ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 174

Proceedings of the 8th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2020)

Machine Learning Algorithm for Anthropomorphic Manipulator Control System

Vyacheslav Petrenko
Institute of Information Technologies
and Telecommunications
North-Caucasus Federal University
Stavropol, Russian Federation
vip.petrenko@gmail.com

Fariza Tebueva*
Institute of Information Technologies
and Telecommunications
North-Caucasus Federal University
Stavropol, Russian Federation
fariza.teb@gmail.com

Andrey Pavlov
Institute of Information Technologies
and Telecommunications
North-Caucasus Federal University
Stavropol, Russian Federation
losde5530@gmail.com

Nikolay Svistunov
Institute of Information Technologies
and Telecommunications
North-Caucasus Federal University
Stavropol, Russian Federation
svistunovn4@gmail. com

***Abstract*—Service robots are one of the relevant areas of modern robotics. Many service robots are equipped with a pair of anthropomorphic manipulators, so that they are able to perform complex operations. However, this approach leads to new challenges in development of the robot control systems. In this paper we propose an algorithm for training the control system of two anthropomorphic manipulators with 7 degrees of mobility having intersecting work areas. The algorithm is based on deep reinforcement learning approach applied to the artificial neural network (ANN). The paper also describes the practical implementation of the ANN-based manipulator control system that avoids collisions and achieves an average accuracy of reproducing target positions of manipulator end effector of 98.3%. The ANN training was carried out using Keras framework. The obtained results indicate the promise of applying the proposed method for the development of control systems for anthropomorphic manipulators based on deep reinforcement learning.**

Keywords—anthropomorphic manipulator, forward kinematics, artificial neural network, machine learning, deep reinforcement learning

I. INTRODUCTION

One of the relevant areas of modern robotics are service robots. Service robots are used to perform operations both in extreme conditions [1-3] and in everyday life: for cooking [4], cleaning [5] in catering establishments [6], [7], hotel business [8], banks and other services [9].

To perform operations with environmental objects, many service robots are equipped with one [10] or two anthropomorphic manipulators [11]. Performing targeted operations with manipulators requires solving problems of motion planning, kinematics and dynamics. Movement planning consists in searching for such a trajectory of the manipulator's grip, which on the one hand would ensure the performance of the target operation, on the other hand, bypassing the obstacle and the absence of collision of the manipulator links with each other. The problem of avoiding obstacles in the joint execution of target operations simultaneously by two manipulators is supplemented by the need to avoid collisions of links of one manipulator with links of the second manipulator. The advantage of anthropomorphic manipulators is kinematic redundancy, which provides advanced opportunities for avoiding obstacles and links of the second manipulator.

Joint movements of manipulators are of two types. The first type is the movement of manipulators holding the manipulated object together. For the first type of movement, the trajectory along which the manipulation object should move is often known. There are a large number of methods for planning this type of movement [12-18]. The disadvantage of analytical methods for solving this problem [19] is the specialization in a strictly defined kinematic scheme of the anthropomorphic manipulator, as well as low energy efficiency. From the possible paths to bypass the obstacles of the second manipulator, one is selected at which the distance between the manipulators will be the largest.

In this paper, we propose a solution to the problem of controlling two anthropomorphic manipulators based on an artificial neural network (ANN) and deep reinforcement learning. The use of ANN is due to neural networks showed high efficiency in solving problems that are difficult to solve analytically [20-21]. To achieve the result, we developed a training algorithm and structure of the control system for anthropomorphic manipulators.

II. THE PROPOSED ALGORITHM

The control object is a combination of two anthropomorphic manipulators that perform target operations together. The kinematic diagram of one of the manipulators under consideration is shown in Figure 1, coordinate systems associated with joints – in Figure 2; Denavit-Hartenberg parameters – in table 1, where θ_i is the angle by which the x_{i-1} axis needs to be rotated around the z_{i-1} axis so that it becomes aligned with the x_i axis (the sign is determined by the rule of the right hand); α_i - the angle by which you want to rotate the z_{i-1} axis around the x_i axis so that it becomes co-directional with the z_i axis; a_i is the distance between the intersection of the z_{i-1} axis with the x_i axis and the beginning of the i-th coordinate system, counted along the x_i axis, i.e. the shortest distance between the axes z_{i-1} and z_i; d_i is the distance between the intersection of the z_{i-1} axis with the x_i axis and the beginning of the $(i-1)$-th coordinate system counted along the z_{i-1}axis.

Table 1 – Denavit-Hartenberg parameters

Joint	θ_i	α_i	a_i	d_i	Range of θ_i
1	180°	90°	0	0	−90° … 90°
2	−90°	90°	0	0	−20° … 90°
3	−90°	90°	0	−30	−45° … 45°
4	−90°	90°	0	0	−70° … 90°
5	−90°	90°	0	25	−90° … 90°
6	−90°	90°	0	0	−45° … 45°
7	0°	0°	−10	0	−70° … 70°

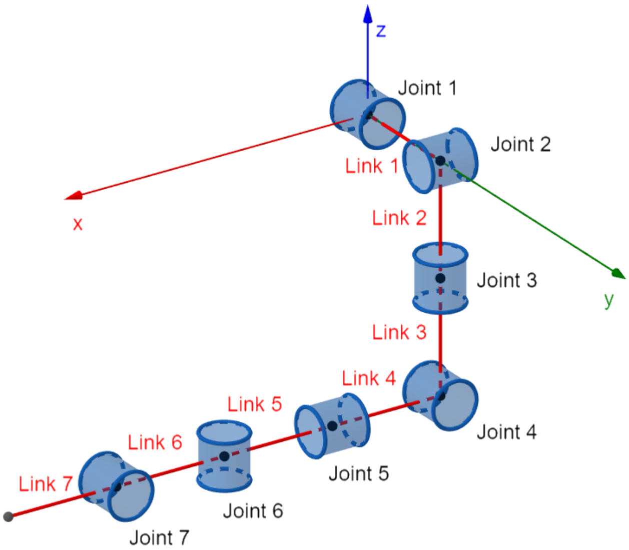

Figure 1 – Kinematic diagram of the anthropomorphic manipulator

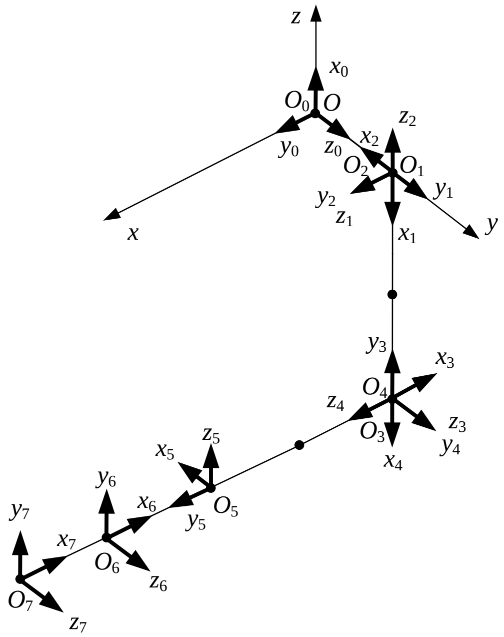

Figure 2 – Coordinate systems associated with the links of the anthropomorphic manipulator

The direct kinematics problem of this manipulator can be solved using the following formulas:

$${}^{i}\boldsymbol{T}_j = \prod_{k=i+1}^{j} {}^{i-1}\boldsymbol{A}_i\,, i < j,$$

$${}^{i-1}\boldsymbol{A}_i = \boldsymbol{T}_{z,\theta}(\theta_i)\boldsymbol{T}_{z,d}(d_i)\boldsymbol{T}_{x.a}(a_i)\boldsymbol{T}_{x,\alpha}(\alpha_i),$$

$$\boldsymbol{T}_{z,\theta}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{T}_{z,d}(d) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{T}_{x.a}(a) = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{T}_{x,\alpha}(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where ${}^{i}\boldsymbol{T}_j$ is the homogeneous transformation matrix from the i-th to the j-th coordinate system, compiled in accordance with the Denavit-Hartenberg representation;

${}^{i-1}\boldsymbol{A}_i$ is a homogeneous complex transformation matrix for adjacent coordinate systems;

$\boldsymbol{T}_{z,\theta}(\theta)$ is a homogeneous matrix of elementary rotation about the z axis by an angle θ;

$\boldsymbol{T}_{z,d}(d)$ is a homogeneous matrix of elementary shift along the z axis by a distance d;

$\boldsymbol{T}_{x.a}(a)$ is a homogeneous matrix of elementary shift along the x axis by a distance a;

$\boldsymbol{T}_{x,\alpha}(\alpha)$ is a homogeneous matrix of elementary rotation about the x axis by the angle α.

Thus, the position of the grip of the manipulator is determined by the matrix ${}^{0}\boldsymbol{T}_7$, the fourth column of which contains the homogeneous coordinates of the end effector.

Control signals for anthropomorphic manipulators are generalized coordinates

$$q(t) = \langle q_1(t), q_2(t)\rangle,$$

where $q_1(t)$, $q_2(t)$ are generalized coordinates of the first and second manipulator at time t, respectively. The goal of controlling a pair of manipulators is to refine the necessary trajectories of end effectors in their configuration space:

$$l(t) = \langle l_1(t), l_2(t)\rangle,$$

where $l_i(t) = \langle p_i(t), \phi_i(t)\rangle$ is end effector trajectory of the i-th manipulator in the Cartesian space p_i and the space of Euler angles ϕ_i.

Further in the work, the combined values are used:

$$p(t) = \langle p_i(t) \rangle_{i=1,2},$$

$$\phi(t) = \langle \phi_i(t) \rangle_{i=1,2},$$

$$l(t) = \langle p(t), \phi(t) \rangle.$$

The process of controlling two anthropomorphic manipulators under conditions of complete observability of the environment can be considered as the Markov decision-making process (MDMP), which can be described by a tuple of five elements, $\langle S, A, R, P, \rho_0 \rangle$, where:

S is the set of all possible states of the medium;

A is a set of all possible actions;

$R: S \times A \times S \to \mathbb{R}$ is the reward function when, at time t, in state s_t, action a_t is executed, resulting in state s_{t+1}:

$$r_t = R(s_t, a_t, s_{t+1});$$

$P: S \times A \to \mathcal{P}(S)$ is the transition probability function, where $P(s_{t+1} \mid s_t, a_t)$ is the probability of transition to the state s_{t+1} from the state s_t when the action a_t is performed;

ρ_0 is the probability distribution of the initial state s_0.

The architecture of the reinforcement learning system applied to the described task is shown in Figure 3.

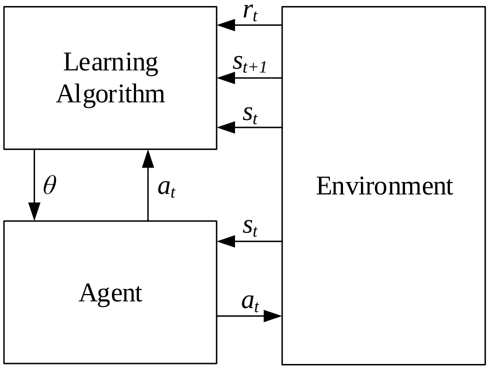

Figure 3 – The structure of the learning environment

The control system of anthropomorphic manipulators is considered as an agent operating in a simulation environment. The agent is able to obtain the current state s_t of the environment at time t, on the basis of which the action a_t is taken based on some decision-making policy μ:

$$a_t = \mu(s_t).$$

The simulation environment informs the agent with the current state s_t , perceives the selected action a_t and simulates the consequences of the action a_t, which leads to the transition of the medium to state s_{t+1}. State s_t includes a set of current generalized coordinates of anthropomorphic manipulators q_t and a point l_{t+1} of the trajectory l, which must be reached at time $t + 1$:

$$s_t = \langle q_t, l_{t+1} \rangle.$$

The action a_t corresponds to the vector q_{t+1} of generalized coordinates of the manipulators that must be reached at time $t + 1$:

$$a_t \equiv q_{t+1}.$$

Also, the simulation environment determines the reward r_t received by the agent. The remuneration consists of the following values:

1) The cost of displacement proportional to the modulus of change of generalized coordinates during the transition from the position q_t to the position q_{t+1}.

2) Penalty for the collision of manipulators. The collision of manipulators is considered to have occurred if the manipulators come closer to a distance less than a certain threshold value D. To calculate the values of the distances between the links of the manipulators, the modified method proposed in [22] was used.

3) The accuracy of positioning the grip of the manipulator, expressed as the distance between the coordinates of the target point and the actual coordinates of the grip.

In this paper we are mainly focused on factors 2 and 3.

The learning algorithm is designed to optimize decision-making policy μ in order to maximize the amount of remuneration received by the agent. The agent's decision policy is modeled using an artificial neural network. The input signals of the ANN are: the current position of the manipulators q_t and the point l_{t+1} of the trajectory l, which must be reached at time $t + 1$. The task of the ANN is to search for such action a, which will have the greatest value.

The function $Q^\mu\,(s_t, a_t)$ of the value of performing the action a_t at the state of the medium s_t at time t in accordance with the policy μ is based on the Bellman equation:

$$Q^\mu\,(s_t, a_t) = \underset{a_{t+1}}{\operatorname{argmax}}[r_t(s_t, a_t) + \gamma Q^\mu(s_{t+1}, a_{t+1})],$$

where $\underset{x}{\operatorname{argmax}}[f(x)]$ is the maximization argument, i.e. the value of x at which $f(x)$ reaches its maximum value;

$r_t(s_t, a_t)$ is the amount of remuneration for performing the action a_t with the environment s_t at time t;

$\gamma \in [0; 1]$ is training discounting factor. The smaller it is, the less the agent takes into account the benefits of its future actions.

The value of $Q^\mu\,(s_t, a_t)$ is determined by the set of ANN weights. The goal of machine learning with agent reinforcement is to optimize weights so that the decision policy μ strives for the optimal μ^*:

$$\mu(s_t) \to \mu^*(s_t).$$

Thus, the machine learning algorithm of the control system for anthropomorphic manipulators consists of the following steps:

1) Choosing the values of constants that determine the parameters of the learning process:

– sizes of data sets for training L and validation V of an artificial neural network (a larger sample size implies achieving greater accuracy, but reduces the speed of learning));

– the number of learning iterations I;

– the values of the coefficients of the reward function, which determine the degree of influence of the cost of movement, the penalty for a collision and the accuracy of positioning of end effectors;

– minimum permissible distance D between manipulators.

The coefficients of the reward function and the number of learning iterations are being selected empirically.

2) Preparation of $L + V$ pairs of the form $((C_1, C_2), (A_1, A_2))$, where C_1, C_2 are the grip coordinates of the first and second manipulator, respectively, A_1, A_2 are the arrays of rotation angles θ_i of the joints of the first and the second manipulator, respectively. Each pair is produced as follows:

2.1) Filling arrays A_1, A_2 using a pseudo-random number generator with uniform distribution. The range of permissible values of θ_i specified in Table 1 is used as the generation range for the i-th angle.

2.2) Calculation the values C_1, C_2 by solving the direct kinematics problem for the first and second anthropomorphic manipulator based on the arrays of angles A_1, A_2, respectively.

3) Carrying out I reinforcement learning iterations with on a data set of L elements. At each iteration, the following actions are being performed:

3.1) Sending the coordinates of end effectors to the input of the ANN.

3.2) Obtaining output values of the angles of the manipulators.

3.3) Solving the direct kinematics problem based on the output values of the angles for determining the coordinates of joints and end effectors of manipulators.

3.4) Collision verification.

3.5) Calculating the distance between the reference positions of the grips and the positions calculated based on the angles generated by the ANN.

3.6) Determination of the amount of remuneration.

3.7) Correction of the weights of the ANN neurons.

4) Testing the trained model by repeating the steps 3.1 – 3.6 on the validation data set.

5) Saving the trained ANN.

We proposed to use a control system free of an environmental model, since this approach will avoid the problem of constructing an accurate dynamic model of the anthropomorphic manipulators. The accurate model is not needed as long as inertia influence is negligible. This is the subject of further research.

III. Results

To test the proposed algorithm, a simulator of the simultaneous operation of two manipulators with the ability to visualize their position was developed in Python (Figure 4). The blue color shows the position of the links of the manipulators according to the initial data, the magenta color – the position generated by the ANN.

The simulation was carried out in accordance with the following conditions: the position of the target was randomly selected in three-dimensional space; the manipulator was controlled by transferring final coordinates to it. The purpose of the machine learning algorithm is to achieve the given points of space with the grips of the manipulators without the collision of the manipulators.

Machine learning was performed using the Keras neural network library [23]. As a model ANN, a direct distribution neural network was used (Figure 5). It has 6 neurons with a linear activation function in the input layer, 14 neurons with a linear activation function in the output layer, and 6 hidden layers containing 1024 neurons each. The activation function of the hidden layers is rectified linear unit (ReLU). We conducted 25 epochs of training. As training data, 1 million pairs “angles – coordinates” were used. 100 thousand such pairs were used for validation. To speed up the training process, we set the batch size to 1000.

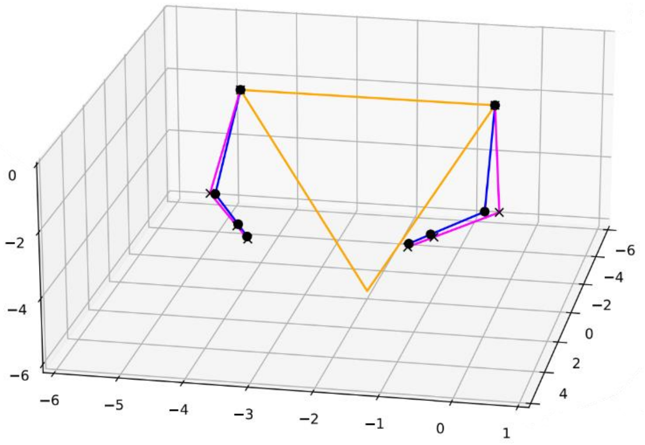

Figure 4 – Simulation of the collaboration of two manipulators

As a loss function, the average value of the sum of the distances between the coordinates of the end effectors of the manipulators and the corresponding target points, expressed in hundredths of a millimeter, was used. For each generated position, a check was made for the collision of the manipulators, as well as for the exit of the angle values beyond the permissible limits. In such situations, a fine of 1,000 units was added to the loss value. The change in the value of the loss function during the passage of 25 epochs of training is presented in Figure 6. After passing through 9 epochs, the generated datasets ceased to contain arrays of angles leading to incorrect positions and collisions.

Figure 7 shows the change of the accuracy metric. This metric was calculated as the average value of the scalar products of the normalized radius vectors of end effectors and normalized vectors corresponding to the reference positions from training and validation datasets.

The achieved accuracy values indicate sufficient training of the model, as well as the lack of overtraining. The resulting average accuracy is 98.3%.

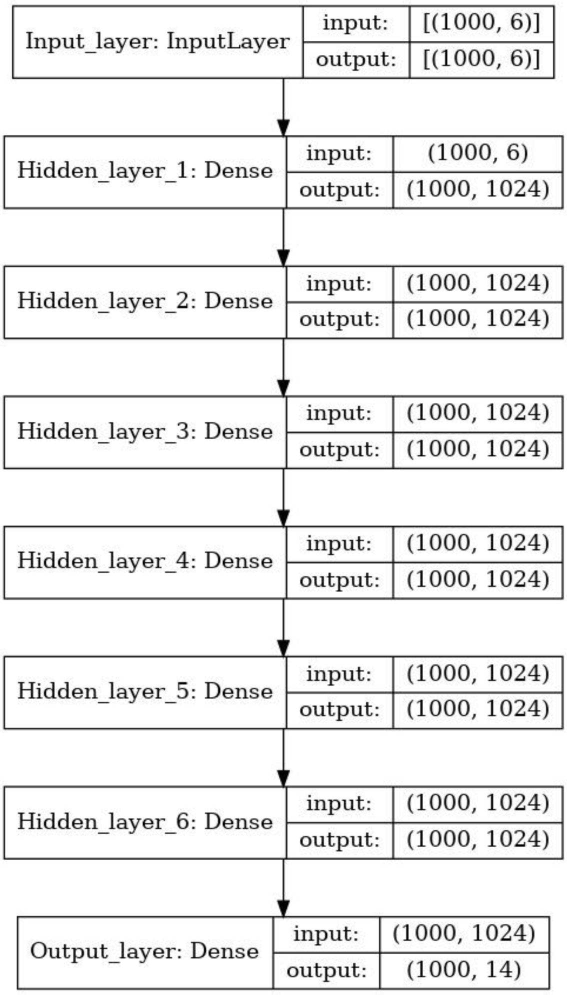

Figure 5 – Structure of the trained ANN

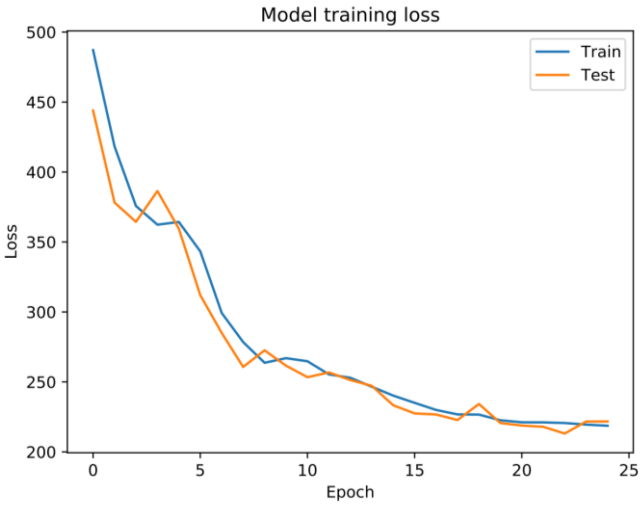

Figure 6 – Loss values during the ANN training

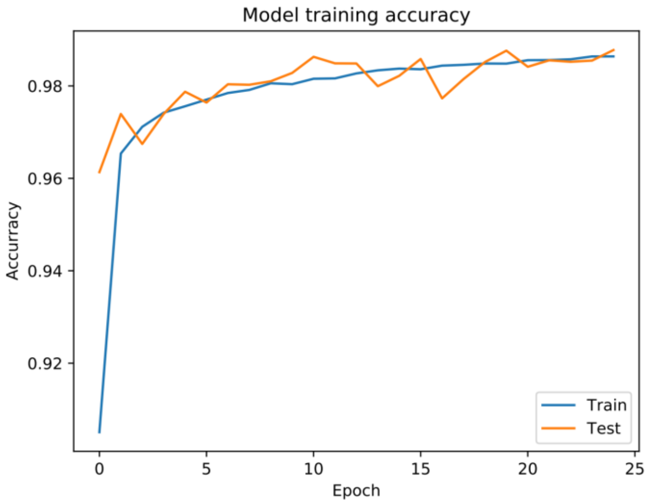

Figure 7 – The accuracy of achieving the target positions of the end effectors

Thus, the results of the implementation of the proposed method show its ability to control dual manipulator systems with sufficient accuracy.

IV. Conclusion

Using the proposed algorithm, we trained the ANN as part of a control system for two anthropomorphic manipulators. During the movement of anthropomorphic manipulators, there were no collisions between the links. The obtained results indicate the promise of applying the proposed method for the development of control systems for anthropomorphic manipulators based on deep reinforcement learning. The directions of further development are: complicating the task by introducing additional static and then dynamic obstacles into the operating environment; adaptation of the proposed algorithm to optimize the movement of anthropomorphic manipulators according to the criteria of energy efficiency, efforts and time of movement by taking into account the dynamic model of manipulators in the simulation environment.

References

[1] O. Saprykin, E. Baksheeva, V. Safronov, and O. Tolstel, “About the concept of using anthropomorphic robots during human exploration of the Moon,” *ARPN J. Eng. Appl. Sci.*, vol. 11, no. 16, pp. 9674–9679, Aug. 2016.

[2] I. M. Kutlubaev, A. A. Bogdanov, N. V. Novoseltsev, M. V. Krasnobaev, and O. A. Saprykin, “Control system of the anthropomorphous robot for work on the low-altitude earth orbit,” *Int. J. Pharm. Technol.*, vol. 8, no. 3, pp. 18193–18199, Sep. 2016.

[3] I. M. Kutlubaev, I. G. Zhydenko, and A. A. Bogdanov, “Basic concepts of power anthropomorphic grippers construction and calculation,” in *2016 2nd International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2016 - Proceedings*, 2016, doi: 10.1109/ICIEAM.2016.7910963.

[4] K. Junge, J. Hughes, T. G. Thuruthel, and F. Iida, “Improving Robotic Cooking Using Batch Bayesian Optimization,” *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 760–765, Apr. 2020, doi: 10.1109/LRA.2020.2965418.

[5] iRobot Vacuum Cleaning, Mopping & Outdoor Maintenance [Electronic resource]. URL: https://www.shopirobot.com.au/ [Accessed: 24-Mar-2020].

[6] D. K. Limbu *et al.*, "Experiences with a barista robot, FusionBot," in *Communications in Computer and Information Science*, 2009, vol. 44 CCIS, pp. 140–151, doi: 10.1007/978-3-642-03986-7_17.

[7] S. Hedaoo, A. Williams, C. Wadgaonkar, and H. Knight, "A Robot Barista Comments on its Clients: Social Attitudes Toward Robot Data Use," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2019, vol. 2019-March, pp. 66–74, doi: 10.1109/HRI.2019.8673021.

[8] R. de Kervenoael, R. Hasan, A. Schwob, and E. Goh, "Leveraging human-robot interaction in hospitality services: Incorporating the role of perceived value, empathy, and information sharing into visitors' intentions to use social robots," *Tour. Manag.*, vol. 78, Jun. 2020, doi: 10.1016/j.tourman.2019.104042.

[9] Promobot V.4 | PROMOBOT [Electronic resource]. URL: https://promo-bot.ru/production/promobot-v4/ [Accessed: 24-Mar-2020] (in Russian).

[10] M. Gonbata, F. Leonardi, and P. Aquino Junior, "Robotic Manipulators Mechanical Project For The Domestic Robot HERA," *II Brazilian Humanoid Robot Work. III Brazilian Work. Serv. Robot.*, pp. 30–35, 2019.

[11] Baxter | Redefining Robotics and Manufacturing | Rethink Robotics [Electronic resource]. URL: https://web.archive.org/web/20140826071530/http://www.rethinkrobotics.com/products/baxter/ [Accessed: 24-Mar-2020].

[12] A. Kimmel, R. Shome, Z. Littlefield, and K. Bekris, "Fast, Anytime Motion Planning for Prehensile Manipulation in Clutter," in *IEEE-RAS International Conference on Humanoid Robots*, 2019, vol. 2018-Novem, pp. 874–880, doi: 10.1109/HUMANOIDS.2018.8624939.

[13] J. A. Haustein, K. Hang, J. Stork, and D. Kragic, "Object Placement Planning and optimization for Robot Manipulators," in *IEEE International Conference on Intelligent Robots and Systems*, 2019, pp. 7417–7424, doi: 10.1109/IROS40897.2019.8967732.

[14] T. McMahon, R. Sandstrom, S. Thomas, and N. M. Amato, "Manipulation planning with directed reachable volumes," in *IEEE International Conference on Intelligent Robots and Systems*, 2017, vol. 2017-Septe, pp. 4026–4033, doi: 10.1109/IROS.2017.8206257.

[15] V. Petrenko *et al.*, "Analysis of the effectiveness path planning methods and algorithm for the anthropomorphic robot manipulator," in *2019 International Siberian Conference on Control and Communications, SIBCON 2019 - Proceedings*, 2019, doi: 10.1109/SIBCON.2019.8729657.

[16] V. Petrenko, A. Pavlov, F. Tebueva, and M. Gurchinsky, "The method of the kinematic structure reconfiguration of a multifunctional modular robot based on the greedy algorithm," in *Proceedings - International Conference on Developments in eSystems Engineering, DeSE*, 2019, vol. October-20, pp. 42–47, doi: 10.1109/DeSE.2019.00018.

[17] V. I. Petrenko, F. B. Tebueva, M. M. Gurchinsky, A. S. Pavlov, V. O. Antonov, and S. S. Ryabtsev, "Kinematic analysis anthropomorphic gripper with group drive," in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 862, no. 3, doi: 10.1088/1757-899X/862/3/032057.

[18] Petrenko, V., Tebueva, F., Gyrchinsky, M., Antonov, V., Shutova, J. The method of forming a geometric solution of the inverse kinematics problem for chains with kinematic pairs of rotational type only // IX International Multidisciplinary Scientific and Research Conference Modern Issues in Science and Technology / Workshop Advanced Technologies in Aerospace, Mechanical and Automation Engineering. IOP Conference Series: Materials Science and Engineering (2018). doi: 10.1088/1757-899X/450/4/042016

[19] V. Petrenko, F. Tebueva, V. Antonov, M. Gurchinsky, S. Ryabtsev, and A. Burianov, "Cooperative Motion Planning Method for Two Anthropomorphic Manipulators," *7th Sci. Conf. Inf. Technol. Intell. Decis. Mak. Support (ITIDS 2019), Part Ser. Adv. Intell. Syst. Res.*, vol. 166, no. Itids, pp. 146–151, 2019, doi: 10.2991/itids-19.2019.27.

[20] G. Kovács, D. Bogdanova, N. Yussupova, and M. Boyko, "Informatics tools, AI models and methods used for automatic analysis of customer satisfaction," *Stud. Informatics Control*, vol. 24, no. 3, pp. 261–270, 2015, doi: 10.24846/v24i3y201503.

[21] N. Yussupova, G. Kovács, M. Boyko, and D. Bogdanova, "Models and methods for quality management based on artificial intelligence applications," *Acta Polytech. Hungarica*, vol. 13, no. 3, pp. 45–60, 2016.

[22] V. I. Petrenko, F. B. Tebueva, M. M. Gurchinskiy, V. O. Antonov, and N. U. Untewsky, "The method of the quasioptimal per energy efficiency design of the motion path for the anthropomorphic manipulator in a real time operation mode," in *CEUR Workshop Proceedings*, 2018, vol. 2254, pp. 245–252.

[23] "Keras: the Python deep learning API." [Online]. Available: https://keras.io/. [Accessed: 21-Jun-2020].