# Application of Linear Algebra Approaches for Predicting Self-Collisions of Dual-Arm Multi-Link Robot

Vadim Kramar, Aleksey Kabanov, and Vasiliy Alchakov
Sevastopol State University, Sevastopol, Russia
Email: kramarv@mail.ru, kabanov@sevsu.ru

*Abstract*— **The article proposes a method for solving the self-collisions predicting problem of dual-arm multi-link robot. The method is based on the analysis of the distance between the skew vectors. The vectors describe the longitudinal axis of the cylinders - the links of the manipulators. The proposed approach will allow the prediction of possible self-collision of robot's manipulators in online mode. Also, the application of the proposed approach is possible when implementing robot control based on the PbD approach in case of wrong actions of the operator ("teacher"), leading to a self-collision of manipulators. An example of using the presented approach is considered.**

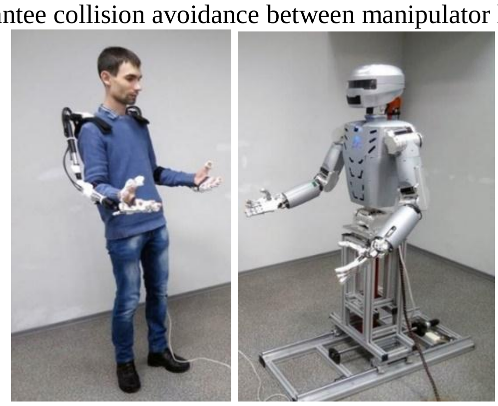
*Index Terms*— **self-collisions, predicting problem, dual-arm robot, multi-link manipulator, linear algebra, robot control**

## I. Introduction

The use of robotic systems has been a trend in recent years. One of the areas of robots' application is associated with the use and development of the ocean's resources. Multi-link manipulators are a species of robotic systems. They are installed on the remotely operated underwater vehicle and on the autonomous underwater vehicle. In connection with the problem of automation of underwater technical works, the problem of the position and orientation control of the manipulator end-effector is solved.

In [1-15], approaches to design of various types of manipulators and methods for their control are considered. The design of modern multi-link manipulators is focused on the use of anthropomorphic structure [1, 6]. One example of such robots is the SAR-401 [16] (Fig. 1). It is the dual-arm robot with manipulators of anthropomorphic type. The dual-arm design leads to the possible collisions of manipulators with each other or with the robot base (body) during the work. Further, we shall call such collisions as self-collisions. Self-collisions are possible in case of incorrect actions by the human-operator or failures in the control system. So, one of the key control problems of a dual-arm robot is to design the control law of the manipulator's motion from the initial state to the target position preventing self-collisions. Although the problem of motion planning of manipulators is relatively simple, but it is computationally complex too [17]. Various algorithms have been developed for motion planning of manipulators (see for example [18, 19]).

Also, approaches to the analysis of manipulators collisions have been developed in [20-23]. For collision avoidance one should use an "occupied-free" approach [20]. According to this approach, a state of the operational area has two options "occupied" and "free". If one of the manipulators is in the operational area, then this area gets the state "occupied" (it is defined as a prohibited zone for the other manipulator). If the occupying manipulator leaves this area, it gets the state "free". In [21] the problem of dual-arm manipulation control with the help of the constraint-based programming is presented. These approaches are mainly based on the analysis of the interaction of only the end-effectors of manipulators. Obviously, they do not guarantee collision avoidance between manipulator links.



a) b)

Figure 1. The robotic system, a) the operator with copy-type setting device; b) the robot SAR-401.

In [22] an approach which is based on the modelling of the manipulator and the objects by simple primitives like "cylinder" or "sphere" is suggested. Using such primitives different types of collisions are introduced. A simple method of detection of such collisions is obtained. The self-collisions detection method for dual-arm robot that is based on the division of operational space into small discrete volume spaces is considered in [24]. It is assumed that the control system generates restriction signals when the motion along the planed path will bring

Manuscript received November 14, 2019; revised September 1, 2020.


doi: 10.18178/ijmerr.9.11.1521-1525

the manipulator into occupied volume spaces. In our opinion, the considered algorithms have a certain restriction for implementation in real-time.

The article proposes an approach of predicting possible self-collisions of dual-arm robot manipulators. This approach is based on methods of linear algebra. This will allow, in our opinion, to make the simple and suitable algorithm for implementation of the self-collision prediction mode of the dual-arm robot in real-time.

## II. The Method of Self-Collisions Prediction for dual-arm robot

We will consider the dual-arm multi-link robot with manipulators of anthropomorphic type.

Let us consider two skew vectors $L_1$ and $L_2$ that define the longitudinal axes of the cylinders (it is used for modeling the links of manipulators). An example of skew vectors is shown in Fig. 2.

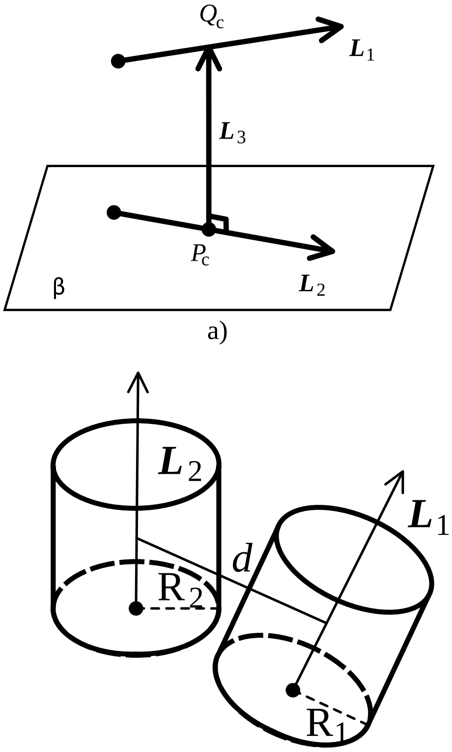



Figure 2. An example of skew vectors, a) skew vectors in the space; b) the cylinders as links of the robot manipulators.

To implement the algorithm for predicting the self-collision of manipulators, we use the relations for the distance between skew vectors $L_1$ and $L_2$. The distance between two skew vectors $L_1$ and $L_2$ is determined by the length of the common perpendicular $L_3$ drawn to these vectors. Let find the indicated common perpendicular.

Let the vector $L_1$ be determined by the canonical equation

$$L_1(t_1) = P_1 + t_1 V_1, \qquad (1)$$

where $P_1 = (x_1, y_1, z_1)$ is the radius vector that defines the starting point of the vector $L_1$, $V_1 = (l_1, m_1, n_1)$ is the direction vector, $0 \le t_1 \le 1$ is a real coefficient.

Similarly, for the vector $L_2$:

$$L_2(t_2) = P_2 + t_2 V_2, \qquad (2)$$

where $P_2 = (x_2, y_2, z_2)$ is the radius vector that defines the starting point of the vector $L_1$, $V_2 = (l_2, m_2, n_2)$ is the direction vector, $0 \le t_2 \le 1$ is a real coefficient.

The vector $L_3$ has the form

$$L_3(t_1, t_2, t_3): \quad P_1 + t_1 V_1 + t_3 V_3 = P_2 + t_2 V_2 \;. \qquad (3)$$

In (3) $V_3$ is defined as a vector product of $V_1$ and $V_2$, i.e.

$$V_3 = V_1 \times V_2 \,. \qquad (4)$$

Transforming (3), we obtain a system of linear algebraic equations in a matrix form

$$t_1 V_1 - t_2 V_2 + t_3 V_3 = P_2 - P_1. \qquad (5)$$

Solving matrix equation (5) we find the intersection points $P_c, Q_c$ between the skew vectors $L_1$, $L_2$ and $L_3$

$$\begin{aligned} P_c &= P_1 + t_1 V_1, \\ Q_c &= P_2 + t_2 V_2. \end{aligned} \qquad (6)$$

The distance $d$ between the intersection points $P_c, Q_c$ is defined as the length of the vector $L_3$. The analytical expression for $d$ is described by the equation

$$d = \frac{\left|((P_2 - P_1) \cdot [V_1 \times V_2]\right|}{\left\|V_1 \times V_2\right\|} \qquad (7)$$

If the condition $d > R_1 + R_2$ is fulfilled (where $R_1$ and $R_2$ are radiuses of the cylinders for left and right manipulator's links correspondingly, Fig. 2b), then these links of the manipulators do not collide.

The collisions check is carried out at the intersection of all links of the left manipulator with all links of the right manipulator.

This approach will make it possible to predict not only a typical collision (body-body collision [22], Fig. 3a), but also a possible collision in the region of the base of the cylinders (base-body and base-base collisions [22], Fig.3b).

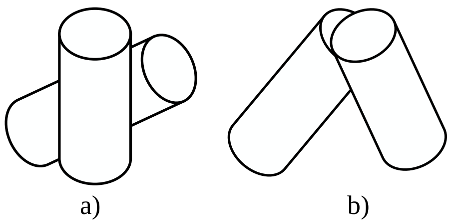



Figure 3. The intersection of cylinders, a) body-body collision; b) base-base collisions.

This feature is especially important for solving motion control problems for a multiple cooperative manipulator system in real practical applications.

## III. Example of Self-Collisions Prediction Approach Implementation

We will consider the underwater dual-arm robot with manipulators of anthropomorphic type. The 3D concept of the robot is shown in Fig. 4. The robot is controlled by the copy-type setting device that is used by the operator (Fig. 1a). It is necessary to predict self-collisions of manipulators that may arise as a result of incorrect operator actions. When self-collisions are predicted, the system must block the operator's incorrect action. It can be mentioned that anthropomorphic robot SAR-401 (Fig. 1b) may be considered as a prototype of this underwater robot due to the fact that the same copy-type setting device is used. Also proposed methods and algorithms were initially worked out on SAR-401.

The manipulator parameters of considered underwater dual-arm robot are given in Table 1. The kinematic model in the Denavit-Hartenberg notation [26, 27] is shown in Table 2.

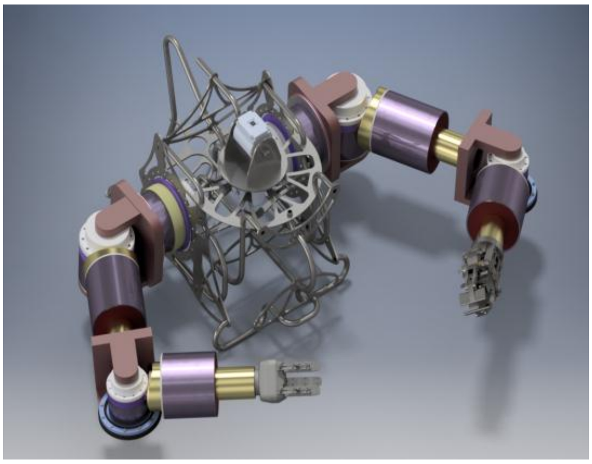


Figure 4. The 3D concept of underwater dual-arm robot with anthropomorphic manipulators.

Based on the proposed method the Matlab program was developed using the freely distributed RoboticsToolbox [28]. The schematic representation of the underwater robot manipulators simulated in Matlab is shown in the Fig. 5.

TABLE I. Parameters of the Manipulators

| Name | Robot manipulators | |
|---|---|---|
| | Left | Right |
| The length of the links, the analog of the human shoulder, mm | 700 | 700 |
| The length of the links, the analog of the human forearm, mm | 700 | 700 |
| Degree of freedom (excluding the end-effector)9 | 6 | 6 |
| Center distance between the left and the right arm, mm | 1000 | |

TABLE II. Parameters of the Manipulators

| Link, i | $A_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | π/2 | 0 | 75° |
| 2 | 0 | π/2 | 0 | 160° |
| 3 | 0 | -π/2 | 0.7 | 0 |
| 4 | 0 | π/2 | 0 | -90° |
| 5 | 0 | -π/2 | 0.5 | 0 |
| 6 | 0.2 | 0 | 0 | -90° |

As an example of the implementation of developed self-collision prediction method we simulate two cases. In case 1 it is assumed that the robot manipulators should collide in the point with coordinates (-0.3678, 0.7666, 0.4184). As a result of the program we get the coordinates of the manipulators self-collision point. The movement that leads to the collision is blocked. The Fig. 6a displays this case.

In case 2 it is assumed that the robot manipulators should not collide. The end-effector of the right manipulator will come to the point with coordinates (-0.3678, 0.7666, 0.4184), and the left manipulator end-effector will come to the point with coordinates (-0.3678, 0.7666, 1.4184). As a result of the program we get the information that manipulators do not collide. The Fig. 6b displays this case.

In the proposed self-collision prediction method, it is possible to take into account the diameter of the cylinders which modeling the manipulator, controlling the approach of the segments to a distance $d > R_1 + R_2$, where $R_1$ and $R_2$ are radiuses of the cylinders.

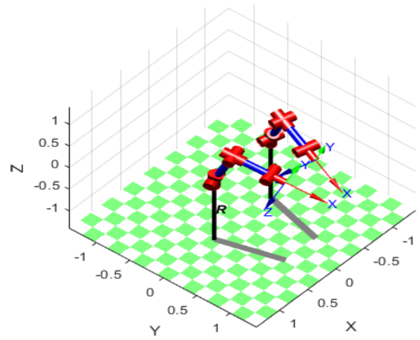
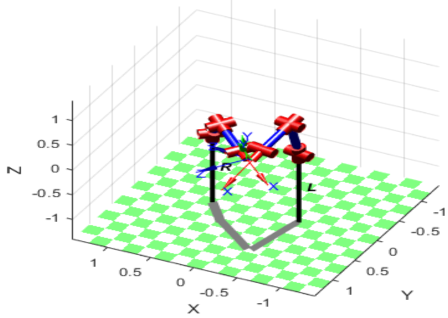
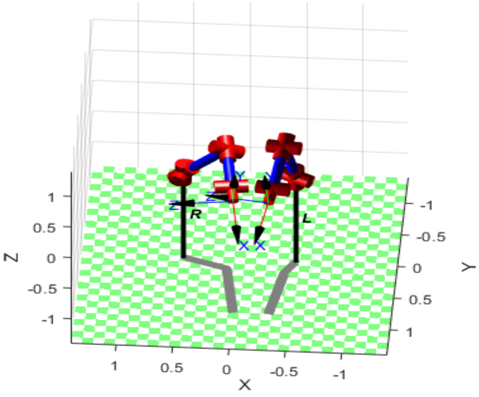



Figure 5. The initial position of the manipulators.




a)




b)

Figure 6. Manipulators collision prediction result, a) case 1; b) case 2.

The proposed method is also tested on an underwater robot simulator (see Fig. 7). As noted earlier, the proposed method of the self-collisions prediction for manipulators will be effectively used in design of the control law for the manipulator's motion path to prevent their collisions.

In addition, using a modern manipulator control method called «Programming by demonstration» will also require the application of the self-collision prediction method proposed in the article. As noted in [29] «Programming by demonstration» (PbD) provides a natural way of programming robots showing the desired task. The PbD method is based on training the robot to perform various operations based on the results of the “teacher". In our case the operator with copy-type setting device can be considered as a “teacher” (see Fig. 1a). Obviously, in the process of training, the “teacher” can make mistakes. For example, due to poor calibration of the copy-type setting device the operator may collide robot manipulators with each other. So, the self-collision prediction algorithm must block this type of actions.

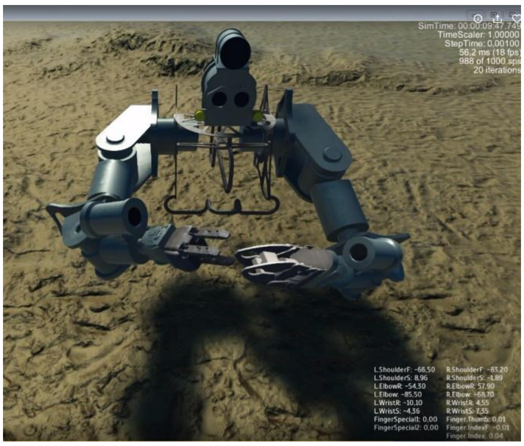


Figure 7. An underwater dual-arm robot simulator.

## IV. CONCLUSION

The article proposes the method to predicting possible self-collisions of dual-arm robot manipulators. This method is based on the analysis of the distance between the skew vectors that modeling the axes of the cylinders. And the links of manipulators are described by these cylinders. Before the control signal is generated, a possible collision point of the manipulators is calculated. If the collision is predicted, then control signal is corrected. The proposed approach allows predicting the self-collision of manipulators both at the stage of path planning and during the generation of control signals. Also, the application of the proposed approach is possible when implementing robot control based on the PbD approach in case of wrong actions of the operator (“teacher”), leading to a self-collision of manipulators.

## CONFLICT OF INTEREST

The authors declare no conflict of interest.

## AUTHOR CONTRIBUTIONS

Vadim Kramar identified point of research, supervised the research and its execution and wrote the paper. Aleksey Kabanov and Vasiliy Alchakov built models and performed numerical experiments, performed validation and editing. All authors contributed to the testing and had approved the final version.

## ACKNOWLEDGMENT

Studies were supported by the Ministry of Science and Higher Education of the Russian Federation (agreement № 14.578.21.0264, project RFMEFI57818X0264).

## REFERENCES

[1] B. Siciliano and O. Khatib, *Robot Learning*, *Handbook of Robotics*, Springer-Verlag Berlin Heidelberg, 2016.

[2] M. Crenganiu, R. Breaz, G. Racz, and O. Bologa, “The inverse kinematics solutions of a 7 DOF robotic arm using fuzzy logic,” in *7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2012, pp.518-523.

[3] M. Shahinpoor, *Robot Engineering Textbook*, Harper & Row Publishers, Inc., New York, 1987.

[4] K. S. Fu, R. C. Gonzalez and C. S. G. Lee, *Robotics: Control, Sensing, Vision and Intelligence*, McGraw-Hill, 1987.

[5] C. Carignan, “Trajectory optimization for kinematically redundant arms,” *Journal of Robotic Systems*, vol. 6, no. 5, 1993, pp. 221-248.

[6] B. Siciliano, O. Khatib, *Robot Programming by Demonstration, Handbook of Robotics*, Springer-Verlag Berlin Heidelberg, 2008.

[7] M. J. Mahmoodabadi and A. Ziaei, “Inverse dynamics based optimal fuzzy controller for a robot manipulator via particle swarm optimization,” *Hindawi Journal of Robotics*, pp.1-10, 2019.

[8] M. Mendili and F. Bouani, “Predictive control of the mobile robot using kinematic and dynamic models,” *Journal of Control Science and Engineering*, pp.1-11, 2017.

[9] M. W. Spong and M. Fujita, “Control in robotics,” *IEEE Control Systems Society*, vol. 10, no. 2, pp.1–25, 2011.

[10] J. J. E. Slotine and W. Li, “On the adaptive control of robot manipulators,” *International Journal of Robotics Research*, vol. 6, no. 3, pp. 147–157, 1987.

[11] R. Ortega and M. W. Spong, “Adaptive control of robot manipulators: A tutorial,” in *Proc. 27th IEEE Conference on Decision and Control*, 1989, pp. 1575–1584.

[12] M. B. Pillai, S. Nakdhamabhorn, K. Borvorntanajanya, and J. Suthakorn, “Enforced acceleration control for DC actuated rescue robot,” in *Proc. XXII International Conference on Electrical Machines*, 2016, pp. 2640–2648.

[13] X. Q. Lv and . Zhao, “Application of improved BQGA in robot kinematics inverse solution,” *Hindawi Journal of Robotics*, pp.1-7, 2019.

[14] M. Zhao and Y. Dai, “Application of fuzzy ant colony algorithm to robotics arm inverse kinematics problem,” *ICIC Express Letters*, vol. 10, no. 1, pp. 43–49, 2016.
[15] X. Li, Y. Guo, J. Zhang, and S. Guo, “Inverse kinematics solution and verification of modular 6-DOF manipulator,” *Transactions of the Chinese Society for Agricultural Machinery*, vol. 44, no. 4, pp. 246–251, 2013.
[16] A. A. Kabanov and A. N. Balabanov, “The modeling of an anthropomorphic robot arm,” *MATEC Web of Conferences*, vol. 224, pp.1-6, 2018.
[17] J. H. Reif, “Complexity of the mover’s problem and generalizations,” *IEEE Symp. Found. Comput. Sci*., pp. 421–427, 1979.
[18] H. H. Gonzalez-Banos, D. Hsu, and J. C. Latombe, *Autonomous Mobile Robots: Sensing, Control, Decision-Making and Applications*, CRC, Boca Raton, 2006.
[19] S. R. Lindemann and S. M. LaValle, “Current issues in sampling-based motion planning,” in *Proc. 11th Int. Symp. Robotics Res*., Berlin, Heidelberg, 2005, pp.36–54.
[20] M. E. Jefferies and W. Yeap, *Robotics and Cognitive Approaches to Spatial Mapping*, New York: Springer-Verlag, 2008.
[21] Y. Wang, P. Ogren, C. Simth, F. Vina, and Y. Karayiannidis, “Dual-arm manipulation using constraint-based programming,” in *Proc. 19th World Congress IFAC*, Cape Town, vol. 19, 2014, pp.311-319.
[22] R. V. Patel, F. Shadpey, F. Ranjbaran, and J. Angeles, “A collision-avoidance scheme for redundant manipulators: Theory and experiments,” *Journal of Robotic Systems*, vol. 22, no. 12, pp. 737–57, 2005.
[23] A. Y Afaghani and Y. Aiyama, “On-line collision avoidance between two robot manipulators using collision map and simple escaping method,” in *Proc. IEEE/SICE Int. Symposium on System Integration*, Kobe, 2013, pp. 105-110.
[24] A. A. Kabanov, D. A. Tokarev, “Collision detection and avoidance method for two cooperative robot manipulators,” *IOP Conference Series: Materials Science and Engineering*, vol. 709, 2020.
[25] G. Zhidenko, I. M. Kutlubayev, A. A. Bogdanov, and V. B. Sychkov, “Basis of structural scheme selection of space application robots,” *Reshetnev’s Readings*, pp. 278-280, 2017.
[26] J. Denavit and R. S. Hartenberg, “A kinematic notation for lower-pair mechanisms based on matrices,” *Transactions ASME Journal of Applied Mechanics*, vol. 23, pp. 215–221, 1955.
[27] P. I. Corke, “A simple and systematic approach to assigning Denavit-Hartenberg parameters,” *IEEE Transactions on Robotics*, vol. 23, pp. 590–594, 2007.
[28] P. I. Corke, *Robotics, Vision and Control Fundamental Algorithms in Matlab*, Springer International Publishing AG, 2017.
[29] V. Alchakov, V. Kramar, and A. Larionenko, “Basic approach to programming by demonstration for an anthropomorphic robot,” *IOP Conf. Series: Materials Science and Engineering*, vol. 709, pp. 1-8, 2020.



**Vadim Kramar** has been inaugurated in the field of Technical cybenetics. He is a full professor at Sevastopol state university at the Department of informatics and control systems. His research interests are focused mainly on the robust control, process modeling, and simulations of complex robotic systems. He has been a leading investigator of a number of successfully completed scientific projects. He actively publishes the results of his activities in many international journals and at international conferences. He is a member of IEEE..

**Aleksey Kabanov** received his Ph. D. degree in Systems and Processes of Control from the Sevastopol National Technical University, Sevastopol, Ukraine in 2012. He is a director of at the Department of informatics and control systems. His current research interests include robotics, mobile object control, singular perturbation theory and methods in control, robust and adaptive control. He is a member of IEEE.

**Vasiliy Alchakov** received the Ph. D. degree in Systems and Processes of Control from the Sevastopol National Technical University, Sevastopol, Ukraine in 2013. His research interests include data processing, neural networks, terminal and adaptive control. He is a member of IEEE.