

Evaluating a computational support tool for set-based configuration of production systems: results from an industrial case

Johannes Uglert, Sipke Hoekstra, Juan Jauregui-Becker

Department of Design, Production and Management

Faculty of Engineering Technology, University of Twente

Enschede, The Netherlands

j.m.uglert@utwente.nl

Abstract – This paper describes research conducted in the context of an industrial case, dealing with the design of reconfigurable, cellular manufacturing systems. Reconfiguring such systems represents a complex task due to the interdependencies between the constituent subsystems. A novel, computational tool was developed to support the production engineers in (sub)system configuration by enabling to consider multiple alternative configurations simultaneously. The tool was tested by applying it in two realistic system engineering problems and conducting interviews to evaluate its effects. The prototype was found to be an effective and efficient approach to support exploring, evaluating and selecting sets of system configurations. The findings suggest that the approach is applicable in practice and represents a means to strategically leverage the flexibility in production system design, as well as to improve the efficiency of the engineering process. Hence, further research could examine if the approach is useful in additional systems engineering domains.

Keywords: Set-based concurrent engineering, software tool, production systems, design space exploration, evaluation.

1 Introduction

Reconfigurable Manufacturing Systems (RMS) are one class of production systems that attempt to combine the advantages of two different manufacturing paradigms: economies of scale, which is the dominating focus in product-specific high-volume production, and product mix flexibility, which often is the objective for producing smaller volumes [1]. An important incentive for adopting the RMS concept is to allow for better utilization of investments into production hardware, enabled by the reuse of the system's components [2]. Nevertheless, a relevant implication that comes with the changeability of the production systems is to consider the overall system and its performance when modifying the constituent subsystems. For instance, altering the manufacturing process capabilities of a single production subsystem can affect the flow of the

products in the factory as well as the allocation of bottlenecks. Even though the focus of modifications might be on the single subsystem, the engineers should be aware of the consequences for the System-of-Systems (SoS) and capable of evaluating if the resulting global effects are synergistic, represent acceptable trade-offs or even deteriorate the overall performance. To support an automotive supplier in considering these effects holistically, a software prototype has been developed with the objective to facilitate the recurrent configuration and modification of the production system. This paper presents the evaluation results of the software prototype in the industrial case study. Therefore, section 2 highlights the problem that motivated the development of the tool, depicting particular challenges associated with designing factories as reconfigurable SoS. Also, Set-Based Concurrent Engineering (SBCE) is contrasted with other development approaches, followed by the outline of the prototype as computational support for production system design. Section 3 as the main part of this paper firstly describes the hypothetical effects of the approach with respect to the system design procedure. Thereafter, the evaluation set-up is explained, the results presented and the implications discussed.

2 Background

2.1 Industrial case of production system design

The presented research took place in the context of a longitudinal case study with an automotive supplier, in the following referred to as *the company*, and builds upon the work of three master students who contributed a basis for the presented approach with their graduation assignments. The demand of the company's customers is characterized by a high variety of products that are assembled-to-order in low, yet fluctuating volumes. Additionally, the company has limited certainty about future orders for new products of their customers. To create the capabilities to produce newly requested products in an economical way, the company adopted the concept of a cellular manufacturing system, in which production cells can be added, removed or reconfigured in response to the development of the business [2]. These production cells have a standardized structure

and consist of modular production machines and tools, enabling to reconfigure the shop floor by moving the hardware modules to different cells. Introducing new products into production may involve to add or modify cells to accommodate the processing resources additionally required. Also the capacities required for the new products' production steps have to be allocated, either to the existing or novel cells. Thus, the capability to reconfigure the production subsystems offers opportunities to react to changes in the product portfolio by re-routing also the production of other parts. In this manner, the production cells' capacities and their utilization can be customized by the engineers. However, deciding to reconfigure a larger number of subsystems increases the complexity of the involved system design and analysis activities, eventually causing more effort for the engineers. Moreover, the many opportunities for grouping the hardware modules and allocating the products give rise to a huge combinatorial space of configurations that are technically feasible and potentially applicable. Yet, the suitability of each configuration can be only determined by quantifying its performance.

Factories are an instance of complex SoS, since they can be characterized by “integral ownership, lack of overview, emerging properties and many stakeholders making decisions locally” [3]. Fig. 1 sketches out these characteristics for the case company’s factory. The red dots indicate the utilized types of operational systems involved in converting the raw materials into finished products, indicating the integral ownership of the complete production process. The Roman numerals identify the planning and operation phases that should develop solutions that are acceptable from the various system perspectives. In the company, the responsibilities for the design and operation of the system types are distributed, so that decisions concerning production, logistics and storage are made locally within the respective functional departments. In this context, the change of any aspect of the production system, such as adding additional production cells or modifying the capabilities of the existing ones, can have further reaching effects due to the changes induced in the product flow and its ramifications for the other systems. The production engineers must consider that each re-allocation of production resources might affect the routing of products using these resources. Thus, each reconfiguration of the production system may result in novel, product-specific flows through the factory that also influence the required logistics and storage capacities. Hence, the effects on the performance of the neighboring systems and their capabilities have to be taken into account to maintain a well-synchronized factory.

Therefore reconfiguring the production system allows to face the uncertain customer demand. At the same time, the magnitude of reconfiguration directly affects the complexity and duration of the system engineering process

that needs to consider the multifaceted dependencies of the subsystems among each other. This puts an emphasis on providing efficient and effective engineering processes alike, which becomes relevant when considering the vast number of possible system configurations that may be feasible, yet need to be quantified beforehand. The company’s habitual practice, which was to manually design and analyze individual production cell configurations, does not represent an efficient approach to recurrently designing systems that require to consider a higher number of interrelationships than in the past, while also offering more configuration opportunities. Hence, a computational support tool was identified as possible means to support the production engineers in effectively and efficiently assessing alternative system configurations.

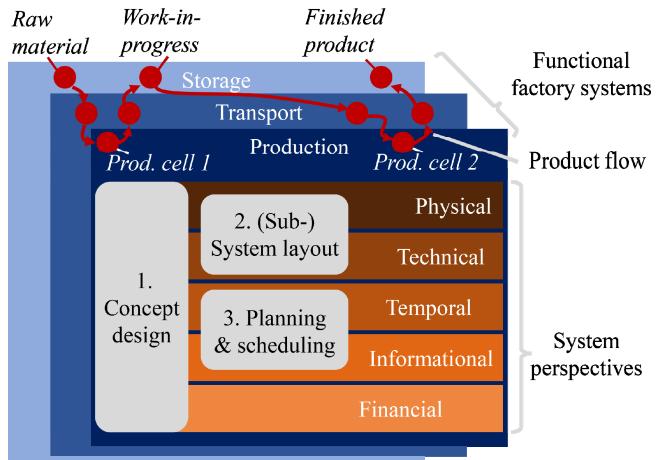


Fig. 1. High-level systems, perspectives and product flow in the factory.

2.2 Set-based concurrent engineering

SBCE describes an approach to product development, which started emerging in the 1990s. As the name suggests, the core idea of SBCE is to develop multiple solutions (a set) in parallel, considering the various functional requirements by eliminating infeasible or dominated solutions from this set in the course of the design process [4]. Hence, it can be contrasted to point-based design, where different departments integrate their functional requirements for new designs subsequently (see Fig. 2, top). Also in comparison to conventional concurrent design (see Fig. 2, middle) the result of SBCE is supposed to be a more flexible design process; rather than modifying solutions when they do not meet the specification, another solution is taken forward (Fig. 2, bottom). Researchers found that set-based development allowed to reduce the probability of iterations in the development process and thus could reduce lead times compared to point-based approaches, which promotes to apply SBCE [5]. Moreover, applications of SBCE in industrial problems have indicated high practical value, yielding an increased performance of the developed systems and an increased robustness concerning specification changes during the design process [6].

Implementing the SBCE rationale in the tool appeared beneficial to face the previously described problems because of two reasons: (i) to simultaneously compare potential configuration alternatives that all fulfill the requirements and preferences of the decision-makers; (ii) to retain alternative configurations in case the chosen candidate solution proved infeasible or disadvantageous during the following detailed design process.

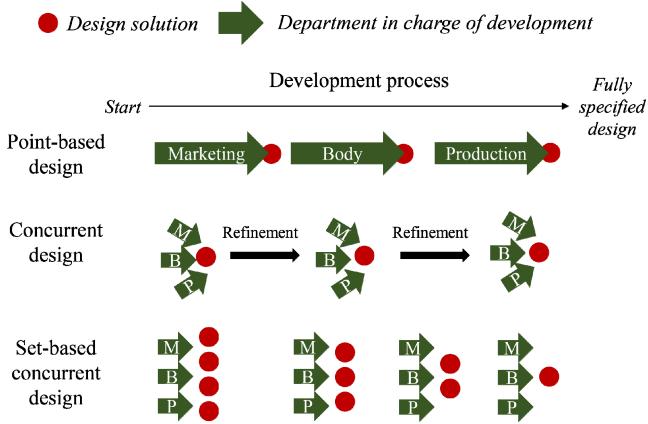


Fig. 2. Development approaches with respect to involved functions and considered solutions (partly based on [4]).

2.3 Prototype application

The developed prototype software application reflects the principle of set-based concurrent engineering by automatically generating various configurations of the production system. Its objective is to support the production engineers in the described system configuration problem by allowing to consider all subsystems in their context in the SoS. Multiple stakeholders from different departments were interviewed to model the production system and its components, as well as the relevant performances, the dependencies between the subsystems and influence factors in the system's environment. To understand the current practice at the company, the systems engineering knowledge and process was documented. Based on this, a knowledge base and multiple heuristic algorithms were developed. In combination they can automatically propose system configurations with distinct system architectures applicable to the situation of the company and analyze their performance. Each type of system architecture exhibits its specific profile of production resource and product allocation. The resulting configurations can range from production hardware distributed among multiple cells and product-specific routing to system architectures consisting of individual production lines for product families. For each of these architectures, many configurations can be generated due to the combinatorial opportunities for configuring the modular production resources to multiple cells and allocating the production steps. The detailed procedure and heuristics implemented to obtain various system configurations are described in previous work and

will be only outlined here [7]. At the beginning of the software workflow, the users specify the number of configurations to be generated and analyzed. The users can also control the design synthesis procedure to ensure that the eventual results are applicable, for instance, by prescribing the design heuristic applied by the algorithm to obtain solely specific system architectures. Additionally, the users can indicate their requirements for the performance or design of the configurations, such as maximum investment of configurations or the number of production cells. Hence, they can guide the generation of solutions by choosing specific design heuristics and filter solutions based on their requirements.

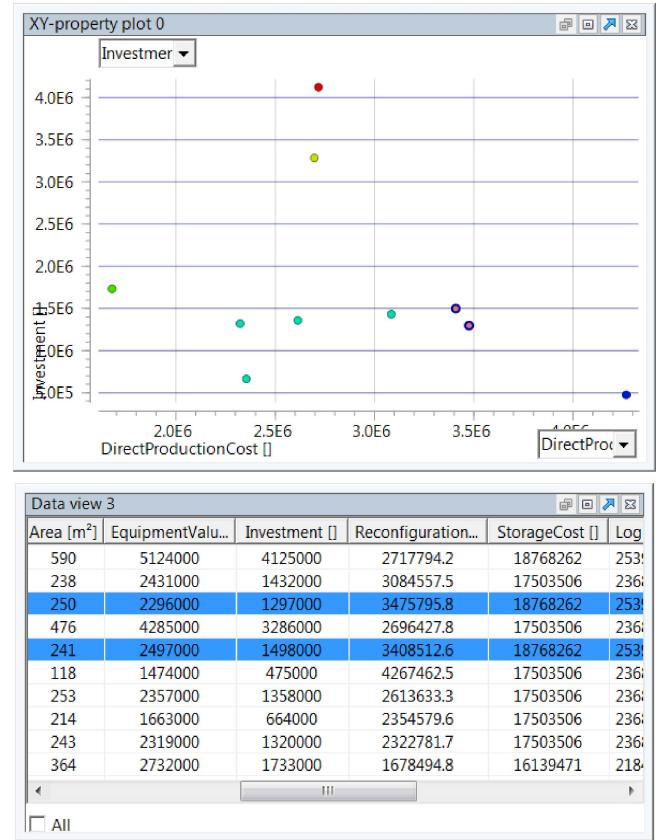


Fig. 3. Graphical interfaces for evaluation of solutions by user: x-y plot (on top); table (below).

As output of the tool, the resulting sets of solutions are visualized in graphs such as shown in Fig. 3. Each dot in the x-y-plot (top) represents one system configuration and indicates the values of the configuration's properties selected by the user for visualization, with the investment as property on the y-axis and the direct production cost on the x-axis in the presented example. In the table-view (below), each line displays the properties of one generated configuration, such as space requirement of the system or additionally required investments. Marking one or multiple solutions in either view will highlight the selected configurations in the respective other graph. By

customizing the graphs and assessing the generated set of configurations with respect to their various design and performance properties, the users are supposed to explore the generated system configurations. Moreover, the tool can highlight configurations with Pareto-efficient properties, to facilitate the simultaneous assessment of multiple properties. The intended usage of the tool is that at first only the most relevant requirements are specified by the users. After this, the users should iteratively explore the sets of solutions and – if intended – add further requirements to generate solution sets with specific and more homogenous properties in their target dimensions of performance or design.

3 Evaluation

The following sections present the hypothetical impact of the support tool and the evaluation set-up. After that, the evaluation results are presented and discussed.

3.1 Hypothetical model

The presented tool was developed to support the company in assessing various alternative configurations of their reconfigurable system. An additional objective was to provide an efficient and effective means for applying the tool on a regular basis. Hence, the tool is based on a hypothetical model that describes the individual features and their respective, expected effects with regard to efficiency and effectiveness of applying the tool. This model was used as reference for evaluating the tool and was established as part of the Design Research Methodology, which served as development framework [8]. Fig. 4 shows how (Arabic numerals) and in which phase (Roman numerals) of the system design process the factors were intended to take effect. The efficiency of the approach is expected to result from the high speed of configuring (factor 1, fig. 4) and analyzing sets of system designs in an automated fashion (2). Since solutions with design properties or performances that do not match the requirements of the user are discarded and a new candidate created for each failed attempt, also the automated retry is expected to be performed efficiently (3). The user-interfaces for evaluating and selecting configurations (4) are expected to allow an efficient approach to assessing and choosing solutions. A factor aiming to provide effectiveness is the opportunity for users to formulate requirements for design or performance properties of solutions (5). The configurations that do not match these requirements are not added to the solution set visualized to the user. Hence, this factor should allow to focus the generation of solutions onto – or away from – solutions with specific properties. For instance, the maximum available shop floor area or maximum allowed investment for configurations can be prescribed. If multiple configurations satisfy these requirements, the users are supposed to compare them with respect to their performance and design properties (6),

using the interfaces. Moreover, the performance sensitivity of each system configuration to user-defined product demand scenarios should be assessable (7). Lastly, the approach was designed to support the users in finding advantageous configurations by visually highlighting solutions based on their design or performance property values, e.g. using color scales (cf. fig.3) (8). In this manner, the graphical interfaces are supposed to assist in comparing configurations and selecting the preferred ones.

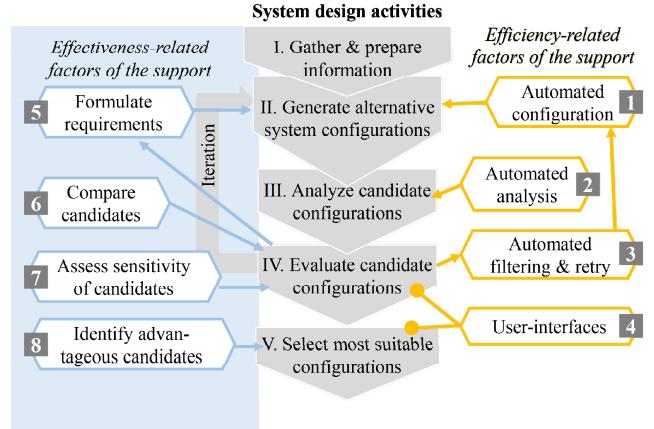


Fig. 4. Design activities and the related factors of the support tool.

3.2 Evaluation approach

The evaluation of the approach was divided into two phases, case testing and validation.

I. Case testing: The participants in the first phase of evaluation were the two lead production engineers of the company, one responsible for designing the system configurations and the other for implementing the selected concepts. At the beginning of the session, they were introduced to the tool by explaining the features and interfaces. After that, they solved two cases using the tool. The specifics of these cases are presented in Fig. 5 and were supposed to give a realistic context for applying the tool. For each problem case, they were asked to suggest a minimum of three alternative system configurations suitable for producing the case-specific input set of products. The cases differed in the number of products to be produced; additionally, the system configurations had to match case-specific performance and embodiment requirements. The two cases were used to evaluate if the support tool can be used with various problem statements, differing in the sizes of the problem and the design space. The requirements were supposed to embody the dependencies with neighboring systems, as well as potential production strategy objectives. Following each case, semi-structured interviews were conducted to evaluate the influence of the features implemented in the tool (Fig.4) and to document the subjective opinions of the participants. For instance, one question asked was whether the ability to formulate requirements had an influence on the quality of the

suggested solutions and, eventually, the effectiveness of the approach.

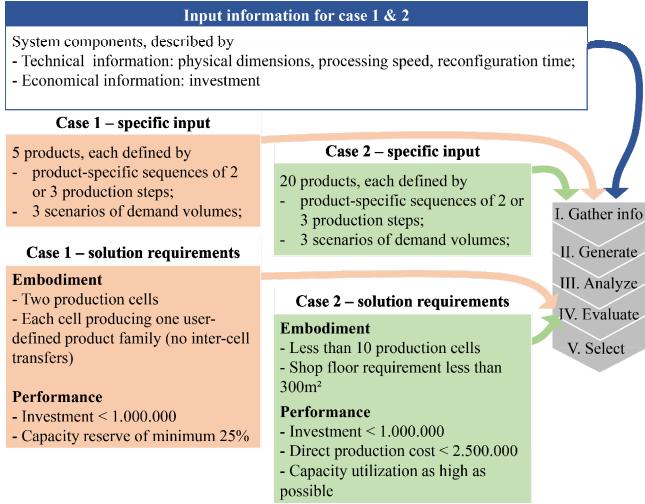


Fig. 5. Input information and solution requirements for evaluation cases.

II. Validation: The second phase of the evaluation aimed at validating and substantiating the findings of the first phase. Participants were the engineers of the first phase and a senior manager responsible for the entire process chain of the production department. In this phase, the results of the first session were discussed to verify the hypothetical model and validate the conclusions. Also, unclarified aspects and ambiguous answers were addressed, and, the realism of the applied case scenarios was evaluated.

3.3 Evaluation results

The evaluation cases proceeded as intended. First, the engineers examined the input information to get familiar with the problem. Then, they generated the first set of solutions and screened it for candidates that fulfilled the case-specific embodiment and performance requirements. Since the set did not contain such configurations, they iteratively entered the requirements for solutions and generated and evaluated further sets of solutions. Eventually, they managed to select three applicable solutions for both cases. During the following interviews, they confirmed the effect of the individual factors that should provide for an efficient support of the design process. Also, the individual, effectiveness-related factors of using the tool to explore and select sets of system configurations were confirmed. Besides establishing the factors individually, the answers also verified the combined effects of the factors on the engineering process. The engineers stated that the complementary functions were well-suited to enable an efficient and effective approach to design space exploration. While the solution requirements for resulting system designs were approved as realistic for both cases, the input information was described realistic only for the first case (Fig.5). The experts expressed that

the twenty products used as input for the second case scenario represented a system design problem much larger – and thus more complex – than considered in traditional practice. In summary, the participants could recognize the purpose of the features to support the design strategy. It was acknowledged that applying the approach allowed to consider the system configurations more holistically at an early stage of the design process, allowing to evaluate more options than previously and to include the requirements and preferences from the various perspectives of the functional departments. Lastly, the opportunity to assess the influence of multiple demand scenarios onto the performance of solutions gave insight into sensitivity and would represent a more extensive evaluation than currently practiced.

3.4 Findings & discussion

While the results of the evaluation can be described as mainly positive, an unexpected finding was that the size of the second evaluation case was perceived unrealistically large by the engineers. This appraisal is not shared by the authors. A possible explanation for their opinion was identified in the lacking awareness of the implicit SoS engineering, as reported in other cases [8]. The company had little relevant experience with the new shop-floor strategy and the implicit opportunities for system configuration. The only reference for problem sizes was the traditional engineering process that only considered two options: adding additional cells for new products; or allocating additional products to existing cells without changing any hardware. Thus, the many products of the existing portfolio and hardware of almost 100 existing production cells were not considered again for the design of additional subsystems, leading to the statement. This argumentation was confirmed by the engineers during the validation. They acknowledged that for obtaining an overall well-performing system configuration, it may be beneficial to consider all subsystems of the factory possibly qualified for reconfiguration, probably affecting a higher number of products. By considering the factory more holistically and as a SoS, larger synergies could be detected and exploited and thus lead to additional benefits in terms of system performance or design opportunities.

As described before, the engineers followed their established procedure that limited the focus of attention to either individual products or production cells. The main reason for this were the increasing complexity of considering larger problems, for instance when designing multiple, related production cells for multiple products at the same time. It could be valuable to increase the awareness of the engineers and management of the consequences of their standard engineering routines with regard to its influence on the entire production system. This could help to build an understanding of the criticality of the practiced engineering and management approach, its limitations and benefits. Another discovery made was that

the implicit, novel approach to developing system configurations requires decision-makers to apply specific strategies to narrow down the candidates. Due to the large number of potential solutions and the quasi-random design instantiation through the algorithm, the experts found it difficult to interpret if there were more solutions than presented in the generated set, which satisfy the requirements and potentially exhibit interesting properties. We recommend to face this issue by discussing and determining the relevant properties of the overall system and their relative priority with the stakeholders of the various subsystems, and formulating corresponding search strategies.

One limitation of this research was that the model used during the evaluation of the tool did not contain the full level of detail required to let the engineers conclusively evaluate the technical feasibility of proposed solutions. Even though the authors do not perceive the adding of further design details as a delicate task, the applicability of the approach in the real setting yet has to be tested.

Despite these limitations, one aspect worth considering is that the rationale of the approach could make it suited also to support in further (production) system engineering problems. The main principle is to instantiate the parametric design model of the system by an algorithm to yield various, possible system configurations that are visualized thereafter. Transferring this principle to additional parametric models of engineered systems appears an interesting endeavor. In this context, attention must be paid to the expected costs and benefits of making such approach operational, while the specific merits of adopting the approach may be difficult to quantify beforehand. Lastly, it has to be considered that despite our best efforts, the applied qualitative research methods limit the generalizability of our findings.

4 Conclusions

The paper presented the evaluation results of a software tool for multi-disciplinary system design that was developed to support a company from the manufacturing industry in the early phase of production system configuration. We described the objectives of the evaluation and the underlying hypothetical factors, aiming at an effective and efficient support of the process. In both evaluation cases, the tool was found to be suitable to consider more candidate system configurations than previously practiced. It allowed to examine the implications of numerous production system configurations for the overall system, logistics and storage early in the decision process. The intended effects of an efficient and effective decision-support could be reached, yet more research is required to validate the tool's effects in context of the practical systems engineering process. Thus, the results obtained from the evaluation sessions encourage further

work scoping the application in other system configuration problems that are characterized by multi-disciplinary engineering and large, various design spaces.

Acknowledgment

The authors acknowledge the contributions of the master students and partial funding by the European Union (grant no. 609087). The first author would like to thank Alberto Martinetti for his helpful comments.

References

- [1] Y. Koren, et al., "Reconfigurable manufacturing systems", *Annals of the CIRP*, vol. 48, no. 2, pp. 527-540, 1999.
- [2] M.G. Mehrabi, A.G. Ulsoy, and Y.Koren, „Reconfigurable manufacturing systems: Key to future manufacturing”, *Journal of Intelligent Manufacturing*, vol. 11, pp. 403-419, 2000.
- [3] G. Muller, and J. H. Andersen, "Factory production line as SoS; a case study in airplane engine component manufacturing", 10th System of Systems Engineering Conference 2015, pp. 42-46, May 2015.
- [4] D. K. Sobek II, A. C. Ward, and J. K. Liker, "Toyota's principles of set-based concurrent engineering", *Sloan Management Review*, vol. 40, no. 2, pp. 67-83, Winter 1999.
- [5] D. Shahani, and C. C. Seepersad, "Implications of Alternative Multilevel Design Methods for Design Process Management", *Concurrent Engineering: Research and Applications*, vol. 18, no.1, pp. 5-18, 2010.
- [6] D. Raudberget, "Practical Applications of Set-Based Concurrent Engineering in Industry", *Journal of Mechanical Engineering*, vol. 56, no.11, pp. 685-695, 2010.
- [7] J. Unglert, J. Jauregui-Becker, and S. Hoekstra, "Computational Design Synthesis of reconfigurable cellular manufacturing systems: a design engineering model", *Procedia CIRP*, vol. 57, pp. 374-379, May 2016.
- [8] L.T.M. Blessing, A. Chakrabarti, *DRM, a Design Research Methodology*. London: Springer-Verlag, 2009.
- [9] G. Muller, "Are stakeholders in the constituent systems SoS aware? Reflecting on the current status in multiple domains", 11th System of Systems Engineering Conference 2016, pp. 1-5, June 2016.