



HAL
open science

Towards Model Synchronization between system architecture design and Multiphysics simulation

Aroua Berriche, Faïda Mhenni, Abdelfattah Mlika, Jean-Yves Choley

► To cite this version:

Aroua Berriche, Faïda Mhenni, Abdelfattah Mlika, Jean-Yves Choley. Towards Model Synchronization between system architecture design and Multiphysics simulation. Design and Modeling of Mechanical Systems CMSM'3019, Mar 2019, Hammamet, Tunisia. hal-02298956

HAL Id: hal-02298956

<https://hal.science/hal-02298956>

Submitted on 27 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Model Synchronization between system architecture design and Multiphysics simulation

Aroua Berriche^{a,b}, Faïda Mhenni^a, Abdelfattah Mlika^b, Jean-Yves Choley^a

^a Quartz Laboratory, Supmeca, 93400 Saint-Ouen, France, aroua.berriche@supmeca.fr;
faida.mhenni@supmeca.fr; jean-yves.choley@supmeca.fr.

^b LMS Laboratory, ENISo, 4054 Sousse, Tunisie, abdelfattah.mlika@gmail.com

Abstract – The design process of complex systems involves the use of different languages and tools to model and simulate systems structures and behaviors. Different models are thus used such as system models (using high level modeling tools such as SysML), multiphysics models (Modelica) and safety models (Altarica) and so on. The main consequence is a high risk of inconsistency between the different views of the system. In this context, we need to ensure the exchange between the different actors interacting in the development of a complex system and to verify the consistency between the different multi-view modeling systems. In this article, we have proposed a model synchronization methodology to detect inconsistencies between the different views of a system. This Method is composed of three steps: first, the abstraction of entry models to a common representation, second the comparison process which permits to identify the inconsistencies between different views of a system and finally the concretization that allows to manage inconsistencies. This approach is illustrated with a case study from the automotive industry, which verifies the effectiveness of this proposal to improve the cooperation between designers developing a complex system.

Keywords: Multi-view Modeling / SysML / Modelica / Consistency management/ Mechatronic systems.

1 Introduction

Developing mechatronic systems requires the consolidation of models from a variety of domains such as mechanics, electronics and software engineering. These models are often created using different formalisms and by different designers having different viewpoints on the overall system. To manage consistency between different models of a mechatronic system, this work proposes a methodology to detect and manage differences and inconsistencies between different models of a given complex system.

The remainder of the paper is organized as follows. Section 2 describes similar works that deal with the assuring consistency between models. Section 3 gives a succinct presentation of the proposed methodology. Section 4 presents the case study of the Electronic Throttle Body (ETB). Conclusion is given in the last section.

2 Related work

Recently, considerable attention has been paid to the problem of consistency of models elaborated with different languages involved in the design process of a mechatronic system. Researches explored several clues to solve the problem.

Some works considered the use of profiles or SysML (System Modeling Language) extensions to enrich SysML with some semantics of other tools and then to transform system (enriched) models into the other language or tool. The two most well-known profiles to make a link between SysML and Modelica are SysML4Modelica [1] and ModelicaML [2].

There also exists a research work proposing federative approaches [3] that attempts to establish relationships between elements of models with different concerns. In [4] a framework is proposed to implement the federation approach by using the powerful and rich semantics of the SysML language.

In this paper, we have selected two particular but representative modeling tools to apply our cooperative approach: SysML for systems engineering and Modelica for modeling and simulation.

3 Methodology

Our methodology consists in identifying, detecting and managing differences and inconsistencies in the process of Model Driven Engineering MDE. This automatic process is based on three phases: abstraction, comparison and concretization. These three phases will be described in the following.

3.1 Abstraction

The first phase includes the representation of entry models (SysML, Modelica) in a common formalism using graph theory [5]. We assume that the abstraction applies to model-to-model transformation [6].

3.2 Comparison

In order to identify differences and inconsistencies between abstracted models, a subgraph isomorphism algorithm must be developed inspired from [7]. This algorithm execute three principal activities:

- Search for component mapping between the two abstracted models.

- Executes a comparison process based on graph properties (label node, number of entering and leaving edges for every nodes...).

- detect the differences (inconsistencies) between the two abstracted models.

3.3 Concretization

The last phase, allows refining the source models using abstracted models. This latter will be associated with operations proposed by the designers to manage consistency between models. This phase will be implemented using model-to-model transformation technique. As a result, we obtain consistent information between the different views.

4 Case study

This methodology is illustrated in a case study from the automotive industry. The Electronic Throttle Body (ETB), is an important actuator which control the air supply to the engine which varies the engine torque output .The internal design architecture of the ETB is shown in Figure 2.

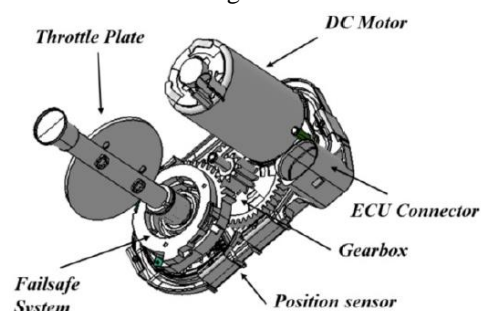


Figure 1. Electronic Throttle architecture

We will study the consistency between SysML and two Modelica models elaborated with different team (T1, T2) for different concerns.

4.1 Abstraction

We transform automatically the three models to topological graphs, as a result we obtain the three graphs represented in the following figure 2, 3, 4.

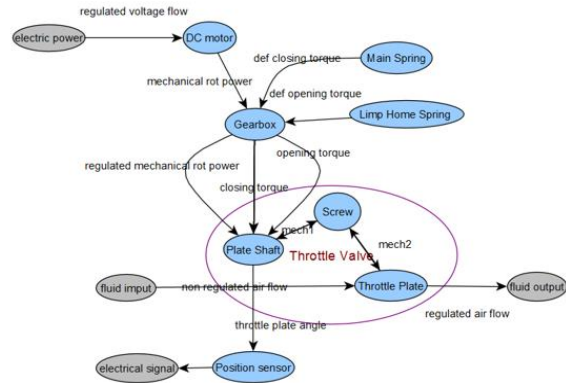


Figure 2. SysML-IBD topological graph

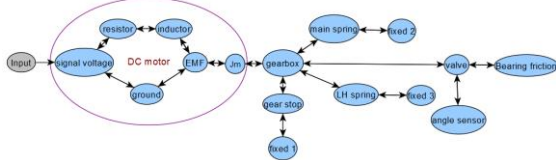


Figure 3. Modelica topological graph (T1)

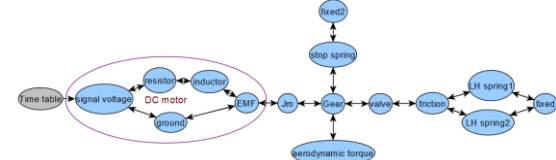


Figure 4. Modelica topological graph (T2)

4.2 Comparison

We compare the models using their topological graph representation.

We represent in the Table 1 the results of the comparison of nodes between different models.

Table 1. Comparison of nodes

SysML	Modelica		Differences	Inconsistencies
	Team1	Team2		
input	input	input	Type	-
DC motor	DC motor	DC Motor	-	Decomposition
-	-	Jm	-	-
Gearbox	Gearbox	Gearbox	-	-
Main spring	Main spring	Main spring	-	-
-	Fixed2	Fixed2/3	Language specification	-
LH spring	LH spring	LH spring	-	-
-	Fixed3	Fixed2/3	Language specification	-
Valve	Valve	Valve	Decomposition	-
-	Friction	Friction	concern	-
Position sensor	Position sensor	-	concern	-
Fluid input	-	-	concern	-
Fluid output	-	-	concern	-
Output	-	-	concern	-
-	Gear stop	Gear stop	concern	-
-	Fixed1	Fixed1	Language specification	-
-	-	Aerodynamic torque	concern	-

In the same way, we obtained a table that contains the results of edges comparison and that is not included here for space saving.

4.3 Concretization

This step, allows to correct entry models with chosen compromises proposed by designers taking into account the results of the comparison.

We represent in Figures 5 and 6 the new topological graphs with corrections in red to manage inconsistencies.

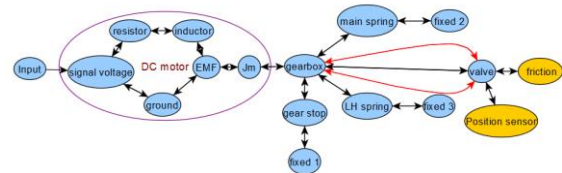


Figure 5. Modelica topological graph refined (T1)

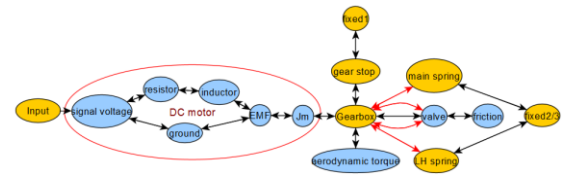


Figure 6. Modelica topological graph refined (T2)

5 Conclusion

The paper proposed a methodology to evaluate consistency of multi-view modeling approach for complex system. It has been shown how the proposed approach cover all phases of early detection of inconsistency problem of mechatronic systems. The first step transform the different views of a system in a common representation. The second phase permit to define the mapping of components between models and compare their structure. The final phase consisted of managing inconsistencies to provide consistent information between the viewpoints.

In order to consolidate our work, a case study on the Electronic Throttle Body (ETB) was given.

6 References

- [1] C. J. Paredis, Yves Bernard, Roger M Burkhart, Hans-Peter de Koning, Sanford Friedenthal, Peter Fritzson « An Overview of the SysML-Modelica Transformation Specification », in *INCOSE International Symposium*, 2010, p. 709–722.
- [2] A. Pop, D. Akhvediani, and P. Fritzson, « Towards Unified System Modeling with the ModelicaML UML Profile », 2007.
- [4] K. Thramboulidis, « Overcoming mechatronic design challenges: the 3+ 1 SysML-view model », *Comput. Sci. Technol. Int. J.*, p. 6–14, 2013.
- [5] F. Havet, « Théorie des Graphes », 2008.
- [6] T. Mens et P. Van Gorp, « A Taxonomy of Model Transformation », *Electron. Notes Theor. Comput. Sci.*, vol. 152, p. 125-142, 2006.
- [7] M. Asiler et A. Yazıcı, « BB-Graph: A New Subgraph Isomorphism Algorithm for Efficiently Querying Big Graph Databases », 2017.