Challenges in modeling an automatic gear box in Modelica

A. Mehlhase[1] and A. Ehrich[2]

[1]TU Berlin, a.mehlhase@tu-berlin.de
[2]TU Berlin, ehribcpc@mailbox.tu-berlin.de

Modeling and simulation of a physical system which is under development is a common procedure. But common simulation environments such as Simulink, Dymola and so forth still exhibit limitations. In particular it is not possible to change the set of the equations of a model during a simulation run. This is, however, an important feature in many applications. For example, when simulating an automatic gear box it is necessary to change the equation system for the different gears. In the work of [1] such a gear box model was implemented in FORTRAN. The disadvantage of this way of proceeding is, however, that the code is not reusable for other models, and FORTRAN is not a common simulation environment in the industry (bycontrast to environments like Dymola and Simulink, which are widely used). With today's state of technology it is feasible to simulate systems like the gear box example in a language like Modelica, which is an acausal, multiphysics modeling language. But also Modelica has the deficit of not supporting a change of the equation system during a simulation run. The simulation tool MOSILAB [2] and the experimental language SOL [3] can handle variable-structure models but are restricted experimental systems and therefore are not yet usable as standard modeling tools.
We therefore use a Python framework (called 'Strucs'), which builds around exisitng standard modeling tools such as Dymola. Strucs allows the user to create variable-structure models. This means that a model can run through different 'modes', where each mode effectively is a model with its own set of equations. When switching from one mode to the next one, the new mode is initialized by the end values of the old mode.
With Strucs we are able to reimplement the gear box and use a common simulation language such as Modelica for our models. Each gear of our box is separately implemented in Modelica. The models are composed from different components such as breaks, clutches and planet sets. In Strucs the shifting through the gears is defined by using a Python template, which specifies the switching conditions, the initialization of the new modes and the solvers for each simulation. We use the same switching conditions and initialization as in [1]. With the help of Strucs we are now able to model and simulate the gear box in a common simulation environment such as Dymola. In the original FORTRAN model different solvers were used to simulate the model; now it is possible to use the existing Dymola solvers. With the framework it is also possible to use the open source simulation environment OpenModelica to simulate the gear box. Therefore the new model is much easier to handle and to reuse than the old and specialized Fortran version.
In the original work the FORTRAN models simulation results where compared to the simulation results of a reference system called ASIM. It turns out that our model does not have the exact same results as the FORTAN model, but we are as close to the ASIM results as the Fortran model was. The big advantage of the new model is that each gear model and its components can easily be reused for new gear box models. The specific Python template for the gear box can easily be reused. Furthermore the models can be used in any simulation environment that supports Modelica. For the time being only Dymola and OpenModelica are integrated into Strucs, but other environments can easily be added.

References

[1] P. Hamann, *Modellierung und Simulation von realen Planetengetrieben*. Diplom Thesis, TU Berlin, Germany, 2003

[2] C. Nytsch-Geusen, T. Ernst, A. Nordwig & et al., *MOSILAB: Development of a Modelica based generic simulation tool supporting model structural dynamics*. Proceedings of the 4th International Modelica Conference, Hamburg, TU Hamburg-Harburg, 2005

[3] D. Zimmer, *Equation-Based Modeling of Variable-Structure Systems*. PhD Thesis, Swiss Federal Institute of Technology, 2010