TAKING INTO ACCOUNT THE CHANGE OF GEOMETRY IN SYSTEM SIMULATION PROCESSES

Mauser, Kristian; Breitsprecher, Thilo; Hasse, Alexander; Wartzack, Sandro
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany

Abstract

In product development process of dynamic systems more than one technical field (i.e. mechanical, electrical or combustion technologies) affects the product. It is of particular importance to ensure the functionality of such multi-domain products especially after changing one of the domain-specific sub-process (i.e. CAD-data in mechanical design). This is possible by simulating the system behaviour of the overall product in a multi-domain system simulation. The consideration of the detailed information of each technical domain in the system simulation is not possible at moment. Therefore the relevant information in each sub-process have to be identified and insert into the system simulation. In this contribution focus is set on mechanical sub-process. An application is developed that identifies all changed information of a product design in the CAD-environment and provides this information to the system simulation. A proof of the functionality of the developed procedure in shown in a use case. The piston of a single-cylinder combustion engine model is changed by the designer. After that the developed application is executed and the changed simulation results are discussed.

Keywords: Systems engineering (SE), Design engineering, Design process

Contact:
Thilo Breitsprecher
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)
Chair of Engineering Design
Germany
breitsprecher@mfk.fau.de

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

Development of dynamic systems has gained a lot of importance recently and its valuable benefits are continuing to grow with the advent of new engineering processes, products and technologies. In addition most dynamic products involve different technical domains. Examples include among other things automation (e.g. pick- and place applications), automotive (e.g. crank mechanism in combustion engines) or medical technologies (e.g. active prosthesis). A multi-domain approach for the product development process (PDP) allows the view of the interactions between different technical domains which affect a product. Therefore, a high degree of integration of the different domains and flexibility in the product development can be achieved. Due to the consideration of different domains the complexity of the product development process increases. Hence, major challenges arise during the development of such products, for instance a suitable product representation with an adequate product data model is essential for each technical domain. Therefore the product development of interdisciplinary and dynamic systems requires a methodology which handles these challenges. Consequently, current processes, methods and IT-solutions must be reconsidered and adapted to interdisciplinary dynamic systems (Eigner et al., 2012). One design methodology for the development process of interdisciplinary products is given with the V model for mechatronic systems according to VDI 2206 (see section 2.1) (2004). Basically the V model describes a very general procedure (Breitsprecher et al., 2010) that starts with the requirements of the target product. After this, a multi-domain system design describes the operating principle of the product. Thereafter, the V model divides the multi-domain PDP into domain-specific sub-processes to find a solution for each domain-specific problem. Different technical domains use different methodologies. For example, in engineering design the VDI 2221 approach (1993) and the approach of Pahl et al. (2007) are the most common methodologies for product development. But also the Munich Procedural Model (Lindemann, 2009) can be used. After this step the domain-specific solutions are integrated into an initial design of the overall system. Several iterations of the V model are necessary to achieve the final design.

Domain-specific changes affect the overall product. Therefore it is important to ensure the functionality of the overall system after each change or iteration step. In this context the functional verification in the product development is of particular importance, because several technical domains are affected. This increases the complexity and the product designer needs to be supported by computer based tools that integrate or link the domain-specific and the multi-domain levels. There are several simulation tools like Modelica or Simulink which enable the design engineer to model the principle solution of an interdisciplinary system and to simulate its behaviour.

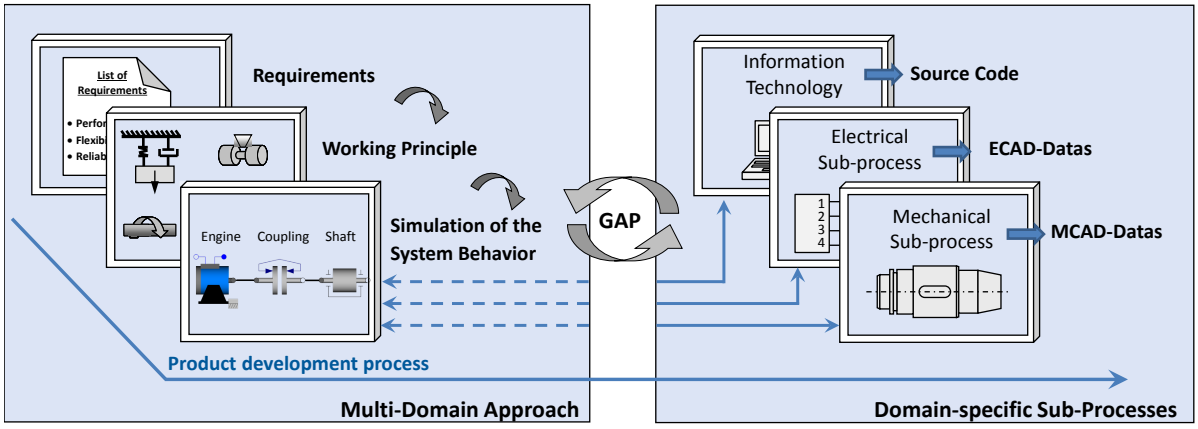

Figure 1. Gap between multi-domain system simulation and domain-specific sub-processes. Each sub-process has specific results, such as source code, ECAD data or MCAD data

The created multi-domain model of the system behaviour (see left side of Figure 1) represents the basis for the domain specific sub-processes (right side of Figure 1) and is integrated in the V model as part of the system design as seen in the left branch of the V model (compare Figure 2). It concretizes the operating solution and exhibits the overall multi-domain system behaviour of the proposed product solution. According to the V model the overall multi-domain system behaviour is now divided into

domain-specific sub-processes. Each sub-process embodies specific output data and representations such as mechanical computer-aided-design (CAD) data. After a change in one sub-process it is of great importance to ensure the functionality of the overall multi-domain system and of the other sub-processes in order to avoid failure. Therefore appropriate interfaces and data formats are needed to ensure a bidirectional exchange of information between the domain-specific sub-processes and the overall multi-domain system. But there is a gap in the bidirectional communication between the multi-domain system behaviour model and the domain-specific sub-processes as illustrated in Figure 1. The missing linkage leads to an inaccurate simulation of the system behaviour. Therefore this gap has to be bridged to ensure the functionality after each iteration step of the V model (Breitsprecher et al., 2010) and after each change in one of the sub-processes. Several disadvantages result in an inadequate linkage. For example more iterations of the V model, a loss of information and a longer development process can occur. In this paper focus is set on mechanical sub-processes. The missing linkage between this sub-process and the overall multi-domain model is represented by an inadequate consideration of the geometry of the product (MCAD) in the overall multi-domain simulation model. Without detailed information about the product's geometry the simulation results of the overall multi-domain model are limited in detail. Therefore the gap needs to be bridged to provide detailed information about the product's geometry to the overall simulation. The same problems occur in other sub-processes (e.g. ECAD-data in electrical sub-processes) but these are not part of this paper. The overall objective is to close the gap step by step. The first step is represented in this paper.
Objective of this contribution is to bridge the mentioned gap between mechanical sub-processes and the overall multi-domain system to ensure the functionality after each change in one of the mechanical sub-process and iteration step of the V model. Therefore, detailed physical information of this sub-process has to be available to the overall multi-domain system simulation model. The paper starts with a brief description of the V model. Furthermore, well-known methods and tools for the modelling of dynamic systems are presented (Chapter 2). Chapter 3 describes the procedure to integrate the physical information of the mechanical sub-process into the multi-domain system simulation model. This is followed by a use case that shows a change in the physical information of a single-cylinder combustion engine (MCAD) and the resulting changes in overall multi-domain system simulation (Chapter 4). Conclusion of the paper is in Chapter 5.

2 BACKGROUND AND RELATED WORKS

2.1 Variant design as the main focus of this work

Pahl et al. (2007) distinguish three degrees of novelty for a product: original, adaptive and variant design. Original design involves the fulfilment of new tasks or problems by means of new combinations of known solution principles. The complete PDP is performed more or less from the beginning (requirement elicitation) until the end (manufacturing documentation). In contrast to that adaptive design features an unchanged principle solution with only the embodiment to be changed. Variant design is performed by changing the sizes and arrangements of product component whilst not exceeding previously defined design limits. Recently, Feldhusen and Grote (2013) added repetitive design to this understanding, which means the remanufacturing of a developed product that has been produced before. At the present state our work focuses on adaptive design and the analysis how the system's behaviour changes with a different embodiment.

2.2 The V model for the development of mechatronic systems

Typical engineering design methodologies like from Pahl et al. (2007) are limited when it comes to the development of multi-domain products because each domain prefers to follow a specific procedure. As depicted in Figure 2 the V model is a recommended procedure description for the development of mechatronic systems and has evolved from the eponymous model from software engineering. This work is embedded in the domain-specific branch of mechanical engineering and furthermore assists during the assurance of properties.

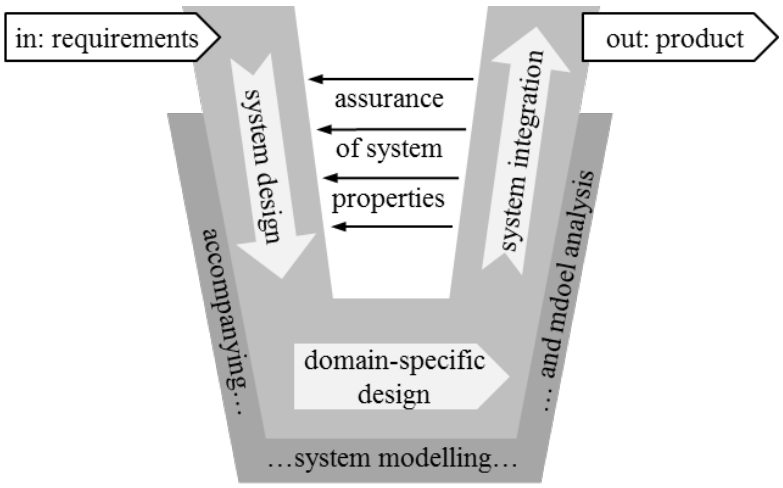

Figure 2. The V modell as described in VDI (2004) is a macro-cycle process model for the development of multi-domain systems.

2.3 Modelling of technical systems

A variety of methods is available to model and analyse multi-domain systems. Pahl et al. (2007) assign a specific function to every technical system and advise to break this function down into subfunctions, depending on the complexity. A function, however, is the combination of a defined flow of energy, matter and information. Based on this function structure working principles can be identified and combined to form the principle solution.

An approach which has recently grown in importance is presented by Albers et al. (2008) and is commonly known as the Contact and Channel Model (C&CM). The C&CM enables design engineers to combine abstract function structure models with definite, almost spatial geometric models which they are used to work with. A C&CM consists of pairs of working surfaces and support structures. Working pairs can be understood as mechanical interfaces where forces are exchanged, e.g. the contact of gear teeth for the transmission of tooth-to-tooth forces. This force is "channelled" through the support structure to the next pair of working surfaces, e.g. the shaft-hub connection. In this example the support structure corresponds to the gear itself.

2.4 Analysis tools for multi-domain systems

In order to model multi-domain systems engineers can fall back on different methods. Block diagrams, as described in Karnopp (2012), are a graphic representation for the description of mathematical interrelations between system components. Block diagrams are not limited to any physical correlation, however, they are rather used for higher level than detailed descriptions. Furthermore, they lack of transparency and traceability for extensive systems.

To overcome such shortcomings the object-oriented modelling language Modelica was introduced by Elmqvist (1978) and since then it has gone through a severe development process. Modelica enables to model a system whilst taking different domains into account such as electric, magnetic, mechanic, fluidal or thermal components. Furthermore, elements for control engineering can be integrated. The user can easily model the system of interest either by programming it in an object-oriented manner or by simply choosing blocks from domain-specific libraries which are described in detail inter alia in Fritzson (2004). For our work we will use Modelica, since it is not only a flexible language but also has become a quasi-standard and can be interpreted both by open-source (e.g. JModelica, OpenModelica) and commercial tools (e.g. Dymola, SimulationX, Wolfram SystemModeler).

3 DESCRIPTION OF APPROACH

The following chapter describes the general method of linking the CAD-information with the system behaviour model. Therefore a general approach is developed. In this case the CAD-information represents the domain-specific parameters. The system behaviour is described by the multi-domain simulation model.

According to Weber (2005) modelling products and product development processes can be distinguished between characteristics and properties. Characteristics describe the shape and structure of a product, the spatial arrangement of the components, dimensions, material and surface parameters. The characteristics can be influenced directly by the product developer. In contrast to that, the properties embody the product behaviour, for example weight, stiffness, reliability or production costs of a product. The product developer cannot influence the properties directly but he can manipulate them indirectly by changing the characteristics. According to Weber (2005) in product development process it can be distinguished between the two core activities analysis and synthesis. Based on given characteristics of a product the analysis determines the product behaviour. The analysis can either be implemented by experiments or simulations (Vajna, 2009). However, the synthesis is the main task in engineering design. Based on predetermined properties the characteristics of the solution can be determined. For instance, the predetermined properties can be given by the conceptual formulation. The design engineer constantly iterates between both activities until the properties meet the requirements.
Our use case contains a system behaviour model as multi-domain approach and a 3D-CAD-model for the engineering design sub-process. The system behaviour can be described by a special modelling language for complex systems named Modelica (Fritzson, 2004). This language offers a variety of modelling elements in multi-domain libraries such as electrical, mechanical, thermal, control and electronic components. Consequently the language is well suited to describe multidisciplinary products in a system simulation. For a detailed system simulation accurate information about the geometry is needed.

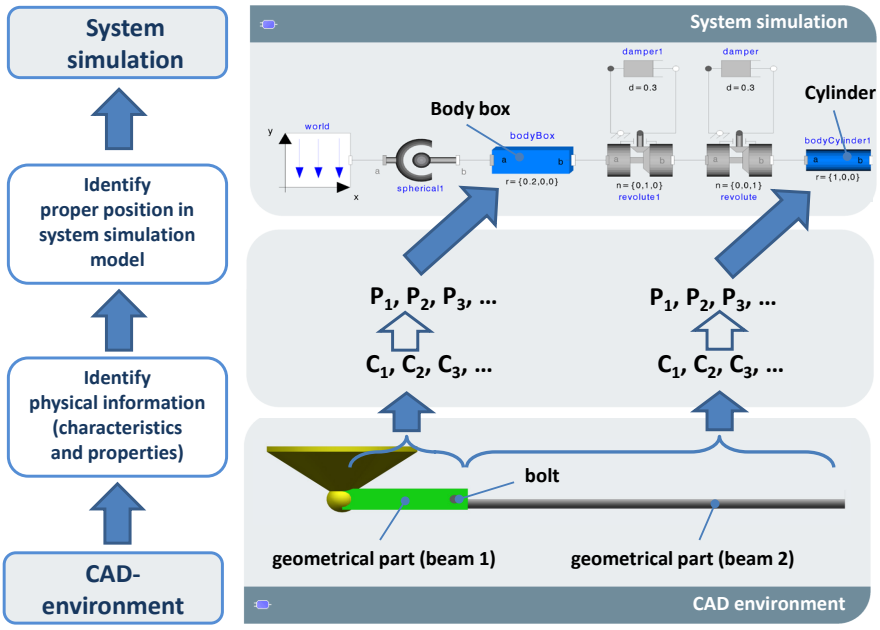

Figure 3. Steps of the workflow are shown on the left side. On the right side a CAD-product (example: double-pendulum) is illustrated in the lower section. The characteristics (C) and properties (P) have to be identified for each part of the product and provided to the proper position in the system simulation model

In most cases system simulation is not shape-oriented and there are just limited capabilities to describe geometry. For instance, in the upper right side of Figure 3 the blue boxes represent a body box and cylinder in a system simulation model. The body box represents beam 1 of a double-pendulum (see lower centre of Figure 3). But a consideration of the hole for the bolt is not possible in the system simulation due to the limited capabilities to describe geometry. This is a quite simple example but it emphasise that complex geometry cannot be modelled with such languages. Consequently the simulation results have a restricted accuracy. In case of this context the gap between the non-shape-oriented system simulation and the detailed 3D-CAD design need to be bridged. To increase the accuracy of the system simulation results the characteristics (C in Figure 3) and properties (P in Figure 3) of the CAD-environment have to be linked to the system simulation to simulate the impact

on the system's behaviour. Initially the characteristics (e.g. geometric dimensions, material) and properties (e.g. inertia, centre of gravity) for each part of the assembly have to be identified in the CAD-environment. These characteristics and properties embody all relevant information of the parts and represent the input of the system simulation. Inside the system simulation the identified information of each part needs to be available in the right place. That means the product designer has to decide which physical information belongs to which component in the simulation model. After changing the characteristics, the mentioned workflow needs to be performed to update the simulation results. To sum up, the workflow is listed below:

1. Identification of the necessary characteristics and properties of each part in an assembly
2. Storage of characteristics and properties with regard to their part
3. Identification of the position in the system simulation that represents the parts
4. Delivery of the right information to the right position in the system simulation

In this context the gap between the characteristics of a conventional CAD-system and the results of a system simulation should be bridged. To reduce the time commitment of the product designer this process should be automated. Especially step three in the workflow is challenging to realize in an automated way because the application has to identify by itself which physical information belongs to which component of the system simulation. In the following chapter a use case is presented which shows a prepared single-cylinder combustion engine model as demonstrator in a CAD-environment. In this use case the characteristics of the single-cylinder combustion engine are changed by the product designer. Thereafter the developed application performs the workflow mentioned above by itself and the results of the system simulation are automatically adapted. This case is comparable to a new iteration step of the V model or to the transmission of product changes (CAD-level) to the system simulation.

4 USE CASE: SINGLE-CYLINDER COMBUSTION ENGINE

Within the scope of this paper the CAD-system CATIA V6 of Dassault Systèmes is used. This application implies a special method for system engineering which is called RFLP. This acronym stands for Requirement Functional Logical and Physical and represents a system engineering process. It is based on the V model design process for multidisciplinary engineering process. Similar to the extended V model according to Eigner et al. (2012) the RFLP environment divides the left part of the V model in four steps. The requirement approach describes the demands on a product. The functional section specifies the functional structure of the system. The logical approach concretizes the components. In addition the system behaviour can be described by a special modelling language for complex systems named Modelica (compare Figure 4). In the physical section the geometry of the part can be modelled comparable to a conventional CAD-system. All familiar design tools and features are available. Consequently the parts can be fully modelled and detailed in this layer. In the following figure the specification tree of the RFLP-environment in CATIA V6 is shown on the left side. This figure illustrates the relation between the specification tree and the multi-domain (logical) and domain-specific (physical) progress.

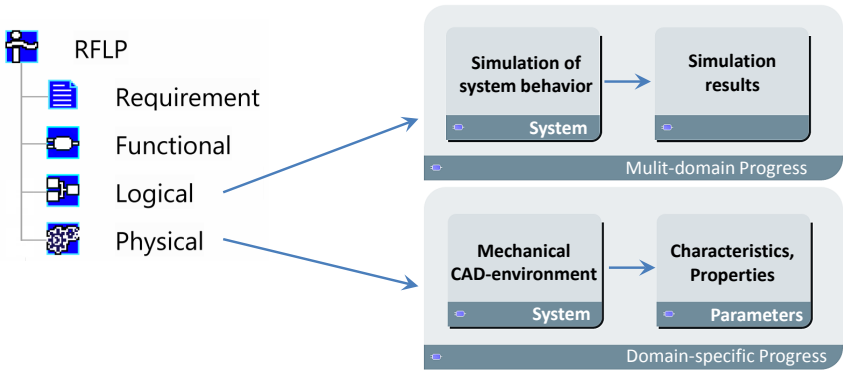

Figure 4. Specification tree (left) of the CATIA V6 RFLP-environment. Domain-specific progress, in this case the 3D-CAD environment with its parameters and the multi-domain progress with the simulation model of the system behaviour and the simulation results

As starting point for modification and variant design (focus of this paper) the description of an entire product model can be assumed. In this context the entire product is described in the RFLP-environment with its requirements, its functional structure, its system behaviour and its physical shape. Currently, there is no possibility to link the relevant information of the physical layer (domain-specific progress) with the logical level (multi-domain progress). Consequently, the relevant physical information, the characteristics and properties of the domain-specific (mechanical) sub-process, will not be considered in the multi-domain system simulation or just in a simplified way, because of the non-shape oriented modelling language Modelica. Relevant information from the physical level includes material attributes and the geometrical shape. This information (e.g. inertia, density, geometrical dimensions) is needed in the logical level to set up a detailed system simulation. For this progress the essential steps in CATIA V6 are based on the workflow mentioned above and performed automatically. Therefore additional software is developed for CATIA V6. In the following the demonstrator is presented.

4.1 Single-cylinder combustion engine

In this context a conceptual model of a single-cylinder combustion engine is used as demonstrator. Due to the early stage of development the demonstrator exhibits a limited complexity. Nevertheless, several characteristics may be manipulated to realize a change in the system simulation results and consequently in the product's behaviour.
The single-cylinder combustion engine represents a multidisciplinary product and involves mechanical and combustion technologies. The model of the engine consists of a crankshaft, a locating pin, a connection rod and a piston as shown in the illustration below. These parts are represented in the physical layer of the RFLP environment.

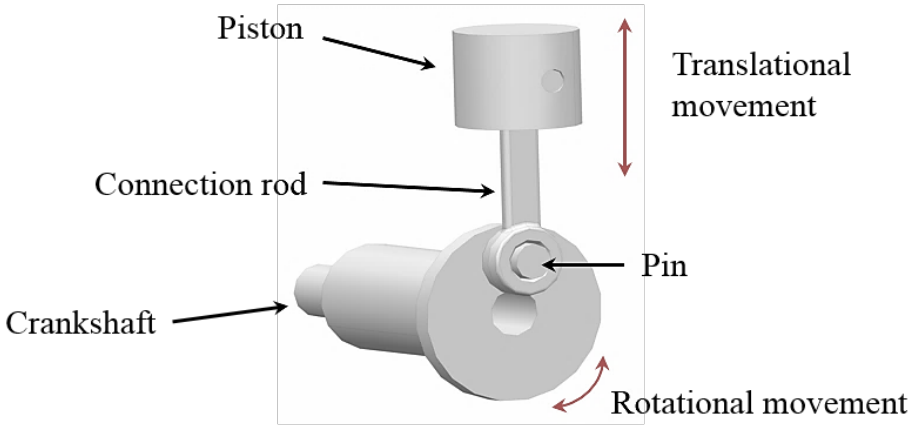

Figure 5. Model of a single-cylinder combustion engine (Physical Layer)

The combustion chamber is part of the system simulation and not integrated into the physical model. The force produced due to the combustion acts on the piston and induces a translational movement. This force is transmitted via the connection rod to the crank shaft and the translational movement of the piston is converted into a rotational movement of the crankshaft. The combustion chamber is integrated in the logical layer of the RFLP-environment and described with the modelling language Modelica. In the logical section the complete system behaviour is described. For a detailed system simulation the characteristics and properties of the physical layer need to be integrated into the simulation to verify the functionality of the overall system. Therefore, components are integrated into the simulation, which describe the geometry. Input data of these special components are the characteristics and the properties of the product of the physical layer in order to describe the system behaviour properly. In the following the use case is described in detail and the results are presented.

4.2 Result presentation

In this paper a prepared RFLP model is the starting point. In the first step the characteristics of the demonstrator mentioned above will be changed by the product designer. After this step a developed application will be executed. This application identifies the changed characteristics and properties in

the physical layer of the RFLP-environment of CATIA V6. Thereafter all relevant information is stored temporarily in the cache. In the next step the correct position will be identified to place the information for each part properly. It is important, that the information of the physical section is provided for the associated components in the system simulation. These components represent the design of the parts in the system simulation and require the detailed physical information about the parts of the physical layer for a proper simulation.
In this process the changed characteristics are the length and material of the piston. These two parameters have influence on the mass of the piston. Also other properties like inertia, point of centre, velocity of the piston or bearing forces of the crankshaft are affected. In the following the velocity of the piston is shown with higher and reduced piston length. For a better comparability of the two following graphs the larger piston will be made of steel and the material of the smaller piston will be an aluminum alloy. This leads to a major change of weight of the piston and to a better visualization of the change in the diagramme.
Due to the idealized, simplified model the velocity describes a cycled graph. By decreasing the mass of the piston the curve progression will not change significantly but the values, the amplitudes and the cycle duration will change. The velocity of the piston over the time with the higher piston length and virtually manufactured of steel is shown as follows.

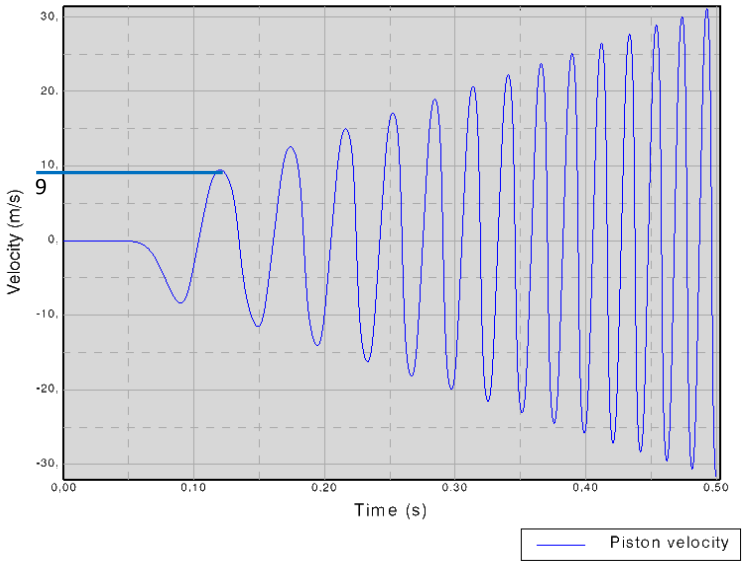

Figure 6. Piston velocity with high piston length made of steel

The velocity behaviour of the larger piston is represented in the diagramme above. The axis of abscissas shows the duration in seconds. In this case the maximal duration is half a second. The ordinate shows the velocity of the piston in meter per seconds. The movement of the piston starts at the initial position. This point is the top dead centre of the single-cylinder combustion engine. After starting the simulation there is a small time delay until the piston moves at a noticeable velocity. At start most of the combustion force affects in vertical direction on the crankshaft. The crankshaft needs a force in horizontal direction to initial the rotational movement. Then the piston moves downwards to the bottom dead centre. After passing this point the slope of the velocity turns positive and the piston moves upwards. With reaching the upper dead centre the combustion force increases the velocity and the next cycle begins.
To imitate a new variant or a modification of the combustion engine the characteristics will be manipulated in the physical layer with the goal of achieving an improved performance by simultaneously less consumption. Therefore the product designer decreases the length of the piston and changes the material to an aluminium alloy in order to reduce the mass of the piston. By executing the developed application the changed characteristics of the parts of the product are integrated into the simulation of the system behaviour. This change in weight leads to a velocity behaviour of the piston.
The basic structure of the following chart is similar to the structure of Figure 6. But the velocity behaviour has changed. The reduced mass leads to an increased velocity of the piston. In the first cycles just a small difference between the two plots is visible. The difference increases significantly

with more cycles. The changes in the product characteristics lead to a total reduction of the piston weight of 400 percentages. This high degree of modification is necessary in order to make a change between the graphs visible in the first cycles. Comparing the extreme values clarifies the differences between the two curves. The maximum velocity of the first cycle of the light-weight piston amounts 13 m/s. The heavy weight piston has a maximum value in the first cycle of 9 m/s. In later cycles the velocity difference between these two pistons increases.

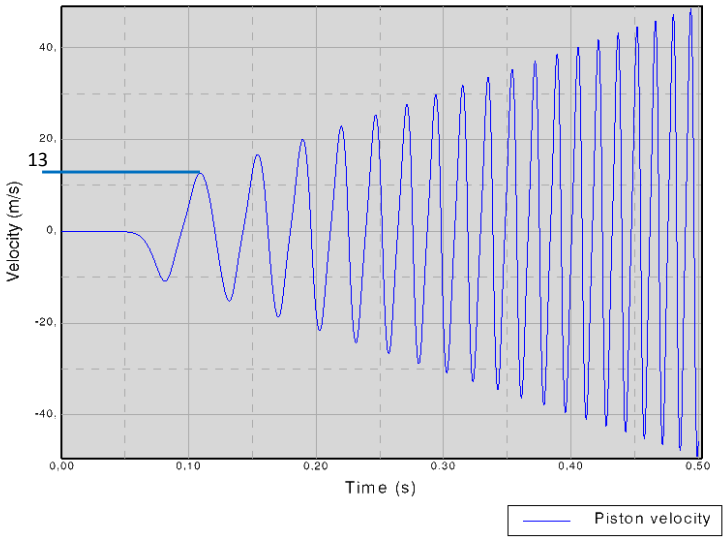

Figure 7. Piston velocity with large piston diameter

This chapter illustrates the results before and after implementing the characteristics and properties of a product automatically into the system simulation. The automation is done with a separate developed application.

5 CONCLUSION AND OUTLOOK

This paper is about modification and variant design. In this design process the designer needs to be supported to save time and costs in order to ensure market shares. This leads to a competitive advantage for the company. Special focus in modification and variant design is set on the verification of the design results by using a system simulation. Such simulations describe the system behaviour by means of modelling languages like Modelica or Simulink. The problem with these languages is that the geometry of the product can be represented only with a limited degree of detail. Consequently, the geometry of the parts are simplified and represented with basic geometries. For example, in the system simulation a piston will be represented by a simple geometrical cylinder. Components like an hole-feature, ring grooves or a complex surface shape cannot be modelled in the system simulation. According to Walter et al. (2014) such abstractions lead to higher level of uncertainty in modelling and simulation. This leads to a limited quality of the simulation results. The quality of the results can be improved by integrating detailed geometrical information of the parts into the system simulation. Conventional CAD-systems provide no linkage between the physical models and the system simulation. In order to increase the quality of the simulation results the gap between the CAD-models and the simulation model need to be bridged to integrate detailed geometric information into the simulation. In the interest of saving time and costs this progress should be performed automatically.
According to Eigner et al. (2005) the left wing of the V model diagramme can be divided into the four steps Requirements, Functional, Logical and Physical. These steps are integrated in the so called RFLP-environment of the CAD-system CATIA V6. In this paper special focus is set on the logical and the physical layer. The first describes the system behaviour with the modelling language Modelica. The latter exhibits a conventional CAD-environment to design products. To bridge the gap between the domain-specific physical layer and the multi-domain logical section a basic procedure is developed. This procedure explains the fundamentally steps which need to be performed to integrate the detailed information of the product into the system simulation to gain more detailed simulation results. The mentioned procedure exhibits the workflow which is integrated in a developed application. The use case presented above shows an application example and imitates a modification

or variant design task or a further iteration step of the V model product development process. A simplified model of a single-cylinder combustion engine is used and shows the simulation results before and after the automated integration of the changed physical information into the system simulation. In the use case presented above the length and the material of the piston represent the changed physical information. The regarded parameter of the system simulation results is the velocity of the piston. A change of the simulation results could be shown by comparing the velocity of the piston with different weight.
For the next phase of the project an automated optimization process could be implemented in order to find the best possible solutions for a given problem. For example the length and material of the piston as well as the injection parameters could be optimized with focus on a low-consumption combustion engine. Therefore the parameters in the physical layer would be changed by the optimization algorithm. After this step the developed application would identify the changes and integrates them into the system simulation. According to the simulation results the optimization algorithm can choose the values for the next optimization step. This process is repeated until the optimum is found.
In a further step automated manipulation of the design could be done according to the results of the system simulation. This may be possible by implementing features, templates and knowledge based engineering tools. For example the pin joining the piston and the connection rod is a standard part. The force on this pin can be figured out with the results of the system simulation. By changing the physical information of the singe-cylinder engine the combustion force could be increased. Consequently excessive forces on the pin could occur. If this is the case knowledge based tools could decide to implement the next larger standard bolt to prevent the pin and the overall product from damage. In a future phase of this project not only variant and modification design may be in focus. The current point of view may be extended to make new constructions possible. Based on given requirements and a functional principle of the product a huge part of the design process may be automated by implementing knowledge based engineering tools. This may be possible through a linkage between all layers of the extended V model by Eigner et al. (2005) and a consequent use of standard parts.

REFERENCES

Albers, A., Alink, T., Matthiesen, S., Thau, S. (2008) Support of system analyses and improvement in industrial design trough the contact & channel model. In DS 48: Proceedings DESIGN 2008, the 10th International Design Conference, Dubrovnik, Croatia.

Breitsprecher, T., Röhner, S., Wartzack, S. (2010) Konzept für eine simulationsgetriebene-wissensbasierte Produktentwicklung im Umfeld mechatronischer Produkte, 21th DfX-Symposium, Buchholz (Germany), 23./24.09.2010, Conference Proceeding: TuTech Innovation, 221-233

Eigner, M., Gilz, T. and Zafirov, R. (2012) Proposal for functional product description as part of a PLM solution in interdisciplinary product development, International Design Conference (DESIGN 2012), Dubrovnik (Croatia), University of Zagreb/The Design Society, pp. 1667-1675.

Elmqvist, H. (1978) A structured model language for large continuous systems. Lund Institute of Technology.

Feldhusen, J., & Grote, K. H. (2013). Pahl/Beitz Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung. Methoden und Anwendung. Springer-Verlag.

Fritzson, P. (2004) Principles of Object-Oriented Modeling and Simulation with Modelica 2.1. Piscataway, New York: Wiley-IEEE Press.

Karnopp, D. C., Margolis, D. L., & Rosenberg, R. C. (2012). System Dynamics: Modeling, Simulation and Control of Mechatronics: Modeling, Simulation, and Control of Mechatronic Systems. John Wiley & Sons.

Lindemann, U. (2009) Methodische Entwicklung technischer Produkte. Berlin: Springer.

Vajna, S. (2009) CAx für Ingenieure - Eine praxisbezogene Einführung. Berlin: Springer.

Verein Deutscher Ingenieure (VDI) (2004) VDI 2206 Entwicklungsmethodik für mechatronische Systeme - Design methodology for mechatronic systems. Düsseldorf: Beuth Verlag.

Verein Deutscher Ingenieure (1993) VDI 2206 Design methodology for mechatronic systems. Berlin: Beuth.

Verein Deutscher Ingenieure (1993) VDI 2221 Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte. Berlin: Beuth Verlag.

Walter, M., Storch, M., & Wartzack, S. (2014). On uncertainties in simulations in engineering design: A statistical tolerance analysis application. Simulation, Vol. 90, No. 5, 547-559.

Weber, C. (2005). CPM/PDD–an extended theoretical approach to modelling products and product development processes. 2nd German-Israeli Symposium on Advances in Methods and Systems for Development of Products and Processes, Berlin: Fraunhofer IPK, pp. 159-179.