Byzantine-Resilient Real-Time Reliable Broadcast on Partially Connected Topology Cases

Authors

Thom Breugelmans
t.l.breugelmans@student.tudelft.nl

Responsible Professor

Jérémie Dechouchant
j.decouchant@tudelft.nl

Supervisor

Bart Cox
b.a.cox@tudelft.nl

1 Introduction

Many of our physical structures are automated and monitored by digital systems, so called *Cyber-Physical Structures* (CPS).

Information and commands need to be relayed to the processes in these structures quickly and reliably.

CPS are often victim of attacks by malicious parties, despite attacks correct processes need still be able to reliably communicate.

Such applications, e.g. SCADA or IoT, can contain hundreds to thousands of devices and are networked, often wirelessly. As a result bandwidth consumption becomes high as well as the cost of connecting the devices.

2 Objective

Construct a protocol that is capable of real-time execution in partially connected topologies and has low bandwidth consumption.

3 Related Work

Previous literature presented either real-time reliable broadcast (e.g. [1]) that cannot tolerate attacks, or asynchronous byzantine-reliable broadcast [2],[3], which does not support real-time.

Two protocols, named RT-ByzCast [4] and PISTIS [5] by Kozhaya et al., present a real-time byzantine reliable broadcast protocol. These protocols tick most of the boxes we require, however both require of fully connected networks to function.

4 Protocol Overview

These protocols ([4],[5]) work under the assumption that a network is of size 3f+1. The f is the number of byzantine processes in the network.

Both protocols flood the network with 3 types of messages:

Echos

Heartbeats **Delivering**

5 Method

Lowering connectivity

Figure out performance implications of protocols when lowering connectivity to as low as f+1

Reducing bandwidth consumption

The method works as follows:

1. A wants to send a message to B
2. For any signature A has received
   a. Compute the probability B has received the signature from A or the signing process
   b. If the probability is lower then some threshold, then send the message as B has probably not received the signature.

6 Results

A *Lowering connectivity*

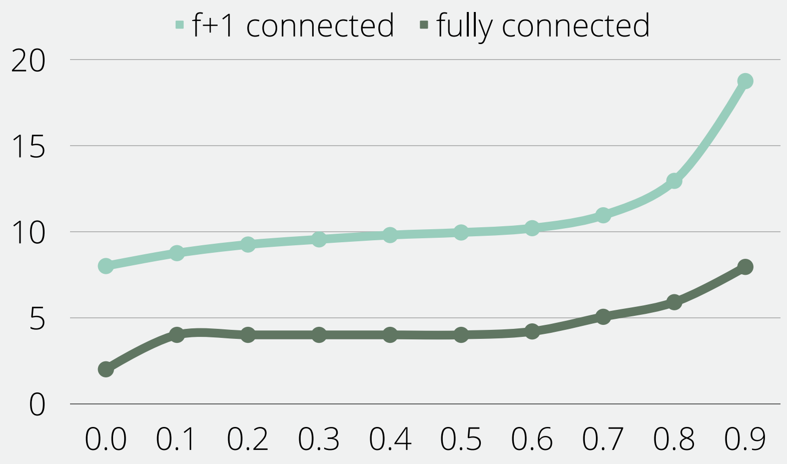

Figure 2: rounds until successful broadcast over increasing probabilistic link failure N=73 (f=24)

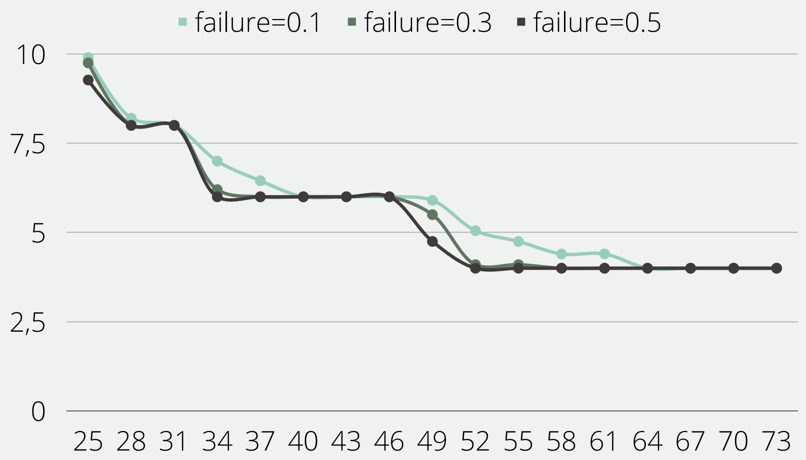

Figure 1: rounds until successful broadcast with increasing connectivity on networks of N=73 (f=24)

B *Reducing messages*

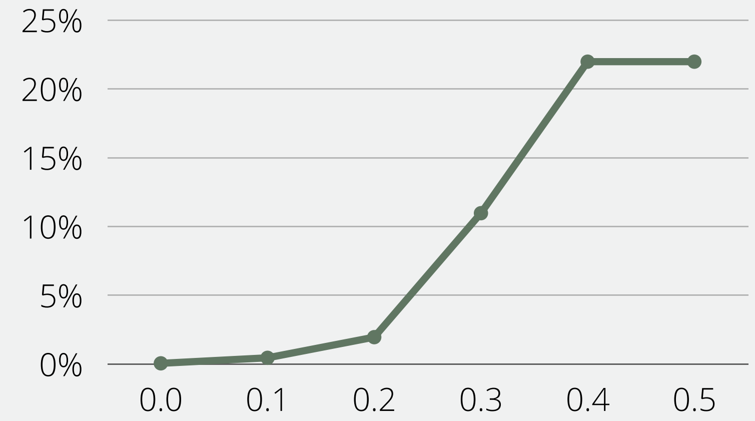

Figure 3: percentile message decrease compared against different threshold values for our proposed message decrease method. (f=49, prob. linkfailure=0.6)

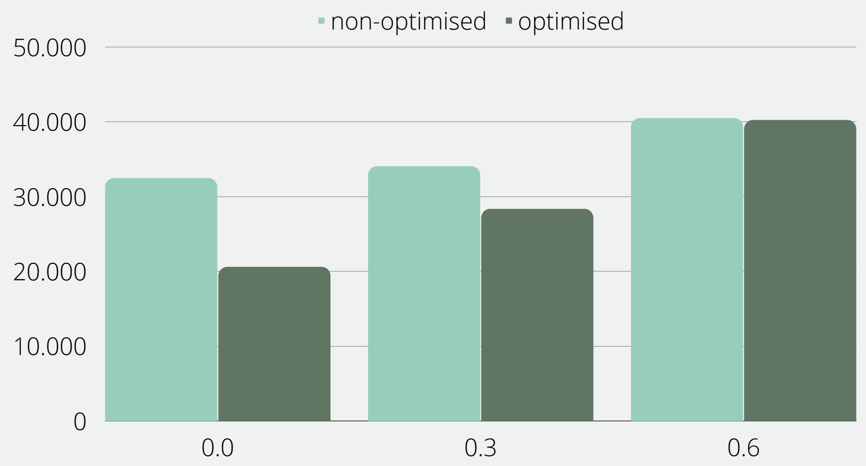

Figure 4: histogram showing the amount of messages sent for non-optimised (light) versus optimised (dark) on probabilistic link failures of 0.0, 0.3 and 0.6

7 Conclusion

Protocols RT-ByzCast and PISTIS do not require a fully connected network but can support partially connected networks with connectivities as low as f+1.

Performance drops by ~50%, however a trade-off between connectivity and performance can be made.

Our proposed method can decrease the messages by around 20-30% with the right parameters.

Abbreviations

CPS	- Cyber-Physical Structure
SCADA	- Supervisory Control And Data Acquisition
IoT	- Internet of Things

References

[1] R. Jacob, M. Zimmerling, P. Huang, J. Beutel, and L. Thiele, "End-to-end real-time guarantees in wireless cyber-physical systems," in *RTSS*, 2016, pp. 167-178.

[2] G. Bracha, "asynchronous byzantine agreement protocols," *Inf. Comput.*, vol. 75, no. 2, pp. 130-143, 1987.

[3] D. Dolev, "The byzantine generals strike again," Stanford University, CA, USA, Tech. Rep., 1981.

[4] D. Kozhaya, J. Decouchant, and P. Esteves-Verissimo, "RT-ByzCast: Byzantine-resilient real-time reliable broadcast," *IEEE Trans. Comput.*, vol. 68, no. 3, pp. 440-454, Mar. 2019.

[5] D. Kozhaya, J. Decouchant, V. Rahli, and P. Esteves-Verissimo. "PISTIS: An Event-Triggered Real-Time Byzantine-Resilient Protocol Suite." IEEE Transactions on Parallel and Distributed Systems 32, no. 9 (2021): 2277-2290.