# Commentary on "The Challenge of Configuring Model-Based Space Mission Planners" by J. Frank, B. Clement, J. M. Chachere, T. Smith and K. Swanson

**Simone Fratini**
ISTC-CNR, National Research Council of Italy
Institute for Cognitive Science and Technology, Rome, Italy
simone.fratini@istc.cnr.it

## Challenge

A very "hot" topic is addressed in this paper: the difficulty of modeling when fielding in space domains automated planning systems. This paper contains a general overview of the mission planning process and focuses on the challenges of model based planners configuration. Model design is difficulty in general because the development of system models is an activity usually disconnected from the real system (which lead to inaccuracy) and, of course, because of modeling representation language limitations (which add complexity). In the case of space systems, there are very often several additional issues: multiple sources of information (often specified a very different levels of abstraction), nobody often really in control of the "whole" model (many specialists in control of different aspects of the problem), many different rationales behind rules and constraints (physical, operational, common usage, even politics some times!), multiple users that aim at using the system with different views and needs (often conflicting with each other). Given the generality of the issues taken into account, I would say that the challenge that this paper discusses is central to the fielding of planners in general, not only in the case of space domains.

## Approach

The idea proposed in this paper is to support model design by connecting tightly the "real world" being modeled with the abstract description being designed by developing the domain model in a loop with an application simulator. The paper describes an hypothetical Interactive Model Development Environment (IMDE) aimed at simplifying validation of models within the development cycle, to make modeling for space mission planning more efficient. The authors goes a bit further, wondering an integration between the planner and the simulator aimed at helping automate model development and validation. Given that the paper presents only a high level design and generic use cases of the proposed architecture, but at the same time discusses some existing potential enabling technologies for implementing it, the IMDE in my opinion can be seen as a potential road map for improving automated mission planners, for a space applications in this case, but also more in general.

The proposed design is made of many modules, but the most critical in my opinion are those implementing the connection between the planner and the simulator: the "Abstraction" and "Refinement" engines, the "Validator" and the "Fixer". The Refinement Engine transforms a plan into simulator input. The Abstraction Engine transforms simulator output into an 'as-executed plan' to be compared with the original plan. A "Validator" module compares the two plans and assesses discrepancies and a "Fixer" module analyzes mismatches between the plan and simulation output and identifies model and abstraction elements for possible change. The "Validator" module is foreseen also able to take as input any constraints not explicitly checked in the simulator (e.g., flight rules) that are part of the planning model.

These modules are critical in my opinion because this translation back and forth hides not trivial issues (some of them discussed at a high level in the paper). The planner and the simulator work on two different models, the first being build and the second supposed unknown (either implicit or explicit, observable or not, depending on the technology of the simulator). The problem is to align the two models to control in the proper way the physical system modeled by the simulator, taking into account only the outcome of the simulation (to be translated back).

## Conclusions

This paper constitutes an interesting contribution to KE for planning and scheduling. The challenges proposed in the paper are central to the modeling for real world domains and to the fielding of P&S technologies.

In my opinion two points are worth to be discussed: the pros and cons of the approach based on simulation with respect to formal model V&V and the focus of the approach on the modeling process more than on the modeling language.

The integration of formal verification techniques to model based planning systems brings evident advantages on applicative fields where formal models of physical systems are available. I believe this is the case of space domains, at least at some levels (like flight dynamic management for instance) and in fact there is a significant amount of work already done in this directions for space applications. On the other side, on some application fields, like autonomous robotics for instance, the approach based on simulation in already in use with obvious advantages. In my opinion it would be interesting to discuss for which type of problems the proposed approach will bring advantages and where instead already existing approaches are probably more convenient. Or perhaps they can be complementary?

The second point is related to the focus of the approach. Modeling representation language limitations add complexity and force inelegant workarounds or rough approximations as pointed out in the paper. There is a quite an amount of work already done on KE environments and languages (NDDL and ANML at NASA for instance) aimed at bringing "the modeler close to the system" by giving him the chance of representing interesting features of the domain in a way suitable for being processed and reasoned upon. These works are focused on working on the language to reduce the distance between the modeling tools and the real world. The approach proposed here makes no assumptions on the modeling language, being based on a general domain independent planning language, and aims at supporting the modeling process by bridging the gap between the system and the modeler through the use of a simulator. Does the use of a modeling language more semantically close to the simulator make sense or are you suggesting to go on a different direction?