User-Friendly Method to Optimize the Network of a Cyber-Physical System

STETSON UNIVERSITY

Samantha Harris (Stetson University) and Levi Welch (University of Michigan)

Mentor: Matt Bunting (University of Arizona)

Abstract

Many cyber-physical systems (CPS), such as self-driving cars, require numerous components working together, which can result in a complex network. This research focused on user-friendly network optimization methods on the autonomous vehicle at the University of Arizona. The method verified that the network operated under cost, bandwidth, latency (time delay in the transfer of information) and processing power constraints. Operating within these constraints ensures that the system is safe and efficient. The optimization method discussed in this poster can be customized for any cyber-physical system.

Project Objectives

- Simplify the network optimization process
- Ensure the safety of a CPS

Introduction

Cyber-physical systems (CPS) are devices that use computational and physical skills to function in an environment[1]. Many CPS's require numerous components working in unison, which can result in a convoluted network. CPS developers mainly concentrate on functionality and dynamics rather than network optimization. It is important to optimize the network of the CPS in order for the device to run efficiently. This task is usually done by experts in the domain which can cause an inconvenience to developers without network optimizing expertise to ensure that their CPS is safe to operate. One example of a CPS is a self-driving car. The purpose of this project is to present a user-friendly method to optimize the network of the University of Arizona's autonomous vehicle, which can be applied to any cyber-physical system.

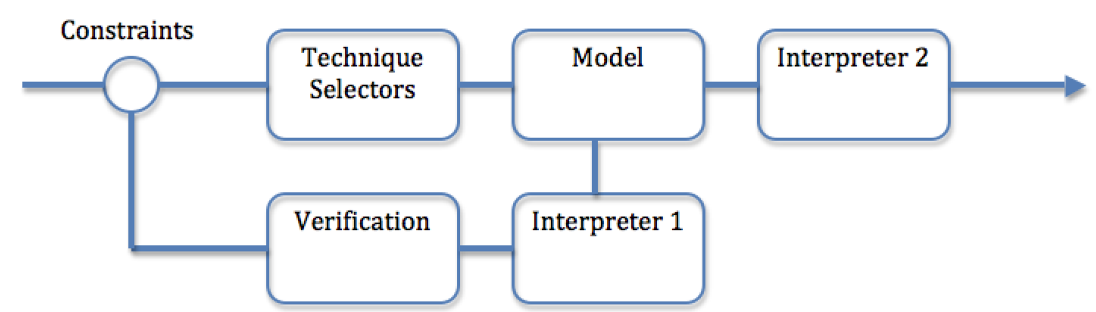

Figure 1: Figure 1 displays the optimization process, starting with the block labeled "model" and continuing in a clockwise manner.

Methods and Materials

The user-friendly method begins by creating a model of the CPS's system. A model of the network is created with a Domain Specific Modeling Language on a Web-based Generic Modeling Environment (WebGME). Two different interpreters, written in JavaScript, are used to verify that the system functions adequately and produce template code to operate the autonomous vehicle. To ensure adequacy, four constraints are implemented to the CPS's network. The constraints are cost, bandwidth, latency (time delay in the transfer of information) and processing power. If the constraints are not met, a technique on how to improve the system is selected and provided to the user (technique selectors)[2]. The technique can be selected from a variety of possible solutions, such as replacing hardware or rewiring connections. The new structure of the CPS's system can be tested in a simulation called the Robotic Operating System (ROS). The process is repeated until the network operates within each constraint. The second interpreter uses the information in WebGME and generates template code and configuration files that are used to control the vehicle.

Figure 2: The Cognitive and Autonomous Test (CAT) Vehicle at the University of Arizona.

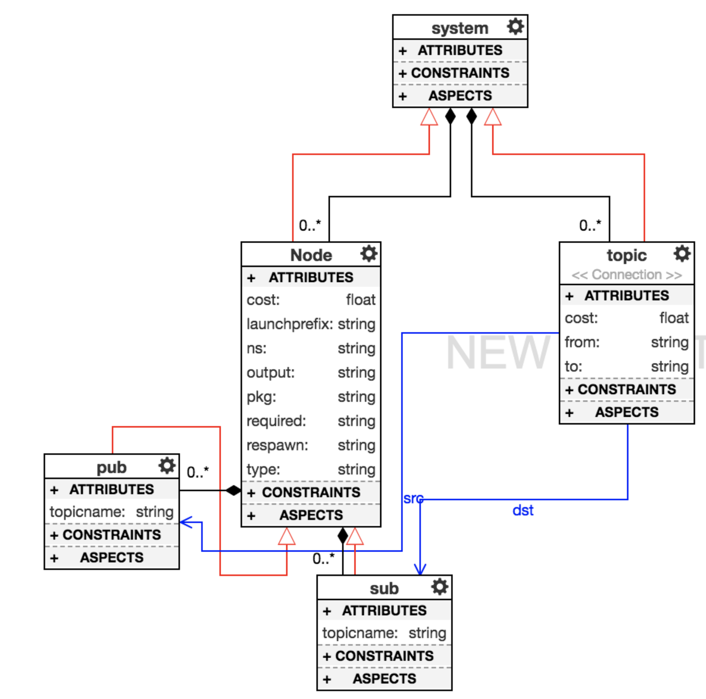

Figure 3: Example of a modeled network in WebGME.

Summary of Results

The optimization method discussed in this poster can be applied to any cyber-physical system. It allows CPS developers to optimize the network of the system themselves, which can cut development costs and increase productivity. The tools that WebGME supplies allow for an easy way to plug interpreters into the model and generate code. The interpreters and technique selectors can be customized to ensure that the network is structured safely and generate template code to run the system a specific CPS. More research is essential for technique selectors, however, the groundwork has been established for user-friendly optimization methods.

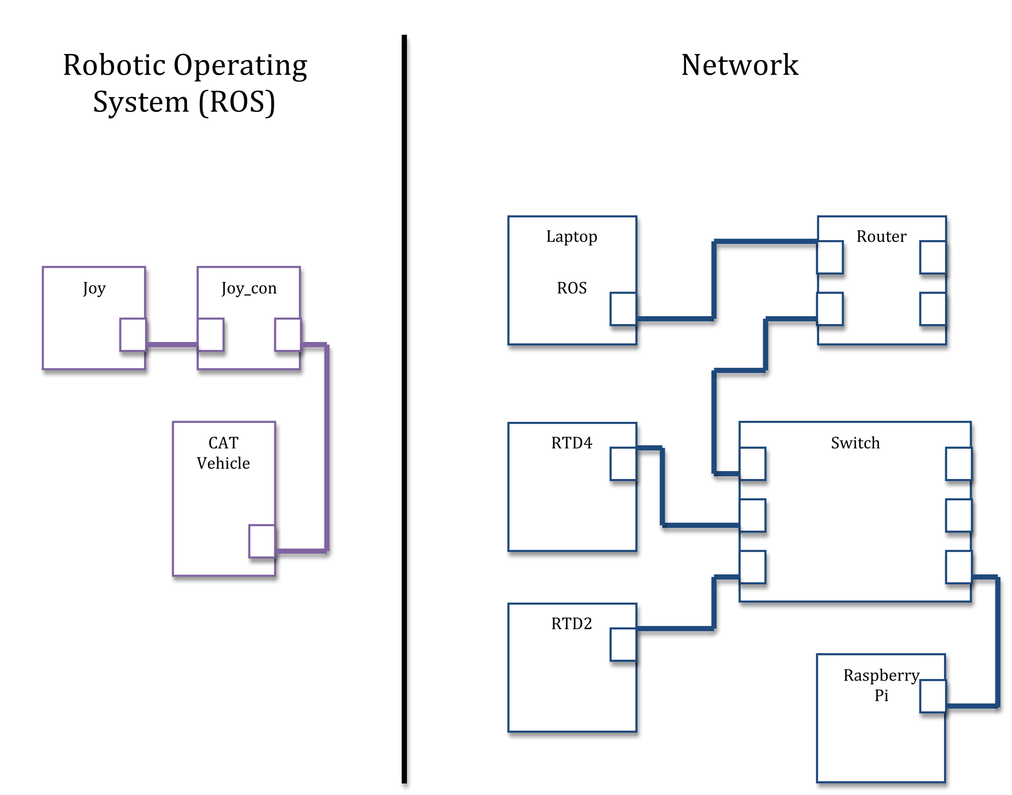

Figure 4: One example of communication between the Robotic Operating System and the network of the vehicle.

Future Work

One way to improve efficiency is to automatically measure and enter the constraint values into WebGME. This process would require extracting measurements from multiple different platforms and using an interpreter to update them into WebGME. In addition, an interpreter that automatically uploads a file with the template code would lead to faster test results in the Robotic Operating System. Furthermore, rewriting the JavaScript files in a way that requires fewer lines of code would decrease the lines of instruction for the computer to read and follow. Lastly, this project did not reach the stage of implementing technique selectors. Due to the existence of more than one way to restructure the network, a weighing process of each technique could help determine which to select.

Figure 5: Using a joystick controller to navigate the vehicle was a first step in transitioning to a fully autonomous vehicle.

References

[1] R. Baheti and H. Gill, "Cyber-physical systems," *The Impact of Control Technology*, vol. 12, no. 1, pp. 161–166, 2011.

[2] M. Bunting, private communication.

Acknoledgements

- University of Arizona NSF REU Program
- The Southeastern Section of the American Physical Society
- Stetson University Dean's Fund
- Stetson University Physics Department