Multi-concern Dependability-centered Assurance for Space Systems via ConcertoFLA

Barbara Gallina and Zulqarnain Haider, Mälardalen University Sweden
Anna Carlsson, OHB Sweden
Silvia Mazzini and Stefano Puri, Intecs Italy

Space systems need to be engineered in compliance with ECSS standards and need to ensure a certain degree of dependability. European Cooperation for Space Standardization (ECSS) provides standards for engineering, management and qualification of space systems. More specficaly, ECSS has dedicated standards for product assurance for dependability and safety, i.e. ECSS-Q-ST-30C [2] and ECSS-Q-ST-40C [4] respectively. Assurance of all these properties, require the modelling of various system characteristics and their co-analysis, in order to enable the management of trade-offs between them.

CHESS [1] an open source supporting toolset, which is the result of several R&D projects, promotes a component-based development methodology in which particular emphasis is given to the ability to specify the non-functional properties of components, including critical properties such as real-time and dependability related characteristics. In particular, ConcertoFLA [2] [5], which is part of CHESS toolset, allows users (system architects and dependability engineers) to decorate component-based architectural models with dependability-related information, execute Failure Logic Analysis (FLA) techniques, and get the results back-propagated onto the original model. ConcertoFLA is built on top of Failure Propagation Transform Logic (FPTC) [7] and CHESSFLA [10]. ConcertoFLA is a compositional technique to qualitatively assess the dependability of component-based systems, and partially combines and automatizes traditional safety analysis techniques (e.g., Fault Tree Analysis). Its analysis results have also been exploited to semi-automatically generate arguments fragments for safety assurance [8].

In this paper, we present the customization of the CHESS methodology [11] and ConcertoFLA in the context of the ECSS standards to enable architects and dependability experts to define a system and perform dependability-centered co-analysis for assuring the required non-functional properties of the system according to ECSSS requirements. Figure 1 shows a high-level view of the overall workflow of the proposed customization. The initial step is to define the system by modelling its components and the interactions. Then, the system design is annotated with information specifically related to system reliability, safety and security, associated directly with the involved architectural elements, at the chosen level of architectural detail, and the overall dependability analysis is performed and its results are back propagated. In the next step, the results of the analysis are interpreted for multi-concern e.g., reliability, safety and security. Based on this interpretation a decision is made for introducing the dependability means by refactoring the system. This process is iterated, until the sufficient level of these concerns is met. Further, the paper reports on the usage of the proposed customization in the context of the Attitude Control Systems Engineering (ACS). The ACS plays an important role within satellites, by contributing to maintaining the orientation of the satellite in three-dimensional space and needs to be engineered in compliance with the ECSS standards to ensure a certain degree of dependability. Figure 2 shows a component-based architecture of ACS, where the Sun sensor measurements and angular velocity are provided as an input and the system computes the control torque to be applied on the satellite body to achieve the target attitude.

This work presented is partially supported by the AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems) project [6], whose main objectives were elaborated in [7].

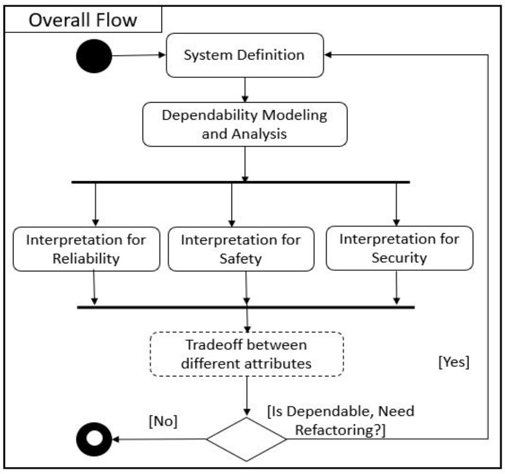
Overall Flow
System Definition
Dependability Modeling and Analysis
Interpretation for Reliability
Interpretation for Safety
Interpretation for Security
Tradeoff between different attributes
[Yes]
[No]
[Is Dependable, Need Refactoring?]

Figure 1 Dependability Co-Analysis via CHESS

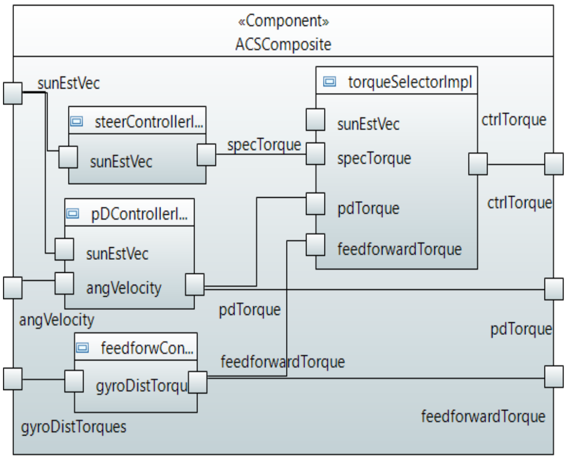

Figure 2 CHESS based architecture of ACS

Acknowledgements

This work is supported by the EU and VINNOVA via the ECSEL project AMASS (No 692474) [6].

References

[1] PolarSys CHESS, https://www.polarsys.org/chess/

[2] Gallina B., Sefer E., Refsdal A. Towards Safety Risk Assessment of Socio-Technical Systems via Failure Logic Analysis. IEEE International Symposium on Software Reliability Engineering Workshops, Naples, pp. 287-292. 2014.

[3] ECSS-Q-ST-30C Space product assurance – Dependability, March 2009.

[4] ECSS-Q-ST-40C Space product assurance – Safety, March 2009.

[5] CONCERTO Deliverable D3.3 Design and implementation of analysis methods for nonfunctional properties – Final version, 2015.

[6] AMASS, http://www.amass- ecsel.eu.

[7] Ruiz A., Gallina B., de la Vara J.L., Mazzini S., Espinoza H. Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems. Computer Safety, Reliability, and Security. SAFECOMP. LNCS, vol 9923. Springer. 2016.

[8] Alajrami S., Gallina B., Sljivo I., Romanovsky A., Isberg P. Towards Cloud-Based Enactment of Safety-Related Processes. Computer Safety, Reliability, and Security. SAFECOMP. LNCS, vol 9922. Springer. 2016.

[9] Wallace M. Modular architectural representation and analysis of fault propagation and transformation. Electronic Notes in Theoretical Computer Science, volume 141 n.3, pp. 53-71, December, 2005.

[10] Gallina B., Javed M.A., Muram F.U., Punnekkat S. A Model-Driven Dependability Analysis Method for Component-Based Architectures. 38th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) Cesme, Izmir, pp. 233-240. 2012.

[11] Mazzini S., Favaro J., Puri S., Baracchi L. CHESS: an open source methodology and toolset for the development of critical systems. Third Workshop on Open Source Software for Model Driven Engineering. OSS4MDE. 2016.