DOI: 10.13009/EUCASS2022-6101

Synthesis of landing gear simulation models based on architectures and templates

Elias Allegaert, Yves Lemmens* and Emre Ongut**
**Siemens Digital Industries Software*
Interleuvenlaan 68, 3001 Leuven, Belgium

Abstract

A methodology and software tool are presented to speed up the process of creating CAD models, FE simulations, and multibody simulations by using template models. This enables MBSE for systems that require analyses on 3D geometry. Templates make it possible to reuse simulation models and use software to automatically assemble them. This leads to a large reduction in modeling effort when there are many configurations that need to be modeled and optimized. Moreover, the formalized approach improves the traceability of design choices and facilitates documentation. The technology is employed to synthesize the FE analysis and multibody simulation models needed for landing gear design.

1. Introduction

Reduced design cycles are an engineering challenge in the aerospace industry. Simulation-driven design puts the simulation engineer under pressure to analyze and report on the newest design iteration. Simulation lag incentivizes to reuse proven designs and avoids exploring new designs to minimize time, risk, and cost. Therefore, a reduction in the time needed to create simulation models will reduce the overall time spent on the simulation of a design iteration. As a consequence, the design process will be more streamlined and design iterations can follow up faster. This is the aim of the methodology and the software tool that is presented in this paper.

Today, engineering design theories focus on many important aspects of designing complex products. Therefore, research on engineering design makes use of state of the art technology and new insights into many scientific fields. For example, innovations in computer science and software engineering have a big impact on aerospace engineering. As an example, educational aerospace and computer science curricula are becoming more connected [1][2]. The work presented in this paper fits this view and builds further on work related to systems architecting [3] and the importance of modularity [4]. Lessons from software engineering have taught that management of interfaces between systems is crucial [5]. In aerospace engineering, the concept of the interface is known as contract-based design [6].

The presented methodology builds further on the Model-Based Systems Engineering (MBSE) paradigm which is a framework to improve the design process by providing tools to handle the increasing complexity of systems. In section 2 a brief view is cast on MBSE and simulation model synthesis. It is explained how the template-based synthesis of CAD and simulation models fits in this framework. In section 3, the software tool is presented. In section 4, a couple of examples of use case scenarios are presented. They demonstrate the various ways in which landing gear simulation data can be obtained from the synthesized models.

2. Simulation model synthesis

2.1 The need for MBSE

As our civilization climbs the technological ladder, our systems, aerospace and defense systems in particular, are more complex than their predecessors. This puts the design methodologies under pressure because more than once, this led to schedule delays and cost overruns [7]. The growth in complexity can be seen in the rise of the number of subsystems, interfaces, and lines of source code [8].

In essence, the strategy of MBSE is to make it easier for people to collaborate by reducing the perceived system complexity. Domain experts have access to all needed information, but they should only expose the relevant

DOI: 10.13009/EUCASS2022-6101

information to non-experts through interfaces. This is a proven technique in software engineering of which object-oriented programming is probably the most obvious example. A consequence is that the system and its engineering activities are easier to formalize. This makes it easier to structure the design process and design variants resulting in the engineering process becoming more machine-interpretable. This leads to design automation because, with a formalized definition, it is easier to express the design as a tree of decisions that can be simulated to find the most performant variant.

MBSE is often explained with the help of the "V" model, see Figure 1. One of the most important concepts of the MBSE strategy is the system or logical architecture. Typically, only a few architectures are considered and worked out. One way to obtain system architectures is by decomposing existing products. Another way is to analyse the functional requirements to eventually obtain a list of components that can satisfy those functional requirements. A software solution exists for automatic system architecture generation [9][10]. This is done by converting the problem of finding a valid architecture into a mathematical formulation. In this way, the entire design space can be formally described and searched efficiently. The next step is to evaluate the performance of each architecture by designing and sizing the individual components. Validation is performed by integrating all the components into a system and performing system simulations. The synthesis of these system simulations is performed by dedicated software, usually closely coupled with a specific simulation tool.

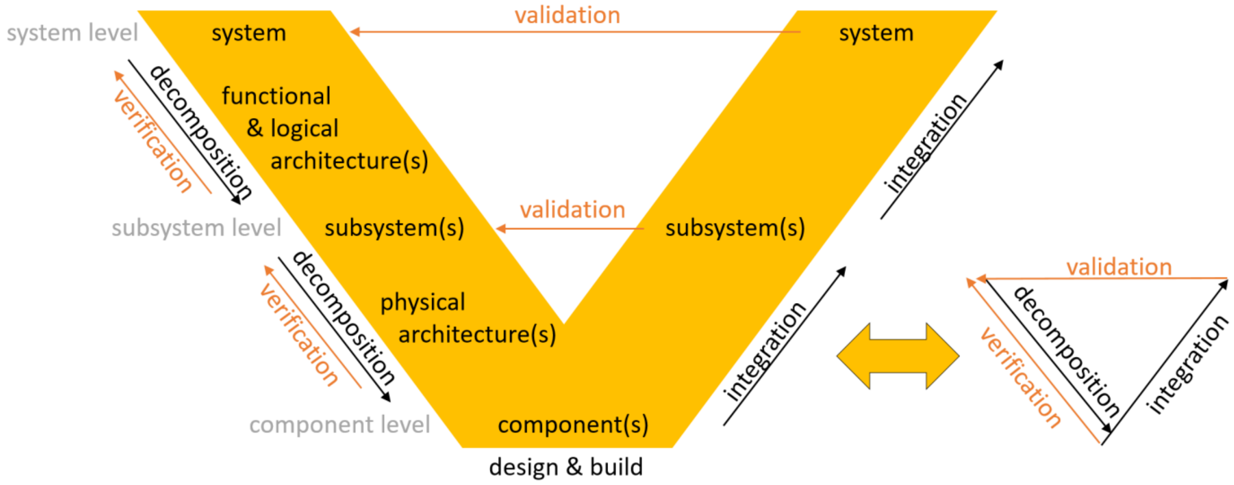

Figure 1: The V model of MBSE. The principal strategy is decomposition and integration, a.k.a. synthesis. Figure from [11].

MBSE is often advocated for the purpose of front loading the design. Indeed, MBSE goes hand in hand with bond-graph simulation models that serve as a first approximation for the behavior of components in a system. These bond-graph models are also called 0D models because there are no spatial dimensions taken into account. The geometry is abstracted away and compressed into a parameter of the system's equations. In the early stages of design, there is no CAD model available and only global phenomena are modeled in the Ordinary Differential Equations (ODE) or Differentiable Algebraic Equations (DAE). However, the goal of the presented methodology is to make it possible to actually have a CAD model as early as possible in the design. How this is done is explained in the next section.

2.2 MBSE with CAD-based simulation models

Although the MBSE approach is gaining traction in the aerospace industry and success stories get reported, its adoption is slow. One of the reasons might be that there are few examples of 3D systems being designed and optimized using an MBSE approach. Most MBSE-related papers focus on systems simulation, i.e. 1D simulation based on a bond-graph representation of the physics [12]. There exist frameworks where CAD and CAE models can be included. However, the CAD geometry is not able to adapt to the system it needs to fit in [13][14] or the methodology only provides a language to define CAD parameters without an implementation to automatically create the assemblies and simulation models [15]. To summarize: it is a well-known weakness of many existing MBSE frameworks and tools that they only provide a language to describe systems but that they lack the software implementations to perform manipulations on the models and most importantly synthesize the system models from individual component models. A software environment that provides all these functionalities for 0D models is Simcenter Studio [16][17] and Simcenter System Architect.

DOI: 10.13009/EUCASS2022-6101

Besides the fact that 0D models solve quickly, they are also a compact representation of the model and therefore they are easy to comprehend. The compactness of the models makes them very suitable to build a library of models that can be used as building blocks for large systems. With a relatively small number of component models, a large variety of systems can be modeled. It is in this explosion of combinations that a specific language and software tool such as Simcenter Studio is needed to explore all possible combinations. How to bring CAD models and their derived Finite Element (FE) models and multibody simulation models in this framework?

First, we have to look at the CAD and simulation models themselves. They will need to fit in the system modelling language. This language defines that a component has ports to exchange information with other components. For 0D models, these ports exchange physical quantities such as position, velocity, pressure, … These quantities are also called flow, effort and data [18]. 3D models, as geometry-based models will be denoted in this paper, only exchange geometry such as points, axes and curves when they are synthesized. Of course, after solving, there are also forces and moments exchanged between components. However, simulation model synthesis only needs to consider the geometric dependencies.

Model templates use the shared geometrical elements to build on and provide new elements for other models. Additionally, a group of geometrical elements can be assigned to represent a physical joint. For example, a point and an axis can be one of the inputs of a model on which other geometry is constructed, see Figure 2. When that CAD model is the basis for a FE model, that point and the axis are used to create a joint with one rotational degree of freedom denoted by the axis. It was found that interfaces between components should be defined as generally as possible to increase the reuse of templates. Similar to 0D models, the ports have a causality, i.e. there are input and output ports.

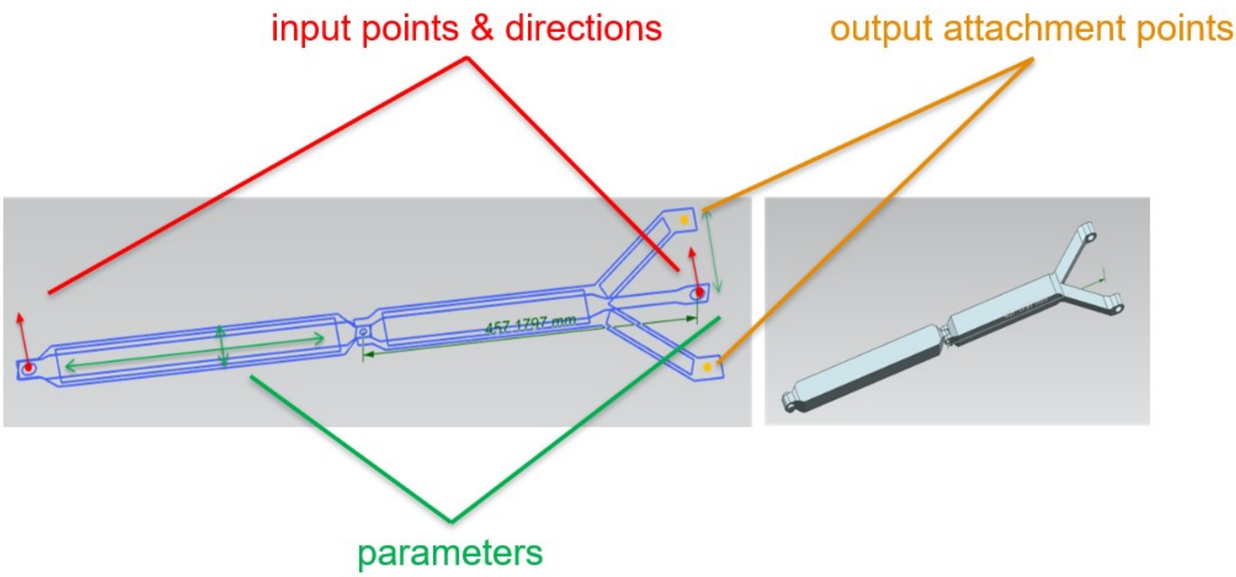

Figure 2: Sketches on which the extrusions are built which are visualized on the right. Red: input geometry, Green: parameters, Orange: output geometry which cannot be directly controlled because it is the result of input geometry and parameters. The orange points are used to make a joint between the brace and the airframe.

The parametrization of the CAD model is the main difficulty of using CAD templates. It is important to spend enough time on creating a robust parametric model. This can be done by keeping some design principles in mind that are well known by professional CAD modelers [19]. A Knowledge-Based Engineering approach might have more tools to mitigate the problem of breaking geometry. However, KBE relies on writing code to generate the geometry. Many engineers experience reading another's code and the debugging process negatively [20][21]. The work described in this paper uses a template-based approach to synthesize the simulation models. The main advantage is that the expert modeler can use the Graphical User Interface (GUI) of the CAD program instead of programming in a Integrated Development Environment (IDE).

Considering template-based model synthesis, the generation of system architectures, and KBE, this suggests that more innovations are to be made when treating engineering as a discipline of information processing and part of computer science. Literature shows that this is an active field of research: graph-based approaches [22] and ontologies [23][24].

DOI: 10.13009/EUCASS2022-6101

2.3 The synthesis process

The previous section has shown how a CAD model can be made to fit in the MBSE framework with a model template library and that this way of modeling allows them to be synthesized automatically. A synthesis tool is needed that can assemble the simulation models. Prototyping of this tool started in the CONCEVAL project [25] and has further been developed in the Soft Landings project. The tool can synthesize CAD, FE models, and multibody simulation models in Simcenter 3D.

The main tasks of the synthesis tool are:

- **Cloning** template models from the model library. The cloned models are not connected at this point.
- **Creating parent file.** A new parent part is created to which all cloned component models are added. The child components are still stacked on top of each other.
- **Linking geometrical elements** of the CAD model through WAVE links in Siemens NX. WAVE links are a means of associatively linking geometry from one part into another. These can be bodies, faces, curves, sketches, routing objects, etc. which can then be used in the receiving part to design from or refer to. Now the components are no longer stacked on each other but correctly connected by WAVE links.
- **Parameter updates** set the correct parameter values. Parameters change the geometry but not the "hard points" that were exchanged by the WAVE links.
- **Adding FE model constraints and multibody model joints.** Although the geometry appears connected, the components are not physically linked during a simulation. The synthesis tool adds joints between components.
- **Copying or reworking simulation elements.** Some simulation elements in the FE template model or multibody simulation template model need to be copied over from the component level to the parent level. For example, there is only one overarching FE analysis file so the loads from the templates need to be copied over to the system-level analysis file.

The tasks described above indicate that there are a lot of practicalities that need to be taken care of depending on the simulation software platform. This is immediately one of the main reasons why MBSE frameworks lack an implementation to synthesize the simulation models.

3. Landing gear synthesis tool

The presented methodology was used for the creation of a tool that enables a non-expert user to generate the simulation models for elementary landing gear simulations. There is a dedicated Graphical User Interface (GUI), see Figure 3, that guides the user in choosing a landing gear architecture, assigning model templates, and configuring the simulation. Landing gear models are synthesized for CAD, Finite Element Analysis, and multibody simulation in Simcenter 3D. A screenshot of the tool can be seen in Figure 5. The contents of the libraries can be extended and modified by adding or removing template models and text files (in JSON format). Even better, the tool does not limit itself to the synthesis of landing gear models because all design-specific information such as system architecture and template models is specified in JSON files. Therefore, virtually any scenario can be synthesized if it can be expressed in parametrized template models. The user is guided through the tool in a logical manner by moving through tabs.

DOI: 10.13009/EUCASS2022-6101

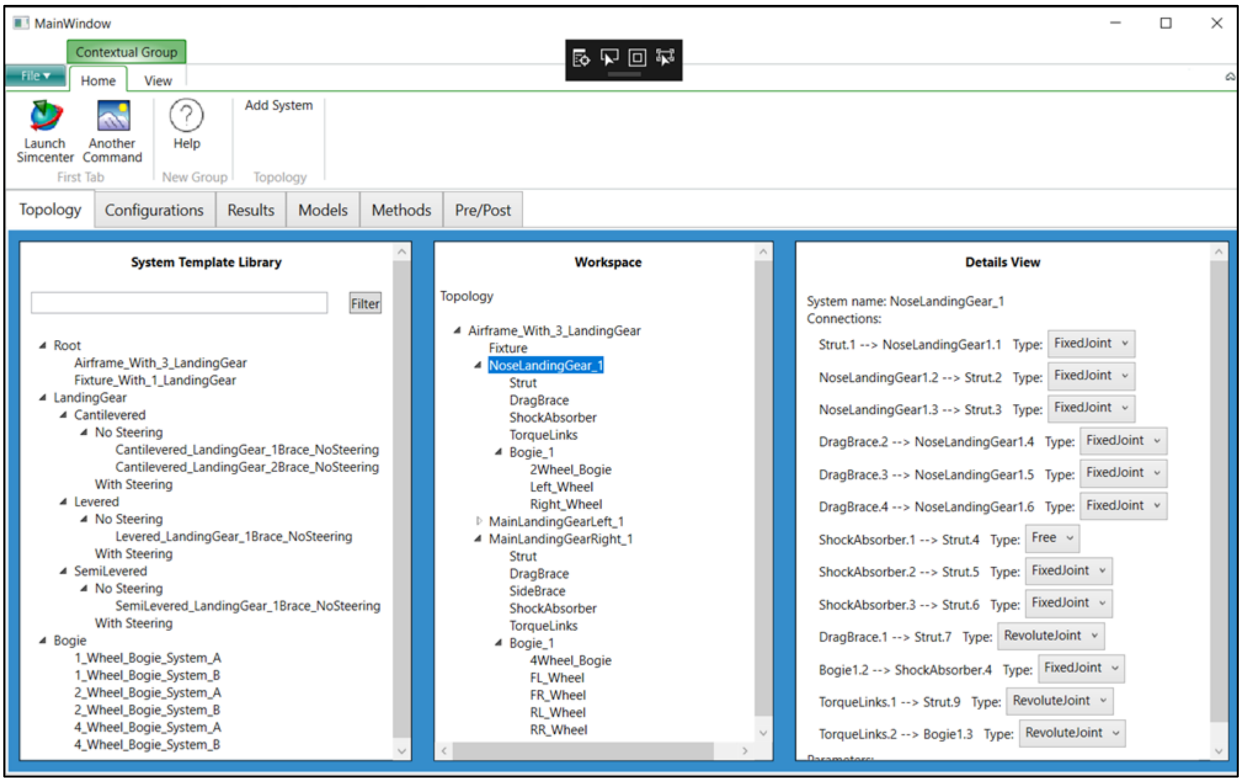

Figure 3: Screenshot of the GUI of the landing gear synthesis tool. Here it shows the selection of a landing gear topology on the left pane, the component tree in the middle pane, and details of the selected component on the right pane.

In the first tab, a system topology is built dynamically by dragging and dropping (sub-)system definitions onto the workspace. Figure 4 shows 3 top-level architectures that can be created for a landing gear. Initially, the bogie component is left open for the user to decide to drag a bogie with multiple wheels or a fork with one or two wheels. Note that on the highest level a choice needs to be made between an architecture of a mounting with a single landing gear or an aircraft with multiple landing gear.

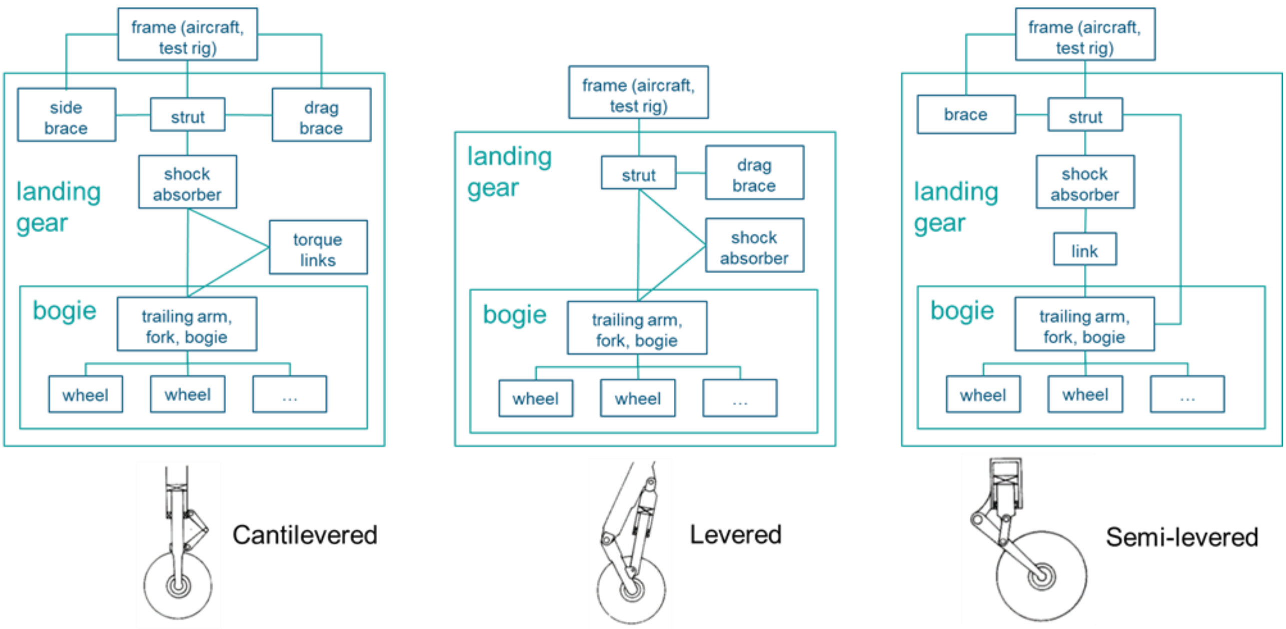

Figure 4: Three landing gear architectures that are available in the synthesis tool. Many variations on these three are possible by adding more or less braces and wheels.

DOI: 10.13009/EUCASS2022-6101

Many variations on these architectures can be made efficiently in the GUI. For example, the telescopic landing gear architecture can be a variant with a steering mechanism or the mounting for the landing gear can be a drop test tower. Typically, a nose landing gear has a cantilevered architecture, while the main landing gears can have all three architectures as in Figure 4. Main landing gear typically have a drag brace and a side brace. This is also easily selectable from the GUI. Figure 5 shows a synthesized CAD model of an aircraft with 4-wheel bogies as main landing gear.

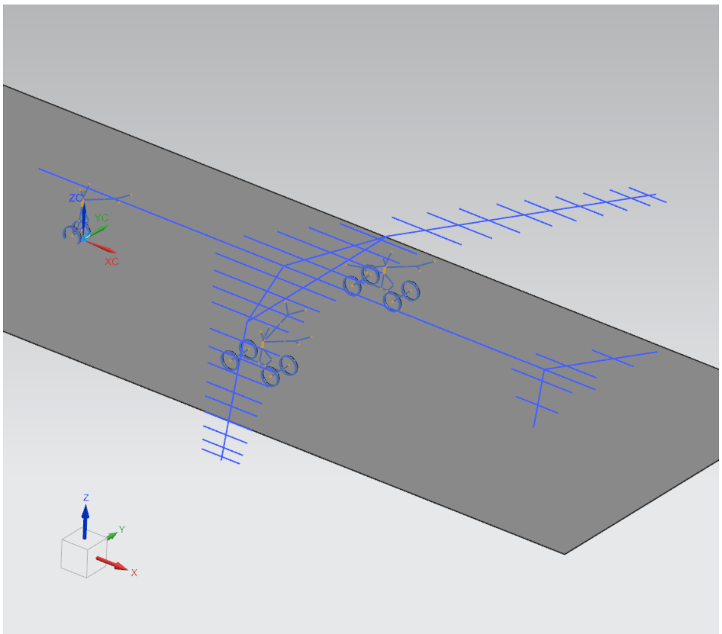

Figure 5: Synthesized multibody model of an aircraft with 3 landing gear of which the main landing gear have bogies with 4 wheels.

The second tab is used for creating and managing configurations of the topologies. The synthesis tool saves a lot of modeling time when there are multiple configurations that need to be modeled. For example, the architecture is fixed but some components have multiple variants with variations in cross-sectional shapes, materials or even different geometry. Modeling, simulating, and managing all these variations looks more like bookkeeping than engineering. Employing an MBSE methodology in combination with automated model synthesis saves a lot of time.

The other tabs serve to define the configuration: assign models to components, link handbook methods and scripts, request results, add load cases and initial conditions. Result requests can be for example: maximum displacement of a component or resultant force at a joint. The result requests serve two purposes. First, it can serve as a check of the completed simulation that the result is within expectations without the need of opening the simulation model. Second, the user can link output results to handbook methods such as Simcenter 3D's Margin of Safety (MoS) module. There it is possible to link simulation results to a script or pre-packaged MoS method.

The tool has already a rich library of template simulation models. An example of a template model of a brace was already shown in Figure 2. Other examples of template models can be found in Table 1. It is expected that a user would add custom made models. Though, these models serve as help to set up first simulations. It is also possible that these models suffice for obtaining first estimates of ground loads by performing multibody simulations.

Table 1: Overview of template models in the template model library.

Component	***Name***	***1D / beam elements***			***3D volumetric elements***		
		cad	**fea**	**mbs**	**cad**	**fea**	**mbs**
Mounting	1LG_Fixture	x	x	x			
	1LG_Testbench	x	x	x	x	x	x
	3LG_Airframe	x	x	x	x	x	x
	3LG_Flexible_Airframe	x	x	x			

DOI: 10.13009/EUCASS2022-6101

	4LG_Airframe	x	x	x			
Strut	Strut_Cantilevered	x	x	x	x	x	x
	Strut_Semilevered	x	x	x	x	x	x
	Strut_Levered	x	x	x	x	x	x
Shock Absorber	SA_Cantilevered	x	x	x	x	x	x
	SA_Levered	x	x	x	x	x	x
	SA_Semilevered	x	x	x	x	x	x
	SA_Cantilevered_Cosim			x			x
	SA_Levered_Cosim			x			x
	SA_Semilevered_Cosim			x			x
Torque links	TorqueLinks	x	x	x	x	x	x
Bogie	1Wheel_Bogie	x	x	x	x	x	x
	2Wheel_Bogie	x	x	x	x	x	x
	4Wheel_Bogie	x	x	x	x	x	x
Wheel	Wheel_SimpleTyre	x	x	x	x	x	x
Brace	Brace_Straight	x	x	x	x	x	x
	Brace_Y-shape	x	x	x	x	x	x
Steering system	Steering_cylinders_cosim	x	x	x	x	x	x
	Steering_input_torque	x	x	x	x	x	x

As a final note, the size of the library might grow over time. There is a filter button to reduce the number of template models available at a certain time. However, as more and more variants are being created for components, a strategy might be needed to reduce the size of the library. Compression of the library can be obtained by identifying hierarchies and similarities between the models. An example of a hierarchy is a stick model CAD part that serves as the basis for many other parts: a 3D volumetric CAD part, a stick model FE model, a stick model multibody model etc. If there are multiple 3D detailed CAD parts that rely on the same simple stick model, then structure is given to the library and repetitive models are eliminated. This is essentially applying template-based design to itself. It is generally accepted in the field of cognitive science that data compression and by extension library compression is an elementary part of intelligence [26][27]. Therefore, we need to keep the template library tidy.

4. User scenarios for landing gear simulation synthesis

In this section, a number of user scenarios are presented that showcase how the landing gear synthesis tool can be employed to obtain the simulation models needed for design. The simulation model files should be seen as the result of the tool because they serve as the basis for a Design Space Exploration (DSE) or optimization.

4.1 Multibody simulation of a landing

In this first scenario, a landing of a business jet needs to be simulated to obtain the ground loads on the main landing gear during a two-wheel landing where the main landing gear absorbs most of the energy. Flight regulations define many load cases that are proportional to the limit vertical inertia load [28]. If this load can be obtained through simulation, then the structural design department can derive a list of hundreds of load cases. The maximum vertical descent velocity is determined to be 3 meters per second. The horizontal approach velocity is determined to be 1.3 times the reference stall speed. In the case of the business jet, a value of 50 meters per second is chosen for the horizontal velocity. The airframe mass is 2500 kg.

Figure 6 shows the synthesized CAD model of the aircraft just before touch-down. The template of the airframe does not have landing gear but provides attachment points for the struts and of the nose and main landing gear. The airframe is oriented parallel to the ground in the template. The configuration defined 15 degrees pitch angle. The user chose a cantilevered (a.k.a. telescopic) nose landing gear and levered main landing gear. In this example, the shock absorber is defined with a stiffness curve and a constant damping coefficient.

DOI: 10.13009/EUCASS2022-6101

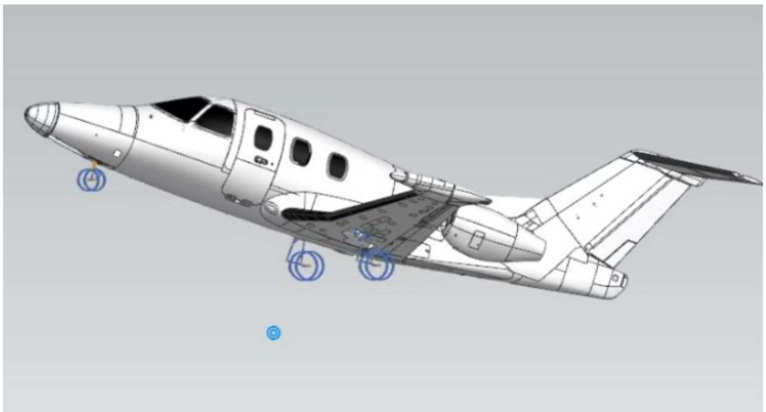

Figure 6: Synthesized CAD and Simcenter 3D Motion model of a landing business jet.

A simulated landing can provide a lot of information. For example, in Figure 7, the drag force and vertical force on the wheel hub is plotted together with the vertical force on the nose landing gear. The pitched position makes the main landing gear come first in contact with the runway while the nose wheels make contact later. Simulating these landings multiple times for slightly varying conditions, e.g., different yaw angles, different velocities etc. will produce a distribution of load sets that are needed for sizing.

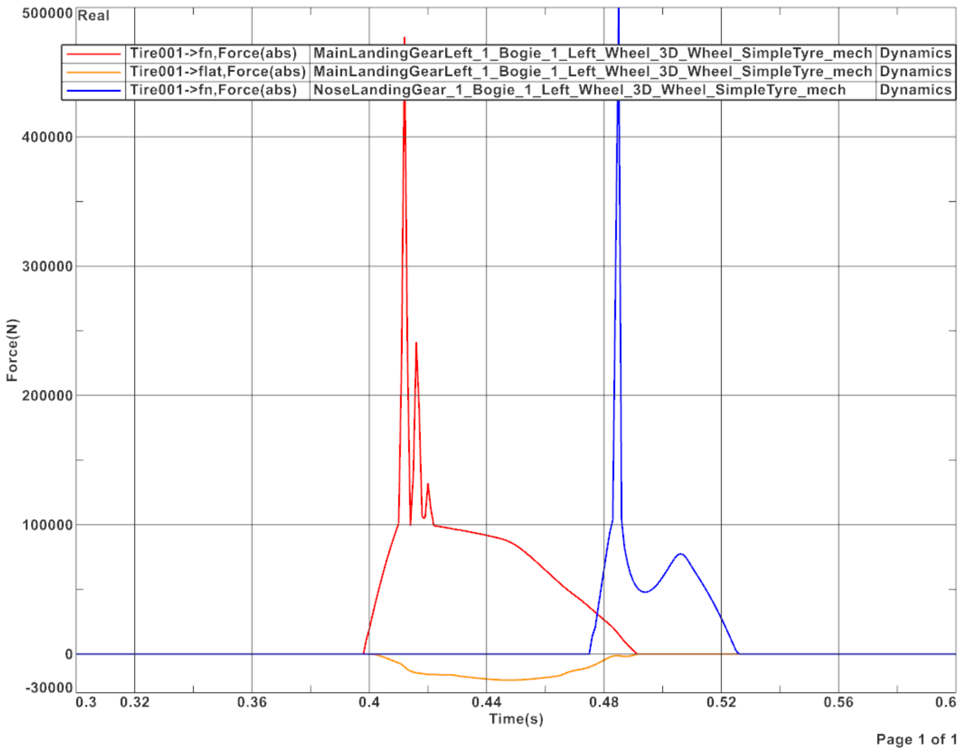

Figure 7: Forces on the wheel hub during 2 point landing. (red) vertical force of the main landing gear. (orange) drag force on the main landing gear. (blue) vertical force on the nose landing gear wheel.

4.2 Multibody simulations of taxing over a bump and turning

Another set of critical loads might occur during taxi. A turn can be simulated by driving the aircraft forward while rotating the nose wheel. The simulation model can also be synthesized by the presented tool. In this case, a system architecture with a nose landing gear with a steering mechanism is selected. The same business jet model can be used because a driver with a forward velocity profile can be selected from the tool. Similarly, the steering angle profile is also selected from the tool. Figure 8 shows the synthesized business jet model taxiing over a bump.

DOI: 10.13009/EUCASS2022-6101

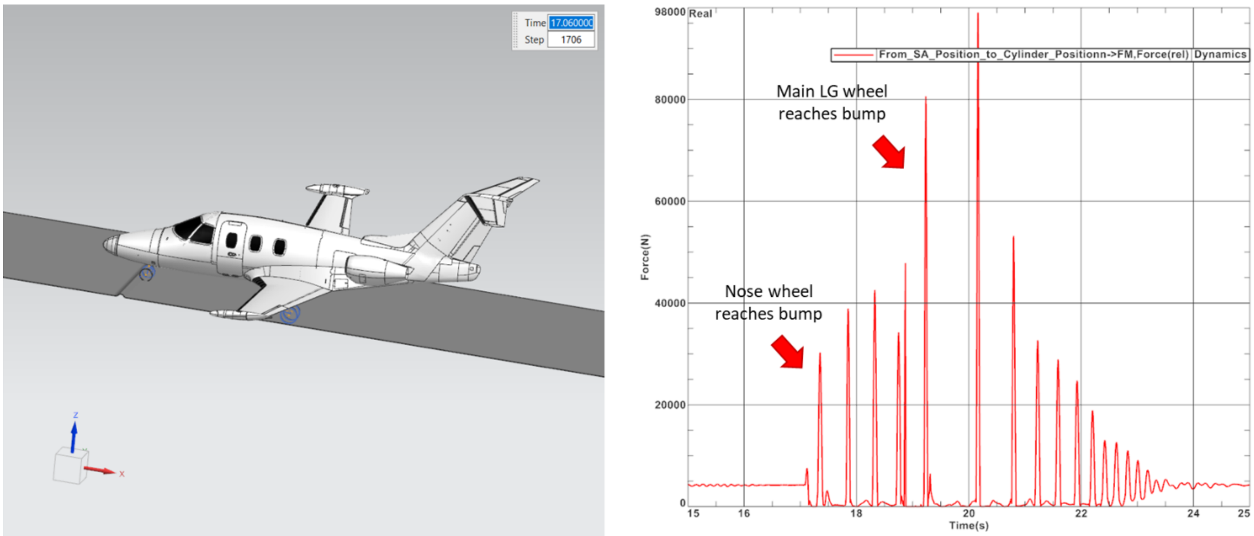

Figure 8: The synthesized multibody simulation in Simcenter 3D Motion. The aircraft reaches a bump and drives over it. The same models were used as for the landing simulation. The road profile and driver is controlled from the tool's GUI.

4.3 Conceptual and preliminary sizing based on FE analysis and multi-body kinematics

Probably most important for the landing gear design engineer is a tool that can manage and synthesize the FE models and simulations. The managing of hundreds of load cases is cumbersome and error-prone. The synthesis tool can import a spreadsheet and apply all loads automatically. Figure 9 shows the synthesized FE analysis model on the left and the Von Mises stresses on the right.

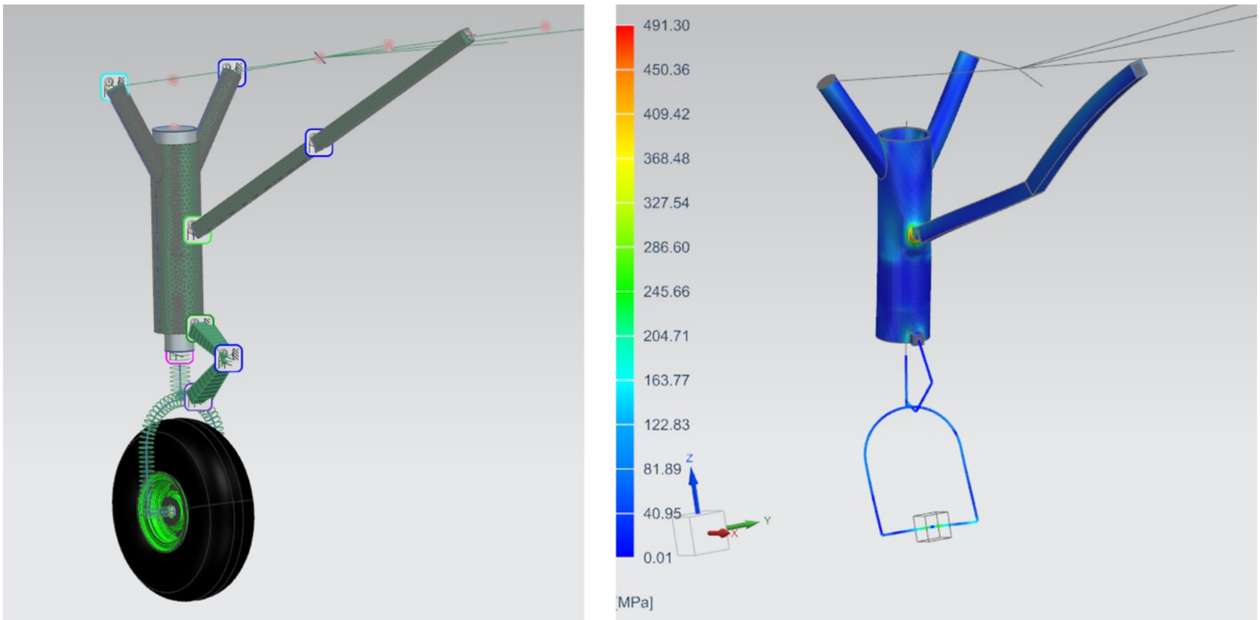

Figure 9: On the left, a synthesized FE model of a nose landing gear which consists of a mix of component models with varying fidelity level. The strut consists of tetrahedron elements based on 3D CAD geometry. The shock absorber only holds some 3D CAD geometry but the FE model consists of a few Simcenter Nastran RBE 2 elements. The torque links consist of beam elements as does the fork. The wheel is just a simple FE model that contains detailed CAD geometry, but that is simplified to a single lumped mass. On the right, the Von Mises stresses results due to a force and a moment at the wheel hub.

DOI: 10.13009/EUCASS2022-6101

Furthermore, the tool makes it possible to link FE analysis results to handbook methods in the form of scripts. In Simcenter 3D this is known as Margin of Safety (MoS) calculations. As an example, a margin of safety calculation is created for pins and lugs sizing. In the early stages of preliminary design, a FE analysis based on beam elements is much faster than a full-fledged simulation with 3D mesh. Beam elements cannot properly model the pins and lugs of the hinges of the landing gear. Therefore, a margin of MoS calculation is ideal. The method receives the resultant forces from the beam elements and calculates the MoS for the hinge. The MoS calculation mechanism can even be used to optimize the pin and lug dimensions for a given load and desired MoS.

This last scenario shows an important capability of the tool: associativity between multiple simulation models. This means that the same underlying CAD model is shared between the FE model and the multibody simulation model. The consequence is that a parameter change in the CAD model will invoke a model update in both models. This makes an optimization more streamlined because the parameters of the CAD model need to be tagged only once for the optimization software. For example, in HEEDS a workflow can now easily be set up with the synthesized simulation models.

5. Conclusion

A software tool was developed that synthesizes CAD models, FE simulations, and multibody simulations in Simcenter 3D by using a library of template models. The tool allows a non-expert user to select a landing gear architecture, assign models to the components, change parameters, add extra boundary conditions that are not already in the model template, request analysis outputs, and add engineering handbook methods that can use simulation results for calculations. The software also has an intuitive graphical user interface. The software unites FE analysis and multibody simulation by allowing them to share the underlying CAD model. Furthermore, the tool promotes the capturing and reuse of in-house knowledge because template models require the creator to attend to interfaces with other components, parametric robustness, design intent, and causality of the CAD features. Consequently, a user can take advantage of the expert's knowledge and quickly piece together a simulation model that suits their needs.

The tool aims to accelerate the setup of full system analyses such as FE ground load cases and dynamic multi-body simulations of maneuvers like landing, taxi, and towing. Multibody simulation models can be structurally flexible and can express complex dynamics e.g., tyre models. Templates for the shock absorber can include a co-simulation with Simcenter Amesim. The focus of the tool is to create the simulation models needed for Design Space Exploration (DSE) and optimization studies. The tool could also be useful for OEMs who want to study high-level landing gear decisions to optimize load transfer to the airframe.

With this work, a step is made toward a heterogeneous MBSE methodology combing 0D and 3D simulation models. The approach is generic and is applicable for the design and analysis of any mechanism that can be modeled using templates. When system architectures and template models can be reused, a significant speedup of the simulation process can be achieved.

6. Future work

A future task is to demonstrate an industrial workflow where a landing gear simulation model is synthesized and optimized in a dedicated software tool like HEEDS. It should be possible to create an optimization workflow where first a static linear FE analysis is performed to retrieve resultant forces. Additionally, the margin of safety methods in Simcenter 3D can be used to size the dimensions of the pins and lugs. Subsequently, the pin and lug dimensions will be used to update a detailed CAD model onto which a detailed FEM and multibody model are built.

A logical next step would be to use the methodology and tool for other applications besides landing gear design. There are many systems, also outside aerospace industry, that could benefit from this design strategy. Work is already done on synthesizing high-lift devices. Other applications that come to mind are: satellites, lander spacecraft, heavy equipment etc.

Work can also be done on more automation. As with many technologies today, machine learning could be used to speed up some tasks. For example, model parameters can be pre-tuned by the model depending on the architecture that is chosen. Another task to automate is the inference of causality of the models. Now, the causality of the connections between components needs to be defined on the architectural level. However, the causality of the architecture could maybe be left undefined and then solved once there are simulation models defined.

DOI: 10.13009/EUCASS2022-6101

Finally, the data model of the tool and the code that drives the synthesis process can be used for many other tasks that are typically seen as facets of the digital twin. In the Assisted DfA project, a framework is set up to assess the assemblability of a CAD assembly. The presented tool could be extended to also "simulate" the procedure to assemble the system. In this case, the CAD model is a simulation where collision detection is performed.

7. Contact and acknowledgment

The authors can be contacted at elias.allegaert@siemens.com, yves.lemmens@siemens.com and emre.ongut@siemens.com.

The research presented in this paper has been supported by the Soft Landings project (Solution for Fast Landing Gear Simulation) and has received funding from the Flemish government through the VLAIO O&O Programme under grant agreement (ref. HBC.2020.2011) and the Assisted DfA project (Automatic Assisted Design for Assembly) through the VLAIO ICON Programme (ref. HBC.2021.0014)

References

[1] Atkins, E.M. and Bradley, J.M., 2013. Aerospace cyber-physical systems education. In AIAA Infotech@ Aerospace Conference, pp. 1-8
[2] Long, L.N., 2015, On the need for significant reform in university education, especially in aerospace engineering. In 2015 IEEE Aerospace Conference, pp. 1-7
[3] Komoto, H. and Tomiyama, T., 2012. A framework for computer-aided conceptual design and its application to system architecting of mechatronics products. Computer-Aided Design, 44(10), pp.931-946.
[4] Van Beek, T.J., Erden, M.S. and Tomiyama, T., 2010. Modular design of mechatronic systems with function modeling. Mechatronics, 20(8), pp.850-863.
[5] Singh, H. and Hassan, S.I., 2015. Effect of solid design principles on quality of software: An empirical assessment. International Journal of Scientific & Engineering Research, 6(4), pp. 1321-1324
[6] Benveniste, A., Caillaud, B., Nickovic, D., Passerone, R., Raclet, J.B., Reinkemeier, P., Sangiovanni-Vincentelli, A., Damm, W., Henzinger, T. and Larsen, K.G., 2015. Contracts for systems design: Theory, Research Report RR-8759, Inria Rennes Bretagne Atlantique.
[7] Warwick G, Norris G, 2010. Designs for success, systems engineering must be rethought if program performance is to improve. *Aviation Week Space Technology*, 172(40), pp.72–75
[8] Hagen, C. and Sorenson, J., 2013. Delivering military software affordably. *Defense AT&L*, 42(2), pp.30-34.
[9] Menu J. and Nicolai, M., 2012. A framework for automated design, verification, and simulation of electrical power systems for aircraft. International workshop on aircraft system technologies, Hamburg, Germany.
[10] Menu J, Nicolai M, Zeller M., 2018. Designing Fail-Safe Architectures for Aircraft Electrical Power Systems. Proc of AIAA/IEEE Electric Aircraft Technologies Symposium (EATS), Cincinnati, pp. 1-14.
[11] Neumaier, M., Kranemann, S., Kazmeier, B. and Rudolph, S., 2022. Automated piping in an Airbus A320 landing gear bay using graph-based design languages. *Aerospace*, 9(3).
[12] Zeigler, B.P., Mittal, S. and Traore, M.K., 2018. MBSE with/out Simulation: State of the Art and Way Forward. *Systems*, 6(4).
[13] Graignic, P., Vosgien, T., Jankovic, M., Tuloup, V., Berquet, J. and Troussier, N., 2013. Complex system simulation: proposition of a MBSE framework for design-analysis integration. Procedia Computer Science, 16, pp.59-68
[14] Laukotka, F., Hanna, M. and Krause, D., 2021. Digital twins of product families in aviation based on an MBSE-assisted approach. Procedia CIRP, 100, pp.684-689.
[15] Bougain, S. and Gerhard, D., 2017. Integrating environmental impacts with SysML in MBSE methods. Procedia CIRP, 61, pp.715-720.
[16] Santos, C.A.R.D., Schrijvers, T., Saleh, A.H. and Nicolai, M., 2021, Divide et Impera: Efficient Synthesis of Cyber-Physical System Architectures from Formal Contracts. In International Symposium on Formal Methods (pp. 776-787).
[17] O'Hara, C., Menu, J. and Van Den Brand, M., 2022, April. COGENT: A Concurrent Engineering and Generative Engineering Tooling Platform. In 2022 IEEE International Systems Conference (SysCon) (pp. 1-8). IEEE.
[18] Gawthrop, P., Bevan, J. and Geraint, P. 2007. Bond-graph modeling: a tutorial introduction for control engineers. IEEE Control Syst. Mag, 27, pp.24-45.
[19] Camba, J.D., Contero, M. and Company, P., 2016. Parametric CAD modeling: An analysis of strategies for design reusability. Computer-Aided Design, 74, pp.18-31.

DOI: 10.13009/EUCASS2022-6101

[20] van den Berg, T., Beijer, B. and Moerland, E., 2019. Application of an integrated and distributed multidisciplinary product development framework to a multi-tier aircraft design case. In AIAA Aviation 2019 Forum (p. 3327).

[21] La Rocca, G., 2012. Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design. Advanced engineering informatics, 26(2), pp.159-179.

[22] van Gent, I., La Rocca, G. and Veldhuis, L.L., 2017. Composing MDAO symphonies: graph-based generation and manipulation of large multidisciplinary systems. In 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, pp. 1-23.

[23] Boussuge, F., Tierney, C.M., Vilmart, H., Robinson, T.T., Armstrong, C.G., Nolan, D.C., Léon, J.C. and Ulliana, F., 2019. Capturing simulation intent in an ontology: CAD and CAE integration application. Journal of Engineering Design, 30(10-12), pp.688-725.

[24] Zhang, M., Jungo, A., Gastaldi, A.A. and Melin, T., 2018. Aircraft geometry and meshing with common language schema CPACS for variable-fidelity MDO applications. Aerospace, 5(47), pp. 1-22.

[25] Allegaert, E., Menu, J., Lemmens, Y., Dutré, S., Ongut, E. and Wilson, W., 2020. Architecture-based conceptual design for mechanical systems applied to landing gear. In Aerospace Europe Conference, Bordeaux

[26] Ghahramani, Z., 2015. Probabilistic machine learning and artificial intelligence. Nature, 521(7553), pp.452-459.

[27] Maguire, P., Moser, P. and Maguire, R., 2016. Understanding consciousness as data compression. Journal of Cognitive Science, 17(1), pp.63-94.

[28] European Aviation Safety Agency, 2012. Certification Specifications for Normal Utility, Aerobatic, and Commuter Category Aeroplanes—CS23.