

Description of the Revision of the Manuscript Submitted to Electronics Journal and Response to the Reviewers' Comments

Carlos Egas Acosta¹, Luis Criollo¹, Christian Tipantuña¹,
and Jorge Carvajal-Rodriguez¹

¹ Departamento de Electrónica, Telecomunicaciones y Redes de Información, Escuela Politécnica Nacional.

January 19, 2024

Paper Information

- **Authors:** Carlos Egas Acosta, Luis Criollo, Christian Tipantuña, and Jorge Carvajal-Rodriguez
- **Journal:** Electronics
- **Paper's title:** Efficient Default Route Configuration in IEEE 802.15.4 Protocol: A Smart Algorithmic Approach

Dear Editorial Board of the Electronics Journal, we sincerely thank you for reviewing our paper and the invaluable comments provided. These inputs significantly contribute to improving the quality of our manuscript. We have thoroughly addressed each comment and incorporated the necessary modifications accordingly.

Reviewer 3

Comment 3.1: The authors have corrected some of the results included in the previous version of the paper. They have now clearly stated the different stages involved in the transmission of a message and added some diagrams in order to clarify it. The order of the stages, however is not correct (clear channel assessment and backoff actually happen before packet transmissions, not after.), but it's an improvement respect to the previous version.

Answer 3.1: We appreciate your constructive feedback on our paper. We acknowledge the error and will promptly rectify the sequence to represent the chronological order of these stages accurately in the revised version of the paper.

Action 3.1: In the new manuscript version, we have correctly positioned the precise channel assessment time in figure 9.

Comment 3.2: I agree with the authors that packet's "hoping time" in the routers/packet forwarding nodes can be measured with the sniffer, but what I don't believe can be measured with the sniffer is the time the originating/source node takes to send a packet to the first relay node, because the sniffer can only detect the packet when it is already "in the air". Consequently (or, also), If there is only one hop, the end-to-end delay can't be measured using the sniffer.

Answer 3.2: You raise a valid concern about the ability of the sniffer to measure the time it takes for the source/source node to send a packet to the first relay node. We agree with your observation that it is indeed not possible to accurately measure the time when the source issues the frame since the time provided by the sniffer is the time when the frame arrived and was stored in the sniffer for processing and does not take into account the time when the frame left the node nor the propagation time. However, the subsequent measurement taken at the next node already considers these times, which are similar in all retransmissions due to the test scenario.

Action 3.2: In the new manuscript version, we added the following text in section 3.7

The time recorded by the sniffer denotes the moment when the frame arrives and is stored for processing, excluding the duration from when the frame departs the source node and the propagation time. It's important to note that the subsequent measurements at the next node encompass these times, including the frame's departure from the source node and the propagation time. This consistency is maintained across all retransmissions due to the test scenario.

Comment 3.3: The authors have provided the source code for their testbed, which is nice and a good practice in general. However, the source code in the github repositories they have linked in the references is quite poorly documented. There is not a README file explaining the code or the structure, and the source code itself is commented in Spanish, not in English. The code could be better formatted, presented and documented if it is going to be cited in the references.

Answer 3.3: Thank you for your suggestion. We commit to enhancing the documentation and structure of the source code in our GitHub repositories. Specifically, we will add a comprehensive README file explaining the organization of information.

Action 3.3: In the new GitHub code, we included the README file, where we translated comments from Spanish to English and improved the presentation

Comment 3.4: In addition to this, I must say that the implementation seems a "minimum viable prototype" (and very "minimum", indeed), in which a lot of values corresponding to the tested scenario (number of nodes, addresses, etc) have been "hardcoded", array limits are not correctly tested, so they could be "overflowed" (in the C code for the microcontroller), and so on.

Answer 3.4: Thank you for your insights on this matter. Indeed, the prototype is designed with minimal feasibility regarding the number of nodes.

However, it allows for the testing of the algorithm under the assumption that node identifiers are pre-assigned. The code sets the information regarding the number of nodes for testing purposes. Note that when implementing the algorithm for a real-world application, the variables within the code need to be appropriately dimensioned to enable the algorithm to operate seamlessly with a more significant number of nodes. This adjustment is essential for the scalability and practicality of the algorithm in real-world scenarios.

Action 3.4: In the new manuscript version, we added the following text in section III paragraph II.

The network diagram used to perform performance tests of the algorithms is shown in Fig 1. The prototype is minimally feasible regarding the number of nodes, but it facilitates algorithm testing, assuming that node identifiers are pre-assigned and tested in controlled scenarios.

Comment 3.5: All this do not help to support the claims made by the authors, but since the other reviewers think the paper is suitable for publication, I am not going to be the one hindering it. However, I recommend the author to tidy and better document their source code, and clearly state that is a very basic minimum viable prototype in the documentation.

Answer 3.5: Thank you for your feedback and consideration. We have included a basic viable prototype in the document that allows the algorithm to be tested in controlled scenarios.

Action 3.5: In the new manuscript version, we added the following text in section III.

The prototype is minimally feasible regarding the number of nodes, but it facilitates algorithm testing, assuming that node identifiers are pre-assigned and tested in controlled scenarios.