Universität Rostock

Traditio et Innovatio

On the Reliability of Time-Sensitive Network Infrastructures

Willi Brekenfelder, Tom Reincke, Helge Parzyjegla, Peter Danielis, Omer Hanif Khan, Gero Mühl

Introduction

To counteract faulty links and switches, Frame Replication and Elimination for Reliability (FRER) is defined in the IEEE 802.1CB sub-standard that is part of the Time-Sensitive Networking (TSN) series of standards. FRER provides mechanisms for sending the same data frame redundantly over multiple network paths from a source to a destination. For reliable communication, it is sufficient that at least one of the replicated frames finally reaches the destination. Since the network load should be kept as low as possible, it is important to replicate the packets only as often as necessary to achieve the required reliability. To ensure this, it is crucial to calculate the probability that the message will reach the receiver as efficiently as possible.

Decomposition Algorithm

The decomposition (decomp) algorithm alternately simplifies and decomposes the network. Only a small network structure, as shown in Fig. 1 can be simplified and multiple links can be replaced by a single link.

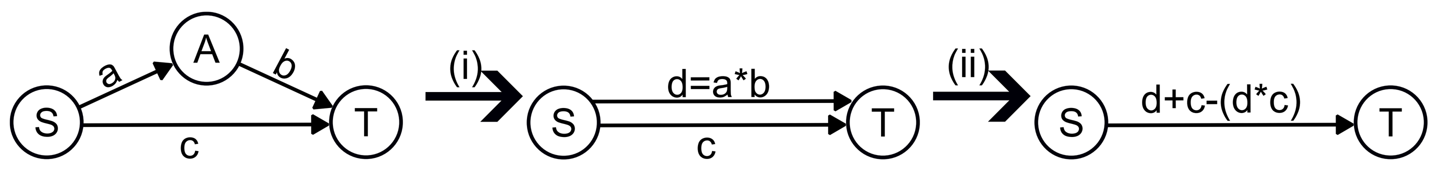

Figure 1: Simplification of (i) serial and (ii) parallel components.

If a network contains subnets that are neither serial nor parallel, the algorithm decomposes the network as shown in Fig. 2. It selects a link and considers two cases; one where the link is faulty (a) and therefore removed, and one where it is not faulty (b) and therefore the two adjacent nodes are combined. Both resulting networks are simplified again until there is only one link left, or it has to be decomposed again. Knowing the reliabilities of the two decomposed networks, the overall reliability of the initial network can be calculated as shown in Fig. 2.

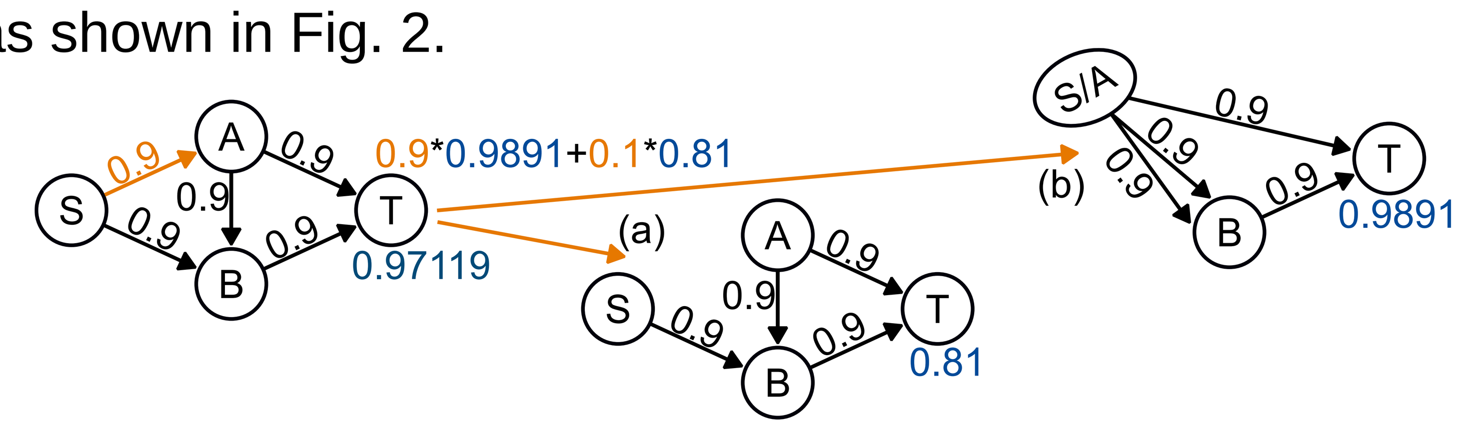

Figure 2: Decomposition using link from S to A.

Extension of the Decomp Algorithm

In addition to the link failure rates, it is also possible to take node failure into consideration. To apply the same decomposition algorithm, it is useful to transform a network with fallible nodes into a network with only link failures. This can be accomplished by splitting a node into two distinct nodes. One node retains all ingress links and the other node receives the egress links. Both nodes are then connected by a new link that stores the nodes' survival rate.

Evaluation

The decomp algorithm was compared to a brute-force approach. The brute-force algorithm checks every simple state of the network to determine if there is at least one successful route from source to destination. The resulting probabilities of those states are added up to calculate the overall reliability. Both algorithms were evaluated using the same networks build from the three base topologies shown in Fig. 4. The line and ring topologies are simple networks whose reliability can be calculated through network simplification. However, the Ladder is a more sophisticated approach that requires decomposition.

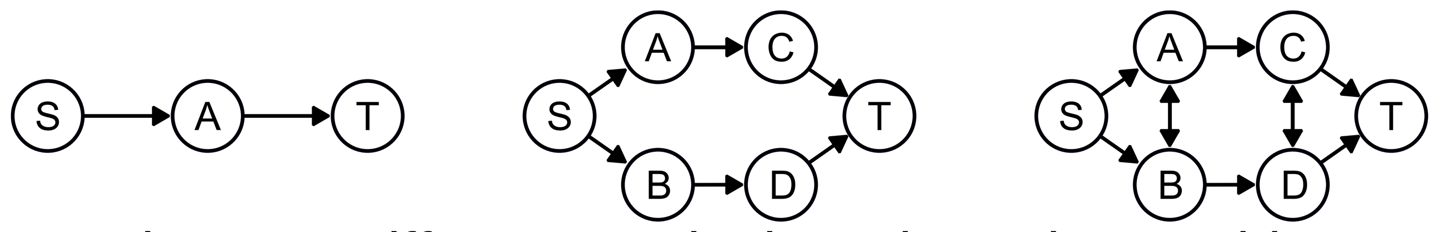

Figure 4: Different topologies: Line, Ring, Ladder.

All networks used to determine the runtimes were generated from the base topologies with different numbers of links and different link survival rates. The calculation of reliabilities delivered the same results. However, the runtimes, differed between the two algorithms, as shown in Fig. 5. It is important to note that the runtime of the brute-force algorithm is only dependent on the number of links, whereas the decomp depends on the number of decomposition steps.

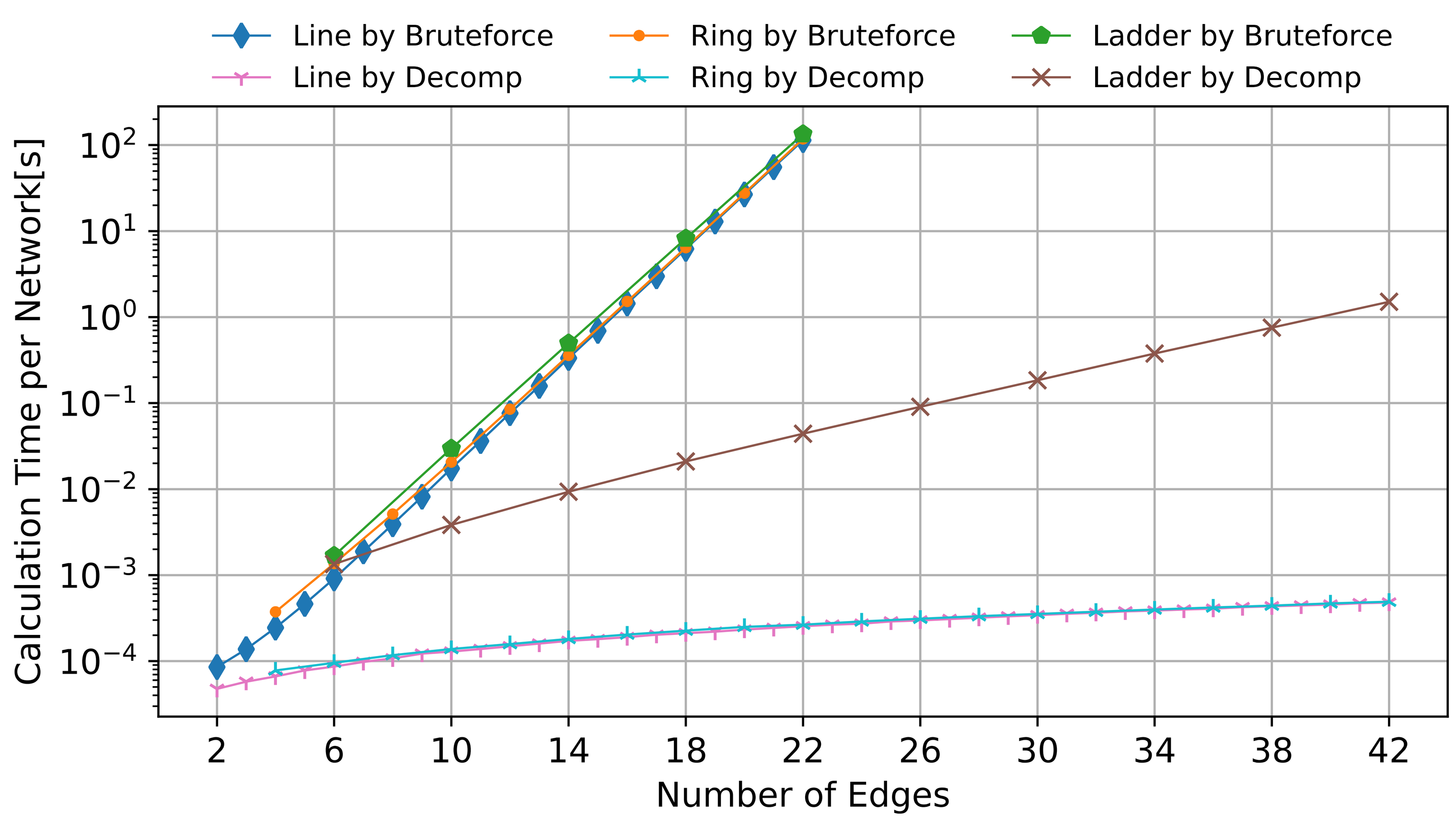

Figure 5: Runtime vs. number of edges for different topologies calculated by brute-force and decomp algorithm.

Conclusions

The decomp algorithm accurately calculates the network reliability while maintaining an acceptable runtime for many network topologies. It is particularly fast when analyzing networks that contain many serial and parallel structures that can easily be simplified. The decomp algorithm is a valuable tool for planning and scheduling realtime traffic in converged network infrastructures that have to fulfill latency constraints and reliability objectives

Faculty of Computer Science and Electrical Engineering | University of Rostock
Institute for Computer Science | Architecture of Application Systems
Albert-Einstein-Straße 22 | 18059 Rostock | Germany