



**HAL**  
open science

# Application of a Standardized Design Procedure in the Development of Automated Micro-assembly Processes

Gordon Saunders, Tobias Müller

► **To cite this version:**

Gordon Saunders, Tobias Müller. Application of a Standardized Design Procedure in the Development of Automated Micro-assembly Processes. 9th International Precision Assembly Seminar (IPAS), Dec 2020, Held virtually, Unknown Region. pp.25-35, 10.1007/978-3-030-72632-4\_2 . hal-03520402

**HAL Id: hal-03520402**

**<https://inria.hal.science/hal-03520402v1>**

Submitted on 11 Jan 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# **Application of a Standardized Design Procedure in the Development of Automated Micro-Assembly Processes**

Saunders, Gordon; Müller, Tobias. Fraunhofer Institut für Produktionstechnologie.

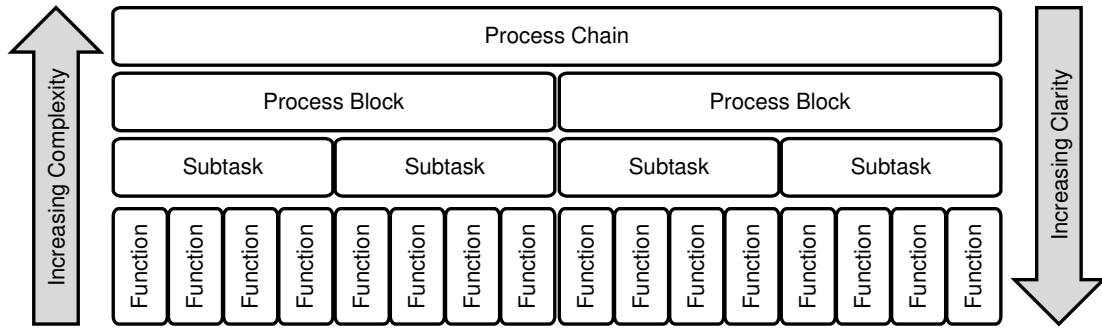
## **1 Introduction**

Automated precision assembly of e.g. optical systems requires development efforts in manifold domains, such as part feeding, handling, alignment, bonding and quality control. The use of systematic design procedure enables the rapid and complete development of new applications and use cases using existing equipment. Combined with modular equipment and subsystems, the use of a standardized design process significantly reduces development time and therefore costs. A generic methodology based on the functional decomposition of assembly task requirements and a coherent synthesis of functional process building blocks can be an answer to reduced process ramp-up time, more stable processes and enable concurrent engineering for novel tool and process development.

The objective of this paper is to present a systematic design procedure for the development of automated production processes to the point of Technology Readiness Level 5, or TRL 5, as defined by the European Union's Horizon 2020 framework [1]. This development process includes the analysis of the assembly task, the identification and selection of promising process concepts, and the testing of said process concept feasibilities within the assembly environment. As a result, individual component processes solving the most substantial automation-specific issues are tested in a production-relevant environment. These results prove the practical validity of the assembly process on the component-level, thereby fulfilling the TRL 5 criteria [2]. Ultimately, the results of these component validation tests are combined into a cohesive system, laying the groundwork for the testing and validation of the entire process chain and paving the way to meeting TRL 6 criteria.

## **2 A Function-Based Design Procedure**

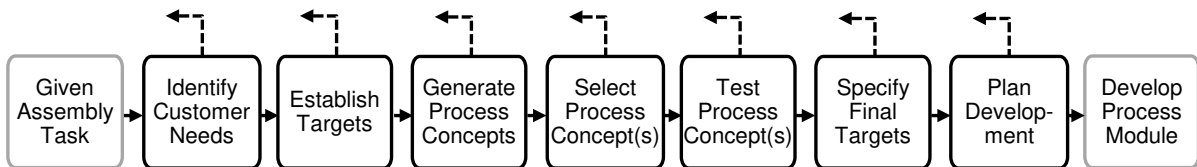
An assembly task must be separated into its constituent parts to allow for a dynamic and complete concept generation phase. Doing so allows for the restructuring of a once homogenous and complicated problem into many individual and clearly-defined subroutines. This clarity in purpose enables a more complete examination and problem-solving process for each of the individual functions inherent within the assembly task without inadvertently omitting functional considerations by working on too complicated of a design problem. In this respect, the design procedure presenting in this paper aims to decompose a typical micro-assembly process chain into its constituent process blocks, subtasks, and functions. These functions support their associated process subtasks which in turn make up their associated process blocks. Hence, these components can be viewed as hierarchical in nature and are depicted as such in Figure 2.1. [3] [4]



**Figure 2.1: Functional Decomposition Hierarchy**

This work follows an adapted procedural design strategy originally laid forth by Ulrich and Eppinger (2013) [3]. Given a specific assembly task, this adapted strategy follows a systematic and iterative procedure to develop a process design concept. That is, if the development process stalls at any given stage, work can be resumed at any previous stage using the same previously developed materials and preventing a complete restart.

The core principle of this design strategy revolves around the identification of customer needs and the hierarchical functional decomposition of the assembly task to generate process concepts. Using these techniques allows for an adaptable problem-solving procedure while ensuring a complete solution via systematic function matching [4]. Each step of the overarching procedural design strategy is summarized in Figure 2.2 and further clarified in the following descriptive text.



**Figure 2.2: Design Procedure for Use in Process Development**

*Identify Customer Needs* - The identification of explicit and latent customer- and task-requirements through internal and external interviews, background research, as well as an examination of existing process concepts and the given assembly task.

*Establish Targets* - The transformation of previously identified customer and task requirements into appropriate functional requirements that a successful assembly process must satisfy.

*Generate Process Concept(s)* - The generation of process concept solutions that have the potential to satisfy the functional requirements of the assembly task.

*Select Process Concept(s)* - The refining of the pool of potential solutions by comparison and analysis of the suitability of each solution with regards to the previously identified functional and customer requirements.

*Test Process Concept(s)* - The feasibility testing of the subset of plausible process solutions.

*Specify Final Targets* - The determination of specifications for a complete process module and process chain based on the results of process concept testing and functional requirements.

*Plan Downstream Development* - The planning of a construction and testing process for a prototype of the combined process module.

### **3 The Design Procedure in Practice**

Ultimately, this procedure introduced in the previous section progresses from identifying basic needs of a given assembly task to providing a technically feasible proof of concept for the continued development of an automated micro-assembly process module. This section dissects and further explores each of these individual design steps in further detail while discussing the tools and practical applications of each step as it relates to a theoretical process design challenge.

#### **3.1 Identify Customer Needs**

In order to fully understand the needs of the customer and therefore the requirements of an automated micro-assembly process, the given assembly task must first be understood within its context. In other words, knowledge of the status quo of both existing assembly processes, as well as knowledge of automated micro-assembly systems in general, facilitates a thorough analysis of the options present in micro-assembly automation. This is an exploratory role undertaken by the designer as the customer alone cannot provide or enumerate a complete and trustworthy list of his or her own needs. Rather, the customer is only aware of his or her own explicit requirements and may be unaware of their interpretation within the context of automated micro-assembly process. Research of the status quo helps to reveal this hidden, or implicit, process requirements and, combined with knowledge of automated micro-assembly processes, provides a basis through which future process concepts are generated as a component of the overarching procedural design strategy in Figure 2.2.

A review of the present status quo involves many approach vectors which can be simplified into two main categories: those pertaining to the given assembly task and those pertaining to automated micro-assembly in general. First, the analysis of the given assembly task can involve, among others, the inspection and documentation of the part to be assembled, the existing manual or semi-automatic assembly processes, the production processes of the component parts to be assembled, or the eventual use case of the assembly part. Such analysis should take care to document limitations and bottlenecks of existing process and interfaces. In particular, it must be noted that an automated assembly process only serves to replace the existing assembly processes, leaving the production routines and customer use cases unaffected. Therefore, special attention should be given to first, how initial component error can be mitigated through module design, and second, that the part quality is preserved throughout use.

The second main approach to need identification involves needs based on automated micro-assembly processes in general. In this respect, designers must be well-educated with regards to process and equipment varieties present in the field of automated micro-assembly. This familiarity provides multiple benefits: not only does it assist in the generation of concepts due to a breadth of options, but also allows the designer to interpret customer needs specific to

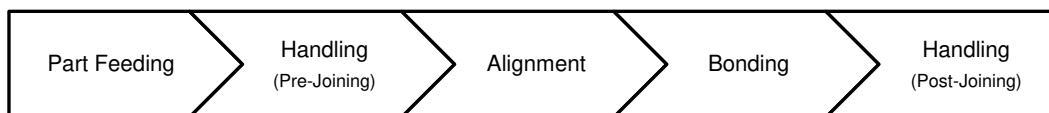
automated micro-assembly and add additional unspoken needs to support the new demands of automation.

This research of the status quo can be accomplished via multiple means. Both internal and external interviews as well as site visits, academic background research, market research, and self-examination and experimentation with the existing process all assist the designer in the identification of customer needs. Additionally, limitations of the development or production environment can be considered as needs. Examples of these limitation-based needs include cost-limits, time-limits, and the reality of machine availability.

### 3.2 Establish Target Specifications

A typical micro-assembly process chain can consist of five primary process blocks: part feeding, handling, alignment, bonding and handling again. Within the part feeding process block, parts to be assembled are first transported and delivered to the working space. Following this, the handling process involves the gripping and movement of parts inside the working space. Handling can be separated into two primary blocks to account for differences in functional demands—one pertaining to the the handling of multiple components prior to joining and a second block pertaining to the handling of the joined part.

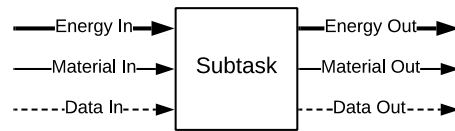
In contrast to the handling process, alignment focuses on the precise measurement and fine local manipulation of parts to match its desired configuration. The bonding process, performed in conjunction with the alignment process, involves the mechanics used to join parts into a final assembly. Collectively, these five process blocks are pictured in Figure 3.1. [5]



**Figure 3.1: Typical Micro-Assembly Process Chain**

These process blocks allow for a simplified view of the process chain but are divided on an inherently subjective basis. These overarching process blocks are too complex to allow for a thorough discussion of the inherent functional requirements present in each. For this reason, the process blocks are further reduced into their subordinate tasks, offering a more nuanced lens through which to view the process chain.

A useful exercise to refine the process chain and therefore define subtasks is to create a process map. In such a map, blocks representing subtasks are characterized by the inputs, outputs, unique functional requirements of each and are connected by the flow of inputs and outputs to and from blocks. Subtasks are intentionally represented as generalized black boxes in order to avoid assuming process concepts, and are connected by energy, material, and data flows with the use of arrows leading into and out of various subtasks such as in Figure 3.2. Through this exercise, the intricacies of the process dependencies are extracted. Furthermore, as this process map is a decomposition of the standard micro-assembly process chain, it serves as a generalized foundation for micro-assembly solutions that can be further adapted to match the reality any given assembly task.



**Figure 3.2: Subtask Block with Charted Inputs and Outputs for Use in Process Maps**

The benefits of using subtask identification as an intermediary tool are not immediately shown but rather reflected later in the design chain. By refining and redefining the process chain using subtasks, functional requirements unique to a specific subtask are identified that may have otherwise been overlooked. These constituent functions serve to continually narrow the scope of the each process step and form the basic building block of the typical micro-assembly process chain. Functions at this level are single, task-specific, and solution-independent transformations of a given material, energy, or data input into a prescribed output. These functions are required actions that a subtask or process must fulfill and take the form of “the process module must \_\_\_\_.” Examples of functions specific to the present assembly task include “the process module must *grip the part*” or “the process module must *calculate the required movement*.”

The initial phase of function identification comes from three sources: the customer, the assembly task, and the decomposition of the subtasks found in the generalized process map. Functions are created from customer needs through a transformation in perspective. In this vein, a customer need that requires “batch sizes of at least 25 parts” is represented in two functional requirements, that “the process module must *accept a batch size of at least 25 of each part*” and “the process module must *store at least 25 assembled tools*”. Meanwhile, task-based functions stem from the technical requirements of the assembly task as well as the procedural and safety requirements of the working environment. These first two sources account for the customization of the development process specific to a given assembly task. Meanwhile, the third source, the standard functional decomposition of typical micro-assembly subtasks, ensures that the customized requirements are implementable within an automated assembly process. In later stages of development, specific solution configurations may require an additional fourth source of functions and reconsideration of the identified functional components.

Following the work of Galvao and Sato (2005), this body of functional parts can be matched with their associated process subtasks and vice versa [6]. Should there be a perfect 1:1 matching of functions and subtasks, the marked intersections on the matrix would lay perfectly on the diagonal. However, in most systems that is not the case. Portraying the function-subtask relationship in this way enables the analysis of subtask and function interdependencies along the process chain. These interdependencies result from multiple subtasks sharing an individual function or from multiple functions sharing an individual subtask and are represented on the matrix by rows or columns with more than one point of intersection. This results in vertical or horizontal sections that cause clumps or deviations from a perfect diagonal line, such as the identified regions in Figure 3.3. These regions are of particular interest in the concept generation phase, as these areas of interdependency identify complex problem areas and require solutions that satisfy multiple requirements simultaneously. Shared functions and

subtasks are accounted for in the design of process concepts by ensuring that the shared function is satisfied in all associated subtasks and that subtask concepts balance the needs of multiple functions.

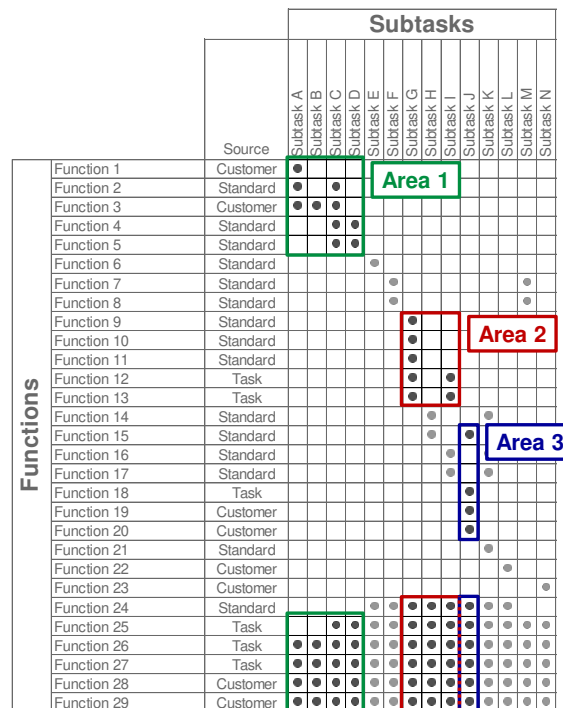


Figure 3.3: Areas of Functional Interdependence for Further Investigation

Therefore, a function-subtask matrix is the ultimate result of the functional decomposition of the assembly task and has a two-fold purpose in the design procedure. First, it is a listing of all identified functional requirements which serve as a basis for generating and evaluating process concept solutions for their feasibility. Secondly, the matrix shows process interdependencies and identifies critical problem areas. As these areas of interdependent functions often form the most complex problems, they are the first problems to be accessed for feasibility in the development of a new process module.

### 3.3 Generate Process Concepts

The concept generation step of the design procedure is focused wholly around solving the identified functional requirements. These functions provide both a springboard from which concept generation and discussion can begin as well as a metric by which the concepts can be later evaluated. However, to maximize designer creativity and therefore novelty of the solution reached, the concept generation phase is a purely theoretical phase wherein no idea is necessary false. Suitability determination and the development of in-depth testing procedures are omitted until the later selection and testing phases respectively.

Thus, for each identified critical problem area, existing solutions are identified and new concepts are developed. This process can take place using a variety of different approaches including but not limited to individual or group brainstorming, interviews with both internal and

external experts, a search of published literature and patents, benchmarking of existing processes, or systematic exploration of technical solutions [3].

### 3.4 Select Process Concepts

In order to reduce the generated solutions to a testable process concept, the various solutions are screened via a qualitative selection matrix. The selection matrix is a novel approach to decision making in an engineering design environment as it provides a quantitative evaluation method to interpret otherwise qualitative data. Furthermore, a selection matrix aims to select the optimal solution variant out of a given set by analyzing each variant's ability to fulfill the functional requirements specific to each process block [3]. Each solution is rated based on its satisfaction of the functional requirements relative to the other variants. Rating each functional requirement on a scale and summing the grades of each solution allows for the scoring of the each generated concept and the determination of the most suitable solution such as in Figure 3.4.

	Possible Solutions						
	Concept A	Concept B	Concept C	Concept D	Concept E	Concept F	Concept G
Function 1	◆	◇	◆	◆	◇	◆	◆
Function 2	◆	◇	◆	◇	◇	◆	◆
Function 3	◇	◇	◆	◇	◆	◆	◆
Function 4	◆	◆	◆	◆	◆	◆	◆
Function 5	◆	◆	◆	◆	◆	◆	◆
Function 6	◆	◆	◆	◆	◆	◆	◆
Function 7	◆	◆	◆	◆	◆	◇	◆
Testability	◆	◆	◇	◆	◆	◇	◇
Dependency Compatability	◆	◆	◆	◆	◆	◆	◆
Expert Opinion	◆	◆	◇	◇	◇	◇	◇
<b>Totals</b>	<b>7.5</b>	<b>6</b>	<b>6</b>	<b>5</b>	<b>6</b>	<b>5</b>	<b>5.5</b>

Key	
0	◇
0.5	◆
1	◆

**Figure 3.4: Example of a Standard Selection Matrix with Discrete Grading**

In addition to strictly functional requirements, the selection matrix can include additional relevant metrics. In this way, the additional concepts of testability, compatibility with dependencies, and the consensus opinion of the experiment interviews are considered in the solution narrowing procedure [4]. Doing so expands on the functional adherence of the chosen process concept by ensuring that the process can be tested with the available time and equipment, does not inhibit or hinder dependencies elsewhere in the process chain, and is considered a reasonable and plausible option by those with experience in the field.



	~ Weights ~	Concept A	Concept B	Concept C	Concept D	Concept E	Concept F	Concept G	
Function 1	x2.0	0.7	0.2	0.4	0.4	0	1	0.5	continuous/ quantitative grading
Function 2	x2.0	0.6	0	0.5	0.1	0.1	0.5	0.4	
Function 3	x2.0	0.2	0.2	0.6	0.3	0.9	0.4	1	
Function 4	x1.0	0.8	0.7	0.8	0.9	1	0.4	1	
Function 5	x1.0	0.5	1	1	1	1	1	1	
Function 6	x1.0	1	0.5	0.5	0.5	0.5	0.5	0.5	discrete/ qualitative grading
Function 7	x1.0	1	1	1	1	1	0	0.5	
Testability	x2.0	1	1	0	0.5	1	0	0	
Dependency Compatability	x1.0	1	1	1	0.5	0.5	1	0.5	
Expert Opinion	x1.0	1	0.5	0	0	0	0	0	
<b>Totals</b>		<b>10.3</b>	<b>7.5</b>	<b>7.3</b>	<b>6.5</b>	<b>8</b>	<b>6.7</b>	<b>7.3</b>	

**Figure 3.5: Example of a Selection Matrix Variation with Varied Grading and Weighting**

A major benefit in using a selection matrix is its customizability to the demands of any given assembly task or development environment. This customization is achieved through three primary means. First, the variation of evaluation parameters such as the previous examples of accounting for testability and dependencies allows for the consideration of any metric. Second, the grading scales can be adjusted based on each metric. In the example shown, a binary, discrete scale of 0, 0.5 and 1 was used to quantify subjective judgement of functional suitability. However, alternative grading systems such as ranking, or the transformation of quantitative data into a continuous grade scale can also be used. Lastly, functional requirements can be weighted with a blanket multiplier to denote those functional requirements of special importance such as in Figure 3.5. Through these variations, selection matrices expand upon their ability to aid the designer but the parameters must be thoughtfully chosen as they affect the final selection.

### 3.5 Test Process Concepts

Following concept selection, functional requirements continue to be used as a basis and a metric by which process concepts are judged. Evaluating how test results adhere to the original functional requirements ensures that a final process module using those concepts will fulfill the functions required by the customer and the given assembly task. Thus, the previously selected concept variants that were determined to be most suitable for their associated functional requirements are implemented in prototypical process module subsystems corresponding to each problem area. In order to fulfill TRL 5 criteria, these subsystem prototypes are implemented and tested in production-relevant conditions.

The evaluation methodology used to determine concept suitability varies depending on the specific functional requirements of the subtask or subtasks the prototypical subsystem aims to fulfill. Thus, each problem area merits its own custom experimental methodology. However, in the design of these experiments, the determination of exact process uncertainties or production parameters are not the goal of these experiments. Rather, the testing phase seeks to determine only that the previously chosen concepts are feasible in a production-ready environment and that its associated functional requirements will be fulfilled. Thus, feasible

results of these tests denote a production-ready proof of concept for the associated subsystem and enables the further development, integration, and refinement of the tested subsystem.

### **3.6 Specify Final Targets**

In the specification of final targets the findings of each separate testing process are combined into a single, cohesive list of specifications a combined process module must fulfill. This phase is a purely analytical phase of the development process wherein requirements and dependencies inherent in the combined process module are identified by examining the performed tests and revisiting the decomposition and definition of functional requirements and subtasks.

In these testing phase, certain equipment and processes are determined as being necessary for final process feasibility. As these demands are found separately, they must be combined along with their dependent inputs and outputs and be recorded in a systematic way to enable the expedient development of combined process module incorporating the required elements of all preliminary tests. In addition, these equipment and process demands can impart additional functional requirements necessary for their functioning or impart changes to the subtask map relating to their place in the process chain. Thus, the decomposition and definition of subtasks and functions should be revisited and reevaluated at this stage in the development process to ensure that these documents and metrics remain relevant to the evolving design.

### **3.7 Plan Development**

The continued development of an assembly process module is facilitated by the development of a combined process module concept based on the consolidated demands and requirements specified in the previous development phase. This combined process module represents one cohesive concept for the complete automated assembly process, including straightforward or standard solutions for the subtasks and functional requirements falling outside of the tested problem areas.

In the development of this combined process module, the competing demands of processes and equipment must be weighted and judged, in order to tactfully design a module without impeding previously solved functions. The determination of consistent reference frames, the minimization of travel times, and consideration of tool envelopes can all be deciding features in the combination of module subsystems with regards to physical location. Similarly, the rudimentary order of subtask functioning must also be determined. Thus, a combined process module and process chain concept is produced as a result of this development step.

## **4 Conclusion**

This paper introduces a systematic design framework that enables the flexible but thorough development of automated assembly processes based on identified customer- and task-specific needs and their associated functions. The first step of this design procedure involves the identification of the base functions of the assembly task through the decomposition of a generic automated assembly process chain. These functions are integrated with the customer- and task-specific needs, forming a complete model of the requirements of a final process

module. From these functions, problem areas are identified for testing by focusing on complex regions of interdependencies found within the process requirements. Process concepts corresponding to each of these problem areas are developed via a three-step procedure. First, solution concepts are generated through the exploration of existing techniques and technologies. Second, these solutions are screened by a qualitative evaluation of each concept's fulfillment of the requirements. Finally, the remaining concepts are systematically tested and evaluated for each's feasibility within its process block. These process concepts are merged into a combined process module incorporating the required elements of all preliminary tests. This process module represents one cohesive concept for the complete automated assembly process. Collectively, this investigation and practical validation of process components in a production-relevant environment determines a feasible automated assembly process concept and satisfies the requirements of Technology Readiness Level 5.

This module is not necessarily a production-ready design but rather a model for the future development of a final process concept. In order to expand this process to Technology Readiness Level 6 and beyond, the continued development of the module and assembly process involves the construction, testing, evaluation, and optimization of the module concept based on its fulfillment of the functional requirements of the task. Doing so validates the developed process on a system-wide level as opposed to the compartmentalized component level tests done earlier in the development procedure and enables the realization of the assembly process in a production environment.

## 5 References

- [1] European Commission: G. Technology readiness levels (TRL), Horizon 2020—Work Programme 2014-2015. Extract from Part 19 - Commission Decision C(2014)4995. 2014.
- [2] European Space Agency: Technology Readiness Levels Handbook for Space Applications, First Edition. 09.2008.
- [3] Ulrich, K.; Eppinger, S.: Product Design and Development, Fifth Edition. New York, NY: McGraw-Hill Irwin, 2013, p. 16, 118-164.
- [4] Pahl, G.; Beitz, W.; Feldhusen, J.; Grote, K.: Engineering Design: A Systematic Approach, Third Edition. London: Springer Berlin-Heidelberg, 2007, p. 9-10, 53, 77-103, 107, 146-153, 169-178.
- [5] Bergs, T.; Sauer, S.; Hoeren, M.: Lecture 13, Automated Precision Assembly of Optical Systems. Lecture Notes, High Precision Glass Optics Manufacturing, RWTH Aachen University, Winter Semester 2018/19.
- [6] Galvao, A.; Sato, K.: Affordances in Product Architecture: Linking Technical Functions and Users' Tasks. In: Proceedings of IDETC/CIE 2005, ASME 2005 International Design Engineering Technical Conference and Computers and Information in Engineering Conference. Long Beach, CA, 28.09.2005.