Intelligent Static Calibration of Industrial Robots using Artificial Bee Colony Algorithm

M. A. Khanesar, Minrui Yan, Peter Kendal, Mohammad Isa, Samanta Piano, David Branson
Faculty of Engineering
University of Nottingham, UK
{ezzma5, ezxmy5, ezzpak, ezzmi2, ppzsp1, ezzdtb} @exmail.nottingham.ac.uk

***Abstract*—This paper proposes an industrial robot calibration methodology using an artificial bee colony algorithm. Open loop industrial robot positions are usually calculated using joint angle readings and industrial robot forward kinematics, where feedback control systems are then use iteratively to improve performance. This can often be time consuming and risks unstable control, so the preference is to enable as accurate open loop control as possible. Industrial robot forward kinematics include Denavit–Hartenberg (DH) parameters. However, assembly and manufacturing tolerances may result in differences between actual and nominal DH parameters. To improve industrial robot positional accuracies, it is required to better estimate its DH parameters. A highly accurate laser tracker system provides the positional measurement required to perform calibration of the DH parameters. For this purpose, a Leica AT960-MR, a laser tracker which works based on interferometry principles, is used to provide end effector 3D position measurements. An artificial Bee colony algorithm is then used to improve the cost function associated with the forward kinematic error by estimating more accurate industrial robot DH parameters. The implementation results demonstrate that using calibrated industrial robot DH parameters, it is possible to improve the open loop positional accuracies of the robot compared to uncalibrated forward kinematics mean absolute error for test data from 75.4 μm to 60.1 μm (20.3% improvement).**

Keywords—Industrial robot calibration, laser tracker system, intelligent optimization, artificial bee colony

I. INTRODUCTION

The usage of industrial robots in different industrial sectors is growing with their ever-increasing degree of collaboration, connection simplicity, ease of run, safety, operational ease, and efficiency. According to the international federation of robotics (IFR), over 2.5 million industrial robots were used in 2019 [1]. However, manufacturing and assembly tolerances result in differences between the real robot parameters and its nominal values. Such tolerances make Denavit–Hartenberg (DH) imprecise and consequently result in industrial robot geometrical forward kinematics (FK) errors. Closed loop controllers working in an iterative process may be used to improve positional accuracies. To improve the speed of response and compensate for DH parameter errors, it is required to perform geometrical calibrations to determine DH parameter values.

Kinematics is defined as the study of motion of an object neglecting its causes [2]. Geometrical robot FKs identifies the position of robot link frame relative to its original coordinate. Using industrial robot forward kinematics and current joint angle values, it is possible to determine robot current configurations including robot link positions and orientations. However, industrial robot FK depends on its DH parameters. Uncertainties caused by mismatch between real robot dimensions and its nominal values causes uncertainty in industrial robot FK. Hence, it is required to perform a geometrical calibration process to increase the positional accuracy of industrial robots.

Artificial neural networks have already been used to calibrate industrial robot FKs. The calibration of PA10 robot arm using neural networks with feedback data gathered from a Leica SMART310 [3] as well as neural networks for the calibration purpose of IRB1410 and a collaborative industrial robot using Leica AT960 and Leica AT960-MR have already existed in literature [4, 5]. Similarly, the calibration of Hyundai HH800 robot, a heavy duty industrial robot, using a laser tracker system [3, 6] is done using artificial neural network approaches. These approaches are non-parametric approaches to improve positional accuracies of industrial robot. However, parametric approaches to tune geometrical FKs of industrial robot result in the estimation of its actual physical parameter values. Moreover, the geometrical FKs of industrial robots is compact and easier to calculate as compared to black box models such as artificial neural networks.

In this paper, artificial bee colony (ABC) as a powerful metaheuristic optimisation approach is used to estimate DH parameters of an industrial robot. The ABC algorithm imitates the foraging behavior of bees in gathering nectar and uses it for optimization purposes [7, 8]. The position feedback is provided using a precise laser tracker system, Leica AT960-MR, capable of performing 3D measurements with error less than $3\mu m/m$ [9]. The position readings from the industrial robot joint angles and its forward kinematics are compared with the precise 3D position measurements from the laser tracker. The mean absolute value of positioning errors is then used as the cost function for ABC to perform optimization. The optimization results show that using the calibrated DH parameters for an industrial robot it is possible to improve the positional accuracies from $75.4\mu m$ to $60.1\mu m$ for test data (20.3% improvement). To have a performance comparison, another recent metaheuristic optimization algorithm of gravitational search algorithm (GSA) [10, 11] is chosen. It is observed that ABC algorithm slightly outperforms GSA [10, 11] for industrial robots calibration purposes. Therefore, the ABC algorithm is the suggested optimization algorithm for parametric industrial robot calibration purposes in this instance.

This paper is organized as follows: the methodology part is presented in Section II. Section III presents the experimental setup for the experiments, experimental results are provided in Section IV and concluding remarks are presented in Section V. The acknowledgement part and the references are provided in Sections VI and VII.

II. METHODOLOGY

The overall calibration methodology including industrial robot FK, the cost function associated with industrial robot FK error, and ABC algorithm which is used to perform the optimisation are explained in this section.

A. Industrial Robot DH Parameters

A serial manipulator with n number of joints has $n+1$ number of links. The link attaching the robot to the base is link 0 and the number of links is increasing up to n. There exists a local coordinate associated with each joint. DH system uses four parameters to describe the spatial relationship between the successive link coordinate frames: joint angle θ_i, link offset d_i, link length a_i, and link twist α_i (see Fig. 1).

- Joint angle θ_i: the angle between x_{i-1} and x_i axes about the z_{i-1} axis
- Link offset d_i: the distance from the origin of frame $(i-1)$ to the x_i axis along the z_{i-1} axis
- Link length a_i: is the distance between the z_{i-1} and z_{i-1} axis along the x_i axis; for intersecting axis is parallel to $z_{i-1} \times z_i$;
- Link twist α_i: is the angle between the z_{i-1} and z_i axes about the x_i axis

B. FK Model of UR5

Joint angle measurements using the rotary encoder sensors on joint shafts are used as the input to UR5 industrial robot FKs to express the Cartesian coordinates of robot within its 3D workspace. The link transformation matrix from the link i-1 to the link i using its DH parameters depends on the corresponding joint angle of the industrial robot and its D-H parameters [12, 13].

$$ {}^{i-1}_{i}T = \begin{bmatrix} cq_i & -c\alpha_i sq_i & s\alpha_i sq_i & a_i cq_i \\ sq_i & c\alpha_i cq_i & -s\alpha_i cq_i & a_i sq_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

where $q_i's, i = 1, \dots, 6$ represent the joint angle i , $\alpha_i's, i = 1, \dots, 6$, $a_i's, i = 1, \dots, 6$, and $d_i, i = 1, \dots, 6$ are other DH parameters of robot which present physical robot dimensions in terms of the angles between the links and distances (see Section 2-A). Furthermore, cq_i, sq_i, $c\alpha_i$, and $s\alpha_i, i = 1, \dots 6$ represent $cos(q_i)$, $sin(q_i)$, $cos\alpha_i$, and $sin(\alpha_i), i = 1, \dots, 6$, respectively. Overall robot transformation matrix in robot base coordinates is obtained as follows.

$$ T_e = {}^{0}_{6}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T \quad (2) $$

The values of α_i's are given as follows.

$$ \alpha_1 = \alpha_4 = -\alpha_5 = \pi/2 $$
$$ \alpha_2 = \alpha_3 = \alpha_6 = 0 \quad (3) $$

The 3D end effector coordinates are obtained as follows,

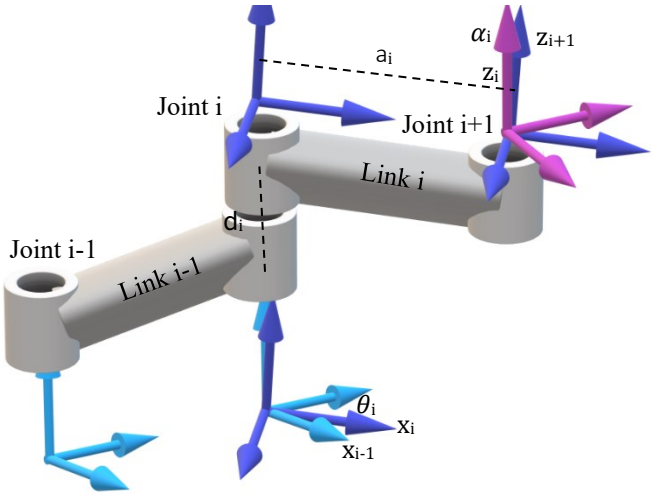

Figure 1 DH parameters

$$ \begin{aligned} x_r = {} & d_4 s_1 + a_2 c_1 c_2 + d_6 c_5 s_1 + a_3 c_1 c_2 c_3 \\ & - a_3 c_1 s_2 s_3 + d_5 c_1 c_2 c_3 s_4 + d_5 c_1 c_2 s_3 c_4 \\ & + d_5 c_1 s_2 c_3 c_4 - d_5 c_1 s_2 s_3 s_4 - d_6 c_1 c_2 c_3 c_4 s_5 \\ & + d_6 c_1 c_2 s_3 s_4 s_5 + d_6 c_1 s_2 c_3 s_4 s_5 \\ & + d_6 c_1 s_2 s_3 c_4 s_5 \end{aligned} $$

$$ \begin{aligned} y_r = {} & a_2 s_1 c_2 - d_6 c_1 c_5 - d_4 c_1 + a_3 s_1 c_2 c_3 \\ & - a_3 s_1 s_2 s_3 + d_5 s_1 c_2 c_3 s_4 + d_5 s_1 c_2 s_3 c_4 \\ & + d_5 s_1 s_2 c_3 c_4 - d_5 s_1 s_2 s_3 s_4 - d_6 s_1 c_2 c_3 c_4 s_5 \\ & + d_6 s_1 c_2 s_3 s_4 s_5 + d_6 s_1 s_2 c_3 s_4 s_5 \\ & + d_6 s_1 s_2 s_3 c_4 s_5 \end{aligned} $$

$$ \begin{aligned} z_r = {} & d_1 + a_2 s_2 + a_3 c_2 s_3 + a_3 s_2 c_3 \\ & - d_5 c_2 c_3 c_4 - d_5 c_2 s_3 s_4 + d_5 c_2 s_3 s_4 \\ & + d_5 s_2 c_3 s_4 + d_5 s_2 s_3 c_4 \\ & - d_6 c_2 c_3 s_4 s_5 - d_6 c_2 s_3 c_4 s_5 \\ & - d_6 s_2 c_3 c_4 s_5 + d_6 s_2 s_3 s_4 s_5 \end{aligned} \quad (4) $$

Although the values of FK parameters are unknown and will be estimated in this paper, their numerical values according to the robot manufacturer are as follows[1].

$$ d_1 = 0.08916m, a_2 = -0.425m, $$
$$ a_3 = -0.392m, d_4 = 0.1092m, $$
$$ d_5 = 0.0947m, d_6 = 0.0823m + d \quad (5) $$

where d is the distance between the centre of the retroreflector and the centre of the robot end-effector (see Figure 2) which is approximately equal to $0.1695m$. Furthermore, $d_2 = d_3 = 0$, and $a_i = 0$, $i = 1,4,5,6$. To conduct the calibration, the orientation of the robot is considered on its downward orientation with its tool centre point (TCP) axis-rotation vector equal to $(\pi \quad 0 \quad 0)$.

C. Artificial Bee Colony

The Artificial Bee Colony (ABC) algorithm is based on the principle of foraging which is the ecology behaviour model to predict how bees work together to search to find

[1] https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/ (visited: 1/5/2022)

the best hive and maximize the colony yield [7]. The usage of their behaviour in ABC is summarised here.

1) Real Bee Behavior

The foraging model for honeybees consists of three different components [14]. The first component is food sources. The quality of a food source depends on its distance to the hive, its nectar taste, its energy, and ease of energy extraction from its nectar [8]. The duty of employed foragers is to extract and share the information associated with the food source with other hive members such as its direction, distance, and its quality. Unemployed foragers or onlookers use the information gathered by employed foragers and exploit the food source. Knowledge exchange is an important part of the ABC algorithm which occurs using waggle dance on the dancing area within the hive [8]. Each food source is assigned a probability proportional to its quality. Onlookers choose food sources based on their associated probability. Hence, there is a high probability for onlookers to choose food sources with higher quality.

When a bee finds a food source, it memorizes its location, becomes an employed bee, and starts exploiting it. The foraging bee takes loads of nectar back to the hive and unloads it in a food store where all nectars are accumulated. The following three options are ahead of a bee after unloading the nectar:

- to become an uncommitted follower
- to communicate to other members through dancing and recruiting them to the food source
- to continue to forage for food source without recruiting after bees

2) Artificial Bee Colony in Summary

Artificial Bee colony is categorized as a swarm intelligence algorithm which imitates the honeybee foraging behavior [8]. Employed bees, onlookers, and scout bees are the three groups of bees performing optimization. Pseudocode of the ABC algorithm is given as follows:

1. initialize population as $x_i, i = 1, \dots, SN$
2. calculate the fitness associated with each member of population
3. repeat the following loop:
    a. produce new set of industrial robot DH parameters as the solutions for the optimization problem using the employed bee using $v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}), k \epsilon 1, \dots, SN, j \epsilon 1, \dots, D,$ where $\varphi_{ij} \epsilon [0, 1]$ is a uniform random number.
    b. calculate the fitness function associated with each solution $f(x_i)$ as the sum of squared positional error (see Section III)
    c. for each solution calculate its selection probability value as follows.

$$p_i = \frac{f(x_i)}{\sum_{i=1}^{SN} f(x_i)} \quad (6)$$

    d. produce the new solutions v_i for the onlookers from the solutions x_i selected depending on p_i and evaluate them
    e. apply a greedy selection process for onlookers
    f. find possible abandoned food sources for scouts and replace them with new food source using $x_{ij} = x_{i,min} + r_j(x_{i,max} - x_{i,min})$ where $r_j \epsilon [0, 1]$ is a uniform random number
    g. compare the best solution in this iteration with overall best solution and replace if necessary
    h. if maximum number of iterations is achieved stop otherwise continue the loop

D. Gravitational Search Algorithm

Gravitational search algorithm (GSA) is a powerful optimisation algorithm chosen to perform a comparison to ABC [11]. This optimisation algorithm is a physics inspired optimization algorithm which imitates the Newtonian gravitational forces between objects. In this algorithm, each object is associated with a mass value inverse proportional to its fitness function such that the object with the minimum fitness function is associated with the higher mass value. Other than a mass value, each solution benefits from a position value which is updated using velocity and acceleration. The acceleration vector is updated such that the objects with smaller weight values are accelerated towards the heavier weights. During their journey objects with smaller weight value scan space. If during their scanning phase, they come up with a better solution, their mass values are updated, and they are assigned with a higher mass value.

Each object in this algorithm benefits from several properties of mass, position, velocity, and accelerations. Solutions in d −dimensional solution space are represented by object positions.

$$X_{GSA}^i = \left(d_1^i, a_2^i, a_3^i, d_4^i, d_5^i, d_6^i\right), \; i = 1, \dots, N \quad (7)$$

The mass value corresponding to i^{th} particle at iteration number t is called the non-normalized mass value and it is represented by $m_i(t)$ [13].

$$m_{GSA}^i(t) = \frac{f(X_{GSA}^i) - f_{worst}(t)}{f_{best}(t) - f_{worst}(t)} \quad (8)$$

where $f_{worst}(t)$ is the overall worst fitness function value and $f_{best}(t)$ represents the overall best fitness function value. Therefore, $m_{GSA}^i(t)$ satisfies $m_{GSA}^i(t) \in [0,\ 1]$ with the mass value corresponding to the best solution being equal to one and the mass value corresponding to worst solution being equal to zero. The parameters $f_{worst}(t)$ and $f_{best}(t)$ are updated at every iteration as follows.

$$f_{worst}(t) = \max \{f(X_{GSA}^i(\mathrm{t}))\}_{i=1,\dots,N}$$

$$f_{best}(t) = min\{f(X_{GSA}^i(\mathrm{t}))\}_{i=1,\dots,N}$$

where x_i^j is the j^{th} component of the particle position and N is the total number of particles. Each particle mass is updated and normalized at t^{th} [13].

$$M_{GSA}^i(t) = \frac{m_{GSA}^i(t)}{\sum_{k=1}^{N} m_{GSA}^k(t)} \quad (9)$$

The overall gravitational force ($F_{GSA}^i(t)$) acting on the i^{th} particle is calculated using gravitational law of force.

$$F_{GSA}^{i}(t) = \sum_{j\in\{1,\dots,k_b\}} r_j G(t) \frac{M_{GSA}^{j}(t) M_{GSA}^{i}(t)(X_{GSA}^{j}(t) - X_{GSA}^{i}(t))}{\| X_{GSA}^{i}(t) - X_{GSA}^{j}(t) \|^{r_p} + \varepsilon}$$

where k_b is the number of selected best solutions, $\|.\|$ represents Euclidean norm, ε is a small value added to prevent singularity, r_p is the power value for the Euclidean distance between the two particles, $G(t)$ is the gravitational constant and $r_j \in [0,\ 1]$ is an uniform random value. The gravitational constant is updated at each iteration using the following equation.

$$G(t) = G_0 exp\left(-\beta \frac{t}{t_{max}}\right) \quad (10)$$

where G_0 has a constant real value and t_{max} is the maximum value of the iterations of the algorithm. Acceleration term for each object is calculated according to Newton's second law of motion by dividing the applied force to i^{th} mass by its mass value.

$$A_{GSA}^{i}(t) = \frac{F_{GSA}^{i}(t)}{M_{GSA}^{i}(t)} = \sum_{j\in\{1,\dots,k_b\}} r_j G(t) \frac{M_{GSA}^{j}(t)\left(X_{GSA}^{j}(t) - X_{GSA}^{i}(t)\right)}{\|X_{GSA}^{i}(t) - X_{GSA}^{j}(t)\|^{r_p} + \varepsilon} \quad (11)$$

where $A_{GSA}^{i}(t) \in R^d$ is d −dimensional acceleration of the particles. The velocity value correspodning to each object is updated using the acceleration term and velocity vector.

$$V_{GSA}^{i}(t+1) = p_i\ V_{GSA}^{i}(t) + A_{GSA}^{i}(t) \quad (12)$$

where $V_{GSA}^{i}(t) \in R^d$ is the d −dimensional acceleration of i^{th} particles at t^{th} iteration and $p_i \in [0,\ 1]$ is a uniform random number. Finally, the position value of each particle is updated using the previous position value and the velocity vectors as follows.

$$X_{GSA}^{i}(t+1) = X_{GSA}^{i}(t) + V_{GSA}^{i}(t+1) \quad (13)$$

III. Experiment Setup

A. Hardware Setup

The calibration test is performed on an industrial robot using a laser tracker. The detailed specifications of the industrial robot as well as the laser tracker are presented in this section.

1) Laser tracker

The calibration test is performed using a laser tracker system. The laser tracker system used in this experiment is a Leica absolute tracker AT960-MR from Hexagon metrology GMBH, Wetzlar which is a widely used measurement device in industry to inspect critical distances, locations and surfaces [9] (see Figure 2).To perform distance measurement, it is required to mount the laser target on the industrial robot. A precision Leica 1.5" red ring reflector detectable through the laser tracker is used for this purpose. The connectivity of the laser tracker is provided by Wifi connectivity to a windows 10 PC and data are collected through Spatial Analyzer® software. The operation frequency of the laser tracker is 10Hz, its maximum measurement distance is *60 m,* and its precision is $3\mu m/m$. The distance between the laser tracker and the robot base coordinate is 2.8m. The environmental working condition for the laser tracker is IP54 which guarantees ingress protection against dust and other contaminants. The laser tracker benefits from a wide operating temperature range of -15 to 45 degrees Celsius. The laser used in this system is laser class II.

B. Industrial robot

The industrial robot used in this experiment is a collaborative one which means that it can work within a close proximity to human without extra safety measures. This collaborative robot is named UR5 and is manufactured by universal robots®. It is capable of handling up to 5Kg of

(a) (b)

Figure 2 Laser tracker: a) camera b) controller

load with maximum no-load angular velocity of $180°/sec$. The connectivity of UR5 to PC is provided through Wifi connectivity and the software used to gather data is ROS Melodic operating under Linux 18.04 operating system. Joint angle values, angular velocities, and joint efforts in terms of motor currents are gathered from the industrial robot. The sample time for the data transfer from robot to PC slightly varies but its mean value is equal to $8msec$. Overall, 38 waypoints are programmed for the robot, and it travels them linearly in *600 sec*. It is required to resample position data from the robot to match laser tracker frequency (10Hz).

1) Data Resampling and Synchronization

As the data recording start time and sample time of the laser tracker and the industrial robot are different, it is required to shift and resample data from UR5 to synchronize them with the laser tracker data. The points at which the linear velocities of the robot are less than *2mm/sec* are extracted.

2) Cost Function to be Optimized

The cost function used to estimate robot DH parameters include mean absolute position error associated with its FK constructed upon the DH parameters given by intelligent optimization algorithms either ABC or GSA. The parameters suggested by either GSA [11] or ABC are used to find the industrial robot coordinate within 3D workspace out of its joint angle values. The industrial robot positions in laser tracker coordinate are calculated using an appropriate transformation matrix as follows,

$$[x_{rl} \quad y_{rl} \quad z_{rl}]^T = Tr_{rl}[x_{rr} \quad y_{rr} \quad z_{rr} \quad 1]^T \quad (14)$$

where x_{rl}, y_{rl}, and z_{rl} are the robot end-effector positions using laser tracker in laser tracker coordinate, Tr_{rl} is the transformation matrix from the robot base coordinate

system to the coordinate system of the laser tracker. The transformation matrix Tr_{rl} can be easily calculated using a least squares algorithm. Using the calculated transformation matrix Tr_{rl}, the end effector positions in laser tracker coordinates are calculated as follows.

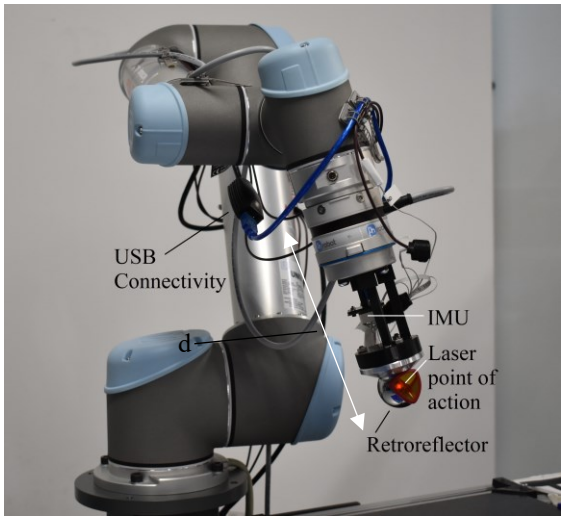

Figure 3 UR5 with retroreflector mounted on it as the target for laser tracker

$$\begin{bmatrix} x'_{rl} \\ y'_{rl} \\ z'_{rl} \end{bmatrix} = Tr_{rl} \begin{bmatrix} x_{rr} \\ y_{rr} \\ z_{rr} \\ 1 \end{bmatrix} \quad (15)$$

where x'_{rl}, y'_{rl}, and z'_{rl} represent the robot end effector position in *x, y,* and *z* axis using robot joint encoders in laser tracker coordinates. The position errors (er) is calculated as follows.

$$er = \sqrt{(x'_{rl} - x_{rl})^2 + (y'_{rl} - y_{rl})^2 + (z'_{rl} - z_{rl})^2}$$

The overall mean absolute value of position error for all measured points associated with each of these industrial robot FKs need to be calculated using (11). Finally, the cost function obtained as the mean value over all measured points is optimised using either ABC and GSA optimisation algorithms.

IV. Experimental Results

Joint angle values gathered from the UR5 and 3D position data for the UR5 end effector are gathered from the laser tracker. The data sample measurements with linear velocities less than 2mm/sec are used for quasi-static calibration. Totally 209 data samples are studied from which 70% is used for train and the rest is the test data. The total population number (*Np*) for ABC is equal to 150 and its total iteration number is equal to 300. The parameters chosen for the GSA are as follows.

$$r_p = 1, \varepsilon = 2.22 \times 10^{-16}, \beta = 20$$

$$k_b = 2, t_{max} = 200, N = 150$$

The mean absolute value of measurement error for the uncalibrated FK, FK calibrated with GSA, and FK calibrated with ABC are presented in Table 2. While the mean absolute error (MAE) value of positions for FK calibrated using ABC for the train data is 55.2 μm, the MAE value of positions for uncalibrated FK is equal to 70.2μm. Hence, using the calibration algorithm in this paper which benefits from ABC, it is possible to improve the uncalibrated positional accuracies for 21.4%. The optimization results are validated by using the test data which shows the generalization capabilities of the proposed calibration algorithm which benefits from ABC algorithm. It is observed that for the test data MAE decreases from 75.4 μm to 60.1 μm which is 20.3% improvement. It is further observed that ABC algorithm slightly outperforms

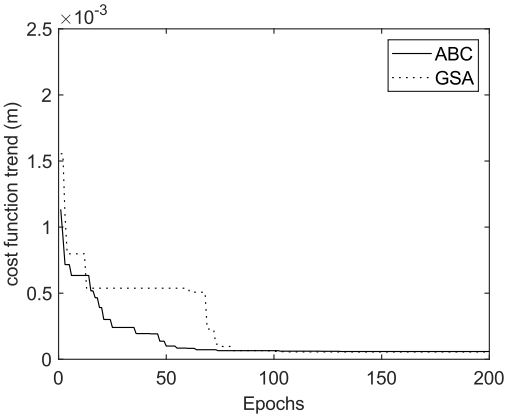

Figure 4 Optimization trend for ABC and GSA

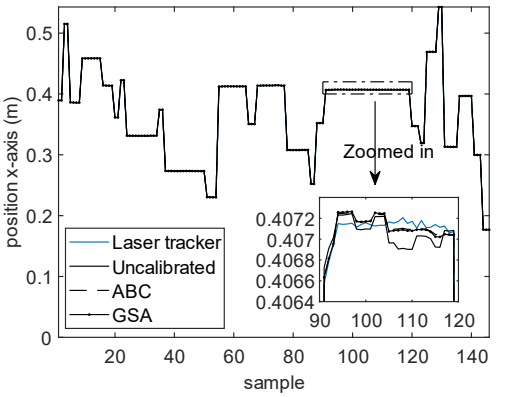

Figure 5 Results of calibrations in x-axis

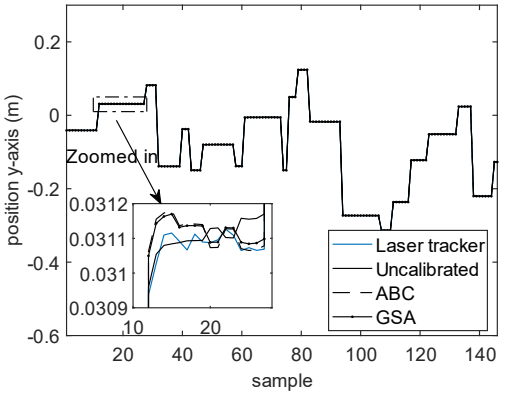

Figure 6 Results of calibrations in y-axis

Table 1 Calibration performance of the optimization algorithms

Optimization method	MAE (train)	MAE (test)
ABC	55.2 μm	60.1 μm
GSA	56.0 μm	60.9 μm
Uncalibrated	70.2 μm	75.4 μm

Table 2 Calibration results for ABC algorithm

Performance indexes		Calibrated (μm)			Uncalibrated (μm)		
		train	test	overall	train	test	overall
MAE	X	62.0	64.7	63.4	95.9	90.3	94.6
	Y	53.2	56.7	54.4	64.0	77.5	68.0
	Z	50.4	58.9	52.8	50.7	58.4	53.3
	3D	55.2	60.1	56.9	70.2	75.4	71.9
σ_i	X	74.5	80.0	76.5	125.5	117.8	124.3
	Y	75.3	73.9	75.8	94.3	105.1	99.2
	Z	64.2	74.2	67.8	64.2	73.5	67.6
	3D	71.5	76.1	73.5	97.9	100.6	99.8

GSA for calibration purpose of industrial robot FK. The results of calibration in each dimension are presented in Table 1. As can be seen from the table, the calibration error along x dimension is slightly better than other dimensions. Figures 4-6 illustrate the positions obtained after calibration using ABC, and GSA. As can be seen from these figures, using ABC algorithm for calibration, industrial robot positions are closer to the laser tracker measurements as compared to uncalibrated positions and positions using FK calibrated with GSA. Figure 8 illustrates the UR5 end-effector positions in x-y and x-z plane for the test data. As can be seen from the figure, the end-effector positions using the proposed calibration algorithm are closer to the positions measured using the laser tracker.

V. Conclusions

One of the major sources of open loop error in industrial robot FK is the robot dimension error. In this paper using laser tracker 3D position measurements, more accurate DH parameters for industrial robot are estimated. The algorithm used for this estimation is ABC algorithm which is a powerful optimization algorithm. The laser tracker used in this experiment is a high precision non-contact metrology equipment one. Using the optimization algorithm of ABC for calibration purposes, it is possible to decrease the industrial robot positioning error from MAE 75.4 μm to 60.1 μm which is a 20.3% improvement. It is further observed that FK calibrated with ABC slightly outperform the one calibrated with GSA. As a future work, motivated by the high performance of the calibration algorithm investigated in this paper, it will be used to calibrate other industrial robots. More intelligent optimization algorithms for robot moving at higher speed will be investigated in future research as well.

VI. Acknowledgement

This work is funded and supported by the Engineering and Physical Sciences Re-search Council (EPSRC) under grant number: EP/T023805/1— High-accuracy robotic system for precise object manipulation (HARISOM).

VII. References

[1] F. Compagnucci, A. Gentili, E. Valentini, and M. Gallegati, "Robotization and labour dislocation in the manufacturing sectors of OECD countries: a panel VAR approach," *Applied Economics,* vol. 51, no. 57, pp. 6127-6138, 2019.

[2] K. Russell, J. Q. Shen, and R. S. Sodhi, *Kinematics and Dynamics of Mechanical Systems: Implementation in MATLAB® and SimMechanics®*. CRC Press, 2018.

[3] S. Aoyagi, A. Kohama, Y. Nakata, Y. Hayano, and M. Suzuki, "Improvement of robot accuracy by calibrating kinematic model using a laser tracking system-compensation of non-geometric errors using neural networks and selection of optimal measuring points using genetic algorithm," in *2010 IEEE/RSJ International conference on intelligent robots and systems*, 2010: IEEE, pp. 5660-5665.

[4] M. Bai, M. Zhang, H. Zhang, M. Li, J. Zhao, and Z. Chen, "Calibration Method Based on Models and Least-Squares Support Vector Regression Enhancing Robot Position Accuracy," *IEEE Access,* vol. 9, pp. 136060-136070, 2021.

[5] Q. K. Duong, T. T. Trang, and T. L. Pham, "Robot Control Using Alternative Trajectories Based on Inverse Errors in the Workspace," *Journal of Robotics,* vol. 2021, 2021.

[6] H.-N. Nguyen, J. Zhou, and H.-J. Kang, "A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network," *Neurocomputing,* vol. 151, pp. 996-1005, 2015.

[7] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied soft computing,* vol. 8, no. 1, pp. 687-697, 2008.

[8] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review,* vol. 42, no. 1, pp. 21-57, 2014.

[9] S. Kyle, "Operational features of the Leica laser tracker," 1999.

[10] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information sciences,* vol. 179, no. 13, pp. 2232-2248, 2009.

[11] E. Rashedi, E. Rashedi, and H. Nezamabadi-Pour, "A comprehensive survey on gravitational search algorithm," *Swarm and evolutionary computation,* vol. 41, pp. 141-158, 2018.

[12] K. Kufieta, "Force estimation in robotic manipulators: Modeling, simulation and experiments," *Department of Engineering Cybernetics NTNU Norwegian University of Science and Technology,* 2014.

[13] J.-D. Sun, G.-Z. Cao, W.-B. Li, Y.-X. Liang, and S.-D. Huang, "Analytical inverse kinematic solution using the DH method for a 6-DOF robot," in *2017 14th international conference on ubiquitous robots and ambient intelligence (URAI)*, 2017: IEEE, pp. 714-716.

[14] V. Tereshko and A. Loengarov, "Collective decision making in honey-bee foraging dynamics," *Computing and Information Systems,* vol. 9, no. 3, p. 1, 2005.

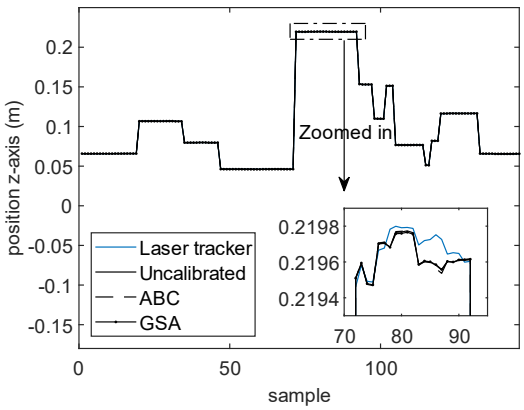

Figure 7 Results of calibrations in z-axis

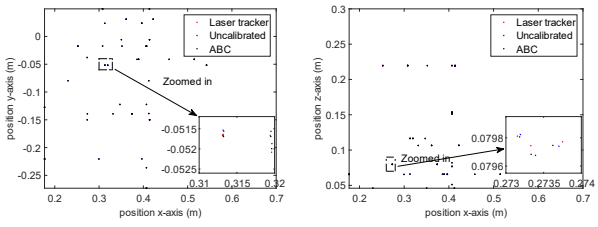

Figure 8 a) validation results on x-y plane b) validation results on x-z plane