MARAE: Component-based proven-by-construction robust control software for space autonomy

Jean-Paul Blanquart[(1)], Saddek Bensalem[(2)], Félix Ingrand[(3)], David Powell[(3)]
(1) Astrium Satellites. 31, avenue des cosmonautes, F-31402 Toulouse Cedex 4, France
Email contact: Jean-Paul.Blanquart@astrium.eads.net
(2) Verimag, Université Joseph Fourier, 2 avenue de Vignate, 38610 Gières, France
(3) LAAS-CNRS, 7 avenue du Colonel Roche, F-31077 Toulouse Cedex, France

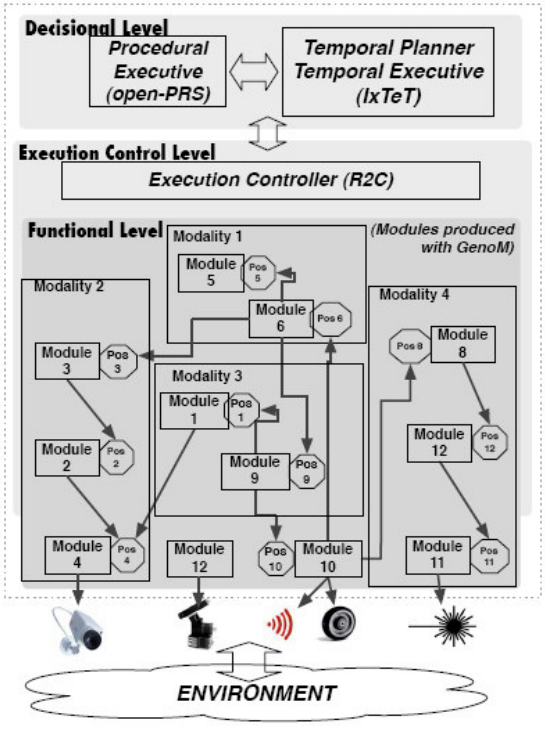
1. Introduction

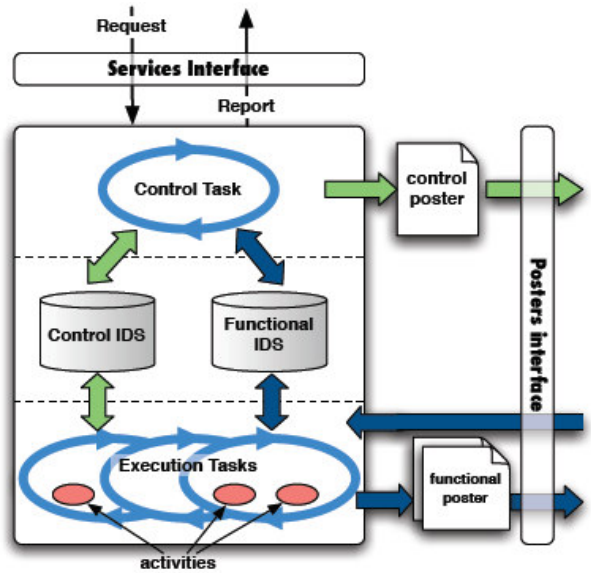
With the increasing complexity of exploration rovers, satellites, and space probes, it has now become a challenge to prove that these systems will behave appropriately/safely in all situations that they may encounter. Such systems make use of complex hardware equipment, but also increasingly complex software which must be trusted at the appropriate level of confidence in terms of correctness as well as robustness. To this end, we present an evolution of the LAAS architecture for autonomous systems, and its tool GenoM. This evolution relies on the BIP component-based design framework, which has been successfully used in other domains such as embedded systems. Particularly, we address the componentization of the functional level, the synthesis of an execution controller for it, and how we verify whether the resulting functional level conforms to properties such as deadlock-freedom. Our approach has been fully implemented in the LAAS architecture, and the implementation has been used in several experiments on a real rover and a simulated satellite. Finally an extensive campaign of robustness testing has been conducted.

2. Background

2.1. Layered modular architecture

The complexity of advanced autonomous systems is often handled through a hierarchical architecture such as the LAAS (LAAS Architecture for Autonomous Systems) architecture featuring in particular a functional layer i.e., the basic perception and action capacities activated by the decisional layer through the intermediate execution control layer that controls the proper execution of the services according to predefined rules and constraints (figure on the right).



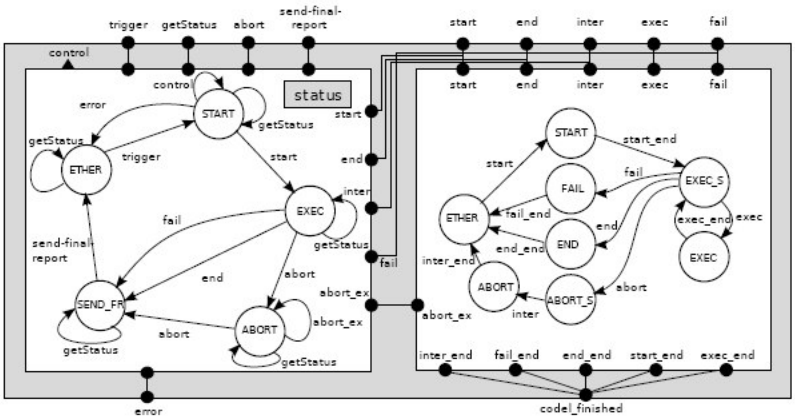


We focus here on the functional layer, aiming in particular at improving the execution control capabilities by addressing the internal execution of the modules. In the current approach, the modules are generated thanks to a tool called GenoM (Generator of Modules), according to a generic scheme (figure on the left).

2.1. The BIP framework

BIP (Behaviour, Interactions and Priorities) is a framework for modelling, generating and executing heterogeneous real-time programs. It uses model-based and component-based paradigms with rigorous definition and separation of structure and behaviour. The composition operator allows incremental construction and preservation at run-time of proven-by-construction properties. The objectives of the study are to benefit from

these characteristics, without losing the modular and layered organisation. Towards this end we elaborated a tool that automatically transforms GenoM modules into BIP, thanks to generic BIP models of GenoM services.



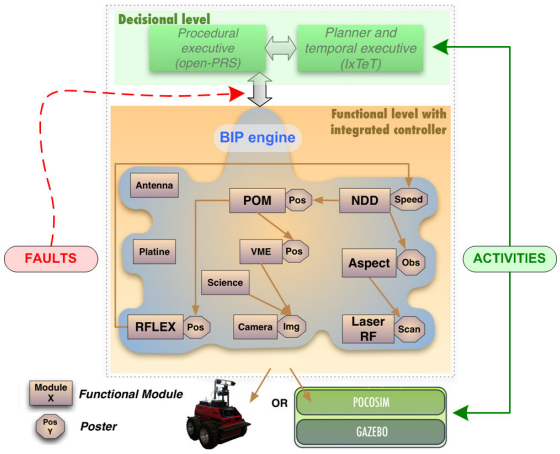
3. Experimentation

We have experimented successfully the approach on two case studies, one using a simulator of an earth observation satellite (with representative mission software previously developed as a layered decisional architecture with GenoM modules at functional layer) and one on a real robot and its simulator. The experimentation confirmed the feasibility and interest of the approach. The automatic transformation covers all the software (of the functional level), completed of course by the manual definition and incorporation of the properties to be ensured at run-time

4. Validation, Robustness testing

In addition to the application of formal validation techniques (with D-Finder, a compositional incremental verifier of safety properties and absence of deadlocks), we set up an extensive robustness testing campaign. We focused on time related errors (modifying the ordering or timing in sequences of requests to the functional layer), using the incorporated safety properties as robustness oracle.

In addition to direct results in terms of robustness evaluation the approach proved very useful as a support to design through the identification of subtle faults.



5. Conclusion and perspectives

The study confirmed the expectations with respect to formal component-based approaches with proven-by-construction preservation at run-time of stated properties. We elaborated an automatic transformation tool and a complete framework and tools for robustness testing, and experimented the complete approach on real case studies representative of complex autonomous space systems. Further steps are still needed to prepare direct application in space programs or other critical autonomous systems, including the qualification and real-time performance of generated code and the maturity (ease of use) and qualification of support tools. We are also investigating a support to the definition, formalisation and incorporation of the properties of interest. Finally we are working on the robustness testing approach which is very promising even beyond the scope of this study but needs additional techniques and tools in particular to facilitate the analysis of observed results (“false alarms”).

References

[1] “"Rock Solid" Software: A Verifiable and Correct-by-Construction Controller for Rover and Spacecraft Functional Levels”, Saddek Bensalem, Lavindra de Silva, Matthieu Gallien, Félix Ingrand, and Rongjie Yan, i-SAIRAS 2010, The 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space, August 29 - September 1, 2010, Sapporo, Japan.

[2] “Toward a More Dependable Software Architecture for Autonomous Robots”, Saddek Bensalem, Matthieu Gallien, Félix Ingrand, Imen Kahloul and Thanh-Hung Nguyen, Special issue on Software Engineering for Robotics of the IEEE Robotics and Automation Magazine, March 2009.

[3] “Robustness Evaluation of Robot Control Software”, Hoang-Nam Chu, Jean Arlat, Marc-Olivier Killijian, Benjamin Lussier, David Powell, European Workshop on Dependable Computing (EWDC), Toulouse, France, 14-15 May 2009.