

Article

Knowledge-Based Automated Mechanical Design of a Robot Manipulator

Robert Pastor , Milan Mihola , Zdeněk Zeman and Adam Boleslavský 

Department of Robotics, Faculty of Mechanical Engineering, VSB-TU Ostrava, 70800 Ostrava, Czech Republic; milan.mihola@vsb.cz (M.M.); zdenek.zeman@vsb.cz (Z.Z.); adam.boleslavsky@vsb.cz (A.B.)

* Correspondence: robert.pastor@vsb.cz; Tel.: +420-737-570-794

Abstract: Design methods have been improving with an increasing level of algorithmic support for some time. The most recent advances include generative design and various optimization methods. However, the automated design tools are often focused on a single stage of the design process, for example, kinematics design, mechanical topology, or drive selection. In this paper, we show the whole design process of a robotic manipulator in an automated workflow. The method consisted of two main parts: a genetic optimization of the kinematic structure and an iterative automated CAD design. The method was then applied to a case study in which a manipulator with five degrees of freedom for a handling task was designed.

Keywords: kinematics optimization; modular manipulator; genetic optimization; knowledge-based system; automated CAD design



Citation: Pastor, R.; Mihola, M.; Zeman, Z.; Boleslavský, A. Knowledge-Based Automated Mechanical Design of a Robot Manipulator. *Appl. Sci.* **2022**, *12*, 5897. <https://doi.org/10.3390/app12125897>

Academic Editors: Ahmad Barari and Marcos de Sales Guerra Tsuzuki

Received: 8 April 2022

Accepted: 7 June 2022

Published: 9 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

When robots in manufacturing are concerned, we often think of their universality as a good thing. However, a case can be made for the utility of a robot which is tailored for a specific task, in the sense of a single-purpose machine. Such a robot would have the kinematic structure optimized for the task at hand and hypothetically should be more energy efficient than a universal robot, which must carry its drive units even if the task does not require their use.

This single-purpose robot would need to be designed on a task-specific basis. Nevertheless, optimizing the kinematics, designing, manufacturing, and assembling a whole new robot is a lengthy process that requires a lot of expertise and time. The process of creating a new robot for a specific task would require considerable resources, and the profitability of such a system is uncertain, even if it promises lower operating costs. The return of investment becomes interesting when we start lowering the cost of designing the task-specific robot. To reduce the cost of the design process cost, we need a robust methodology with as much automation as possible. Ideally, instead of having a team of expert engineers designing the robot, we would have a computer program that one person could use to explore the design possibilities.

Parametric CAD software packages can alleviate many of the routine design processes in general engineering tasks, e.g., bolt placement. For more domain-specific design tasks, there is a method usually referred to as Knowledge-Based Engineering (KBE). There are many definitions of KBE systems [1]. An elaborate description is given by Chapman and Pinfold [2] in which they state that 'KBE represents an evolutionary step in Computer-Aided Engineering (CAE) and is an engineering method that represents a merging of Object-Oriented Programming (OOP), artificial intelligence (AI) and Computer-Aided Design (CAD) technologies, giving benefit to customized or variant design automation solutions.

Many KBE methods are built on a set of design rules or a decision tree [3]. However, some systems do incorporate optimizations of the design. Xu et al. [4] have implemented a

multi-objective pareto optimization algorithm to deal with diverse design objectives and constraints. A knowledge-based system for containership cargo tank structural design was presented by Cui and Wang [5]. Their approach employs Multi-island Genetic Algorithms as a last step in the design process. La Rocca et al. [6] used structural analysis and multi-level optimization of aircraft wings in their generative model for aircraft wings. Optimizations in KBE can take many forms, whether it is an optimization of mechanism topology and structure [5,7] or a topological optimization of individual parts or multiple parts at once [8].

In Figure 1, a generic structure of a KBE system is indicated, similar structures can be found in [1,9–11]. An interesting feature of optimizations in KBE systems is the placement of the optimization step. Some researchers have placed the optimization function block towards the beginning of the design process. Some consider optimization to be one of the last steps in the process [5]. This difference is most likely due to the design rules that are being applied in the process and the nature of the optimization.

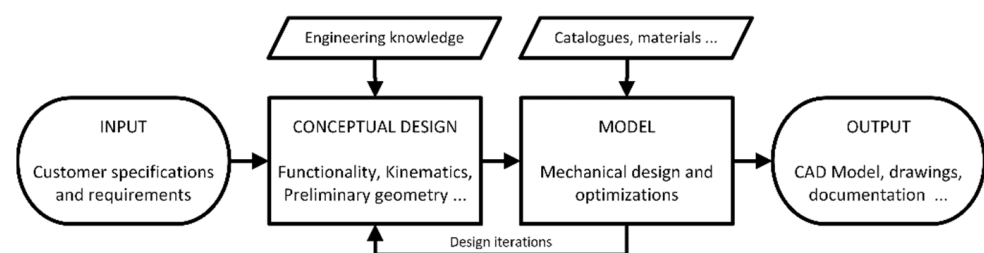


Figure 1. Generic structure of a Knowledge-Based Engineering System.

Kinematic structures of robots are often a subject of optimizations in the field of evolutionary robotics [12–14]. In this paper we took a similar evolutionary approach to the design of kinematic structure of manipulators.

Although KBE methods are used across engineering fields, the authors have not found a similar approach to produce a viable mechanical design of task-based robots in the literature. It should be stated that in most case studies in the literature, a KBE system is built on top of the already functioning design process in order to make it faster and more efficient. In this study, the authors aimed to present a KBE method to design single-purpose manipulators to replace universal robot manipulators for simple tasks.

This paper is structured as follows; Section 2 introduces the method overview, Section 2.1 describes genetic optimizations of a kinematic structure, Section 2.2 discusses a CAD model design and iterative design of individual mechanism links, and Section 3 then uses these methods in a case study.

2. Materials and Methods

The method overview is shown in Figure 2. First, the basic requirements and input parameters needed to start the robotic arm design were defined, such as the weight and dimensions of the manipulated object, target positions for the trajectory, the required movement time, and the dimensions of the workspace with obstacles. In Step 2, we estimated the size of the end effector based on the manipulated object dimensions. This step was intentionally chosen to be very generic. Since we did not yet know the forces acting on the end-effector to design or select the appropriate end-effector, we took only the size into consideration and estimate the volume envelope for the end-effector. The size of the end-effector volume was estimated to be 1.5 times the maximum dimensions of the manipulated object.

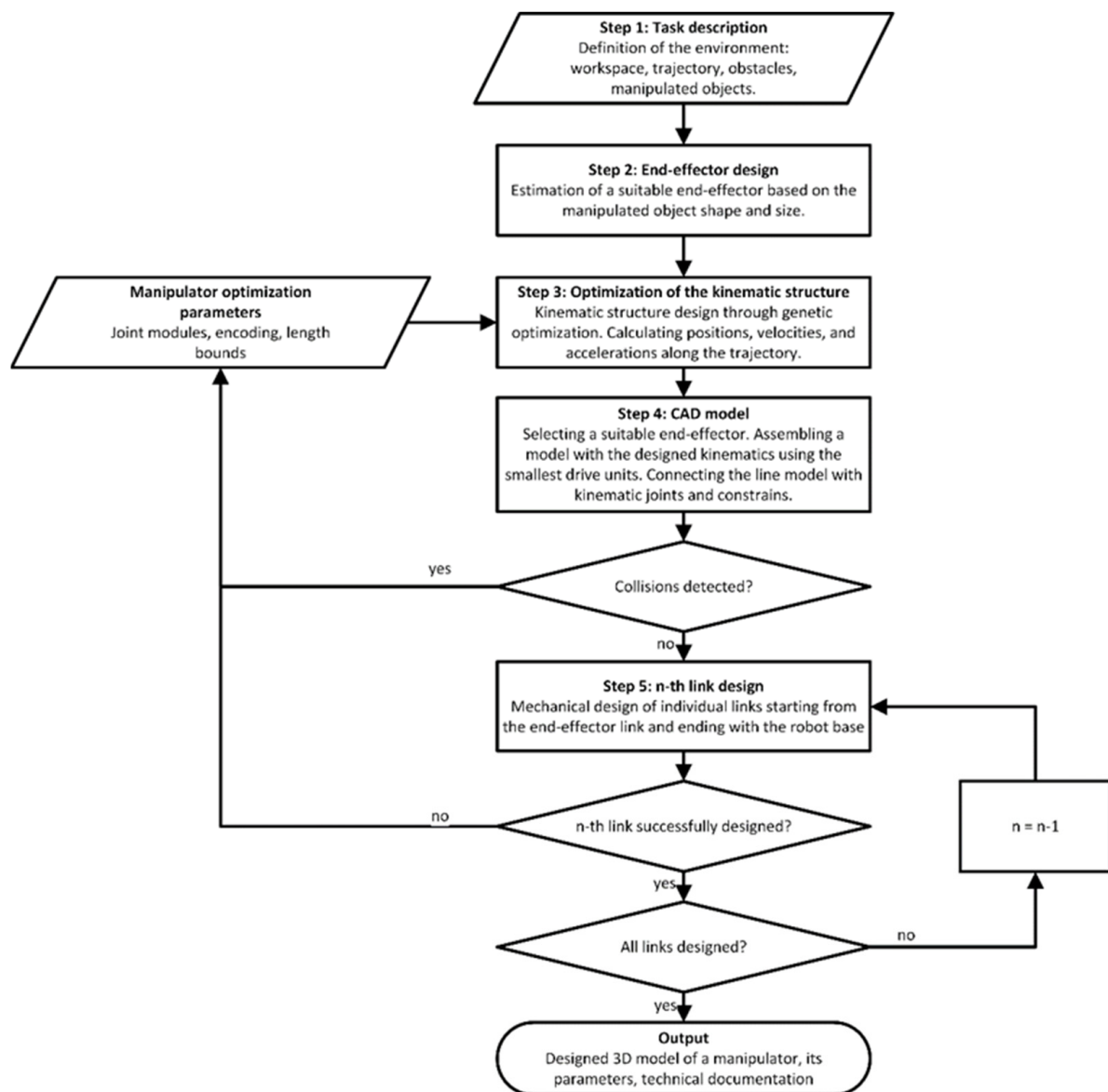


Figure 2. Our KBE system structure for designing robot manipulators.

Step 3 of the process was a genetic optimization of the kinematic chain. The dimensions of the manipulated objects, the end-effector, and the obstacles were used for collision checking during this step. The output of this block was a kinematic chain described as transformation between joints and the positions, velocities, and accelerations in joint space. Section 2.1 describes this step in more detail.

In Step 4 the end-effector was chosen, based on the manipulated object parameters, i.e., dimensions, weight, and accelerations during manipulation. The selection of the end-effector followed the manufacturer's instructions. The last manipulator link was designed to be compatible with the selected end-effector. A preliminary 3D model of the manipulator was designed to fit the optimized kinematic structure. A database of available drive units was prepared in advance. However, the preliminary design was assembled from only the smallest available drive units in the database. The structural components were also minimized by selecting the smallest available cross section. This preliminary 3D model was analyzed for possible collisions. If a collision is detected in this step, it is necessary to change the kinematic structure of the manipulator. If no collisions are detected, the design can continue to Step 5.

In this step, the individual links of the manipulator were iteratively modeled. The iterations included the selection of drive units, kinematic and dynamic analyses, shape optimizations, and other necessary steps. In the case that all the individual parts are designed successfully, the result is a full 3D model of the manipulator. If at any point the design iteration fails, it is necessary to modify the kinematic structure. The parameters for optimizing the kinematic structure are updated, and a new optimization is carried out.

2.1. Kinematic Synthesis through Optimization

The optimization took place as the second function block in our KBE system. It optimizes a kinematic structure of a manipulator for the functionalities specified in the first block. The optimization is based on a genetic algorithm implementation from MATLAB Global Optimization Toolbox. The algorithm creates a random population of individuals, evaluates their fitness, and creates a new population from the individuals with the best fitness value. In this case, the individuals are descriptions of robot kinematics. The evaluation and calculation of the fitness function was implemented on top of the MATLAB Robotics Systems Toolbox.

2.1.1. Robot Genotype Encoding

The robot kinematic structures are in the GA optimization represented by a genotype vector. The structure of a genotype vector for a 3 degree-of-freedom (DOF) manipulator is shown in Table 1. The genotype is a combination of link lengths and joint type. The length of the genotype depends on the number of DOFs in the robot and is set before optimization.

Table 1. Robot Genotype Structure.

Parameter	a_1	α_1	a_2	α_2	a_3	α_3
Description	Link 1 length	Joint 1 type	Link 2 length	Joint 2 type	Link 3 length	Joint 3 type
Upper bound	1000	2	1000	2	1000	2
Lower bound	275	0	250	0	225	0

The genotype vector is a vector of integers. The length is given in millimeters. The lower bound of the link length is different for each link. This is because we expected the joints towards the base to have more powerful and therefore bigger drive units. Towards the end effector, the requirements for joint torque are generally lower than in the base, and we could expect smaller drive units. The bounds can be modified for different applications. With our selected drive units, we have settled on lower bounds driven by a linear function (1). The maximum length is less important, since the fitness function is trying to minimize the length of the whole manipulator. For our experiments we have used 1000 which was more than enough for the used workspaces.

$$a_n = 300 - n \cdot 25 \text{ [mm]} \quad (1)$$

where: n is the link number and a_n is the length of the n th link.

The joint type describes the rotation of a joint relative to the previous joint, or in the case of the first joint relative to the world coordinate frame. We have used three different orientations shown in Figure 3. The transformation matrices for these joint types are defined in Equation (2).

$$\begin{aligned}
 \alpha_i = 0 \rightarrow T_i &= \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) & 0 & a_i \\ -\sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \alpha_i = 1 \rightarrow T_i &= \begin{bmatrix} \cos(\theta_i) & 0 & \sin(\theta_i) & a_i \\ -\sin(\theta_i) & 0 & \cos(\theta_i) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \alpha_i = 2 \rightarrow T_i &= \begin{bmatrix} 0 & 0 & -1 & a_i \\ -\sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ \cos(\theta_i) & \sin(\theta_i) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \quad (2)$$

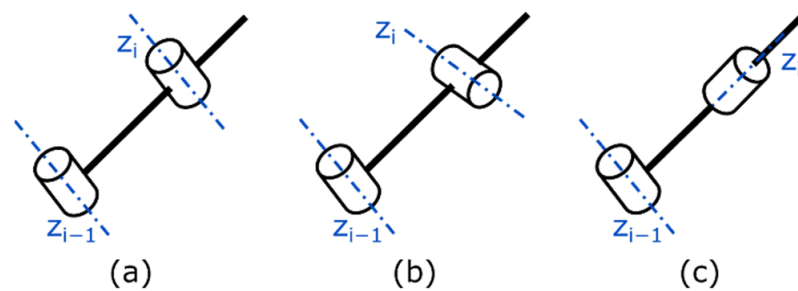


Figure 3. Joint types; (a) $\alpha = 0$, (b) $\alpha = 1$, (c) $\alpha = 2$.

Transformation matrices for an example 3 DOF kinematic structure with the genotype [300, 0, 400, 0, 200, 1] are shown in (3).

$$T_E = \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) & 0 & 0.3 \\ -\sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta_2) & \sin(\theta_2) & 0 & 0.4 \\ -\sin(\theta_2) & \cos(\theta_2) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) & 0.2 \\ -\sin(\theta_3) & 0 & \cos(\theta_3) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

where T_E is the transformation matrix from the base coordinate frame to the end effector coordinate frame and $[\theta_1, \theta_2, \theta_3]$ is the current joint configuration.

2.1.2. Evaluation

The genetic algorithm (GA) works by evaluating each potentially optimal manipulator kinematics using a cost function. This cost function is a weighted sum of three values (3): the positioning error of the manipulator (4), its length (5), and a measure of collisions (Figure 4). The weights were experimentally set as $weight_1 = 100$, $weight_2 = 1$, $weight_3 = 10$.

$$cost = weigh_1 \cdot cost_1 + weigh_2 \cdot cost_2 + weigh_3 \cdot cost_3 \quad (4)$$

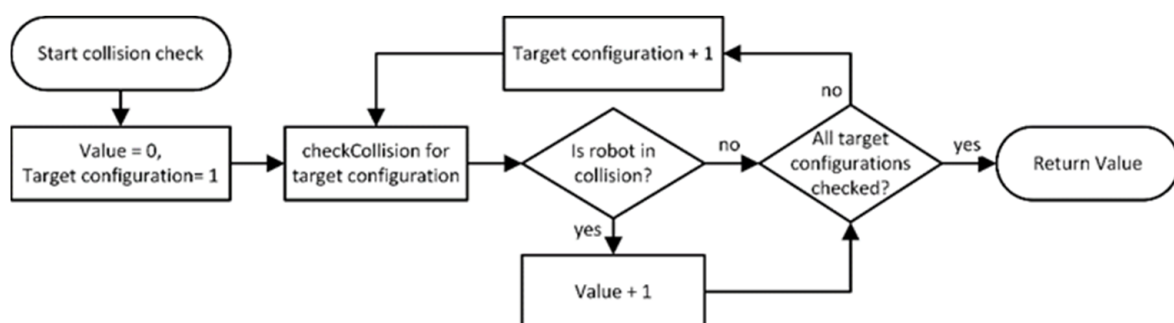


Figure 4. Collision measure calculation.

The evaluation was performed using the Robotic System Toolbox in MATLAB. For each individual in a GA generation, a rigidBodyTree object was created based on the encoded genes. This object represents the robot and is used to calculate the inverse kinematics (IK) and collisions. A joint configuration was calculated using IK for all target points. The magnitude of the pose error was returned by the IK solution as PoseErrorNorm. The value of cost₁ is the average of PoseErrorNorm values for all trajectory points.

$$cost_1 = \frac{1}{n} \sum_{i=1}^n PoseErrorNorm_i \quad (5)$$

The second part of the cost function is a simple sum of the lengths of the manipulator segments.

$$cost_2 = \sum_{i=1}^n a_i \quad (6)$$

The third part of the cost function is a measure of collisions. This value uses the checkCollision function to measure collisions between the manipulator and the workspace obstacles. This collision check was done for all target configurations.

2.2. Cad Model

In previous steps, the dimensions of the end-effector were estimated. The first step in designing a robotic arm is the selection of a real usable end-effector. End-effectors for manipulation tasks were selected based on the shape and dimensions of the handled object, on the position of the gripping surfaces, on the accelerations and external forces, and on the way that the handled object is placed in the environment. In this study, we considered only available off-the-shelf end-effectors. Their selection followed the instruction of the manufacturer, using the software tools available on the manufacturers' websites [15–17].

Based on the end-effector connection flange and the proposed kinematic structure, a preliminary CAD model of the whole arm was created. The model was assembled using the smallest drive units available in the database and were used in all the joints. The dimensions of the structural elements of the robotic arm then depend on the drive units connecting dimensions. This manipulator model with the smallest drive units was then subjected to analysis, with the aim of finding possible collisions between the individual parts of the manipulator itself and within the entire workplace. If collisions are found, the proposed kinematic structure is not accepted, and the design process returns to the kinematic optimization step. However, the input parameters of the optimization were adjusted in such a way as to reflect the reasons why it was not possible to proceed to the next design stage. For example, the lower and upper bounds for the link lengths were adjusted. If no collisions are found, the process continues to the next step in which the individual parts of the robotic arm are designed.

2.2.1. *n*-th Link Design

In this step, the force and torque effects were taken into account. The individual elements of the manipulator were designed iteratively, beginning from the end-effector towards the base. Thus, the first proposed element is the part connecting the end effector with the next drive unit. Figures 5 and 6 are flow charts that describe the design and general procedure of the *n*-th component, including the respective drive units. The design process depends on the link length. Short links consist of one structural element and long links from three elements.

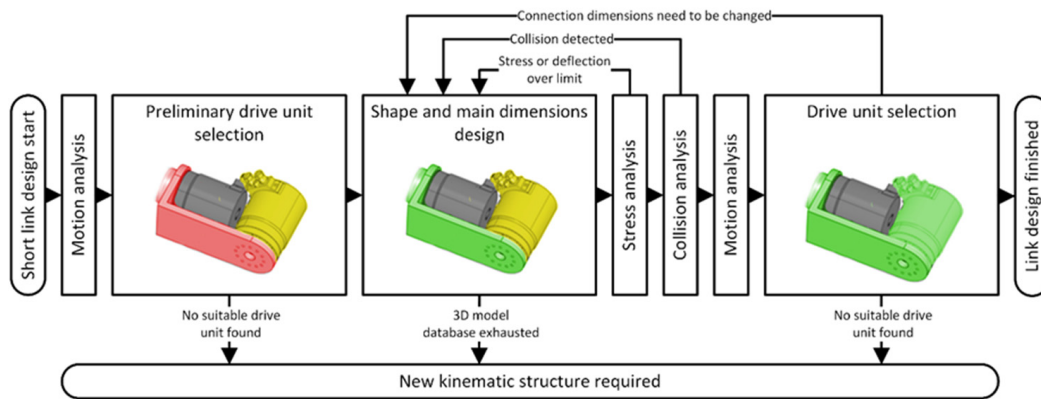


Figure 5. Design procedure of the n -th link if the link consists of one element.

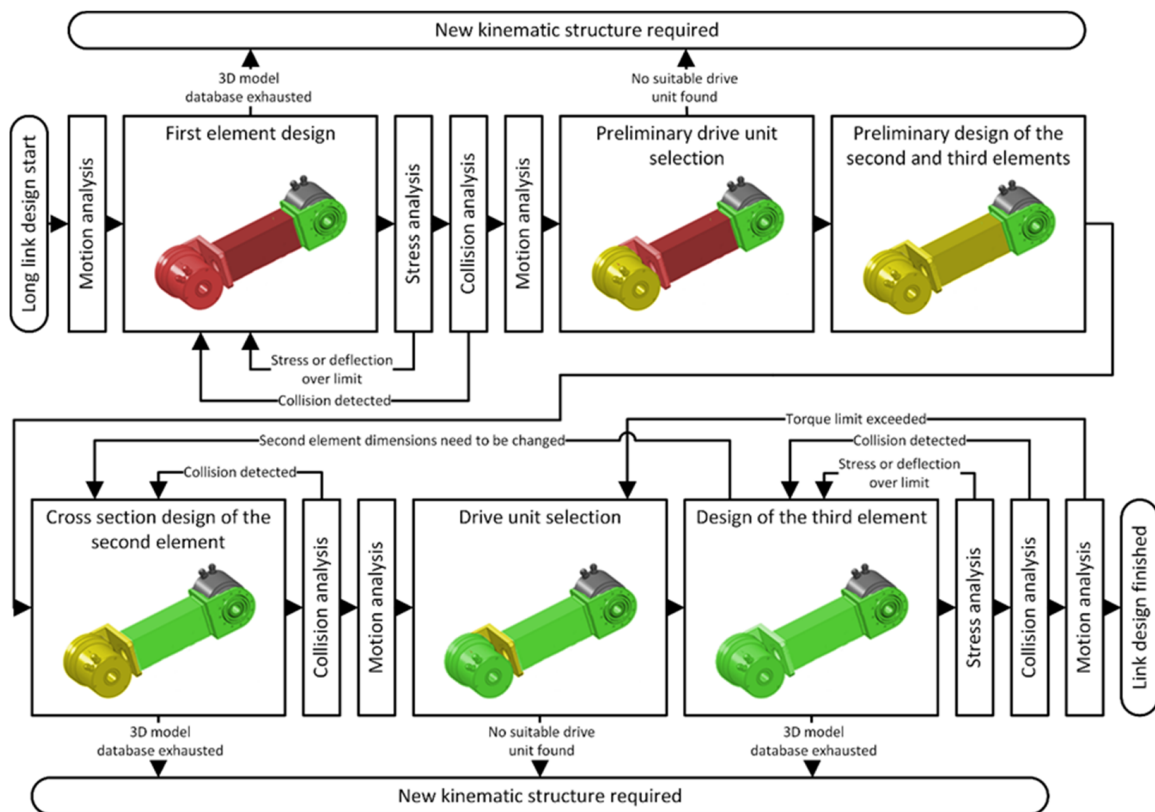


Figure 6. Design procedure of the n -th link if the link consists of three elements.

As part of this procedure, we divided the elements of the robotic arm into two main categories. The first contains off-the-shelf or otherwise reused over elements (drive units, end effectors, sensors, etc.), for which we cannot change the parameters, shape, or individual dimensions. The second category contains mainly load-bearing and connecting elements, which were designed on the basis of predefined rules, but we can change the shape, dimensions, and selected parameters (e.g., material). Based on the lengths of the individual links in the proposed kinematic structure of the robotic arm, one of the two design paths of the n th element was chosen.

The length of the kinematic n -link was compared to the diameter of the previous ($n - 1$) link's drive unit, or in the case of the first link, the end-effector length was used. If the length of the n -link exceeds the size of the drive unit (or end-effector) size by more than a factor of four, we considered it long enough to be built from three elements bolted, welded, or glued together. Otherwise, the n -th link will be built from just one structural element. Figure 7 shows examples of links designed from one and three structural elements.



Figure 7. Examples of the n -th link; (a) from one part, (b) from three parts.

If the link consists of one piece of material, kinematic and dynamic analysis of the manipulator's 3D model is performed first. Within these analyses, a 3D model created on the basis of the proposed kinematic structure with the smallest possible overall dimensions was used. Within this model, the original elements were gradually replaced by elements designed according to the procedure indicated in Figures 5 and 6. The data obtained from these analyses were then used in the preliminary design of the drive unit located in the following joint (using a modified version of the DrivePicker software tool [18]). Based on the drive unit dimensions, relative position to the previous joint, and the respective connection dimensions (flanges), a suitable parameterized 3D model was selected from the database. The model has its parameters set appropriately for this use case. The next step of the design was similar to a design study in CAD system SolidWorks, which aimed at the dimensional optimization of the element in such a way as to suit both in terms of strength and especially in terms of maximum deformation, while achieving the lowest possible weight. If it is not possible to get the desired results with this element, another candidate 3D model is selected from the database as the basis of the n -th element and the process is repeated until a suitable solution is found. Then, a collision analysis is performed again. The purpose of this analysis was to determine whether the model update caused no new collision states. If no collisions are found, the kinematic and dynamic analysis is performed again, and the selected preliminary drive unit is checked. If it is possible to use the originally designed drive unit or a unit with the same connection dimensions, it is possible to end the design of the n -th element and continue with the design process for the next link in the manipulator. In case it is necessary to use a drive unit with different connection dimensions, the design process needs to be repeated from the step in which the basic shape and dimensions of the n th element were designed, and the design process is repeated until a suitable solution is found.

The process is similar in the case where the n -th link consists of three elements. Again, individual analyses and optimization processes are gradually performed, on the basis of which both parts of the n -th element itself and the drive unit located in the respective joint were designed.

Regardless of whether the n -th link consists of one or three elements, the iterative process is currently almost completely automated. This was made possible by the existence of the already mentioned database, in which the necessary information about the drive units, end-effectors, and other used elements is stored. This database also contains parameterized 3D models from which the manipulators can be assembled. The automated design has been implemented using the SolidWorks CAD API, which can be used to automate the individual processes outlined in the block diagram in Figures 5 and 6. Currently, the only process that is not fully automated is performing the kinematic and dynamic analyses. This is a process that is quite problematic due to the variety of possible settings and still requires control and possible interventions from a human. Nevertheless, even this part of the design of the n -th element has been significantly simplified and accelerated thanks to the automation of data transfer between the design steps. The process also considered the

case that it will not be possible to complete the design of the n -th member of the robotic arm, whether due to the overloading of the drive units, deformations exceeding the required level, or due to collisions. In that case, the process returned to the kinematic optimization step with updated optimization parameters.

3. Results

The functionality of our method from previous chapter was tested on a case study. The chosen task is a manipulation between two industrial machines. The models of this workspace are shown in Figure 8. The manipulated object is a cylindrical steel object weighting 2.15 kg. The workspace environment was bounded by the industrial machines, which acted as obstacles. The task trajectory was defined by four target points, two on each machine, which define the position and orientation of the manipulated object in space. The drive unit database consists of compact drive units from HarmonicDrive CanisDrive series and Spinea DS series.

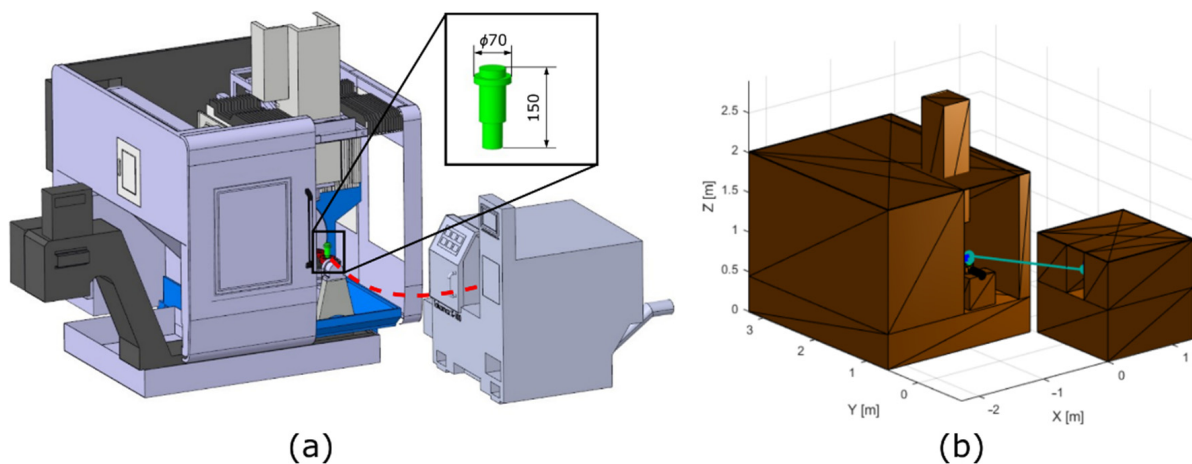


Figure 8. Task workspace; (a) CAD model, (b) model in MATLAB.

This workspace has been modeled in MATLAB as an environment for the evaluation in the genetic optimization algorithm described in Section 2.1. The output of the genetic algorithm is a 5 DOF manipulator shown in Figure 9. For clarity, let us call this kinematic structure “A”. The output also includes joint angle values throughout the movement. These values were then used for motion analyses in CAD software.

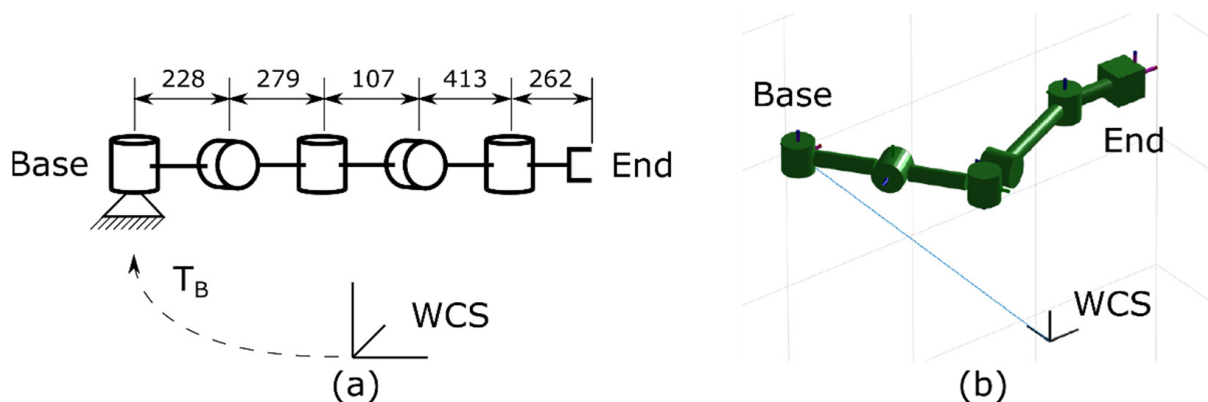


Figure 9. Optimized kinematic structure A; (a) schematic, (b) view in MATLAB.

As stated in previous chapters, the kinematic structure was used to build a 3D model of the manipulator with the use of the smallest drive units available in the element database. This model was then used for motion analysis to measure the forces and torques acting on

drive units and mechanical elements in the manipulator. Then the iterative process from Section 2.2.1 was run to check and optimize the design. The preliminary design is shown in Figure 10.

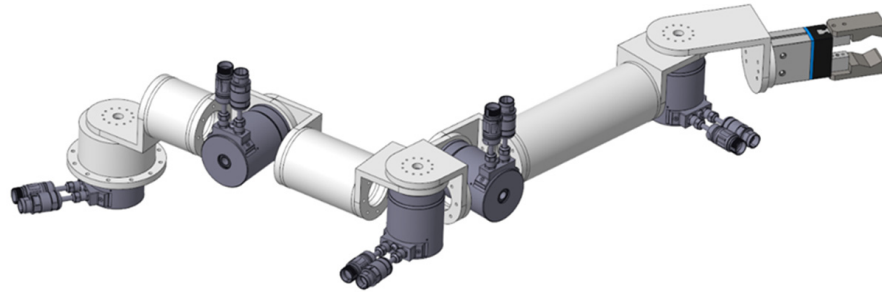


Figure 10. Initial 3D model for kinematic structure A.

The design process of the CAD model for the kinematic structure A failed in the design iteration of the third link due to system collisions (Figure 11). When designing the drive unit of the third joint, a larger drive unit was chosen, for which the third arm member with the required length of 107 mm could not be realized, and therefore the kinematic structure A was eliminated by the algorithm and the design returns to the design of a new kinematic structure. The options for the optimization algorithm to generate a new kinematic structure were changed to reflect this collision. The minimum length for a manipular link was extended to 200 mm.



Figure 11. A drive unit change during design iterations; (a) preliminary, (b) after change, with the resulting collision (red ellipse).

After changing the optimization options, a new kinematic structure B was generated (Figure 12). Again, it is a 5 DOF manipulator. The difference between Structures A and B is not only in link lengths, but also in the rotation of individual joints.

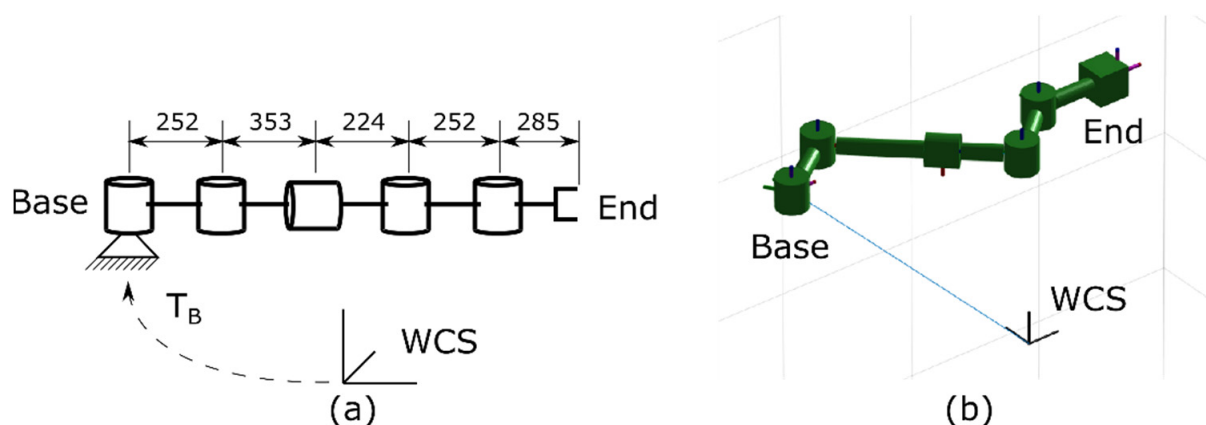


Figure 12. Kinematic structure B; (a) schematic (left), (b) the same structure in Matlab (right).

The new kinematic structure B went through all iterations of the design successfully. Figure 13 shows the design in the preliminary phase and Figure 14 shows the final design. The change in components can be seen in Figure 14; the structural components have been replaced with thinner beams or brackets. The structural components towards the base of the manipulator have been replaced with steel parts. As part of the arm design, the drive unit in the third link was moved closer to the base, in order to minimize the force effects of its weight on the arm. The resulting courses of torque, velocities, and forces in the individual joints of the arm can be seen in Figures 15 and 16.

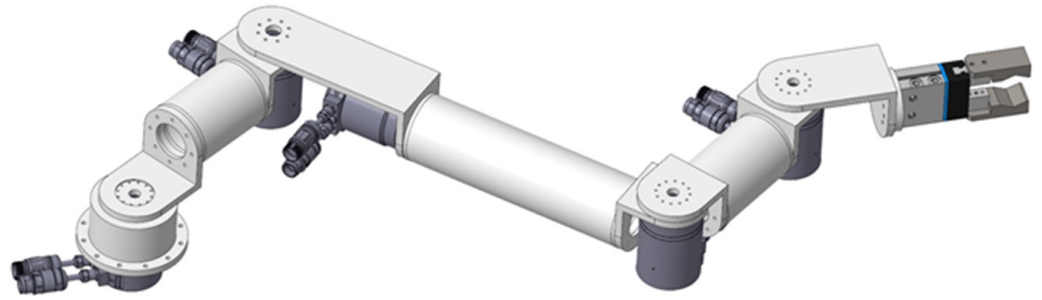


Figure 13. Kinematic structure B: initial preliminary CAD model. (All aluminum profiles and the smallest drive units).

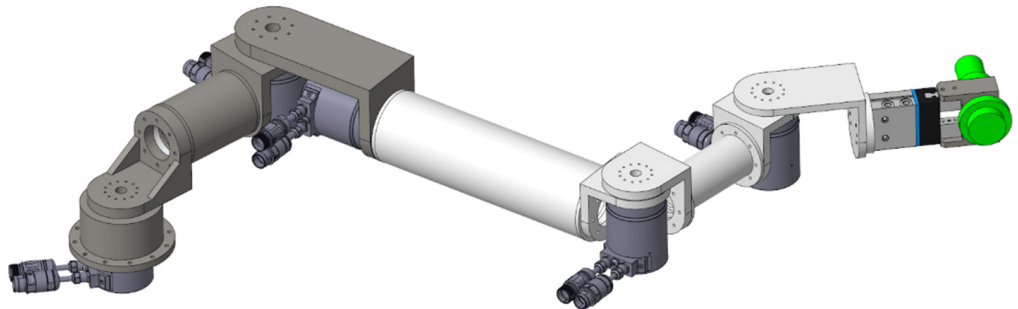


Figure 14. Kinematic structure B: resulting CAD model, after all design iterations. (darker parts: steel, lighter parts: aluminum).

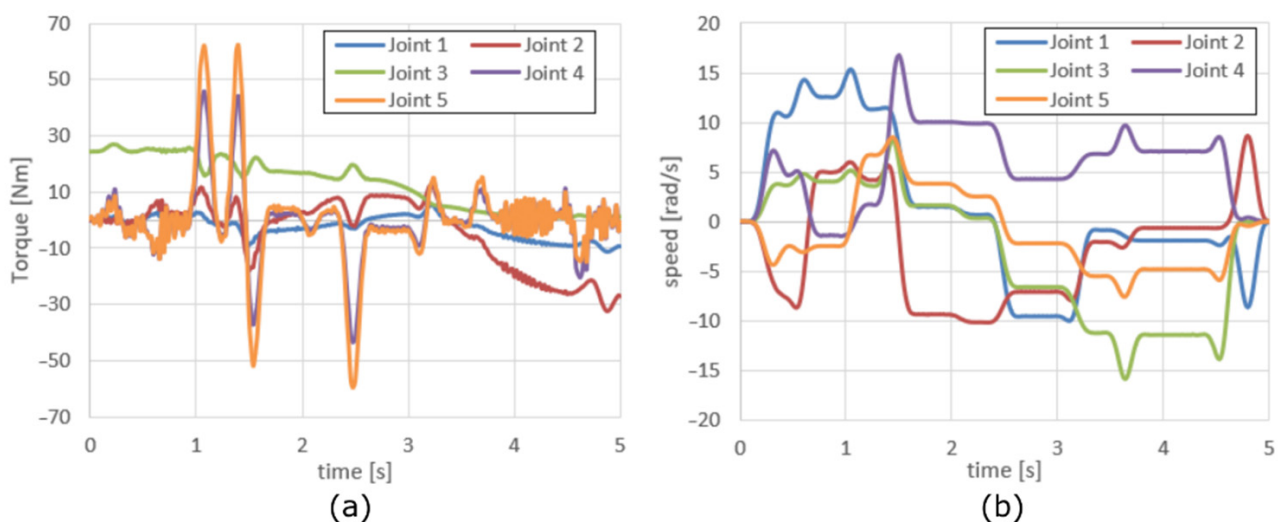


Figure 15. Measurements during movement; (a) torque, (b) joint velocity.

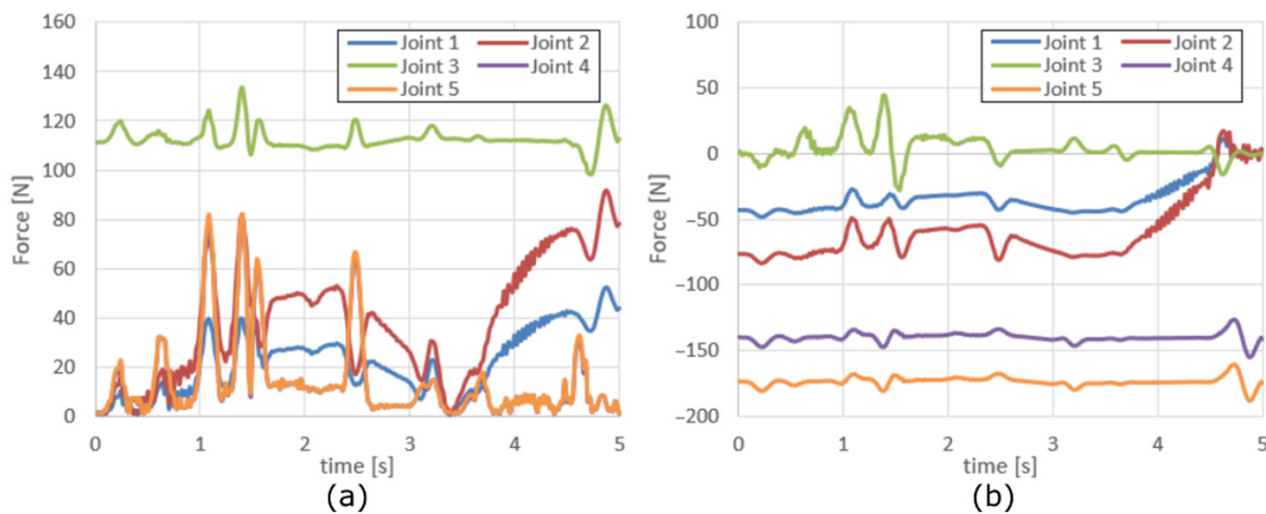


Figure 16. Measurements during movement; (a) Radial forces, (b) axial forces.

Table 2 shows the basic parameters of the designed robotic arm; these are the selected end-effector and drive units, the designed cross sections of individual links, and the total weight of the manipulator.

Table 2. Basic parameters of the designed robotic arm.

End-Effector	Festo DHPS 25-A-N
Drive unit 1	HD Canis Drive 14A (Ratio 80)
Drive unit 2	HD Canis Drive 14A (Ratio 100)
Drive unit 3	HD Canis Drive 17A (Ratio 80)
Drive unit 4	HD Canis Drive 17A (Ratio 80)
Drive unit 5	HD Canis Drive 17A (Ratio 100)
Link 2 structural profile	Ø 60 × 2 mm, aluminium
Link 3 structural profile	Ø 100 × 3 mm, aluminium
Link 4 structural profile	Ø 90 × 3 mm, steel
Total weight	29.75 kg

The sizes of the structural flange elements are determined by the sizes of the drive unit flange. These elements can be, however, optimized by changing the material thickness without changing the overall dimensions. These changes will only affect the total weight and are therefore not included in the resulting table.

4. Discussion

The use of optimization algorithms for kinematics and automated CAD design of robotic arms is a viable process to achieve better results in terms of reducing operating and acquisition costs of robotic workplaces. This mainly applies to applications where it is possible to use robotic arms with a lower number of DOFs. With manipulators that have six DOFs or more, these algorithms can still significantly help in designing a suitable kinematic structure, especially in cases with a large number of collision objects within the workplace. However, most of the algorithms available do not consider the real dimensions of the elements from which the robotic arm should be built. As a result, the kinematic structures designed in this way cannot be used. Therefore, a system was designed that takes into account real elements. The kinematic structure was optimized using genetic algorithms first and then real elements were iteratively selected from a database and evaluated. Therefore, it is a complex system that aims to simplify and speed up the design process of robotic

arms. Its functionality was presented on the design of an arm for handling the workpiece between two industrial machines.

In articles dealing with knowledge-based design of robotic arms, the authors often propose the dimensions of individual links, joints, and other elements of robotic arms according to some generic rules or set fixed dimensions for given design elements. However, this procedure will significantly affect the achieved results, especially in terms of possible collisions between individual parts of the robotic arm or between the robotic arm and other elements of the robotic workplace. It turns out that many approaches from the literature produce kinematic structures for which the mechanical design with all necessary components cannot be completed. The dimensions of the individual elements of the arms tend to be significantly larger than originally planned. The design itself may then fail, as was the case of the first proposed kinematic structure “A” in Section 3. However, the inclusion of a selection of elements with dimensions and overall parameters corresponding to real robotic arms significantly complicates and prolongs the process of designing kinematic structures. For this reason, procedures, algorithms, and software tools have been developed to automate and greatly simplify this process. At the same time, we are moving much closer from theoretical designs of kinematic structures to the design of usable devices, which we consider to be a contribution to the field.

Section 3 describes the design of a robotic arm with five DOFs. There is a lot of competition in the area of robotic arms with five and six DOFs. Due to the fact that commonly available power units were used in the design and so far, the individual structural elements have not been fully topologically optimized, the resulting design cannot compete with the commonly available and optimized robotic arms of established manufacturers. However, the situation could change significantly when an arm with three or four DOFs will be sufficient for the selected application. For these applications, it is still possible to use an arm with a greater number of degrees of freedom. From the point of view of acquisition and operating costs, a manipulator with lower number of DOFs and with an optimized kinematic structure offers significantly better results. If a task requires a manipulator with seven DOFs or more, the commercial options are very limited. In these situations, designing your own robotic arm is the only viable option. Additionally, tools for designing a suitable kinematic structure can be of great help in these situations.

Author Contributions: Conceptualization, M.M. and R.P.; methodology, M.M.; software, R.P. and Z.Z.; validation, R.P., Z.Z. and A.B.; formal analysis, Z.Z.; investigation, R.P.; resources, R.P. and Z.Z.; data curation, Z.Z. and R.P.; writing—original draft preparation, R.P., Z.Z. and M.M.; writing—review and editing, R.P. and Z.Z.; visualization, R.P.; supervision, M.M.; project administration, M.M.; funding acquisition, M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Regional Development Fund in the Research Centre of Advanced Mechatronic Systems project, project number CZ.02.1.01/0.0/0.0/16_019/0000867 within the Operational Programme Research, Development and Education. This article has been also supported by specific research project SP2022/67 and financed by the state budget of the Czech Republic.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Verhagen, W.J.C.; Bermell-Garcia, P.; van Dijk, R.E.C.; Curran, R. A critical review of Knowledge-Based Engineering: An identification of research challenges. *Adv. Eng. Inform.* **2012**, *26*, 5–15. [[CrossRef](#)]
2. Chapman, C.B.; Pinfold, M. The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure. *Adv. Eng. Softw.* **2001**, *32*, 903–912. [[CrossRef](#)]

3. Reddy, E.J.; Sridhar, C.N.V.; Rangadu, V.P. Knowledge Based Engineering: Notion, Approaches and Future Trends. *Am. J. Intell. Syst.* **2015**, *5*, 1–17. [[CrossRef](#)]
4. Xu, Q.L.; Ong, S.K.; Nee, A.Y.C. Function-based design synthesis approach to design reuse. *Res. Eng. Des.* **2006**, *17*, 27–44. [[CrossRef](#)]
5. Cui, J.-J.; Wang, D.-Y. Application of knowledge-based engineering in ship structural design and optimization. *Ocean Eng.* **2013**, *72*, 124–139. [[CrossRef](#)]
6. La Rocca, G.; Krakkers, L.; Van Tooren, M. Development of an ICAD generative model for blended wing body aircraft design. In Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, Georgia, 4–6 September 2002. [[CrossRef](#)]
7. La Rocca, G.; Van Tooren, M. Enabling distributed multi-disciplinary design of complex products: A knowledge based engineering approach. *J. Des. Res.* **2007**, *5*, 333. [[CrossRef](#)]
8. Kim, B.J.; Yun, D.K.; Lee, S.H.; Jang, G.-W. Topology optimization of industrial robots for system-level stiffness maximization by using part-level metamodels. *Struct. Multidiscip. Optim.* **2016**, *54*, 1061–1071. [[CrossRef](#)]
9. Tor, S.B.; Lee, S.G.; Britton, G.A.; Zhang, W.Y. Knowledge-based functional design of industrial robots. *Int. J. Prod. Res.* **2008**, *46*, 4501–4519. [[CrossRef](#)]
10. Cooper, D.; La Rocca, G. Knowledge-based techniques for developing engineering applications in the 21st century. In Proceedings of the 7th AIAA ATIO Conference, Belfast, Northern Ireland, 18–20 September 2007; Volume 1, pp. 146–167. [[CrossRef](#)]
11. Cooper, S. *Achieving Competitive Advantage through Knowledge-Based Engineering: A Best Practice Guide*; Department of Enterprise Integration Cranfield University: Cranfield, UK, 1999.
12. Chocron, O.; Bidaud, P. Evolutionary algorithms in kinematic design of robotic systems. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Grenoble, France, 11 September 1997; Volume 2, pp. 1111–1117. [[CrossRef](#)]
13. Paredis, C.J.J.; Khosla, P.K. Kinematic Design of Serial Link Manipulators From Task Specifications. *Int. J. Robot. Res.* **1993**, *12*, 274–287. [[CrossRef](#)]
14. Singh, S.; Singla, E. Realization of task-based designs involving DH parameters: A modular approach. *Intell. Serv. Robot.* **2015**, *9*, 289–296. [[CrossRef](#)]
15. Automation Technology and Technical Education Solutions | Festo GB. Available online: <https://www.festo.com/gb/en/> (accessed on 10 March 2022).
16. Competence Leader for Gripping Systems and Clamping Technology—SCHUNK. Available online: https://schunk.com/de_en/homepage/ (accessed on 10 March 2022).
17. Home—Zimmer Group. Available online: <https://www.zimmer-group.com/en/> (accessed on 10 March 2022).
18. Zeman, Z.; Mihola, M.; Suder, J. Design of algorithms for automatic selection of drive units for mechatronic devices. *MM Sci. J.* **2021**, *2021*, 4362–4370. [[CrossRef](#)]