Вісник Тернопільського національного технічного університету
https://doi.org/10.33108/visnyk_tntu
Scientific Journal of the Ternopil National Technical University
2024, № 4 (116) https://doi.org/10.33108/visnyk_tntu2024.04
ISSN 2522-4433. Web: ***visnyk.tntu.edu.ua***

UDC 621.865.82

FEATURES OF SOFTWARE IMPLEMENTATION AND THREE-DIMENSIONAL MODELLING OF THE AUTOMATED CERTIFICATION OF INDUSTRIAL ROBOT METRICS IN COPPELIASIM ENVIRONMENT

Oleksandr Dobrzhanskyi[1]; Valerii Kyrylovych[1]; Anton Kravchuk[1]; Eugene Puhovsky[1]; Volodymyr Savkiv[2]

[1]«Zhytomyr Polytechnic» State University, Zhytomyr, Ukraine
[2]Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine

Abstract. *The paper is devoted to the problem of automated certification of industrial robot metrics (IRM) in the CoppeliaSim software environment. The authors in detail consider the capabilities and tools of the CoppeliaSim software environment for measuring and evaluating such spatial parameters of the IR as the working area and the configuration of the tool's geometric position. The study of mechanical, technical and technological systems and virtual environments has long been the subject of scientific activity. For example, the papers describe mathematical models of the behaviour of flexible mechanical transmissions, the research of which is carried out without taking into account automated implementation. There are also known studies by the method of computer modelling of technical systems for the power load of cutting tools using highly specialized software environments. Moreover, the features of the software implementation of the proposed models are usually not considered. The material proposed in this paper focuses on universal software products and programming languages for three-dimensional modelling. The results of the investigation of methods and approaches to the automated implementation of the metric, which make it possible to ensure reliability and accuracy in the further synthesis of elements of robotic technologies, such as optimisation of equipment placement, formation of the optimal trajectory of movement of links of the manipulation system of the IR with a tool or a gripper are presented in this paper. Rationale for the use of spatial 3D modelling with full-size virtual models of the IR in the CoppeliaSim environment are presented in the paper. The authors analyse the tools and instruments that allow taking into account the influence of primarily spatial factors on the metric of the IR, such as geometric parameters of the IR design, tools, gripper, possible limitations due to the design and technological features of the technological equipment. This paper can be useful for researchers, engineers and students studying IRs in terms of their automated modelling and analysis.*

Key words: *modelling, industrial robot, robotic technology, automated metric certification.*

https://doi.org/10.33108/visnyk_tntu2024.04.135 *Received 25.11.2024*

1. INTRODUCTION

The study of many mechanical (various mechanical transmissions, industrial robots, etc.), technical and technological systems (technological system of MDTP – machine, device, tool, part, etc.) and virtual environments in their various implementations is not new.

For example, in [1], a mathematical model of the behavior of a flexible transmission is compiled and studied at the level of analytical analysis, which the authors consider as a discrete-flexible connection. This mechanism, which can be represented in three-dimensional (3D) space, is considered as a component of the drives of a wide range of technological machines. The main task in this case is the task of studying changes in its kinematic characteristics during operation. At the same time, the automated implementation of the compiled mathematical model is not considered, and therefore the specifics of its software implementation are not indicated.

Corresponding author: Valerii Kyrylovych; e-mail: kiril_va@yahoo.com

In [2], with the method of computer modeling the technical system of power load of knives of end mills of a certain design when processing by cutting certain materials was investigated. The research is carried out using a specialized software environment DEFORM-3D. The latter does not imply originality with an indication of the features of 3D modeling, since it involves data entry and modeling according to formats and procedures characteristic of this.

In general, in the above as an example, but in a far from complete list of available information sources, studies of these and similar systems and environments are mostly carried out full-scale research and/or mathematical models of research objects. At the same time, the peculiarities of the software implementation of the proposed methods, approaches and models using such software products, which are not highly specialized, but are more universal with built-in certain capabilities for the use of known programming languages in terms of displaying the features of programming and three-dimensional modeling, are not indicated. The material presented below is focused on the use of such software environments and programming languages.

An effective modern toolkit for automated certification of industrial robot (IR) metrics in the design of the modern production environment at present involves the use of **virtualisation technologies** [3, 4, 5, 6]. Such approach makes it possible to visualise the obtained results and modelling process itself, allows the **multivariability of the initial data**, and to perform the certification process in the **automated manner**, and at present it is a common requirement when choosing design/synthesis tools for robotic technologies.

The process of certification of the IR metric could be performed in a physical form – physical modelling, which involves measuring and controlling certain performance parameters, including movement accuracy, position change, trajectory reproducibility, etc. In this case, it would be necessary to use a significant array of different sensors and equipment. These can be encoders, accelerometers, force and position sensors, etc. [7, 8]. At the same time, this array of sensor devices would require the use of such resource-intensive procedures as setup and calibration. Physical implementation would make it possible to collect real data, but direct processing and analysis in any case would involve the use of automated computer processing. Implementation of multivariate input data in physical modelling would generally require a special approach, for example, the production of full-size or scalable physical models of technological equipment when solving the problem of metric certification.

The relevance of improving the process of certification of the IR metric is also explained by the value of the results obtained for the following stages of analysis, which may include the identification of anomalies, comparison with standards, statistical analysis, etc. The subsystem of automatic certification of the IR metric can be integrated with the IR management system. The certification data can be used to ensure effective management of IR directly during the performance of technological operations [9]. This may include the automatic adjustment of the IR parameters based on the obtained data. The automated attestation of IR metric can also address the **safety and reliability** of the overall robotic process system. For example, if metric indicates the abnormality, it may indicate possible malfunction of the IR that could result in the emergency.

Thus, at present there is virtually no alternative to the need to develop systems for automated certification of IR metrics based on **virtualisation and computer calculating technologies**.

Studies [3, 4, 5, 6] provide many examples of experiments conducted in virtual space, which result in obtaining data on the parameters of the IR, which can be accepted as certification and form a certain metric of the IR.

The importance of forming metric taking into account the spatial parameters of the accuracy of possible movements of the IR, which is used as an alternative to standard technologies for machining the parts using a set of machine tools is shown in investigation [9]. The considered metric can be supplemented with elements that take into account the IR operating.

ISO standards for the possibility of using them to form a positioning accuracy metric when using IR are considered in studies [10, 11]. To perform the certification, it is proposed to obtain all data during physical experiment. At the same time, the issue of virtualisation of the experiment is ignored. The importance of the spatial parameters of the IR operating areas is not mentioned as well.

Analysis of materials demonstrating the results of experiments carried out by means specialised virtual environments shows that the RoboDK modelling environment for robotic complexes is quite common [13, 14, 15, 16, 17, 18]. It should be noted that RoboDK is rather multitasking environment with API (application programming interface). This makes it possible to organise data exchange between RoboDK and applications written using the most common programming languages [17]. RoboDK can also be integrated with such common software product for mathematical analysis as MATLAB [14].

The idea of using RoboDK software environment for direct performance of automated metric certification is substantiated in [18] and mandatory stages of this process are presented: 1) creation and preparation of 3D model of the analysed IR (for example, using another software environment Solidworks); 2) loading the 3D model of the IR into the RoboDK software environment (which requires element-by-element loading of individual links of the IR, additional scaling procedures, fixing connections, etc; 4) selection of the end element of the gripper (which can be the end point of the gripper, the end point of the gripper with the workpiece, or the end point of the tool on the end flange of the gripper); 5) direct execution of the certification procedure; 6) visualisation and generalisation of the results, development of recommendations for the practical application of the gripper.

Investigations [18, 19] demonstrate certain limitation in terms of the freedom of application of the complex analysis of the spatial elements of the IR, especially when there is the need to develop arbitrary algorithm for the IR certification. It should also be noted that RoboDK is a commercial product with certain licensed terms of use. Thus, for wider application, it is required to search for open-source software products.

CoppeliaSim [20, 21] is a software product that removes certain restrictions on licensed use for research purposes and is a complete environment for computer modelling of robotic operations. The CoppeliaSim software environment, the latest version of which is 4.5.1 – rev.1.0, created and maintained by Coppelia Robotics AG 2006–2024 (www.coppeliarobotics.com), according to the documentation [20], is actually intended for virtualisation and modelling of robotic devices, development of control algorithms, prototyping, verification and simulation of the functioning of various devices based on digital twins (models).

The objective of the paper is to present the results of the investigations concerning the possibility of using the tools of the CoppeliaSim software environment to build the software system for automated certification of the IR metric, which is designed to determine and evaluate the spatial parameters of the IR operation, namely: the working area of the analysed IR, taking into account the configuration of the geometric position of its clamping device and/or tool.

2. METHODOLOGY

The CoppeliaSim virtual model editor system has a built-in Lua language interpreter that can be used to create scripts for controlling the links of the analysed IR model, as well as scripts for collecting and recording simulation results. The presence of the built-in Lua interpreter makes it possible always to make operational changes to the script and run it without previous compiling it into machine code. This approach makes it possible to use the script as a research element and even make changes to the initial data in text – without using an additional graphical user interface (GUI), although the possibility of creating the latter one is also included in CoppeliaSim. The Lua language is one of the most well-known and developed high-level programming languages at the moment, and in terms of its prevalence and speed, it can be compared to such programming languages as C++, C#, and Java.

No less important is the role of software algorithms embedded in the CoppeliaSim virtual model editor itself. These algorithms are able to reproduce the mutual movement of virtualised mechanical elements – the links of the gearbox, taking into account the fact that they are connected by movable joints. At the same time, full correspondence between the movement of the real gear and its virtual model is maintained.

Thus, taking into account specialisation, multitasking, and compliance with the abovementioned criteria for specialised virtual environments, the CoppeliaSim software environment was chosen for further research in the area of creating algorithms for automated performance of IR metric certification operations.

The **actual task** while carrying out research on the specified topic was to conduct the experiment, the stages of which can be presented in the following way:

– downloading and placing the IR model for research in the CoppeliaSim virtual environment;

– creation of the tool for specifying initial data when conducting the experiment concerning the construction of the operating area taking into account additional conditions, for example, fixed angular position of the gripper (tool) of the IR;

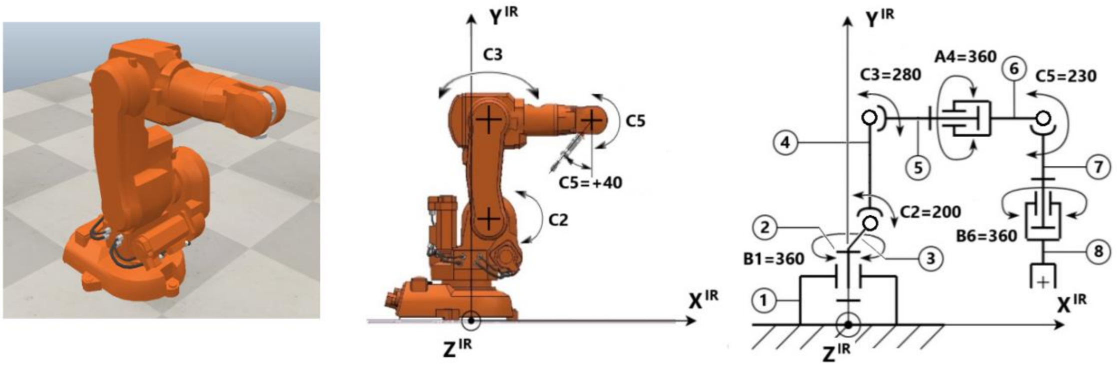
– compiling the program that ensures the execution of the modelling process algorithm, for example, moving the IR links in a certain order, registering angular positions and extreme points, avoiding collisions between moving parts and collisions with foreign surfaces, etc.,

– converting the operating point cloud into 3D surface of the IR operating area and saving this surface in a generally accepted format used for three-dimensional figures, for example, the stl format;

– direct performance of the experiment, i.e. execution of the software algorithm for the hosted virtual IR model and display of the experiment results.

3. RESULTS AND DISCUSSION

At the first stage **the downloading and placement of the analysed IR model** were performed. It is quite possible to develop our own IR model, as described in [18]. To avoid developing of the virtual IR model, we used the digital twin of the ABB IRB 140 robot IR - CAD model of the IR (Fig. 1), specially developed by ABB Robotics [22] for the application in the CoppeliaSim environment and similar ones. As for the characteristics of the ABB IRB 140 IR model, they are given by the manufacturer as follows: positioning accuracy: 0.03mm; number of axes: 6; maximum reachable point by axis link 5: 0.810 m; load capacity: 6 kg; additional load on the lifting links: 1 kg; additional load on the wrist link: 0.5 kg. The virtual ABB IRB 140 model has coordinates, the dimensions of which are given in meters and completely correspond to the dimensions of the real IR. This is also true for the coordinates of the CoppeliaSim environment itself.



a b c

Figure 1. Virtual model of the ABB IRB 140 IR: general view (a), side projection (b) and coordinate system taking into account the attached gripper (tool), frame (thread) model (c) of the kinematic structure of the IR

This virtual model is supplied without specific gripping mechanism. Therefore, the digital model of the ABB IRB 140 robot was supplemented by the authors with the virtual model of RG 2 robotic gripper (Fig. 2), provided by the On Robot manufacturer [23].

It was verified that the further application of the gripper model in the experiment on the construction of the operating area taking into account collisions reveals the model disadvantage: it is possible to limit the real angles of rotation of the IR links, when moving them, by means of the algorithm, but it is impossible to control the collisions of the gripper surface with other IR elements or other objects, since the gripper consists of separate links that already enter into collisions as a result of its assembly. Therefore, the CoppeliaSim collision system will always be activated. It is proposed to overcome this by surrounding the gripper with control planes close to it (Fig. 2, a).

If the IR grip contains the part which defines the outer boundaries of the grip by its dimensions, then it is possible not to use the planes surrounding the grip. The source of collisions will be the part itself, which should not collide with the grip, but should collide with all other elements of the model. This can be ensured by removing the collision control setting for the overlays at the ends of the gripper fingers.

We placed the ABB IRB 140 IR, supplemented by RG 2 gripper, in the center of the virtual scene so that the centre of the coordinate system of the first rotary joint B1 coincided with the global centre of the scene coordinate system.

It is done in such a way that the coordinates of the operating areas, obtained and further processed according to the methodology, can be re-placed on the scene in the same coordinate space for the purpose of additional visual verification and as the final result of the modelling.

a b

Figure 2. Virtual model of the RG 2 IR gripper: general view (a) taking into account the array of collision control planes; general view (b) taking into account the location of the workpiece in the gripper

The next step is the **initial setup of the IR model**. The methodology involves modelling for a certain, predetermined, angular placement of the gripper in the plane of rotation of the main «elbow» rotary articulated joints of the IR. The gripper is rotated by the penultimate rotary joint of the IR C5 (see Fig. 1, b, c), which in practice is intended for this purpose.

The method also involves the initial setting by the user and the software support of the stability of the angular position of the grip relative to the global coordinate system throughout the entire modelling process. For this purpose, the separate element was developed on the virtual scene – the angular position setter (Fig. 3, a).

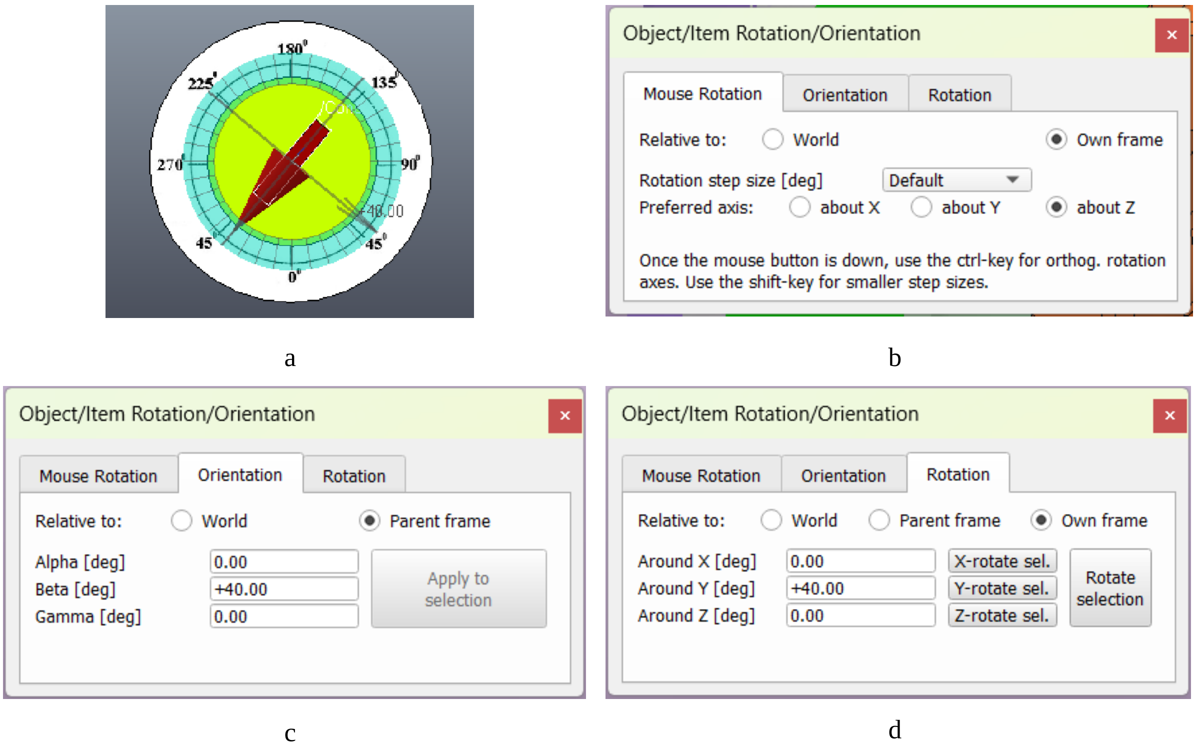


Figure 3. The angular position setter of the IR: direct virtual object-setter (a), the parameters of the setter (b), which enables the mode of setting the angle by dragging the mouse pointer; the parameters of the numerical mode of setting by Euler angles (c); the parameters of the numerical mode of setting by angles around the coordinate axes (d)

This provides a certain convenience for both the user and the further functioning of the model. Due to this, it is possible to avoid unnecessary routine interaction of the user with the virtual space of the model when specifying the angular position of the gripper, as well as direct interaction of the user with the elements of the IR design with the gripper and, as a result, erroneous displacements of other elements of the model. Therefore, a separate setter plays the role of the element of the interface part.

The user interacts with the setter by moving its arrow to a certain angle on a circular scale using dragging movement of the mouse pointer. However, the minimum discreteness of such setter is 1 degree (Fig. 3, b).

The user can also rotate the pointer on the circular scale using special window for specifying the angular positions of virtual elements, which is provided in the CoppeliaSim environment itself. In this case, the minimum specification resolution is 0.001 degrees (Fig. 3, c).

Before starting the simulation, the number of discrete steps is also set when reproducing fixed angular positions in the range from 0 to 180 degrees for each rotary joint of the IR separately. This is done with one line directly in the script: steps={0,90,90,0,90,0}. Some positions are set as 0, since they relate to rotary joints that provide rotations of the IR links in planes other than the plane of rotation of the main rotary joints of the IR: B1, A4, B6 (Fig. 1, d).

If the operation area of the IR is symmetrical, then it is enough to obtain a half-space of 0–180 degrees of the operating area. In this case, the cross-section of the operating area will be clearly visualized on the virtual stage to assess its shape.

When the preparation stage is completed, the modelling process is launched. Such modelling is carried out under the control of the program script. Writing the program script in the Lua language. The tools built into CoppeliaSim for specifying the angular position of the connections of the IR links operate in such a way that when the angular displacement of the certain IR link around the axis of its connection relative to its initial position relative to the previous link is specified (the direction from the IR base is taken into account), then in the coordinate system of this link, all subsequent links connected to it are shifted by the same angle together with it. In this case, the integrity of the model is preserved and the connections themselves are not violated.

Regarding the role of the Lua script directly in the considered virtual model of IR with grip, it is worth noting that its tasks are to control the main stages of the modelling experiment, which involve its execution in automatic mode.

The defining fragments of the developed script for each of the stages are briefly presented below.

1) Script modelling of all possible combinations of rotation angles of the IR links. At the initial stage, it is necessary to obtain the limiting angles of mutual angular displacement of the links in the rotary joints.

For the case of ABB IRB 140 virtual IR model, they are already set by the manufacturer in the properties of virtual objects (IR links) in a special window form for each of the IR links (Fig. 4).

Figure 4. Window form for specifying limit angles for the selected rotary joint in the CoppeliaSim environment

In the case of the own IR model, the angle limits can be pre-set either by means of the window form (Fig. 4) or programmatically in a Lua script:

jointInit[i]=-180*math.pi/180 -- setting the initial angle;

jointRange[i]=360*math.pi/180 -- setting angular displacement limits relative to the initial position.

Then it is possible to perform sequential setting in automatic mode of displacements relative to the initial angle and within the angular displacement for each rotary joint. The simplest version of the software implementation of such task is nested cycles:

for i1=1,steps[1],1 do -- selection of the current step for the first link (B1);

local p1=jointInit[1]+i1*jointRange[1]/steps[1] -- calculation of the current angular displacement;

sim.setJointPosition(jointHandles[1],p1) -- setting the current angular displacement;

for i2=1,steps[2],1 do -- selection of the current step for the second link (C2);

local p2=jointInit[2]+i2*jointRange[2]/steps[2] -- calculation of the current angular displacement;

sim.setJointPosition(jointHandles[2],p2) -- setting the current angular displacement.

Similarly for other rotary joints.

As an alternative to nested loops, it is possible to consider the option of creating the recursive function. This approach will be especially convenient for arbitrary number of links and connections of the IR manipulation system.

2) Holding the fixed angular position of the gripper (tool) relative to the coordinate system of the analyzed IR using script commands. When specifying combinations of the search for angular positions of individual IR links, the script simultaneously monitors the constancy of the

angular position of the gripper, fixed on the tool flange of the last link of the IR manipulation system. Control is performed in the IR coordinate system, i.e., relative to external objects, the angle (orientation) of the gripper will remain constant regardless of the position of other IR links:

irb140 = sim.getObjectHandle('IRB140') -- getting the pointer to IR object;

tool = sim.getObjectHandle('IRB140_link6') -- getting a pointer to the rotary grip connection corresponding to C5;

set = sim.getObjectHandle('SetBlock') -- obtaining the pointer on the angular position setter;

matr = sim.getObjectMatrix(set,irb140) -- obtaining the matrix of the setter angles in the IR coordinate system;

orient = sim.getEulerAnglesFromMatrix(matr) -- obtaining Euler angles from the setter matrix;

setObjectOrientation(tool, irb140, orient) -- setting of gripping angles by the angles of the setter.

3) While specifying combinations of searching for angular positions of individual elements, the script simultaneously avoids exceeding the limited angles of rotation of the links, controls the collision between the links and the gripper (tool) or obstacles (in our case also with the plane of installation of the IR):

coll_check = sim.getCollisionHandle('CollisionBlock') -- getting a pointer to a collision block;
coll=sim.readCollision(coll_check) -- checking for possible collision;

```
if ( coll == 1 ) then -- in case of collision, move to the next angular position.
counter=counter+1
end
```

4) The cloud (set) of points of the collision-free reachable area is written to the file for further targeted processing and analysis:

file = io.open('FileName.xyz','w') -- opening a file for writing;

io.output(file) -- binding the data output stream to the open file;

end_effector = sim.getObjectHandle('end_effector') -- obtaining the pointer to the spatial point of the final element of the IR;

position=sim.getObjectPosition(end_effector,-1) -- obtaining the coordinate vector of the final element of the IR;

io.write(table.concat(position,"\t")) -- writing to the file of the current collision-free position of the final element of the IR;

io.write("\n") -- line selection for the next record

At the same time, the script forms the cloud of points of the operating area of the IR in the form of points of reach of the center of capture around the IR, taking into account the absence of mechanical collisions and the constancy of the angular position of the IR grip:

matrix=sim.getObjectMatrix(end_effector,sim.handle) -- obtaining the position matrix of the final element of the IR;

points[#points+1] = matrix[4] -- isolation of X coordinate;
points[#points+1] = matrix[8] -- isolation of Y coordinate;
points[#points+1] = matrix[12] -- isolation of Z coordinate;

5) And based on the obtained point cloud, 3D figure of the IR operating area is formed:

vertices, indices = sim.getQHull(points) -- formation of vertices and indices of 3D figure;

work_zone=sim.createShape(3,0,vertices,indices) -- obtaining a virtual image of 3D figure of the operating area.

6) When the 3D figure of the workspace is formed, it is possible to create the standard .stl file, i.e. export the virtual spatial model of the workspace to the file of the standardized form for further processing in other applications or for storage:

filePath = 'path/file_name.stl' -- preparing the STL file path and name;

sim.exportShape(work_zone, filePath) -- model export to the STL file.

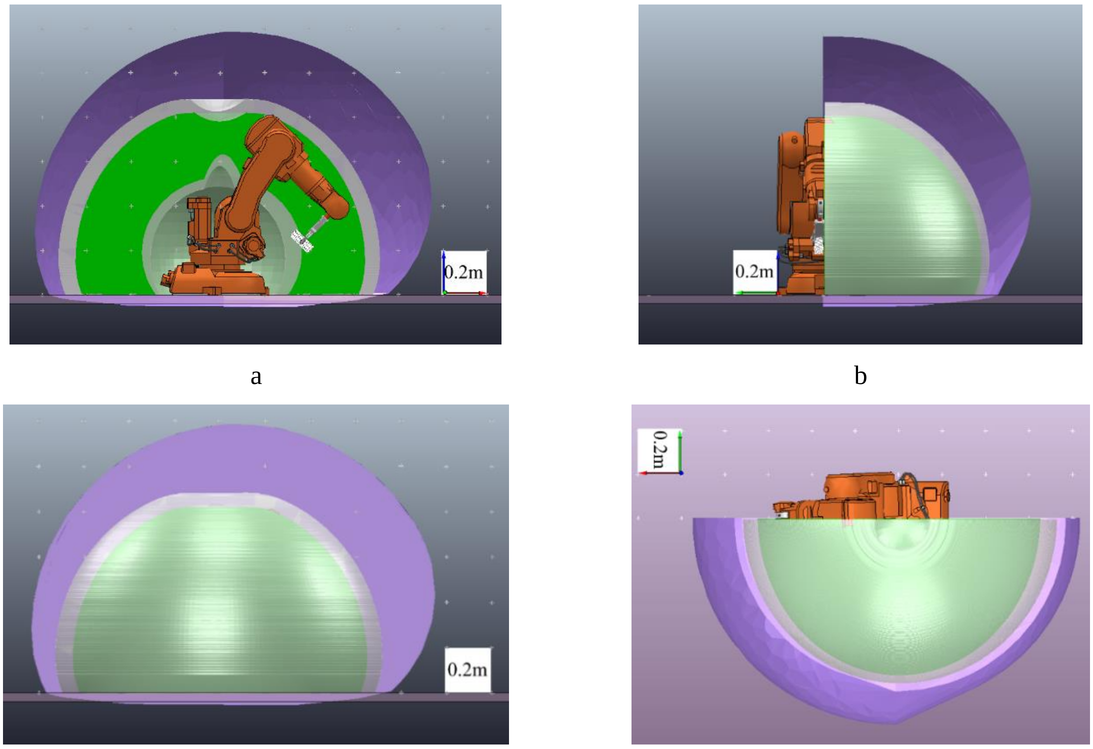
Direct execution of modelling. The modelling is started by the main script launch button.

At the final stage of modelling, 3D figures of the operating areas of the IR grip reach, obtained after processing the cloud of operating points by algorithm commands and presented in the form of .stl files, are loaded onto the virtual scene and combined with the 3D IR model. At the same time, the actual dimensions, coordinates, and centers of the coordinate systems of the present and added elements are maintained (Fig. 5).

To enable spatial verification, the horizontal and vertical grid of marks is applied to the model space scene. The horizontal and vertical mark guides are clearly distinguishable in the display (Fig. 6, a). The grid makes it possible to estimate the actual horizontal and vertical dimensions of the 3D IR models and the resulting operating areas.

The dimensional cube is also placed on the space scene (Fig. 6, b). It shows the current distance in meters between the marks in three spatial directions. Thus, the user can estimate the dimensions of the obtained 3D figures of the operating areas of the IR. For example, using the dimensional grid, it is possible to determine that the area of maximum reach of the IR grip center has horizontal dimensions of 0.87 m to the right and 0.82 m to the left relative to the horizontal axis of the center of the basic IR coordinate system, and the vertical dimension relative to the base surface is: 1.19 m.

It is also possible to load operating areas obtained for different variants of fixed angular grip position into the scene for their visual verification. The tools for moving along the 3D space of the modelling scene, which are built into CoppeliaSim, make it possible to view the operating areas from different angles and projections. The applied grid of marks allows you to estimate the values of parameters or deviation distances between individual 3D figures of the operating spaces, or any other linear dimensions.



a b c d

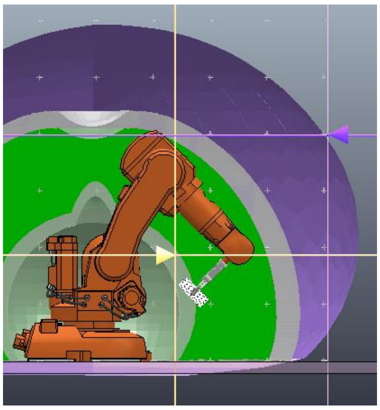
Figure 5. Interface with multiple projections (a, b, c, d), displaying the results of modelling the operating areas of the IR taking into account the angular position of the IR grip (the following elements are presented in different colors: purple – the area of maximum reach of the grip center without taking into account the fixed angular position of the IR grip; green – the working zone taking into account the collision-free reach of the grip center and the fixed angular position of the IR grip; gray – the operating area taking into account the collision-free spatial placement of the workpiece that is in the IR grip, which (the grip) is in the fixed angular position)

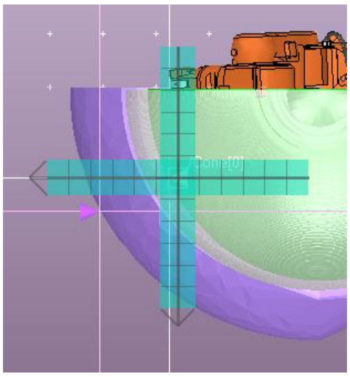
It is important to be able to obtain 3D shape of the operating area taking into account the presence of a certain workpiece grip, taking into account the size of the workpiece and possible collisions during the movement of this workpiece by the IR gripper, including collisions of the workpiece with the elements of the IR gripper itself. Having placed such 3D shape of the operating area in 3D space of the modelling scene, it is possible, for example, to conclude about the expediency of using certain models of the IR precisely because of the possible reduction of their operating space under the conditions of operating workpieces of certain sizes and shapes.

a b

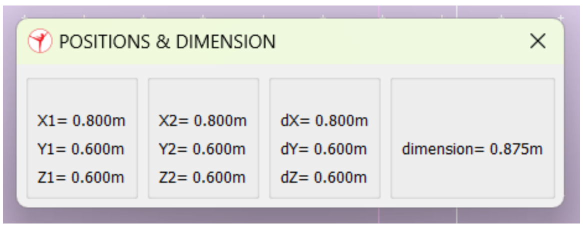
Figure 6. Dimensional vertical and horizontal parts of the grid (or) and dimensional cube (b) for enabling the verification of the actual dimensions of the operating areas

To estimate the dimensions of the working areas relative to the basic coordinate system of the IR, it was necessary to develop the separate tool due to which the user could interact on the virtual 3D scene. Simultaneous display of all dimensions for all elements of the scene of the 3D modelling space would load the scene with a large number of additional objects. Therefore, the decision which is often practiced in similar situations: display of only two coordinate indicators – size indicators (Fig. 7, a, b) is made. The coordinates are counted relative to the center of the basic coordinate system of the IR. In this case, the values of the dimensions and coordinates are displayed in the user interface window (Fig. 7, c).





a b



c

Figure 7. Measuring movable indicators of the dimensions of operating areas: selected positions of the size indicators (projection: side view) (a); visualization of the axes of movement of the indicators (projection: top view) (b); interface window displaying the current measured coordinates of the centers of the indicator axes and the value of the calculated size (c)

For convenience, the user uses marker arrow objects when dragging the size pointers. The marker arrows simultaneously point to the centres of the dimension pointers' coordinate systems. The visible axes of the size pointers make it possible to align their position relative to the visible elements of the scene. You can read the coordinate values in the separate user-defined POSITIONS & DIMENSION window in the axis directions: (X1,Y1,Z1) – for the labelled centre of the first pointer's coordinate system, (X2,Y2,Z2) - for the labelled centre of the second pointer's coordinate system, (dX,dY,dZ) – the value of the difference between the corresponding coordinates, dimension - the value of the calculated size, which is taken as the distance between the axis coordinate centres The size is obviously calculated as the root of the sum of squares of the dX, dY, dZ.

Thus, to use dimensional measurement tools, it is sufficient to have the skills of spatial thinking, movement in the 3D space of the scene and movement of objects on it. This statement is also true for other user tools that make up the scene interface of the 3D modelling space in the automated certification system of the spatial metric of the IR.

4. CONCLUSIONS

The results of the presented investigations indicate the possibility of creating an effective software system for automated metric certification to determine and evaluate the spatial parameters of the IR that characterise the operating area of the analysed IR, taking into account the spatial position of the IR gripper. The practical implementation of such system required solving a number of applied problems: virtualisation of the IR model, algorithmisation of the control of model elements in order to check (verify) the possibility of a particular position of the links of the IR manipulation system while maintaining the predefined parameters of the spatial position of the IR gripper, automatic formation and visualisation of the IR operating area taking into account the imposed initial conditions, and the possibility of quantifying the spatial parameters of the formed operating areas.

The virtualisation of the IR model is ensured by using the CAD models of the IR provided by the manufacturer, which are refined by the connections of rotary joints. The cross-platform software environment CoppeliaSim was used as the implementation environment for experimental modelling, which, as tested and proven in the course of the study, meets the requirements for building the system for automated certification of the IR metric in terms of a set of tools and functions. CoppeliaSim, in addition to direct visualisation of spatial objects, has the functionality of building mechanically connected multi-element structures and a wide range of necessary tools for managing their statics and dynamics, including tools and functions that do not require programming knowledge. These features of the CoppeliaSim software environment ensure its versatility and allow it to be used also in cases where the virtual model of the structure is not provided by the manufacturer.

The task of algorithmisation during the experiment was solved due to the application of the Lua interpreter built into CoppeliaSim. Based on the scripts of this language, the subsystem for managing the model of the IR during the automated certification of the metric is built, which results in spatial models of the IR operating areas, taking into account the imposed restrictions on the angular position of the IR gripper. The Lua scripts have a set of command instructions and implement the so-called ‘high’ level of programming, which allows the user to communicate with the elements of the workover unit structure and control the angular positions during the operation of the system under development without user intervention. In addition, the Lua language editor in CoppeliaSim allows the use of inline instructions in the code, which make it possible to form markup for displaying multi-window user interfaces, which, in turn, can be used to set the initial settings of the automated verification algorithm and to display the results of its execution.

At the current stage of development of the automated metric certification system based on the tools of the CoppeliaSim software environment, the task of initialising the initial data for the main algorithm was carried out by developing a spatial visual 3D object of the fixed angular position of the gripper. This interface tool is specially designed for the application by the user intuitively, withou making changes to the main algorithm's software script.

During the investigations it is also found that the task of automatic formation and visualisation of the working area of the IR, taking into account the imposed initial conditions, can be solved by using special Lua commands to process the cloud of possible points: commands for setting angular positions, commands for controlling collisions, commands for forming 3D surfaces that perform special calculations using Convex Hull algorithms [24].

The resulting point cloud is a sequence of numerical data on the spatial coordinates of the permitted positions of the IR, taking into account the imposed restrictions on the angular position (orientation) of the gripper. The formed areas are ultimately the result of the algorithm for automated certification of the IR metric under the constraints on the angular position of the gripper.

The task of measuring the dimensions of the operating areas is solved by developing the separate tool consisting of two axial dimension gauges moving in 3D space and the window displaying the values of their current coordinates and the distance between their centres, which is interpreted as the measured size. It is provided such measurement method that requires only the skills of orientation and movement of objects in 3D displays, given that the latter one is typical for users from the design staff.

From the authors' point of view the prospects for further researches are as follows:

- to consider the development of the separate single window user interface that would allow setting the angular position of the gripper with arbitrary accuracy, setting the discreteness of angular positions during the search and other initial modelling parameters without direct access to the CoppeliaSim tools;
- to ensure the switching within a certain array of IR models;
- to improve the capabilities of the user interface for setting the angular position of individual links of the manipulation system of the analysed IR by means of unambiguous and simple interactive elements such as digital field and slider;
- to investigate the possibility of automatic measuring the boundary dimensions of the resulting operating areas and displaying the measurement results using the user interface or by outputting the generalised modelling results to the separate report file.

However, the main direction of further researches can be defined as the work on the refinement of the software product to the state of suitability for practical application by users, i.e. for real engineering use, in the field of design, including those who are not trained in programming skills, with orientation towards the requirements of industrial enterprises.

References

1. Hlembotska L., Balytska N., Melnychuk P., Melnyk O. (2019) Computer modelling power load of face mills with cylindrical rake face of inserts in machining difficult-to-cut materials. Scientific Journal of TNTU, vol. 93, no. 1, pp. 70–80. Doi: 10.33108/visnyk_tntu2019.01.070.
2. Lutsiv I., Dubyniak T., Manziy O., Andreichuk S. (2022) Mathematical represantation of the branch kinematics of a transmission with descrete flexible connection. Scientific Journal of TNTU, vol. 106, no. 2, pp. 5–15. Doi: 10.33108/visnyk_tntu2022.02.005.
3. Liu Y., Zhao L., Liang M., Wang F. (2024) Kinematics Study of Six-Axis Industrial Robots Based on Virtual Simulation Technology. Proceedings of the 13th International Conference on Computer Engineering and Networks. CENet 2023. Lecture Notes in Electrical Engineering, Springer, vol. 1125, pp. 520–531. Available at: https://www.researchgate.net/publication/377133228_Kinematics_Study_of_Six-Axis_Industrial_Robots_Based_on_Virtual_Simulation_Technology. https://doi.org/10.1007/978-981-99-9239-3_51
4. Li L., Neau M., Ung T., Buche C. (2024) Crossing Real and Virtual: Pepper Robot as an Interactive Digital Twin. RoboCup 2023: Robot World Cup XXVI. RoboCup 2023. Lecture Notes in Computer Science, Springer, vol. 14140, pp. 275–286. https://doi.org/10.1007/978-3-031-55015-7_23

5. Guanopatin A. V., Ortiz J. S. (2023) Meaningful Learning Processes of Service Robots Through Virtual Environments. Proceedings of the Future Technologies Conference (FTC), Vol. 1. FTC 2023. Lecture Notes in Networks and Systems, Springer, vol. 813, pp. 59–73. https://doi.org/10.1007/978-3-031-47454-5_5
6. Ge Y., Hu Y., Sun X. (2023) Co-Design of Service Robot Applications Using Virtual Reality. Human Factors in Virtual Environments and Game Design. AHFE. International Conference. AHFE Open Access, AHFE International, USA, vol. 96, pp. 65–73. https://doi.org/10.54941/ahfe1003868
7. Liu R., Wandeto J., Nageotte F., Zanne P., de Mathelin M., Dresp-Langley B. (2023) Spatiotemporal Modeling of Grip Forces Captures Proficiency in Manual Robot Control. Bioengineering, vol. 10, 59, pp. 1–18. Available at: https://www.researchgate.net/publication/369021403_Spatiotemporal_modeling_of_grip_forces_captures _proficiency_in_manual_robot_control https://doi.org/10.3390/bioengineering10010059
8. Junya Y., Kenji T., Takahiro W. (2024) Effect of Presenting Stiffness of Robot Hand to Human on Human-Robot Handovers. TechRxiv, April 01, pp. 1–8. https://doi.org/10.36227/techrxiv.171198254.46018996/v1
9. Lu Y., Deng B., Wang Z., Zhi P., Li Y., Wang S. (2022) Hybrid Physical Metric For 6-DoF Grasp Pose Detection, arXiv.2206.11141, vol. 1, pp. 1–7. Available at: https://www.researchgate.net/ publication/361479841_Hybrid_ Physical_Metric_For_6-DoF_Grasp_Pose_Detection. https://doi.org/10.1109/ICRA46639.2022.9811961
10. Barnfather J., Goodfellow M.J., Abram T. (2016) A performance evaluation methodology for robotic machine tools used in large volume manufacturing. Robotics and Computer-Integrated Manufacturing, vol. 37, pp. 49–56. Available at: https://www.researchgate.net/publication/281716706_A_performance_evaluation_methodology_ for_robotic_machine_tools_used_in_large_volume_manufacturing https://doi.org/10.1016/j.rcim.2015.06.002
11. Slamani M., Nubiola A., Bonev I. (2012) Assessment of the positioning performance of an industrial robot. Industrial Robot: An International Journal, vol. 39, no. 1 , pp. 57–68. Available at: https://www. researchgate. net/publication/238308032_Assessment_of_the_positioning_performance_of_an_industrial_robot https://doi.org/10.1108/01439911211192501
12. Panneerselvam S., Karthikeyan R. (2020) Simulation of Robot Kinematic Motions using Collision Mapping Planner using Robo Dk Solver. International Journal of Innovative Technology and Exploring Engineering, vol. 9, pp. 2278–3075. Available at: https://www.researchgate.net/publication/ 348232778_Simulation_of_Robot_Kinematic_Motions_using_Collision_Mapping_Planner_using_Robo _Dk_Solver. https://doi.org/10.35940/ijitee.J7588.0991120
13. Chakraborty S., Aithal S. (2021) ABB IRB 120-30.6 Build Procedure in RoboDK. International Journal of Management, Technology, and Social Sciences, vol. 6, no. 2, pp. 256–264. Available at: https://www.researchgate.net/publication/357158252_ABB_IRB_120-306_Build_Procedure_in_Robo DK. https://doi.org/10.47992/IJMTS.2581.6012.0169
14. Henriques J., Neto E., Paiva J., Carneiro S., Letícia L., Alexandre F., Flávio C., Giuliano. Trajectory Generation Using RoboDK for a Staubli SCARA TS 60 Robot. 2023 11th International Conference on Control, Mechatronics and Automation (ICCMA), Grimstad, Norway. 2023, pp. 121–126. https://doi.org/10.1109/ICCMA59762.2023.10374649
15. Goryl K., Pollák M. (2023) Calibration of Panasonic TM-2000 Welding Robot Using Simulation Software. EAI ARTEP 2023. EAI International Conference on Automation and Control in Theory and Practice. Springer Innovations in Communication and Computing. Springer, pp. 273–284. https://doi.org/10.1007/978-3-031-31967-9_21
16. Salihovic I., Skamo A., Jokic D. (2021). RoboDK to MATLAB Joint Position Transformation. 2021 Selected Issues of Electrical Engineering and Electronics (WZEE), Rzeszow, Poland, pp. 1–6. https://doi.org/10.1109/WZEE54157.2021.9576924
17. Chakraborty S., Aithal S. (2021) Forward and Inverse Kinematics Demonstration using RoboDK and C#. International Journal of Applied Engineering and Management Letters (IJAEML), vol. 5, no. 1, pp. 97–105. https://doi.org/10.47992/IJAEML.2581.7000.0095
18. Kyrylovych V., Kravchuk A., Melnychuk P., Mohelnytska L. (2021). Automated Attestation of Metrics for Industrial Robots' Manipulation Systems. Advanced Manufacturing Processes II. InterPartner 2020. Lecture Notes in Mechanical Engineering. Springer, pp. 813–822. https://doi.org/10.1007/978-3-030-68014-5_79
19. Kyrylovych V., Kravchuk A. (2023) A Three-Tiered Approach to The Initial Stages of Design of Collaborative Robotic. Technical sciences. Technologies Herald of Khmelnytskyi national university, issue 4, no. 323, pp. 180–187. Doi:10.31891/2307-5732-2023-323-4-180-187 81.5.
20. CoppeliaSim Homepage. Available at: http://www.coppeliarobotics.com.
21. Chakraborty S., Aithal S. (2021) An Inverse Kinematics Demonstration of a Custom Robot using C# and CoppeliaSim. International Journal of Case Studies in Business, IT, and Education (IJCSBE), vol. 5, no. 1, pp. 78–87. https://doi.org/10.47992/IJCSBE.2581.6942.0102
22. ABB Homepage. Available at: https://new.abb.com.
23. OnRobot – RG2 gripper. Available at: https://onrobot.com/en/products/rg2-gripper.
24. Escobar L., Kaveh K. (2020) Convex polytopes, algebraic geometry, and combinatorics // Notices of the American Mathematical Society, vol. 67, no. 8, pp. 1116–1123. https://doi.org/10.1090/noti2137

УДК 621.865.82

ОСОБЛИВОСТІ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ТА ТРИВИМІРНОГО МОДЕЛЮВАННЯ АВТОМАТИЗОВАНОЇ СЕРТИФІКАЦІЇ МЕТРИК ПРОМИСЛОВИХ РОБОТІВ У СЕРЕДОВИЩІ COPPELIASIM

Олександр Добржанський[1]; Валерій Кирилович[1]; Антон Кравчук[1]; Євген Пуховський[1]; Володимир Савків[2]

[1]Державний університет «Житомирська політехніка», Житомир, Україна
[2]Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна

Резюме. *Присвячено проблемі автоматизованої сертифікації метрик промислових роботів (МПР) у програмному середовищі CoppeliaSim. Автори детально розглядають можливості та інструменти програмного середовища CoppeliaSim для вимірювання та оцінювання таких просторових параметрів ПР, як робоча зона та конфігурація геометричного положення інструменту. Дослідження механічних, техніко-технологічних систем і віртуальних середовищ давно є предметом наукової діяльності. Наприклад, у роботах описано математичні моделі поведінки гнучких механічних передач, дослідження яких проведено без урахування автоматизованої реалізації. Відомі також дослідження методом комп'ютерного моделювання технічних систем силового навантаження різальних інструментів із використанням вузькоспеціалізованих програмних середовищ. Не розглянуто особливостей програмної реалізації пропонованих моделей. Пропонований в даній роботі матеріал акцентує увагу на універсальних програмних продуктах та мовах програмування для тривимірного моделювання. Представлено результати дослідження методів та підходів до автоматизованої реалізації метрики, які дозволяють забезпечити надійність і точність при подальшому синтезі елементів робототехнічних технологій, таких, як оптимізація розміщення обладнання, формування оптимальної траєкторії руху ланок системи маніпуляції ПР з інструментом або захватом. Наведено обґрунтування необхідності використання просторового 3D моделювання з повнорозмірними віртуальними моделями ПР у середовищі CoppeliaSim. Проаналізовано інструменти та засоби, що дозволяють врахувати вплив на метрику ПР насамперед просторових факторів, таких, як геометричні параметри конструкції ПР, інструменти, захват, можливі обмеження, зумовлені конструктивними та технологічними особливостями технологічного обладнання. Стаття може бути корисною для дослідників, інженерів та студентів, які вивчають ПР з точки зору їх автоматизованого моделювання та аналізу.*

Ключові слова: *моделювання, промислова робототехніка, робототехніка, автоматизована сертифікація метрики.*

https://doi.org/10.33108/visnyk_tntu2024.04.135 *Отримано 25.11.2024*