Middle East Technical University
Department of Computer Engineering

Implementing and Benchmarking Babel Routing Protocol on SDRs

CENG513 Wireless Communications and Networking
2016-2017 Spring
Term Project Report

Prepared by

Ozge Lule	1881366	luleozge@gmail.com
Esref Ozturk	1881762	esrefozturk93@gmail.com
Ilim Ugur	1746460	e1746460@ceng.metu.edu.tr

Computer Engineering
15 May 2017

Abstract

As in any conventional, wired computer network, a routing protocol is needed for a network of wireless devices to achieve communication. This study considers Babel, a known routing protocol, and attempts to find out if it is practical to use Babel in a wireless network setting formed by Software Defined Radios. (SDR) Performance of the routing protocol is attempted to be measured based on the packet transmission from a source node to a destination node. Packet transmission from a single SDR to another with no routing protocol involved is also separately benchmarked to gain insight into the performance of the specific devices being used. Successful packet transmission and transmission duration are considered as the principal metrics for the performance evaluation in both experiments. The results obtained indicate that the successful packet transmission rate from a device to another is too low, even when a routing protocol is not involved. At this rate, packet loss is overwhelming for Babel protocol, and as expected, it fails to function. Based on the results obtained, it is considered by the researchers that it is impractical to use Babel in a wireless network topology, unless specific countermeasures to remedy the packet loss rate are taken.

Table of Contents

List of Figures

List of Tables

1. Introduction

The goal of this project is to quantitatively assess the performance of Babel network routing protocol in a wireless ad hoc network setting using Software Defined Radios (SDR) and to compare it to the performance of other available routing protocols that can run on SDRs. Although RFC for Babel protocol [1] is not that recent, until the usage of SDRs became widespread, research efforts on wireless networks were not as painless. Routing protocol performance analysis on wireless networks was not an exception, and such a comparative benchmarking for Babel protocol was not extensively studied. Thus, an effort on this front can still yield valuable information to be used when choosing which protocol to use in a particular wireless ad hoc network, especially ones that make use of SDRs.

The project scope consists of testing Babel routing protocol on wireless ad hoc networks running on SDRs, and comparing its performance to other routing protocols running on the same setup. However, to achieve that, there is a clear need to set a quantitative basis for the performance of a routing protocol to be tested. As a result, it is recognized that certain limitations may need to be made regarding the structure of the ad hoc network. Moreover, even though the project may include speculations and projections as to the performance of the protocol over a generic ad hoc wireless network, the quantitative testing may be limited to certain configurations, which differ based on the density or the physical range of the network.

The main challenge of the project lies not in the complexity of the code, which is certainly peculiar in itself, but rather in the specifications, configurations and special arrangements to be made while dealing with a problem on wireless networks. In order to develop a proper quantitative analysis, certain configurations should be enforced while setting up the environment on which the routing protocols will be tested.

2. Background and Related Work

2.1 Background

The project is based on the idea of routing. In the simplest sense, routing is the act of estimating a route, or a path between two parties in a network so that those parties can communicate. The concept of routing lies in the core of the Internet, and in fact all the interconnected networks, as the communication between different parties is the key to transferring information.

There are various routing algorithms specifications of which are based on different features of the network itself, including the topology and the density of it. Among the popular approaches in routing is distance-vector routing protocols, which is based on the idea of routers regularly informing its neighbors as to the changes in their topology. Babel routing protocol, which is the main focus of this project, belongs to this class of solutions to the routing problem.

Babel is a distance-vector routing protocol that can be used on wired and wireless networks. This study reveals experimental results on the feasibility of running Babel routing protocol on SDRs.

Before reviewing this project, one should also review the concept of wireless networking and the challenges it entails to transferring information, as wireless networks are at the center of the project. Contrary to passing the information through a wire, communicating without a wire requires dealing with quite a few serious challenges like handling noise and physical obstructions. As a result, in most cases, studying wireless networks turn out to be much trickier than studying its wired counterparts.

Software Defined Radio (SDR) is a tool helping ease studies on wireless networks, overcoming most of the challenges of a wireless networks by providing a rather generic interface for each participant of a network to use. In recent years, SDRs are being used more and more in wireless networking studies and are a valid option to test the performance of a routing algorithm, as it eliminates the overhead of setting up the physical layer configuration of a wireless network.

2.2 Related Work

Performance analysis of routing networks on various settings are not a new domain for academic research. In fact, even for wireless networks, there are studies dating more than a decade back that focus on routing protocols for ad hoc wireless networks. [2] Although they provide an initial perspective on how a review of routing algorithms can be done, their major drawback is the fact that they do not feature more recent routing algorithms such as Babel.

Naturally, not all the studies on wireless ad hoc network routing predate more modern routing protocols. In fact, some provide a solid ground for quantitative performance benchmarking, despite not featuring Babel. [3] Others benchmark routing protocols including Babel, but define a rather specific use case, such as UAVs. [4]

There even exists a study on routing protocols for wireless ad hoc networks featuring a comparison of protocol options including Babel. [5] However, that too differs from this project by not forming the networks using SDRs.

3. Main Contributions

3.1 SDR Performance Analysis

The field of wireless networking features numerous other studies on routing protocols, and quite a few projects on SDRs. As referenced in Section 2.3 and narrated in Section 2.2, there are specific studies to benchmark Babel protocol and to evaluate its relative performance against other routing protocols. However, considering how relatively recently SDRs became widespread, a quantitative performance analysis of Babel routing protocol on SDRs can be considered as an endeavor valuable to the field.

3.2 Extension of Babel Implementation

As detailed in Section 4.1, the study makes use of the official Babel protocol implementation. In order to achieve implementing Babel protocol for SDRs, the study aims to extend the source code to support SDRs. If the study or the open-sourced code related to it gains enough traction, the processes and the code used in this study may eventually be extended by other developers in the Babel community to cover a wider range of SDRs. Although the code produced is rather a by-product from the perspective of the study, it is a noteworthy contribution to the open source developer community of an important routing protocol, considering the increasing usage and prominence of SDRs.

4. Results and Discussion

4.1 Methodology

This section provides further insight into the specific settings used for the experiments that were run, by clearly covering the devices, the software and the methods used in conducting them.

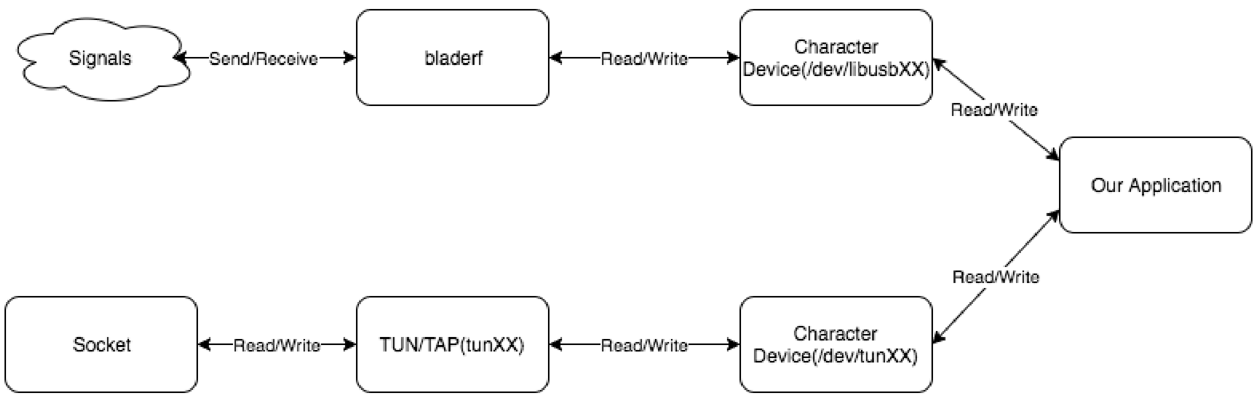
4.1.1 Hardware and Software

The study utilized up to 3 bladeRF x115 devices as SDRs. [6] The software to run on these devices is based on the official sample codes provided by the producer. [7] On top of that boilerplate code, an MIT-licensed open source library called liquid-dsp is utilized to achieve packet-based transmission of data between devices. [8] Furthermore, official 1.8.0 release of Babel protocol daemon is used. [9] Finally, SDRs are connected to computers with operating systems that have a Linux-based operating system, specifically Ubuntu 16.04. [12]

4.1.2 Communication Between Two SDRs

SDR devices constitute the physical layer of the network schema handled in this study. Consequently, how SDRs communicate should be more clearly covered, in order to provide further insight and to make replication of this study possible.

The manner in which this study achieves SDR communication is as follows. Initially, each SDR is connected to a computer, and not essentially to a different one, via a Universal Serial Bus(USB) interface. [13] Then, for each SDR, a network tunnel (TUN) interface is defined, as a virtual kernel network device. The code written as part of this study is placed between the USB and TUN interfaces, and is responsible from transmitting the packets received from one of the interfaces to the other one.



4.1.3 SDR Network Topology

As multiple SDRs are to be used in testing the performance of Babel protocol, how these SDR devices are to be arranged is a question worth considering, given that there are multiple topologies that can be used. For the purposes of the research, given the number of bladeRF devices used, it is deemed reasonable to use centralized messaging topology, where each node of the network can communicate with any other node in a point to point fashion. Once such peer to peer transmission can be achieved, routing protocol is planned to be utilized in settings with 3 nodes, such as placing a third SDR between two separated SDRs.

4.1.4 Benchmarking

The benchmarking method being used in estimating the efficiency of the device communication considers two parameters as significant. The first is the number of packets that were lost during transmission, and the second one is the latency to transmit the packets from one device to another. These parameters may be affected by many factors including distance and interference. Due to having limited resources, among all such possible factors, only distance was considered, and the tests were conducted multiple times, with nodes at different distances between SDRs.

In the test setting, real values for these metrics are obtained by automating the transmission logic using a Python script. [10] In order to benchmark the efficiency of the channel in a simple, straightforward manner, the duplexity of the SDR communication is not utilized. Instead, when running SDR communication tests, one node is asked to transmit, exclusively while another only receives the transmitted packets. Last but not least, to be able to find out an average value of latency and packet loss in the test setting, the script repeats packet transmission multiple times, 1000 to be exact, for each distance values.

As for the performance benchmarking of Babel's routing performance, the planned method to be used is Expected Transmission Count (ETX) which is the one used and endorsed by the original Babel RFC for wireless networks. [1] This metric is introduced in a 2003 study and the original paper is used as guide in implementing the benchmarking to be used in this study. [11]

4.2 Results

Current results of the project include 2 sets of packet transmission logs and their aggregation. Each dataset contains 4 different tests that were conducted by placing two SDRs 30, 60, 90 and 120 centimeters apart. The remainder of the test settings are identical to the content covered in Section 4.1.

A summary of the test data featuring the average duration of a packet to be received can be viewed in Table 1, shown below. The entire SDR performance test data can be found online. [10]

Distance(cm)	Packets sent	Packets received	Average Duration(seconds)
30	1000	29	0.149299
60	1000	20	0.136568
90	1000	15	0.138780
120	1000	14	0.137731

Table 1: Summary of dataset 1 [14]

Distance(cm)	Packets sent	Packets received	Average Duration(seconds)
30	1000	17	0.138616
60	1000	18	0.140821
90	1000	32	0.141074
120	1000	23	0.141280

Table 2: Summary of dataset 2 [15]

Upon setting up Babel protocol daemon, and attempting to transmit a packet from one of the SDRs to the other one, no packets were received. The distance between the SDRs was changed and the experiment was attempted twice more, to no avail. Consequently, Babel performance benchmarking could not be conducted. Possible reasons for the failure of the experiments with Babel protocol are considered in Section 4.3.

4.3 Discussion

There are two main discussions to be made regarding the results obtained and are presented in Section 4.2. The first one considers the results indicated by the datasets representing the packet transmission between SDRs, while the second one is relevant to the failure of running Babel routing protocol using SDRs.

4.3.1 Packet Transmission Amount

The results obtained in two datasets are compelling due to numerous reasons. The first dataset shows that the number of received packets decrease as the distance between

SDRs is increased. Even though there seems to be some conflicting data in the second dataset (i.e. number of packets received for the distances of 30, 60, and 90 centimeters) given that the number of packets received for 120 centimeters is less than the number obtained for 90 centimeters, the results can be considered to, in general, comply with the idea that factors such as interference in wireless networks cause successful transmission rate to decrease as the distance between the endpoints increases.

4.3.2 Packet Loss Rate

Arguably the most significant result to be considered is the amount of packets that were successfully transmitted. Even though the datasets differ in certain ways, they decisively agree on the fact that only a fraction (i.e. less than 4% in the best setting) of the packets sent by an SDR were received by the other one.

There may be a few reasons for a loss rate as high as the ones observed in the datasets provided above. The first reason has to do with the nature of wireless networking, which features dealing with concepts such as interference, reflection, and distortion. Even an environment with sharp corners or many objects can be considered a significant challenge for wireless transmission. Wireless networking solutions typically face these type of challenges by utilizing countermeasures such as repetition of packets. Yet, even though the packets sent could have been repeated, this level of packet loss was not predicted for a setting that does not feature repetition of packets, and the rate constitutes a compelling concern for the practical use of SDRs.

The high loss rate could have also occurred due to how a successful transmission was defined. The setting narrated in Section 4.1 details the usage of Python scripts in benchmarking. Due to said scripts using the default sockets package of Python programming language, no specific error correction is applied to the packets, and modified packets are just omitted, despite they have been, at least partially, transmitted. In other words, all the packets that are modified while in transit are considered unsuccessful. If a special mechanism is used to handle modifications to transmitted packets, it may be possible to recover the original content of some, and consequently achieve lower rates of loss. In the current setting, however, no modified packet is considered successfully transmitted, despite the fact that it was successfully received.

4.3.3 Packet Transmission Duration

The average time it took to transmit the successfully received packets is another result obtained as part of the packet transmission experiment. While the average transmission time increases with the increasing distance in dataset 2, dataset 1 features results which are not clear, such as the average transmission time of SDRs that are 30 centimeters apart being too long or the the average transmission time of SDRs that are 90 centimeters apart being too short. Still, the two datasets can not be considered to completely disagree, and relative discrepancies within dataset 2 can be attributed to environmental factors which can also cause packet loss, and are also mentioned above, in Section 4.3.2.

4.3.4 Babel Transmission Issues

The reason Babel routing protocol did not successfully work with SDRs is not an easy one to find. Nevertheless, the packet transmission experiment discussed above can be considered as a guide in understanding SDRs, and with some help from the RFC of the Babel routing protocol, it is possible to analyze the failure.

As detailed in Section 3 of its RFC, Babel protocol operates by sending numerous packets among the nodes included in the network topology. [1] Therefore, the phenomenon of packet loss can be detrimental to the protocol, as it does not specifically cover packet loss. It is not only possible, but validly proven in Section 4.2 that packet loss rate is too high for the bladeRF devices that were used in this study. Consequently, the principle cause of the failure of Babel protocol running on SDRs is not an error in estimating a route for the packet, but the packets being lost while in transit from an SDR to another.

As wireless network challenges are not a priority for a routing protocol, they are not considered or sufficiently handled by babeld, the official implementation of the protocol. [9] Until the daemon is extended to handle such challenges that occur in wireless networking, or those challenges are handled by application-specific means by the experimenters, it is improper and impractical to attempt using Babel for SDRs, and for wireless devices, generally.

5. Conclusion

Widespread use of the wireless network may lead one to believe that it is possible to build a complex mesh of wireless networks and achieve successful communication between the nodes involved. Despite the initial conjecture, however, the results of the experiments indicate that at below 4%, the packet loss rate is too high to deem using Babel, or any other routing protocol as practical, at least until steps are taken to ensure that the packets can successfully be transmitted. Remedying the packet loss rate, or updating the routing protocols to handle packet losses is the most critical future work that can be conducted in this front to achieve truly widespread use of wireless networks.

Yet, even if the packet loss is remedied, the average transmission durations featured among the results are not optimistic, as it takes considerably long to transmit a packet. Therefore, the researchers conclude that presently, it is not feasible or practical to construct complex wireless networks of many nodes if transmission efficiency is a high priority.

6. References

[1] “RFC 6126 - The Babel Routing Protocol,” RFC 6126 - The Babel Routing Protocol, Apr-2011. [Online]. Available: https://tools.ietf.org/html/rfc6126. [Accessed: 13-May-2017].

[2] E. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Personal Communications*, vol. 6, no. 2, pp. 46–55, 1999.

[3] S. Mohseni, R. Hassan, A. Patel, and R. Razali, “Comparative review study of reactive and proactive routing protocols in MANETs,” 4th IEEE International Conference on Digital Ecosystems and Technologies, 2010.

[4] S. Rosati, K. Kruzelecki, L. Traynard, and B. Rimoldi, “Speed-aware routing for UAV ad-hoc networks,” 2013 IEEE Globecom Workshops (GC Wkshps), 2013.

[5] D. Murray, M. Dixon, and T. Koziniec, “An experimental comparison of routing protocols in multi hop ad hoc networks,” 2010 Australasian Telecommunication Networks and Applications Conference, 2010.

[6] Nuand LLC, “bladeRF x115 Product Specification.” [Online]. Available: https://www.nuand.com/blog/product/bladerf-x115/. [Accessed: 13-May-2017].

[7] Nuand LLC, “libbladeRF Examples.” Github. [Online]. Available: https://github.com/Nuand/bladeRF/tree/master/host/libraries/libbladeRF/doc/examples. [Accessed: 13-May-2017].

[8] J. D. Gaeddert, “liquid-dsp,” Github. [Online]. Available: https://github.com/jgaeddert/liquid-dsp. [Accessed: 13-May-2017].

[9] J. Chroboczek, “The Babel routing daemon,” Github, 06-Dec-2016. [Online]. Available:

https://github.com/jech/babeld/tree/194a5360606a2b70aa47cb37ce5586aa4e459f8b. [Accessed: 13-May-2017].

[10] O. Lule, E. Ozturk, I. Ugur, “sdr-performance-analysis,” Github. [Online]. Available: https://github.com/esrefozturk/babel-performance-sdr/ [Accessed: 13-May-2017].

[11] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” Proceedings of the 9th annual international conference on Mobile computing and networking - MobiCom '03, 2003.

[12] Canonical Ltd, “Ubuntu Desktop 16.04.” [Online]. Available: https://www.ubuntu.com/desktop/. [Accessed: 13-May-2017].

[13] USB Implementers Forum, Inc, “Universal Serial Bus Revision 2.0 specification.” [Online]. Available: http://www.usb.org/developers/docs/usb20_docs/usb_20_033017.zip. [Accessed: 13-May-2017].

[14] O. Lule, E. Ozturk, I. Ugur, “Performance Analysis Results: Dataset 1,” Github. [Online]. Available: https://github.com/esrefozturk/babel-performance-sdr/tree/master/1. [Accessed: 13-May-2017].

[15] O. Lule, E. Ozturk, I. Ugur, “Performance Analysis Results: Dataset 2,” Github. [Online]. Available: https://github.com/esrefozturk/babel-performance-sdr/tree/master/2. [Accessed: 13-May-2017].