A Conceptual Framework for Consistency Management in Model-Based Systems Engineering

Sebastian J. I. Herzig, Dr. Markus Brandstätter (Advisor), Christiaan J. J. Paredis (Advisor)

Georgia Institute of Technology

- **Problem Statement**
  - In Model-Based Systems Engineering (MBSE) formal models are used as the means to specify systems
  - Stakeholders from various domains are involved in the design and development of a complex system
  - Every stakeholder has a different view on the specification (e.g., electrical, software, mechanics)

- **Research Challenges**
  - To the best of knowledge of the authors consistency management in MBSE has never been studied at a fundamental level – there are, however, a few ad-hoc solutions mentioned in the literature

- **Approach**
  - Consistency issues and management methods were identified by asking the questions: How is design done? How are models created? How are consistency issues currently being treated?

- **Consistency in Model-Based Systems Engineering**
  - **Decision-Based Engineering Design**
    - A single decision maker has ideas (concepts), beliefs and preferences
    - These ideas (concepts), beliefs and preferences are abstracted using models
      - Beliefs are predictions about future events and are informed by scientific data
      - Rational beliefs are consistent with scientific data
      - Preferences are rational only if they have transitive qualities and therefore show an explicit ordering
  - **Models and Internal Consistency**
    - In order for models to be consistent they need to conform to modeling languages
    - These modeling languages need to map to formalisms from mathematics semantically
    - Transformations (based on, e.g., rules) are used to keep different models, and hence views, consistent across each other
  - **The Dilemma of External Consistency**
    - Scientific data is based on observations of nature
    - Nature and natural phenomena cannot be described with precision, hence checking for such external inconsistencies is impossible
    - Raises question whether validation is possible at all

Model-Based Systems Engineering Center

- **Formal Systems**
  - **Formal Languages**
    - Formal systems contain formal languages
    - A language L is defined as a set of formulas: $L = \{ w_1, w_2, \ldots, w_n \}$
    - The formulas or words of a language are derived using a formal grammar $G$ that specifies axioms, inference rules and the symbols that may be used to construct words
    - A formula is well-formed if it follows the syntax of the language and is therefore an element of L
    - A formula or word is a theorem of L if it is well-formed and satisfies certain syntactical and logical conditions
    - A semantic meaning may be given to every well-formed formula

- **Modeling Languages**
  - In essence the same as formal languages
  - Models are words or formulas
  - Meta-model is equivalent to grammar
  - Differentiation between abstract ("the essence") and concrete syntax also common in textual modeling languages

- **Relation to State of the Art**
  - **Formalization of Languages**
    - UML and SysML are semi-formal modeling languages - current work includes formalization of these using description logic
    - Most of this work comes from software engineering
  - **Consistency Across Views and Models**
    - Current research focuses on using transformations (TGG) to ensure consistency across views and models
    - Two possible approaches mentioned in related literature:
      - Transformation from and to a central model system
      - Transformation across domain-specific models
  - **Domain-Specific Modeling & Consistency**
    - Consistency checking across models from multiple domains requires formalism
    - Current approaches utilize rules and meta-models
    - Generic, domain- and application-specific concepts needed
    - Illustrative example:
      - Electrical schematic shows electrical connection
      - Hence, the mechanical view must contain a physical connection (of a conductive material)

- **Future Work**
  - Develop a test case to illustrate the different kinds of consistency issues: camera payload of a picosatellite
  - Implement consistency checking methods in tools
  - At what point in the design process should a consistency check be performed? What part of the system specification should be examined?

- **Academic Collaborators**
  - Georgia Institute of Technology
  - Model-Based Systems Engineering Center
  - Technische Universität München
  - Institute of Aeronautics
  - KTH Royal Institute of Technology
  - Department of Machine Design

- **Publications**