

Achieving efficient routing in constrained networks with unidirectional links through Software Defined Networking

Renan C. A. Alves, Cíntia B. Margi

Escola Politécnica – Universidade de São Paulo (USP)
São Paulo – SP – Brasil

***Abstract.** Routing protocols targeting low power wireless networks commonly assume they operate over bidirectional links, ignoring the possibility of routing over unidirectional links. However, experiments show this is not always the case since device heterogeneity and environmental factors can lead to asymmetric links. Our main contribution consists on the design of algorithms that are capable of using unidirectional links to route packets. We took on a Software Defined Networking approach, as its centralized feature allows for a generic and efficient solution to the challenge of routing over unidirectional links. We conducted experiments to investigate the trade-offs between fully using all available links in the network and the additional overhead required in maintaining unidirectional routes. The gist of our findings is that unidirectional links must be considered during protocol design, while actually leveraging such links for routing tends to improve network metrics, especially if the links are long-reaching.*

1. Introduction

Low power and Lossy Networks (LLNs) is an umbrella term referring to wireless networks operating on a power budget while employing specialized link layer standards, such as IEEE 802.15.4 or Bluetooth low energy (BLE). An instance of LLNs are Wireless Sensor Networks (WSNs). The main application of such networks is to gather data from the environment through the deployment of small, cheap sensing devices. Additionally, LLNs are expected to play a role in the realization of the Internet of Things (IoT), supporting the relevance of the topic at hand.

The low power aspect of the communication standards used in these networks, along with the typical affordability of their hardware components, facilitate the occurrence of spontaneous asymmetric and unidirectional links. In fact, we set up a pair of nodes to send packets to each other at varying distances, positions, and power levels [Alves and Margi 2020]. The experiments confirmed that, even in homogeneous circumstances, the asymmetry level (as defined in the paper) can be typically as high as 0.75, while rising to 0.91 when power asymmetries are introduced.

Besides naturally emerging asymmetric links, links can be made purposefully unidirectional – for example, to explore specific devices capabilities, or to supply deployment needs. A given node could use an energy surplus to increase its transmission power, sparing the energy of intermediate nodes, which do not abound energy as much. Another example, is the case of a basestation, typically connected to utility power. This node often carries the burden of configuring other network nodes; a long-reaching unidirectional link can be employed to carry out this task in a fast and energy efficient manner.

Having established why unidirectional routing is a worthwhile challenge, let us review how previous routing techniques dealt with it. We may classify preexistent techniques into two distinct groups: (1) algorithms that are restricted to specific network configurations, thus ones that are non-generic, and (2) algorithms that employ flooding, thus incurring in prohibitive overheads.

The main representatives of the non-generic algorithms category are Chen *et al.* [Chen et al. 2013] and Kim *et al.* [Kim et al. 2016]. The former proposes a probabilistic routing algorithm targeting many-to-one traffic only. The latter assumes a network with a single sink able to reach any node in the network in a single hop, although most nodes cannot reply back to the sink. Their premise is difficult to generalize to a case where any link could be unidirectional and any traffic pattern can be supported.

On the other hand, there are protocols that are generic in that they may support any kind of unidirectional link and topology, but are not efficient in doing so, i.e. they present high overhead. At the time of writing, these two such protocols are BRA [Ramasubramanian and Mosse 2008] and ULC [Karnapke and Nolte 2015]. Both protocols operate similarly in spite of focusing on different layers of the protocol stack: they rely on flooding to seek the reverse path of any unidirectional links in the network.

Given the research challenge of routing over unidirectional links and after surveying the existent body of literature, we identify the research gap: the lack of a generic and efficient routing algorithm that supports unidirectional links.

We resolve this research gap by using the Software Defined Networking (SDN) paradigm. SDN has debuted in wired networks to increase their flexibility and configurability, reducing operational costs and error-prone manual configuration. It has since been ported to LLNs seeking the same advantages over traditional algorithms, while keeping a low overhead profile, comparable to existing distributed algorithms [Alves et al. 2019b]. SDN's distinctive feature is its separation of data and control planes, requiring specialized auxiliary algorithms: neighbor and controller discovery protocols. However, previous discovery protocols do not tackle link asymmetry.

Our contribution is the adaptation of the SDN paradigm to support unidirectional links, along with a set of auxiliary protocols to enable such to be done. More precisely, we have developed four novel assets: (1) a set of experiments regarding link asymmetry, including a unique metric to assess asymmetry level [Alves and Margi 2020]; (2) simple proof-of-concept discovery algorithms [Alves and Margi 2017]; (3) enhanced discovery algorithms, tailored for performance [Alves et al. 2019a]; and (4) an asynchronous radio duty cycling (RDC) algorithm, the first of its kind to support unidirectional links [Alves et al. 2020]. A summary of these contributions is presented in Section 2.

We evaluated the performance of the algorithms in a variety of scenarios. We observed that, without duty cycling, using unidirectional links yields at least 90% packet delivery, whereas the baseline case yielded less than 50% delivery on many cases. On the other hand, the RDC analysis showed that using unidirectional links is beneficial mostly with long-reaching links. We discuss our main results in Section 3. Final remarks are presented in Section 4.

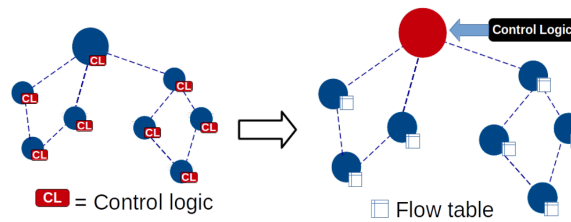


Figure 1. Classic routing vs. Software Defined Networking

2. Contributions Summary

The Software Defined Networking paradigm's key characteristic is its separation of the control plane and data plane. Network devices forward packets according to the contents of a flow table, which inform what action to take by matching specific packet bits. Flow tables are filled by the control plane, which is located at logically centralized controller nodes, responsible for making network decisions. Figure 1 illustrates the differences between the classic routing approach and the SDN approach.

In the context of low power networks, the control plane programs the data plane through an inband channel. In fact, nodes far from the controller need to run a controller discovery algorithm, in order to contact the control plane to request flow table entries. To that end, the control plane, itself, needs topological information to make accurate decisions. This information is gathered by all nodes via a neighbor discovery algorithm. Such an algorithm detects which nodes are directly connected to one another, it may also be responsible for link quality assessment.

Once the controller builds an accurate network representation, calculation routing with unidirectional links may be easily solved with a local algorithm. The challenge is to provide controller and neighbor discovery algorithms that are able to both cope with, and correctly detect unidirectional links.

We adopted simple but effective strategies in designing our algorithms. The neighbor discovery algorithms focus on finding the inbound neighbors, instead of the typical outbound neighbors, as in most discovery algorithms. Concerning controller discovery algorithms, we explored the fact that they are executed mostly during the setup phase.

First, we designed simple proof-of-concept algorithms to demonstrate the soundness of our hypothesis (Section 2.1). Considering the nascent design a success, enhanced algorithm were then designed to target performance efficiency (Section 2.2).

Certainly, a solution targeting low power networks would be incomplete without incorporating a radio duty cycling algorithm. Noting an absence of asynchronous RDCs able to cope with unidirectional links, we endeavored to design our own (Section 2.3).

2.1. Simple algorithms

The goal of the simple algorithms is to show SDN a viable solution to the unidirectional links problem. In the simple neighbor discovery algorithm, every node announces its existence by transmitting periodic beacons. When another node receives a beacon, the sender is added to the inbound neighbor table. Eventually, the complete inbound neighbor table is transmitted to the controller.

In turn, the simple controller discovery algorithm is built on a modified Bellman-Ford algorithm. In short, the controller starts the algorithm by advertising its inbound neighbors. The next steps occur iteratively: each node that discovers how to reach the controller continues to advertise its neighbors, until all nodes eventually converge to a path towards the controller.

Simulation results supported the hypothesis that SDN is a viable solution to the research problem, although, as expected, the obtained control overhead could be improved. These findings were published in a national conference [Alves and Margi 2017].

2.2. Enhanced algorithms

The enhanced algorithms were designed to live up to the potential of SDN, while addressing the overhead issue in the simple algorithms discussed previously. The primary neighbor discovery technique remained unchanged: periodic beacon broadcasts. However, we added the following four enhancements: (1) discovery by overhearing, (2) adaptive beacon intervals, (3) link quality estimation, and (4) neighbor departure detection.

The enhanced controller discovery algorithm was completely redesigned. The main premise was the routes found by the controller discovery process are temporary and used only during the initial network setup. Thus, the adopted strategy was to blacklist unidirectional links only in this early stage. A tree rooted at the controller is built, enabling nodes to quickly obtain a route to the controller.

Our experiments indicated that leveraging unidirectional links, if any, positively affects packet delivery rate and tends to decrease the average data delay. This set of algorithms and experiments was published in an international conference [Alves et al. 2019a].

2.3. Radio duty cycling

Our final contribution was a radio duty cycling algorithm that supports unidirectional links. We focused on asynchronous RDCs, since, in comparison to synchronous ones, they do not require the precomputation of a transmission schedule prior to utilization. In addition, to the best of our knowledge, there were no substantive efforts to provide unidirectional support to asynchronous RDCs in an efficient fashion.

Preamble sampling is a classic asynchronous RDC technique; all nodes wake up periodically to check for transmission, while a sender transmits a preamble to alert the receiver of an incoming packet.

ContikiMAC [Dunkels 2011] is a state-of-the-art RDC, but heavily relies on acknowledgments to implement its features: early acknowledgment, and delayed preamble transmission. We modified the packet headers to include enough information to relinquish acknowledgments. The receiver uses this information to calculate the phase shift, which tells the sender when it should start and end the preamble transmission. The phase shift is relayed to the sender through a route calculated by the controller.

The results illustrate that unidirectional links improve network metrics primarily when the unidirectional links are long; the greatest improvement being seen in the metric of packet delay. However, minor data delivery gains were also observed in cases of short-ranged unidirectional links in medium-sized networks. The asynchronous unidirectional RDC was published within the Elsevier journal of Computer Communications [Alves et al. 2020].

3. Main results

In this section, we summarize the experiments executed during the research span. We used COOJA, a tool that runs native code by emulating WSN motes, while simulating the wireless medium. The full set of results is available in the thesis document [Alves 2020]. The experiments were designed to answer the question of whether using unidirectional links is valuable from a network standpoint.

We split the experiments in two sets: without RDC (pure CSMA), and with RDC. In the first group we tested our simple and enhanced algorithms by comparing them to a baseline approach known as “Collect”. In the latter group we tested the same algorithms, combining them with our RDC, when unidirectional links are used; otherwise, Contiki-MAC is used. None of the non-SDN asymmetric algorithms we have surveyed had an implementation we could use to compare our results.

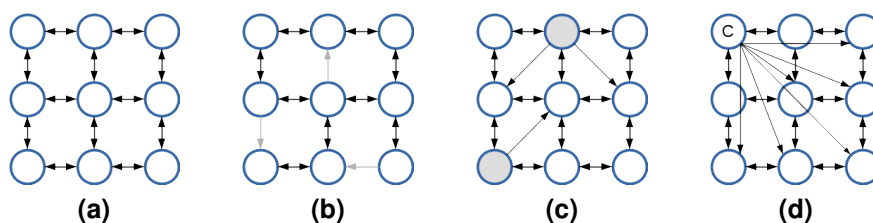


Figure 2. Link variations used in experiments. (a) Bidirectional only. (b) Random unidirectional links. (c) Nodes with increased range. (d) Controller to all.

Each algorithm was run under a combination of the following three overarching parameters: (1) network size: from 16 to 100 nodes; (2) node placement: random or grid; and (3) link configuration: fully bidirectional, random unidirectional links, nodes with extended range, or the controller with a link to every node in the network (Figure 2 exemplifies). The graphs are organized as follows: there is a plot for each link configuration, while the lines within the plots represent a combination of algorithm (colors) and topology (dotted for grid, continuous for random). Horizontal axis represents number of nodes, while vertical axis represents the metric at hand.

Each of the test scenarios was simulated 10 times for 60 minutes, obtaining statistical relevance. The assessed metrics were typical wireless network indicators, namely packet delivery rate, data delay, control overhead, and energy.

Figure 3 displays data delivery results without RDC. In fully bidirectional networks, the enhanced discovery algorithms perform on par with the Collect-based approach (Figure 3a), while the enhanced discovery algorithms deliver more packets when unidirectional links are present and actually used (Figure 3b). The simple ND/CD algorithms presented comparable performance to the enhanced algorithms in small networks (up to 36 nodes). However, their delivery rate decreases in a linear trend as the network size increases, indicating a lack of scalability. The Collect-based algorithm performs poorly in the presence of unidirectional links, regardless of link and topology type.

The best result for the data delay metric is observed in the “controller to all” topology (Figure 4b). Using the enhanced algorithm as a reference, long unidirectional links allow for faster initial route setup, decreasing the overall data delivery delay. However, there is a small performance degradation in comparison to the Collect approach in the fully bidirectional case due to the extra control overhead.

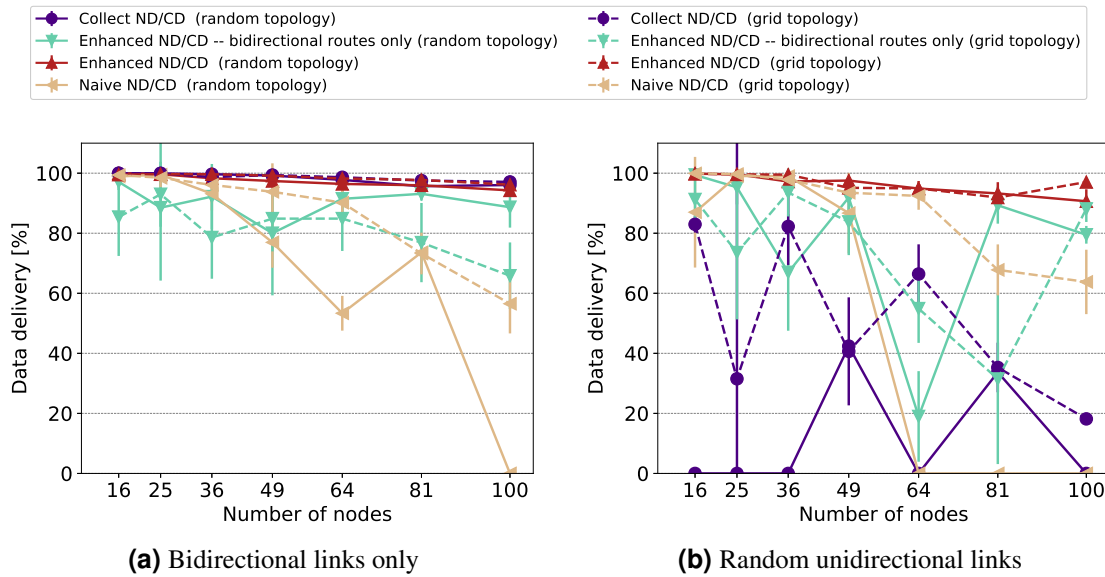


Figure 3. Without RDC (pure CSMA): Data delivery results.

4. Final Remarks

Unidirectional links emerge naturally on low power wireless networks due to radio transceiver characteristics, environmental factors, and device heterogeneity. Neglecting such links may result in performance degradation or protocol failure.

We used the Software Defined Networking paradigm to address the challenge of routing in the presence of unidirectional links. The crucial strategy in our case involved the leveraging of the network centralized view available at the controller to facilitate route calculation.

Our main contribution is a set of discovery algorithms needed to enable SDN to work properly under unidirectional links. Our results show that exploring long-reaching unidirectional links is the most valuable way of exploring unidirectional links. The use of an RDC negates most gains obtained by short unidirectional links, while there are modest gains in the pure CSMA/CA scenarios tested.

4.1. Publications

Four research papers stemmed directly from the PhD research, listed below:

- “Discovery protocols for SDN-based wireless sensor networks with unidirectional links”, presented at the XXXV Brazilian Communications and Signal Processing Symposium (SBrT) [Alves and Margi 2017];
- “No way back? An SDN protocol for directed IoT networks”, presented in the 15th Wireless On-demand Network systems and Services Conference (WONS) [Alves et al. 2019a];
- “Know when to listen: SDN-based protocols for directed IoT networks”, published at the Elsevier Computer Communications Journal [Alves et al. 2020];
- “Can you hear me? A metric for link asymmetry”, published at the Open Journal of Internet Of Things, and presented at the International Workshop on Very Large Internet of Things [Alves and Margi 2020].

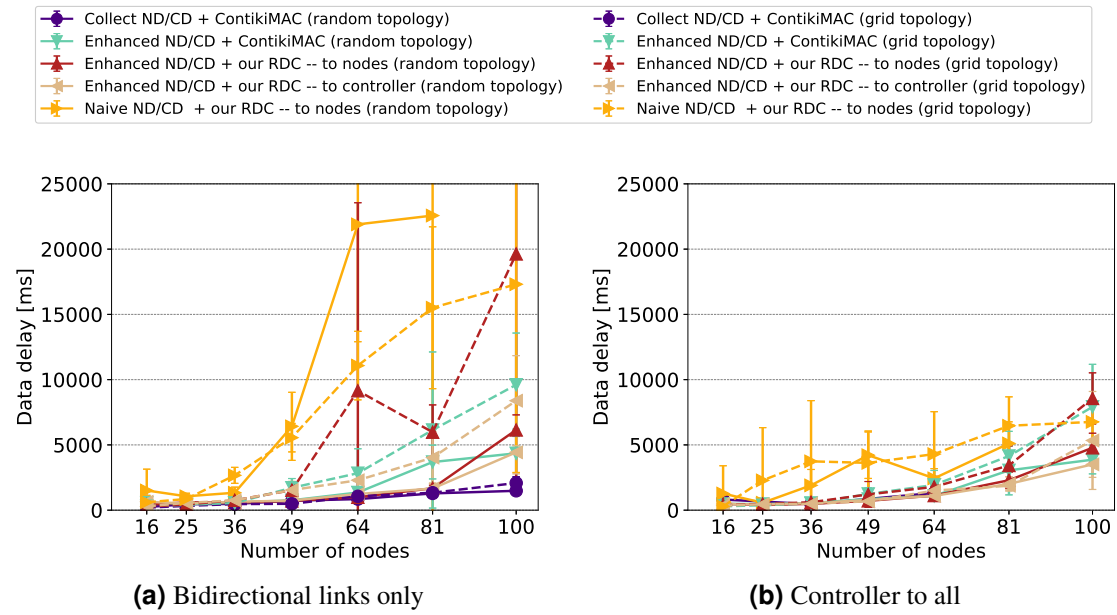


Figure 4. With RDC: Data delay results.

Additionally, three other papers, focused on porting the SDN paradigm to low power wireless networks, which is a vital premise for this research. These paper were published along the PhD timespan in collaboration with research group fellows:

- “IT-SDN: Improved architecture for SDWSN”, presented at the XXXV Brazilian Symposium on Computer Networks and Distributed Systems (SBRC) [Alves et al. 2017];
- “Software-defined wireless sensor networks approach: Southbound protocol and its performance evaluation”, published in the Open Journal of Internet Of Things [Margi et al. 2018];
- “The cost of software-defining things: A scalability study of software-defined sensor networks”, published at the IEEE Access Journal [Alves et al. 2019b].

Lastly, the paper entitled “Cluster-based Flow Control in Hybrid Software-Defined Wireless Sensor Networks” was published in the Elsevier Computer Networks Journal [Liu et al. 2021]. The opportunity to collaborate on this paper arose during the Research Internships Abroad (BEPE – Bolsa Estágio de Pesquisa no Exterior, in Portuguese), under the supervision of Professor Fernando Kuipers. In this paper, we used IT-SDN to provide a proof-of-concept to a system that trades the amount of overhead for the percentage of centrally controlled devices.

Acknowledgments

We would like to thank Professor Fernando A. Kuipers for discussions and opportunities that contributed to the development of this research.

Renan C. A. Alves was supported by grants #2016/21088-1 and #2018/11295-5, São Paulo Research Foundation (FAPESP); and grant n° 155372/2016-5, Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

References

- Alves, R. C. A. (2020). *Achieving efficient routing in constrained networks with unidirectional links through Software Defined Networking*. PhD thesis, Escola Politécnica da Universidade de São Paulo, São Paulo, Brazil. Available at: <https://doi.org/10.11606/T.3.2020.tde-17122020-103717>.
- Alves, R. C. A. and Margi, C. B. (2017). Discovery protocols for SDN-based Wireless Sensor Networks with unidirectional links. In *XXXV SBrT*, São Pedro - SP, Brazil. Sociedade Brasileira de Telecomunicações.
- Alves, R. C. A. and Margi, C. B. (2020). Can You Hear Me? A Metric for Link Asymmetry. *Open Journal of Internet Of Things (OJIOT)*, 6(1):82–88.
- Alves, R. C. A., Margi, C. B., and Kuipers, F. A. (2019a). No way back? An SDN protocol for directed IoT networks. In *15th WONS*, Wengen, Switzerland. IEEE.
- Alves, R. C. A., Margi, C. B., and Kuipers, F. A. (2020). Know when to listen: SDN-based protocols for directed IoT networks. *Computer Communications*, 150:672 – 686.
- Alves, R. C. A., Oliveira, D., Segura, G. N., and Margi, C. B. (2017). IT-SDN: Improved architecture for SDWSN. In *XXXV SBRC*, Belém. Sociedade Brasileira de Computação.
- Alves, R. C. A., Oliveira, D. A. G., Segura, G. A. N., and Margi, C. B. (2019b). The cost of software-defining things: A scalability study of software-defined sensor networks. *IEEE Access*, 7:115093–115108.
- Chen, X., Dai, Z., Li, W., and Shi, H. (2013). Performance guaranteed routing protocols for asymmetric sensor networks. *IEEE Transactions on Emerging Topics in Computing*, 1(1):111–120.
- Dunkels, A. (2011). The ContikiMAC Radio Duty Cycling Protocol. Technical Report T2011:13, Swedish Institute of Computer Science.
- Karnapke, R. and Nolte, J. (2015). Unidirectional link counter - a routing protocol for wireless sensor networks with many unidirectional links. In *Ad Hoc Networking Workshop (MED-HOC-NET), 2015 14th Annual Mediterranean*, pages 1–7.
- Kim, H.-S., Lee, M.-S., Choi, Y.-J., Ko, J., and Bahk, S. (2016). Reliable and energy-efficient downward packet delivery in asymmetric transmission power-based networks. *ACM Transaction on Sensor Networks*, 12(4):34:1–34:25.
- Liu, Q., Cheng, L., Alves, R., Ozcelebi, T., Kuipers, F., Xu, G., Lukkien, J., and Chen, S. (2021). Cluster-based flow control in hybrid software-defined wireless sensor networks. *Computer Networks*, 187:107788.
- Margi, C. B., Alves, R. C. A., Segura, G. A. N., and Oliveira, D. A. G. (2018). Software-defined wireless sensor networks approach: Southbound protocol and its performance evaluation. *Open Journal of Internet Of Things (OJIOT)*, 4(1):99–108.
- Ramasubramanian, V. and Mosse, D. (2008). BRA: A Bidirectional Routing Abstraction for asymmetric mobile ad hoc networks. *IEEE/ACM Transactions on Networking*, 16(1):116–129.