

# Determination of accuracy contour and optimization of workpiece positioning for robot milling

**Gotlih, J.<sup>a,\*</sup>, Brezocnik, M.<sup>a</sup>, Balic, J.<sup>a</sup>, Karner, T.<sup>a</sup>, Razborsek, B.<sup>a</sup>, Gotlih, K.<sup>a</sup>**

<sup>a</sup>Faculty of mechanical engineering, University of Maribor, Maribor, Slovenia

---

## ABSTRACT

Workpiece positioning into a machine's workspace has become a simple task. Advanced CNC machines are equipped with standardized clamping systems, allowed workpiece dimensions are listed in the machine's documentation and tolerance levels of the end produced parts are known. This gives users plenty of information and good confidence that they are choosing the best machine for a specific task. For more universal machines like industrial robots this is not the case. Due to their flexibility industrial robots can be an alternative to specialized CNC machines, but when a specific task should be executed, important information is missing. For a standard industrial robot the mechanisms layout, its dimensions and its reachable workspace is known, but accuracy levels over the robot's workspace are not. If a workpiece should be milled within certain accuracy limits the robot's documentation offers no information on how close it can be located to the borders of the robot's workspace. This article deals with the mentioned problem with a novel methodology. Based on experimental data we found that a standard 6 DOF industrial robot's reachable workspace can be divided into two regions, one with suitable milling accuracy and another with rapidly decreasing milling accuracy. To isolate the suitable accuracy region a regional non-dominated sorting algorithm was developed and an accuracy contour separating the regions was extracted. In the second part of the article a genetic search algorithm based on regional non-dominated sorting was applied to find the biggest arbitrary shaped workpiece's size, position and orientation in the suitable milling accuracy region of the robot's workspace.

© 2017 PEI, University of Maribor. All rights reserved.

---

## ARTICLE INFO

**Keywords:**  
Robot milling  
Accuracy contours  
Workpiece positioning  
Non-dominated sorting  
Genetic algorithm  
Optimization

**\*Corresponding author:**  
[janez.gotlih@um.si](mailto:janez.gotlih@um.si)  
(Gotlih, J.)

**Article history:**  
Received 26 May 2017  
Revised 20 July 2017  
Accepted 24 July 2017

## 1. Introduction

Using industrial robots for less common tasks is becoming increasingly interesting among researchers, as due to their flexibility, machining could become a more regular application for industrial robots. To reach the level where robot machining becomes an acceptable alternative to CNC machining, robot machining accuracy needs to be analysed, understood and improved. The main reasons for inaccuracies in robot machining were classified and divided into three groups by Kubela et al. [1] as environmental errors, robot dependent errors and process dependent errors. Robot dependent errors were further divided into geometrical, non-geometrical and system errors. Their study on wood milling showed that the most significant errors are the result of robots reverse motion, causing end product shape inaccuracies in range from 0.3–0.5 mm. Geometric error compensation has been studied by Wu et al. [2], who added a fifth step to the usual robot calibration approach, used to achieve the desired robot positioning accuracy. The proposed steps include modelling, design of experiments, measurement, identification, and imple-

mentation. Their goal was to improve the standard approach, so that a minimum number of experiments is required for robot's calibration. With the proposed calibration technique an improved positioning accuracy was achieved in absolute range of up to 0.17 mm. Greater positioning accuracy was measured by Józwik et al. [3], who used a high speed camera to investigate the effect of direction of approach on positioning accuracy. In their experiment the authors found that positioning inaccuracy is raising with robot's continuous operation. Accuracy improvement in continuous motion of a welding robot was achieved by Marônek et al. [4] by insertion of additional points into the robot's trajectory. Due to technological reasons constant speed needs to be maintained during welding. With help of experiments the authors found better trajectory matching at higher movement speeds. Jezeršek et al. [5] developed a system for automatic trajectory generation for robotic welding based on robotic laser scanning. Accuracy levels comparable to traditional visually based trajectory generation methods were achieved, but teaching times were shortened more than 30-times. A system for optical accuracy evaluation of robots in static and dynamic operation mode was set up by Papa and Torkar [6]. In their experiment the authors found the same subpixel accuracy in both modes.

To predict a mechanism's accuracy, different dexterity related performance measures have been introduced in the past. Manipulating ability of end-effector of robotic mechanisms was studied by Yoshikawa [7] and a manipulability index was introduced. Manipulability, defined as determinant of the Jacobian matrix of a robotic mechanism is proposed as a robot's performance indicator. Other dexterity measures based on determinant of the Jacobian matrix like condition number and monotone manipulability were proposed, as summarized by Gotlih et al. [8]. In their study authors demonstrated application of velocity anisotropy on a real industrial robot and introduced a four dimensional robotic workspace for visualisation of robot's manipulability. Problems of linking performance measures to a robot's accuracy were exposed by Merlet [9]. In his experiment only the robot's manipulability index has shown the same trend as its measured accuracy, but no connection between manipulability and accuracy was established. To overcome unit based inconsistency, transition from kinematic to dynamic performance measures was proposed and a power manipulability index was introduced by Mansouri and Ouali [10]. The power manipulability index is a combination of kinematic and static parameters and includes translational and rotational components. The authors showed that power manipulability is a homogeneous tensor insensitive to physical units change.

Optimization of robotic systems with algorithms based on performance measures was studied by many authors. Dynamic performance measures were found to be more significant than kinematic by Vosniakos and Matsas [11]. The authors used genetic algorithm for optimal positioning of a predefined milling path into a robot's workspace. They found out that with application of genetic algorithm an improvement in robot's manipulability and a reduction in joint torques can be achieved. Genetic algorithm was also used for robot's path planning to avoid collision of robot's structure with obstacles in its workspace [12]. Another approach of robot path planning considering a mechanical power index and an obstacle avoidance index in a multi-criteria optimization problem was suggested by R. dos Santos et al. [13]. To overcome local minima authors have used a two-phase optimization process and the sequential linear programming method. Results showed good agreement with Pareto optimality as better obstacle avoidance caused an increase in power consumption. A multi-criteria path planning algorithm for robotic laser cutting was developed by Dolgui and Pashkevich [14] and implemented in an industrial software package, yielding significant reduction in process planning and changeover time for end users. A three step optimization procedure for workpiece positioning into a robot's workspace was proposed by Lin et al. [15]. The authors developed an approach based on manipulability and stiffness indexes and introduced a deformation index to refine the final workspace maps, where contour lines dividing the robot's workspace into sub regions indicate optimal workpiece positioning locations.

To position a workpiece into a robot's refined workspace precisely, it is necessary to extract individual contour lines. Techniques for contour extraction are mainly associated with computer vision and shape detection and are used in various engineering fields. An approach for workpiece localization and its shape deviations from nominal geometry in robotic deburring was

proposed by Kuss et al. [16]. A point cloud of a CAD model was compared with a point cloud representing a real workpiece in a robot's workspace. The best matching point clouds were used to generate workpiece contours and trajectories for the deburring process. In their experiment the authors used a milling robot equipped with a stereo camera for shape sensing and achieved a significant improvement in deburring quality. A robotic cell for optical workpiece detection was developed by Klancnik et al. [17]. The authors used neural networks for system calibration where the relation between captured pictures and the robot's coordinate system was learnt. Numerous tests proved the system adequate for use in practice. Karimi and Nategh [18] introduced contour maps to graphically estimate kinematic errors in hexapod machines for an optimal workpiece setup. The authors developed an analytical model for kinematic error estimation with an average 11 % discrepancy from experimental results. Contour line generation from short line segments with a genetic algorithm based method was proposed by Wei et al. [19]. An automatic octree based algorithm for façade and opening contour detection from point clouds obtained with terrestrial laser scanning was developed by Truong-Hong and Laefer [20] where a knowledge database was used for contour validation. Relative errors of maximum 3 % were found between measured drawings and the reconstructed models. A method for building boundary identification based the Delaunay triangulation was developed by Awrangjeb and Guojun [21]. By defining a maximum point to point distance the algorithm proved capable of detecting concave edges and holes of any point cloud shape.

## 2. Material and methods

An experiment to evaluate the effect of different performance measures on robot milling accuracy was set up by Veber [22] in his master's thesis. A standard 6 DOF industrial milling robot equipped with a milling tool and a clamping table was selected for the experiment and the system was calibrated. To reduce influence of human errors each experiment was repeated three times. Workpieces of polymer composite material with low hardness were selected to reduce inaccuracies caused by cutting forces. According to the robot's manufacturer, workpiece hardness should not exceed 35 N/mm<sup>2</sup> and deviations in range of  $\pm 0.12$  mm should be expected due to the robot's rigidity. With all inaccuracy causes controlled, measured inaccuracies were expected to be influenced mainly by the robot's dexterity.

In the experiment five milling locations were selected and at each location three workpieces were milled in three sequential runs. The positioning of the workpieces is presented in Table 1, where x, y, z represent distances in X, Y and Z direction from the robot's base coordinate system and d is the resulting absolute distance.

Edge and circular pocket milling operations were performed on each workpiece. End shapes were then measured with a coordinate measuring machine. Measurement results compared to nominal values are presented in Table 2.

**Table 1** Initial positions of the measured workpieces

Pos.	x (mm)	y (mm)	z (mm)	d (mm)
1	1940	1053	610	2207
2	2060	-498	610	2060
3	1283	0	610	1376
4	1515	821	610	1723
5	1811	-709	610	1945

**Table 2** Robot milling measurement results

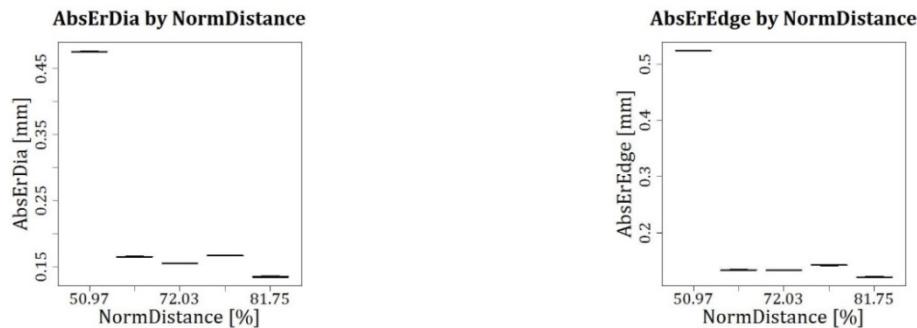
Pos.	Experiment 1		Experiment 2		Experiment 3	
	Nominal value (mm)	20	Nominal value (mm)	20	Nominal value (mm)	20
1	20.134	80.122	20.136	80.122	20.134	80.124
2	20.166	80.143	20.167	80.142	20.167	80.143
3	20.477	80.522	20.476	80.523	20.476	80.523
4	20.165	80.134	20.166	80.134	20.165	80.136
5	20.155	80.134	20.155	80.134	20.155	80.134

### 3. Results and discussion

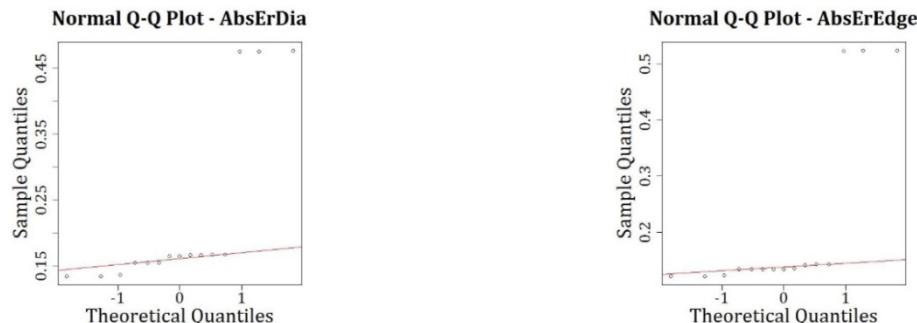
For result analysis absolute errors for measured diameters and edge lengths were calculated and a normalized distance parameter (NDS) expressed in percent was introduced. NDS was calculated as the distance of a workpiece from the robot's base coordinate system divided by the maximum distance the robot could reach with a vertical tool orientation.

Fig. 1 reveals highest absolute errors at milling position 3, located close to 50 % NDS. For workpieces located at 63 % NDS and above, absolute errors are almost constant and do not exceed 0.2 mm, which is close to the inaccuracy caused by the robot's rigidity. In range between 51 % and 63 % NDS, milling inaccuracy quickly raises. Due to high measurement consistency indicated by low result variation, human and cutting force based inaccuracies can be excluded and the quick decrease of accuracy at milling position 3 can be attributed to the robot's dexterity.

Fig. 2 shows result point deviations from a normal distribution. Measurement results are represented by sample quantiles, while theoretical quantiles represent a standard normal distribution. For both measured quantities the highest deviations are found at highest inaccuracies. This indicates a nonlinear dependency between workpiece distance from the robot's base and the robot's milling accuracy, based on which a sharp boundary separating the robot's workspace into two milling accuracy regions is expected.



**Fig. 1** Absolute errors for diameter (left) and edge length (right) versus normalized distance



**Fig. 2** Absolute error deviations for diameter (left) and for edge length (right)

#### 3.1 Accuracy contour for robot milling

In order to extract the suitable milling accuracy region from the robot's workspace, a connection between accuracy and dexterity was established. Because of the unit based inconsistency of manipulability [10] we did not intend to find a model connecting accuracy with dexterity over its complete range, instead we only equated accuracy and manipulability at a constant value. The dexterity measure defined by Yoshikawa [7] was found to be the most reliable and was therefore used in our analysis. By assuming that same manipulability causes same milling accuracy, we were able to extrapolate a milling accuracy boundary value to the robot's three dimensional workspace to create a spatial accuracy contour, separating the region of suitable milling accuracy from the region of quickly reducing milling accuracy.

As an example a standard 6 DOF robot, described by Denavit-Hartenberg (DH) notation, given in Table 3 was chosen.

With inverse kinematics the robot's joint angles and dexterity values at all five milling positions were found. Tool orientation during milling was parallel to the robot's main Z axis, therefore  $\theta_4$  and  $\theta_6$  joints were considered constant. Additionally, all workpieces were projected to the Y = 0 plane, so that rotation about robot's first axis  $\theta_1$  was also considered constant. As a result only  $\theta_2$ ,  $\theta_3$  and  $\theta_5$  contributed to dexterity calculation.

Normalized dexterity ( $ND$ ) was introduced as actual dexterity divided by the highest dexterity of the mechanism.  $ND$  is expressed in percent (Eq. 1).

$$ND = \frac{detJ}{\max(detJ)} \cdot 100 \quad (1)$$

Table 4 shows relative joint angles and  $ND$  values of the robot at all five milling positions. Robot's initial angle positions were  $\theta_1 = \theta_2 = \theta_3 = \theta_6 = 0^\circ$ ,  $\theta_4 = 180^\circ$  and  $\theta_5 = 90^\circ$ .

Results presented in Tables 2 and 4 reveal that the rapid decrease in accuracy happens between 22 % and 45 % normalized dexterity. Based on our conclusion that one region with suitable milling accuracy exists and that the transition between the suitable accuracy region and the rapidly decreasing accuracy region is sharp, 40 % normalized dexterity was chosen as the boundary value for accuracy contour generation.

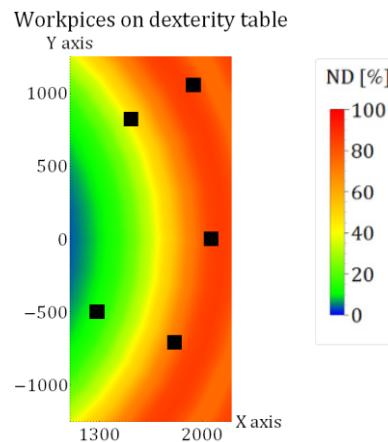
Fig. 3 shows the clamping table covered by normalized dexterity contours for vertical tool orientation. Workpieces at actual milling positions are represented by black squares scaled by 5.

**Table 3** DH notation of the discussed robot

Link	d (mm)	a (mm)	$\alpha$ (°)	$\theta$ (°)	Range (°)
0	0	0	0	0	0
1	750	350	90	0	+/-185
2	0	1250	0	0	0/-146
3	0	-55	90	90	+155/-119
4	1100	0	90	180	+/-350
5	0	0	90	90	+/-125
6	0	0	0	0	+/-350
Tool tip	A	B	C	D	E

**Table 4** Joint angles and ND at milling positions

Pos.	$\theta_1$ (°)	$\theta_2$ (°)	$\theta_3$ (°)	$\theta_4$ (°)	$\theta_5$ (°)	$\theta_6$ (°)	ND (%)
1	0	37.8	17.2	180	125	0	81.8
2	0	43.0	6.0	180	130	0	72.6
3	0	62.0	-35.7	180	153.8	0	22.2
4	0	53.0	-15.8	180	142.8	0	46.1
5	0	46.6	-1.8	180	135.2	0	63.9



**Fig. 3** Top view of workpieces on clamping table with dexterity contours

To generate the accuracy contour, regions in the robot's workspace with at least 40 % normalized dexterity were isolated. As revolution about the robot's first axis does not affect its dexterity it was possible to consider only the workspace cross section and revolve results to three-dimensional space afterwards. According to this simplification a two-dimensional restricted workspace was created, only containing points with normalized dexterity values 40 % and above. For a more general solution tool orientation was considered free.

To isolate contour points of the restricted workspace a regional non-dominated sorting algorithm was developed. The algorithm searches for non-dominated minimum points defined by x and z in the restricted workspace point cloud, which represent to the robot's tool tip positions in X and Z direction relative to the robot's base coordinate system. Extracted two-dimensional contour points form the outer boundary of the subspace in the robot's complete workspace, inside which milling with suitable accuracy is possible. Pseudocode of the algorithm is presented in Fig. 4.  $IC$  stands for iteration count,  $RC$  stands for region,  $LE$  stands for number of local extremes and  $Q1-Q4$  are bool variables representing quadrant 1-4. MA stands for mirror axis, NnD stands for non-dominated and AC stands for accuracy contour. Initialization values are presented in Table 5.

To find the first non-dominated front in the restricted workspace, which corresponds to the first iteration in Fig. 4, regular non-dominated search algorithm was applied. In the following iterations the restricted workspace was mirrored over each axis to expose outer regions and isolate front points contributing to the final accuracy contour. Due to the shape of the restricted workspace shown in Fig. 5, local extreme points in X and Z direction had to be found in order to generate an equally dense contour. In our case one local extreme point was defined in initialization dividing the concave region into two refined regions, but an arbitrary refinement level would be possible if necessary. The refined regions were restricted by squares ranging from the existing regions contour's end point to the enclosed local extreme point. The relevant mirror axis for each refined region was determined based on the adjacent region's mirror axis and the transformation Table 6, where  $Q1-Q4$  represent square quadrants in XZ plane (Fig. 5) sequenced clockwise starting at the top right corner.

---

```

1 Initialization.
2 If  $IC < 5$ :
3   If  $IC = 1 \vee Q3 = 1$ : MA = None, NnD sort,  $Q3 = 0$ ,  $IC + 1$ , go to step 2.
4   If  $IC = 2 \vee Q4 = 1$ : MA = X, NnD sort,  $Q4 = 0$ ,  $IC + 1$ , go to step 2.
5   If  $IC = 3 \vee Q2 = 1$ : MA = Z, NnD sort,  $Q2 = 0$ ,  $IC + 1$ , go to step 2.
6   If  $IC = 4 \vee Q1 = 1$ : MA = X  $\wedge$  MA = Z, NnD sort,  $Q1 = 0$ ,  $IC + 1$ , go to step 2.
7 Else:
8   If  $RC = 0$ :
9     If  $LE > 0$ :  $RC = 1$ , go to step 7.
10    If  $LE = 0$ : Go to step 13.
11    If  $0 < RC < LE + 1$ : RC boundaries. Find mirror axes.  $Q(1-4) = 1$ ,  $RC = RC + 1$ , go to step 2.
12    If  $RC = LE + 1$ : Go to step 13.
13 Join NnD regions into suitable accuracy contour.

```

---

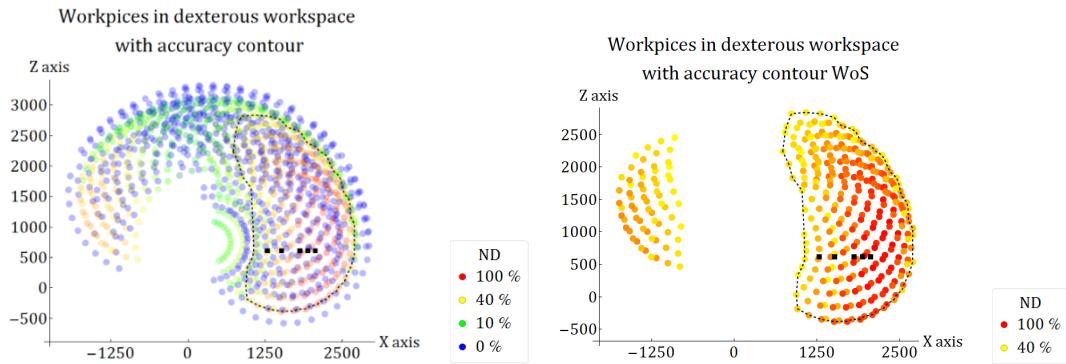
**Fig. 4** Pseudocode for accuracy contour generation

**Table 5** Regional non-dominated sorting algorithm variables and initial values

Regional NnD - variables	Unit	Value
Iteration count ( $IC$ )	-	1
Region count ( $RC$ )	-	0
Local extremes ( $LE$ )	-	1
Quadrants ( $Q1-Q4$ )	-	0

**Table 6** Transformation table for refined region definition

Quadrant	Change direction	Quadrant
Q1	↔	Q2
Q2	↔	Q1
Q3	↔	Q4
Q4	↔	Q3



**Fig. 5** Cross section of robot's complete (left) and restricted dexterous workspace (right) with projected workpiece positions and accuracy contour

After a refined region was mirrored over the relevant axis, the non-dominated sorting algorithm was applied to extract the regions contour points. Finally, all partial results were joined into the accuracy contour.

On the left side of Fig. 5 complete dexterous workspace cross section with accuracy contour and five workpiece positions is shown. Blue dots represent singularity points, which are regions with lowest dexterity, green and yellow dots represent transition regions with medium dexterity and red dots represent regions with highest dexterity. Accuracy contour is marked with a dashed black line. Workpieces are presented in the projected plane and scaled by 5. The plot on the right side shows the suitable milling accuracy region with normalized dexterity values 40 % and above, the accuracy contour and all five projected workpiece positions. For accuracy contour generation only the bigger point cloud was considered as the revolution about the robot's main Z axis covers complete 360 °.

Now, the tool orientation effect becomes visible. Dexterity contours corresponding to a fixed tool axis shown in Fig. 3 are different from dexterity contours shown in Fig. 5, where tool orientation was considered free. Free tool orientation enables another degree of freedom, which increases the size of the suitable region. By allowing free tool orientation, also the workpiece closest to robot's base is located inside the suitable milling accuracy region, meaning that all workpieces could be milled with suitable accuracy, regarding their size and assuming correct tool and technology selection.

### 3.2 Optimization of workpiece positioning

With the separation of the robot's workspace into two milling quality regions, it is interesting to find the size, position and orientation of the biggest workpiece that can be milled with suitable accuracy. As the three dimensional workspace of the robot in consideration can be created by revolving its cross section about its main Z axis, it is obvious that the biggest workpiece with eight corner points is a parallelepiped.

Genetic algorithm was applied as the search tool for the biggest parallelepiped. The algorithm was set up as an extension of the regional non-dominated sorting algorithm used to create the accuracy contour. Pseudocode of the algorithm is presented in Fig. 6 where  $IC$  stands for iteration count,  $MI$  for maximum number of iterations,  $PC$  for population count,  $PS$  for population size and  $CC$  for convergence criteria. GA stands for genetic algorithm, RP for reference point and NnD stands for non-dominated.

The algorithm requires correct initialization as shown in Fig. 6. Maximum allowed iteration number, population size, reproduction, crossover and mutation frequency, mutation strength and convergence criteria were declared initialization variables. Maximum number of iterations defines the maximum allowed loop repetitions, population size defines the quantity of parallelepipeds generated in one iteration, reproduction frequency defines the quantity of individuals to be transferred to the next generation, crossover frequency defines the fraction of individuals to participate in gene exchange and mutation frequency defines the number of individuals to mutate with a magnitude of geometrical change defined by mutation strength. Convergence criteria is an early exit criteria. Values used for initialization are presented in Table 7.

---

```

1 Initialization.
2 If  $IC = 1$ : Boundaries for first generation, go to step 5.
3 If  $1 < IC < MI$ : Boundaries for non-first generation, GA mechanics, go to step 5.
4 If  $IC = MI$ : Go to step 12.
5 Generate parallelepipeds, add RP to restricted workspace list, NnD sort.
6 If RP is on front: Discard parallelepiped, go to step 2.
7 Else: Add parallelepiped to population.
8 If  $PC < PS$ : Go to step 2.
9 Else:
10    $IC = IC + 1$ ,
11   If  $CC = \text{True}$ : Go to step 12.
12   Else: Go to step 2.
12 Find global best.

```

---

**Fig. 6** Pseudocode of the genetic search algorithm**Table 7** Search algorithm variables and their initial values

GA - variables	Unit	Value
Max. iteration number ( $MI$ )	-	100
Population size ( $PS$ )	-	20
Reproduction frequency ( $RF$ )	%	10
Crossover frequency ( $CF$ )	%	60
Mutation frequency ( $MF$ )	%	30
Mutation strength ( $MS$ )	-	0.1
Convergence criteria ( $CC$ )	-	10

To generate a parallelepiped a random corner point and two edge lengths describing a parallelogram in the  $Y = 0$  plane were created. The third dimension was added by extrusion of the parallelogram in  $Y$  direction. For the first iteration different boundary conditions to restrict parallelepiped's sizes closely to the allowed region were used as for the following iterations. In the first iteration parallelepiped's sizes, positions and orientations were generated by Eqs. 2-8:

$$x = \text{randomReal}\{x_{min} + 500, x_{max} - 500\} \quad (2)$$

$$y = 0 \quad (3)$$

$$z = \text{randomReal}\{z_{min} + 500, z_{max} - 500\} \quad (4)$$

$$l_a = \text{randomReal}\{0, 1\} \cdot \frac{x_{max} - x_{min}}{2} \quad (5)$$

$$l_b = \text{randomReal}\{0, 1\} \cdot \frac{z_{max} - z_{min}}{2} \quad (6)$$

$$l_c = \text{randomReal}\{0, 1\} \cdot \frac{x_{max} - x_{min}}{2} \quad (7)$$

$$\varphi = \text{randomReal}\{0, \pi\} \quad (8)$$

where  $x, y, z$  are the biggest parallelogram's corner point coordinates,  $\varphi$  is its orientation about the  $Y$  axis and  $l_a, l_b$  and  $l_c$  are the associated parallelepiped's edge lengths.  $x_{min}, x_{max}, z_{min}$  and  $z_{max}$  are the extreme points of the accuracy contour in  $X$ ,  $Y$  and  $Z$  direction. As the workspace is rotationally symmetrical  $x_{min}$  and  $x_{max}$  are used as extreme points in  $Y$  direction and only one orientation has to be considered variable.

After a parallelepiped was created and positioned in the robot's workspace, its reference points were added to the restricted workspace point cloud, whereby corner points of extruded corners were projected to the  $Y = 0$  plane. Because of the concave shape of the accuracy contour also edge points had to be checked by the contour criteria, therefore three equidistant reference points were created on each edge and added to the restricted workspace point cloud. Then regional non-dominated sorting was applied on the extended point cloud. If a reference point was

found non-dominated it was located outside of the allowed space and a workpiece containing such a point would fall outside of the robot's suitable milling region. To avoid bad individuals proceeding to the next generation, every parallelepiped that failed the above criteria was discarded and a new one was created instead. After as many acceptable parallelepipeds as defined by population size were created, they were sorted by relevance, where bigger volume was considered better.

For the following iterations GA mechanisms selection, reproduction, crossover and mutation were applied to create new generations of parallelepipeds. For reproduction, only the best individuals from the last generation were used. If multiple individuals were tied, a random selection among them was made. In crossover each individual from the last generation could participate multiple times. To avoid reproduction an individual could never be crossed with itself and at last one gene was exchanged by the parents. Eqs. 9-15 were used to generate new individuals by mutation, where index *old* defines the selected variables old value and *MS* defines mutation strength. Each individual from the last generation could participate in mutation with same probability.

$$x = x_{old} + \text{randomReal}\{-100, 100\} \cdot \frac{x_{max}}{x_{min}} \cdot MS \quad (9)$$

$$Y = 0 \quad (10)$$

$$z = z_{old} + \text{randomReal}\{-100, 100\} \cdot \frac{z_{max}}{z_{min}} \cdot MS \quad (11)$$

$$l_a = l_{a,old} + \text{randomReal}\{-100, 100\} \cdot \frac{x_{max}}{x_{min}} \cdot MS \quad (12)$$

$$l_b = l_{b,old} + \text{randomReal}\{-100, 100\} \cdot \frac{z_{max}}{z_{min}} \cdot MS \quad (13)$$

$$l_c = l_{c,old} + \text{randomReal}\{-100, 100\} \cdot \frac{x_{max}}{x_{min}} \cdot MS \quad (14)$$

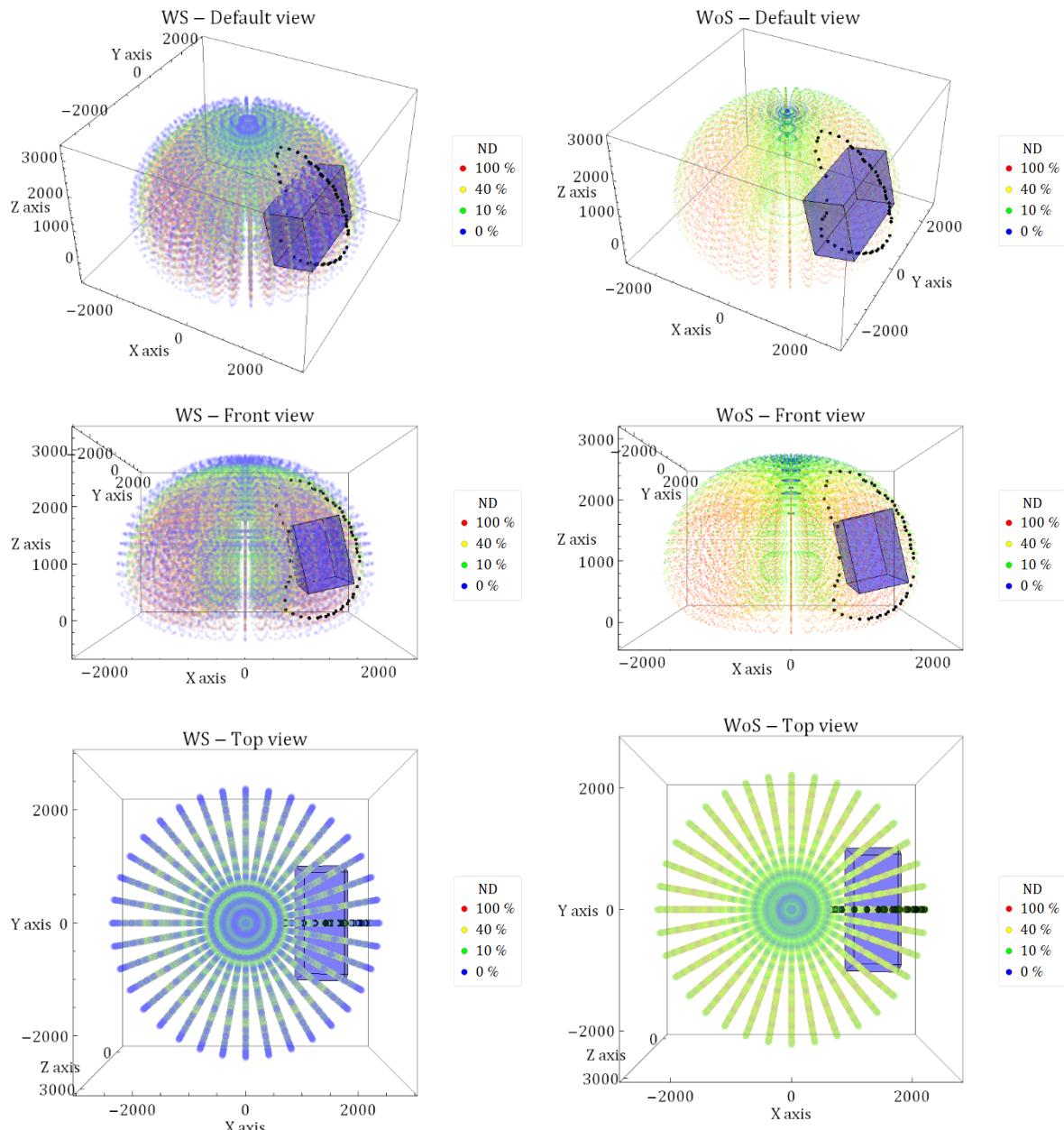
$$\varphi = \varphi_{old} + \text{randomReal}\{0, \pi\} \cdot MS \quad (15)$$

The optimization loop was repeated until a parallelepiped was found to be the best for same iteration count as defined by convergence criteria or until maximum iteration number was reached. On completion the algorithm returned the parallelepiped with the biggest volume as the optimization target. The biggest parallelepiped represents the size, position and orientation of the biggest workpiece that can be milled with the chosen accuracy with the discussed robot.

In the first run the algorithm did not converge with the defined settings. Workpieces still increased in size from generation to generation, indicating space for improvement. For this reason an efficiency mechanism was built into the code, generating a restart file if maximum iteration number was reached before convergence criteria. From the restart file a new optimization run could be started with the individuals from the last generation of the previous run adopted for the initial generation of the new optimization run. In our case, six optimization runs were performed to find the target workpiece.

Locations of two most opposite corner points of the biggest workpiece found by the applied genetic algorithm are  $A_x = 2000.23$  mm,  $A_y = -1162.99$  mm,  $A_z = 1963.87$  mm and  $G_x = 1354.13$  mm,  $G_y = 1162.99$  mm,  $G_z = 292.81$  mm. Edge lengths  $l_a$ ,  $l_b$  and  $l_c$  are  $l_a = 1002.44$  mm,  $l_b = 1486.24$  mm and  $l_c = 2325.98$  mm, where  $l_a$  is the distance between two corner points on the same edge of the parallelepiped in X direction and it is rotated by angle  $\varphi$  about the Y axis in counter clockwise direction in front view,  $l_b$  is the distance between two corner points on the same edge of the parallelepiped in Z direction and it is rotated by angle  $\varphi$  about the Y axis in counter clockwise direction in front view and  $l_c$  is the distance between two corner points on the same edge of the parallelepiped in Y direction. Workpiece rotation about the Y axis is  $\varphi = 3.37^\circ$ . The volume of the workpiece, calculated as the product of its edge lengths is  $3.47 \times 10^9$  mm<sup>3</sup>.

The workpiece is presented in Fig. 7. Graphics on the left hand side show workpiece size, position and orientation in the robot's complete workspace, while graphics on the right hand side show the same workpiece in the robot's workspace without singularities. Accuracy contour is plotted in Y = 0 plane only. Dexterity in blend colours is used for workspace presentation, but for size, position and orientation search only the accuracy contour marked in black was relevant. For each case figures are presented in default, front and in top view.



**Fig. 7** Optimized workpiece in the robot's workspace with accuracy contour: with singularity points (left), without singularity points (right)

#### 4. Conclusion

The article demonstrates a methodology, based on innovative algorithms, to isolate a refined sub region in a robot's complete workspace that allows milling with suitable accuracy. A parallelepiped shaped, maximum sized workpiece and its position and orientation in a robot's refined workspace were found as an example, but any arbitrary shaped object or a number of objects can be positioned into the refined region instead by using the proposed method.

The presented methodology can also be applied to robotic mechanisms in different manufacturing environments. Instead of defining the accuracy contour by milling measurements, other machining processes to create performance contours can be used. Contours for a robot's posture accuracy for grasping and positioning, measurement accuracy contours for tolerance control of big welded parts and many more can be created to optimally restrict a robot's workspace. Some analytical performance measures that can be used for contour generation were also discussed in this article.

It is to be noted that complex contour shapes may require a different extraction approach than the regional non-dominated sorting algorithm presented in this article. The algorithm could be used because of simplicity of the accuracy contour, containing only two inflection points that were easy to identify and was tailored for quick contour extraction as it was intended to serve as the optimization function for the search algorithm. A general contour extraction algorithm would consider each point from the initial point cloud more often and would be less suitable for workpiece size and placement optimization.

Genetic algorithm was applied for workpiece size, position and orientation search, because of good results reported for similar problems by other authors. Discard penalty was introduced for individuals violating the accuracy contour criteria. Because of the penalty each new iteration took longer to complete as workpieces from earlier generations were closer to contour boundaries and discarded more often. An algorithm with faster convergence is being developed, allowing a more general contour generation algorithm to be used as optimization function.

## Acknowledgement

The authors would like to thank the Ministry of Higher Education, Science and Technology of the Republic of Slovenia, the University of Maribor, Slovenia and the Faculty of Mechanical Engineering of Maribor, Slovenia, who made the presented study possible by financial and infrastructural support.

## References

- [1] Kubela, T., Pochyly, A., Singule, V. (2016). Assessment of industrial robots accuracy in relation to accuracy improvement in machining processes, In: *2016 IEEE International Power Electronics and Motion Control Conference (PEMC)*, Varna, Bulgaria, 720-725, doi: [10.1109/EPEPEMC.2016.7752083](https://doi.org/10.1109/EPEPEMC.2016.7752083).
- [2] Wu, Y., Klimchik, A., Caro, S., Furet, B., Pashkevich, A. (2015). Geometric calibration of industrial robots using enhanced partial pose measurements and design of experiments, *Robotics and Computer-Integrated Manufacturing*, Vol. 35, 151-168, doi: [10.1016/j.rcim.2015.03.007](https://doi.org/10.1016/j.rcim.2015.03.007).
- [3] Józwik, J., Ostrowski, D., Jarosz, P., Mika, D. (2016). Industrial robot repeatability testing with high speed camera Phantom V2511, *Advances in Science and Technology Research Journal*, Vol. 10, No. 32, 86-96, doi: [10.12913/22998624/65136](https://doi.org/10.12913/22998624/65136).
- [4] Marônek, M., Bárta, J., Ertel, J. (2015). Inaccuracies of industrial robot positioning and methods of their correction, *Tehnički vjesnik – Technical Gazette*, Vol. 22, No. 5, 1207-1212, doi: [10.17559/TV-20140416123902](https://doi.org/10.17559/TV-20140416123902).
- [5] Jezeršek, M., Kos, M., Kosler, H., Možina, J. (2017). Automatic teaching of a robotic remote laser 3D processing system based on an integrated laser-triangulation profilometry, *Tehnički vjesnik – Technical Gazette*, Vol. 24, No. 1, 89-95, doi: [10.17559/TV-20160504230058](https://doi.org/10.17559/TV-20160504230058).
- [6] Papa, G., Torkar, D. (2009). Visual control of an industrial robot manipulator: Accuracy estimation, *Strojniški vestnik – Journal of Mechanical Engineering*, Vol. 55, No. 12, 781-787.
- [7] Yoshikawa, T. (1985). Manipulability of robotic mechanisms, *The International Journal of Robotics Research*, Vol. 4, No. 2, 3-9, doi: [10.1177/027836498500400201](https://doi.org/10.1177/027836498500400201).
- [8] Gotlih, K., Kovac, D., Vuherer, T., Brezovnik, S., Brezocnik, M., Zver, A. (2011). Velocity anisotropy of an industrial robot, *Robotics and Computer-Integrated Manufacturing*, Vol. 27, No. 1, 205-211, doi: [10.1016/j.rcim.2010.07.010](https://doi.org/10.1016/j.rcim.2010.07.010).
- [9] Merlet, J.P. (2005). Jacobian, manipulability, condition number, and accuracy of parallel robots, *Journal of Mechanical Design*, Vol. 128, No. 1, 199-206, doi: [10.1115/1.2121740](https://doi.org/10.1115/1.2121740).
- [10] Mansouri, I., Ouali, M. (2011). The power manipulability – A new homogeneous performance index of robot manipulators, *Robotics and Computer-Integrated Manufacturing*, Vol. 27, No. 2, 434-449, doi: [10.1016/j.rcim.2010.09.004](https://doi.org/10.1016/j.rcim.2010.09.004).
- [11] Vosniakos, G.-C., Matsas, E. (2010). Improving feasibility of robotic milling through robot placement optimisation, *Robotics and Computer-Integrated Manufacturing*, Vol. 26, No. 5, 517-525, doi: [10.1016/j.rcim.2010.04.001](https://doi.org/10.1016/j.rcim.2010.04.001).
- [12] Aly, M.F., Abbas, A.T. (2014). Simulation of obstacles' effect on industrial robots' working space using genetic algorithm, *Journal of King Saud University – Engineering Sciences*, Vol. 26, No. 2, 132-143, doi: [10.1016/j.jksues.2012.12.005](https://doi.org/10.1016/j.jksues.2012.12.005).

- [13] dos Santos, R.R., Steffen, V. Jr., Saramago, S.F.P. (2008). Robot path planning in a constrained workspace by using optimal control techniques, *Multibody System Dynamics*, Vol. 19, No. 1-2, 159-177, doi: [10.1007/s11044-007-9059-1](https://doi.org/10.1007/s11044-007-9059-1).
- [14] Dolgui, A., Pashkevich, A. (2009). Manipulator motion planning for high-speed robotic laser cutting, *International Journal of Production Research*, Vol. 47, No. 20, 5691-5715, doi: [10.1080/00207540802070967](https://doi.org/10.1080/00207540802070967).
- [15] Lin, Y., Zhao, H., Ding, H. (2017). Posture optimization methodology of 6R industrial robots for machining using performance evaluation indexes, *Robotics and Computer-Integrated Manufacturing*, Vol. 48, 59-72, doi: [10.1016/j.rcim.2017.02.002](https://doi.org/10.1016/j.rcim.2017.02.002).
- [16] Kuss, A., Drust, M., Verl, A. (2016). Detection of workpiece shape deviations for tool path adaptation in robotic deburring systems, *Procedia CIRP*, Vol. 57, 545-550, doi: [10.1016/j.procir.2016.11.094](https://doi.org/10.1016/j.procir.2016.11.094).
- [17] Klancnik, S., Brezovnik, S., Brezocnik, M., Balic, J., Ficko, M., Pahole, I., Buchmeister, B. (2009). Advanced concept of milling with robot, *Scientific Bulletin Series C: Fascicle Mechanics, Tribology, Machine Manufacturing Technology*, Vol. 23, No. 100, 73-78.
- [18] Karimi, D., Nategh, M.J. (2016). Contour maps for developing optimal toolpath and workpiece setup in hexapod machine tools by considering the kinematics nonlinearity, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. 230, No. 9, 1572-1583, doi: [10.1177/0954405415592123](https://doi.org/10.1177/0954405415592123).
- [19] Wei, H., Tang, X.-S., Liu, H. (2015). A genetic algorithm(GA)-based method for the combinatorial optimization in contour formation, *Applied Intelligence*, Vol. 43, No. 1, 112-131, doi: [10.1007/s10489-014-0633-y](https://doi.org/10.1007/s10489-014-0633-y).
- [20] Truong-Hong, L., Laefer, D.F. (2014). Octree-based, automatic building façade generation from LiDAR data, *Computer-Aided Design*, Vol. 53, 46-61, doi: [10.1016/j.cad.2014.03.001](https://doi.org/10.1016/j.cad.2014.03.001).
- [21] Awrangjeb, M., Lu, G. (2015). A triangulation-based technique for building boundary identification from point cloud data, In: *2015 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, Auckland, New Zealand, 1-6, doi: [10.1109/IVCNZ.2015.7761536](https://doi.org/10.1109/IVCNZ.2015.7761536).
- [22] Veber, M. (2016). *Manipulability analysis of milling robot Kuka KR 150-2*, Master thesis, Faculty of Electrical Engineering and Computer Science, University of Maribor, Maribor, Slovenia.