Вісник Тернопільського національного технічного університетуhttps://doi.org/10.33108/visnyk_tntu
Scientific Journal of the Ternopil National Technical University
2021, № 2 (102) https://doi.org/10.33108/visnyk_tntu2021.02
ISSN 2522-4433. Web: ***visnyk.tntu.edu.ua***

UDC 621.865.8:62.505

AUTOMATED PLANNING OF THE OPTIMAL MOVEMENT TRAJECTORIES OF MOBILE MECHATRONIC DEVICES

Valerii Kyrylovych[1]; Petro Melnychuk[1]; Lubomir Dimitrov[2]; Ilona Kryzanivska[1]

[1]ZhytomyrPolitechnic State University, Zhytomyr, Ukraine
[2]Sofia Technic University, Sofia, Republic of Bolgaria

Summary.*The compatible work is considered and the obtained results of the known algorithms generating unobstructed trajectories of different length and smoothness are investigated. Their operation is performed within the framework of the developed software product LSTr. The use of the analyzed set of these algorithms on the set of considered sections of the generated trajectories according to the obtained results allows a differentiated approach to the use of different algorithms on different sections of trajectories, determined by the accepted criteria of length and / or smoothness. The scientific novelty of the work and its practical significance in this area of research are determined.*

Key words: *algorithm, trajectory, optimization, mobile mechatronic device, length, smoothness.*

https://doi.org/10.33108/visnyk_tntu2021.02.064 *Received 11.04.2021*

Statement of the problem. Nowadays, when industrial robotics is intensively developing [1], the essential role is played by the tasks of optimization of various indicators design/synthesis of robotic mechanical assembly technologies (RMAT) implemented in flexible production cells (FPC) and in other technological structures of flexible production systems. Trajectory problems, which are unsolvable without planning the movements PM in FPC and choosing the optimal ones according to pre-selected criterion/criteria, are of particular importance.

To implement the planning tasks, the PM control should to be able to construct a quasi-optimal trajectory (ideally optimal) according to the previously accepted criterion and free from collisions (collisions) with other FPC elements (so-called collision-free trajectory). Construction of unobstructed, close to the optimal is necessary to minimize time and energy costs by reducing the desired length and speed of displacement, especially the grip (Gr) of PM, which in turn is determined by the corresponding displacements of the manipulation system (MS) PM.

Analysis of available research. The problem of finding the shortest path and other parameters of trajectories that bypass obstacles in a given state space is not new and includes many algorithms for its solution.

To solve effectively the problems considered here, a mobile mechatronic device (MMD) is regarded as a kind of Gr PM analogue with or without an object of manipulation (OM) fixed in it, which is displaced by clearly defined reference points. And, knowing the laws of displacement of this finite element of the MS PM, it can be stated that in the final stage, these algorithms and the obtained data can be used for further solution of the inverse problems of kinematics, i.e. by specific position of Gr PM with/without OM, that is a technological robotic kit (TRK) [2], determination of the position occupied by the entire MS PM, not only in order to determine the intersections and collisions at the level of the finite element, but also at the level of all parts of MS PM.

Today there are many algorithms for planning unobstructed trajectories, the most common of which are the following: KPIECE (Kinodynamic Motion Planning by Interior-Exterior Cell Exploration) [3]; Bi-directional KPIECE [4]; Lazy Bi-directional KPIECE [4]; PRM (Probabilistic Roadmap Method) [5–7]; SBL (A Single-Query Bi-Directional Probabilistic Roadmap Planner with Lazy Collision Checking) [6, 7], RRT (Rapidly Exploring Random Tree) [8]; RRT Connect [8, 9];Lazy RRT [10]; EST (Expansive Space Trees) [11].

Semantic analysis of certain algorithms indicates the following. These algorithms examine the state space, starting from the initial configuration q_{init} (in this case, the trajectory component of the displacement) and until they reach the final configuration q_{goal}. The of space investigation is carried out through the propagation of trees, which is a graphical illustration of the possible displacements of Gr, growing from the starting point to the end, while avoiding the existing obstacles. In some of the above mentioned algorithms, two trees T_{init} and T_{goal} are built simultaneously (in the forward direction, from the initial state q_{init}, and in the reverse direction, i.e. from the target q_{goal}, stopping after the two trees are connected).

The collision test (collision, intersection) in each of these algorithms takes place at different stages of their work, but the common is that in case of detection of an obstacle, the segment containing it is removed, and after, the search process continues. In addition, these algorithms are described mainly as those that operate in 2D space.

There is also a number of more recent investigations [12–20], which are devoted to the problem of generating and / or investigationg the trajectories of industrial robots, mobile robots, mechatronic devices, etc.

For example, in paper [12] only a certain rectilinear trajectory of an industrial robot clamping device is considered. It is analyzed in five areas. At each area, the optimal orientation of the clamping device is determined in order to ensure the continuity of the manipulation object transportation using only the force of inertia and gravity. In this case the possibility of using certain algorithms for certain trajectory areas is not analyzed. This indicates the inexpediency of using such an approach to the investigation of trajectories in the context of the following study.

The problems of trajectory planning in robotics are solved outlying in terms of the content of the investigation materials on the example of studying the usefulness of ultra-wideband sensor technologies (theultrawideband (UWB) sensortechnology) [13]. The signals on the way of robots movement recorded here are the basis for generating trajectories of their (robots) movement and achieving logistical goals. This approach “works” only for the known set of predefined routes and is obligatory for the availability of information and hardware for trajectories generation and therefore is not considered here.

In paper [14], the investigation confirmed the possibility of A * -algorithm use for off-line programming of robot movements in the known environment. However, the set of analyzed algorithms is not considered, which does not correspond to the purpose and essence of this paper (see below).

Defining and comparing only two types of trajectories obtained by on-line and off-line programming is the content of paper [15]. At the same time the estimation of the received trajectories on their separate areas is not considered as areas by themselves are not regarded. Therefore, the usefulness of the analysis of such trajectories is questionable in terms of the content of problems considered here.

The trajectory generation of the mobile robot based on the data of infrared distance sensors and the use of artificial neural networks is presented in paper [16]. Neither the set of algorithms by which the obtained trajectories can be evaluated nor the trajectory sections where one of the algorithms can be effective are considered here. This makes this investigation not really useful for this work.

In paper [17] the trajectory of mobile robots is generated on the basis of operating positions map using ADAMS and MATHLAB programs. This is typical for the tasks of positioning and planning of technological equipment (here it is working position). The use of data according to [17] for the investigation of this paper is impossible due to different problem statements, and hence different obtained results.

Paper [18] is devoted to the analysis of optimization methods providing energy-saving trajectories of industrial robots and automatic systems, including mechatronic ones. But due to the obvious informative and methodological value this work can not be used in the given investigation, as it does not involve the use of algorithm sets that can be effective in different areas of it, which are also not considered as areas by themselves.

Kinematic redundancy as a tool for designing energy-saving trajectories is studied in paper [19]. The trajectory of the robotic cell components is spline interpolation by solving the inverse dynamics problem, making it possible to improve the speed and energy criteria. But neither smoothing the trajectory nor dividing it into technologically significant areas are considered. This makes it impossible to use this approach for the below mentioned tasks.

The content of paper [20] consists in the proposed smooth and time-optimal method of planning the S-curve by means of continuous path differentiation at the limits set for the robot and other machines in terms of speeds, accelerations, etc. The impossibility of this method application for the following tasks is determined both by the absence of many similar methods that can generate the appropriate trajectories, and the impossibility of their (methods) application in certain areas of each possible trajectories.

Papers [12–20] are common due to the obvious content and methodological importance of the tasks considered in them, as well as the lack of their software, which allows their direct implementation. This in general makes it impossible to use them for solving the below mentioned problems of this investigation.

Undoubtedly, their use (papers [12–20]) for the investigations given below is fundamentally possible. But this requires significantly large intellectual and time-consuming resources in terms of software development for their operation.

Therefore, taking into account the above mentioned, the content of the following investigation is based on algorithms [3–11].

Objective of the investigation is to increase the validity of technological decisions made in the automated synthesis of RMAT at the stage of solving trajectory problems, represented by the movement of mobile mechatronic devices (MMD) in 3D space by generating trajectories that are effective in terms of their length and/or smoothness.

Statement of the task is to highlight the results of the joint (integrated) use of the above mentioned algorithms, minimizing in certain areas the length and/or smoothness of the implemented trajectories of MMD as a certain analogue of the TRC moving in the pre-generated scene, which in turn is analogous to the analyzed FPC with its design and planning characteristics and parameters. This makes it possible further to solve the direct and inverse problems of kinematics, and hence the corresponding problems of dynamics, which are obligatory components of the synthesis of PMAT in FPC.

Presentation of basic material. For the successful implementation of the MMT trajectory planning process, information support (IS) is proposed, the structure of which is presented in Figure1.

The input information is conventionally divided into constant, variable and conventionally variable.

Constant information contains the basic information on the analyzed algorithms of planning of unobstructed trajectories. Sources of constant information are resources with the content and essence of algorithms, as well as their software implementation, which is freely available (OMPL library).

Variable information contains data on:

- a scene, which is a certain form of limited 3D-space with existing obstacles;
- final and initial coordinates of the object of manipulation (OM), shape and geometric dimensions of OM;
- final and initial coordinates of the MMD, its shape and geometric dimensions.

The sources of variable information are the technical task that must be implemented to achieve this goal.

Depending on the task, MMD can also refer to conventionally variable information. This is possible in case when different OM are used in one MMP.

The sources of conventionally variable information are specific production conditions in which the relevant trajectories are analyzed/designed.

The result of displacement of MMD along the found trajectories in solving the problem is the choice of the optimal ones according to the previously accepted criteria of smoothness *S* or length *L* on 3 sections of the scene (see below).

The main requirements that have to be ensured are smooth (collision-free) displacement of MMD on the shortest trajectory, as well as its continuity and evenness.

The formal statement of the solved problems involves determining the proposed system of criteria that logically follow from the semantic essence of the analyzed algorithms $A_1, \dots, A_9$.

In this regard, the criteria used below are as follows:

- local:

◦the minimum length $min\, L_{A_j}$of local trajectories of MMD displacement in sections 1–2, 2–3 and 3–1, each of which is provided by the corresponding *A*$A_j$algorithm from their final analyzed set $A = (A_j |\, j = \overline{1,9})$:

$$F_{L_{A_j}}(z_i) = min\, L_{A_j}, i = \overline{1,3};\ z \in (1-2, \qquad 2-3, \qquad 3-1); j = \overline{1,9}, \tag{1}$$

where A_1–algorithm *KPIECE*; A_2–algorithm *Bi directionalKPIECE*; A_3–algorithm *LazyBi directionalKPIECE*; A_4–algorithm *PRM*; A_5–algorithm *SBL*; A_6–algorithm *RRT*; A_7–algorithm *Lazy RRT*; A_8–algorithm *RRT Connect*; A_9–algorithm *EST*; z_1 – *1*-st segment; z_2– *2*-nd segment; z_3 – *3*-rd segment of the trajectory(about these segments, see further);

◦ the least smoothness of working off the local trajectories of MMD displacement on segments*1–2*, *2–3* and *3–1*:

$$F_{S_{A_j}}(z_i) = min\, S_{A_j}, i = \overline{1,3};\ z \in (1-2, \qquad 2-3, \qquad 3-1); j = \overline{1,9};\ S_{A_j} \in [0,1], \tag{2}$$

where $A_1, \ \dots, \ A_9$arethe above trajectory planning algorithms; $z_1, \ z_2, \ z_3$ – the first (1–2), the second (2–3) and the third (3–1) sections of trajectories, respectively;

- global:

◦ the total minimum length of the MMD displacement, taking into account the minimum length of MMD displacement in each of the sections 1–2, 2–3 and 3–1 according to one of the analyzed algorithms:

$$F_{L_{min}} = min\, L_{A_j(1-2)} + min\, L_{A_j(2-3)} + min\, L_{A_j(3-1)}, \tag{3}$$

where$L_{A_j(1-2)}$ – the length of the trajectory on the section 1–2; $L_{A_j(2-3)}$ – the length of the trajectory on the section 2–3; $L_{A_j(3-1)}$ – the length of the trajectory on the section 3–1.

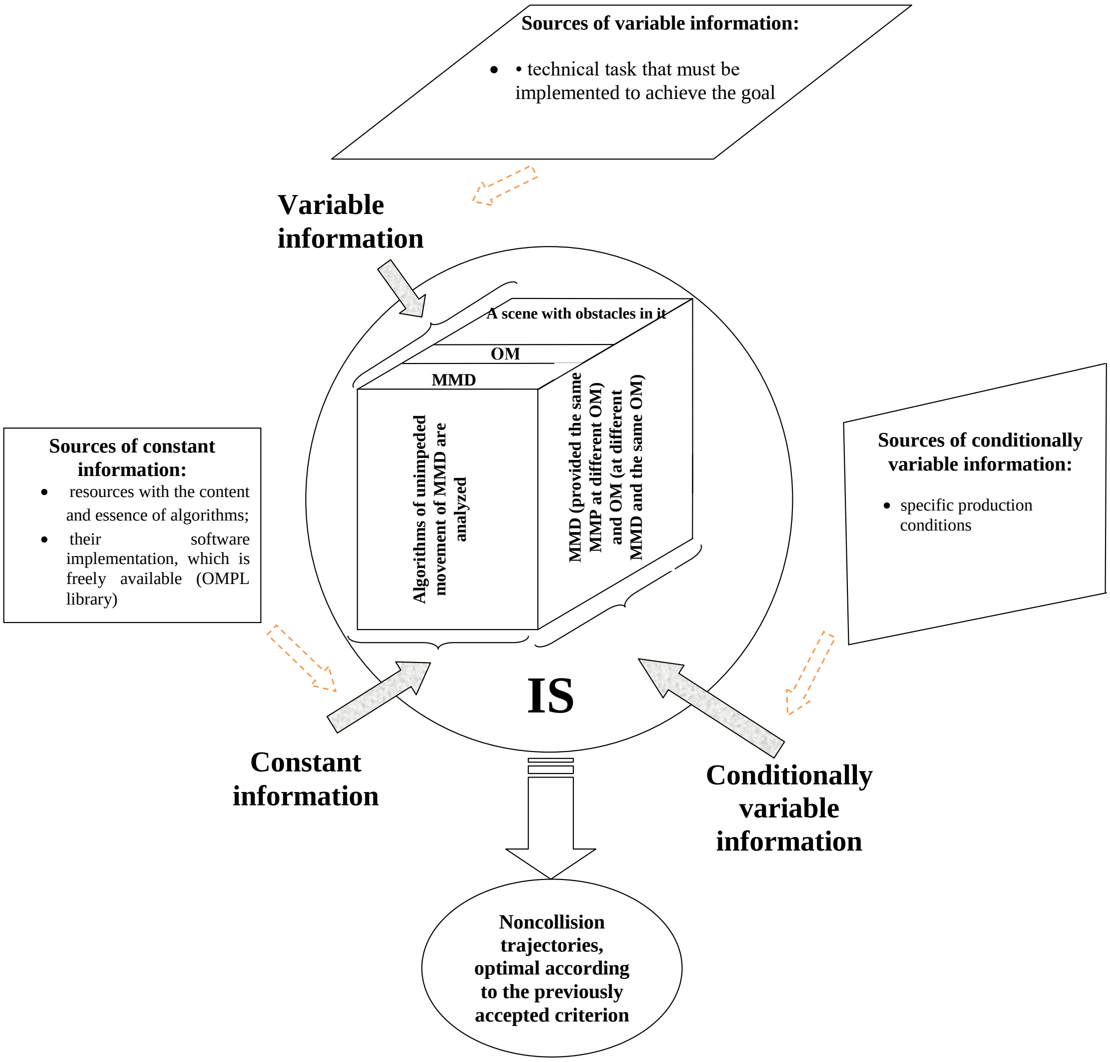

Sources of variable information:
• technical task that must be implemented to achieve the goal
Variable information
A scene with obstacles in it
OM
MMD
Algorithms of unimpeded movement of MMD are analyzed
MMD (provided the same MMP at different OM) and OM (at different MMD and the same OM)
Sources of constant information:
resources with the content and essence of algorithms;
their software implementation, which is freely available (OMPL library)
Sources of conditionally variable information:
specific production conditions
IS
Constant information
Conditionally variable information
Noncollision trajectories, optimal according to the previously accepted criterion

Figure 1. The structure of the proposed information support

It is appropriate to emphasize that the global criterion for the smoothness of the trajectories $\boldsymbol{F_S}$is subject to additional and more detailed study, which is not considered here and in such a statement, but is calculated by the content of each algorithm. The same can be said for the combined criterion $\boldsymbol{F_{LS}}$, which comprehensively takes into account the lengths and smoothness of trajectories, i.e. local criteria $\boldsymbol{F_{L_{A_j}}}$and$\boldsymbol{F_{S_{A_j}}}$.

Below is a description of the developed software product (SP) *LSTr* and the results of solving one of the test examples of its (SP) operation. For software implementation of the content of the tasks the C ++ programming language is chosen with the use of:

- graphic library*OpenGL* (*OpenGraphics Library*);
- *QT*, which is a cross-platform software development toolkit and includes all the basic classes required in the development of graphical interface elements;
- *OMPL* (*Open Motion Planning Library*) libraries for constructing the trajectories themselves;

- *Assimp* (*Open Asset Import Library*) libraries to work with 3D-models;
- *SketchUp*software to create scene 3D-models.

Figure 2 shows a simplified block diagram of the operation of the developed *LSTr* software, which demonstrates the relationship of different application software packages for planning a smooth (collision-free) trajectory of the MMD displacement.

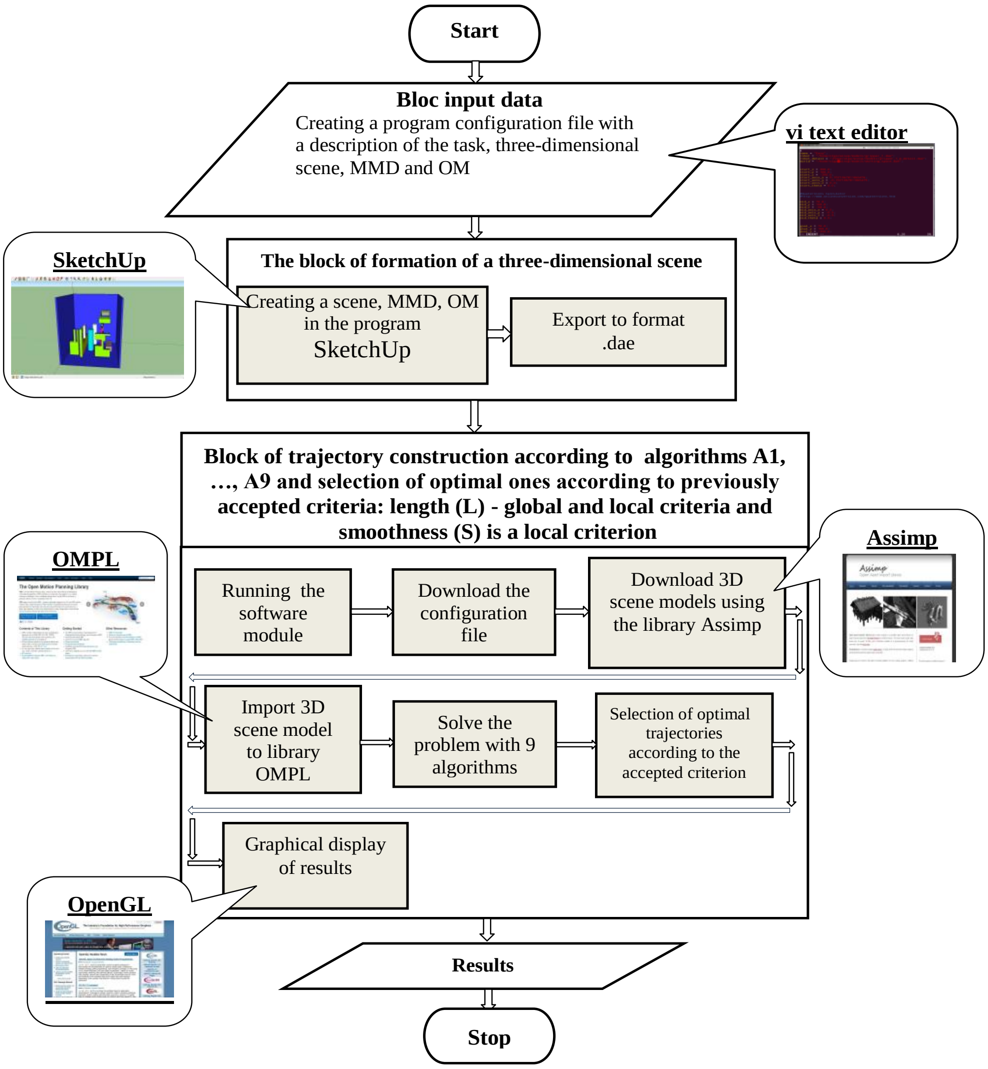

Start
Bloc input data
Creating a program configuration file with a description of the task, three-dimensional scene, MMD and OM
vi text editor
SketchUp
The block of formation of a three-dimensional scene
Creating a scene, MMD, OM in the program SketchUp
Export to format .dae
Block of trajectory construction according to algorithms A1, …, A9 and selection of optimal ones according to previously accepted criteria: length (L) - global and local criteria and smoothness (S) is a local criterion
OMPL
Assimp
Running the software module
Download the configuration file
Download 3D scene models using the library Assimp
Import 3D scene model to library OMPL
Solve the problem with 9 algorithms
Selection of optimal trajectories according to the accepted criterion
Graphical display of results
OpenGL
Results
Stop

Figure 2. The simplified structural scheme of operation of the developed software *LSTr*

In general, the work of the developed *LSTr* software consists of 4 unequal stages in terms of calculated resources: 1 – formation of the corresponding configuration files; 2 – choosing of trajectory selection criteria; 3 – launching of the analyzed set of algorithms *A1,…, A9*; 4 – obtaining results, their analysis and visualization. This is obvious from Figure 2. In this case, any number of algorithms are used to build a trajectory of the user's choice. The obtained results are compared and a decision is made on their (obtained decisions) optimality.

Figure 3 presents the components of the information field developed by *LSTr* SP. According to Figure 3, the problem is divided into three parts, the solution of which is performed in the following sections:

- *1*-st section – displacement of MMDfrom the starting*Initial position*toOM (*Take object of manupulation*) (section 1–2);

- *2*-nd section – MMDwith OM displaces to the aim*GOAL*, where leaves OM (section 2–3);
- *3*-rd section – 'empty', i.e. without OM in Gr, when MMD returns to the initial position (section 3–1) *Initial position*.

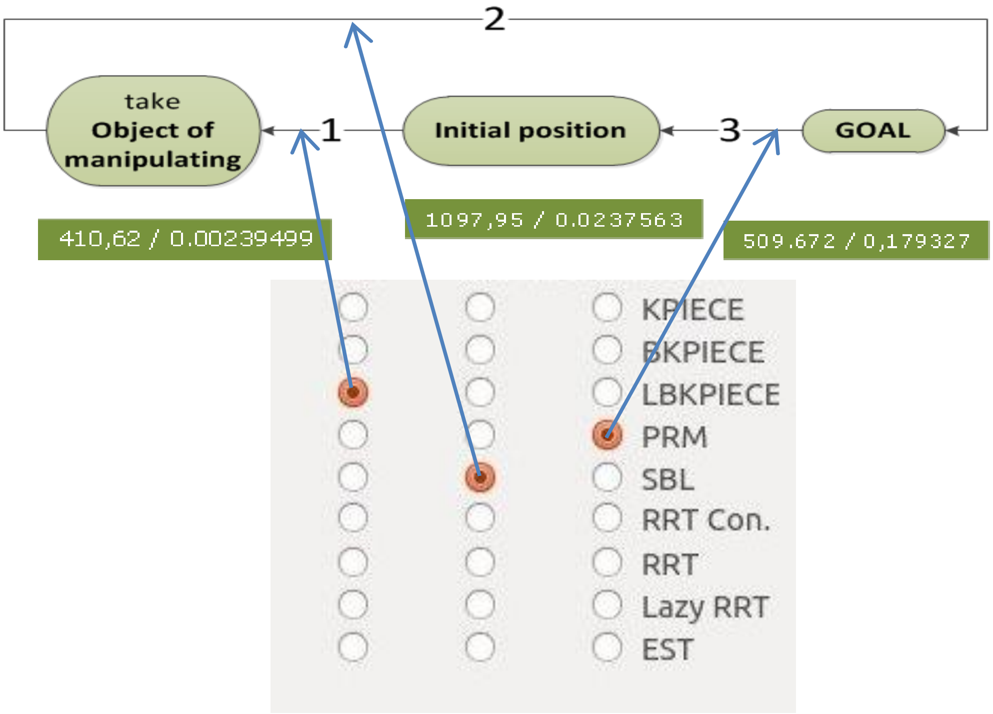


Figure 3. The information field of the developed software *LSTr*

The specified information field with the selected sections 1–2, 2–3 and 3–1 functionally corresponds to the work of the PR during the technological loading / unloading of a certain working position of the mechanical assembly FPC by the corresponding object of manipulation.

In Figure 3, the red buttons indicate the activated appropriate algorithms, which are randomly selected by the user to analize the results. The latter are represented on the green field, where the first of them reproduces the obtained length L of the trajectory, and through the symbol "/" – thedigital value of the smoothness *S* of the trajectory. In fact, by activating the appropriate buttons, the user can select any number of them and then select one or another algorithm for the final calculation of the relevant parameters.

To solve the test problems, a finite set of scenes was created, which includes the state space (similar to the structural components of FPC, which are interconnected by certain coordinates of placement in the coordinate system of PR and technological sequence of PR operations) and parameters of obstacles (design and geometric features of each working position FPC), MMD and OM, the initial and final purpose of MMD, i.e. the initial and final coordinates of the points of movement of MMD.

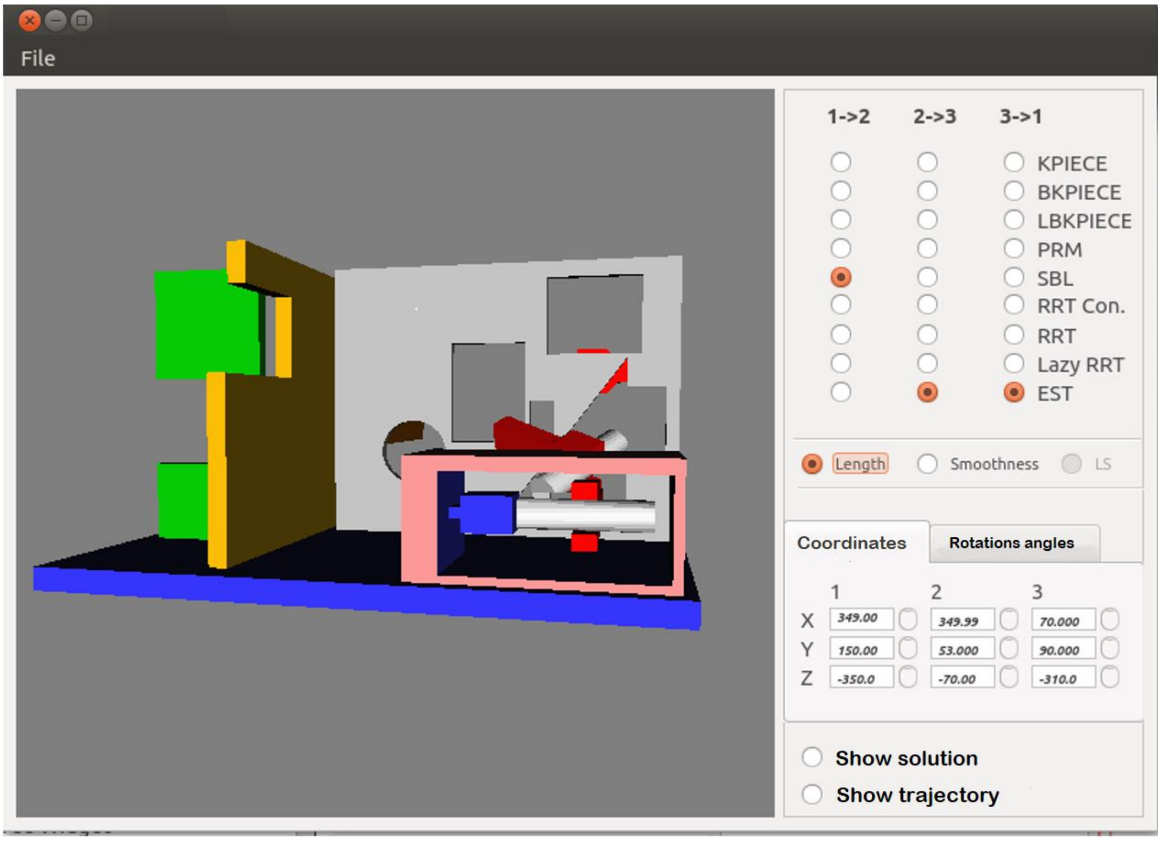
Figures 4–8 show the results of solving one of the test problems in the form of screenshots of the operation of the developed software *LSTr*, which corresponds to one of the constructed scenes from their generated finite set.

According to Figure 4 (length criterion *L*, its values are presented in conventional units of length), where the activated algorithms are randomly selected by the user:

• in the *1*st section (1–2) the best result was provided by the *SBL* algorithm with the value L1min = 336,810; L_{1min} = 336,810;

• in the 2nd section (2–3) the best result was given by the *EST* algorithm with the value L_{2min} = 710,749;

• in the 3rd section (3–1) the best result was tested by the *EST* algorithm with the value L_{3min} = 306,724.



1->2	2->3	3->1	
○	○	○	KPIECE
○	○	○	BKPIECE
○	○	○	LBKPIECE
○	○	○	PRM
●	○	○	SBL
○	○	○	RRT Con.
○	○	○	RRT
○	○	○	Lazy RRT
○	●	●	EST

● Length ○ Smoothness ○ LS

Coordinates | Rotations angles

	1	2	3
X	349.00	349.99	70.000
Y	150.00	53.000	90.000
Z	-350.0	-70.00	-310.0

○ Show solution

○ Show trajectory

Figure 4. The result of the analyzed algorithms on the analyzed scene by the criterion of length *L*

According to Figure 5 (smoothness criterion *S*, its values are presented in conventional units of measurement of the smoothness for the trajectories provided by the corresponding analyzed algorithms *A1,…, A9*):

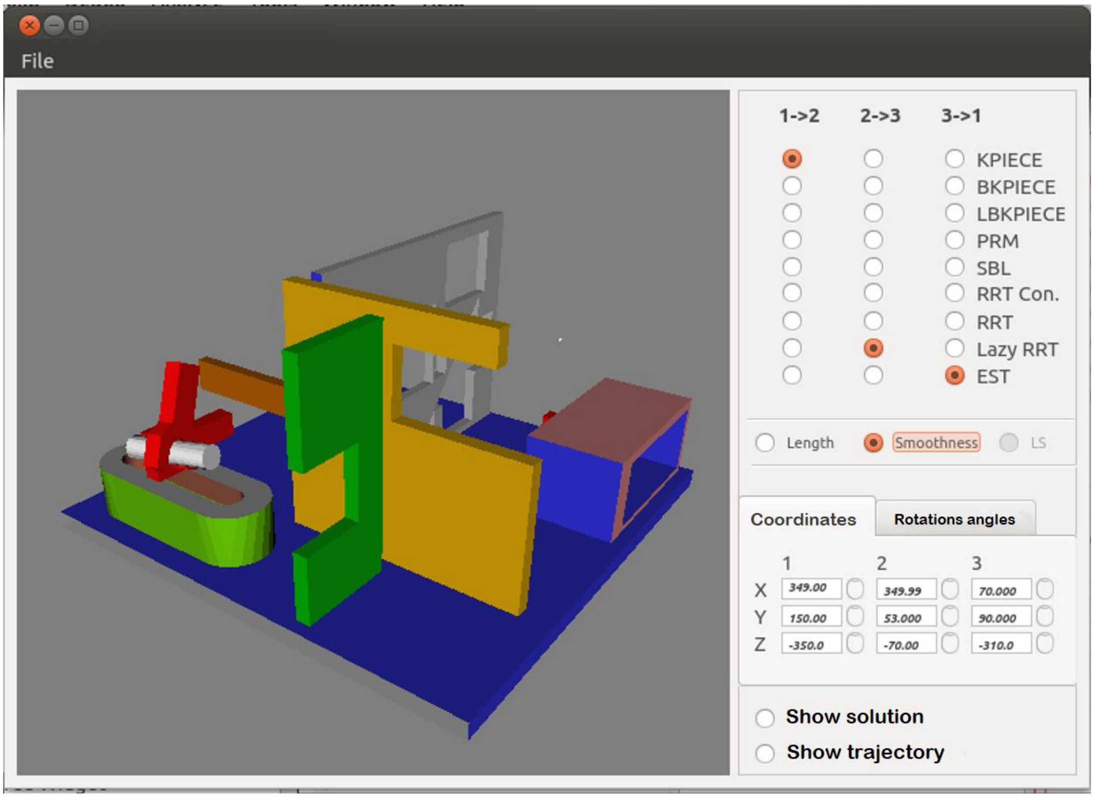
◦ on the *1*st section (1–2) the best result was provided by the *KPIECE* algorithm with the value значеннямS$_{1min}$=0,00614645;

◦ in the 2nd section (2–3) the best result was given by the *Lazy RRT* algorithm with the value S_{2min}=0,00712410;

◦ in the 3rd section (3–1) the best result was shown by the *EST* algorithm with the value S_{3min}=0,015405091.

Graphical representation of the obtained trajectories by the criteria of length *L* and smoothness *S* are shown in Figure 6 and Figure 7, respectively.

The general results of the calculated criteria for trajectories implemented by algorithms *A1,…, A9* according to the accepted criteria of length *L (Solution length)* and smoothness *S (Solution smoothness)*, are shown in Figure 8. Optimal results are determined on the set of sections 1–2, 2–3 and 3–1. Thus, in section 1–2 the shortest path length is provided by the *SBL* algorithm (L_{1min}=336,810 units), in sections 2–3 (L_{2min}=710,749units) and in 3–1 (L_{3min}=306,724units) – by the *EST* algorithm.



1->2	2->3	3->1
●	○	○ KPIECE
○	○	○ BKPIECE
○	○	○ LBKPIECE
○	○	○ PRM
○	○	○ SBL
○	○	○ RRT Con.
○	○	○ RRT
○	●	○ Lazy RRT
○	○	● EST

○ Length ● Smoothness ○ LS

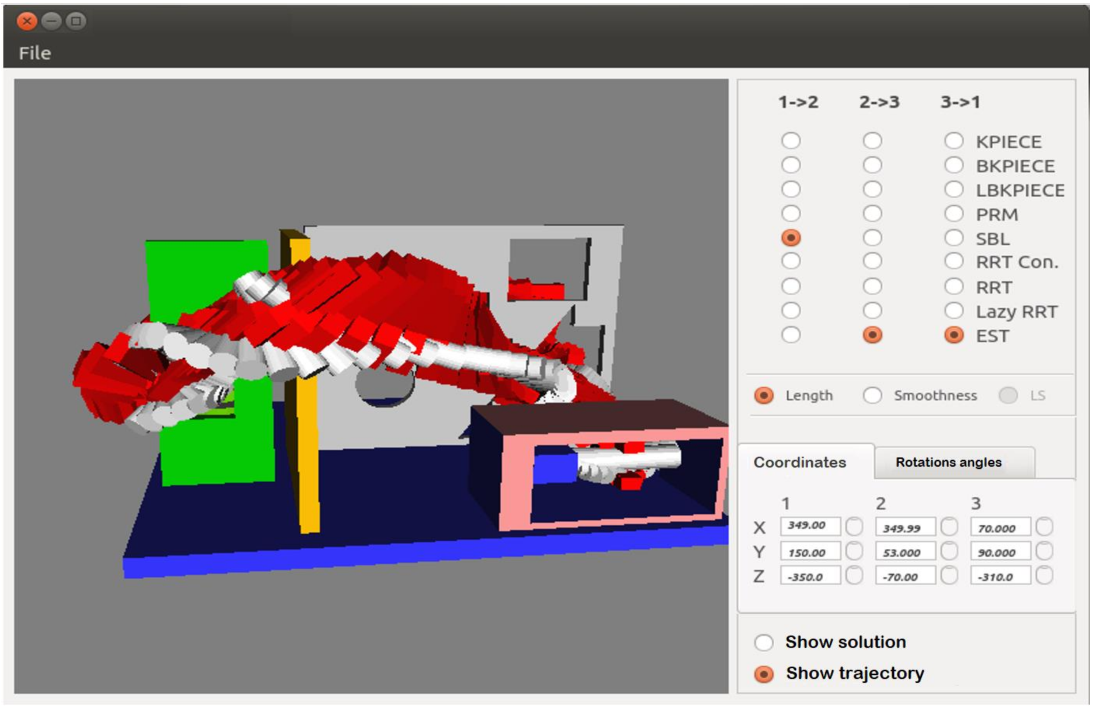
Coordinates | Rotations angles

	1	2	3
X	349.00	349.99	70.000
Y	150.00	53.000	90.000
Z	-350.0	-70.00	-310.0

○ Show solution

○ Show trajectory

Figure 5. The result of the analyzed algorithms on the analyzed scene by the criterion of smoothness *S*



1->2	2->3	3->1
○	○	○ KPIECE
○	○	○ BKPIECE
○	○	○ LBKPIECE
○	○	○ PRM
●	○	○ SBL
○	○	○ RRT Con.
○	○	○ RRT
○	○	○ Lazy RRT
○	●	● EST

● Length ○ Smoothness ○ LS

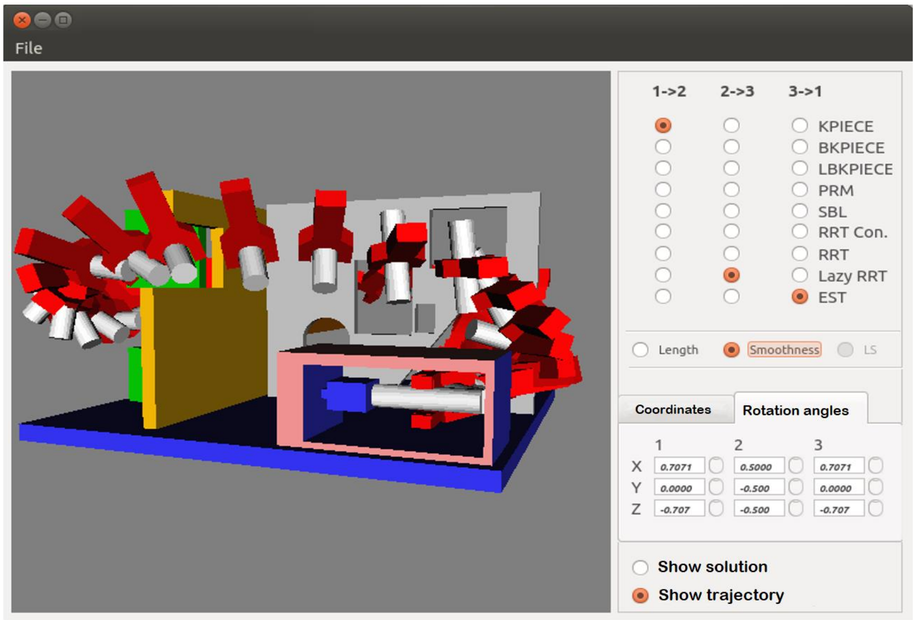
Coordinates | Rotations angles

	1	2	3
X	349.00	349.99	70.000
Y	150.00	53.000	90.000
Z	-350.0	-70.00	-310.0

○ Show solution

● Show trajectory

Figure 6. The graphical representation of the trajectory according to Figure 4



1->2	2->3	3->1	
●	○	○	KPIECE
○	○	○	BKPIECE
○	○	○	LBKPIECE
○	○	○	PRM
○	○	○	SBL
○	○	○	RRT Con.
○	○	○	RRT
○	●	○	Lazy RRT
○	○	●	EST

○ Length ● Smoothness ○ LS

Coordinates | Rotation angles

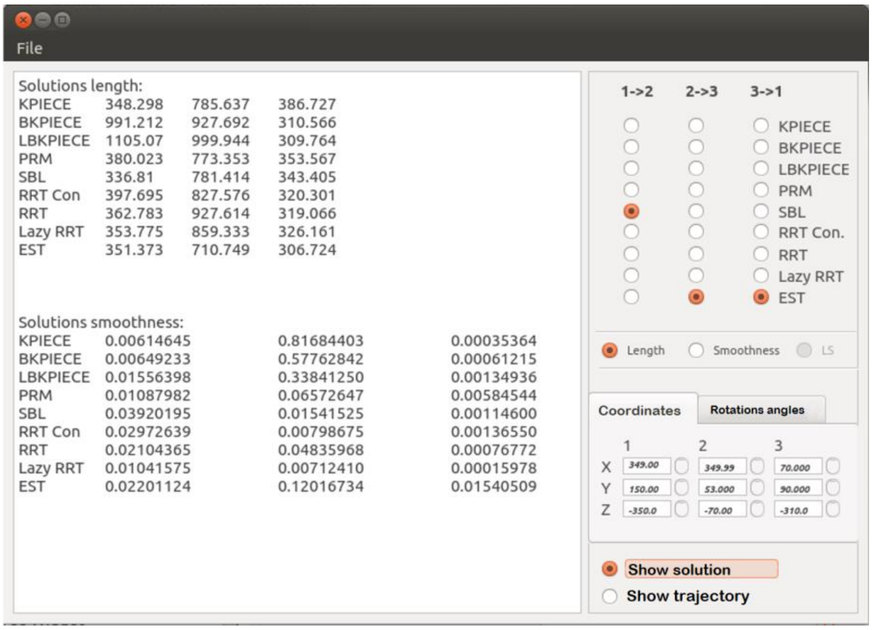
	1	2	3
X	0.7071	0.5000	0.7071
Y	0.0000	-0.500	0.0000
Z	-0.707	-0.500	-0.707

○ Show solution
● Show trajectory

Figure 7. The graphical representation of the trajectory according to Figure 5

The general results of the calculated criteria for trajectories implemented according to algorithms *A1,…, A9* due to the accepted criteria of length *L* (*Solution length*) and smoothness *S* (*Solution smoothness*), are shown in Figure 8. Optimal results are determined on the set of sections 1–2, 2–3 and 3–1. Thus, in section 1–2 the shortest path length is provided by the *SBL* algorithm (L_{1min} = 336,810 units), in sections 2–3 (L_{2min} = 710,749 units) and 3–1 (L_{3min} = 306,724 units) – the *EST* algorithm.

Similarly, conclusions can be drawn for the criterion of smoothness of the trajectory, for which activate the *Smoothness* button. In this case (see Figure 8), in section 1–2 it is advisable to use the algorithm *KRITCE* (S_{1min}=0,00614645), in sections 2–3 (S_{2min}=0,00712410) and 3–2 (S_{3min}=0,00015978) – the algorithm *Lazy RRT*.



Solutions length:

KPIECE	348.298	785.637	386.727
BKPIECE	991.212	927.692	310.566
LBKPIECE	1105.07	999.944	309.764
PRM	380.023	773.353	353.567
SBL	336.81	781.414	343.405
RRT Con	397.695	827.576	320.301
RRT	362.783	927.614	319.066
Lazy RRT	353.775	859.333	326.161
EST	351.373	710.749	306.724

Solutions smoothness:

KPIECE	0.00614645	0.81684403	0.00035364
BKPIECE	0.00649233	0.57762842	0.00061215
LBKPIECE	0.01556398	0.33841250	0.00134936
PRM	0.01087982	0.06572647	0.00584544
SBL	0.03920195	0.01541525	0.00114600
RRT Con	0.02972639	0.00798675	0.00136550
RRT	0.02104365	0.04835968	0.00076772
Lazy RRT	0.01041575	0.00712410	0.00015978
EST	0.02201124	0.12016734	0.01540509

1->2	2->3	3->1	
○	○	○	KPIECE
○	○	○	BKPIECE
○	○	○	LBKPIECE
○	○	○	PRM
●	○	○	SBL
○	○	○	RRT Con.
○	○	○	RRT
○	○	○	Lazy RRT
○	●	●	EST

● Length ○ Smoothness ○ LS

Coordinates | Rotations angles

	1	2	3
X	349.00	349.99	70.000
Y	150.00	53.000	90.000
Z	-350.0	-70.00	-310.0

● Show solution
○ Show trajectory

Figure 8. The general results of *LSTr* software operation according to algorithms A_1–A_9

Conclusions. As follows from the above mentioned, the developed software *LSTr* allows automatic solving the problem of planning the smooth (collision-free) movement of MMD with OM and/or without it within the pre-formed scenes. Different values of the criteria of length and/or smoothness allow the combined use of different algorithms on different sections of trajectories.

In general, the content of the material presented in this research indicates its **scientific novelty**, which can be formulated as follows: further developed an approach to planning optimal trajectories on the criteria of length and / or smoothness, based on the use of primary existing algorithms for collision-free movement of moving objects in the form of MMD taking into account the simulated scene, structural and geometrical parameters of MMD and OM, the work of which involves determining the local criteria of length and smoothness and allows to further solve the direct and inverse problems of kinematics.

The performed developments and the obtained results are considered as a **practical** basis for further creating the approach to planning the trajectories of PM and/or MMD with solving direct and inverse problems of kinematics, and further direct and inverse problems of dynamics, which are the basis for developing control laws ofMS PM, and hence its Gr.

Referenses

1. IFR Press Conference 24th September 2020 Frankfurt. URL: https://ifr.org/downloads/press2018/Presentation_WR_2020.pdf (accessed 10.05.2021).
2. Kyrylovych V. A. Systemnyj pidhid do robotyzovanyx mehanoskladal`nyh tehnologij yak ob'yekta syntezu. Sborny` ktrudov XIX mezhdunarodnoj nauchno-tehnycheskoj konferencyy` "Mashy`nostroenie i` texnosfera XXI veka". Doneczk. 2012. Vol. 2. P. 38–39. [In Ukrainian].
3. Ioan A. Şucan. Kinodynamic Motion Planning by Interior-Exterior Cell Exploration. Lydia E. Kavraki–International Workshop on the Algorithmic Foundations of Robotics. URL: https://www.springerprofessional.de/en/kinodynamic-motion-planning-by-interior-exterior-cell-exploratio/3074572 (accessed 21.05.2021).
4. Ioan A. Sucan. Task and Motion Planning for Mobile Manipulators. Diss.doct. of Ph. Rice University Houston, Texas, August 2011, 153 p.
5. Open Motion Planning Library. A Primer,Kavraki Lab, Rice University, January 22, 2013, p. 25. URL: http://ompl.kavrakilab.org/OMPL_Primer.pdf (accessed 30.05.2021).
6. Gildardo Sanchez, Jean – Claudel Latombe. A Single-Query Bi-Directional Probabilistic Roadmap Planner with Lazy Collision Checking. URL: https://www.researchgate.net/publication/2473120_A_ Single-Query_Biirectional_Probabilistic_Roadmap_Planner_with_Lazy_Collision_Checking (accessed 12.01.2021).
7. Ricardo Sisnett, Gildardo Sánchez. Intelligent Motion Planning for Virtual Characters. URL: http://sisnett.tesisinteractive.com/PosterSGWS.pdf (accessed 25.05.2021).
8. RRT-Connect‒ neoptimalnoe planirovanie traektorii robota. URL: http://robot-develop.org/archives/3835 (accessed 22.04.2021). [In Russian].
9. Howie Choset, James Kuffner. Robotic Motion Planning. URL: https://www.google.com/search?q=9.+Robotic+Motion+Planning+%3A+Howie+Choset%2C+James+Kuffner&rlz=1C1PRFC_ukUA896UA896&oq=9.%09Robotic+Motion+Planning+%3A+Howie+Choset%2C+James+Kuffner&aqs=chrome..69i57.13477j0j4&sourceid=chrome&ie=UTF-8 (accessed 22.05.2021).
10. Eric Heiden, Luigi Palmieri, Kai O. Arras, Gaurav S. Sukhatme and Sven Koenig. Experimental Comparison of Global Motion Planning Algorithms for Wheeled Mobile Robots. Available URL: https://arxiv.org/pdf/2003.03543.pdf (accessed 20.05.2021).
11. Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, Sebastian Thrun. Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents).The MIT Press., 2005, 550 p (accessed 15.05.2021).
12. Mykhailyshyn R. I., Prots' Ya. I., Savkiv V. B. Optimization of bernoulli gripping device's orientation under the process of manipulations along direct trajectory. Scientific Journal of TNTU (Tern.), 2016. Vol. 81. No. 1. P. 107–117.

13. Tao Cheng, Uday Mantripragada, Jochen Teizer, Patricio A. Vela. (2012). Automated Trajectory and Path Planning Analysis Based on Ultra Wideb and Data. URL: https://www.researchgate.net/publication/264626619_Automated_Trajectory_and_Path_Planning_Analysis_Based_on_Ultra_Wideband_Data (accessed19.05.2021).
14. Duchoň, F., Huňady, D., Dekan, M., &Babinec, A. (2013). Optimal navigation for mobile robot in known environment. Applied Mechanics and Materials. Vol. 282. P. 33–39. DOI: https://doi.org/10.4028/www.scientific.net/AMM.282.33
15. Kohrt C., Stamp R., Pipe A. G., Kiely J., & Schiedermeier G. (2013). An online robot trajectory planning and programming support system for industrial use. Roboticsand Computer-Integrated Manufacturing. Vol. 29 (1). P. 71–79. DOI: https://doi.org/10.1016/j.rcim.2012.07.010
16. Palamar M., Poikhalo A., Strembitskyi M., Strembitskyi V. (2016) Method of constructing the navigation system of autonomousmobile robot susing fuzzy logic elements. Scientific Journal of TNTU (Tern.). No. 4 (84). P. 93–98.
17. Sapietová A., Saga M., Kuric I., & Václav Š. (2018). Application of optimization algorithms for robot systems designing. International journal of advanced robotic systems. 15 (1). 1729881417754152. URL: https://journals.sagepub.com/doi/pdf/10.1177/1729881417754152 (accessed 15.05.2021). DOI: https://doi.org/10.1177/1729881417754152
18. Carabin, G., Wehrle, E., &Vidoni, R. (2017). A review on energy-saving optimization methods for robotic and automatic systems. Robotics. 6 (4). 39. URL: https://www.mdpi.com/2218-6581/6/4/39/htm (accessed 15.05.2021). DOI: https://doi.org/10.3390/robotics6040039
19. Boscariol, P., &Richiedei, D. (2019). Trajectory design for energy savings in redundant robotic cells. Robotics. 8 (1). 15. URL: https://www.mdpi.com/2218-6581/8/1/15/htm (accessed 17.05.2021). DOI: https://doi.org/10.3390/robotics8010015
20. Fang Y., Hu J., Liu W., Shao Q., Qi J., & Peng Y. (2019). Smooth and time-optimal S-curve trajectory planning for automated robots and machines. Mechanismand Machine Theory. 137. P. 127–153. DOI: https://doi.org/10.1016/j.mechmachtheory.2019.03.019

Список використаної літератури

1. IFR Press Conference 24th September 2020 Frankfurt. URL: //https://ifr.org/downloads /press2018/Presentation_WR_2020.pdf (дата звернення 10.05.2021).
2. Кирилович В. А. Системний підхід до роботизованих механоскладальних технологій як об’єкта синтезу. Сборник трудов XIX международной научно-технической конференции «Машиностроение и техносфера XXI века». 2012. Донецк. Т. 2. С. 38–39.
3. Ioan A. Şucan. Kinodynamic Motion Planning by Interior-Exterior Cell Exploration / Lydia E. Kavraki– International Workshop on the Algorithmic Foundations of Robotics. URL: https://www.springerprofessional.de/en/kinodynamic-motion-planning-by-interior-exterior-cell-exploratio/3074572 (дата звернення 21.05.2021).
4. Ioan A. Sucan. Task and Motion Planning for Mobile Manipulators: dissertation – Doctor of Philosophy, Rice University Houston, Texas, August 2011.153 p.
5. Open Motion Planning Library: A Primer.Kavraki Lab, Rice University. January 22. 2013. P. 25. URL: http://ompl.kavrakilab.org/OMPL_Primer.pdf (дата звернення 30.05.2021).
6. Gildardo Sanchez, Jean – Claudel Latombe. A Single-Query Bi-Directional Probabilistic Roadmap Planner with Lazy Collision Checking: URL: https://www.researchgate.net/publication/2473120_A_Single-Query_Bi-irectional_Probabilistic_Roadmap_Planner_with_Lazy_Collision_Checking. (дата звернення 12.01.2021).
7. Ricardo Sisnett, Gildardo Sánchez. Intelligent Motion Planning for Virtual Characters. URL: http://sisnett.tesisinteractive.com/PosterSGWS.pdf(дата звернення 25.05.2021).
8. RRT-Connect– неоптимальное планирование траектории робота. URL: http://robot-develop.org/archives/3835 (дата звернення 22.04.2021). [In Russian].
9. Howie Choset, James Kuffner. Robotic Motion Planning. URL: https://www.google.com/search?q=9.+Robotic+Motion+Planning+%3A+Howie+Choset%2C+James+Kuffner&rlz=1C1PRFC_ukUA896UA896&oq=9.%09Robotic+Motion+Planning+%3A+Howie+Choset%2C+James+Kuffner&aqs=chrome..69i57.13477j0j4&sourceid=chrome&ie=UTF-8 (дата звернення 22.05.2021).
10. Eric Heiden, Luigi Palmieri, Kai O. Arras, Gaurav S. Sukhatme and Sven Koenig.Experimental Comparison of Global Motion Planning Algorithms for Wheeled Mobile Robots. URL: https://arxiv.org/pdf/2003.03543.pdf (дата звернення 20.05.2021).

11. Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, Sebastian Thrun. Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents). The MIT Press. 2005. 550 p.(дата звернення 15.05.2021).
12. Mykhailyshyn R. I., Prots' Ya. I., Savkiv V. B. Optimization of bernoulli gripping device's orientation under the process of manipulations along direct trajectory. ВісникТНТУ. ТНТУ, 2016. Vol. 81. No. 1. P. 107–117.
13. Automated Trajectory and Path Planning Analysis Based on Ultra Wideband Data. Tao Cheng, Uday Mantripragada, Jochen Teizer, Patricio A. Vela. URL: https://www.researchgate.net/ publication/264626619_Automated_Trajectory_and_Path_Planning_Analysis_Based_on_Ultra_Wideband_Data. (дата звернення 19.05.2021).
14. Duchoň F., Huňady D., Dekan M., & Babinec A. Optimal navigation for mobile robot in known environment. Applied Mechanics and Materials. Trans Tech Publications Ltd. 2013. Vol. 282. P. 33–39. DOI: https://doi.org/10.4028/www.scientific.net/AMM.282.33
15. Kohrt C., Stamp R., Pipe A. G., Kiely J., & Schiedermeier G. (2013). An online robot trajectory planning and programming support system for industrial use. Roboticsand Computer-Integrated Manufacturing. 2013. Vol. 29 (1). P. 71–79. DOI: https://doi.org/10.1016/j.rcim.2012.07.010
16. Palamar M., Poikhalo A., Strembitskyi M., Strembitskyi V. (2016) Method of constructing the navigation system of autonomousmobile robot susing fuzzy logic elements. Scientific Journal of TNTU (Tern.). No. 4 (84). P. 93–98.
17. Sapietová A., Saga M., Kuric I., & Václav Š. Application of optimization algorithms for robot systems designing. International journal of advanced robotic systems. 2018. Vol. 15 (1), 1729881417754152. URL: https://journals.sagepub.com/doi/pdf/10.1177/1729881417754152 (дата звернення 15.05.2021). DOI: https://doi.org/10.1177/1729881417754152
18. Carabin G., Wehrle E., & Vidoni R. A review on energy-saving optimization methods for robotic and automatic systems. Robotics. 2017. Vol. 6 (4). 39. URL: https:// www.mdpi.com/2218-6581/6/4/39/htm. (дата звернення 15.05.2021). DOI: https://doi.org/10.3390/robotics6040039
19. Boscariol P., & Richiedei D. Trajectory design for energy savings in redundant robotic cells. Robotics. 2019. Vol. 8 (1). 15. URL: https://www.mdpi.com/2218-6581/8/1/15/htm (дата звернення 17.05.2021). DOI: https://doi.org/10.3390/robotics8010015
20. Fang Y., Hu J., Liu W., Shao Q., Qi J., & Peng Y. (2019). Smooth and time-optimal S-curve trajectory planning for automated robots and machines. Mechanism and Machine Theory. 2019. Vol. 137 P. 127–153. DOI: https://doi.org/10.1016/j.mechmachtheory.2019.03.019

УДК 621.865.8:62.505

АВТОМАТИЗОВАНЕ ПЛАНУВАННЯ ОПТИМАЛЬНИХ ТРАЄКТОРІЙ ПЕРЕМІЩЕННЯ МОБІЛЬНИХ МЕХАТРОННИХ ПРИСТРОЇВ

Валерій Кирилович[1]; Петро Мельничук[1]; Любомир Дімітров[2]; Ілона Крижанівська[1]

[1]Державний університет «Житомирська політехніка», Житомир, Україна
[2]Технічний університет-Софія, Софія, Республіка Болгарія

Резюме. *Висвітлено результати сумісного (комплексного) використання загальнодоступних та найбільш відомих алгоритмів, що мінімізують довжину та/або гладкість реалізованих траєкторій переміщення мехатронного мобільного пристрою (ММП) як певного аналога технологічної роботи зованого комплекту. Аналізовані алгоритми та їх програмна реалізація знаходиться у вільному доступі. Запропоновано структуру інформаційного забезпечення, що передбачає автоматизовану сумісну реалізацію алгоритмів. Результати переміщення ММП за знайденими траєкторіями відтворені на 3-ох технологічно значущих її (траєкторії) ділянках з урахуванням аналізованої сцени. Основними* *вимогами, які при цьому забезпечуються, є безперешкодність (безколізійність), безрозривність та плавність. Для підвищення продуктивності та автоматизованої реалізації аналізованих алгоритмів мовою програмування С++ розроблено програмний продукт (ПП) LSTr. Для його повноцінного функціонування використано графічну бібліотеку Open GL (Open Graphics Library), крос-платформовий інструментарій QT, бібліотеку OMPL (Open Motion Planning Library), бібліотеку Assimp (Open Asset Import Library) та програму Sketch Up. Представлено структурну схему функціонування розробленого ПП LSTr, що демонструє взаємозв'язок вказаних прикладних пакетів програмування. Розроблено інформаційне поле з виділеними ділянками, що функціонально є технологічно значущими при роботі ПР. Для розв'язування тестових задач створено кінцеву множину сцен, кожна з яких розглядається як аналог певної ГВК з її конструктивно-планувальними характеристиками та параметрами. Кожна сцена включає в себе простір стану та параметри перешкод, ММП та ОМ, початкові та кінцеві координати точок переміщення ММП. Робота ПП LSTr проілюстрована для однієї із сцен. При цьому для побудови траєкторії за вибором користувача використовується будь-яка кількість алгоритмів. Отримані результати представляються кількісними значеннями довжини та/або гладкості, а також візуалізацією траєкторії. Після їх (результатів) співставлення приймається рішення щодо їх оптимальності на множин із згаданих вище ділянок. Використання аналізованої множини вказаних алгоритмів на множин і розглянутих ділянок та згенерованих на них траєкторій за отриманими результатами дозволяє диференційовано підходити до використання різних алгоритмів на різних ділянках траєкторій. Визначено наукову новизну роботи та її практичне значення для визначених подальших досліджень.*

Ключові слова: *алгоритм, траєкторія, оптимізація, мобільний мехатронний пристрій, довжина, гладкість.*

https://doi.org/10.33108/visnyk_tntu2021.02.064 *Отримано 11.04.2021*