Model-Driven Innovation

Many engineering organizations today are facing major challenges when designing and delivering new products. Time to market and cost control are critical in an increasingly competitive and global market. At the same time, system complexity is rising to accommodate growing customer expectations and increased regulatory constraints.

As a result of these challenges, key and sometimes fundamental design issues are often discovered late in the product development process. This results in budget overruns and project delays, especially if issues are not discovered until the hardware prototyping or system integration stages. Moreover, once a system has been delivered and installed, failures may result in disrupted production and emergency on-site service calls, which can become very expensive and hurt a company's profits.

Many of these issues come about because the design of a complex system involves multiple engineering disciplines, each focused on one aspect of the system, using dedicated tools and methods to determine if their design will comply with the specifications for that aspect of the system. Problems arise when these subsystems are integrated during prototyping or even during final assembly.

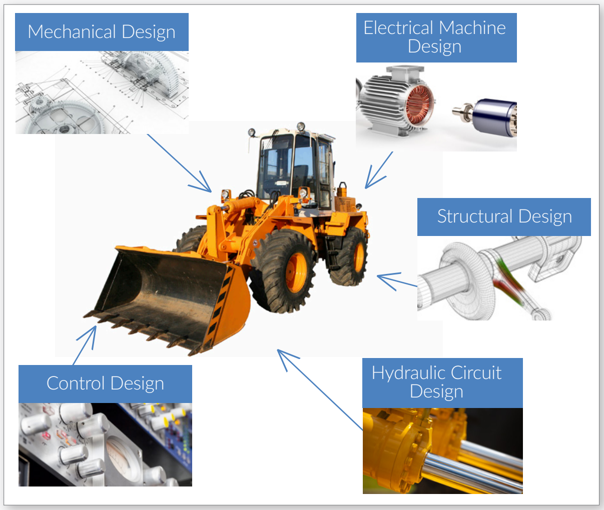

Figure 1. Design issues are often brought to light during the integration of subsystems from different domains.

In order to address these issues, a growing number of organizations are turning towards a **Model-Driven Innovation** process – an approach that makes a multidomain, system-level model the core of their design activities. Using system-level modeling to develop virtual prototypes of their systems, engineers can see how the individual subsystems work together.

They can consider the mechanical, electrical, thermal, and control aspects of the project in unison, to identify weaknesses in the design, and make corrections earlier in the development cycle. Ultimately, such an approach leads to increased product quality, shortens time to market, and contributes to controlling costs by lowering iteration times and reducing the need for hardware prototypes.

Successful implementation of a model-driven innovation process requires robust tools that enable engineers to develop and test high-fidelity virtual prototypes of their system - tools like MapleSim, the multidomain system modeling and simulation tool from Maplesoft. Using such a tool, engineers can successfully address issues early on in the design process that they would otherwise not have detected until much later. For example, a multidomain system model can be used to look at the thermal strain on the battery system of a robot, resulting from the dynamic loads on its joints. Such strains cannot be detected when the battery system and mechanism design are investigated separately.

As mentioned, at the core of a Model-Driven Innovation process is a multidomain, system-level model that describes the dynamic behavior of a design and incorporates all relevant subsystems.

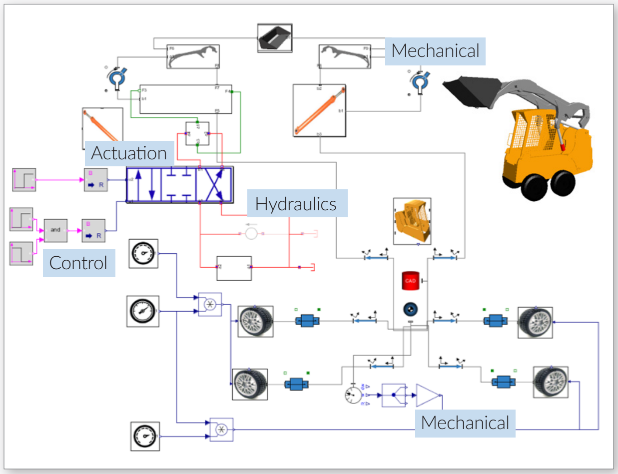

Figure 2. A multidomain system-level model lies at the core of the Model-Driven Innovation process.

The level of detail required for such a system-level model is typically much less than for traditional, single-domain simulations that can be performed, for example, with finite element analysis (FEA). That is because the overall system behavior and how different subsystems interact are now of interest, and properties like strain and stress are only required at a lumped-parameter level. However, system-level simulation results provide forces and loads within the system that can then be used as inputs for a more detailed FEA analysis of a particular component.

This also means that it is critical to understand what questions need to be answered, in order to build a system-level model that will yield those answers. Depending on requirements, such a model is assembled from a variety of sources. Rich libraries of prebuilt components from domains such as electrical, mechanical, hydraulics, thermal, and control are readily available. Industry standards like Modelica and the Functional Mock-up Interface (FMI) provide an ecosystem for component libraries and for reusing models across different tools. Models typically also incorporate empirical data as well as results from detailed FEA studies.

Once such a system-level multidomain model has been developed, what can now be done that was not possible before?

First, a virtual prototype of the design can be studied, and the dynamic behavior of the multidomain system simulated. Probes can be inserted, to show forces, loads, heat, etc., at any point of interest in the system, across the duration of the simulation. Simulation results can be used to estimate overall system behavior, which in turn can be used to work through scenarios with varying parameters and configurations.

But that is just the start. Beyond simulation, the design can now be explored, to obtain deep insight into the system. In the case of Maplesoft tools, MapleSim is tightly coupled with Maple, the world's most advanced symbolic computation engine, which enables users to analyze models at a deeper level. Interactive analysis tools driven by the model can be set up, which allow parameters to be tuned and the impact on the system

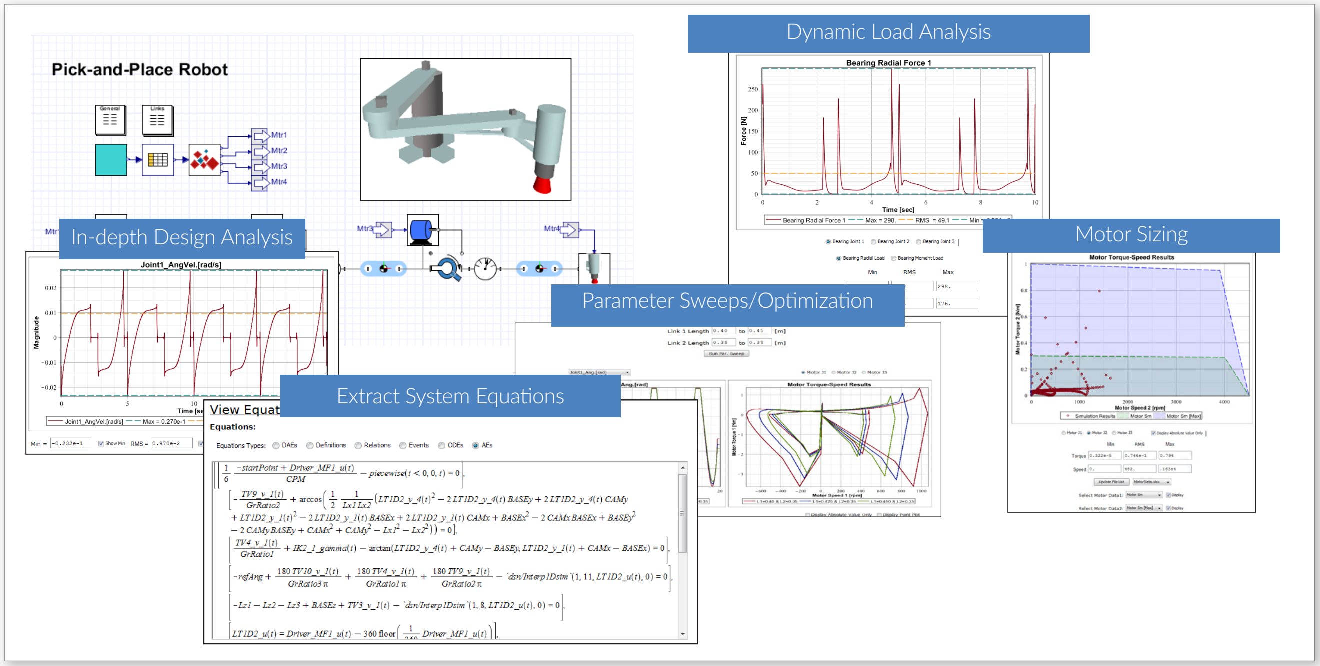

Figure 3. MDI accelerates system-level analysis and optimization

behavior seen instantly while capturing the knowledge that went into the analysis. Optimization and Monte-Carlo techniques can be applied to control a large number of system parameters. It is also possible to access the underlying system equations to solve inverse kinematic and inverse dynamic problems, which are useful for advanced control of mechatronic systems.

Finally, a system-level model facilitates verification and control design. The controller can now be designed and tested, well before a hardware prototype of the system is built. A virtual prototype derived from the system-level model can be substituted via real-time simulation, so that fewer surprises will be discovered late in the process.

Such a virtual prototype of the system can also be used for operator training, before the hardware is available or when the final deployment environment is critical to the operator experience. For example, when developing a robotic space arm, the simulation environment can emulate the effects of different gravitational fields, enabling the user to become familiar with adapting the way they operate the joystick controller.

If you are seeing system complexities arise, threatening your ability to control costs, produce high quality designs, and get your products to market quickly, then a system-level approach can bring value to your organization. We have seen these techniques make a real difference for many customers who now consider Model-Driven Innovation a core part of their process, and indeed a competitive advantage.

Maplesoft
Mathematics • Modeling • Simulation
A Cybernet Group Company

www.maplesoft.com | info@maplesoft.com
Toll-free: (US & Canada) 1-800-267-6583 | Direct:1-519-747-2373