

RESEARCH AND DEVELOPMENT OF A SOFTWARE TOOL FOR PARAMETRIC MODELING OF ROBOTIZED WORKPLACES

ADAM BOLES LAVSKY¹, MILAN MIHOLA¹,
ROSTISLAV WIERBICA¹, JAN BEM¹, TOMAS SPURNY¹

¹VSB - Technical University of Ostrava, Department of Robotics,
Ostrava, Czech Republic

DOI: 10.17973/MMSJ.2022_06_2022061

e-mail : adam.boleslavsky@vsb.cz

Designing a robotic workplace is a time and knowledge intensive process. It requires systems engineers who specialise in the design of this type of workplace and have the appropriate know-how in this area. Due to the time-consuming nature and the small number of experts, a small number of solution options are often created, from which the most suitable one is subsequently developed.

The aim of the development was to create a tool that would simplify the design of robotic workplaces. The result is a software tool integrated into the SolidWorks CAD system. The tool uses a database of models. The connection between the developed software tool and the database is realized through the API of the SolidWorks CAD system.

Thanks to this tool, it is possible to simplify and speed up the work of workplace design. More variant solutions can be developed and overall better results can be achieved.

KEYWORDS

API, SolidWorks, Parametric modelling, Macro, robotized technological workplace, Generating the model, Parametric system design

1 INTRODUCTION

The development of a robotic workplace (RW) involves many requirements that the developer must then work with. Often these requirements change during development and the workstation has to be redesigned or rebuilt completely. The development of a workplace is therefore a very time-consuming process. Currently, the demand for the development of robotic workstations is so great that system integrators have difficulty meeting it. And with the further development of robotics, this problem is getting worse. Therefore, ways are being sought to simplify and speed up at least part of this process.

Different tools and resources can help RW designers. These may include, for example, various libraries and databases of 3D models of both individual parts and complete devices, which in many cases are freely available [KEYENCE 2022], [FESTO 2022], [PARTcommunity 2022], [TraceParts 2022], [GrabCAD 2022]. RW design companies have their own, internal databases of components and devices that have been used in previous designs. In many cases, such pre-prepared elements need to be modified before they can be used. However, this is a better solution in terms of the time required than preparing elements from the very beginning.

Another useful source of 3D models can be the online configurators of parts and assemblies. For example, Hiwin [HIWIN 2022] offers a trolley configurator, igus [igus 2022] offers tools for designing linear guides or motion screws, or INTERROLL [INTERROLL 2022] offers a conveyor configurator. The user selects the parameters that are suitable for him. In this way, not only 3D models can be obtained, but also information about the partial financial costs, which is also very beneficial from the RW design point of view.

Parameterized models are also available within some libraries and databases, often bound by various functions and conditions. By changing their parameters or variables, it is possible to obtain the desired 3D model very quickly and thus again significantly reduce the design time.

Another option is to create your own software tools. In [Zeman 2021] and [Mihola 2020] tools for the design of drive units and robot arms are described, in [Jayakiran 2018] a tool for creating a 3D model of battery packs is described, in [Jingzh 2020] an application for the design of Single Screw Extruders is described, in [Esanakula 2021] a tool for the design of gears and in [Zhang 2019] a tool for the creation of box models.

In all these cases, these are very narrowly focused software tools. The aim of this work was to design a much more complex tool whose output would be 3D models of the RW according to the specified parameters. Creating a similar type of application requires knowledge not only of programming, but also of the subject matter. In the case of creating an application whose output is to be, among other things, a 3D model, it is often advantageous to use the capabilities of existing CAD systems. In many cases, these systems have Application Programming Interfaces (API) or tools for creating modules that can be used to extend the capabilities of the CAD system. Combined with the ability to create parameterizable 3D models, this is a very powerful development tool. It was also necessary to create a database system that would be used to store both the source and the resulting 3D models.

The basis of this paper is based on a master thesis [Boleslavsky 2021]. Its aim is to present the results obtained, which have no known parallel at present.

2 SOFTWARE TOOL FOR CREATING ROBOTIC WORKPLACES

The following chapter is devoted to the description of a newly developed software tool for creating conceptual designs of robotic workplaces. The selected algorithms required for the proper functioning of this tool and its user environment will be described in turn.

2.1 Description of working with models

This tool was created as an add-on module to the SolidWorks CAD system. Thanks to this, its user can also access other functions of this CAD system. It is therefore possible not only to create RWs using the user interface of the designed software tool from elements contained in the prepared database, but also to add custom models or to perform kinematic or strength analyses of selected parts of the workplace. Once the design is complete, a 3D model of the RW is available, which can be further manipulated as a manually created model.

Custom 3D models contained in the database can be edited before use by using macros with SolidWorks CAD API commands.

When creating the RW model, the sub models are inserted relative to the main coordinate system of the assembly. The calculation of the correct placement location and its rotation is

done by calculating the rotation matrices and inserting them into the transformation matrix function of the SolidWorks API.

```
Dim TransformData() As Double
ReDim TransformData(0 To 15) As Double
TransformData(0) = 1
TransformData(1) = 0
TransformData(2) = 0
TransformData(3) = 0
TransformData(4) = 1
TransformData(5) = 0
TransformData(6) = 0
TransformData(7) = 0
TransformData(8) = 1
TransformData(9) = 0.5
TransformData(10) = 0.5
TransformData(11) = 0.5
TransformData(12) = 1
TransformData(13) = 0
TransformData(14) = 0
TransformData(15) = 0
Dim TransformDataVariant As Variant
TransformDataVariant = TransformData
Dim swMathUtil As Object
Set swMathUtil = swApp.GetMathUtility()
Dim swTransform As Object
Set swTransform = swMathUtil.CreateTransform((TransformDataVariant))
boolstatus = swComp.SetTransformAndSolve2(swTransform)
```

Figure 1. API function for device insertion

The matrix is composed of 16 values whose order corresponds to the notation in 4x4 arrangement. The graphical form of this matrix is indicated in Fig.2.

0	1	2	13
3	4	5	14
6	7	8	15
9	10	11	12

Figure 2. Visualization of the transformation matrix

The numbers filled in this notation can be used to control the rotation of the model, its position and scale. Specifically, the entries in fields 0 through 8 control the position and angle of rotation of the model relative to the "x", "y", and "z" axes of the assembly model. The data in fields 9 to 11 are used to determine the position of the inserted model in the assembly workspace. The scale of the model can be adjusted using the data in field 12. The data in fields 13 to 15 are not used.

To place the model where it needs to be by filling in these fields, the user needs to calculate the rotation matrices. These are the rotations about the "x", "y" and "z" axes. Each rotation has its own matrix.

The specific values for fields 0 to 8 of the transformation matrix can be determined using relations (1) to (3). First, the calculations of each matrix are performed separately and then their product is calculated.

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \quad (1)$$

$$R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (2)$$

$$R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

The values obtained in this way are then used to insert the model into the assembly at the desired location and with the desired rotation.

The whole calculation is done automatically, based on six variables that are defined by the user.

The tool also performs other necessary calculations to solve the arrangement, layout, and shapes of some models.

Therefore, in order for the user to work with the models, variables that are part of the tool and control the required dimensions must be created. These variables are entered using a user interface in the form of a floating window, where the user fills in the dimensions and ticks the boxes, thus setting the variables that are then reflected in the prepared model and the overall working scene.

The user fills in the necessary data and confirms it with the appropriate button. The tool processes these data, calls the necessary function, and then loads the model from the library into the SolidWorks environment. Here the model can be modified according to the requirements, and then placed in the defined position in the work set. A schematic of this process is shown in Fig.3.

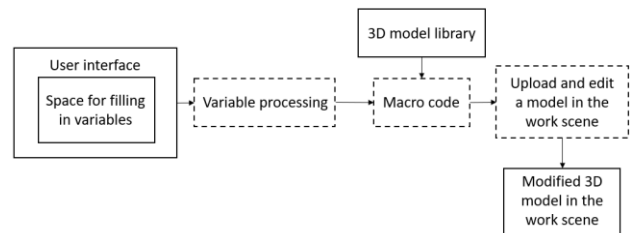


Figure 3. Block diagram of how the tool works

2.2 Basic description of the RW design tool

As already mentioned, the software tool works with the design of a reconfigurable 3D model of the robotic workplace. More precisely, the tool works with certain structural and mechanical elements that generally make up the RW. This tool is intended to serve system integrators of robotic technological workplaces to facilitate and speed up their work. Also, when using this tool, a greater success rate in creating designs and a better layout of machines within the proposed work cell is expected, allowing for a greater success rate of intelligent placement and enforcement of initial designs.

Using the tool, the user can insert things from the database created for the tool. Currently the basic version of the database contains one type of cuboid shaped cell with everything needed for such a model. This means the cell chassis, cover plates, cover glasses and doors. Support tables are also part of the cell design. As for the other models of the database, there are four types of robots, one type of conveyor, nine different positioners, two types of end-effectors, three types of measuring devices, three types of marking devices and three elements from the RW security area. At least one type from each of the most commonly used devices for robotic workstation design. The tool has limitations so far, such that only five pieces from each type of device can be inserted using the tool. However, this is sufficient for the first version and there is no problem to increase this value.

The structure of the user environment consists of a set of floating user windows that are logically arranged and linked.

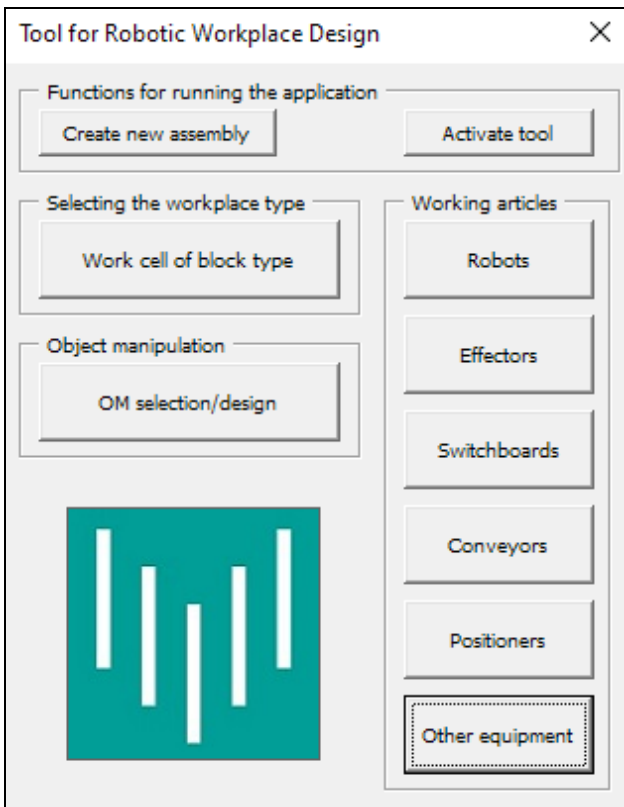


Figure 4. Initial floating window

The initial menu (Fig. 4) contains several buttons with device names. These buttons serve as gateways to other floating windows, which are used to design and deploy models of the same named devices. Fig. 5 shows the structure of the floating windows of the proposed software tool in the form of a block diagram.

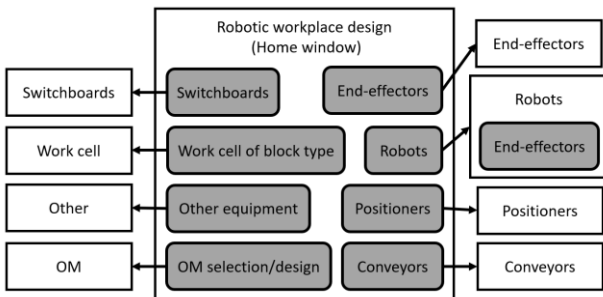


Figure 5. Block diagram of the structure of floating windows

Among other things, the tool also includes a function to create an assembly in which the user will create his model and a function to activate this device. The user can close the tool during his/her design, work on the model within the capabilities of the SolidWorks CAD system, and then turn the tool back on using the icon to continue the RW design.

Work with individual models is done in individual floating windows. There are buttons and fill-in fields that allow you to insert, delete and move models of the named devices according to the user's needs and specifications. All models have these basic functions.

Within the tool, the models are notionally divided into dimensionally invariant and parametric. In the case of dimensionally invariant ones, their layout dimensions cannot be changed. These are mainly models of parts or assemblies that are taken from distributors or external databases. These are

purchased products such as sensors, measuring devices or safety equipment. Parametric models can be changed or further modified according to certain options and rules. Some models have special functions. For example, they can be scaled by changing the value of the transformation matrix variable at field level 12. The special functions of the models are created by adapting to the needs of the workplace design. Which means that models can be further divided into models with and without specific functions.

2.3 Example of working with demonstration models

The preceding text shows that it contains four different types of models. Non-parametric models without special functions, non-parametric models with special functions, parametric models without special functions and parametric models with special functions.

Fig. 6 shows an example of a floating window with a non-parametric model with a special function, which can be used to insert models of industrial robots into the RW model.

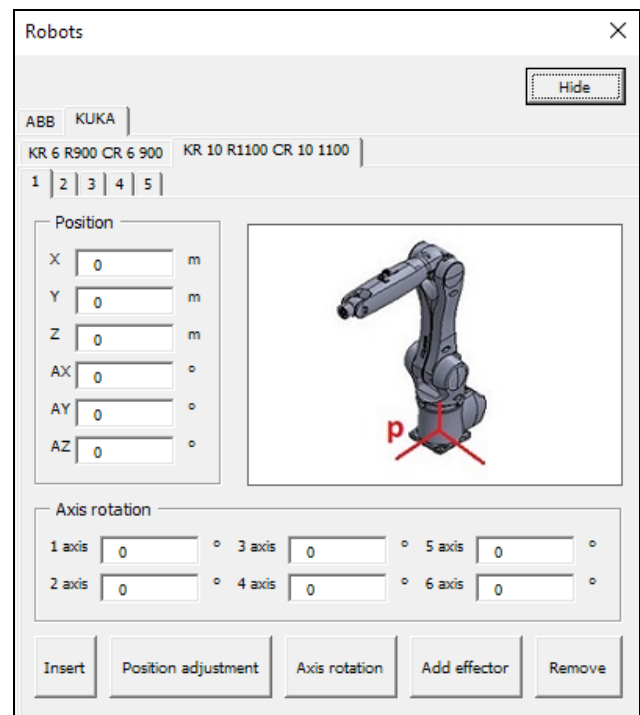


Figure 6. Floating window for robot design

In this case, it is specifically the KR 10 R1100 CR 10 1100 robot from KUKA. The models are always inserted, positioned, or moved relative to the main coordinate system of the assembly. To control the positioning, the user writes values in the "Position" section. A further subdivision of this window can be seen in the upper half of the figure. For better sequence and clarity, this division is indicated in Fig. 7.

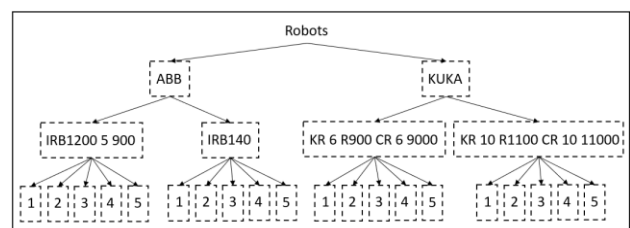


Figure 7. Structure of the floating window tabs demonstrated in the non-floating window for robot design

The user ends up working at the lowest level, the single number level. This ensures the originality of the placed robot. By originality it is meant that the resulting configuration for a robot with attribute one may be different than the same robot model with attribute two or a different number.

For this model, the rotation of its axes can be controlled using the "Axis rotation" section. This feature is its special function. The user can move the robot with the help of the tool, thus checking if the robot reaches where the user needs it to. If the robot does not reach that place, the user can move it or replace it with a robot with a greater reach. Again, this feature for this model came out of the analysis of working with this model, and so this need has been addressed and solved. Almost all models managed by the tool have the same insertion and modification procedure. For this model, however, there is also an "End-effector" button which takes the user to the floating window for designing the end-effector.

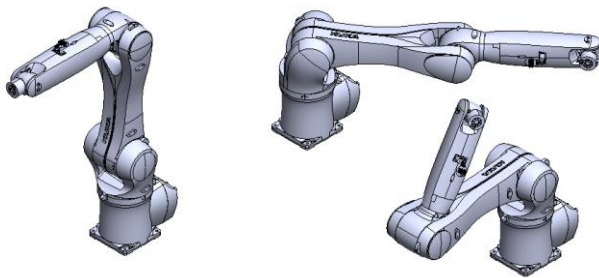


Figure 8. Different rotation of the axis of one robot, which was done using the tool

The conveyor belt design window represents an example of working with a parametric type of model without special functions. Modifications to its dimensions are made in the section "Modification of the conveyor design". The length, width, height and diameter of the drive and driven roller of the conveyor can be modified. The conveyor also includes a base (legs) which can be parametrically adjusted, so that the number of legs per length of the conveyor can be changed at the user's discretion. For example, in cases where the conveyor is subjected to multiple loads, the weight needs to be distributed and therefore more legs and support points are required.

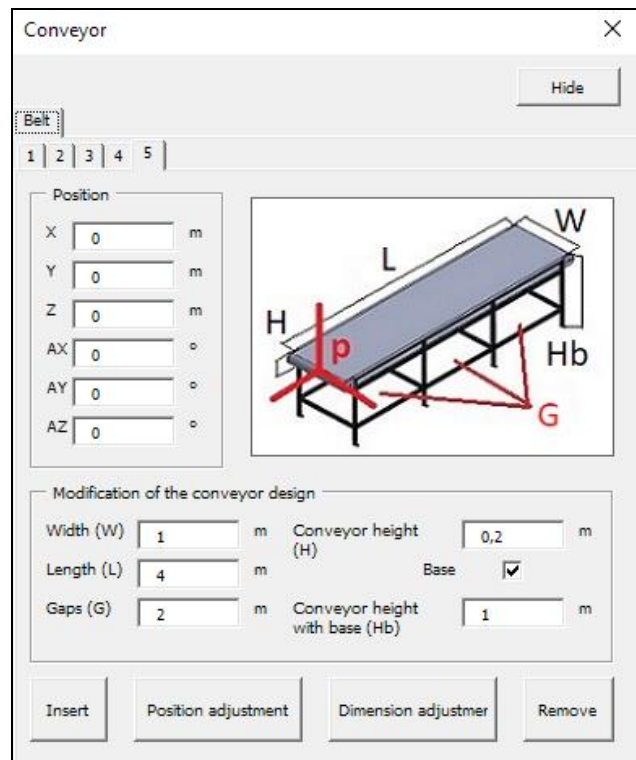


Figure 9. Floating conveyor window

However, there may be cases where the base is not necessary. For this case there is a check box, after unchecking it the base will be suppressed, and the model will be without legs. Fig. 10 shows examples of conveyors created using the software tool.

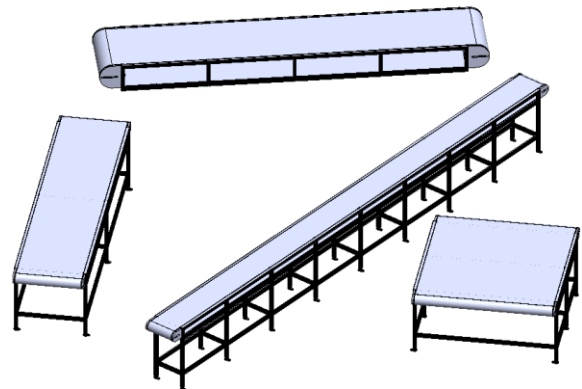


Figure 10. Different conveyors created with the tool

The design of the cell itself is very sophisticated. The menu of the floating window contains its sub-sections in which different parts of the device are designed (Fig. 11).

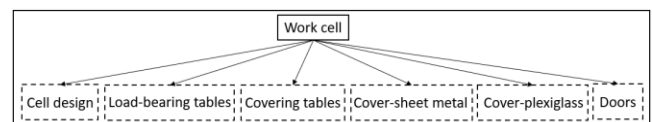


Figure 11. Structure of cell design tabs

In the first part, the cell skeleton is proposed. Here, the width, length, height, rib density of the base and rib density of the cage can be set. In the next tab, the design of the welded workbench takes place. Its dimensions depend on the ribbing of the base. Since the intention is to place this table only in the interior of the cell and to bolt it to the base, the placement of this table in the cell is done by means of a checkerboard

system, in the array. With this method, the user does not have to calculate the table placement; the tool does this calculation itself. The user specifies only the width field and the length field, and the table is placed where the user needs it, or the user can change his decision. If the user changes the cell parameters, the designed tables will also change their dimensions. This is because the dimensions are tied to the design parameters. The same procedure and facilitated type of inserting models through the field is applied in other tabs where additional tables, cell coverings (chassis), visors and service doors are designed. The smart insertion of models through the array makes the process very efficient and fast.

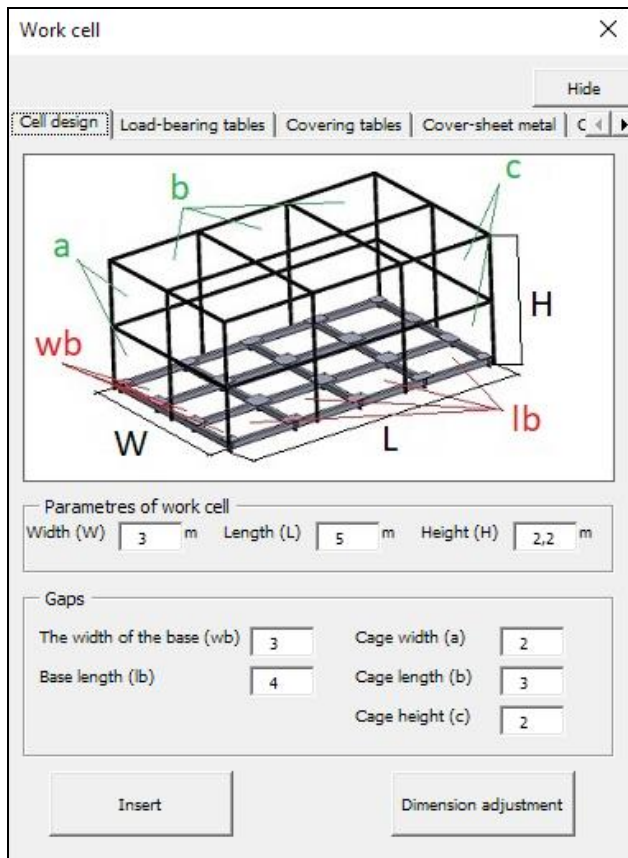


Figure 12. Floating window for cell design

Fig. 13 shows examples of parametric modelling and the use of a special design of this device. This device is taken as an example of a parametric model with a special feature, which is the method of inserting its parts from the cards.

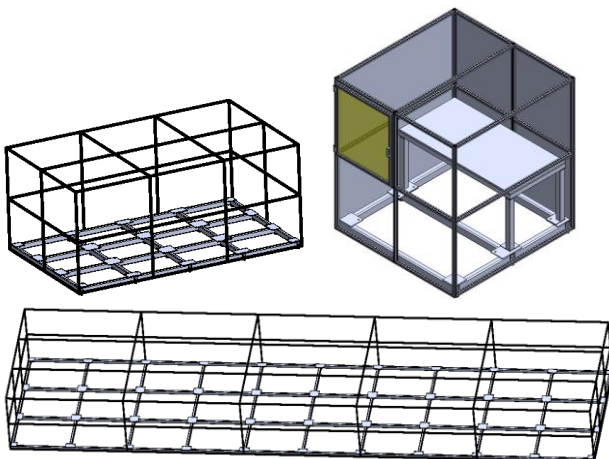


Figure 13. Different types of designed cells that have been designed using the tool

The next challenge was to implement the end-effector design solution. The ideal setup for this tool is that the end-effector which the user chooses and designs can be attached to the robot or device that will perform the desired work with it, and that the tool can be replaced during the production cycle. Thus, the ideal would be to put the tool on a stand next to the device that worked with the tool, if necessary, and to develop another end-effector or process head for the next operation in the production cycle. Another, rather bonus, requirement arises from the situation where one workstation is populated with devices that work with the same end-effector. It would therefore be advantageous if this end-effector, once designed, could be deployed on all the necessary devices without the user having to set it up repeatedly.

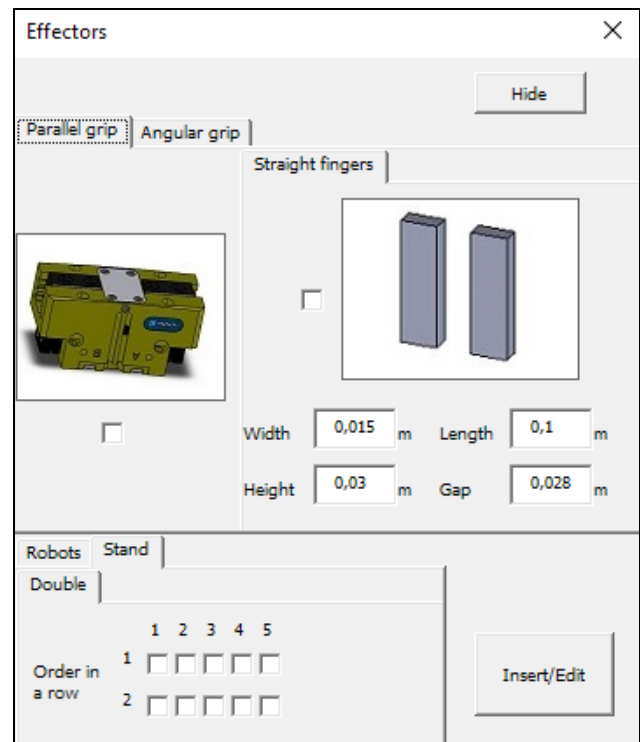


Figure 14. Floating window for end-effector design

We have met this challenge, using an innovative route that is inherently binary in nature. During development, a point was reached where the necessary functionality was missing from the API library. To meet the requirements, the issue was resolved as follows. The robot which the user wants to associate an end-effector with already contains the end-effector, but it is suppressed. The user only activates it and modifies the end-effector's finger parameters. Thus, the whole design procedure works as follows. The user first chooses which type of end-effector he requires in a floating window using the checkbox next to the end-effector, and then the user selects the desired type and dimensions of the fingers. This completes the design of the end-effector and fingers.

The next step is to activate this end-effector at the robot or at the stand that serves as their tray in case of using the end-effector exchange function. For this purpose, there are sections in the lower left corner of the window. Here the user selects the device to which he wants to assign the end-effector and ticks it. The user then presses the "Insert/Edit" button. The end-effector will be activated for the device that was checked in the lower left corner.

If a change is required, the user makes it and presses the "Insert/Edit" button again. The end-effector and finger model can be unchecked by unchecking the boxes in the end-effector

and finger design sections. Then the user selects the model he wants to get rid of his end-effector in the device selection section. Finally, the "Insert/Edit" button must be pressed.

The great advantage of this method is that if an end-effector configured in this way is needed on other devices, the user just checks multiple boxes in the device selection section and the designed end-effector will be assigned to them. Having the carrier device model contain all the components associated with the end-effector may seem a bit redundant, but it meets all the requirements and makes the job much easier and reduces the RW design time.

The proposed software tool works in a similar way for other elements that can be inserted in the RW design.

3 EXAMPLES

As part of the testing of the software, a lot of tests were performed while debugging it. In the final phase, ten defined workstation assignments were built to test the speed of building workstations using the software. These assignments vary in complexity and difficulty in terms of the number of parts and sub-assemblies.

It was decided to compare the process of building the workstations using the software under development with building the workstations manually using databases and manual rebuilding of models. It was decided that in the manual method of building a workplace, the time spent rebuilding the model by the user should be differentiated. This time could be lower if external generators were used. No external generators were used for the workstations that were built.

Three examples of workplaces processed according to the specifications are given in the text below for demonstration. The workstations that were created for this experiment were not designed randomly but were purposefully created to meet the requirements placed on them according to the specified design. The purpose of the workstations and the requirements placed on the workstations are discussed in the paragraphs preceding the workstation figures.

These are followed by Table 1 showing the results of the measurements and descriptions of the data in the table. A graph is also presented in which the data from the table is processed for better visualisation.

At this workplace (Fig. 15), the marking of object of manipulation (OM) occurs. The component arrives at the interior of the cell using a conveyor belt. Here it is picked up by a robot and placed under the marking device. The marked OM is then placed by the robot on a second conveyor belt and the OM leaves the workplace.

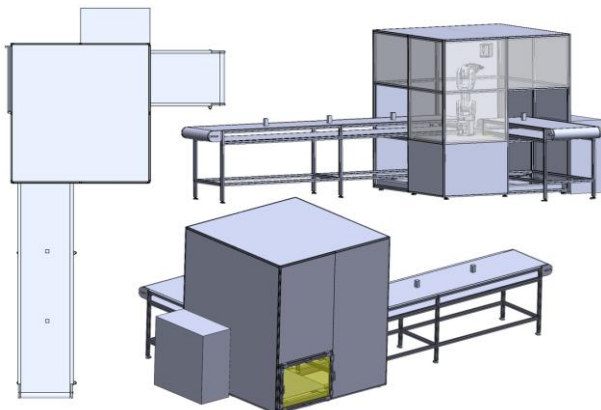


Figure 15. First workplace that was made for test by tool

Another workplace (Fig. 16) is used for assembly and marking. Two components (cylinder and block) enter the workstation via two conveyor belts (low). Here they are removed by a robot which assembles them together on a positioner. The positioner moves up so that the second robot can take the assembled component and place it under the marking device. The OM, assembled and marked, then leaves the workstation along the third (not higher) conveyor belt.

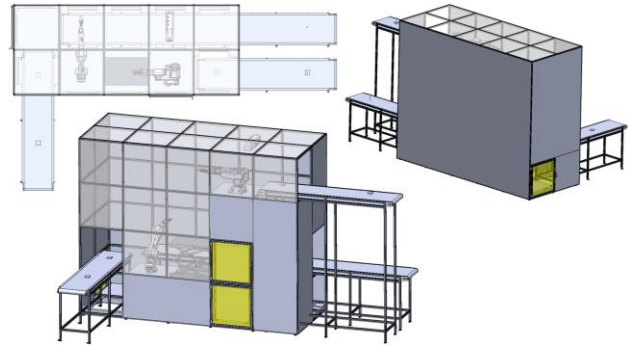


Figure 16. Second workplace that was made for test by tool

This workplace is used for product assembling and marking. The eight conveyors labelled A to H (upper part of Fig. 17) are used to bring the parts into the workplace. Robot "1" assembles parts A and B together and passes them to the table next to it. Robot "2" assembles C and D. It then further assembles AB and CD and passes them to the table between it and robot "5". Robot "3" assembles components E and F together. Next, it places the component thus assembled on the table next to robot "4". Robot "4" first assembles components G and H together, and then attaches component EF to them. It places the element thus assembled (EFGH) into the prepared pallet next to robot "5" which assembles the prepared subassemblies together to form the product marked ABCDEFGH and places it under the marking device where the product is marked and dispatched by a conveyor out of the RW.

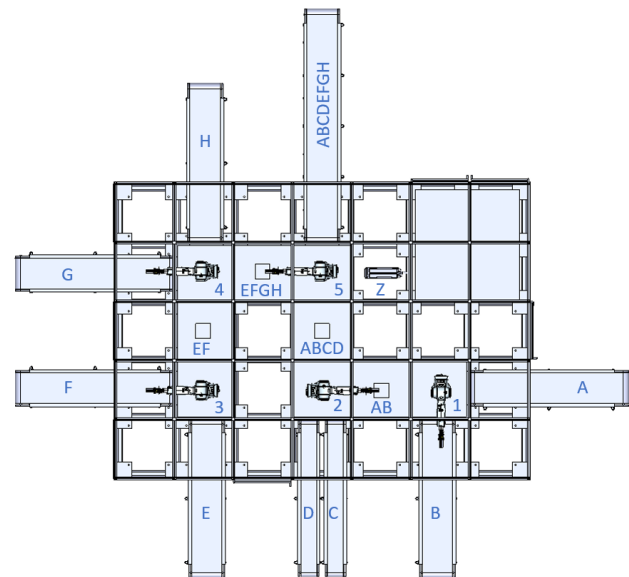


Figure 17. Third workplace that was made for test by tool – upper view

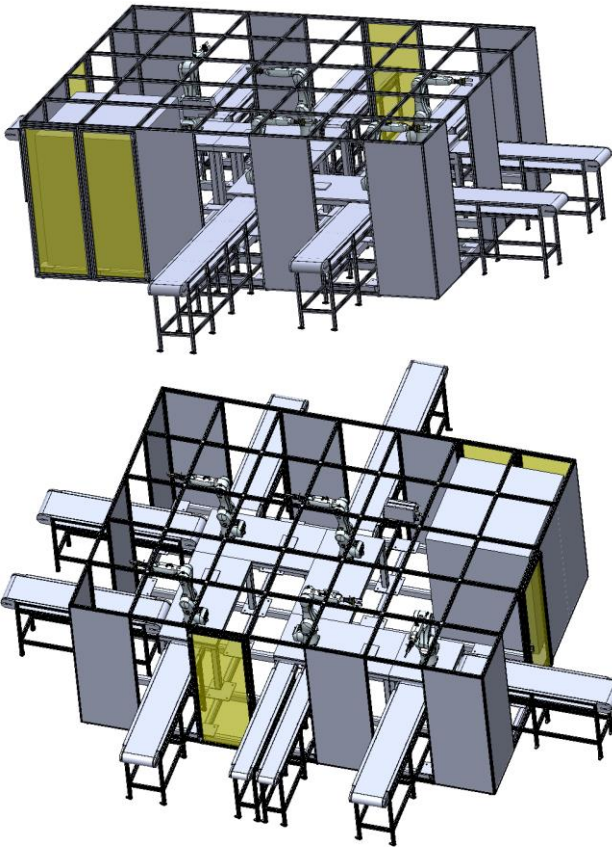


Figure 18. Third workplace that was made for test by tool

During the development of the ten workstations, the mentioned data was measured and is tabulated below. The first row of Table 1 is used to label each workstation. These are ordered according to the complexity of the parts inserted in the assembly. The number of models from the database that the workplace contains is shown in row NM (number of models). The data shown in row T1 is the total time taken to build the workplace using the software tool.

Workplace	1	2	3	4	5	6	7	8	9	10	
NM	24	28	34	38	41	41	46	47	59	64	[pieces]
T1	17	21	25	28	35	36	42	47	51	65	[min]
T2	19	32	35	49	62	64	70	83	101	125	[min]
T3	28	42	48	73	102	113	134	169	185	213	[min]

Table 1. Data from the tests

For a better overview, the time data are plotted in a graph (Figure 19). The horizontal axis of the graph shows the workplace number from the table. The vertical axis is the time in minutes. The graph then shows three curves. All data in the graph is from Table 1. The T1 data is shown in blue. T2 data are shown in orange. T3 data are shown in grey. NM data (number of models) are not shown in the graph. These data are only shown in Table 1 for ordering and an idea of how many models are in the workplace.

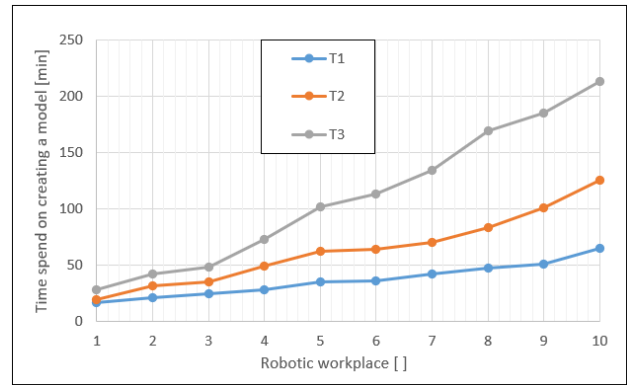


Figure 19. Graph recording the progress of the tests as a function of time

4 CONCLUSIONS

This way to accelerate the design of robotic work cells is very efficient and currently almost unattainable. With this tool, a system integrator can create multiple robotic workstation designs in less time with much more precision and detail.

Among other things, the user does not have to go through databases and search for models. With the robotic cell design software tool, all the database is accessible via a single window directly in the CAD software. Using a logical and user-friendly environment, these models can be selected with absolute precision. There is no need for manual editing of the models from the database, they can be edited using the user interface, which has been shown to reduce time considerably. If the user makes a mistake, it can be corrected in a few clicks. The software has been designed so that its user only needs a basic knowledge of the SolidWorks CAD system to operate it.

The tool includes a comprehensive design of the entire workstation with the ability to turn the tool off, make manual adjustments and turn it back on. All development is done in SolidWorks CAD software, so the user doesn't even need to open any catalogue or database. Everything is in one tool in one CAD software.

However, there is currently no tool that incorporates design and parametric design as extensively as this work.

The tool can also be further developed and can be enriched with any model required if needed. Even the models that are already there can be further developed, for example the parametric model of a conveyor can be further modified and extended to include the capabilities of sensor systems that will be placed in the appropriate location with the appropriate parameters upon user request. It is therefore only a matter of setting additional requirements and constraints.

ACKNOWLEDGMENTS

This article has been elaborated under support of the project Research Centre of Advanced Mechatronic Systems, reg. no. CZ.02.1.01/0.0/0.0/16_019/0000867 in the frame of the Operational Program 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.

REFERENCES

- [Jayakiran 2018] Jayakiran, R. E. and Venkatachalapathi, N. and Pandu, R. V. Development of an approach for Knowledge-Based System for CAD modelling. Materials Today: Proceedings, January 2018, Vol.5, No.5, Part 2, pp 13375-13382. ISSN 2214-7853
- [Jingzh 2020] Jingzh, Z. and Tingzhou, N. and Yanan, G. APPLICATION OF PARAMETRIC MODELING TECHNOLOGY IN SINGLE SCREW EXTRUDER. International Journal of Mechatronics and Applied Mechanics, November 2020, Vol.1, No.8, pp 250-255. ISSN 25594397, 25596497
- [Esanakula 2021] Esanakula, J. and Theja, M. DEVELOPING A BASIC CAD MODEL OF A SPUR GEAR USING AN AUTOMATED APPROACH. International Journal on "Technical and Physical Problems of Engineering" (IJTPE), September 2021, Vol.13, No.3, pp 20-24. ISSN 2077-3528
- [Boleslavsky 2021] Boleslavsky, A. Automated Reconfiguration of a 3D Model of a Robotic Workplace. BOL0039_FS_N2301_2301T013_2021. Ostrava: Vysoká škola báňská – Technická univerzita Ostrava, 2021.
- [Zeman 2021] Zeman, Z. and Mihola, M. and Suder, J. DESIGN OF ALGORITHMS FOR AUTOMATIC SELECTION OF DRIVE UNITS FOR MECHATRONIC DEVICES. MM Science Journal, June 2021, No.2, pp 4362-4370. ISSN 18031269, 18050476
- [Mihola 2020] Mihola, M. and Zeman, Z. and Fojtik, D. Automation of the design of the cross-section of the manipulator arms profile. IAES International Journal of Robotics and Automation, September 2021, Vol.10, No.3, pp 170-181. ISSN 2722-2586
- [Zhang 2019] Zhang, C. Research on 3D Variable Design of Corrugated Box Based on SolidWorks. 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), Dublin, October, 2019. New Jersey: IEEE Xplore, pp 588-591, ISBN: 978-1-7281-4691-1
- [KEYENCE 2022] KEYENCE. Products. Sensors and Machine Vision Systems for Factory Automation [online]. Copyright © 2022 [cit. 2022-02-14]. Available from https://www.keyence.co.uk/products/?ad_local=sit_etopvallbn1
- [FESTO 2022] FESTO [online]. Copyright © 2022 [cit. 2022-02-14]. Available from <https://www.festo.com/gb/en/>
- [PARTcommunity 2022] Partcommunity. 3D & 2D CAD Models [online]. Copyright © 2022 [cit. 2022-02-14]. Available from <https://b2b.partcommunity.com/community/>
- [TraceParts 2022] TraceParts. Free 3D models, CAD files and 2D drawings [online]. Copyright © TraceParts S.A.S. 1990 [cit. 2022-02-14]. Available from <https://www.traceparts.com/en>
- [GrabCAD 2022] GrabCAD. GrabCAD Makes Additive Manufacturing at Scale Possible [online]. Copyright © 2022 [cit. 2022-02-14]. Available from <https://grabcad.com/>
- [HIWIN 2022] HIWIN. 3D konfigurator. Ball screws, Linear technology, Linear guides, Linear motors [online]. Copyright © 2022 [cit. 2022-02-14]. Available from <https://www.hiwin.cz/en/3d-konfigurator>
- [igus 2022] igus. Online Tools and Configurators. Cable carriers, polymer bearings, flexible cable, & linear slides [online]. Copyright © 2022 [cit. 2022-02-14]. Available from https://www.igus.com/info/online-tools#section_6
- [INTERROLL 2022] INTERROLL. Belt Drive Matchmaker. Leading Global Provider of Material Handling Solution [online]. Copyright © 2022 [cit. 2022-02-14]. Available from <https://www.interroll.com/products-solutions/belt-drive-matchmaker/#/conveyor/type>

CONTACTS:

Ing. Adam Boleslavsky
VSB - Technical University of Ostrava, Department of Robotics
17. listopadu 2172/15, Ostrava-Poruba, 708 00, Czech Republic
+420 596 999 363, adam.boleslavsky@vsb.cz,
<https://www.fs.vsb.cz/354/en>

doc. Ing. Milan Mihola, Ph.D.
VSB - Technical University of Ostrava, Department of Robotics
17. listopadu 2172/15, Ostrava-Poruba, 708 00, Czech Republic
+420 596 995 445, milan.mihola@vsb.cz,
<https://www.fs.vsb.cz/354/en>

Ing. Rostislav Wierbica
VSB - Technical University of Ostrava, Department of Robotics
17. listopadu 2172/15, Ostrava-Poruba, 708 00, Czech Republic
+420 596 999 363, rostislav.wierbica@vsb.cz,
<https://www.fs.vsb.cz/354/en>

Ing. Jan Bem
VSB - Technical University of Ostrava, Department of Robotics
17. listopadu 2172/15, Ostrava-Poruba, 708 00, Czech Republic
+420 596 999 362, jan.bem@vsb.cz,
<https://www.fs.vsb.cz/354/en>

Ing. Tomas Spurny
VSB - Technical University of Ostrava, Department of Robotics
17. listopadu 2172/15, Ostrava-Poruba, 708 00, Czech Republic
+420 596 999 443, tomas.spurny@vsb.cz,
<https://www.fs.vsb.cz/354/en>