Paper ID: 25340
oral

SPACE SYSTEMS SYMPOSIUM (D1)
System Engineering - Methods, Processes and Tools (2) (6)

Author: Dr. Jian Guo
Delft University of Technology (TU Delft), The Netherlands, J.Guo@tudelft.nl

Prof. Eberhard Gill
Delft University of Technology, The Netherlands, E.K.A.Gill@tudelft.nl
Mr. Stefano Figari
Delft University of Technology (TU Delft), Italy, stefano.figari.to@gmail.com

MODEL-BASED SYSTEMS ENGINEERING TO SUPPORT THE DEVELOPMENT OF NANO-SATELLITES

Abstract

In the recent decade, the advancement of miniaturization and the revolutionary design philosophy make nano-satellites a success story. However, from the systems engineering methodology point of view, nano-satellites are still following the (simplified) culture of Document-Based Systems Engineering (DBSE) from traditional space industry. The lack of alignment between innovative enabling technologies and traditional systems engineering methodology can procrastinate the further advancement of nano-satellites.

This paper aims at filling the aforementioned misalignment by utilizing and evaluating Model-Based Systems Engineering (MBSE) in support of nano-satellite development. Using the DelFFi mission as a case study, this paper focuses on two particular aspects of applying MBSE: requirements development and design validation, and their link with onboard software development.

The paper consists of three primary parts. The first part is a brief review of MBSE for space projects. In the second part, a Systems Modelling Language (SysML, a representative tool of MBSE) template is developed for the modelling of the DelFFi and the way that requirements are developed is illustrated. Once the SysML description is completed, a specific simulation model is created in SIMULINK and linked through executable SysML. The execution aims at validating (or invalidating) the design by comparing it with requirements automatically, and testing SysML's adaptability to perform this kind of studies. The Attitude Determination and Control System (ADCS) design of DelFFi is of particular interest for this part of investigation.

The third part of the paper analyses the link between MBSE and the onboard software development of the ADCS. In early development phases, the ADCS models are used mainly for running simulations to verify the design, while in later phases all the models and algorithms have to be coded on the ADCS processors. Typically this is done by different engineers using different development tools. In this paper a more efficient way is implemented: the ADCS models that have been developed in SIMULINK for early stage phases and verified using executable SysML are translated automatically to hardware-independent C codes, which are then integrated with low layers of ADCS onboard software. Using this approach, a bridge between MBSE and onboard software development is built up through model sharing. The ADCS models developed earlier can be re-used, and the engineers can focus more on developing the models rather than programming. A second benefit is the increase in quality of the codes since the consistency of the simulation models and the onboard software is structurally assured.