Engineering Transactions, **67**(3): 333–345, 2019, doi: 10.24423/EngTrans.996.20190815
Polish Academy of Sciences • Institute of Fundamental Technological Research (IPPT PAN)
Université de Lorraine • Poznan University of Technology

Approaches to the Determination of the Working Area of Parallel Robots and the Analysis of Their Geometric Characteristics

Dmitry MALYSHEV[1)], Mikhail POSYPKIN[2), 3)], Larisa RYBAK[1)*]
Alexander USOV[2)]

[1)] *Belgorod State Technological University named after V.G. Shukhov*
Belgorod, Russian Federation
*Corresponding Author e-mail: rlbgtu@gmail.com

[2)] *Dorodnicyn Computing Centre, FRC CSC RAS*
Moscow, Russian Federation

[3)] *Moscow Institute of Physics and Technology*
Moscow, Russian Federation

The article presents and experimentally confirms two approaches to the problem of determining the working area of parallel robots using the example of a planar robot DexTAR with two degrees of freedom. The proposed approaches are based on the use of constraint equations of coordinates. In the first approach, the original kinematic equations of coordinates in the six-dimensional space (two coordinates describing the position of the output link and four coordinates – the rotation angles of the rods) followed by projecting the solution onto the two-dimensional plane is used. In the second approach, the system of constraint equations is reduced to a system of inequalities describing the coordinates of the output link of the robot, which are solved in a two-dimensional Euclidean space. The results of the computational experiments are given. As an algorithmic basis of the proposed approaches, the method of non-uniform coverings is used, which obtains the external and internal approximation of the solution set of equality/inequality systems with a given accuracy. The approximation is a set of boxes. It is shown that in the first approach, it is more efficient to apply interval estimates that coincide with the extremes of the function on the box, and in the second approach, grid approximation performs better due to multiple occurrences of variables in inequalities.

Key words: parallel robot; working area; non-uniform coverings; interval analysis; approximation; algorithm; multiple solutions.

1. Introduction

Over the past decades, parallel robots have attracted the attention of a large number of scientists from around the world [9, 11, 12, 18, 20]. Such robots have a number of advantages, such as rigidity and positioning accuracy, which led to their extensive use in industry. An important task to be solved when designing robots is to determine its working area, i.e. sets of points that can be reached by a robot tool (end-effector). The size of the working area is a key characteristic of the robot. The working area itself serves as the basis for working tool path planning.

Several methods for the working area determination have been proposed so far [1, 2, 17]. A simple representation of the constraints for a planar 3-R$\underline{\mathrm{P}}$R (R stands for a passive revolute joint, and $\underline{\mathrm{P}}$ for an actuated prismatic joint) robot was considered in [14]. It is shown that interval analysis is an effective tool for approximating the working area. The non-uniform coverings method is primarily aimed for solving global optimization problems. For the detailed description please refer to [4, 8]. It was successfully tested on some types of parallel robots [3, 5]. The method can be used for constructing coverage for any number of dimensions, including 3D working area [10].

In this work we propose two approaches to determining the robot working area. The first approach (approach A) directly uses the system of kinematic equations. The second approach (approach B) consists in reducing the system of kinematic equations for the rods to quadratic equations. Then the solvability of the system of quadratic equations is formulated as a set of inequalities with fewer variables. A theoretical and experimental comparison of the efficiency of the proposed approaches is performed for the DexTAR robot [19].

2. Formulation of the problem

The planar $\underline{\mathrm{R}}$RRR$\underline{\mathrm{R}}$ parallel robot DexTAR is a four-link mechanism controlled by two actuators (Fig. 1a). The robot has two degrees of freedom. As input variables, we take the rotation angles of the q_A and q_D rods and the output variables are the coordinates x_P and y_P of the output tool P (Fig. 1b).

a) b)

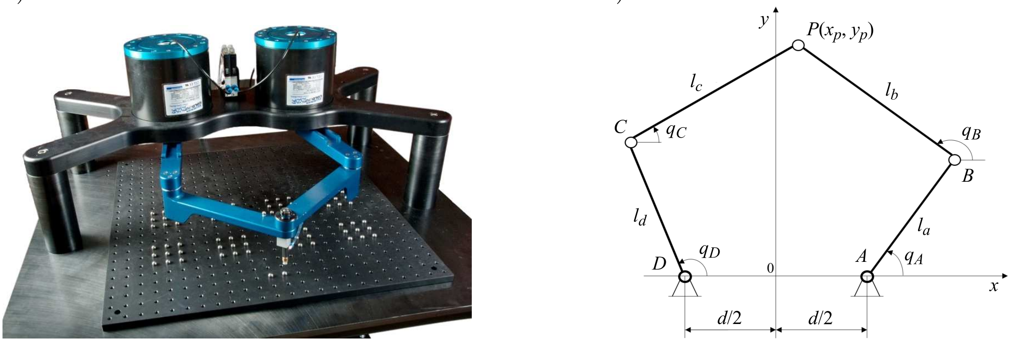

Fig. 1. Robot $\underline{\mathrm{R}}$RRR$\underline{\mathrm{R}}$: a) robot layout, b) block diagram of the robot.

The robot includes 4 rods of constant length l_a, l_b, l_c, l_d, $l_a = l_d$, $l_b = l_c$. Consider the case when the engines are located above the plane of the working area and do not affect it. The distance between the fixed ends of the rods is d. The letters R in the abbreviation $\underline{R}RRR\underline{R}$ designate 5 rotational kinematic pairs (D, C, P, B, A), two of which corresponding to actuators are underlined. Rotation of the actuators determines the forward movement of the working tool P along the axes. Kinematic equations connecting given geometrical parameters of the robot and the rods rotation angles looks as follows:

$$\begin{cases} x_p - l_b \cdot \cos q_B - l_a \cdot \cos q_A - \dfrac{d}{2} = 0, \\ x_p - l_d \cdot \cos q_D - l_c \cdot \cos q_C + \dfrac{d}{2} = 0, \\ y_p - l_a \cdot \sin q_A - l_b \cdot \sin q_B = 0, \\ y_p - l_d \cdot \sin q_D - l_c \cdot \sin q_C = 0. \end{cases} \tag{2.1}$$

The system of Eqs (2.1) has six variables x_P, y_P, q_A, q_B, q_C, q_D. Below we propose a transformation that reduces the dimension (number of variable) of the system. Consider the kinematic chain DCP (Fig. 2) in a $X'O'Y'$ coordinate system, where O' is located in the center of the rotary joint D. Using the transition from the XOY coordinate system to the $X'O'Y'$ coordinate system, we express the coordinates of points C and P:

$$\begin{cases} x'_{C,P} = x_{C,P} + \dfrac{d}{2}, \\ y'_{C,P} = y_{C,P}. \end{cases} \tag{2.2}$$

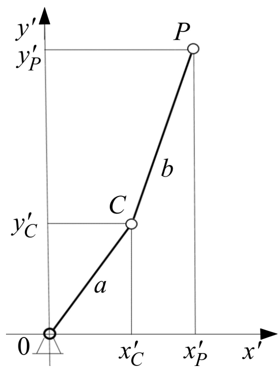

FIG. 2. Kinematic chain DCP.

We write the constraint equations for the DCP circuit, taking into account (2.2):

$$\begin{cases} {x'_C}^2 + {y'_C}^2 = a^2, \\ (x'_P - x'_C)^2 + (y'_P - y'_C)^2 = b^2. \end{cases} \tag{2.3}$$

Subtract from the second equation the first equation of system (2.3) and get

$$ {x'_P}^2 - 2x'_C x'_P + {y'_P}^2 - 2y'_C y'_P = b^2 - a^2. \tag{2.4}$$

In Eq. (2.4) we express y'_C

$$ y'_C = -\frac{x'_C x'_P}{y'_P} + \frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P}. \tag{2.5}$$

Substitute the expression (2.5) in the first equation of the system (2.3):

$$ {x'_C}^2 + \left(-\frac{x'_C x'_P}{y'_P} + \frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P} \right)^2 = a^2. \tag{2.6}$$

The Eq. (2.6) can be rewritten as

$$\begin{aligned} \left(1 + \frac{{x'_P}^2}{{y'_P}^2}\right) {x'_C}^2 + \left[2\left(-\frac{x'_P}{y'_P}\right)\left(\frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P}\right)\right] x'_C \\ + \left[\left(\frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P}\right)^2 - a^2 \right] = 0. \end{aligned} \tag{2.7}$$

From Eq. (2.7) we derive the discriminant expression:

$$\begin{aligned} D_1 = \left[2\left(-\frac{x'_P}{y'_P}\right)\left(\frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P}\right)\right]^2 \\ - 4\left(1 + \frac{{x'_P}^2}{{y'_P}^2}\right) \left[\left(\frac{{x'_P}^2}{2y'_P} + \frac{y'_P}{2} + \frac{a^2 - b^2}{2y'_P}\right)^2 - a^2 \right]. \end{aligned} \tag{2.8}$$

In Eq. (2.8), we move from the $X'O'Y'$ coordinate system to the XOY system and obtain:

$$\begin{aligned} D_1 = \left[2\left(-\frac{x_P + \frac{d}{2}}{y_P}\right)\left(\frac{(x_P + \frac{d}{2})^2}{2y_P} + \frac{y_P}{2} + \frac{a^2 - b^2}{2y_P}\right)\right]^2 \\ - 4\left(1 + \frac{(x_P + \frac{d}{2})^2}{{y_P}^2}\right) \left[\left(\frac{(x_P + \frac{d}{2})^2}{2y_P} + \frac{y_P}{2} + \frac{a^2 - b^2}{2y_P}\right)^2 - a^2 \right]. \end{aligned} \tag{2.9}$$

It is obvious that system (2.3) has a solution if and only if the quadratic Eq. (2.7) is solvable, i.e. when in (2.9) $D_1 \geq 0$. Performing similar transformations for the kinematic chain ABP, we get

$$D_2 = \left[2\left(-\frac{x_P - \frac{d}{2}}{y_P}\right)\left(\frac{(x_P - \frac{d}{2})^2}{2y_P} + \frac{y_P}{2} + \frac{a^2 - b^2}{2y_P}\right)\right]^2 - 4\left(1 + \frac{(x_P - \frac{d}{2})^2}{y_P{}^2}\right)\left[\left(\frac{(x_P - \frac{d}{2})^2}{2y_P} + \frac{y_P}{2} + \frac{a^2 - b^2}{2y_P}\right)^2 - a^2\right]. \tag{2.10}$$

Note that the working area includes all points with coordinates for which inequalities are simultaneously satisfied

$$\begin{cases} g_1(x) \geq 0, \\ g_2(x) \leq 0, \end{cases} \tag{2.11}$$

where $g_1(x)$ and $g_2(x)$ denote $-D_1$ and $-D_2$, respectively.

3. Algorithms for the working area approximation

The proposed approach is based on the non-uniform coverings methods, proposed in [4, 8] for global optimization. To calculate the estimates, we investigate the possibility of using both the interval analysis [13] and the grid approximation.

3.1. *Systems of equations*

To approximate the system (2.1), we used the first approach. Consider a system of nonlinear algebraic equations written in a general form:

$$\begin{cases} g_1(x) = 0, \\ \dots \\ g_m(x) = 0, \\ a_i \leq x_i \leq b_i, \quad i = 1, ..., n. \end{cases} \tag{3.1}$$

The initial box Q that encloses the whole solution set X is defined by interval constraints $a_i \leq x_i \leq b_i$, $i = 1, ..., n$. The proposed approach (approach A) constructs a coverage of the set X of solutions of the system (3.1). The coverage is a set of boxes, with a diameter less or equal to the prescribed accuracy δ.

Consider an arbitrary box B. Let $m(B) = \max_{i=1,...,m} \min_{x \in B} g_i(x)$ and $M(B) = \min_{i=1,...,m} \max_{x \in B} g_i(x)$. If $m(B) > 0$ or $M(B) < 0$ then B contains no feasible points for a system (3.1). The proposed algorithm, shown in the Fig. 3 discards such boxes. If a box cannot be discarded it is partitioned into two smaller boxes unless its diameter is below the prescribed accuracy δ. The algorithm works with three lists of n-dimensional boxes $\mathbb{P}$ (current list), $\mathbb{P}_A$ (the coverage) and $\mathbb{P}_E$ (discarded boxes).

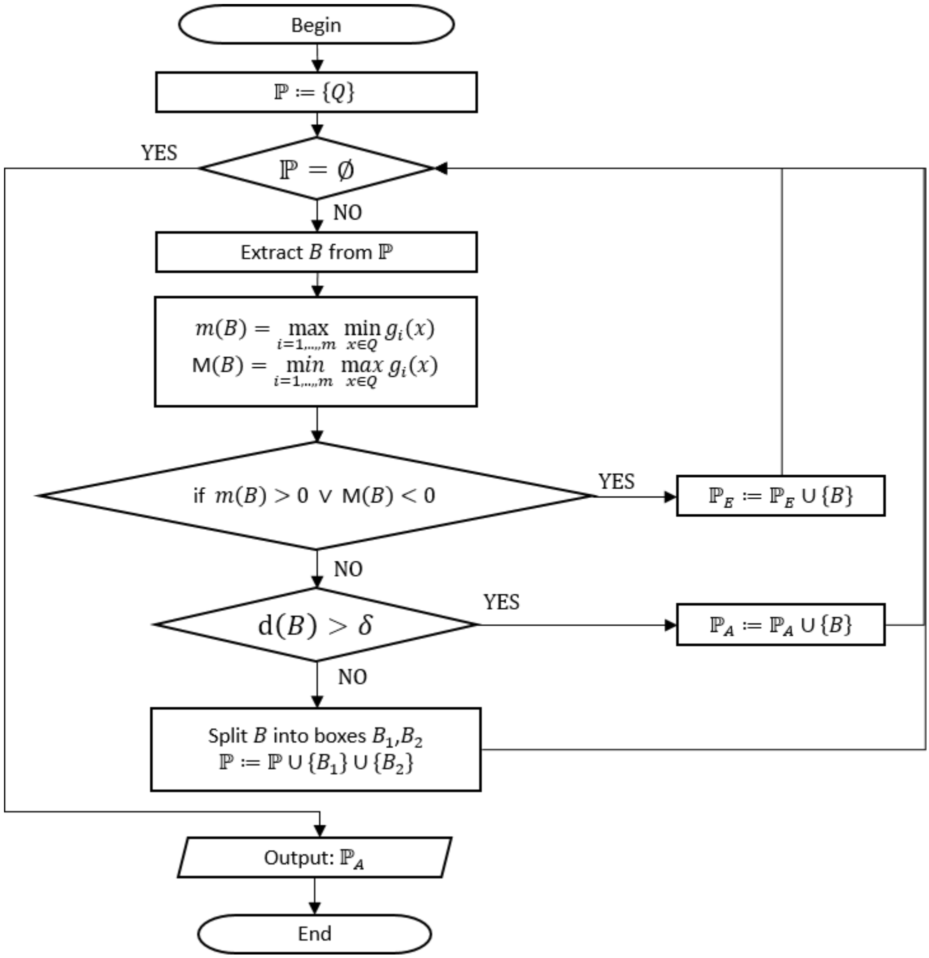

FIG. 3. Algorithm A for approximating a working area.

The algorithm works as follows:

1. At the beginning the list $\mathbb{P}$ consists of only one initial box Q, that includes the whole range of theoretically maximum limits of the set X. Lists $\mathbb{P}_A$ and $\mathbb{P}_E$ are initially empty.
2. Extract from the list $\mathbb{P}$ a box B.
3. Compute $m(B)$ and $M(B)$.

4. If $m(B) > 0$ or $M(B) < 0$, then the box is excluded from further consideration, falling into the list $\mathbb{P}_E$, that is, $\mathbb{P}_E := \mathbb{P}_E \cup \{B\}$.
5. If the box has a diameter less than or equal to the given parameter δ, that is, $d(B) \leq \delta$ characterizing the accuracy of the approximation, it is added to the $\mathbb{P}_A$ list, that is, $\mathbb{P}_A := \mathbb{P}_A \cup \{B\}$.
6. In either case, the box is divided into two equal boxes B_1 and B_2 along the longest edge. Boxes are appended to the end of the list $\mathbb{P}$, that is, $\mathbb{P} := \mathbb{P} \cup \{B\}$.
7. The algorithm terminates when the list $\mathbb{P}$ becomes empty, otherwise steps 2–8 are repeated.

The finiteness of the number of steps of the algorithm follows from the limit on the minimum diameter of a box.

Usually we are interested in a projection of a set X to a set of axes. For instance in the example under consideration the set $\mathbb{P}_A$ is a subset of $\mathbb{R}^6$. Since the task is to find the working area in the parameter space x_P, y_P, it is necessary to project $\mathbb{P}_A$ into $\mathbb{R}^2$. Since the sets are boxes, getting a projection is not difficult.

3.2. Systems of inequalities

To approximate the systems of inequalities (2.11), we used the second approach described in [5, 7]. For the convenience of the reader we reproduce it here. The algorithm works with a system of inequalities written in a general form:

$$\begin{cases} g_1(x) \leq 0, \\ \dots \\ g_m(x) \leq 0, \\ a_i \leq x_i \leq b_i, \quad i = 1, ..., n. \end{cases} \tag{3.2}$$

The initial box Q that encloses the whole solution set X is defined by interval constraints $a_i \leq x_i \leq b_i, i = 1, ..., n$. Consider an arbitrary box B. Let $m(B) = \max_{i=1,...,m} \min_{x \in B} g_i(x)$ and $M(B) = \max_{i=1,...,m} \max_{x \in B} g_i(x)$. Notice that $M(B)$ is defined differently from the equality case. If $m(B) > 0$ then B contains no feasible points for a system (3.2). The proposed algorithm, shown in the Fig. 3 discards such boxes. If $M(B) \leq 0$ then every point of a box B is a feasible solution. Therefore it can be added to the coverage as an inner box. If a box cannot be discarded it is partitioned into two smaller boxes unless its diameter is below the prescribed accuracy δ. Such boxes are added to the boundary approximation. The algorithm (Fig. 4) works with four lists of boxes $\mathbb{P}$, $\mathbb{P}_I$, $\mathbb{P}_A$, and $\mathbb{P}_E$.

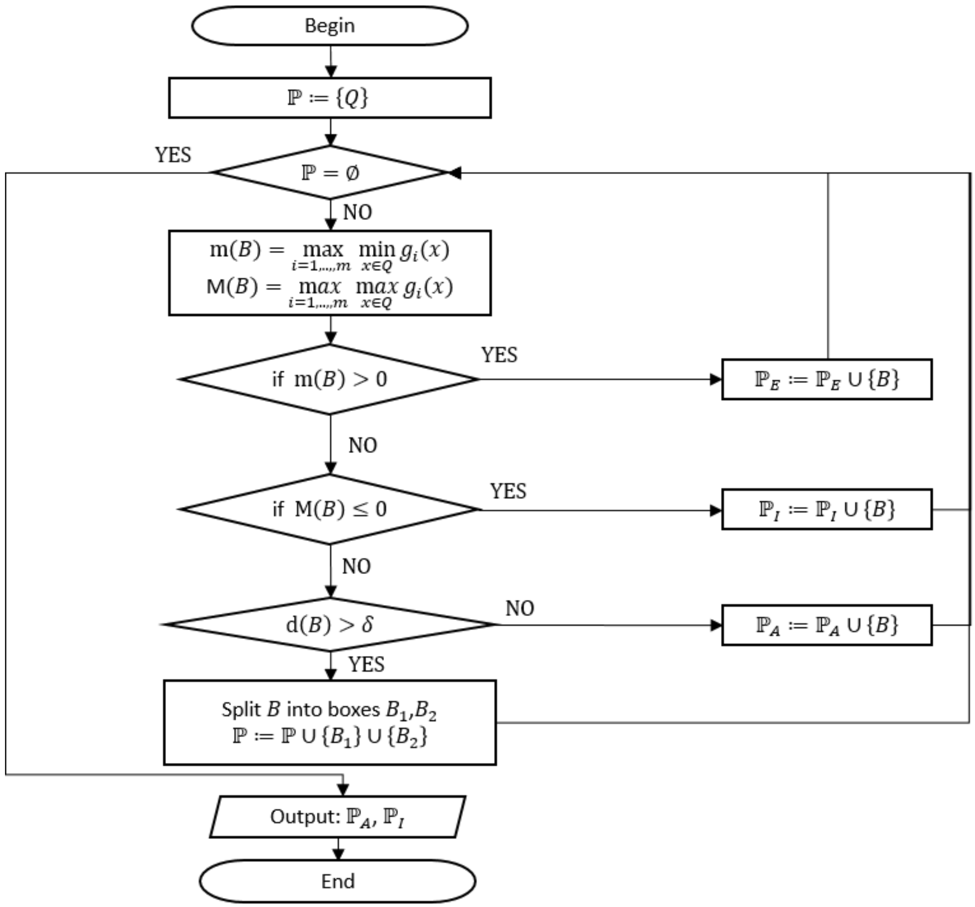

FIG. 4. The algorithm B for approximating the working area.

The algorithm works as follows:

1. At the beginning the list $\mathbb{P}$ consists of only one initial box Q, that includes the whole range of theoretically maximum limits of the set X. Lists $\mathbb{P}_A$, $\mathbb{P}_I$ and $\mathbb{P}_E$ are initially empty.
2. Extract from the list $\mathbb{P}$ a box B.
3. Compute $m(B)$ and $M(B)$.
4. If $m(B) > 0$, then the box is excluded from further consideration, falling into the list $\mathbb{P}_E$, that is, $\mathbb{P}_E := \mathbb{P}_E \cup \{B\}$.
5. If $M(B) \le 0$, it is added to the $\mathbb{P}_I$ list, that is, $\mathbb{P}_I := \mathbb{P}_I \cup \{B\}$.
6. If the box has a diameter less than or equal to the given parameter δ, that is, $d(B) \le \delta$ characterizing the accuracy of the approximation, it is added to the $\mathbb{P}_A$ list, that is, $\mathbb{P}_A := \mathbb{P}_A \cup \{B\}$.

7. In either case, the box is divided into two equal boxes B_1 and B_2 along the longest edge. Boxes are appended to the end of the list $\mathbb{P}$, that is, $\mathbb{P} := \mathbb{P} \cup \{B\}$.
8. The algorithm terminates when the list $\mathbb{P}$ becomes empty, otherwise steps 2–8 are repeated.

The finiteness of the number of steps of the algorithm follows from the limit on the minimum diameter of a box.

As for approach A, the finiteness of the number of steps of the algorithm follows from the restriction on the minimum diameter of the box. The approximation is assumed to be equal to the union of the sets $P_B = \mathbb{P}_A \cup \mathbb{P}_I$.

4. Realization of algorithms and results of computational experiments

The considered algorithms require finding the minimum and maximum of the functions included in the left side of equations and/or inequalities. In general, the exact minimum is difficult to find and, therefore, estimates are used. Initially, interval estimations were applied in both methods.

Interval methods allows to compute an enclosing interval for a function from intervals on its parameters. This process can be automated by applying interval arithmetic rules. The detailed information on interval analysis can be found in [13].

In the approach A, interval estimates are the ideal choice (for our particular robot). This is due to the fact that all variables are included in the evaluated expressions once. Therefore the interval estimates coincide with the extrema of the functions g_i on the box, i.e. cannot be improved.

In the approach B with the system (2.11), expressions contain multiple occurrences of variables. The verification showed that interval estimates do not allow obtaining approximations of acceptable quality. Therefore, another method has been applied, consisting of the approximate estimation of the extrema of a function f_i on a uniform grid. In each box, the function on the left side of the inequality is calculated at the nodes of a rectangular grid of size $N \times N$, where $N = 100$. The minimum and maximum on the grid of the function f_i values on the grid are taken as the estimate of the minimum and maximum of the function on the box.

It can be observed that the size of the grid grows exponentially with increasing the number of parameters. This severe problem can be addressed by the following two approaches. The first approach is to apply parallelization – different grid nodes can be computed on different cores. The second approach is to use of quasi-random low-discrepancy sequences (e.g. [15]) instead of rectangular grids.

The calculations were carried out on a personal computer that has a quad-core Intel i7 processor with a clock frequency of 2.4 GHz and 8 GB RAM. The algorithm is implemented in C++ using our own interval library [16].

The Fig. 5 shows the approximations of the working area obtained using approaches A and B. The simulation was performed for the following parameters: $l_a = l_d = 72$ mm, $l_b = l_c = 87$ mm, $d = 60$ mm. Each figure shows both the entire working area (left) and its enlarged fragment (right).

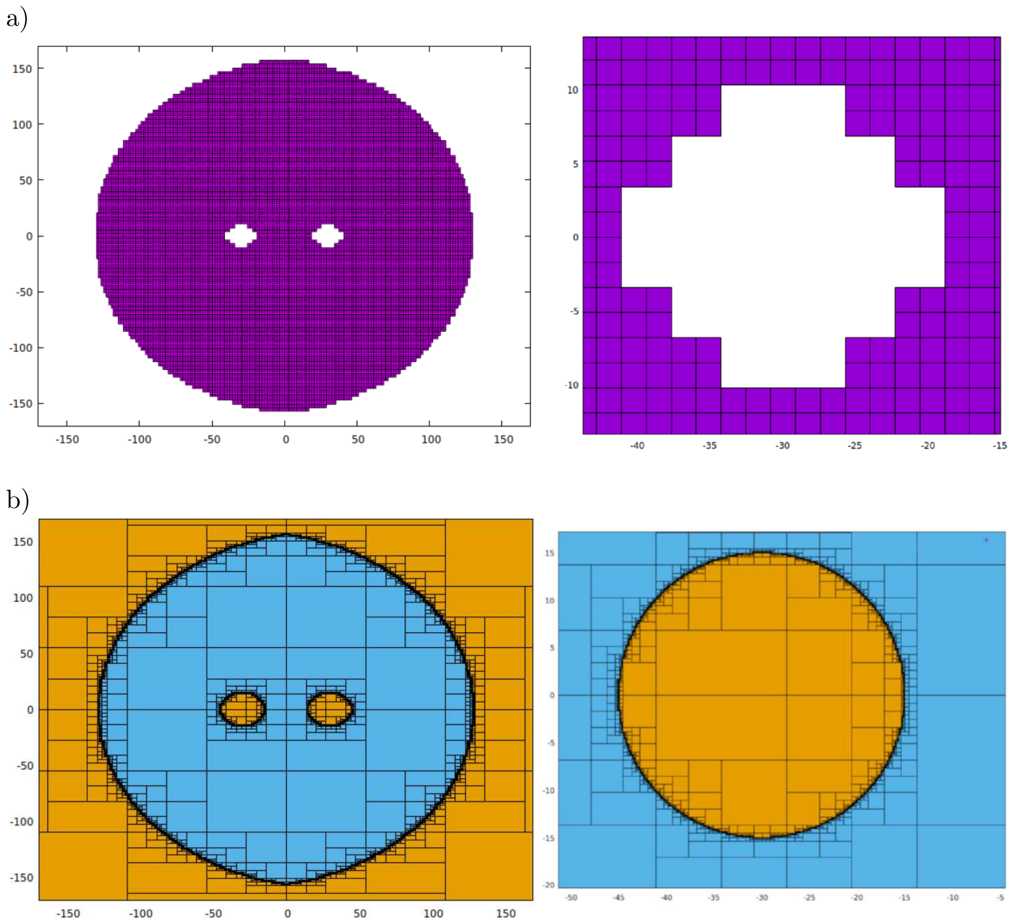

FIG. 5. Working area: a) based on approach A; b) based on approach B.

The calculation results allow us to draw the following conclusions. The working area with both approaches has the same shape and size. The quality of the approximation obtained with the help of the approach B is significantly higher. This can be easily explained by the higher number of dimensions in the first case (6 *vs.* 2). Thea approach A requires significantly more resources: the time of calculations by approach A was 39 minutes 40 seconds, and approach B – 3 minutes and 50 seconds.

It is worth noting the constructed approximaton can be used for computing the volume of the robot's workspace. Clearly this area is a function of the rod's lengths. Using approach B, we determine the area of the workspace for different ratio of lengths l_b/l_a, when the sum of the lengths of the rods is fixed: $l_a + l_b =$ 160 mm. Figure 6 shows the value of the workspace's area when the ratio l_b/l_a ranges from 0.7 to 1.25.

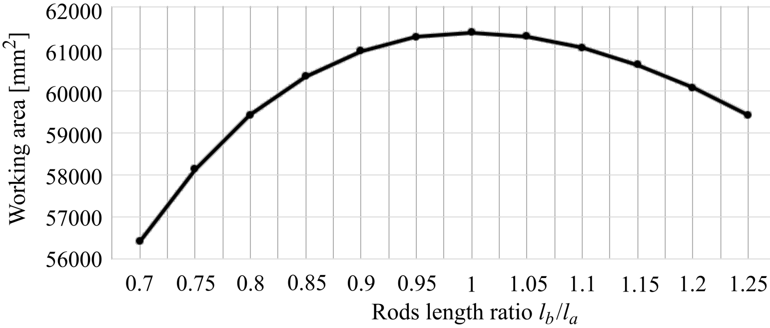

FIG. 6. The dependence of the area of the working area on the ratio of the length of the rods.

It can be seen from the figure that the maximum area of the working area is achieved with the ratio of lengths $l_b/l_a = 1$, i.e. $l_a = l_b = l_c = l_b = 80$ mm. The area of the working space is 61400.3 mm^2.

5. Conclusion

The article proposed and tested two approaches to the automated approximation of the robot's working area. The resulting approximation can be used for the robot's path planning and for working area volume computation.

The approaches can be used for constructing coverage for any number of dimensions. Approach A determines a set of solutions of a system of kinematic equations. Approach B requires a preliminary transformation of a system of constraint equations to a system of inequalities. This normally results in a decrease of the number of parameters. Both approaches are implemented in software. The performed computational experiments showed that approach B allows obtaining much better approximations w.r.t. approach A in a less time. The disadvantage of approach B is the need to transform the system of inequalities to the desired type manually.

It should be noted that the proposed approaches are quite general and can be applied to compute the working areas for a potentially arbitrary manipulator. Indeed it is sufficient to derive a system of kinematic Eqs (3.1) or (optionally) system of inequality (3.2). However when applying to complex robotic systems the approach may be very resource demanding. This can be mitigated by applying high-performance computing methods, e.g. by employing techniques elaborated in parallel global optimization [6].

Acknowledgments

This work was supported by the Russian Science Foundation, the agreement number 16-19-00148.

References

1. Tagiyev N.R., Alizade R.I., Duffy J., *A forward and reverse displacement analysis of an in-parallel spherical manipulator*, Mechanism and Machine Theory, **29**(1): 125–137, 1994.

2. Angeles J., Bulca F., Zsombor-Murray P.J., *On the workspace determination of spherical serial and platform mechanisms*, Mechanism and Machine Theory, **34**(3): 497–512, 1999.

3. Turkin A., Rybak L., Evtushenko Y., Posypkin M., *On the dependency problem when approximating a solution set of a system of nonlinear inequalities*, 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), pp. 1481–1484, 2018, https://doi.org/10.1109/EIConRus.2018.8317377.

4. Posypkin M., Evtushenko Yu., *A deterministic approach to global box-constrained optimization*, Optimization Letters, **7**(4):819–829, 2013, https://link.springer.com/article/10.1007/s11590-012-0452-1.

5. Rybak L., Evtushenko Yu., Posypkin M., Turkin A., *Approximating a solution set of nonlinear inequalities*, Journal of Global Optimization, **71**: 1–17, 2017, https://doi.org/10.1007/s10898-017-0576-z.

6. Sigal I., Evtushenko Yu., Posypkin M., *Framework for parallel large-scale global optimization*, Computer Science-Research and Development, **23**(3–4): 211–215, 2009, https://doi.org/10.1007/s00450-009-0083-7.

7. Rybak L.A., Turkin A.V., Evtushenko Yu.G., Posypkin M.A., *Numerical method for approximating the solution set of a system of non-linear inequalities*, International Journal of Open Information Technologies, **4**(12): 1–6, 2016, http://injoit.org/index.php/j1/article/view/369.

8. Evtushenko Yu.G., *Numerical methods for finding global extrema (case of a non-uniform mesh)*, USSR ComputationalMathematics and Mathematical Physics, **11**(6): 38–54, 1971, https://doi.org/10.1016/0041-5553(71)90065-6.

9. Kraynev A.F., Glazunov V.A., Koliskor A.Sh., *Spatial Mechanisms of Parallel Structure* [in Russian: *Prostranstvennye mekhanizmy parallelnoi struktury*], Moscow: Nauka, 1991.

10. Malyshev D.I., Posypkin M.A., Gorchakov A.Yu., Ignatov A.D., *Parallel algorithm for approximating the work space of a robot*, International Journal of Open Information Technologies, **7**(1): 1–7, 2019.

11. Merlet J.P., *Parallel robots*, Springer Publishing Company, Incorporated, 2006. https://www.springer.com/kr/book/9781402041327.

12. Gosselin C., Kun S., *Structural synthesis of parallel mechanisms.* M.: Fizmatlit, 2012.

13. Jaulin L., Kieffer M., Didrit O., Walter É., *Applied interval analysis*, Springer Science & Business Media, 2001, https://doi.org/10.1007/978-1-4471-0249-6.

14. Husty M.L., *On the workspace of planar three-legged platforms*, World Automation Congress, **3**: 339–344, 1996.

15. Niederreiter H., *Low-discrepancy and low-dispersion sequences*, Journal of Number Theory, **30**(1): 51–70, 1988.

16. Posypkin M., Usov A., Implementation and verification of global optimization benchmark problems, Open Engineering, **7**(1): 470–478, 2017.

17. Clavel R., *Design of a fast parallel robot at 4 degrees of freedom* [in French: *Conception d'un robot parallele rapide a 4 degres de liberte*], EPFL, Lausanne, 1991. http://dx.doi.org/10.5075/epfl-thesis-925.

18. Chichvarin A.V., Rybak L.A., Erzhukov V.V., *Effective methods for solving problems of kinematics and dynamics of a machine tool with a parallel structure* [in Russian: *Эффективные методы решения задач кинематики и динамики робота-станка параллельной структуры*], Fizmatlit, 2011.

19. Stan S.D., Maties V., Balan R., *Optimization of a 2 dof micro parallel robot using genetic algorithms*, *Frontiers in Evolutionary Robotics*, I-Tech Education and Publishing, pp. 465–490, 2008.

20. Kuzmina V.S., Virabyan L.G., Khalapyan S.Y., *Optimization of the positioning trajectory of the planar parallel robot output link*, Bulletin of BSTU named after V.G. Shukhov, **9**: 106–113, 2018, http://vestnik_rus.bstu.ru/shared/attachments/184909.

Received December 10, 2018; accepted version July 19, 2019.