

## 1 Introduction

End-to-end communication is in many ways the basics of communication networks. Sending a message from one point, the sender, to another, the receiver. It is what we call a basic primitive in communication networks. The information must be able arrive at the receiver, one part of the whole message at a time. No omissions, duplications or reordering of the data are allowed.

But just like two persons, talking to each other, can't understand each other anymore when there's too much noise around them, the same goes for communication networks. Because of this, errors tend to unfold in the network between the communicating entities. It is especially at moments when the communication network experiences high loads, that the chance of errors becomes very high, and eventually we can't guarantee stable communication anymore.

In this category of problems, Wireless Sensor Networks (WSNs) are one of the technologies that are inherently the most prone to interference. The deployment of WSNs has been steadily on the rise in recent years, particularly due to the availability of sensors that keep getting smarter, cheaper, and more intelligent due to big advances in sensor technologies. Another reason of this sudden rise has everything to do with two buzzwords that have been circulating in the media in the last couple of years: IPv6 and the Internet of Things (IoT). But WSNs and sensors in general are often deployed in outside environments and very harsh conditions, like volcanoes[?]. It's no surprise then that these little devices are quite prone to catching noise. Luckily, their increase in popularity has brought about a field of researchers that aim to reduce this inherent interference.

The aim of research in this field has been to remove outside interference out of the equation as much as possible, so we are able to guarantee reliable and well performing end-to-end communication. Removing the interference is not always easy to accomplish, since we often have little control over the outside world. But we can, however, target the errors in the communication channel that this interference causes. Thus, error detection and error correction are two field where research is most focused, in the hopes of minimizing errors as much as possible. In combination with these technologies, research looks toward the concept of self-stabilization. This means that an algorithm should be able to recover from any arbitrary state, after encountering an error for example.

Chalmers University, and specifically my supervisor, Elad Michael Schiller, have developed an algorithm that employs these two main technologies: error correction codes and self-stabilization. The algorithm they present can be applied to dynamic networks of bounded capacity that omit, duplicate and reorder packets.

## **1.1 Background and related work**

Gulliver[?], is a platform for studying vehicular systems on a large scale open source test-bed of low cost miniature vehicles that use wireless communication and are equipped with onboard sensors. It was developed at Chalmers University of Technology, and my supervisor was part of the team that presented it. The paper about self-stabilizing end-to-end communication that is the building block of this thesis, was written as part of a series of papers presented as part of the Gulliver project.

At the time of presenting the algorithm, this was the only algorithm of its kind. There many attempts in the general direction, but none could satisfy all the guarantees that this algorithm can[?]. My assignment is to implement the algorithm and make practical proof regarding the theoretical statements the original paper makes in terms of guaranteeing stable communication.

After doing thorough background work, we have come to the conclusion that there are still no efforts or algorithms produced that can guarantee the same kind of stable communication. At the time of writing this report, the best efforts concerning this type of application are still the ones referred to in the original paper, so I will not repeat them here. There is also, to the best of my knowledge, no practical implementation of any kind available.

## **1.2 My contribution**

This thesis investigates the practical implementation in TinyOS and proof of the presented self-stabilization algorithm. This thesis presents the first, to the best of my knowledge, practical implementation and proof of a self-stabilizing end-to-end algorithm for reliable FIFO message delivery over bounded non-FIFO and duplicating channel.