Hybrid Calibration of Industrial Robot Considering Payload Variation

Maxime Selingue, Adel Olabi, Stéphane Thiery, Richard Bearee

Hybrid calibration of industrial robot considering payload variation

Maxime Selingue[1*], Adel Olabi[1], Stéphane Thiery[1] and Richard Béarée[1]

[1]Arts et Metiers Institute of Technology, LISPEN, HESAM Université, F-59000 Lille, France.

*Corresponding author(s). E-mail(s): maxime.selingue@ensam.eu;
Contributing authors: adel.olabi@ensam.eu;
stephane.thiery@ensam.eu; richard.bearee@ensam.eu;

Abstract

Absolute accuracy of industrial robot is required for most of industrial applications. However, positioning errors of several millimeters are induced by many factors. Hybrid calibration, combining analytical model and learning-based regression, can compensate for most of the positioning error, including payload effects. However, when the payload changes, hybrid calibration has to be performed again. In this paper, hybrid calibration is applied on an industrial robot in two different sub-workspaces, with two different payloads. The results of this method have been compared to other calibration approaches, and highlight that hybrid calibration provides a higher final accuracy. Moreover, two data-efficient and pragmatic approaches are proposed, to address the issue of changing payload. Both methods are based on hybrid calibration. The first one uses previously-acquired knowledge to drastically reduce the number of measurements necessary to update a trained learning model with another payload. The second one uses a model trained separately for two different payloads and interpolates the outputs to compensate for new payloads without any additional measurement. The datasets used are available at: https://doi.org/10.57745/DWUC0H. The methods have been experimentally validated using a compensation algorithm and compared to other approaches, and show that the positioning error can be reduced by 95%.

Keywords: robot's accuracy, hybrid calibration, artificial neural network, industrial manipulators

1 Introduction

In the context of industry 4.0, the use for industrial robots evolved from traditional pick-and-place applications to more complex operations such as machining, drilling and assembly. To meet the requirements of those tasks, robots must be accurate. However, industrial robots are known to have good repeatability but poor absolute accuracy. For tasks requiring high accuracy, robots are usually programmed using online teaching methods. With the growing demand for digital twins, robots are more and more programmed off-line. However, the effectively reached cartesian position of the end-effector can deviate from the expected one because of various factors. Among these factors, there are geometric errors, coming from manufacturing and assembly tolerances of the robot's parts; robot's link compliance; backlash, compliance in robot's joints and thermal effects. In the literature, calibration methods are proposed to reduce the positioning error, through four steps: modeling, measurement of the end-effector's position, identification of the model's parameters, compensation of the positioning error. Calibration methods can be classified in two categories: *model-based* and *model-less* calibration [1].

Model-based calibration relies on an accurate physical model of the phenomena that reduce a robot's positioning accuracy. The most known *model-based* calibration method is the geometric calibration. It is based on the identification of the geometric parameters (i.e. position and orientation of each joint with regard to the previous one) [2–4]. This model can be enhanced with stiffness identification, which enables self-mass and payload compensation. The compliant behavior of industrial robots can be modeled by a linear torsion spring on each joint of the robot. This model is known as the Virtual Joint Method (VJM). In [5], the effect of joint stiffness on the positioning error was modeled by implicitly linearizing the relationship between torque applied and axis deformation. Thus, stiffness identification requires either torque measurement on each joint or torque estimation thanks to the mass model of the robot. In [6], authors proposed a method to identify joint stiffness based on the mass model (i.e. mass and center of gravity of each link) of the robot. In [7], the joint stiffness of an industrial robot are identified using a global identification method; while in [8], each stiffness was identified individually. Later, in [9], authors identified the stiffness of a collaborative robot using its torque sensors. In [10], stiffness of each axis of an ABB IRB 1600 was identified using a Laser Tracker and a loaded double ball bar system.

The previously described methods that deal with payload effect compensation present drawbacks. First, they are only valid in specific conditions, e.g. static positioning with relatively small payload variation, due to the non-linear relationship between torque applied and axis deformation [11, 12]. Second, they are difficult to apply in practice, as they require either the mass model of the robot or torque sensor on each joint, which are rarely available. In addition to VJM, other methods exist to compensate for the compliant behavior of industrial robots. In [13], a Matrix Structural Analysis (MSA) method was proposed. It extended the VJM in that each axis can deflect in any of the six

possible directions (displacement or rotation). However, the number of parameters to identify is significantly higher, depending on the degrees of freedom of the composing element [14]. Finally, Finite Elements Analysis methods are very effective for payload compensation [15–17], nonetheless they require to have an accurate CAD model of the robot, which is not the case for most of industrial robots.

Model-less calibration relies on a "blackbox" model used to compensate for positioning error. As Artificial Neural Networks (ANNs) are known to be an effective tool for complex regressions, they are often used for positioning error compensation. In [18], authors showed in simulation that ANNs were able to compensate for kinematics error of a 2 degrees of freedom robot. In [19], a method based on an ANN was proposed. The ANN predicts the angular offset to apply to each one of the joints in function of the desired cartesian position. The method had been tested in simulation and showed a significant reduction of the positioning error. In [20], authors showed that only a few amount of data is required to train an ANN for error compensation in a small area. In [21], a BP-neural networks was used to directly predict the position of the end-effector, given a joint position, for a robotic poly-articulated arm coordinate measuring machines.

Recently, hybrid calibration methods were presented in the literature. The principle of these methods is to combine *model-based* calibration for some well-known errors, such as geometric errors and/or joints compliance, and to approximate the residual errors with *model-less* calibration. In [22], authors identified the geometric parameters of a PA-10 robot and used an ANN to predict the angular offset to apply, given a joint position, so that the desired cartesian position is reached accurately. In [23, 24], authors used an ANN to predict the difference between real measurements and an advanced model of the robot, with geometric parameters and stiffness identification. The method was tested on a HH800 robot to compensate for only robot's self-mass (i.e. no payload). In [25], authors trained an ANN for error prediction using massive measurements, made possible by the automation of the measurement process with a trajectory that keeps the contact between the laser tracker and the reflector. In [26], authors showed that hybrid calibration can reduce the positioning error for a 5-bar parallel robot. They optimized the training phase of the neural network to reduce the number of data required using a Levenberg-Marquardt combined with accelerated particle swarm for weights optimization. In [27], authors improved the positioning and the orientation accuracy of an industrial robot using ANN and geometric parameters identification across the whole workspace of the robot, by using massive data. In [28], an hybrid calibration method also based on geometric parameters identification and artificial neural network was proposed for payload compensation. As the payload was an entry neuron of the ANN, it had to be trained with 5 different payloads, leading to a complex and time-consuming experimental process.

One current concern in industry is the data-efficiency of measurement and calibration processes. To meet this specification, several researches focused

on optimal planning of *model-based* calibration experiments, using observability indices (for geometric calibration [22, 29, 30] or elastostatic calibration [31, 32]). These methods focus on carefully choosing the training data, to maximize their contribution in the fitting process. On the other hand, fine-tuning and transfer learning are commonly used methods in machine learning applied to robotics, in order to limit the number of measurements needed. The general idea is to first train machine learning models using simulation data, and complete the training on a few real data. These methods are known as sim-to-real transfer learning. In [33], an Adaptive Domain Adversarial Neural Network with Dual-Regressions model was trained to compensate for deflection caused by the robot's payload. It was first trained with simulated data, generated with a common VJM with identified stiffness. Then, training is completed using a few real data. Sim-to-real transfer learning requires an estimation of some parameters (e.g. axis stiffness in [33], friction and center-of-mass of links in [34]), which are rarely publicly available, and hence involves model parameters identification through measurements beforehand.

In our previous work [35], the interest of hybrid calibration methods have been experimentally shown on a collaborative robot, the KUKA *iiwa*. A comparison between model-based, model-less and hybrid calibration have been made. The workspace has been divided into sub-workspaces, leading to a light ANN architecture. Moreover, since the robot handled a 5kg payload during the measurements, it has been shown that hybrid calibration methods can compensate for errors due to the payload and the joints compliance.

In this paper, the method described in [35] have been applied to a traditional industrial robot, the KUKA KR300, and extended to tackle the issue of a changing payload with two pragmatic and data-efficient approaches. The main objective is to propose methods that allow a fast and data-efficient reconfiguration of existing calibration models, while maintaining a high level of accuracy. The first method adapts, in a real-to-real transfer learning fashion, an existing hybrid model to a new payload using only a few measurement, making the experimental measurement process faster. The second method uses a model previously trained for two different payloads and interpolates the outputs to compensate for new payloads, so that no additional measurements are necessary. Compared to state-of-the-art methods, the ones presented hereafter focus on updating an existing and in-use hybrid model to a new environment, here a new payload, with minimum time, material, data, and experimental setup complexity. In this way, the proposed methods match the main concerns of robots end-users, that are fast reconfigurability and accuracy. Besides, contrary to most of state-of-the-art methods for robotic calibration, the ones developed in this paper require only a position measurement device, i.e. no torque sensors, CAD model or mass model of the robot are needed. Hence, the proposed methods are suitable for any industrial manipulator. Experimental validations have been carried out on a traditional industrial robot, a KUKA KR300, and the performances of the proposed hybrid calibration approaches

have been discussed and compared to the one from [35], obtained on a smaller, more compliant robot, a KUKA *iiwa*.

The hybrid calibration method is composed of two steps: first, global geometric parameters identification compensates for the geometric errors; second, an ANN is used for residual error approximation. The principle of the hybrid calibration is summarized in Figure 1: to compensate for any positioning error, the nominal Inverse Kinematics (IK) function is replaced by a compensation algorithm that relies on both a more accurate model than the nominal forward kinematics one, and an ANN that is to approximate the residual positioning error. In Section 2, the hybrid calibration method is presented, experimentally tested and compared to both model-based and model-less calibration methods. Moreover, the method has been tested while the robot handled two different payloads $P_1 = 4kg$ and $P_2 = 120kg$, in two different sub-workspaces, to assess the effectiveness of the method in various sub-workspaces with various payloads. This leads to four different measurements configurations for ANN's training, then to four different set of weights. The hybrid's model final accuracy is close to the robot's repeatability, similarly than in [35]. However, if the payload changes, the hybrid model is no longer valid, since the ANN's have been trained for a specific payload. For such scenarios, instead of applying the complete hybrid calibration method described previously, two novel methods based on hybrid models are proposed in Section 3. These two methods have been evaluated in simulation and compared to other approaches, and experimentally validated using the compensation algorithm in Section 4. Finally, Section 5 concludes.

2 Hybrid calibration

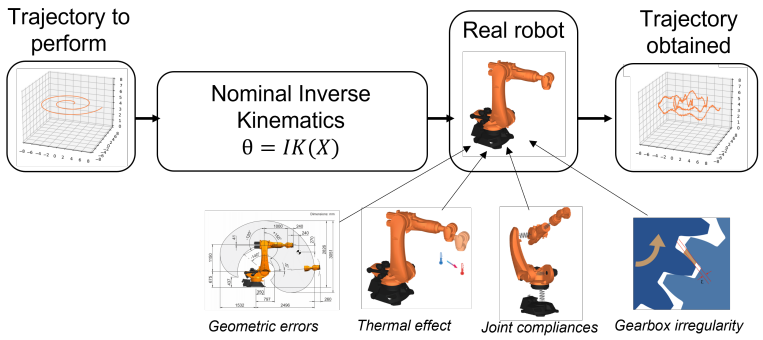
Calibration methods consist in using a compensation algorithm to reduce the positioning error through a modeling of the sources of errors. This algorithm will be presented in Section 4. The hybrid model proposed in this paper relies on a global identification of the geometric parameters and an artificial neural network for residual error approximation in local sub-workspaces. Measurements have been made in two different sub-workspaces, denoted S_1 and S_2, with two different payloads $P_1 = 4kg$ and $P_2 = 120kg$.

2.1 Geometric model of the robot

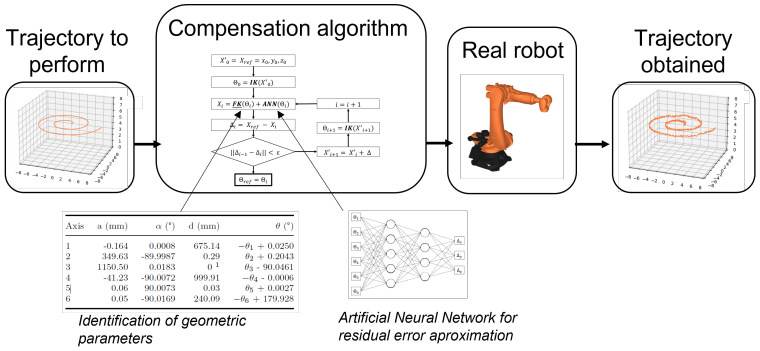
The geometric model used is the DH-model [36], in which four parameters (α, a, r, θ) are used to describe the transformation between two consecutive joints. The DH parametrization of the KR300 is depicted in Figure 2, and its nominal geometric parameters are presented in Tab.1.

The geometric parameters of the robot are gathered in a vector denoted ξ. The transformation matrix between joint $i-1$ and i is:

$$T_{i-1}^{i} = \boldsymbol{Rot}(\overrightarrow{x_{i-1}}, \alpha_i) \cdot \boldsymbol{Trans}(\overrightarrow{x_{i-1}}, a_i) \cdot$$



(a) Off-line programming without calibration using nominal Inverse Kinematics



(b) Off-line programming with hybrid calibration using compensation algorithm

Fig. 1: Hybrid calibration principle

$$\boldsymbol{Rot}(\overrightarrow{z_i}, \theta_i) \cdot \boldsymbol{Trans}(\overrightarrow{z_i}, r_i) \quad (1)$$

where $\overrightarrow{x_{i-1}}$ and $\overrightarrow{z_i}$ are defined accordingly to [37], and depicted on Figure 2. The cartesian position of the end-effector X is given by the serial multiplication of these transformation matrices over all axes for a given joint position θ. For an n-axes robot, the forward kinematics (FK) is:

$$X = FK(\xi, \theta) = T_0^n(\xi, \theta) = \prod_{i=1}^{n} T_{i-1}^i \quad (2)$$

However, since the nominal geometric parameters $\xi_{nominal}$ are inaccurate (due to manufacturing and assembly tolerances), they must be identified through geometric calibration [38]. To obtain the identified geometric parameters $\xi_{identified}$, the generalized Jacobian matrix, denoted J_ξ, is computed. After having measured a set of cartesian positions $x_{i,\,measured}$, the theoretical cartesian position $x_{i,\,th}$ is computed from the commanded joint positions θ_i, using Eq.(2). Then, $\Delta\xi = J_\xi \backslash \Delta x$, where $\Delta x = x_{measured} - x_{th}$ is numerically computed iteratively, until Δx no longer evolves. The identified geometric

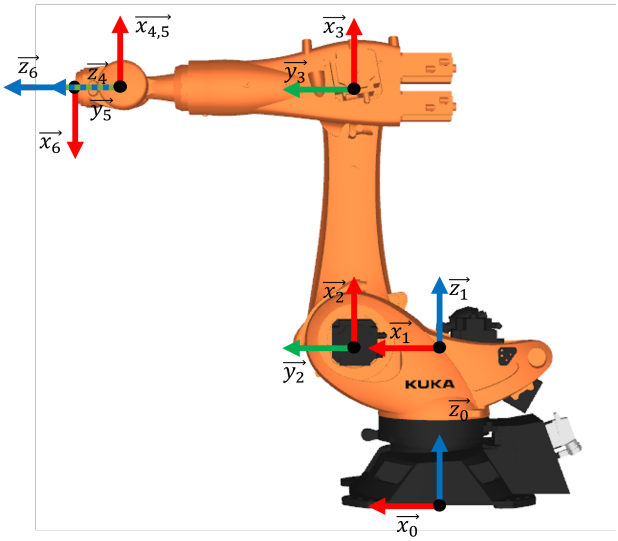


Fig. 2: DH parametrization of the KR300

Table 1: DH model of the KR300 robot (nominal geometric parameters)

Joint	a (mm)	α (°)	d (mm)	θ (°)
1	0	0	675	$-\theta_1$
2	350	-90	0	θ_2
3	1150	0	0	θ_3 - 90
4	-41	-90	1000	$-\theta_4$
5	0	90	0	θ_5
6	0	-90	240	$-\theta_6$ + 180

parameters are then:

$$\xi_{identified} = \xi_{nominal} + \Delta\xi \tag{3}$$

and $\xi_{identified}$ have been obtained from 60 measurements made across the whole workspace of the robot. Identified geometric parameters are shown in Tab.2.

2.2 Artificial neural network for residual error compensation

The geometric model described previously does not take into account other phenomena that produce positioning errors, such as joints compliance, thermal effects, kinematic errors in gearboxes etc. As ANNs are known to be an effective tool for complex regression, they are used for approximating the residual positioning error.

The proposed ANN's architecture is summarized in Figure 3. It takes as input the joint position of the robot, and has to predict the difference between the associated measured cartesian position and the theoretical cartesian position, given by the analytical model (i.e. Eq.(2) with $\xi_{identified}$). The chosen architecture is similar to the one described in [25]: the ANN has an entry layer of 6 neurons (as the robot has 6 joints), two hidden layers of 80 and 40 neurons, with respectively *tanh* and *elu* activation function, and an output layer of 3 neurons with no activation function. Weights are initialized with the Glorot initializer [39], and optimized using *adam* optimizer, with a learning rate of 0.001. The loss function is the Mean Squared Error (MSE). The datasets are composed of 1100 joint positions and their associated measured positioning error. Datasets are randomly split into 1000 data for training and 100 data for model validation. The neural network has been trained over 100 epochs. ANN's hyper-parameters tuning have been performed empirically without observing any significant variation of the model's accuracy. Moreover, hyper-parameters tuning have been performed with a grid-search method in [25], where only the training time have been significantly impacted. The architecture used in this paper is trained within a minute on a common laptop.

In this paper, the robot's workspace has been divided into sub-workspaces, and in each sub-workspace, the robot have been calibrated for two different payloads. ANN's training in a specific sub-workspace S with a payload P leads to a corresponding set of weights denoted $w_{P,\,S}$.

2.3 Experimental setup for measurements

To get all the required data for the identification of the geometric parameters and the training of the neural network, different measurement devices could be used, such as 3d camera [40], Laser Scanners [41] or Laser Trackers [9, 25, 27]. In this paper, a Laser Tracker API T3 and a Spherical Mounted Reflector (SMR) have been used. Its accuracy is 0.015mm + 0.005mm/m, and the repeatability of the robot is 0.06mm (according to the manufacturer). The described calibration method have been performed in two different sub-workspaces, with two different payloads $P_1 = 4kg$ and $P_2 = 120kg$ for each. Consequently, there is one dataset for the global identification of the geometric parameters (made up of 60 points) and four datasets for ANN's training (made up of 1100 points each). The datasets are publicly available[1].

The robot configuration during the measurements of the datasets is depicted in Figure 4 for both sub-workspaces, and the measurement sets are represented in Figure 5. S_1 is located between $x = 1200$, $y = -500$, $z = 1450\ mm$ and $x = 1600$, $y = 500$, $z = 1900\ mm$ in the robot's base frame, while S_2 is located between $x = -250$, $y = 1050$, $z = 1000\ mm$ and $x = 300$, $y = 1400$, $z = 1600\ mm$. Both sub-workspaces have a comparable volume. The choice of working with several sub-workspaces has been made for a fast reconfiguration of the experimental process. Concretely, for industrial

[1] https://doi.org/10.57745/DWUC0H

Table 2: DH Model of the KR300 robot (identified geometric parameters)

Joint	a (mm)	α (°)	d (mm)	θ (°)
1	-0.164	0.0008	675.14	$-\theta_1$ + 0.0250
2	349.63	-89.9987	0.29	θ_2 + 0.2043
3	1150.50	0.0183	0 [1]	θ_3 - 90.0461
4	-41.23	-90.0072	999.91	$-\theta_4$ - 0.0006
5	0.06	90.0073	0.03	θ_5 + 0.0027
6	0.05	-90.0169	240.09	$-\theta_6$ + 179.928

[1]This parameter cannot be identified because of a loss of rank in the generalized Jacobian, as explained in [42]

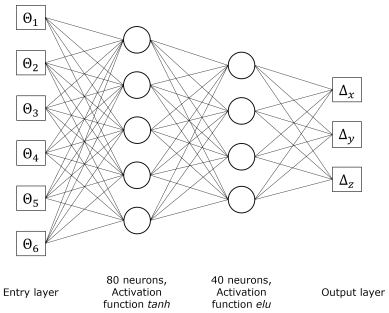
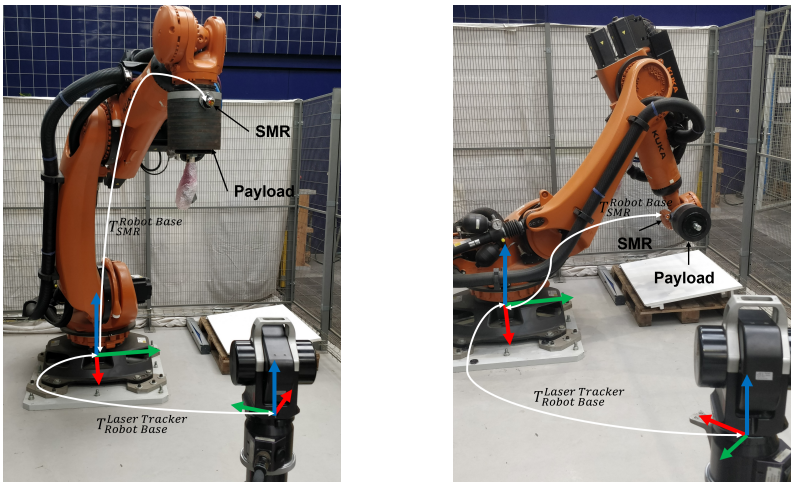


Fig. 3: Architecture of the used ANN

deployment of the proposed methods, sub-workspaces can be defined wherever the robot has to perform accurately. The robot has been warmed up during three hours before measurements, to avoid any unexpected intern thermal variation during the process, as advised in [43]. After training this ANN separetely in both sub-workspaces with both payloads, four different sets of weights w_{P_1,S_1}; w_{P_1,S_2}; w_{P_2,S_1} and w_{P_2,S_2} are obtained. Depending on the sub-workspace in which the task is performed, and the payload, the corresponding set of weights can be dynamically loaded. Recording dataset required approximately 2.5 hours for each. In addition, setting up the Laser Tracker takes 30 minutes.

2.4 Evaluation of the model's accuracy

Before any experimental validation using the compensation algorithm, the model is evaluated in forward kinematics, using 100 measurements. After geometric parameters identification and ANN's training, the theoretical cartesian position given by the model from the commanded joint position is compared to the measured one. Four different models have been evaluated and compared, in the two sub-workspaces S_1 and S_2 and with the two payloads $P_1 = 4kg$ and $P_2 = 120kg$:



(a) Robot in S_1 (b) Robot in S_2

Fig. 4: Configuration of the robot during the measurements.

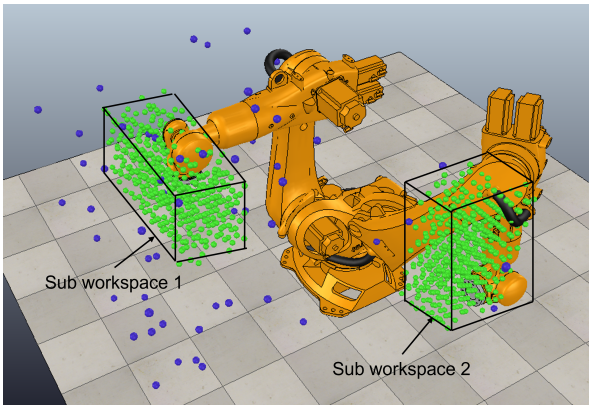


Fig. 5: Measurements repartition for geometric parameters identification (blue) and ANN's training (green)

- Nominal Geometric Parameters (NGP) : raw accuracy of the robot

$$X = FK(\xi_{nominal}, \Theta) \tag{4}$$

- Identified Geometric Parameters (IGP) : model-based calibration

$$X = FK(\xi_{identified}, \Theta) \tag{5}$$

Table 3: Accuracy of models (4) to (7) in the four different conditions of measurement for the KR300

Method	Sub-workspace 1				Sub-workspace 2			
	Mass P_1		Mass P_2		Mass P_1		Mass P_2	
	Mean error	Max error	Mean error	Max error	Mean error	Max error	Mean error	Max error
NGP	4.230	5.125	4.554	5.67	2.831	3.975	3.090	4.43
IGP	0.399	0.57	0.189	0.381	0.299	0.408	0.483	0.611
NGP + ANN	0.083	0.155	0.09	0.204	0.084	0.261	0.124	0.401
IGP + ANN	0.078	0.199	0.083	0.17	0.058	0.125	0.071	0.217

- NGP + ANN : model-less calibration

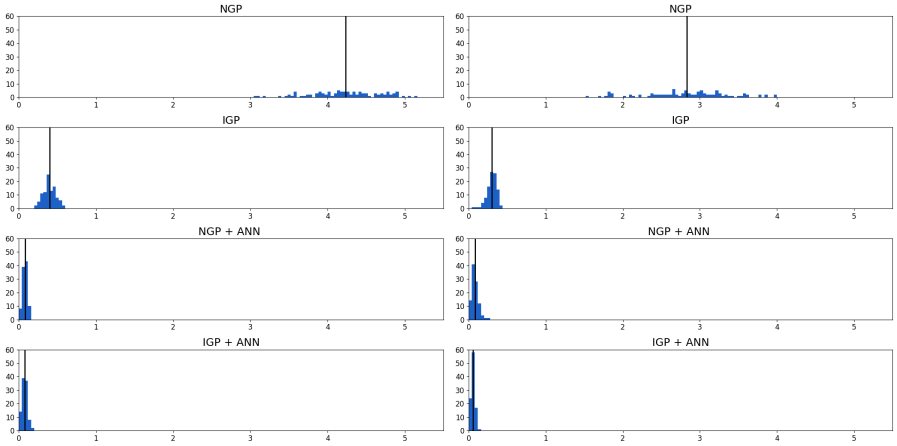
$$X = FK(\xi_{nominal}, \Theta) + ANN(\Theta) \tag{6}$$

- IGP + ANN : hybrid calibration

$$X = FK(\xi_{identified}, \Theta) + ANN(\Theta) \tag{7}$$

Tab 3 summarizes the results and Figure 6 shows the error distribution for each calibration method. For readability purposes, only the positioning error distribution for payload P_1 are depicted, as the distributions are very similar for payload P_2. Figure 6 shows that calibration of industrial robot is necessary for off-line programming, as the positioning error without any compensation is high and widely distributed. Geometric calibration allows a significant reduction of the mean error and tightens the error distribution. Black-box modeling reduces further the mean error. However, the best results are consistently obtained with hybrid calibration the mean and the max error are the lowest, and the error distribution is the tightest around the mean. This can be seen on Tab.3. Hybrid calibration methods can reduce the mean positioning error by at least 97%, leading to a mean error close to the robot's repeatability (0.06mm according to the manufacturer) after calibration. Moreover, comparing Tab.3 and Tab.4 (Tab.4 summarizes the results obtained in a previous works [35] for a KUKA iiwa robot), the accuracy of the hybrid model is very similar for both robots, despite their differences: the KR300 is bigger (reach of 2500mm, compared to 820mm for the iiwa), stiffer [7, 9], and has a better repeatability (0.06mm versus 0.15mm for the *iiwa*). A study on the required number of data for both model (6) and (7) have been made. Figure 7 shows that hybrid calibration converges faster than model-less calibration, in every measurement configuration.

Since the experimental setup and the hybrid model are very similar for both robots (see Figure 8 and [35]), one can assume that applying the described hybrid calibration method to any other robot leads to similar results: hybrid



(a) Sub-workspace S_1, payload P_1 (b) Sub-workspace S_2, payload P_1

Fig. 6: Positioning error (mm) distribution (over 100 validations points, with a bar width of 0.04mm) for models (4) to (7) for payload P_1. Vertical black lines represent the mean error.

Table 4: Accuracy of models (4) to (7) for the KUKA iiwa [35]

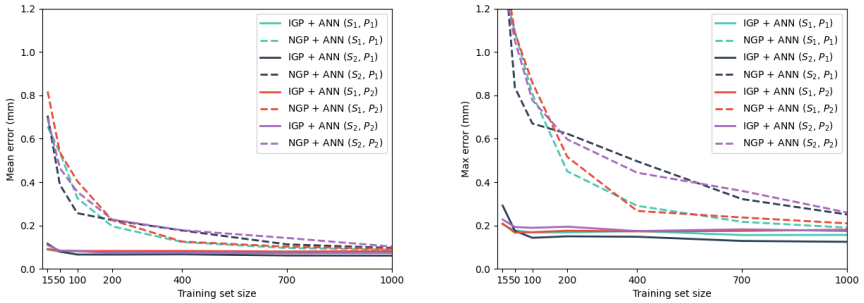
Method	Sub-workspace 1		Sub-workspace 2	
	Mean error	Max error	Mean error	Max error
NGP	3.25	4.48	1.95	2.75
IGP	0.96	1.40	1.63	2.05
NGP + ANN	0.094	0.22	0.079	0.185
IGP + ANN	0.091	0.24	0.083	0.175

calibration reduces significantly the positioning error, while being more data-efficient than *model-less* calibration, in any sub-workspace with any payload.

3 Payload variation compensation

3.1 Considerations and principle of the proposed methods

In many industrial applications robots must handle different payloads. These payloads induce torque variations on each joint, depending on the joint configuration, thus causing torsion on each axis, leading to positioning error. Assuming that the payload has a negligible effect on other sources of errors, the positioning error ΔX is decomposed as follow:



(a) Mean error (b) Max error

Fig. 7: Influence of the number of data on the mean (left) and max (right) positioning error.

$$\Delta X = f(\Delta X_{geometric} + \Delta X_{self\ mass} + \Delta X_{gearboxes} + \Delta X_{thermal\ effects} + \Delta X_{residual}) + \Delta X_{payload} \quad (8)$$

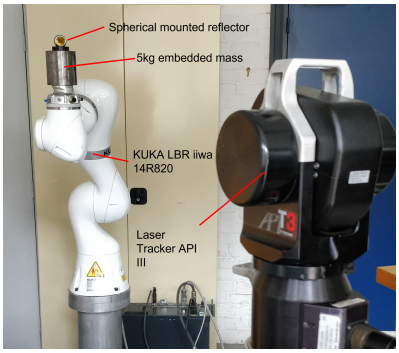
The effect of the payload is commonly modeled:

$$\Delta X_{payload} = J \times K_{\Theta}^{-1} \times J^{-1} \times P \quad (9)$$

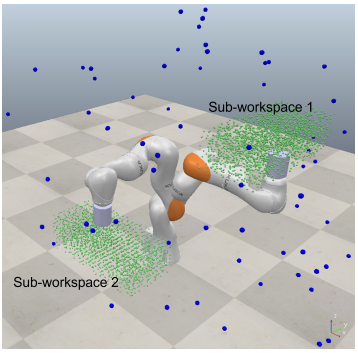
with J being the kinematic Jacobian, K_Θ the stiffness matrix, and P the payload vector expressed in the robot's base coordinates and applied to the flange. This model requires the identification of the stiffness of all joints, contained in K_Θ.

However, joint stiffness are difficult to identifiy in practice, as it requires torque sensors on each joint or the mass model of the robot. Moreover, as joint stiffness is mostly due to gear drives stiffness, it is known to be non-linearly dependent on the applied torque [11, 12]. These elements make any analytical compensation of $\Delta X_{payload}$ difficult.

The classical hybrid calibration as described in section 2, referred as *classical approach* hereafter, requires a full dataset to train the ANN to get a set of weights adapted to a new payload. For data and time efficiency, as discussed in 2.3, two novel methods for payload compensation are investigated. The first one relies on the use of pre-trained weights with an other payload for ANN's weights initialization in a real-to-real transfer learning fashion, to minimize the number of data needed, thus reducing the experimental complexity. The second method directly interpolates two outputs of an ANN by using two distinct sets of weights obtained from previous training with different payloads, under the assumption that the gravity center of the end-effector does not vary significantly. By this way, a new payload could be handled without any additional measurement, making the implementation in real industrial scenarios fast and easy.



(a) Experimental setup with laser tracker API III, KUKA *iiwa* 14R820, 5kg payload and a SMR



(b) Measurement distribution for geometric parameters identification (blue) and ANN's training (green)

Fig. 8: Experimental setup and measurement repartition for the *iiwa* used in [35]

3.2 Method 1: using pre-trained weights for initialization

In this method, when dealing with a new payload P_{new}, a fast training process is done. ANN's weights are initialized with the ones coming from training with a previous payload. By this way, the ANN could re-use previously-acquired knowledge, requiring only few measurements with the new payload to efficiently train the ANN.

The ANN's weights obtained with the classical approach in a sub-workspace S with a payload P is denoted $w_{P,\,S}$. As $w_{P_1,\,S_1}$ are already fitting the joint configuration to the cartesian error accurately for a payload P_1, initializing the ANN for a new payload P_{New} with $w_{P_1,\,S_1}$ allows the model to start the learning phase with a lower loss value. Indeed, $w_{P_1,\,S_1}$ is able to compensate for the first term of ΔX in Eq.(8), and they only need to be updated with few new data to compensate for $\Delta X_{payload}$. This method is depicted in Figure 9.

In practice, after the training leading to $w_{P_1,\,S_1}$ and $w_{P_1,\,S_2}$, two new set of weights, denoted $w_{P_{1\rightarrow new},\,S_1}$ and $w_{P_{1\rightarrow new},\,S_2}$, are initialized with $w_{P_1,\,S_1}$ and $w_{P_1,\,S_2}$, respectively. Using $n_{data} << 1000$ measurements from the datasets $(P_{new},\ S_1)$ and $(P_{new},\ S_2)$, the training is done to get new adapted weights.

3.3 Method 2: by interpolation of two ANN's predictions

This method relies on the interpolation of two predictions coming from two different set of weights of the ANN, trained in the same sub-workspace with two different payloads. As the ANN is able to compensate for all non-geometric errors (according to section 2.4), and according to Eq.(8), if the payload varies, only $\Delta X_{payload}$ varies, and all other terms remain constant for a given joint position. Thus, interpolating the predictions of the ANN is mathematically equivalent to interpolate $\Delta X_{payload}$, as long as the payload vector P in Eq.(9)

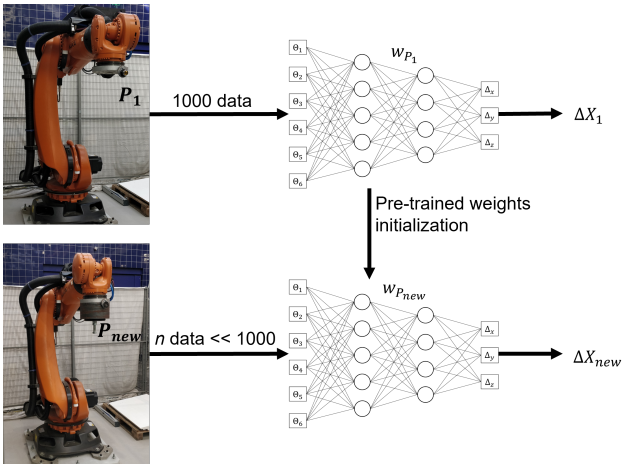


Fig. 9: Illustration of method 1 principle

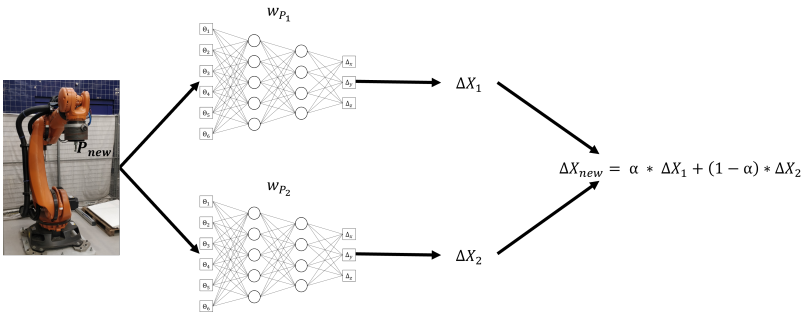


Fig. 10: Illustration of method 2 principle

evolves linearly, i.e. the gravity center of the end-effector does not vary. For the new known payload P_{new}, the model becomes:

$$X = FK(\xi_{identified}, \Theta) + \alpha \times ANN_{P_1}(\Theta) + (1-\alpha) \times ANN_{P_2}(\Theta) \quad (10)$$

with $\alpha = (P_2 - P_{new})/(P_2 - P_1)$. This method is depicted in Figure 10.

The main advantage of this model is that it does not require any additional measurements. Moreover, it can be coupled with method 1: first, ANN can be trained, according to the method described in Section 2, with data measured while the robot handle a light payload P_{light}, resulting in a set of weights $w_{P_{light},S}$. Then, the ANN can be initialized with $w_{P_{light},S}$, and trained with measurement data while the robot handle a heavy payload P_{heavy}, so that the total amount of data required for two training phases is significantly reduced.

4 Validations of the proposed models

4.1 Evaluation of the proposed methods performances

Before any experimental validation using the compensation algorithm, the performance and relevance of methods 1 and 2 are assessed. For comparison purposes, two new datasets of 1100 measurements have been acquired in each sub-workspaces $S = S_1$ and $S = S_2$ with a new payload $P_{New} = 72kg$. For each sub-workspace, a comparison have been made between:

- the classical approach, i.e. $w_{P_{new},S}$ obtained with training on the whole dataset.
- a non-updated set of weights $w_{P_1,S}$, trained following the classical approach, to highlight the need for adaptation to new payloads.
- the classical approach if the training is made with the same amount of data that method 1, to assess its relevance. In practice, comparison is done with $n_{data} = 15$.
- Method 1, as described in section 3.2, applied with 15 training points.
- Method 2, as described in section 3.3, with $\alpha = \frac{120-72}{120-4} \simeq 0.414$.

Results are shown in Tab.5 and Figure 11. First, the results of the classical approach, trained on the whole dataset, are similar to the ones obtained with $P_1 = 4kg$ and $P_2 = 120kg$, in both sub-workspaces. This is consistent with the conclusion of Section 2. On Figure 11, the error distribution for method 1 is very similar to the one for classical hybrid calibration described in Section 2, showing the interest of this method for data-efficiency. Moreover, method 1 performs significantly better that the classical approach but using the same number of data than method 1 (15 training data). This means that, with an equal number of training data, re-using previously-trained set of weights is better than initializing new weights with the common Glorot initializer [39]. In both sub-workspaces, methods 1 and 2 perform better than using a model trained with a different payload (e.g. ANN_{P_1}), highlighting the fact that it is necessary to adapt the model to new payloads. However, methods 2 performs better in S_1 than in S_2. This can be explained by the fact that the configuration of the robot and of the tools in S_1 do not entails additional torque due to a change in the tool's gravity center position. Indeed, for the three used payloads P_1, P_2 and P_{new}, the position on the gravity center is along $\overrightarrow{z_{flange}}$. As for S_1, the configuration of the robot makes $\overrightarrow{z_{flange}}$ and $\overrightarrow{z_{world}}$ co-linear, there is no lever arm that would induce additional torque, contrary to the robot's configuration in S_2, where $\overrightarrow{z_{flange}}$ is orthogonal to $\overrightarrow{z_{world}}$. Thus, the relation between the torque applied on each joint evolves linearly with the payload in S_1, while in S_2 it depends on both the payload and the gravity center's position. In scenarios where it is possible to measure new data with a new payload, method 1 enables a fast measurement process, with performances similar to the classical hybrid approach with a much lower number of data. In situations where no measurement device is available, method 2 performs

Table 5: Mean and max error of different models for a new payload $P_{new} = 72\,kg$ in both sub-workspaces $S = S_1$ and $S = S_2$

Method	Sub-workspace 1		Sub-workspace 2	
	Mean error	Max error	Mean error	Max error
$ANN(w_{P_{New},\,S})$ (classical approach)	0.077	0.173	0.059	0.132
$ANN(w_{P_1,\,S})$	0.256	0.375	0.410	0.587
$ANN(w_{P_{New},\,S})$ (15 training points)	0.110	0.253	0.109	0.253
Method 1	0.081	0.152	0.077	0.144
Method 2	0.147	0.228	0.256	0.359

better than using a non-updated set of weights, but may be performing less well than classical approach due to a displacement of the tool's gravity center.

4.2 Experimental validation using compensation algorithm

4.2.1 Compensation algorithm

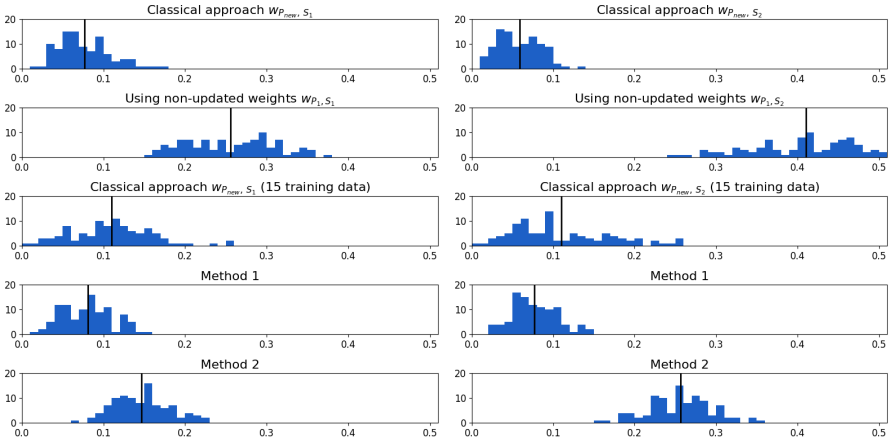
The proposed approaches are tested using the compensation algorithm mentioned before. Indeed, in most of industrial scenarios, robots are programmed with cartesian coordinates, and the robot's controller converts them to joint position. As the models take as input joint positions to return cartesian coordinates, a compensation algorithm is used instead of the nominal inverse kinematics model. The compensation algorithm supplants the nominal IK model, as depicted in Figure 1. The algorithm is described in Figure 12.

This algorithm takes as input the desired cartesian position. Through the nominal inverse kinematics (IK) model, a joint position is deduced. Using the hybrid model, a theoretically reached cartesian position X_i is computed. The difference between X_{ref} and X_i (denoted Δ_i) is then deduced, and added to X_{ref} to give an intermediate cartesian position X_i'. From this cartesian position, a new joint position is given by the **IK**, and the previous steps are repeated until Δ_i no longer evolves. Finally, the last computed joint position reaches the desired cartesian position, according to the model used. This algorithm can be used with any model that can map robot's joint position to cartesian position of its end-effector.

4.2.2 Results

Five different evaluations have been carried out in S_1, to assess the effectiveness of the proposed models in real scenarios.

- Classical hybrid approach with payload $P_1 = 4kg$
- Classical hybrid approach with payload $P_2 = 120kg$
- Method 1 ($w_{P_{1\rightarrow New},\,S_1}$) for $P_{new} = 72kg$



(a) S_1 (b) S_2

Fig. 11: Positioning error (mm) distribution (over 100 validations points extracted from the training sets, with a bar width of 0.01mm) of different methods for a new payload of 72kg. Vertical black lines represent the mean error.

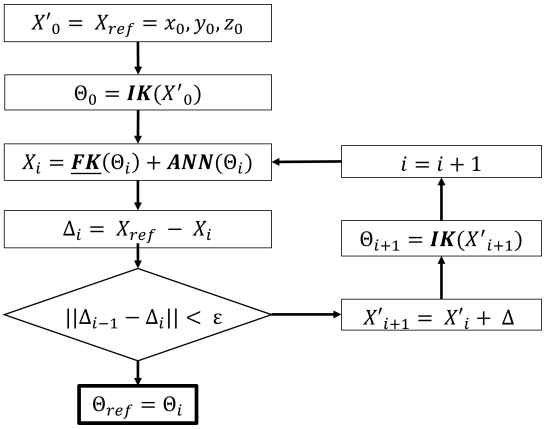


Fig. 12: Compensation algorithm based on ANN error prediction.

- Method 2 (using w_{P_1, S_1} and w_{P_2, S_1}) for a new payload $P_{new} = 72\,kg$
- Method 2 (using w_{P_1, S_1} and w_{P_2, S_1}) for a new payload $P_{new} = 128\,kg$

Results of the experimental validations are depicted in Figure 13 and summarized in Tab.6. All the proposed approaches have a similar error distribution, and are all consistent with the previous results. In practice, Δ_n is of the order of magnitude of $10^{-4}mm$, so the output joint position reaches the desired cartesian position perfectly according to the model used. The slight difference

between these results and the one presented in Section 4.1 comes from the robot's repeatability and the laser tracker's accuracy. Results show that hybrid calibration reduces by 97% the positioning error. The proposed methods are more accurate (final accuracy 0.05 and 0.06mm away from the robot's repeatability for methods 1 and 2 respectively, while state-of-the-art methods are 0.1-0.3mm away of the considered robot's repeatability [8, 9, 23]), and require no specific information or sensors (CAD-model, torque sensors) compared to state-of-the-art methods. In particular, the proposed methods are able to use previously acquired knowledge to compensate for new payloads, reducing the time and effort needed to perform all the measurements. Indeed, four hours of measurements were necessary for classical approach but only half an hour for Method 1 (after having applied at least one time the classical approach).

4.3 Discussion

The results presented in Section 4 show that the two proposed methods are effective for payload effects compensation. Method 1 presents consistent results in both sub-workspaces in simulation (as shown in Tab.5 and in Figure 11) and in experimental validation using the compensation algorithm, with only 15 new data. Method 2 does not require any new measurements, and provides interesting results in S_1 and in experimental validation using the compensation algorithm. However, method 2 relies on the assumption that the gravity center of the object handled do not vary significantly, or remains on the $\overrightarrow{z_{world}}$ axis. This is not a strong limitations since for many applications, such as machining or pick-and-place, the tool's center of gravity is usually located along $\overrightarrow{z_{world}}$. This limit have been experienced in S_2, as shown in Tab.5 and in Figure 11. Moreover, this method could be ineffective for extrapolation to very high payload, since the relationship between applied torque and axis deformation is linear up to a certain torque, after which the relationship is still smooth and could be linearly approximated [11, 12]. Thus, after the linear part of the relationship between applied torque and axis deformation, the curve could still be approximated by a piece-wise linear model, hence interpolation of the output of two ANN trained for different payloads can still be effective for high payloads.

Hence, the two methods are fast, data-efficient and allows the reconfigurability of existing hybrid calibration models to adapt to new payloads while keeping a high level of accuracy. Even though method 2 suffers from limitations and hypothesis, it can still be effective for a wide range of application (e.g. pick-and-place, drilling, machining), without any additional data.

5 Conclusion

In this paper, hybrid calibration methods have been investigated on a traditional industrial robot, a KUKA KR300, and two methods for payload variation compensation are proposed. First, it has been shown that hybrid calibration provides consistently better results than model-based or model-less

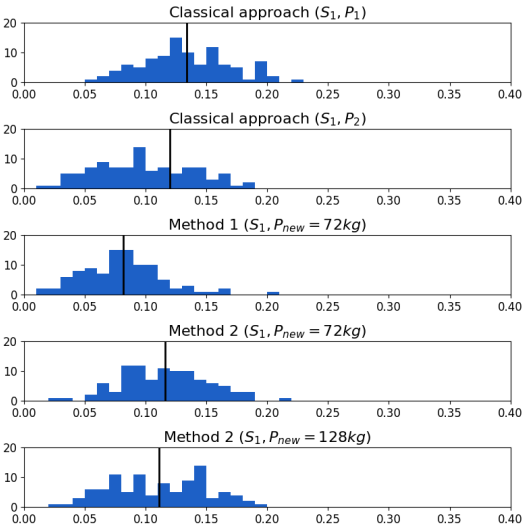


Fig. 13: Positioning error (mm) distribution (over 100 experimental validation points computed with the compensation algorithm, with a bar width of 0.01mm) of different methods. Vertical black lines represent the mean error.

Payload	Method	Mean error	Max error
$P_1 = 4kg$	Classical approach	0.13	0.23
$P_2 = 120kg$	Classical approach	0.10	0.19
$P_{new} = 72kg$	Method 1	0.08	0.20
$P_{new} = 72kg$	Method 2	0.12	0.22
$P_{new} = 128kg$	Method 2	0.11	0.19

Table 6: Experimental validation results

calibration. Moreover, the number of data required for ANN's training is significantly reduced with hybrid calibration. Since this study have been carried out previously on a KUKA iiwa 14R820 (which is smaller and more compliant) with similar conclusions, one can assume that they will be valid for any industrial robot. For both robots, the mean positioning error is reduced by 95% in two different sub-workspaces.

Moreover, two hybrid calibration approaches are proposed to reduce the positioning error of industrial robots in scenarios involving payload variations. In such cases, the hybrid models previously trained are no longer valid. Instead of starting over a new hybrid calibration process (which is time-consuming), the two proposed methods re-use former ANNs to speed up the experimental processes. The first one relies on an efficient initialization of a new set of weights, using a previously-trained one for another payload. The new weights can be trained with only a few measurement made with a new payload. The second method is based on the interpolation of two predictions coming from

two different set of weights of the ANN, trained in the same sub-workspace with two different payloads, thus requiring no additional data for new payloads. Both methods are fast, data-efficient and accurate. Experimental validation shows that method 1 is able to reduce the mean positioning error from 4.23mm to 0.08mm with only 15 data, and Method 2 to 0.12mm without any data.

6 Declaration

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Code or data availability (software application or custom code)

The data used in this paper can be uploaded at: https://doi.org/10.57745/DWUC0H.

Authors' Contributions

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Maxime Selingue. The first draft of the manuscript was written by Maxime Selingue and all authors commented on previous versions of the manuscript.

Ethics approval

Not applicable

Consent to participate

Not applicable

Consent for publication

Not applicable

References

[1] Elatta, A.Y., Gen, L.P., Zhi, F.L., Daoyuan, Y., Fei, L.: An overview of robot calibration. Information Technology Journal **3**, 74–78 (2004)

[2] Xuan, J.-Q., Xu, S.-H., *et al.*: Review on kinematics calibration technology of serial robots. International journal of precision engineering and manufacturing **15**(8), 1759–1774 (2014)

[3] Mooring, B., Roth, Z., Driels, M.: Fundamentals of manipulator calibration, 27 (1991)

[4] Li, Z., Li, S., Luo, X.: An overview of calibration technology of industrial robots. IEEE/CAA Journal of Automatica Sinica **8**(1), 23–36 (2021)

[5] J. Kenneth, S.: Active stiffness control of a manipulator in cartesian coordinates. 1980 19th IEEE Conference on Decision and Control including

the Symposium on Adaptive Processes, 95–100 (1980)

[6] Khalil, W., Besnard, S.: Geometric calibration of robots with flexible joints and links. Journal of Intelligent and Robotic Systems **34**, 357–379 (2002)

[7] Dumas, C., Caro, S., Garnier, S., Furet, B.: Joint Stiffness Identification of Six-revolute Industrial Serial Robots. Robotics and Computer-Integrated Manufacturing **27**(4), 881–888 (2011)

[8] Olabi, A., Damak, M., Bearee, R., Gibaru, O., Leleu, S.: Improving the accuracy of industrial robots by offline compensation of joints errors. In: 2012 IEEE International Conference on Industrial Technology, pp. 492–497 (2012)

[9] Besset, P., Olabi, A., Gibaru, O.: Advanced calibration applied to a collaborative robot. In: 2016 IEEE International Power Electronics and Motion Control Conference (PEMC), pp. 662–667 (2016)

[10] Theissen, N.A., Laspas, T., Archenti, A.: Closed-force-loop elastostatic calibration of serial articulated robots. Robotics and Computer-Integrated Manufacturing **57**, 86–91 (2019). https://doi.org/10.1016/j.rcim.2018.07.007

[11] Tuttle, T.D., Seering, W.P.: A nonlinear model of a harmonic drive gear transmission. IEEE Transactions on Robotics and Automation **12**(3), 368–374 (1996). https://doi.org/10.1109/70.499819

[12] Kircanski, N., Goldenberg, A.A., Jia, S.: An experimental study of nonlinear stiffness, hysteresis, and friction effects in robot joints with harmonic drives and torque sensors, vol. 16, pp. 326–340 (1993)

[13] Klimchik, A., Pashkevich, A., Chablat, D.: Fundamentals of manipulator stiffness modeling using matrix structural analysis. Mechanism and Machine Theory **133**, 365–394 (2019). https://doi.org/10.1016/j.mechmachtheory.2018.11.023

[14] Rezaei, A., Akbarzadeh, A.: Compliance error modeling for manipulators considering the effects of the component weights and the body and joint flexibilities. Mechanism and Machine Theory **130**, 244–275 (2018). https://doi.org/10.1016/j.mechmachtheory.2018.08.012

[15] Klimchik, A., Pashkevich, A., Chablat, D.: Cad-based approach for identification of elasto-static parameters of robotic manipulators. Finite Elements in Analysis and Design **75**, 19–30 (2013). https://doi.org/10.1016/j.finel.2013.06.008

[16] Wang, Y.Y., Huang, T., Zhao, X.M., Mei, J.P., Chetwynd, D.G., Hu, S.J.: Finite Element Analysis and Comparison of Two Hybrid Robots-the Tricept and the TriVariant, pp. 490–495 (2006). https://doi.org/10.1109/IROS.2006.282522. https://www.scopus.com/inward/record.uri?eid=2-s2.0-34250625329&doi=10.1109%2fIROS.2006.282522&partnerID=40&md5=db128af8200a51e5aa64ac8c1f4f753b

[17] Cao, W.-a., Yang, D., Ding, H.: A method for stiffness analysis of over-constrained parallel robotic mechanisms with scara motion. Robotics and Computer-Integrated Manufacturing **49**, 426–435 (2018). https://doi.org/10.1016/j.rcim.2017.08.014

[18] Kumar, P., *et al.*: Artificial neural network based geometric error correction model for enhancing positioning accuracy of a robotic sewing manipulator. Procedia Computer Science **133**, 1048–1055 (2018)

[19] Takanashi, N.: 6 dof manipulators absolute positioning accuracy improvement using a neural-network. In: EEE International Workshop on Intelligent Robots and Systems, Towards a New Frontier of Applications, pp. 635–6402 (1990)

[20] Josin, G., Charney, D., White, D.: Robot Control Using Neural Networks. In: IEEE 1988 International Conference on Neural Networks, pp. 625–631 (1988). https://doi.org/10.1109/icnn.1988.23980

[21] Gao, G., Zhang, H., San, H., Wu, X., Wang, W.: Modeling and error compensation of robotic articulated arm coordinate measuring machines using bp neural network. Complexity **2017** (2017)

[22] Aoyagi, S., Kohama, A., Nakata, Y., Hayano, Y., Suzuki, M.: 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-, 5660–5665 (2010)

[23] Nguyen, H.-N., Zhou, J., Kang, H.-J.: A calibration method for enhancing robot accuracy through integration of an extended kalman filter algorithm and an artificial neural network. Neurocomputing **151**, 996–1005 (2015)

[24] Nguyen, H.-N., Le, P.N., Kang, H.-J.: A new calibration method for enhancing robot position accuracy by combining a robot model–based identification approach and an artificial neural network–based error compensation technique. Advances in Mechanical Engineering **11**, 168781401882293 (2019)

[25] Zhao, G., Zhang, P., Ma, G., Xiao, W.: System identification of the nonlinear residual errors of an industrial robot using massive measurements.

Robotics and Computer-Integrated Manufacturing **59**, 104–114 (2019)

[26] Nguyen, H.X., Cao, H.Q., Nguyen, T.T., Tran, T.N.-C., Tran, H.N., Jeon, J.W.: Improving robot precision positioning using a neural network based on levenberg marquardt–apso algorithm. IEEE Access **9**, 75415–75425 (2021). https://doi.org/10.1109/ACCESS.2021.3082534

[27] Gadringer, S., Gattringer, H., Müller, A., Naderer, R.: Robot calibration combining kinematic model and neural network for enhanced positioning and orientation accuracy. IFAC-PapersOnLine **53**(2), 8432–8437 (2020). https://doi.org/10.1016/j.ifacol.2020.12.1436

[28] Hsiao, J.-C., Shivam, K., Lu, I.-F., Kam, T.-Y.: Positioning accuracy improvement of industrial robots considering configuration and payload effects via a hybrid calibration approach. IEEE Access **8**, 228992–229005 (2020)

[29] Sun, Y., Hollerbach, J.M.: Observability index selection for robot calibration. In: 2008 IEEE International Conference on Robotics and Automation, pp. 831–836 (2008). IEEE

[30] Joubair, A., Bonev, I.A.: Comparison of the efficiency of five observability indices for robot calibration. Mechanism and Machine Theory **70**, 254–265 (2013)

[31] Klimchik, A., Wu, Y., Pashkevich, A., Caro, S., Furet, B.: Optimal Selection of Measurement Configurations for Stiffness Model Calibration of Anthropomorphic Manipulators. Applied Mechanics and Materials **162**, 161–170 (2012)

[32] Klimchik, A., Caro, S., Pashkevich, A.: Optimal pose selection for calibration of planar anthropomorphic manipulators. Precision Engineering **40**, 214–229 (2015)

[33] Ye, C., Yang, J., Ding, H.: High-accuracy prediction and compensation of industrial robot stiffness deformation. International Journal of Mechanical Sciences **233** (2022). https://doi.org/10.1016/j.ijmecsci.2022.107638

[34] Yu, W., Kumar, V.C.V., Turk, G., Liu, C.K.: Sim-to-Real Transfer for Biped Locomotion, pp. 3503–3510 (2019). https://doi.org/10.1109/IROS40897.2019.8968053

[35] Selingue, M., Olabi, A., Thiery, S., Béarée, R.: Experimental analysis of robot hybrid calibration based on geometrical identification and artificial neural network. In: IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society, pp. 1–6 (2022). https://doi.org/10.1109/IECON49645.2022.9968704

[36] Denavit, J., Hartenberg, R.S.: Notation for lower-pair mechanisms based on matrices. A. Kinematic, ASME Journal of Applied Mechanics **22**, 215–221 (1995)

[37] Siciliano, B., Khatib, O.: Springer Handbook of Robotics. Springer, Berlin, Heidelberg (2007)

[38] Khalil, W., Dombre, E.: Chapter 11 - geometric calibration of robots. In: Khalil, W., Dombre, E. (eds.) Modeling, Identification and Control of Robots, pp. 257–289. Butterworth-Heinemann, Oxford (2002)

[39] Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Teh, Y.W., Titterington, M. (eds.) Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 9, pp. 249–256. PMLR, Chia Laguna Resort, Sardinia, Italy (2010). https://proceedings.mlr.press/v9/glorot10a.html

[40] Meng, Y., Zhuang, H.: Autonomous robot calibration using vision technology. Robotics and Computer-Integrated Manufacturing **23**(4), 436–446 (2007)

[41] Mazzoni, F., Olabi, A., Bearee, R., Ernst-Desmulier, J.-B.: Calibration methodology for multirobot assembly cell. In: IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society, pp. 1–5 (2022). https://doi.org/10.1109/IECON49645.2022.9968339

[42] Dombre, E., Khalil, W.: Robot Manipulators: Modeling, Performance Analysis and Control. Control Systems, Robotics and Manufacturing Series, (2007)

[43] Gong, C., Yuan, J., Ni, J.: Nongeometric error identification and compensation for robotic system by inverse calibration. International Journal of Machine Tools and Manufacture **40**(14), 2119–2137 (2000). https://doi.org/10.1016/S0890-6955(00)00023-7