9th International Conference on Mass Customization and Personalization – Community of Europe (MCP - CE 2020)
Re-innovating Business in the Digital Era
September 23-25, 2020, Novi Sad, Serbia

DESIGN AUTOMATION CASE STUDY: TRASH RACK CLEANER

Paul Christoph Gembarski
Institute of Product Development, Leibniz Universität Hannover,
An der Universität 1, 30823 Garbsen, Germany

Abstract: *Product configurators are a widely used tool to let customers specify an individual variant that best fits to their requirements. The basis for this is a product model that has well defined degrees of freedom and options and so describes the possible solution space. Related to this are design automation systems that aim at automating the design process in order to mimic human designers and assist them in routine design activities. While a lot of configurators and design automation systems is settled in the field of business to consumer applications, some implementations for complex goods like in plant engineering are reported. The present article reports about a design automation system for trash rack cleaners which are used in water management. The basis for the system is a knowledge-based engineering system which uses spreadsheet, constraint and macro technology to create a fully detailed CAD model of the trash rack cleaner.*

Key Words: ***Design Automation, Product Configuration, Knowledge-Based Engineering, Trash Rack Cleaner***

1. INTRODUCTION

Product configurators are a widely used tool to let customers specify an individual (product) variant that best fits to their requirements [1]. Applications range from online sales configurators for clothing, furniture and cars to configurations systems for industrial goods [2]. Technical product configurators are usually linked to a design system and deliver an artefact description like a bill-of-material, a virtual product model or even a complete set of production data [3].

The basis for configuration systems is a product model that has well defined degrees of freedom and options, e.g. a parametric model [4-6]. Related to this are design automation systems that aim at automating the design process in order to mimic human designers and assist them in routine design activities [7-9]. Both configurators and design automation systems can be seen as particular knowledge-based engineering (KBE) systems which use representations and models of engineering knowledge to e.g. automate routine design tasks [10].

KBE is rather not new, one of the first documented applications already dates back to the early 1980ies: The R1/XCON was implemented as rule-based configurator for VAX data processing units. Rule-based means that it was constructed by IF-THEN-ELSE statements [11]. The system was active for about nine years and contained at its end more than 17.500 rules and more than 31.000 components as building blocks for configuration [12]. In the beginning of the 1990ies, XRAY, a configurator for x-ray systems, was implemented based on the PLAKON expert system shell which used constraint satisfaction techniques as knowledge model [13]. The special feature of XRAY was its ability to jointly configure product, software as well as service features and so can be understood as an early instantiation of configurable product-service systems [14].

The aforementioned systems were documented in an extensive way, including descriptions of the underlying knowledge models, reasoning mechanisms and architectural considerations. Although today's computer-aided design systems allow for integrating KBE without the necessity of using specialized software, there is only a limited number of applications reported from e.g. aerospace and automotive engineering or niche design activities like fixture design [15, 16]. Focusing on detailed application examples, the number of implementations that exceed simple machine element assemblies, single parts or conceptual product models is rare.

This article aims at contributing to close this gap and discusses the implementation of a design automation system for trash rack cleaners which are used in water industry. The basis for the system is a knowledge-based engineering system which uses spreadsheet, constraint and macro technology to create a fully detailed CAD model of the trash rack cleaner. The system was modelled as part of an industrial case study. Its primary objective was to evaluate the effort and competences necessary for creating a knowledge-based in comparison to a traditional singular product model in an engineer-to-order company. A secondary objective of the study was to compare the prior ETO process with the KBE supported one regarding lead time, design artefact quality and solution space.

The remainder of the article is organized as follows: In the subsequent section 2, the theoretical background is presented which comprises knowledge-based engineering, basic engineering problem-solving tasks and the implementation of knowledge-based CAD models. In section 3, the industrial case study is outlined and the design of trash rack cleaning systems is introduced. Section 4 then shows the implementation before section 5 presents the discussion and conclusion of the case study. The final section 6 contains the summary and further research possibilities.

2. THEORETICAL BACKGROUND

2.1. Knowledge-Based Engineering and Design

Fig. 1 shows the basic setup of a knowledge-based system which can be understood as computer-aided problem-solving tool [8]. Knowledge-based engineering (KBE) systems use knowledge representations and models for the automation of design processes, for dimensioning and design optimization or for decision-support [10]. KBE can be comprehended as evolutionary step in computer aided engineering which is created by the combination of object-oriented programming, artificial intelligence and computer aided design (CAD) systems [17].

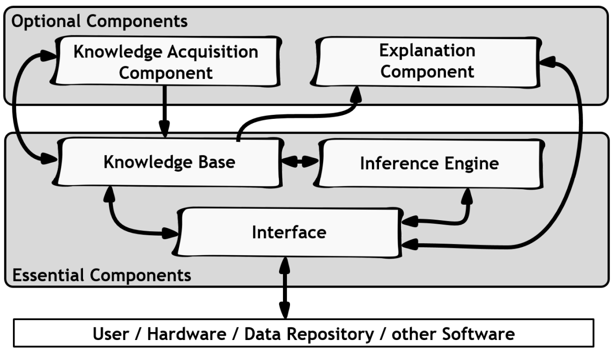

Fig. 1. *Main Components of a Knowledge-Based System (acc. to [8])*

To automatically perform design tasks, a KBE system must have the ability of reasoning [12]. Therefore, two very basic kinds of knowledge need to be implemented [18]: (1) Domain knowledge constitutes a solution space in which a particular solution for a defined set of requirements can be found. It is modelled e.g. by parameter constraints, formulae and design rules for product models [4, 8, 19]. (2) Control knowledge states how this solution space is explored and integrates reasoning techniques [3, 7, 12].

A particular KBE implementation is knowledge-based CAD [4]. It integrates design rules, dimensioning formulae, spreadsheets, macros and interactive applications into the CAD system itself [20]. Although rather not new, only single detailed reports about applications exist. Exemplarily stand works from design synthesis in aerospace and automotive engineering [15], conceptual or configuration design in plant engineering [3] or niche design activities like automating fixture design [16]. All of these approaches have in common that knowledge artefacts as well as models have to be implemented explicitly.

Especially technical product configuration systems also belong to KBE systems. Here, a common master model which uses formalized engineering knowledge is modelled as an image of the product solution space instead of a single variant [21].

Design automation systems differ from this: These are able to fully automate a design task from specification over conceptual design to detailed design and definition of product and production data [7, 22]. A relatively new development in this field is, in contrast to traditional reasoning, the implementation of more complex problem-solving mechanisms and artificial intelligence. One example is the application of multi-agent systems for the analysis and optimization of CAD models regarding design guidelines [23, 24]. In this context an agent represents a software entity that operates autonomously without intervention of a human user to complete a task [25]. In order to do so, an agent needs to perceive the environment relevant to his task, to react on changes to the environment and know about consequences [26].

2.2. Problem-Solving in Knowledge-Based Engineering and Design Automation Systems

The decomposition of problem-solving tasks into single steps and mechanisms usually involves synthesis and analysis operations [27]. From a top-level perspective, KBE systems have to deliver artefact descriptions by either one or a combination of the three basic synthesis tasks [7, 18]:

1. Synthetic design is designing a system that meets specified requirements. These are first formulated by the user and then operationalized by the KBE system. Hereby hard requirements enable the KBE system to filter possible system designs that have been generated on the basis of knowledge about system creation. Soft requirements enable the system to evaluate and rank multiple valid system designs [18].
2. Configuration means creating a system out of fully predefined building blocks that are integrated via standardized interfaces [28]. Although the building blocks used in configuration themselves do not have any degree of freedom, a very large solution space can be created with an appropriate design [29]. The decisive factor here is the number of combination interfaces and rules. From an information science point of view, configuration tasks can be written and solved as constraint satisfaction problem [30, 31].
3. Parametrization aims to eliminate degrees-of-freedom (e.g. with regard to dimensions or activation of individual design elements or components) in a variable product model, step by step by setting parameter values [32]. The same as for configuration, a basic representation for parametrization is a constraint satisfaction problem [13].

2.3. Implementation of Knowledge-Based CAD

A common basis to build knowledge-based CAD models is a parametric design system which allows automatic change propagation, the embedding of necessary domain knowledge and thus the design of a solution space (fig. 2) [4, 5, 19, 33].

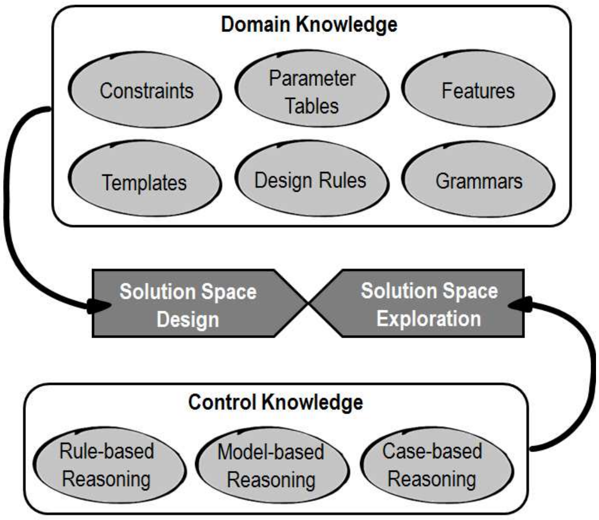

Fig. 2. *Knowledge Modelling in KBE and KBD* [4]

Parameters can be linked by arithmetic, logic or geometric constraints [34]. Especially in complex CAD models that contain a huge number of model features and relations, a control and configuration concept is beneficial that structures the way how parameters are calculated, related and referred to each other [4, 5]. One way is to integrate mathematical formulae, e.g. for dimensioning of machine elements, another way is to externalize the parameter calculation e.g. into a spreadsheet application [19, 21, 35]. The latter commonly offers additional mathematical and statistical operations compared to those implemented in the CAD system itself. Another advantage is that relevant data for the definition and specification of components, e.g. parameter tables, can be stored on different worksheets and then be linked by use of matrix-operations like VLOOKUP in MS Excel [36]. Additionally to organize e.g. multiple parts within a CAD assembly, a skeleton model can define component positioning or superordinate geometrical characteristics, e.g. based on the structural design [21].

Other ways of formalizing design knowledge is templates that have to be understood as reusable, updatable building blocks in a virtual prototype [6]. As such, geometry templates are further distinguished into rigid and variable geometry templates. The first represent carry-over-parts or library components that have additional process parameters available which cover knowledge about application, design interfaces or technical data in general. The latter is taken as predefined starting point for embodiment or detailed design that includes all necessary design rules and features. Beside geometry templates there also exist structural and functional ones. A structural template includes e.g. a basic generic product structure and different delimited physical design solution spaces. So, the design process is parallelized in a standardized way. A functional template represents the implementation of specific problem-solving methods and simulation tools, additionally to the geometry description [19].

The implementation and formulation of design rules strongly depends of the CAD system. Basically, a rule is an IF-THEN-ELSE-statement known from software development and is used to e.g. exclude model features in relation to parameter values or to execute commands in order to modify the geometry [35, 36].

Design grammars are another, but not widespread way of formalizing engineering knowledge. In the academic field, synthesis systems for electricity pylons, wheel rims, heat sinks or robot kinematics have been implemented [37]. The idea of a design grammar in this context is to implement coherent synthesis operations through a vocabulary of elements in combination with a set of alteration rules. Applying this on a starting design, which is either developed by a human expert or created by algorithms, a huge number of alternative designs may be generated by the system, thus this approach is also known as generative design [38].

In contrast to domain knowledge, control knowledge determines the way a solution space is explored [4]. This can be done e.g. by the integration of reasoning mechanisms:

1. Rule-based reasoning also relies on IF-THEN-ELSE-statements, like discussed in context with design rules. The major difference is that in rule-based reasoning the rules are linked to a decision tree or decision network [12]. Although it’s one directional and simple nature, instantiation and loops form complex rule bases where rules activate sub-ordinate rules or exclude them from further processing [15]. It is often reported that rule-bases with several hundred rules are difficult to maintain [11].
2. Model-based reasoning operates on a logical, physical or resource allocation/consumption model [12]. A common implementation uses constraint networks and constraint satisfaction techniques [30]. A constraint represents the relationship between two model elements and may have a rule for value assignment [3]. Values applied to the constraint network can then be propagated, which means that the values of all other model elements are calculated on the basis of them. The representation allows to model the relationships in an undirected way so that it is of no importance which variables are given and which are searched. From a logic point of view, the constraint network can be written as equation system [13].
3. Case-based reasoning mimics the human ability to work with analogies [39]. It uses an implicit knowledge representation of problem statement-solution-pairs, so a case can be understood as previously solved problem. Depending on the degree of maturity of the reasoning mechanism the system is either limited just to search for exactly matching existing solutions or to find solutions that are similar [36]. This must be expressed by e.g. a classifiation or indexing system or a mathematically formulated distance measure [40]. Usually, such a similar solution needs to be modified , e.g. by a human expert, to be applicable to a new problem. After Validation, this case is stored again in the case base so that the system can be considered as self-learning [32].

3. PROJECT DESCRIPTION

The case study reported here was carried out during an industry research project from water industry. In this section, the trash rack cleaner design, which had to be automated, is described as well as the project boundaries and project scope.

3.1. Trash Rack Cleaner and Main Components

In water industry, trash rack cleaners are important systems that ensure a high efficiency of e.g. pumping stations, sewage treatment plants or hydroelectric power plants [41]. The inlet screen of a hydraulic structure, which usually is built as trash rack, serves as a coarse filter to retain alluvial debris or flotsam such as leaves and branches to protect e.g. turbines [42]. A regular removal of this material is necessary to minimize flow losses. For mechanical cleaning, stationary or mobile trash rack cleaners are used for this purpose, which are either excavating machines with articulated arm, telescopic boom or cable cleaners [43]. The example modelled in this case study corresponds to the third design (Fig. 3).

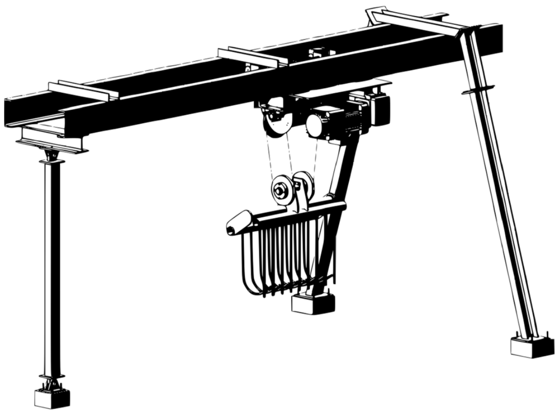

Fig. 3. *Cable Trash Rack Cleaner with Gripper Head*

Depending on the design and purpose, different gripper types are used. Grabs for floating materials are used for picking up larger flotsam such as branches near the surface while cleaning rakes with closable cage are applied for the removal of debris, sludge and lighter flotsam along the whole trash rack [44]. The cleaning process begins with the lowering of the rake onto the trash rack. There the rake slides from the water surface to the sole and carries the flotsam with it. At the reversal point the cage is closed and the rake is raised again to the initial height. With the rake closed, the trolley travels to the disposal site and releases the removed flotsam there. Usually, the rake is narrower than the trash rack, so this cycle is repeated with small overlap until the whole rack is cleaned.

The engineer-to-order process for such a trash rack cleaner involves the following basic problem-solving steps for the main components:

(1) The design starts with dimensioning of the hoist unit. Here, the rack's grid space, the width of the rake and the cleaning length (dimension water surface to sole) are the characteristic variables. The cleaning length determines, together with the load of the rake and the material to be conveyed, the dimensioning of the lifting gear and of the cables. Depending on the expected loads and the installation space, versions with double winch or pulley block are possible.

(2) The determined design and size of the hoist is one of the input variables for the design of the trolley. The trolley carries, in addition to the hoist, the travelling drive and the hydraulics for the cylinder of the grab. The arrangement of the aggregates primarily depends on the width of the trolley. This in turn has a reference to the width of the grab, since only limited diagonal pull is permitted for safe reeling of the steel cables. The placement of the aggregates are usually based on a best-fit old project, which are adapted to the current case.

(3) Support structures and tracks for the trolley are primarily based on constructional conditions at the installation site (use of portal supports or cantilever), the disposal site (disposal on the ground, conveyor belt or into skip) and operating mode. Furthermore the stiffness of the construction is a determining factor. Support structures and tracks are individually developed for each project and verified with a test statics.

3.2. Project Boundaries

The company, where this case study was carried out, is set up as a small-series manufacturer. Besides other hydraulic structures constructions and components, such as vanes, weirs and pumping stations, approximately 30 cable trash rack cleaners are designed and manufactured as engineer-to-order projects per year.

Sales are made worldwide, apart from the trash rack cleaning system itself, commissioning and the installation at the destination is offered by the company. In addition to the engineer-to-order business there is a standard program of four variants for small pumping stations and hydroelectric power plants. The design engineering is carried out in-house, only the verification of the statics of support structures and overhead track is executed by external experts. The production is also largely carried out in-house, the entire system is then assembled and tested before being dispatched to the construction site for final assembly.

3.3. Project Scope

Although the boundary conditions for each project regarding construction site, material handling, service strategy and regional standardization differ and justify an engineer-to-order approach, sub-processes in engineering can be considered as stable and routine activities. This is particularly true for the design of hoist and trolley. In order to evaluate the automation potential and thus the reduction of lead time in sales and engineering, a case study was initiated. As boundary conditions was agreed to keep existing software and not to introduce new systems into the design department as well as to formalize engineering knowledge in plain text and simple algorithms that can be maintained by the design staff without additional in-depth software engineering competences.

After a potential assessment that was carried out along two design projects of the industry partner, the scope of the project was defined as:

- Implementation of a prototype KBE systems for the hoist unit;
- Definition of templates for characteristic trolley designs that use the hoist configuration as input;
- Integration of KBE functionalities, in particular calculation routines, to the design of support structures and the track in order to facilitate the statics report;
- Preparation of a KBE system for configuration of standard variants of the trash rack cleaner.

In an additional step, the performance of the corresponding new project approach should be tested with three old completed projects. At these chosen projects, engineering times have been recorded in detail so that a comparison is possible regarding cycle time and quality of the design artefacts.

4. IMPLEMENTATION OF COMPONENT AND TRASH RACK CLEANER KBE-SYSTEMS

Since the design department is the main user of the later KBE system, it was chosen to directly work with the available design tool, namely the CAD system Autodesk Inventor. Regarding domain knowledge integration, all in sect. 2.3 introduced mechanisms but grammars are offered. For reasoning, rule-based and model-based approaches are available, realized via iLogic, a proprietary script language, VBA macros and a spreadsheet integration.

4.1. Hoist Unit

As stated in the initial situation sub-section, the design of the hoist unit largely follows the recommendations and processes described in the German standards for material handling and conveying technology (e.g. DIN 15020-1). Based upon the given formulae, equation systems can be formalized and integrated into a spreadsheet configurator. The main worksheet (Fig. 4) contains input parameters and controls for the selection of factors and coefficients for each type of hoist. Other worksheets contain parameter tables for machine elements, auxiliary calculations and additional plausibility checks.

As input parameters, the user enters the design parameters of the corresponding standards in addition to the desired dimensions of the rake. The configurator first determines the dimensions of the cables and the cable drum on the basis of the forces that occur, taking into account the prescribed safety factors. Then the shaft of the cable drum is dimensioned.

Based on the calculated minimum diameter in the bearing seat (critical cross section), a reasoning algorithm determines the diameters of the individual shaft segments and checks them in each case against the design parameters of the intended machine elements. The reasoning is directly integrated into the spreadsheet via VBA macros and operates on the cell values. If the necessary verifications are not met, the algorithm increases the relevant shaft diameter and adapts the design accordingly. Depending on the predefined maintenance interval the configurator chooses the rolling bearings for the cable drum and calculates its dimensions (Fig. 5). In addition to the drum, the spreadsheet determines the parameters for all assembly groups (supports, couplings, etc.) and selects a suitable electric motor from a catalog.

Lastangaben			
Parametername	Wert	Einheit	Kommentar / Zwischenergebnis / Plausibilitätsprüfung / Hinweis
Laufzeitklasse	v_3		4,1 bis 8,0 h/d im Jahr
Lastenkollektiv	leicht		selten größte Last
Triebklassengruppe	2_m		
Transporte	Übliche Transporte		
Nennfestigkeit der Drähte	1770		
Beiwert c	0,095	mm/N^(1/2)	0 entspricht "Nicht vorhanden"
Masse Last	200	kg	ab Gesamtgewicht von 650kg Doppelrolle
Masse Gesamt	414	kg	unter Gesamtgewicht von 650kg und keine Kurven einfache Laufkatze
Sicherheitsfaktor	6	oE	
Leistungsangaben			
Parametername	Wert	Einheit	
Drehmoment	386	Nm	
Gesamtkraft S	4873,608	N	
Umdrehungen/Minute	8,8	1/min	
Hubgeschwindigkeit	0,88	m/min	
Laufzeit für gesamte Auf-/Abwicklung	4,63102314	min	
Trommelwirkungsgrad	0,99	oE	
Lagerlebensdauer in Stunden	155891,299	h	
Lagerlebensdauer in Jahren	53,3874312	y	
Angaben Seiltrommel			
Parametername	Wert	Einheit	
minimaler Seildurchmesser d	7	mm	

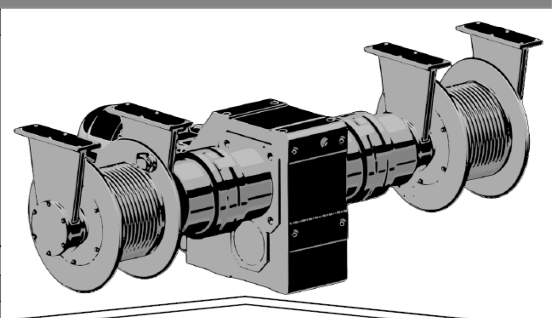

Fig. 4. *Configurator for Hoists with Double Winch*

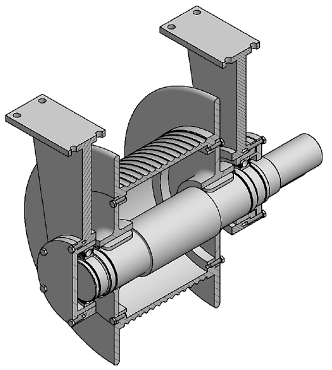

Fig. 5. *Cable Drum (Sectional View)*

The calculated outputs are stored as transfer parameters and then passed to the CAD assembly. There, the assembly model is regenerated with all parameters accordingly, machine elements are exchanged following the selections of the spreadsheet and another plausibility check is carried out to check the assembly. Machine elements are modeled as iPart families so that the change of a size can be executed by iLogic rules (Fig. 6).

In addition to the model, a set of technical drawings is generated and relevant machine elements are verified mathematically.

A total of four different hoists were implemented as independent KBE systems. In addition to the hoist with double winch, these include a hoist with single winch for low lifting loads and versions with single and double winch with pulley block to reduce loads and installation space.

4.2. Trolley

The configured hoist is the starting point for the synthesis of the trolley and provides the initial mounting dimensions for the base frame. This frame carries, in addition to hoist unit, also the rollers and the cross-travelling drive as well as the supply hydraulics for the closing mechanism of the rake.

Due to the de facto standardization of the four hoists, it was possible to create CAD templates for the architecture of the trolley. The selection of an architecture depends primarily on the possible installation space. First of all, it must be decided whether the hoist and cross-travelling drive must be mounted on the same level in order to achieve a flat architecture or a slim trolley as shown in Fig. 7 is beneficial, where the hoist is mounted under the travelling drive. This is e.g. advantageous for trash rack cleaners with monorail track and short cleaning length. The selection of the CAD template relies on a rule base that argues selection characteristics and is implemented into a top-level assembly in the iLogic-language.

```
If Parameter("__Layout:1", "S:01")<10 Then
        iPart.ChangeRow("DIN625_hohe_Tragzahl:1", "DIN625-1-600" &
                Parameter("__Layout:1", "S:01") & "-2RSR")
        iPart.ChangeRow("DIN625_hohe_Tragzahl:2", "DIN625-1-600" &
                Parameter("__Layout:1", "S:01") & "-2RSR")
Else
        iPart.ChangeRow("DIN625_hohe_Tragzahl:1", "DIN625-1-60" &
                Parameter("__Layout:1", "S:01") & "-2RSR")
        iPart.ChangeRow("DIN625_hohe_Tragzahl:2", "DIN625-1-60" &
                Parameter("__Layout:1", "S:01") & "-2RSR")
End If
```

Fig. 6. *iLogic Rules for Component Exchange (Excerpt)*

As special feature, each CAD template for a trolley has a reasoning mechanism implemented which determines the center of gravity for the current configuration. If it is not centered under the track, the algorithm starts to shift the individual subassemblies in a controlled way in their position (the mechanism is comparable to a truth-maintenance system) until an optimally balanced trolley is found. As verifications, the weld seams between base frame and mounting brackets are calculated as well as the service life of the wheel blocks.

Other equipment such as hose guides, piping or fastening elements are not included in the assembly templates due to the variability of the sub-assembly positions. The template is then further detailed by a human designer.

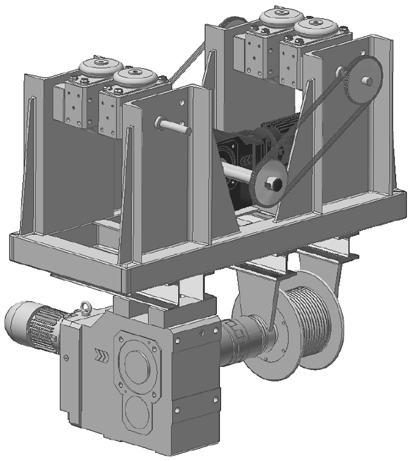

Fig. 7. *Slim trolley with under-mounted hoist for monorails (illustration without casing)*

In total twelve variants of the trolley are available, which can be paired with either monorail or double beam tracks.

4.3. Support Structure and Track

The parameterization of the track is carried out on the basis of the weights of the trolley, grab and expected flotsam. The trolley travels either on a HEM wide flange beam as monorail track or in two UPE profiles as external double beam track. For both track types model-based design wizards were implemented within Inventor. The user specifies the lane type and the position of the support points. The system then determines the required profile cross sections in order not to exceed a given deflection of the track. For each support point, the corresponding structure, e.g. a cantilever, must be inserted from a library. The system sets the cross-sections according to an assignment matrix between the track profiles and the support structures. Adjustments to structural conditions, such as different heights of the foundations or the drilling patterns for bolting to the foundations must then be carried out manually. For this purpose, parametric skeletons were stored in the respective subassemblies. Due to the numerous variation parameters, the strength verification of the supports is not carried out by the system but externally following the design as before.

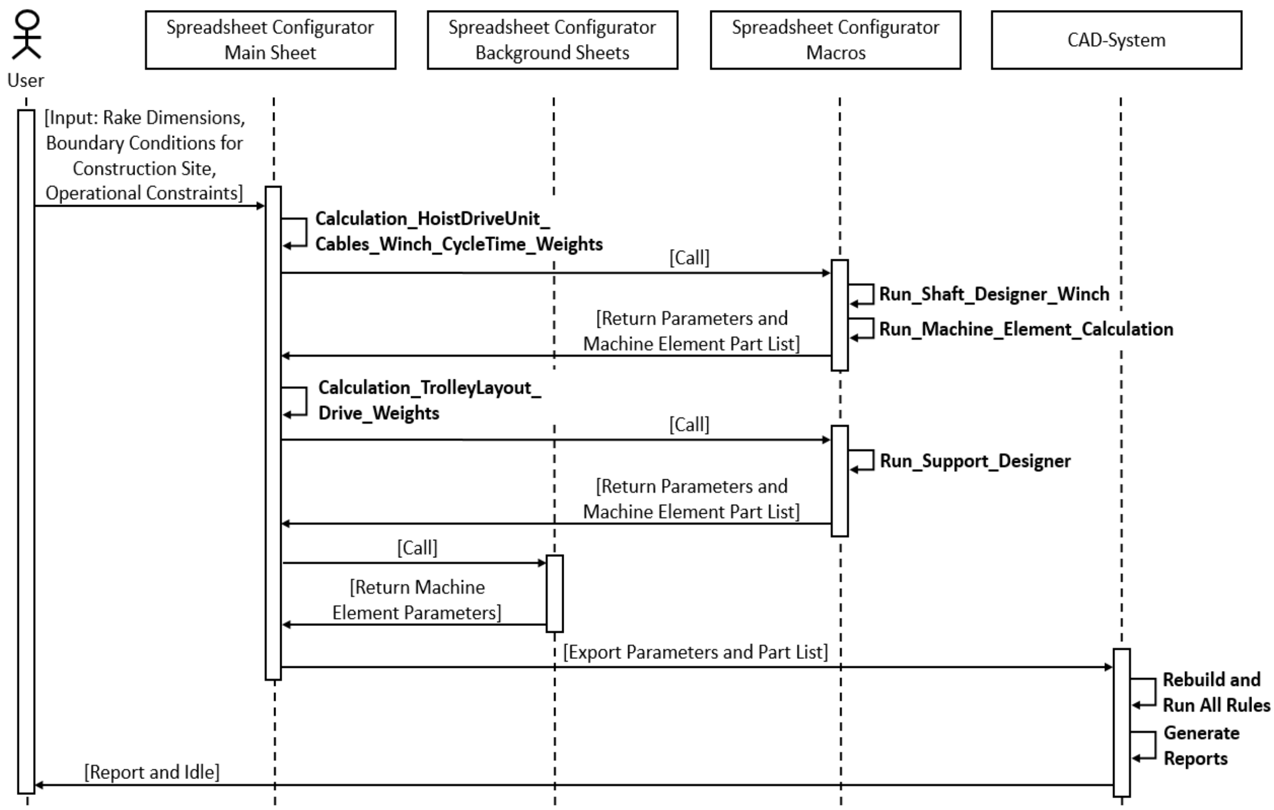

Fig. 8. *Sequence Diagram for Trash Rack Cleaner Configuration*

4.4. Trash Rack Cleaner Configuration System

For the integration of the individual KBE systems into a standard configurator, the hoist configurator was selected as the basis and supplemented by the other parts. The central configuration tool in the KBE system is thus again a spreadsheet, which has been extended by macros. The configuration process is shown in Fig. 8 as sequence diagram.

The starting point of the configuration is the user input regarding number of rakes, rake dimensions, boundary conditions of the construction site (areas for support foundations, installation space, environmental conditions) and operational constraints (information on flotsam, frequency of use, operation time, manual or automatic operation). Based upon this the spreadsheet executes the hoist configuration and calls macros for shaft design and machine element choice, which have the same functionality as in the hoist configurator. Afterwards the calculated and optimized dimensions are coded as parameters and then passed back to the spreadsheet. The same counts for the machine elements. In the next step, the trolley is chosen based upon a rule-base, similarly to the template choice mentioned above. After calculation and when the weights are determined, the macro for support design is called and based upon the information on the installation boundary conditions, the beams and tracks are dimensioned and verified. Relevant parameters are returned to the spreadsheet and all machine element parameters are collected from the background worksheets containing the corresponding standards. All geometrical and topological parameters are then passed to the CAD system, where the force rebuild command and the execution of the internal iLogic rule base is triggered. After the generation of all verification reports, the configuration process ends.

It has to be noted that this standard configurator contains only a few template configurations for the supports. These have been synthesized from a former project overview and represent nearly 60% of projects that have been realized in the former five years.

Final check and further detailing of the sub-assemblies is carried out manually afterwards. To those tasks belong the electrical and hydraulic equipment including hose routing, power supply and control system, placement of fasteners and finalizing the housing.

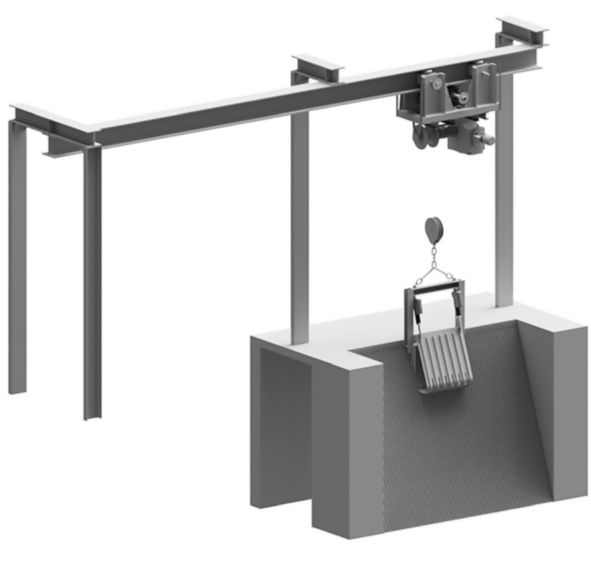

Fig. 9: *Configured Trash Rack Cleaning System (Illustration without Cables, Casing, Foundations, Top Plates and Screw Connections)*

The standard configurator is also accessible to the employees of the sales department for rapid calculation and tendering. Therefore, additional plausibility checks and a scheme for pricing were implemented. A configuration of the trash rack cleaning system generated by the KBE system is shown in Fig. 9.

5. DISCUSSION AND CONCLUSION

The implemented systems basically fulfilled the expectations of the industry partner. From a result perspective, the engineer-to-order process was supplemented by two configuration processes. The first involves the three single KBE systems, operated by a human designer. This process allows degrees of freedom in particular for the support structures and tracks. The second configuration process refers to a standardized product model and less degrees of freedom but covers already a majority of the solution space for many projects.

Regarding the cost/benefit dimension, the average processing time of the design task could be reduced from 140 to 30 hours already after the initial modelling of the solution space. From these, about 24 hours were spent on the detailing of the trolley and another four on the detailing and verification of the track. The basic configuration of the trash rack cleaning system can be generated by the KBE systems within two hours, so that incoming inquiries in the sales department can be answered immediately and precisely. Subsequent adaptation of the design to the actual conditions on the construction site (e.g. deviating executed foundations) can also be investigated and carried out in an accelerated manner. The implementation of the KBE systems by an experienced knowledge engineer took about three times as much design time as a "classical" single variant design, the costs for KBE system implementation are therefore amortized again within the first year after deployment.

During the design automation project, the industry partner was able to form an innovation cell since the design department was freed from routine activities. This innovation cell identified a demand for cleaning systems, which are able for cornering in order to save installation space and realize more complex operations. A corresponding prototype trolley was developed already using routines and methods from the existing KBE systems, in particular the dimensioning routines. After testing, the new trolleys were implemented into new KBE systems by project staff following the above principles (Fig. 10).

Despite the success of the project, there are still limitations. First, manufacturing knowledge was only integrated to a very limited extent into the KBE systems (verification of weld seams). The small-series character and the high flexibility in production allows large degrees of freedom here and as a consequence there was no additional benefit considered in restricting design parameters or formalizing assembly procedures.

Second, the level of detail of the final CAD models is not 100%. Fasteners were included into the CAD assemblies only where necessary for verification, major parts of the electrical engineering were skipped. Here also, the industry partner's design team estimated either no real advantage for raising the quality of the designed artefacts or the cost/benefit ratio was apparently not profitable.

Third, the digital master models of the main-components of the trash rack cleaner are functional but monolithic. All parameters and all component occurrences are hardcoded in the KBE system what makes a transfer to another application difficult or results in change expenditures respectively. The exchange of components, or e.g. the integration of other electrical drives, also leads to an interference with the knowledge model and thus the code of the KBE systems. Nonetheless of the digital master character of the assemblies, the rebuild times are satisfactory.

Fourth, the engineering environment is constituted only by two standard tools of a mechanical engineer, a CAD and a spreadsheet system. No further analysis systems, like finite element analysis etc., were included. In this particular case this was acceptable because the verifications all can be done via traditional mechanical calculations.

The implementation of the KBE systems and the corresponding new procedures in order processing had an impact on the business model of the industry partner as well. On the one hand, the evaluation of previous projects and enquiries to the sales department lead to the extension of the standard program. In particular trolleys and rakes could be standardized or quickly derived from the configurators so that only the support structures and tracks had to be designed individually for the specific project. On the other hand, the market share could be increased because the response times in technical sales got short and the quality of tender documents, since generated in parts by the KBE systems, raised. The implementation of KBE systems for other business segments is planned.

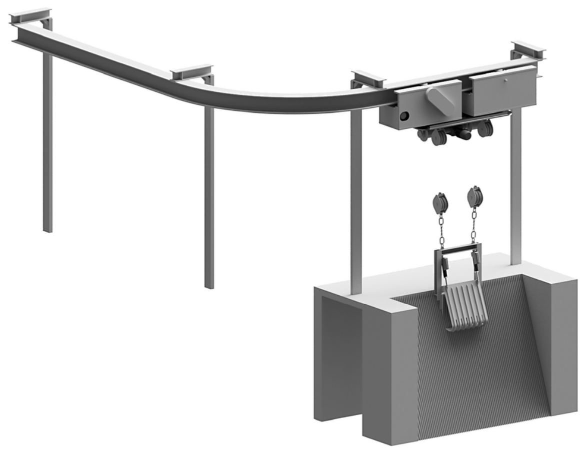

Fig. 10: *Configured Trash Rack Cleaning System for Cornering (Illustration without Cables, Casing, Foundations, Top Plates and Screw Connections)*

6. SUMMARY AND FUTURE RESEARCH

This contribution addresses the implementation of a design automation system that is integrated into a standard CAD systems. Therefore, a case study was presented and discussed which shows such a design automation system for a trash rack cleaner. The knowledge model was integrated into spreadsheets and the CAD assembly itself, using available functionalities like parameter constraints, integration of formulas and design rules as well as the use of API-based macros.

This case study offers multiple avenues for further research. The transfer from the initially modelled trolleys to the ones for cornering showed deficiencies in maintaining the knowledge model and adapting the KBE system. An interesting question is if an ontology as mediator between CAD system and knowledge model could provide additional functionalities. This could also simplify the addition of new model elements.

Another question focusses on product data management. Since the industry partner is experienced in engineer-to-order projects, all data management processes are aligned to that. In the end, a copy of the digital master is stored for each ordered trash rack cleaner for documentation. Regarding product data management of CAD-based configurators or design automation systems the question is what actually needs to be stored. From the author's point of view, it could be sufficient to save input variables and the version number of the design automation system since the input of the same parameters must lead to the same configuration.

7. REFERENCES

[1] C. Forza, F. Salvador, *Product information management for mass customization: connecting customer, front-office and back-office for fast and efficient customization*, Basingstoke, United Kingdom: Palgrave Macmillan, 2007.

[2] P. Blazek, C. Streichsbier, C., and M. Partl, *Configurator Database Report 2016*. Morrisville, USA: lulu.com, 2017.

[3] K. Kristjansdottir, S. Shafiee, and L. Hvam, "How to identify possible applications of product configuration systems in Engineer-to-Order companies," *International Journal of Industrial Engineering and Management*, vol. 8, no.3, pp. 157-165, 2017.

[4] P.C. Gembarski, R. Lachmayer, "Solution Space Development: Conceptual Reflections and Development of the Parameter Space Matrix as Planning Tool for Geometry-based Solution Spaces," *International Journal of Industrial Engineering and Management (IJIEM)*, vol. 9, no. 4, pp. 177-186, 2018.

[5] J.J. Shah, "Designing with parametric cad: Classification and comparison of construction techniques" *Geometric Modelling – Proceedings of the Sixth International Workshop on Geometric Modelling, Tokyo, Japan, December 7–9, 1998*, pp. 53-68, 2001.

[6] J.J. Cox, "Product Templates - A Parametric Approach to Mass Customization" *CAD Tools and Algorithms for Product Design*. Berlin, Heidelberg, Germany: Springer, pp. 3-15, 2000.

[7] P.C. Gembarski, and R. Lachmayer, "Business Models for Customer Co-Design," in *Proceedings of the 7th International Conference on Mass Customization and Personalization in Central Europe (MCP-CE 2016)*, 2016, pp. 76-83.

[8] A.A. Hopgood, *Intelligent systems for engineers and scientists*. Boca Raton, USA: CRC press, 2016.

[9] T. Brockmöller, P.C. Gembarski, I. Mozgova, and R. Lachmayer, "Design catalogue in a CAE environment for the illustration of tailored forming," in *Proceedings of the 59th Ilmenau Scientific Colloquium*, Ilmenau, 2017, pp. 11-15.

[10] W.J.C. Verhagen, P. Bermell-Garcia, R.E.C. van Dijk, and R. Curran, "A critical review of Knowledge-Based Engineering: An identification of research challenges," *Advanced Engineering Informatics*, vol. 26, no. 1, pp. 5-15, 2012.

[11] J. McDermott, "R1: A rule-based configurer of computer systems," *Artificial intelligence*, vol. 19, no. 1, pp. 39-88, 1982.

[12] D. Sabin, R. Weigel, "Product configuration frameworks-a survey," *IEEE intelligent systems*, vol. 13, no. 4, pp. 42-49, 1998.

[13] R. Cunis, A. Günter, and H. Strecker, *Das PLAKON-Buch: ein Expertensystemkern für Planungs- und Konfigurierungsaufgaben in technischen Domänen*, Berlin, Heidelberg, Germany: Springer, 1991.

[14] P.C. Gembarski, and R. Lachmayer, "Product-service-systems – what and why developers can learn from mass customization," *Enterprise Modelling and Information Systems Architectures*, vol. 13, no. 16, pp. 1–16, 2018.

[15] G. La Rocca, "Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design" *Advanced Engineering Informatics*, vol. 26, no. 2, pp. 159-179, 2012.

[16] I. Boyle, Y. Rong, and D.C. Brown, "A review and analysis of current computer-aided fixture design approaches," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 1, pp. 1-12, 2011.

[17] C.B. Chapman, and M. Pinfold, "The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure," *Advances in Engineering Software*, vol. 32, no. 12, pp. 903-912, 2001.

[18] G. Schreiber, B. Wielinga, R. de Hoog, H. Akkermans, W. Van de Velde, "CommonKADS: A comprehensive methodology for KBS development" *IEEE expert*, vol. 9, no. 6, pp. 28-37, 1994.

[19] M. Hirz, W. Dietrich, A. Gfrerrer, and J. Lang, *Integrated computer-aided design in automotive development*. Berlin, Heidelberg, Germany: Springer, 2013.

[20] S.K. Chandrasegaran, K. Ramani, R.D. Sriram, I. Horváth, A. Bernard, R.F. Harik, W. Gao, "The evolution, challenges, and future of knowledge representation in product design systems," *Computer-aided design*, vol. 45, no. 2, pp. 204-228, 2013.

[21] H. Li, P.C. Gembarski, R. Lachmayer, "Template-Based Design for Design Co-Creation" *Proceedings of the 5th International Conference on Design Creativity (ICDC2018)*, Bath, United Kingdom, 31.01.-02.02.2018, 2018.

[22] M. Bibani, P.C. Gembarski, and R. Lachmayer, "Ein wissensbasiertes System zur Konstruktion von Staubabscheidern," in *DFX 2017: Proceedings of the 28th Symposium Design for X*, 4-5 October 2017, Bamberg, Germany, 04.10.-05.10.2017, 2017.

[23] E. Dostatni, J. Diakun, D. Grajewski, R. Wichniarek, and A. Karwasz "Multi-agent system to support decision-making process in design for recycling," *Soft Computing*, vol. 20, no. 11, pp. 4347-4361, 2016.

[24] P.C. Gembarski, "On the conception of a Multi-Agent Analysis and Optimization Tool for Mechanical Engineering Parts," in G. Jezic, J. Chen-Burger, M. Kusek, R. Sperka, R. Howlett, and L. Jain (eds) *Agents and Multi-Agent Systems: Technologies and Applications 2020. Smart Innovation, Systems and Technologies, vol 186*. 2020, pp. 93-102.

[25] N.R. Jennings, and M.J. Wooldridge, *Agent Technology - Foundations, Applications, and Markets*. Berlin, Heidelberg, Germany: Springer, 1998.

[26] Y. Chen, Z.-L. Liu, and Y.-B. Xie, "A multi-agent-based approach for conceptual design synthesis of multi-disciplinary systems," *International Journal of Production Research*, vol. 52, no. 6, pp. 1681-1694, 2014.

[27] M. Biedermann, and M. Meboldt, "Computational design synthesis of additive manufactured multi-flow nozzles," *Additive Manufacturing*, vol. 35, 101231, 2020.

[28] N.R. Milton, *Knowledge Technologies*, Monza, Italy: Polimetrica sas, 2008

[29] B. Durhuus, and S. Eilers, "On the entropy of LEGO®", *Journal of Applied Mathematics and Computing*, vol. 45, no. 1-2, pp. 433-448, 2014.

[30] R. Barták, M.A. Salido, and F. Rossi, "Constraint satisfaction techniques in planning and scheduling", *Journal of Intelligent Manufacturing*, vol. 21, no. 1, pp. 5-15, 2010.

[31] M. Stumptner, G.E. Friedrich, G. E., and A. Haselböck, "Generative constraint-based configuration of large technical systems". *Ai Edam*, vol. 12, no. 4, pp. 307-320, 1998.

[32] P.C. Gembarski, M. Bibani, and R. Lachmayer, "Design Catalogues: Knowledge Repositories for Knowledge-Based-Engineering Applications.", in D. Marjanović, M. Štorga, N. Pavković, N. Bojčetić, and S. Škec (eds) *Proceedings of the DESIGN 2016 14th International Design Conference, The Design Society, 2016*, pp. 2007-2016.

[33] I. Stroud, H. Nagy, *Solid modelling and CAD systems: how to survive a CAD system*, Berlin, Heidelberg, Germany: Springer, 2011.

[34] S. Vajna, *CAx für Ingenieure: eine praxisbezogene Einführung*, Berlin, Heidelberg, Germany: Springer, 2009.

[35] W. Skarka, "Application of MOKA methodology in generative model creation using CATIA." *Engineering Applications of Artificial Intelligence*, vol. 20, no. 5, pp. 677-690, 2007.

[36] P.C. Gembarski, "Three Ways of Integrating Computer-Aided Design and Knowledge-Based Engineering." *Proceedings of the Design Society: DESIGN Conference*, vol. 1, pp. 1255-1264, 2020.

[37] F. Hoisl, *Visual, interactive 3D spatial grammars in CAD for computational design synthesis*. PhD-Thesis, Technical University Munich, 2012.

[38] K. Shea, R. Aish, and M. Gourtovaia, "Towards integrated performance-driven generative design tools," *Automation in Construction*, vol. 14, no. 2, pp. 253-264, 2005.

[39] M.M. Richter, and R.O. Weber, *Case-based reasoning*, Berlin, Heidelberg, Germany: Springer, 2016.

[40] A. Aamodt, E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches," *AI communications*, vol. 7, no. 1, pp. 39-59, 1994.

[41] H. Nøvik, L. Lia, and M. Wigestrand, "Horizontal trash rack–a new intake concept for efficient backflushing," *Journal of Applied Water Engineering and Research*, vol. 2, no. 2, pp. 81-90, 2014.

[42] S.R. Abt, T.E. Brisbane, D.M. Frick, and C.A. McKnight, "Trash rack blockage in supercritical flow," *Journal of Hydraulic Engineering*, vol. 118, no. 12, pp. 1692-1696, 1992.

[43] W. Radhuber, "Trashrack cleaning: The past the present the future," *Proceedings of the 15th International Seminar on Hydropower Plants 2008*, Vienna, Austria, 2008.

[44] A. Schoklitsch, *Handbuch des Wasserbaues: Zweiter Band*, Berlin, Heidelberg, Germany: Springer, 1962.

CORRESPONDENCE

Dr.-Ing. Paul Christoph Gembarski
Institute of Product Development,
Leibniz Universität Hannover,
An der Universität 1
30823 Garbsen, Germany
gembarski@ipeg.uni-hannover.de