Combining model-based development and formal verification of a complex ROS2 multi-robots system using Timed Rebeca

Hiep H. Trinh[1,†], Marjan Sirjani[1,*,‡], Fereidoun Moradi[1,‡], Antonio Cicchetti[1,‡] and Federico Ciccozzi[1,‡]

[1]School of IDT, Mälardalen University, Västerås, Sweden

Abstract

ROS2 is an increasingly popular middleware framework for developing robotic applications. A ROS2 application is composed of nodes that run concurrently, coordinate with each other through asynchronous interfaces, and can be deployed in a distributed manner. Rebeca is an actor-based language for modelling asynchronous and concurrent systems. Timed Rebeca adds timing features to deal with timing requirements of real-time systems. The similarities in concurrency and message-based asynchronous interactions of reactive nodes, and the ability of modelling timed behaviours justify using Timed Rebeca models to assist the development and verification of ROS2 applications. Model-based development and model-checking techniques allow faster prototyping and earlier detection of system errors before developing the entire real system. However, there are challenges in bridging the gap between continuous behaviours of robotic systems and discrete states in a model for automatic verification, and between complex robotic computations and inequivalent programming facilities in a modelling language like Timed Rebeca. We investigated the problem systematically and have succeeded in modelling a realistic multiple autonomous mobile robots (AMR) system using Timed Rebeca, creating corresponding ROS2 demo code, and showing the match between the model and the program. Based on experiments, we demonstrated the value of the model in development and automatic formal verification of correctness properties (target reachability, collision freedom, and deadlock freedom). Our model-checking results show that certain system problems are not always detected through simulation. The modelling principles, modelling and implementation techniques that are created and used in this work can be generalized for many other cases.

Keywords

Model-based development, formal verification, model checking, ROS2, robotics, Timed Rebeca, autonomous mobile robots

In complex software systems, complicated logic and timing issues cannot always be detected by testing due to the enormity of the test space and concurrent interactions. Model checking helps by formally verifying the expected correctness properties. A model is created first, and the problems are checked at design time. Discovering errors at later stages of development is generally more expensive.

Timed Rebeca [1, 2] is an actor-based language for modelling concurrent reactive systems with support for timing requirements. Rebeca has an object-based syntax and comes with an IDE to help create models and model-check them in a developer-friendly way [3]. ROS2 is a middle framework for developing robotic applications on top of its node architecture and communication protocols. A ROS2 application consists of nodes that run concurrently and interact asynchronously, typically through publish-subscribe channels and service interfaces.

In one of our industrial collaborations, we witnessed a case where a company invested considerable effort in developing a complex robotic system based on ROS2 and later found concurrency and timing problems affecting correctness and safety that were not aware of earlier. The story inspired us to use Timed Rebeca to address such problems. Comparing to the previous work, we step forward to consider a realistic multiple autonomous robots system in ROS2 with all essential components and complex behaviors [4].

Our modelling and development flow follows an iterative process. After choosing a robotic problem, we design the architecture in ROS2 node topography. We then develop a Timed Rebeca model where robots reach destinations without collision and deadlock. We then develop ROS2 simulation code based on the model, refining for smoother robotic movements. We revise the model's semantics and fine-grained level until its behavior matches the code in selected scenarios. The model checking result is only validated when adequate matching tests are passed. We model all integral parts of a multiple autonomous mobile robots system [5]: projecting static and mobile obstacles/robots onto a 2D occupancy grid, converting speeds to travel times, detecting collision by intersection of shadows, simulating LIDAR-based obstacle detection, generating paths based on A* algorithm. We apply different discretization strategies to map continuous variables to discrete state variables in Rebeca, balancing between accuracy and performance: map to occupancy grid, robot orientation from arbitrary angles to 8 typical directions, robot movement step from micro to per cell. We work out a human-like algorithm for moving and avoiding collision and a backoff algorithm for resolving congestions. We devise several workaround and optimization tricks to control the state space explosion and boost performance in the model (e.g. use a helper script, combine numbers, precompute trigonometric values).

We set up a system of five robots traveling along crossing paths to their destinations which exposes a high risk of collision and congestion to test correctness properties (collision freedom, deadlock freedom, target reachability) and the intelligence of the algorithms. The model and the program are set up to work on the same parameters. The fine-grained level of the model (cell size vs. robot length) is set properly to retain the same behaviors. Two kinds of cases are tested: working and non-working. In working cases, parameters are set within safety ranges, the robots reached their destinations without collision and congestion, the model and the program behaved the same. In non-working cases, parameters are set out of safety ranges, a collision was detected consistently by the model but did not always show up in the simulation. The model has helped us discover configuration parameters, their impact on the behaviours of the robots and their safety thresholds.

International Workshop on Reliability Engineering Methods for Autonomous Robots – REMARO 2024

*Corresponding author

† Principal investigator

‡ Advisors

hiep.hong.trinh@mdu.se (H. H. Trinh); marjan.sirjani@mdu.se (M. Sirjani); fereidoun.moradi@mdu.se (F. Moradi); antonio.cicchetti@mdu.se (A. Cicchetti); federico.ciccozzi@mdu.se (F. Ciccozzi)

References

[1] A. H. Reynisson, M. Sirjani, L. Aceto, M. Cimini, A. Jafari, A. Ingólfsdóttir, S. H. Sigurdarson, Modelling and simulation of asynchronous real-time systems using Timed Rebeca, Sci. Comput. Program. 89 (2014) 41–68. URL: https://doi.org/10.1016/j.scico.2014.01.008.

[2] M. Sirjani, E. A. Lee, E. Khamespanah, Model checking software in cyberphysical systems, in: 44th IEEE Annual Computers, Software, and Applications Conference, COMPSAC 2020, Madrid, Spain, July 13-17, 2020, IEEE, 2020, pp. 1017–1026. URL: https://doi.org/10.1109/COMPSAC48688.2020.0-138.

[3] M. Sirjani, Power is overrated, go for friendliness! expressiveness, faithfulness, and usability in modeling: The actor experience, in: M. Lohstroh, P. Derler, M. Sirjani (Eds.), Principles of Modeling - Essays Dedicated to Edward A. Lee on the Occasion of His 60th Birthday, volume 10760 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 423–448. URL: https://doi.org/10.1007/978-3-319-95246-8_25.

[4] H. Trinh, Model based development and verification of ROS2 robotic applications using Timed Rebeca, Master Thesis, Malardalen University (2023).

[5] H.Trinh, Released rebeca model (public) (2023). URL: https://github.com/thhiep/ros2rebeca_model.