NORDDESIGN CONFERENCE
REYKJAVIK, ICELAND, 12-14 AUGUST 2024

Automated Retrieval of Functional Relations from Machine Parts CAD Data

Jonas Fastabend, Daniel Roth, Matthias Kreimeyer

Instutute for Engineering Design and Industrial Design, University Stuttgart

Abstract: Existing products are analysed in order to develop the functionality of new products. This is done with effortful manual methods like the contact and channel model. Digital product models, particularly CAD models, are widespread in development processes. To reduce the effort for analysing products functionality, this research is investigating whether functional relations can be automatically revealed from CAD models. The physical architecture is created from the CAD model and combined with standardised main functions of individual components to form a database. This database is analysed for two specific questions regarding the functionality of a product. The design knowledge required to interpret the database regarding these questions is defined, formalised and implemented as an algorithm. Where appropriate, machine learning methods are used for implementation. The resulting algorithms reveal functional relations regarding relative motions and mechanical energy transfers. The approach is evaluated with the CAD model of a gearbox.

Keywords: Design Automation, Reverse Engineering, Functional Modelling, Computer Aided Design (CAD), Artificial Intelligence (AI)

1 Introduction

The purpose of a technical product is defined by the product function (Pahl et al., 2007). The physical components are defined by the product design in such a way that the product function can be fulfilled (Matthiesen, 2021). The functional architecture structures the product function hierarchically and is materialised by the physical architecture of the components (Pahl et al., 2007). Product function, structure, associated functional architecture and physical architecture of the components are therefore elementary components of widely used development models (Gero and Kannengiesser, 2004; Pahl et al., 2007; Farid and Suh, 2016). The development of products is based generationally on predecessor products and similar products (Matthiesen et al., 2018), so that existing products are analysed in order to develop the functionality of new products (Vajna et al., 2018; Matthiesen, 2021). It is therefore common practice to generate the functional architecture from a physical architecture through reverse engineering (Weck et al., 2011; Farid and Suh, 2016). Methods such as the contact and channel approach (C&C²) are useful for modelling functional relations from geometric product models, but must be carried out manually by the engineer (Matthiesen, 2021) so that they do not lead to a processable data structure for functional relations (Grauberger et al., 2020). This causes effort for each individual modelling of functional relations and limits the reuse of given product models. The CAD model of a product describes the product geometry and, unlike functional models, can be processed digitally (Vajna et al., 2018). So, the CAD model might form a suited and executable input for a data based functional analysis. A data-based link between functional product analysis and the CAD model of a product was therefore identified as a research gap (Grauberger et al., 2020). This publication examines how functional relations can be automatically retrieved from the CAD model of a product. The aim is to define a method for the automated extraction of functional relations from CAD data. A prototype for functional relations retrieval is then designed, implemented and tested based on this method. The defined method is validated by the capabilities of the prototype for functional relations retrieval. The CAD model of a gearbox with 27 components enables the evaluation of the prototype.

2 Research Method

As a framework for the research method, the relevant state of the art on functional-design relationships is first briefly described. The overall embodiment-function relationship (EFR) of a product is given by the product architecture (Pahl et al., 2007). The product architecture consists of the functional architecture, the physical architecture and the transformation relationships between the physical and functional architecture (Pahl et al., 2007). The functional architecture describes the logical model of a products function hierarchically, as well as the flow of inputs and outputs of the product behaviour (Buede, 2009). The physical architecture describes the components of a system and the relationships between them (Farid and Suh, 2016). Individual EFRs can be identified by analysing the product geometry using C&C² approach (Matthiesen et al., 2018; Grauberger et al., 2020). This approach is based on the fact that components realise a functional interaction via pairs of working surface pairs (Grauberger et al., 2020). In this way, the fulfilment of a function can be distributed across several components. These working surfaces often correspond to the contact surfaces that two components share with each other. This is the case, for example, when a mechanical force is transmitted by direct physical contact. In order to be able to determine EFRs from the shape, a semantic must be formed for the product function to be analysed

(Matthiesen et al., 2018). It must be clarified which question about the functionality of the product is to be answered. When analysing a gearbox, for example, it makes a difference whether the lubrication sealing concept or the torque transmission is being examined. Different engineering experts with different expertise are required for both scenarios. The formation of EFRs therefore requires a specific question about the functionality, a model of the physical shape of the product and an engineer with the necessary knowledge to interpret the geometric model semantically (Matthiesen, 2021). As this research follows an automated approach, not all aspects of a components shape can be considered. Therefore, the extensive modelling of EFRs is limited to the modelling of functional relations. With regard to this publication, a functional relation is defined as quantitative information on the fulfilment of a function through the emergent interaction of a set of machine parts.

The CAD data of a product is an automatically processable version of the design documentation (Siemens, 2016). Information on implicit product knowledge like the product functionality is integrated into the CAD model and can be retrieved (Krahe et al., 2021). If it is therefore possible to convert a heuristic for extracting functional relations into an algorithm, functional relations can be generated automatically from the CAD data of a product. This argumentation can be confirmed by the realisation of a prototype. The research objective is therefore to define a method for the automated extraction of functional relations and to validate it with a working data-based prototype for identifying functional relations from CAD data. This leads to the following research question:

How must data processing be structured in order to automatically retrieve functional relations from the CAD data of a mechanical product?

The method in Figure 1 is a hypothesis for answering this question and forms the methodological basis for building the prototype. It is implemented and evaluated for two types of functional relations in order to identify the possibilities and limitations of this approach. The method is briefly explained below.

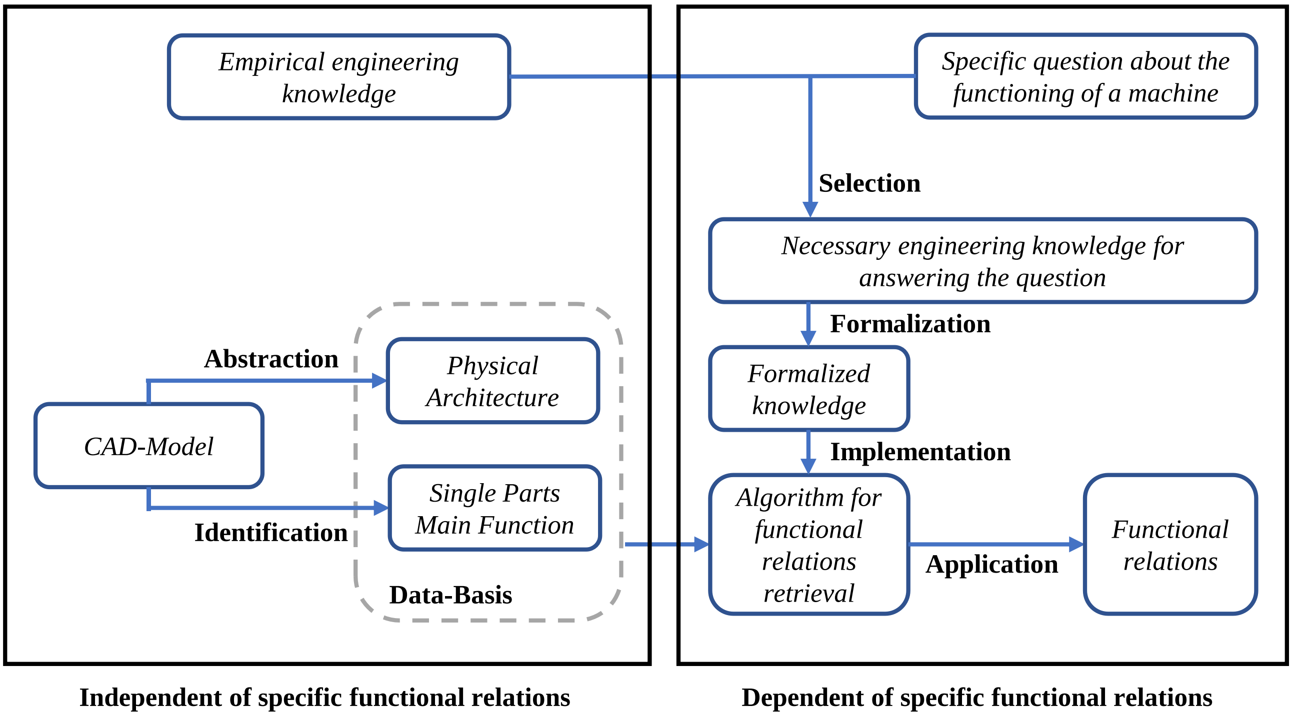

Figure 1: Method for automated retrieval of functional relations from CAD-Data

Three relevant input variables for the identification of functional relations are derived from the state of the art (Matthiesen, 2021). The process requires a geometrical design documentation which is realized through using the CAD model. A specific question about the products functionality and the necessary engineering knowledge to answer this question basedon the geometrical product design documentation is required as well. The method is separated into two parts, depending on whether the implementation of individual steps relies on a specific functional relations question. On the one hand, the preprocessing of the CAD model must enable the transfer between the geometrical and the functional domain. On the other hand, the preprocessing of the CAD model must map the interactions between machine components so that the modelling of cross-component functional relations is subsequently possible. Both processing steps should be universally valid in order to be valid for different functional relations and different CAD models. The interactions of the components are therefore modelled in a physical architecture. This physical architecture abstracts the CAD model and enables a statement to be made about which components have physical contact. This information is implicitly contained in the CAD data of a product. Hence, there is a potential to automatically retrieve it by a suited algorithm. Component classes are formed in order to generalise the geometry to function transfer for different products and functional relations. In mechanical engineering, solution concepts for certain functions are often repeated (Roth, 2001; Pahl et al., 2007), so that there are common component classes or standard parts whose function does not change for different products. Retaining structures, springs, ball bearings, screws and gears, for example, fulfil the same function in most cases. An individual product function is only created through the interaction of these individual elements. The component functions can therefore be assigned on the basis of the component type using rules, so that a catalogue of component class and component function is created for common machine components and component concepts. Subsequently, it can be analysed for specific functional relations whether superordinate and individual functions of a product can be identified based on the arrangement of these components in the physical architecture. The functions of the individual components combined with the physical architecture of the product form the data basis for further investigation of specific functional relations.

Answering questions about specific functional relations requires associated engineering knowledge to be identified so that the database can be interpreted correctly with regard to specific functional relations. This is done by modelling the cognitive process of an engineer when capturing the design model. As all members of the research group are graduated engineers, this step is carried out manually by the author and his colleagues for specific questions regarding functionality. This knowledge is then formalised in order to transfer it into an algorithm. Once the algorithm has been implemented, the functional relations can be identified from the CAD data automatically.

In order to check whether the entire method leads to the determination of functional relations, the generation of the database is first implemented using automatic data processing. Two functional relations are then defined that are to be recognised from the database. An algorithm is developed, programmed and tested by the method to extract them from the database. Automated extraction of functional relations demonstrates that the method works. The CAD model of a two-stage spur gearbox with 27 components serves as a use case (Vajna et al., 2020) for the validation. This use case is simple enough to enable a manual validation of the identified functional relations quality and at the same time has a sufficient number of functional and physical relationships between the components to verify the data processing. The CAD software Siemens NX and the associated programming interface NXopen are used to automatically analyse the gearbox with C# programs. Suitable Python libraries are applied for further data processing and .csv tables are used the data transfer.

3 Results

The exemplary realisation of the formulated method by means of a working prototype for functional relations retrieval forms the result of this research. Both the function-independent and the function-dependent part of the method are designed, implemented and tested. Firstly, the generation of the physical architecture and the overview of the component functions are explained and evaluated using the gearbox example. Subsequently, two specific questions on the functionality of a product are formulated and the function-dependent part of the method is designed, implemented and tested for both questions using the gearbox model.

3.1 Definition of the functionally independent data basis from the CAD data

According to Figure 1, the creation of the functionally independent data basis consists of the abstraction of the CAD data into a physical architecture based on the physical component contacts and the definition of main functions for standard components.

3.1.1 Abstraction of the CAD product model into a physical architecture

The physical architecture forms a suitable basis for analysing the functionality of a product (Farid and Suh, 2016). In order to create an abstraction of the physical shape of the product as a physical architecture that is independent of various functional relations, the definition based on the physical component contacts is a suited option. A common form of representation for this is the Design Structure Matrix (DSM) (Eppinger and Browning, 2012). The components are assigned symmetrically to the rows and columns in a matrix which in turn receives an entry when the components physically touch each other. The resulting matrix can be processed as an adjacency matrix so that the physical architecture can be represented as a graph (Diestel, 2018). This representation offers the advantage that algorithms from graph theory can be used to analyse component relationships. In order to generate the physical architecture of a CAD product model as a DSM or DSM graph based on the component contacts, it is necessary to extract from the CAD model which individual component models touch each other geometrically. This is realised by comparing the components in pairs. For each component pair, it is checked whether a two-dimensional or three-dimensional sectional geometry of the individual part models can be generated on the basis of the modelled 3D bodies. In the API interface of the Siemens NX CAD software, a function can be called that automatically checks this property for two 3D bodies. The resulting DSM is saved as a matrix in an Excel table and can be visualised as a graph using a Python script. The evaluation of the algorithm using the gearbox leads to an exact representation of the component contacts. The DSM graph does not contain any errors. The result for the gearbox is shown in Figure 2. The DSM is schematic in order to maintain the clarity of the representation. A node in the graph corresponds to a component in the CAD model. The colouring of the components matches the colouring of the corresponding nodes in the graph. The DSM is saved as csv. file and can be visualised as a graph using the Gephi program.

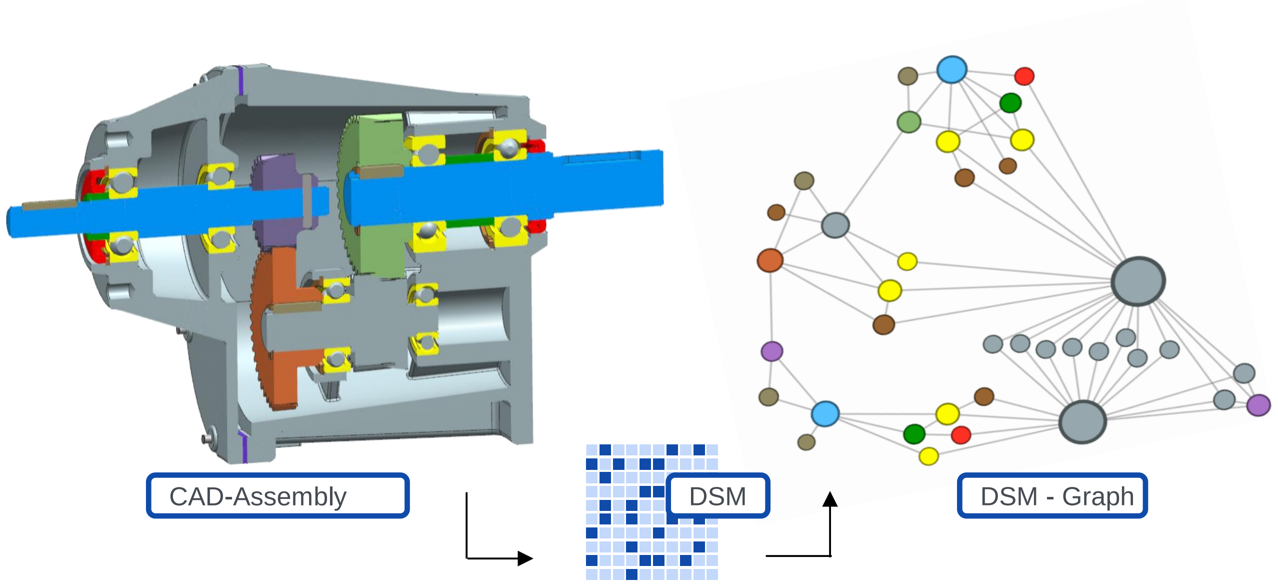

Figure 2: Generating the physical architecture automatically from a CAD-Assembly for the gearbox

3.1.2 Derivation of single part functions from the CAD single part models

In mechanical engineering, many components and solution concepts are used to fulfil a standardised function. Standard parts such as screws, springs or ball bearings fulfil a predictable function in the context of the entire product. Screws fix two components in place, springs couple two components dynamically and ball bearings enable rotational movements. However, there are also numerous standardised solution concepts for frequently occurring sub-functions in the context of the overall product. Holders spatially fix adjacent components; cam discs realise a cyclical motion sequence and housing parts transfer reaction forces to the environment. The overall function of a product results from the emergent interaction of the functions from individual components (Pahl et al., 2007). The use of such standard functions to identify functional relations of the entire product assumes that a large proportion of components fulfil a standard function. This limits the applicability of the method. For the identification of standard functions from individual part data, a tabular catalogue of standard components and associated functions is created. This is shown Table 1 for the components of the gearbox.

Table 1. Functions of the gearbox components

Component	**Function**
Shaft	Transmit torque
Gear wheel	Convert torque
Pinion shaft	Transmit torque; Convert torque
Housing	Taking up reaction forces; Spatial positioning of the powertrain; Store lubricant
Bearing	Enable rotation; Taking up reaction forces;
Radial shaft seal	Dynamic separation of lubricant and environment
Screw	Connecting components
Pin	Transmit shear forces
Static seal	Static separation of lubricant and environment
Feather key	Transmit shear forces
Retaining ring	Positioning components
Sleeve	Avoid abrasion

In this way, the problem of identifying component functions is reduced to identifying component classes. Two methods are available for identifying component classes (Winkler et al., 2023). The components can be classified on the basis of their designation or on the basis of their geometric properties. Both options were tested in the context of the study. The name of a component tells an engineer what type of component it is. It is therefore possible to classify components based on their name and this must be formalised for an algorithm. This formalisation of component classes based on the component names is rule-based using keywords, suffixes or prefixes that can be assigned to a specific class. Examples of keywords are "bearing", "seal" and "shaft". A corresponding catalogue for these keywords must be created manually, but can be used for various products and functional relations.

Based on geometric properties, two machine learning methods were considered for component classification. Firstly, a property vector is created from the geometry for each component using the Siemens NX API interface. Part properties for this are the number of edges, number of faces, number of bodies, body volume, body surface and volume of a bounding box. This property vector forms the basis for the classification. When the location of these vectors in the six-dimensional solution space is clear cut, a cluster analysis can be used as an unsupervised learning method to identify the components. A k-means clustering algorithm was tested for the gearbox example. However, the component classes are not clearly defined, as the number of clusters must be defined in advance. In addition, the individual clusters must be interpreted semantically after clustering. This approach is therefore more effortful than classification by component name. In addition to this approach to component identification, the classification can also be tested using a trained neural network (Krahe et al., 2021). This requires a training data set consisting of property vectors and classified CAD data. A short experiment has shown a general applicability but has a lack of classification accuracy. This can probably be improved with more extensive training data and optimised learning algorithms.

For the research structure of this publication, the classification of components by component name is the simplest and most robust method and is therefore used for the further identification of functional relations. However, the implementation effort for a scaled application to many CAD product models is relatively high, as the data processing would have to explicitly map all naming rules. For more extensive use cases, a better cost-benefit ratio can be expected for machine learning models, especially for well-trained neural networks because their output does not receive a manual evaluation. As shown in Table 1, twelve different component classes are installed in the gearbox, so that twelve naming rules are sufficient for identifying the components. The naming rules are implemented in such a way that different component classes can be selected for the application of further algorithms. The application of algorithms based on component classes and their functions can be easily automated. However, the decision as to which component classes contribute in which way to specific functional relations must be made in advance by an engineer for each question relating to the functionality of the product. This knowledge needs to be transferred in an algorithm for automated function retrieval for each specific functional relation.

3.2 Definition of functionally dependent algorithms for the identification of functional relations

The component functions and the component contacts can be derived from CAD data. The design of the subsequent process steps is based on a specific question about the functionality of the product. Therefore, a suitable question on the functionality must be defined at this point. A question is suitable if it can be answered by a formalised process on the basis of the DSM graph and the functions of standard components. Two requirements for a question on functional relations follow from the realised generation of the data basis from the CAD product model:

- Emergent functional relations between individual components are adequately represented by physical component contacts in the physical architecture if they are relevant for answering the question
- The functions of individual components contribute directly to answering the question, so that their identification can form the basis for the transfer between form and function

In order to demonstrate a broader applicability of the method from Figure 1, two questions on the functionality of a product are formed for further analyses and answered automatically by an algorithm. Each question is checked against the two requirements using the author's existing design experience and fulfils them. The following questions are analysed:

1. Through which components does the main flow of mechanical energy take place?
2. Between which components are there relative motions?

To answer the questions, the necessary design knowledge for the interpretation of the physical architecture and the component functions must be defined and formalised. The formal definition of this knowledge is then transferred into an algorithm and tested using the physical architecture and component functions of the gearbox. This procedure is first implemented for the first and then for the second question.

3.2.1 Use case 1: Identification of the main flow of mechanical energy

Functions are based on the main flows of energy, material and information (Pahl et al., 2007). If the main flow is present as mechanical energy, there is the potential to identify the components involved on the basis of the component contacts. At first it is to clarify what empirical design knowledge an engineer can use to recognise the main flow of mechanical energy in a product. Mechanical forces, torques and power are transmitted through physical contacts between different components. In addition, a mechanical force takes the shortest path through a mechanical structure (Ehrlenspiel, 2009). For the transmission of mechanical forces, torques and power, there are component classes that serve this function directly, such as levers, shafts, gears and housing parts. Other components, such as screws or retaining rings, only serve this function indirectly, as they connect or fix force-conducting components together. Components such as seals or sleeves do not contribute directly to the transmission of mechanical energy.

These considerations can be formalised on the basis of the physical architecture and the component functions. The physical architecture defines the existence of physical component contacts. Relationships in the physical architecture thus represent a pair of working surfaces of two components and can be interpreted as an interface for the transmission of forces and torques, as well as mechanical power. The main flow of energy thus corresponds to a path through the DSM graph. The beginning and end of this path are formed by the interfaces of the system for the input and output of mechanical power. The main flow of mechanical energy is identified when this path is found in the DSM graph. Due to the principle of direct force conduction (Ehrlenspiel, 2009), it is convenient to first look for the shortest path between the input and output of mechanical energy. However, since there is no guarantee that this principle is valid for any product, the combinatorial explosion of the path search must be limited in another way. This is done by eliminating components from the DSM graph whose main function does not contribute directly to the transfer of mechanical forces. In the gearbox, this concerns screws, retaining rings, sleeves and seals. The paths between the input and output shafts are then searched for in the remaining DSM graph. The formalised process is shown in Figure 3 for the energy flow of the gearbox.

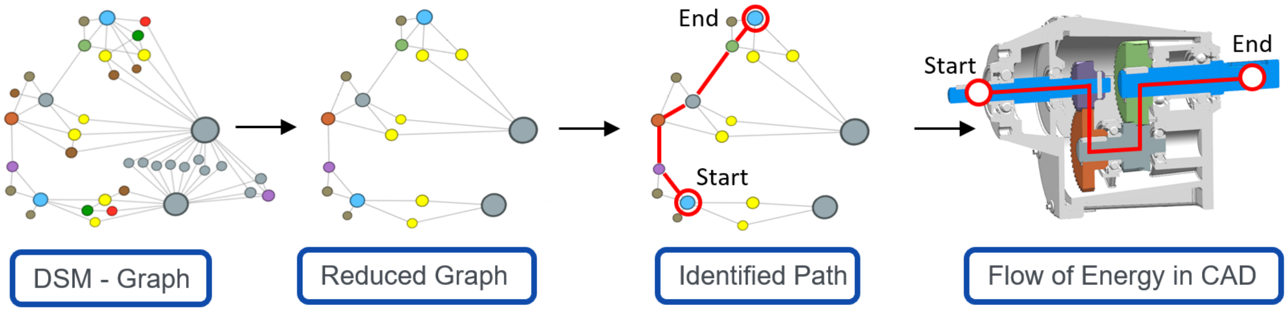

Figure 3: Automated retrieval of mechanical energy flows from the DSM-Graph and part functions

The technical implementation of the component elimination is done by a filter that operates on the basis of the component names. The paths are searched for in the reduced DSM graph using a brute-force algorithm starting at both interface components. As soon as both paths arrive at the same component, a path is found from the input component to the output component. For the gearbox, the shortest path corresponds to the main flow of mechanical energy. The path consists of the six components input shaft, gear wheel, gear wheel, pinion shaft, gear wheel and output shaft. The automated identification of functional relations can therefore be demonstrated for the formulated question using the gearbox example. However, feather keys and pins are not part of the identified path, although they are involved in energy transmission. The path found is therefore valid, but does not perfectly represent the flow of mechanical energy.

3.2.2 Use case 2: Identification of relative motions

The aim of the second use case is the automated identification of relative motions and is based on preliminary work of the author on the identification of components for additive manufacturing (Winkler et al., 2023). The first step is to clarify what empirical design knowledge an engineer needs to be able to recognise the distribution of relative motions based on the DSM graph and the component functions. The working surfaces of component contacts can either be statically positioned relative to each other or move relative to each other. Therefore, an entry in the DSM corresponds to either a static or a dynamic coupling of components. There are also components whose main function requires an internal relative motion of their working surfaces to each other, such as ball bearings, springs and dampers. There are also components, such as the sprags of a freewheel or dynamic seals, which are always located at the boundary of two static component groups. The elimination of these components from the DSM graph promotes the separation of the physical architecture into static component groups so that there are no more relative motions within individual component groups. However, this does not yet lead to the complete formation of component groups without relative motions. On the one hand, this may be due to inaccuracies in the CAD data and the DSM graph. On the other hand, there are still relationships in the DSM graph between two components that represent a relative motion, such as rolling gears. By eliminating components with internal relative motion, the coupling within static component groups is greater than the coupling between component groups with relative motion. The concept of modularity can be applied to a graph to decompose such a topology into individual communities of the graph (Blondel et al., 2008). In this context, a community describes a connected set of individual nodes of the graph. The formula for calculating the modularity of a graph is given in formula (1) and a more detailed explanation of the mathematical relationships can be found in the relevant literature (Newman, 2006).

$$Q = \frac{1}{2m}\sum_{i,j}\left[A_{i,j} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j) \quad (1)$$

m	=	Sum of edge weights in the graph
k	=	Summed adjacency edge weight of a node
$A_{i,j}$	=	Entry of the adjacency matrix
c	=	A community of the graph
δ	=	Kronecker Delta

If the modularity Q of the graph is maximised, the coupling between the communities is weak in relation to the coupling within the communities. Therefore, this mathematical property is suitable for partitioning the graph into static groups based on the reduced DSM graph as data input. The analytical optimisation of modularity is associated with a very high computational effort, so that the Louvain algorithm is used as an unsupervised machine learning method to optimise the modularity of the graph and to form static component groups (Blondel et al., 2008). Once the graph has been broken down into components that represent the static component groups, the DSM graph contains a statement about the components between which relative motions exist and which components are statically coupled. Statically coupled components are part of the same community, components with relative motions are assigned to different communities. The entire algorithm was implemented and evaluated using the gearbox example. The various data processing steps are visualised in Figure 4.

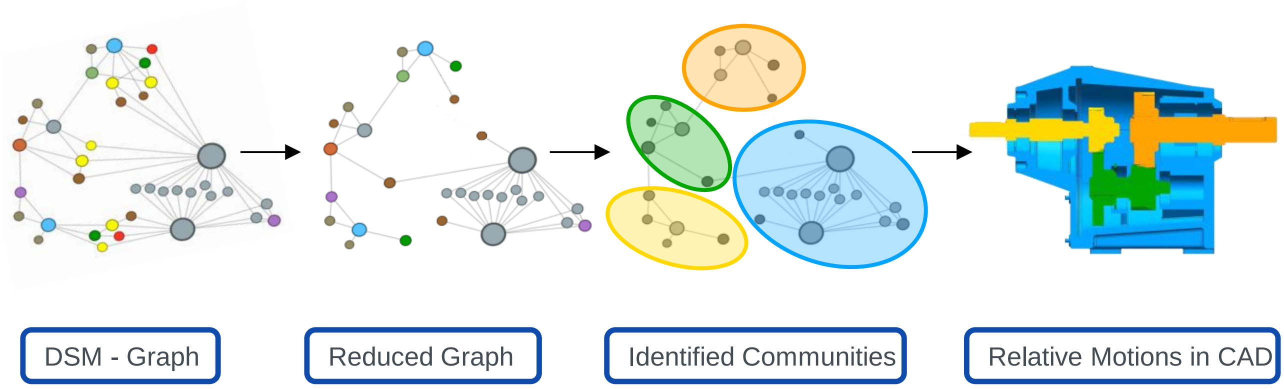

Figure 4: Automated retrieval of relative motions from the DSM-Graph and part functions

Relevant components for the separation of relative motions in the gearbox are the ball bearings and the radial shaft seals. Ball bearings and dynamic seals were therefore automatically removed from the DSM graph using a filter based on the component names. In the next step, the reduced DSM graph was broken down into various components using the Louvain algorithm so that modularity was optimised. The paths between the components of the graph were then removed. For the gearbox, the implemented algorithm resulted in an almost error-free distribution of relative motions. Only one retaining ring was assigned to the wrong component group. The formulated question about the functionality of the product could therefore be answered in good quality on the basis of the CAD data. However, the gearbox is a comparatively simple product and it cannot be concluded that the algorithm is generally valid for more complex use cases.

4 Discussion

Overall, the objective of the study was achieved. A prototype for the automated extraction of functional relations from the CAD data of a product was realised through the formalisation and implementation of engineering knowledge and evaluated using the CAD data set of a two-stage spur gear. Two specific questions about the functionality of the product can be answered. The main flow of mechanical energy and the relative motions of the product can be identified automatically. The proposed method for generating functional correlations was thus successfully implemented so that it sufficiently answers the research question. However, a general potential for the automated identification of any functional relations from the CAD data of a product cannot be concluded from the result. Various limiting factors must be considered with regard to further significance.

One limiting factor is the information content of CAD data. CAD data documents the physical shape, so it is primarily mechanical functional relations that can be recognised. If a function is solved electrically or by software, no statement can be made. In addition, the CAD model usually only depicts one operating state of the product. In the case of a manual gearbox, for example, it is confusing if all potentially force-transmitting gears in the CAD model are modelled with a physical body contact at the same time. This phenomenon is already known in the modelling of EFRs and must also be considered here (Matthiesen, 2021). Theoretically, a CAD model would be required for each operating state. Furthermore, the modelling may be imprecise so that the component contacts in the model do not represent the real product. Further limitations of the solution space result from the generated database, so that the question of functional relations cannot be freely chosen. The interactions of the components are only modelled by physical component contacts. Functional relations for the flow of forces or mechanical energy can be mapped well using this. This does not apply to the flow of electrical or thermal energy, for example. Other main functional flows, such as matter or signal, are also likely to be difficult to extract from physical component contacts. In the methodology used, the transfer between form and function is realised through the main functions of standard components. This limits the solution space of the method to products that are primarily composed of classic machine parts such as brackets, gears, shafts, ball bearings or screws. Assigning the main functions to the components also implies an initial manual effort, so that scalability as an important motivator for process automation (Emmer, 2023) only arises through the use of the component catalogue for different functional relations and different CAD models. In order to avoid repeating this manual effort for every data set, it is crucial to restrict the application to common component concepts for any CAD data sets. The method can therefore not be used for highly innovative products with special components without manually entering their specific functions. Two requirements for formulating questions about functional relations were defined in this paper. By fulfilling these requirements, a question is validated against the generated data set. However, this does not guarantee that an algorithm can be created to answer the question automatically. For each question, a heuristic must first be found that can be formalised and implemented as an algorithm. So far, there is no method for the specific formulation of questions, so this criterion is tested by trial and error. Outside of academic research, the question should therefore be of great benefit to product development, so that the automation effort required to answer it is worthwhile. This is not a limitation for the chosen research structure, as the focus is on proving the feasibility in principle within a methodological framework. However, statements on the extraction of any functional relations from CAD data remain uncertain.

Despite the limitations mentioned, the result shows that it is possible in principle to extract functional relations from CAD data with the given method. The implemented generation of the physical architecture and the use cases for the flow of mechanical energy and relative motions identification offer several interfaces to make operational processes in digital product development more efficient. The generation of the physical architecture by the implemented algorithm is a very stable method for generating the DSM graph, as it is based on geometric interferences. If products are developed in the context of model-based systems engineering (MBSE), a corresponding SysML diagram for the physical architecture can be generated automatically from the CAD data. This might be helpful for system engineers, when it comes to the conceptualization of a new product generation. In addition, the identified component contact surfaces might be a suited input for finite element simulations to simplify modelling preprocessing steps. This can be helpful for simulation engineers. The identification of mechanical energy flows and relative motions make it easier for a new designer, user or customer to understand the machine. In addition, the identified relative motions can facilitate modelling steps for kinematic simulations of a machine. The overarching method is a suited tool for anyone who wants to model functional relations automatically based on CAD data. Further use cases for example might be the identification of hydraulic or pneumatic energy transfers. That might be helpful for computational fluid dynamic simulation. The automated extraction of functional relations therefore appears to be an academically relevant field of research, as the resulting functional relations can be transferred to various scenarios in the operational development of products and have the potential to make these processes more efficient. Further research into the existing possibilities and limitations of the developed functional relations retrieval thus appears to make sense. It might be an approach to integrate simulation models into the database, as they document the behaviour of the product and can therefore usefully supplement the data basis for identifying functional relations.

5 Summary and Outlook

As part of this publication, a method for the automated extraction of functional relations from the CAD data of a product was developed and tested for two specific questions relating to the functionality of a spur gearbox. The method was used to develop, implement and successfully validate a procedure for analysing mechanical energy transmission and a procedure for detecting relative motions using the gearbox CAD model.

An algorithm for abstracting the CAD data as a physical architecture was implemented on the basis of physical component contacts. In addition, a procedure was developed for identifying standard components based on the component designations in order to assign a standardised component function to components. This creates a database consisting of a physical architecture and a component catalogue with component functions to answer specific questions about how a machine works. Two requirements were formulated for the identification of specific functional relations and two questions on the functioning of a machine were defined that fulfil these requirements. The necessary design knowledge was formulated and formalised for the investigation of mechanical energy transfers and the detection of relative movements, considering the existing database. Algorithms were implemented for both applications in order to solve the tasks automatically. The algorithms were tested with the CAD data of the gearbox. Components with mechanical energy transfer and relative motions between the components were recognised reliably overall, meaning that the validation was successful. The method for retrieving functional relations was therefore successfully implemented and it was shown that the functional relations of a product can be automatically extracted from the CAD data of a product.

Future research work can address the identified limitations of the existing methods and expand them. For example, the database can be expanded to include simulation data in order to enable more comprehensive conclusions to be drawn about how products function. An extension with SysML diagrams on the physical and functional architectures of products can also be an approach. It may be possible to generate functional models in SysML from the CAD data of a product in order to reduce the effort for MBSE. For the physical architecture, this transfer has already been reduced to an operational problem. Methodologically, research can be conducted into how suitable questions about the functionality of products can be formulated, as these can only be assessed a posteriori using the current method. In this research work, machine learning methods were only used selectively for individual sub-processes. The comprehensive use of machine learning methods for the generation of functional models could also be an interesting issue if it is possible to generate sufficiently large data sets on the physical and functional properties of products. It can also be investigated to what extent existing methods for optimizing designs, such as the design for assembly approach according to Boothroyd and Dewhurst, can be optimized through the automated extraction of functional relations.

References

Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. J. Stat. Mech. 2008 (10), P10008. 10.1088/1742-5468/2008/10/P10008.

Buede, D.M., 2009. The engineering design of systems: Models and methods. John Wiley & Sons, Hoboken, N.J.

Diestel, R., 2018. Graph Theory. Springer, Berlin.

Ehrlenspiel, K., 2009. Integrierte Produktentwicklung: Denkabläufe, Methodeneinsatz, Zusammenarbeit, 4th ed. Hanser, München.

Emmer, M., 2023. The Seven Principles Of Scalability. Forbes 2023. https://www.forbes.com/sites/forbesbusinesscouncil/2023/12/05/the-seven-principles-of-scalability/, 13.02.2024

Eppinger, S.D., Browning, T.R., 2012. Design structure matrix methods and applications. MIT Press, Cambridge Mass.

Farid, A.M., Suh, N.P., 2016. Axiomatic Design in Large Systems. Springer International Publishing, Cham.

Gero, J.S., Kannengiesser, U., 2004. The situated function–behaviour–structure framework. Design Studies 25 (4), 373–391. 10.1016/j.destud.2003.10.010.

Grauberger, P., Wessels, H., Gladysz, B., Bursac, N., Matthiesen, S., Albers, A., 2020. The contact and channel approach – 20 years of application experience in product engineering. Journal of Engineering Design 31 (5), 241–265. 10.1080/09544828.2019.1699035.

Krahe, C., Kalaidov, M., Doellken, M., Gwosch, T., Kuhnle, A., Lanza, G., Matthiesen, S., 2021. AI-Based knowledge extraction for automatic design proposals using design-related patterns. Procedia CIRP 100, 397–402. 10.1016/j.procir.2021.05.093.

Matthiesen, S., Grauberger, P., Hölz, K., Nelius, T., Bremer, F., Wettstein, A., Gessinger, A., Pflegler, B., Nowoseltschenko, K., Voß, K., 2018. Modellbildung mit dem C&C²-Ansatz in der Gestaltung - Techniken zur Analyse und Synthese. KIT SCIENTIFIC WORKING PAPERS.

Matthiesen, S., 2021. Gestaltung – Prozesse und Methoden, in: Bender, B., Gericke, K. (Eds.), Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung, 9th ed. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 397-467

Newman, M.E.J., 2006. Modularity and community structure in networks. Proceedings of the National Academy of Sciences of the United States of America 103 (23), 8577–8582. 10.1073/pnas.0601602103.

Pahl, G., Beitz, W., Feldhusen, J., Grote, K.-H., 2007. Engineering design: A systematic approach, 3rd ed. Springer, London.

Roth, K., 2001. Konstruieren mit Konstruktionskatalogen, 3rd ed. Springer, Berlin.

Siemens, 2016. Getting Started with NXopen.

Vajna, S., Weber, C., Zeman, K., Hehenberger, P., Gerhard, D., Wartzack, S., 2018. CAx für Ingenieure. Springer Berlin Heidelberg, Berlin, Heidelberg.

Vajna, S., Wünsch, A., Pilz, F. (Eds.), 2020. Siemens NX für Einsteiger - kurz und bündig, 4th ed. Springer Vieweg, Wiesbaden, Heidelberg.

Weck, O.L. de, Roos, D., Magee, C., Charles Vest, C., 2011. Engineering systems: Meeting human needs in a complex technological world. MIT Press, Cambridge, Mass.

Winkler, M., Jacobs, G., Fastabend, P.J., Konrad, C., 2023. Automated Identification of Geometric Structures with Potential for Functional Integration, in: Lachmayer, R., Bode, B., Kaierle, S. (Eds.), Innovative Product Development by Additive Manufacturing 2021. Springer International Publishing, Cham, pp. 243–256.

Contact: Jonas Fastabend, Institute for Engineering Design and Industrial Design, jonas.fastabend@iktd.uni-stuttgart.de