, Ghavami M, Aghvami AH (2008) A collision-free scheduling scheme for sensor networks arranged in linear topologies and using directional antennas. Sensor technologies and applications, 2008. SENSORCOMM '08. Second international conference on, 25–31 Aug 2008, pp 18–22
7. Wang H, Yang Y, Ma M, Wu D (2007) Network lifetime optimization by duality approach for single-source and single-sink topology in wireless sensor networks. Wireless and optical communications networks, 2007. WOCN '07. IFIP international conference on, 2–4 July 2007, pp 1–7
8. Koutsakis P, Papadakis H (2006) Efficient medium access control for wireless sensor networks. Wireless pervasive computing, 2006 1st international symposium on, 16–18 Jan 2006
9. SAFESPOT Integrated Project. http://www.safespot-eu.org/
10. Maroti M (2004) Directed flood-routing framework for wireless sensor networks. Proceedings of the 5th ACM/IFIP/USENIX international conference on middleware, Toronto, Canada, 2004, pp 99–114