DEVS BASED ROBUST COMMUNICATION PROTOCOL FOR INTER-SIMULATION COMMUNICATION IN CADMIUM

Sasisekhar M. Govind[a]

[a]Advanced Real-Time Simulation Laboratory, Carleton University, Canada
sasisekharmangalamgo@cunet.com

SUMMARY

Intercommunication between processes in a distributed hard real-time system have to meet extremely tight timing deadlines. One of the bottlenecks in the communication process is the protocol used within the system. This paper presents a formal definition and implementation of a communication protocol that is reliable and predictable for use with distributed hard real-time systems and by extension, distributed real- time simulations. The protocol presented in this paper was modeled using the Discrete Event System specification and simulated using the Cadmium simulator. We present a case study wherein we distribute a centralized model using the protocol. Upon simulation, the behavior of the system before and after distribution were observed and their congruence determined.

Keywords: DEVS, Distributed systems, RT-DEVS, Communication Protocol, Distributed Simulation

ACKNOWLEDGMENTS

I wish to express my sincere thanks and deep sense of gratitude to my thesis supervisor Dr. Gabriel Wainer for their consistent encouragement and valuable guidance.

Proc. of the 2024 Annual Simulation Conference (ANNSIM'24), May 20-23, 2024, American University, DC, USA
P.J. Giabbanelli, I. David, C. Ruiz-Martin, B. Oakes and R. Cárdenas, eds.

PHD COLLOQUIUM ABSTRACT

Hard real-time (RT) distributed systems are decentralized over multiple inter-connected compute nodes, wherein, each node runs various processes, each of which need to be completed within an explicit deadline. These processes may or may not be dependent on the output produced by another node within the system. When two processes running on different nodes are inter-dependent on each other, one of the prime factors that obstruct the system's capability of meeting the deadlines is the communication network employed within the system. One of the major steps to be taken to improve the communication network, would be to develop a robust, deterministic, and reliable communication protocol.

In the industry, where hard RT distributed systems are prevalent, this issue was over come by using simple yet robust communication protocols, like the Field-bus or CAN protocols. However, since Ethernet has become the standard communication protocol for businesses and educational institutions, the cost of setting up an Ethernet-based communication network has dropped significantly. With the infrastructure widely and cheaply available, there is incentive to move forward from outdated protocols like RS485[1].

However, Ethernet, as a protocol was not developed with hard RT communication in mind. Hence, there is a need to slightly modify the protocol stack to mold it to the needs of a hard real time distributed system. The authors in [2] go into detail about the various paths to be taken to develop such an RT protocol by removing or modifying a few layers of the Ethernet communication stack.

The modification of the Ethernet protocol allows us to establish RT communication between various nodes in a distributed system. Nevertheless, the development of a communication protocol that can transfer data deterministically and reliably requires the developer to follow a rich modelling formalism. Modelling allows the developer to simulate the protocol per the environment it will be deployed in, enabling the developer to verify and validate the performance of their protocol. This would allow for the development of a deterministic and reliable protocol, wherein, the developer is completely aware of the possible behaviors of their model, hence enabling one to develop a protocol that is not only real-time, but also predictable in its faults.

This thought process can be extended towards the development of complex distributed systems. As systems increase in complexity, modelling the same leads to simpler and broader understanding of the system, hence the prevalence of UML diagrams etc. Despite this, there always seems to exist a disconnect between the model and the implementation of the system. Hence, this leads us to find a rich modelling formalism that can accurately and simply describe the behavior and structure of a system, so that the disconnect between implementation and modelling can be completely neglected [3].

One such modelling formalism is the Discrete Event System Specification (DEVS), which deals with systems that are continuous in time, but expect discrete events[4]. The Modeling and Simulation Based Engineering (MSBE) is a methodology that draws on the DEVS formalism to create a development paradigm for embedded and cyber physical systems [5].

This work portrays the development of a robust, deterministic, and reliable communication protocol based on the DEVS formalism to facilitate communication between different DEVS models running on different execution nodes. The DEVS-Inter Atomic Communication (DEVS-IAC) protocol borrows wisdom from the Ethernet communication stack, and formalizes the layers using the DEVS formalism, improving and modifying the protocol to fit the needs of DEVS models, and implementing the same using MSBE in a tool called CADMIUM, on the ESP32S3 Microcontroller platform.

The DEVS-IAC protocol provides a fully flexible DEVS based protocol that can be modified per the modeller's requirement. In Hard-RT systems, the delays involved in communication would be a defining factor in the system meeting its deadlines. This protocol's integration into DEVS facilitates precise latency com-

putations by modelers, thereby allowing for an incorporation of these delays directly into the models. This approach minimizes the discrepancy between simulated and actual behavior, enhancing model accuracy.

By establishing a statistically significant dataset of latency measurements, researchers can accurately characterize communication delays. Once these latencies are rigorously defined, it becomes feasible to execute distributed real-time models that perform closely to their simulations. This advancement extends the scope of MBSE, empowering engineers across various disciplines to deploy their systems more effectively and ensure operational stability.

Future research directions will focus on the comprehensive formalization of latencies encountered during model execution on a DEVS engine. Following the formalization, the development of an optimized simulation algorithm is planned, which will incorporate these latencies to improve RT performance. The objective is to refine this algorithm to such an extent that it can simulate all models with an accuracy that mirrors their real-time execution, thereby enhancing the accuracy of DEVS-based simulations.

REFERENCES

[1] P. Neumann, “Communication in industrial automation-what is going on?” *IFAC Proceedings Volumes*, vol. 37, no. 4, pp. 63–71, 2004.

[2] P. Pedreiras, L. Almeida, and J. A. Fonseca, “The quest for real-time behavior in ethernet.” *The Industrial Information Technology Handbook*, vol. 2005, pp. 1–14, 2005.

[3] H. Saadawi and G. Wainer, “Verification of real-time devs models,” in *Proceedings of the 2009 Spring Simulation Multiconference*. Citeseer, 2009, pp. 1–8.

[4] B. Zeigler, “Theory of modeling and simulation,” *Academic Press google schola*, vol. 2, pp. 1779–1785, 2000.

[5] G. A. Wainer, “Applying modelling and simulation for development embedded systems.” in *MECO*, 2013, pp. 1–2.