



A novel step-by-step procedure for the kinematic calibration of robots using a single draw-wire encoder

Giovanni Boschetti^{1,2} · Teresa Sinico³

Received: 25 October 2023 / Accepted: 6 February 2024 / Published online: 23 February 2024
© The Author(s) 2024

Abstract

Robot positioning accuracy is a key factory when performing high-precision manufacturing tasks. To effectively improve the accuracy of a manipulator, often up to a value close to its repeatability, calibration plays a crucial role. In the literature, various approaches to robot calibration have been proposed, and they range considerably in the type of measurement system and identification algorithm used. Our aim was to develop a novel step-by-step kinematic calibration procedure — where the parameters are subsequently estimated one at a time — that only uses 1D distance measurement data obtained through a draw-wire encoder. To pursue this objective, we derived an analytical approach to find, for each unknown parameter, a set of calibration points where the discrepancy between the measured and predicted distances only depends on that unknown parameter. This reduces the computational burden of the identification process while potentially improving its accuracy. Simulations and experimental tests were carried out on a 6 degrees-of-freedom robot arm: the results confirmed the validity of the proposed strategy. As a result, the proposed step-by-step calibration approach represents a practical, cost-effective and computationally less demanding alternative to standard calibration approaches, making robot calibration more accessible and easier to perform.

Keywords Robot calibration · Kinematic calibration · Positioning accuracy · Robot kinematics · Draw-wire encoder

1 Introduction

In the last two decades, the applications of robots in high-precision manufacturing tasks such as grinding [1], milling [2], or riveting [3] have expanded extensively, and so has the demand for higher precision manipulators.

The precision of a robot can be quantified in terms of repeatability or accuracy: repeatability is defined as the

precision with which the robot's end-effector returns to a previously taught position, while accuracy is defined as the precision with which the robot's end-effector moves to a commanded position with respect to a spatial coordinate frame. Manipulators currently used in industry are characterized by very high repeatability, but poor accuracy. The latter is rarely given by robot manufacturers and can assume a value of some millimeters.

High repeatability is the main requirement in a variety of manufacturing and handling applications where the required robot end-effector poses are manually taught by joggling the robot through a teaching pendant. Along with high repeatability, high accuracy is required in manufacturing tasks that involve offline programming, such as drilling and laser cutting, where the robot's end-effector poses are defined with respect to an absolute or relative reference frame. Indeed, robot positioning accuracy may not be adequate for performing these types of task, and conventional machine tools may still be preferred, despite their higher cost and lower flexibility. For this reason, increasing robots' accuracy is crucial for their application in flexible and reconfigurable manufacturing systems.

Teresa Sinico contributed equally to this work.

✉ Giovanni Boschetti
giovanni.boschetti@unipd.it
Teresa Sinico
teresa.sinico@phd.unipd.it

¹ Department of Industrial Engineering (DII), University of Padua, Via Venezia 1, Padua 35131, Italy

² Department of Information Engineering (DEI), University of Padua, Via Giovanni Gardenigo 6b, Padua 35131, Italy

³ Department of Management and Engineering (DTG), University of Padua, Stradella S. Nicola 3, Vicenza 36100, Italy

The accuracy of a robot depends on the accuracy of the robot's mathematical model in the controller: this model computes the joint angles of the robot given an end-effector pose with respect to a reference frame. The inaccurate knowledge of the geometric and non-geometric parameters of this mathematical model is the major source of the discrepancy between the actual pose of the robot's end-effector and the pose predicted by its controller. Errors associated with geometric parameters are related to the deviation between the nominal and actual dimensions of the robot's mechanical links, misalignment of the joint axes, and incorrect joint variable offset values used to describe the manipulator's home position [4]. Errors associated with non-geometric parameters are caused by deformation in the mechanical components of the robot from external load and self-gravity, mechanical wear, thermal variation, sensors and servos precision, friction, and other non-linearities, including hysteresis and gear backlash [5, 6].

A practical approach to improve the robot accuracy — up to a value close to its repeatability — is to re-evaluate the parameters in the robot's mathematical model by using a calibration scheme. This procedure mainly involves four steps: modeling, measuring, identifying parameters, and implementing error compensation.

1.1 Modeling

The first step of robot calibration is to derive a mathematical model that relates the robot's joint angles to its end-effector pose and takes into account the geometric and non-geometric parameters that need to be identified. A kinematic model suitable for robot calibration should meet the three principles of completeness, continuity, and minimality [7]. The standard Denavit-Hartenberg (DH) convention [8] is widely used for kinematic modeling in robotics; however, this model does not meet the continuity condition when two consecutive joint axes are parallel. Since most industrial robots possess this feature, significant efforts have been made to solve this problem: authors either proposed to use a simplified version of the DH model to make it continuous [9, 10] or a modified version of the standard DH model (MDH) [11], adding an additional parameter to the original convention. In addition, other kinematic models that satisfy the continuity condition have been proposed, such as the S-model [12], the complete and parametrically continuous (CPC) model [13, 14] and the product of exponential (POE) based model [15, 16]. These models meet the conditions of completeness and continuity but usually do not meet the minimality condition, i.e., some parameters are redundant; those redundant parameters must be determined and excluded from the model before identification.

1.2 Measuring

Once a complete, non-singular and minimal kinematic calibration model has been derived, it is then used to compute the predicted end-effector pose based on the nominal kinematic parameter set. The predicted end-effector pose is compared with the actual end-effector pose measured by an external measurement system and an error quantity is defined. To measure the actual end-effector pose, a variety of different measurement systems can be used, which differ considerably in their cost, accuracy, ease of use, and type of data collected. In particular, the measurement systems used to calibrate a robot can be classified into complete pose measurement and partial pose measurement: a complete pose measurement of the robot's end-effector pose consists of three position coordinates and three orientation angles, while a partial measurement of the robot's end-effector pose consists of less than six measured values per observation (typically ranging from 3D to 1D). The most common measurement systems used for robot calibration are laser tracking systems [9, 17–19], vision systems [20, 21], ballbars [10, 22], theodolites [23–25], and coordinate measuring machines (CMM) [26, 27].

1.3 Identifying error parameters

When a sufficient number of end-effector poses have been measured, the unknown parameters can be estimated. This identification can be achieved by determining the analytical relationship between the end-effector coordinates and the parameters in the form of a Jacobian and then inverting the equation to calculate the deviation of the parameters from their nominal values. Alternatively, the identification can also be viewed as a constrained non-linear optimization problem. In this case, a cost function that relates the parameter to a quantity that is an overall measure of optimality is defined, and the parameter set is systematically changed to reduce the cost function to zero. This problem can be solved using different optimization algorithms, such as Levenberg-Marquardt [28], extended Kalman filter (EKF) [29, 30], or particle swarm optimization (PSO) [31, 32].

1.4 Implementing error compensation

The final step of robot calibration involves implementing error compensation: this can be done by modifying the nominal parameters embedded in the robot controller or by using error compensation techniques. Since most robot controllers use DH parameters, the former approach is typically possible only if the standard DH convention is used in the modeling phase and if the identification is limited to kinematic parameters. However, access to modify the kinematic parameters is not always possible and may be limited (i.e., not all

parameters can be modified). On the other hand, error compensation techniques need to be implemented if a different kinematic model is used in the modeling step, if access to modify some or all the kinematic parameters is not possible, and also if the identification is extended to non-kinematic parameters. In this case, the pose deviation is first calculated using the mathematical model of the manipulator with the estimated parameters and then a compensated pose is obtained [33, 34].

1.5 Scope and contribution

In this paper, we present a novel step-by-step procedure for the kinematic calibration of robots using a single draw-wire encoder. We were interested in developing a kinematic calibration procedure with this measurement system because it offers a good balance of accuracy, resolution, measurement range, cost, and usability. In particular, it is much less expensive and requires an easier setup than laser trackers, which are still the most used measurement instrument for the calibration of robots. Several studies have used multiple draw-wire encoders for robot calibration: they can be arranged as complete pose measurement devices [35–37] or partial pose measurement devices [38–40]. However, few studies have considered the possibility of using a single draw-wire encoder [40–45], which only provides a 1D radial measurement of a point with respect to a fixed reference frame. In this case, the identification step is typically viewed as an optimization problem, where the cost function is the sum of the difference between the measured and predicted distance of the robot's end-effector from a fixed point over a number of calibration poses.

When calibrating a robot using a single draw-wire encoder, two practical choices have to be made: the encoder location and the set of calibration points where measurements are taken. These two aspects greatly influence the resulting accuracy after calibration, but previous studies did not explore them in depth. In [40] the sensor is located outside the manipulator workspace, while in [43] it is located in an arbitrary place: in both cases, the location of the measurement system is an unknown parameter that is found through the optimization algorithm along with the other kinematic parameters. Unfortunately, in [41, 42, 44, 45] the authors did not provide complete details of sensor location. Moreover, in [40] the calibration points have been chosen from a uniformly sampled grid that covers most of the working envelope in front of the robot, while in [41–45] the details about the calibration points are not reported.

Our step-by-step calibration procedure differs significantly from previously proposed approaches using a single draw-wire encoder, and the heart of our approach lies in the careful selection of encoder location and calibration points. The draw-wire encoder is placed in a position in the

robot's workspace specified by a set of joint coordinates: this allows the estimation of kinematic parameters without explicit knowledge of the coordinates of the encoder location. This approach is more robust than the approaches proposed in [40, 43], as the identification of the fixed location of the encoder could introduce errors that could propagate to the final results. In addition, we developed a novel analytical approach to find a set of calibration points for each parameter that needs to be identified: in this set of points, the discrepancy between predicted and measured distances depends only on that parameter (and possibly on those that have already been identified). Consequently, our calibration procedure can be viewed as a step-by-step procedure through which the unknown parameters are subsequently identified: the manipulator is first moved to the first set of calibration points, where the measured distance error depends only on one of the kinematic parameters. This unknown parameter is identified, and the manipulator is moved to a second set of calibration points, where the error depends on a second kinematic parameter (and possibly also on the just identified parameter). The second kinematic parameter is then estimated, and, following the same procedure, all kinematic parameters are identified. The proposed approach offers two major advantages: only one of the unknown parameters is estimated at a time, and fewer measurements are processed at the same time. The consequence of the former is that the parameters can be estimated more accurately than with previously proposed approaches, and their identification does not require complex optimization algorithms; the consequence of the latter is that the measurement data can be computationally processed faster. As a result, the proposed calibration procedure can be implemented directly into the robot's controllers, which typically have limited computational power, without the need for extra hardware to process the measurement data.

1.6 Paper structure

The remainder of this paper is organized as follows: in Sect. 2 our novel step-by-step calibration approach is presented. The materials that we used in our experiments to test and validate the calibration procedure are described in Sect. 3. The experimental results and related discussion are presented in Sects. 4 and 5 respectively. Finally, in Sect. 6 our conclusions are drawn.

2 Proposed kinematic calibration approach

The proposed calibration procedure requires the measurement of the distance between the end-effector and a fixed location, for several poses of the robot. For this purpose, we used a draw-wire encoder fixed to the table on which the robot

stands; the other end of the wire is attached to the robot's flange by a specifically designed fixture. If the fixed location of the draw-wire encoder is known, the distance between the end-effector and the draw-wire encoder exit point at each calibration point can be easily computed as

$$d_i = \sqrt{\Delta x_{i-0}^2 + \Delta y_{i-0}^2 + \Delta z_{i-0}^2} \\ = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2}, \quad (1)$$

where (x_i, y_i, z_i) are the coordinates of the i th calibration point and (x_0, y_0, z_0) are the coordinates of the fixed location of the draw-wire encoder. For a n degrees-of-freedom (DOF) manipulator, the coordinates of the i th calibration point with respect to the robot's base frame can be computed through the direct kinematics, which is a non-linear function of the manipulator's joint variables and geometric parameters

$$\begin{cases} x_i = f_x(\boldsymbol{\theta}_i, \boldsymbol{p}) \\ y_i = f_y(\boldsymbol{\theta}_i, \boldsymbol{p}) \\ z_i = f_z(\boldsymbol{\theta}_i, \boldsymbol{p}) \end{cases}, \quad (2)$$

where $\boldsymbol{\theta}_i = [\theta_i, \dots, \theta_n]$ are the joint coordinates of the robot at the i th calibration point and $\boldsymbol{p} = [p_1, \dots, p_j]$ is the vector containing the geometric parameters of the manipulator, which depend on the model chosen to describe the robot's kinematics. At each calibration point, the computed distance will be different from the measured distance, due to the discrepancy between the nominal and actual values of the model parameters. We focused on identifying only kinematic parameters, neglecting non-kinematic parameters, as inaccurate knowledge of kinematic parameters yields about 90% of the total positioning inaccuracy [24, 46]. In particular, both the joint coordinates $\boldsymbol{\theta}$ and the geometric parameters \boldsymbol{p} are affected by errors, which can be seen as offsets from their nominal values. At this stage, we need to define the set of kinematic error parameters that we wish to estimate (for example, we may only be interested in identifying joint offsets). These error parameters can be grouped in vector $\boldsymbol{e} = [e_1, \dots, e_m]$, while their offsets from their nominal value can be grouped in the error vector $\boldsymbol{\delta e} = [\delta e_1, \dots, \delta e_m]$. Given the nominal geometric parameters of the manipulator \boldsymbol{p}_n and the joint coordinates at the i th calibration point $\boldsymbol{\theta}_i$, the spatial coordinates in Eq. 2 can be rewritten as a function of the error vector $\boldsymbol{\delta e}$ and the distance in Eq. 1 can be rewritten as

$$d_i(\boldsymbol{\delta e}) = \sqrt{(x_i(\boldsymbol{\delta e}) - x_0)^2 + (y_i(\boldsymbol{\delta e}) - y_0)^2 + (z_i(\boldsymbol{\delta e}) - z_0)^2}. \quad (3)$$

The difference between the computed $d_i(\boldsymbol{e})$ and measured distance \tilde{d}_i , known as the residual, is calculated at each

calibration point. The aggregate sum of squares of the residuals over N calibration points is then calculated as

$$f_{cost}(\boldsymbol{\delta e}) = \sum_{i=1}^N (d_i(\boldsymbol{\delta e}) - \tilde{d}_i)^2. \quad (4)$$

Calibration may be posed as the problem of systematically varying the error parameters \boldsymbol{e} in order to minimize the error in wire length over the set of measurement points: the quantity in Eq. 4 is used as a cost function to be minimized by error parameter estimation. The optimal error vector $\boldsymbol{\delta e}^*$ is calculated as

$$\boldsymbol{\delta e}^* = \arg \min_{\boldsymbol{\delta e} \in \boldsymbol{\Omega}} f_{cost}(\boldsymbol{\delta e}), \quad (5)$$

where $\boldsymbol{\Omega}$ is the set of allowable deviations of the error parameters. This is a non-linear least-squares optimization problem that can be solved using various optimization algorithms.

2.1 Encoder location

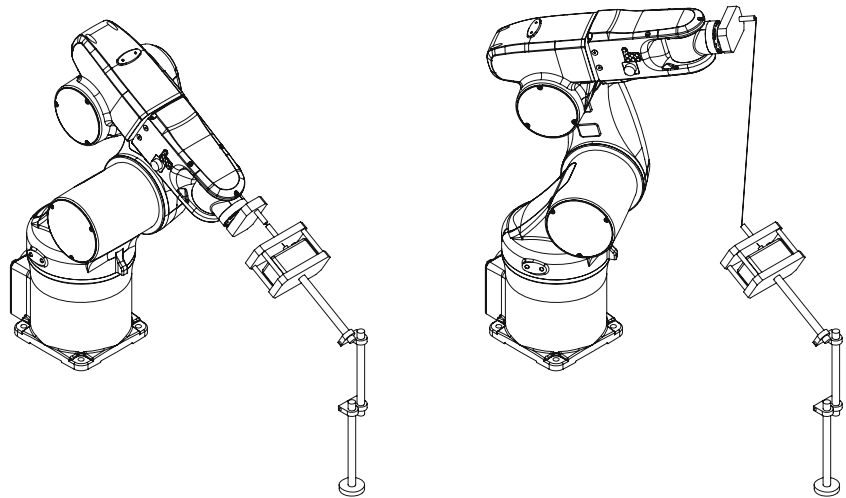
Equation 3 requires the precise knowledge of the encoder location with respect to the robot's base frame, which is not easy to obtain. In previous studies [40, 43], encoder location coordinates were treated as additional unknown parameters that were identified by the optimization algorithm after an initial estimate. In the work proposed here, we first moved the robot to the configuration specified by the joint coordinates $\boldsymbol{\theta}_0$ and then mounted the draw-wire encoder so that the two tips touch when the robot is in this configuration (i.e., the measured distance is zero), as depicted in Fig. 1. In this case, (x_0, y_0, z_0) are not independent additional parameters, but they depend on the manipulator's parameters and joint variables through the direct kinematics as well

$$\begin{cases} x_0 = f_x(\boldsymbol{\theta}_0, \boldsymbol{p}) \\ y_0 = f_y(\boldsymbol{\theta}_0, \boldsymbol{p}) \\ z_0 = f_z(\boldsymbol{\theta}_0, \boldsymbol{p}) \end{cases}. \quad (6)$$

However, the actual encoder coordinates are not exactly the ones calculated through Eq. 6, as the joint coordinates $\boldsymbol{\theta}$ and geometric parameters \boldsymbol{p} are affected by errors. Given the nominal geometric parameters of the manipulator \boldsymbol{p}_n and the joint coordinates $\boldsymbol{\theta}_0$, the spatial coordinates in Eq. 6 can also be written as functions of the error vector $\boldsymbol{\delta e}$. Therefore, the computed distance expressed in Eq. 3 can be rewritten as

$$d_i(\boldsymbol{\delta e}) = \sqrt{(x_i(\boldsymbol{\delta e}) - x_0(\boldsymbol{\delta e}))^2 + (y_i(\boldsymbol{\delta e}) - y_0(\boldsymbol{\delta e}))^2 + (z_i(\boldsymbol{\delta e}) - z_0(\boldsymbol{\delta e}))^2}. \quad (7)$$

Fig. 1 System at the configuration specified by the joint coordinates θ_0 (left) and at the i th calibration configuration, specified by the joint coordinates θ_i (right)



Both the i th calibration point and encoder location coordinates are written as a function of the same set of error parameters e , without the need for three additional parameters.

2.2 Calibration points

Once we defined the cost function and the location of the encoder, we were interested in finding an optimal set of calibration points, which is a well-known issue of robot calibration. In fact, the resulting accuracy after calibration is strongly dependent on the selection of the measurement poses [47, 48]. Depending on the selected pose in the robot’s workspace, a variation in one of the error parameters could produce a small or large error on the end-effector pose. Ideally, a variation in any of the error parameters should cause the maximum possible error on the gripper pose so that the effect of noise (due to unmodeled error sources and measurement errors) can be minimized and, consequently, all of the error parameters can be accurately identified. To measure the goodness of a set of calibration points, different observability indices [49], which are based on the singular value decomposition (SVD) of the Jacobian identification, have been proposed. Unfortunately, this approach is not applicable in our case, as we do not measure the end-effector pose directly, and it is not possible to derive a Jacobian matrix relative to the distance error. Both the coordinates of the i th calibration point and of the fixed location of the encoder depend on the direct kinematics of the manipulator; these coordinates appear in Eq. 7 subtracted from each other, then squared and added. For this reason, rather than evaluating the contribution of the different error components on the end-effector pose, we evaluated their contribution on the distance from a fixed location specified in the joint space.

First, to understand the influence of the different error parameters at the i th calibration point, we can calculate the partial derivatives of Eq. 2 with respect to the vector e

$$\Phi_i = \begin{bmatrix} \partial f_x / \partial e \\ \partial f_y / \partial e \\ \partial f_z / \partial e \end{bmatrix}_{\substack{\theta = \theta_i \\ p = p_n}} = \begin{bmatrix} \partial f_x / \partial e_1 & \dots & \partial f_x / \partial e_m \\ \vdots & \ddots & \vdots \\ \partial f_z / \partial e_1 & \dots & \partial f_z / \partial e_m \end{bmatrix}_{\substack{\theta = \theta_i \\ p = p_n}} \quad (8)$$

This is a $3 \times m$ matrix (corresponding to half of the analytic Jacobian of the robot), where m is the number of unknown parameters that we are trying to estimate, and it is evaluated at nominal conditions, i.e., $\theta = \theta_i$ and $p = p_n$. Similarly, we can calculate the partial derivatives of Eq. 6 with respect to the vector e

$$\Phi_0 = \begin{bmatrix} \partial f_x / \partial e \\ \partial f_y / \partial e \\ \partial f_z / \partial e \end{bmatrix}_{\substack{\theta = \theta_0 \\ p = p_n}} = \begin{bmatrix} \partial f_x / \partial e_1 & \dots & \partial f_x / \partial e_m \\ \vdots & \ddots & \vdots \\ \partial f_z / \partial e_1 & \dots & \partial f_z / \partial e_m \end{bmatrix}_{\substack{\theta = \theta_0 \\ p = p_n}} \quad (9)$$

In Eq. 7 the coordinates of the i th calibration point and the coordinates of the encoder location appear subtracted from each other. The contribution of the different error parameters on the quantities Δx_{i-0} , Δy_{i-0} and Δz_{i-0} that appear in Eq. 1 is given by

$$\Phi_{i-0} = \Phi_i - \Phi_0 \quad (10)$$

The draw-wire encoder only supplies a 1D radial measurement of the wire: rather than the influence of the different parameters on Δx_{i-0} , Δy_{i-0} and Δz_{i-0} , we are interested

in the influence of the different parameters on the direction of the wire. For this purpose, let us define

$$\begin{cases} \Delta x_{i-0,th} = f_x(\boldsymbol{\theta}_i, \mathbf{p}_n) - f_x(\boldsymbol{\theta}_0, \mathbf{p}_n) \\ \Delta y_{i-0,th} = f_y(\boldsymbol{\theta}_i, \mathbf{p}_n) - f_y(\boldsymbol{\theta}_0, \mathbf{p}_n) \\ \Delta z_{i-0,th} = f_z(\boldsymbol{\theta}_i, \mathbf{p}_n) - f_z(\boldsymbol{\theta}_0, \mathbf{p}_n) \end{cases}, \quad (11)$$

i.e., the theoretical values of Δx_{i-0} , Δy_{i-0} and Δz_{i-0} (evaluated with nominal parameters). With these quantities, we can define the vector

$$\mathbf{v}_i = [\Delta x_{i-0,th} \ \Delta y_{i-0,th} \ \Delta z_{i-0,th}], \quad (12)$$

and the corresponding unit vector $\hat{\mathbf{v}}_i$, which represents the theoretical direction of the wire at the i th calibration point

$$\hat{\mathbf{v}}_i = \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}. \quad (13)$$

If the error vector caused by one error component is nearly perpendicular to the direction of the wire, this error component will have little influence on the residual that appears in Eq. 4. On the other hand, if the error vector caused by that error component is parallel to the direction of the wire, this error component will have a great influence on the residual. For this reason, to evaluate the contribution of the j th error component at the i th calibration point, we can compute the projection of the j th column of Φ_{i-0} in the direction of the theoretical distance of the wire

$$\Psi_{ij} = \langle \Phi_{i-0_j}, \hat{\mathbf{v}}_i \rangle. \quad (14)$$

Consequently, the influence of the different error parameters e on the wire length can be expressed by the following matrix

$$\Psi_i = [\Psi_{i_1} \ \dots \ \Psi_{i_m}] = [\langle \Phi_{i-0_1}, \hat{\mathbf{v}}_i \rangle \ \dots \ \langle \Phi_{i-0_m}, \hat{\mathbf{v}}_i \rangle]. \quad (15)$$

This matrix allows the evaluation of the contribution of each error component on the residual at a definite pose in the robot's workspace: the higher each component of Ψ_i , the higher the contribution of the related error component on the wire length. When analyzing the matrix in Eq. 15 it is important to properly compare terms related to angular errors and terms related to linear errors. If Ψ_j is related to an angular error, it is expressed in mm/rad, while if Ψ_k is related to a linear error, it is dimensionless. Considering that angular errors are typically in the range of $\pm 1^\circ$ while linear errors are in the range of ± 1 mm, we can convert Ψ_j in $\text{mm}/^\circ$: in this way, both Ψ_j and Ψ_k multiply a term that is within the range ± 1 of their respective measurement units.

The matrix Ψ_i not only allows the evaluation of the contributions of different error parameters on the residual at a

specific point in the workspace but also gives us a key insight: since the position of the draw-wire encoder is specified by the joint coordinates $\boldsymbol{\theta}_0$, by appropriately choosing the joint coordinates $\boldsymbol{\theta}_i$, some of the terms in Ψ_i may become null. In particular, we can find a set of joint coordinates $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_i$ so that

$$(\boldsymbol{\theta}_i, \boldsymbol{\theta}_0) = \{\boldsymbol{\theta}_i, \boldsymbol{\theta}_0 \in \mathbb{R}^n \mid \Psi_{ij} \neq 0, \Psi_{ik} = 0 \ \forall k \neq j\}. \quad (16)$$

This means that all the elements of Ψ_i are null except for Ψ_{ij} and this is possible if some of the joint coordinates between $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_i$ are kept constant while some others are changed (typically the last two in the case of a 6 DOF anthropomorphic robot arm). In this case, the difference between the predicted and measured distance depends only on the j th error parameter: the cost function becomes a function of δe_j only, and the remaining parameters are kept constant at their nominal values. By choosing a proper set of calibration points $S_j(\boldsymbol{\theta})$ that satisfy Eq. 16, the j th error parameter can be accurately estimated by minimizing the cost function as:

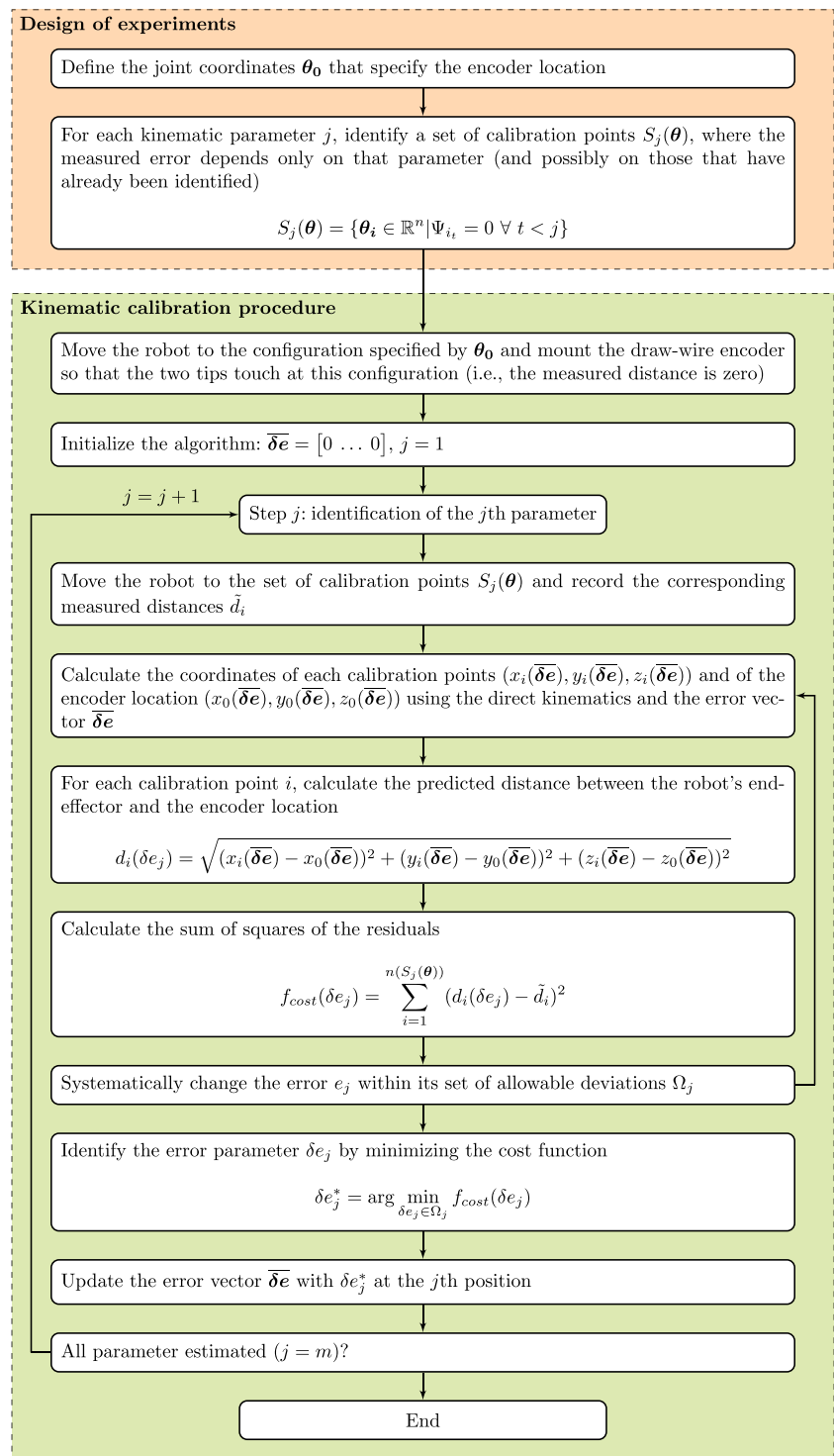
$$\delta e_j^* = \arg \min_{\delta e_j \in \Omega_j} f_{cost}(\delta e_j). \quad (17)$$

After identifying the j th error parameter, we can find a second set of calibration points $S_w(\boldsymbol{\theta})$ where all the elements of Ψ_i are null except for Ψ_{ij} and Ψ_{iw} : this allows the estimation of the w th parameter. In fact, the cost function becomes a function of δe_w only, while the remaining parameters are kept constant (δe_j is kept at the value found at the previous step with Eq. 17 and the remaining parameters are kept at their nominal value). Following this procedure and finding different sets of points with these properties, we can estimate the remaining error parameters, one at a time. In addition, once we determine the order in which the parameters will be identified, e can be rearranged in vector \bar{e} and Ψ_i in $\bar{\Psi}_i$, so that the order of the columns reflects the order in which the parameters are estimated. For each unknown parameter j , the corresponding set of calibration points can be expressed as:

$$S_j(\boldsymbol{\theta}) = \{\boldsymbol{\theta}_i \in \mathbb{R}^n \mid \Psi_{it} = 0 \ \forall t > j\}. \quad (18)$$

As a result, we propose the step-by-step kinematic calibration procedure described in Fig. 2, through which the error parameters are subsequently identified, one at a time. This constitutes a major advantage over standard calibration approaches, where the measured error depends on all the parameters and the unknown parameters are estimated all together, often with complex optimization algorithms. In fact, if the measured error depends on all the error parameters, it is more difficult to accurately estimate each of them, and the optimization algorithm may converge to a set of error

Fig. 2 Flowchart of the proposed step-by-step calibration procedure



parameters that minimize the cost function, but do not represent the actual error parameter values.

The validity of the proposed step-by-step calibration procedure can also be demonstrated with a different approach, described in the Appendix A.

3 Materials and methods

3.1 Industrial manipulator: the Adept Viper S650

To test the proposed calibration approach, we used the Adept Viper S650 robot and the eMB-60R controller, both

Table 1 DH parameters of the Adept Viper S650 manipulator

i	α_{i-1} (°)	a_{i-1} (mm)	θ_i (°)	d_i (mm)
1	0	0	θ_1	0
2	−90	75	θ_2	0
3	0	270	θ_3	0
4	90	−90	θ_4	295
5	−90	0	θ_5	0
6	90	0	θ_6	80

manufactured by Omron Adept Inc. The Adept Viper S650 is an articulated robot with six degrees of freedom and has a repeatability of $\pm 0.02\text{mm}$. The robot controller uses DH parameters, and the DH table according to the convention described in [50] is presented in Table 1.

3.2 Measurement system: draw-wire encoder

The measurement system used to obtain 1D distance measurements consists of a draw-wire encoder manufactured by Unimeasure Inc. (model EP-50-N20-N30-10C) and a specifically designed tool to be attached directly to the robot's flange. The draw-wire encoder is composed of a reel on which the wire is wound, an incremental encoder, and electronic components for signal conditioning. The incremental encoder measures the rotation of the reel, and hence the amount of extracted wire, while the spring maintains the proper tension cable. The main specifications of this measuring system are a resolution of 0.025488mm and a maximum measured length of 1250mm , which is suitable for our experiments, as the manipulator's workspace is approximately a hemisphere with a radius of about 600mm . The draw-wire encoder can be easily placed in different positions and orientations, thanks to its holding fixture. In addition, the draw-wire encoder is connected to the robot controller through the belt encoder port (which is typically used for conveyor tracking tasks). Measurement data are acquired directly from the robot controller, which then proceeds to process the data and identify error parameters.

3.3 Kinematic calibration model

Since we wanted our kinematic calibration procedure to be directly transferable to the robot controller, we used the DH convention to derive the kinematic model of the manipulator. In particular, we followed the approach described in [40] to derive the so-called DH(-) calibration model. As stated in Sect. 1, a kinematic calibration model should meet the three principles of completeness, continuity, and minimality. The standard DH model is not parametrically continuous for the Adept Viper S650 as joint axes 2 and 3 are parallel: a minor

misalignment between those axes can result in major changes to the remaining DH parameters. For this reason, as in [9, 10, 40], we chose not to perturb the link twist α_2 . In addition, in our robot controller it was not possible to modify the nominally zero-valued DH parameters: consequently, we did not consider those parameters in our calibration model, as in [40]. Furthermore, using the proposed calibration approach, the error related to the first joint angle $\delta\theta_1$ cannot be estimated, as this quantity shifts all calibration points by the same amount, while the distances between these points (which are the measurements used) remain unchanged. Finally, to avoid redundancy, either the error parameters $\delta\theta_6$ and δd_6 or the error parameters related to the position of the tool frame ($\delta x_{tool}, \delta y_{tool}, \delta z_{tool}$) can be estimated. If the tool used during calibration is not the same as that used during robot operations, finding the error parameters related to the tool position is not effective. For this reason, we used a tool of known dimensions mounted directly on the robot and decided to estimate the errors associated with the last link, i.e., $\delta\theta_6$ and δd_6 . The resulting kinematic calibration model is presented in Table 2 and is composed of only 10 error parameters, which can be grouped as

$$\delta e = [\delta\theta_2 \ \delta\theta_3 \ \delta\theta_4 \ \delta\theta_5 \ \delta\theta_6 \ \delta a_1 \ \delta a_2 \ \delta a_3 \ \delta d_4 \ \delta d_6] \quad (19)$$

Although this kinematic calibration model is incomplete, it has been proven to be robust and effective [40].

3.4 Encoder location and calibration points

The proposed calibration approach first requires the definition of the joint coordinates θ_0 that identify the position of the draw-wire encoder. The draw-wire encoder should be placed in a position within the robot's workspace where it can be easily set up and where it does not hinder the robot's movements, taking into account potential obstacles that may be present near the robot. For this reason, we chose $\theta_0 = [0, -90, 210, -90, 0, -90]$ as the initial configuration of the robot. Once θ_0 has been defined, the kinematic

Table 2 DH kinematic calibration model for the Adept Viper S650 manipulator

i	α_{i-1} (°)	a_{i-1} (mm)	θ_i (°)	d_i (mm)
1	0	0	θ_1	0
2	−90	$75 + \delta a_1$	$\theta_2 + \delta\theta_2$	0
3	0	$270 + \delta a_2$	$\theta_3 + \delta\theta_3$	0
4	90	$-90 + \delta a_3$	$\theta_4 + \delta\theta_4$	$295 + \delta d_4$
5	−90	0	$\theta_5 + \delta\theta_5$	0
6	90	0	$\theta_6 + \delta\theta_6$	$80 + \delta d_6$

calibration procedure then requires the identification of a different set of calibration points for each unknown parameter. This is carried out by analyzing matrix Ψ_i , which for our manipulator can be expressed as

$$\Psi_i = [\Psi_{i\theta_2} \ \Psi_{i\theta_3} \ \Psi_{i\theta_4} \ \Psi_{i\theta_5} \ \Psi_{i\theta_6} \ \Psi_{ia_1} \ \Psi_{ia_2} \ \Psi_{ia_3} \ \Psi_{id_4} \ \Psi_{id_6}], \tag{20}$$

with different values of joint coordinates θ_i . This analysis also determines the order in which the error parameters are identified. Because our robot is an open chain manipulator, the first error parameters that can be estimated are the one closest to the end of the kinematic chain (i.e., the ones related to the last link) and subsequently all of the remaining ones. In fact, the analysis of Ψ_i with different values of θ_i determined that the order in which the parameters are identified is: $\delta\theta_6$, δd_6 , $\delta\theta_5$, $\delta\theta_4$, δa_3 , δd_4 , δa_2 , $\delta\theta_3$, $\delta\theta_2$ and δa_1 . Consequently, the error vector δe can be rearranged as

$$\bar{\delta e} = [\delta\theta_6 \ \delta d_6 \ \delta\theta_5 \ \delta\theta_4 \ \delta a_3 \ \delta d_4 \ \delta a_2 \ \delta\theta_3 \ \delta\theta_2 \ \delta a_1], \tag{21}$$

and matrix Ψ_i can be rearranged as

$$\bar{\Psi}_i = [\Psi_{i\theta_6} \ \Psi_{id_6} \ \Psi_{i\theta_5} \ \Psi_{i\theta_4} \ \Psi_{ia_3} \ \Psi_{id_4} \ \Psi_{ia_2} \ \Psi_{i\theta_3} \ \Psi_{i\theta_2} \ \Psi_{ia_1}]. \tag{22}$$

After determining the order in which the error parameters are identified, we defined the 10 sets of calibration points, one for each unknown error parameter. Because the analytical expressions for the calibration points that satisfy Equation (18) and ensure the highest value of Ψ_{ij} is quite difficult to obtain, we used an optimization program to find such points. The joint coordinates θ_i and the corresponding matrix $\bar{\Psi}_i$ of each calibration point are reported in Table 3.

Before the experiments were performed, computer simulations were conducted to verify the correct functioning of the step-by-step kinematic calibration procedure and the goodness of the sets of calibration points. This was achieved by perturbing the nominal parameters by known amounts, producing artificial measured distance data (also taking into account the encoder’s resolution and random noise), and checking that the identification process converged to the

Table 3 Joint coordinates θ_i and corresponding matrix $\bar{\Psi}_i$ of the different calibration points

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	$\Psi_{i\theta_6}$	Ψ_{id_6}	$\Psi_{i\theta_5}$	$\Psi_{i\theta_4}$	Ψ_{ia_3}	Ψ_{id_4}	Ψ_{ia_2}	$\Psi_{i\theta_3}$	$\Psi_{i\theta_2}$	Ψ_{ia_1}
Step 1: identification of $\delta\theta_6$															
0	-90	210	-90	-26	-180	-1.07	0	0	0	0	0	0	0	0	0
0	-90	210	-90	-26	-170	-1.08	0	0	0	0	0	0	0	0	0
0	-90	210	-90	-26	-190	-1.07	0	0	0	0	0	0	0	0	0
0	-90	210	-90	26	0	1.07	0	0	0	0	0	0	0	0	0
0	-90	210	-90	26	10	1.07	0	0	0	0	0	0	0	0	0
0	-90	210	-90	26	-10	1.08	0	0	0	0	0	0	0	0	0
Step 2: identification of δd_6															
0	-90	210	-90	-90	-90	0	1.41	0	0	0	0	0	0	0	0
0	-90	210	-90	-90	-85	0.03	1.41	0	0	0	0	0	0	0	0
0	-90	210	-90	-90	-95	-0.03	1.41	0	0	0	0	0	0	0	0
0	-90	210	-90	90	-90	0	1.41	0	0	0	0	0	0	0	0
0	-90	210	-90	90	-85	0.03	1.41	0	0	0	0	0	0	0	0
0	-90	210	-90	90	-95	-0.03	1.41	0	0	0	0	0	0	0	0
Step 3: identification of $\delta\theta_5$															
0	-90	210	-135	10	-90	0.09	0.17	0.2	0	0	0	0	0	0	0
0	-90	210	-135	-5	-90	0.1	-0.08	-0.22	0	0	0	0	0	0	0
0	-90	210	45	0	-270	0	0	-4.19	0	0	0	0	0	0	0
0	-90	210	45	10	-270	0.29	-0.13	-3.9	0	0	0	0	0	0	0
0	-90	210	45	-5	-270	-0.06	0.08	-4.17	0	0	0	0	0	0	0
0	-90	210	-45	-5	-90	0.06	0.08	-0.13	0	0	0	0	0	0	0
0	-90	210	-45	10	-90	-0.29	-0.13	0.62	0	0	0	0	0	0	0
0	-90	210	135	0	-270	0	0	4.19	0	0	0	0	0	0	0
0	-90	210	135	-5	-270	-0.1	-0.08	4.15	0	0	0	0	0	0	0
0	-90	210	135	10	-270	0.09	0.17	4.13	0	0	0	0	0	0	0

Table 3 continued

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	$\Psi_{i\theta_6}$	Ψ_{id_6}	$\Psi_{i\theta_5}$	Ψ_{id_4}	Ψ_{ia_3}	Ψ_{id_4}	Ψ_{ia_2}	$\Psi_{i\theta_3}$	$\Psi_{i\theta_2}$	Ψ_{ia_1}
Step 4: identification of $\delta\theta_4$															
0	-90	206	0	45	-90	-1.3	0.18	2.82	0.43	-0.04	0	0	0	0	0
0	-90	206	10	45	-90	-1.29	0.17	2.8	0.46	-0.03	0.01	0	0	0	0
0	-90	206	-10	45	-90	-1.29	0.22	2.81	0.38	-0.05	-0.01	0	0	0	0
0	-90	239.93	0	-78.93	-90	-0.46	0.18	0.99	-3.04	0.27	0.08	0	0	0	0
0	-90	239.93	10	-78.93	-90	-0.16	0.21	0.34	-2.62	0.36	0.03	0	0	0	0
0	-90	239.93	-10	-78.93	-90	-0.58	0.22	1.26	-3.15	0.22	0.13	0	0	0	0
0	-90	239.93	0	-78.93	90	0.46	0.18	-2.81	3.04	0.27	0.08	0	0	0	0
0	-90	239.93	10	-78.93	90	0.58	0.22	-2.67	3.15	0.22	0.13	0	0	0	0
0	-90	239.93	-10	-78.93	90	0.16	0.21	-2.93	2.26	0.36	0.03	0	0	0	0
0	-90	206	0	45	90	0.13	0.18	-0.97	-0.43	-0.04	0	0	0	0	0
0	-90	206	10	45	90	0.29	0.22	-0.49	-0.38	-0.05	-0.01	0	0	0	0
0	-90	206	-10	45	90	0.29	0.17	-1.19	-0.46	-0.03	0.01	0	0	0	0
Step 5: identification of δa_3															
0	-90	230	-65	-80	0	-1.15	1.15	0.19	-1.72	0.25	0	0	0	0	0
0	-90	230	-60	-80	0	-1.13	1.13	0.1	-1.69	0.26	-0.02	0	0	0	0
0	-90	230	-70	-80	0	-1.16	1.17	0.26	-1.73	0.25	0.02	0	0	0	0
0	-90	230	65	-80	0	1.15	1.15	-2.92	1.72	0.25	0	0	0	0	0
0	-90	230	70	-80	0	1.16	1.17	-2.88	1.73	0.25	0.02	0	0	0	0
0	-90	230	60	-80	0	1.13	1.13	-2.96	1.69	0.26	-0.02	0	0	0	0
Step 6: identification of δd_4															
0	-90	130	-90	0	-90	0	1.28	0	0	-0.09	1.28	0	0	0	0
0	-90	135	-90	0	-90	0	1.21	0	0	-0.09	1.21	0	0	0	0
0	-90	140	-90	0	-90	0	1.14	0	0	-0.08	1.14	0	0	0	0
0	-90	125	-90	0	-90	0	1.35	0	0	-0.1	1.35	0	0	0	0
0	-90	120	-90	0	-90	0	1.41	0	0	-0.1	1.41	0	0	0	0
0	-90	150	-90	0	-90	0	1.0	0	0	-0.07	1.0	0	0	0	0
Step 7: identification of δa_2															
0	-110	230	-90	0	-90	0	0	0	0	0	0	0.35	0	0	0
0	-110	235	-90	0	-90	0	0	0	0	0	0	0.43	0	0	0
0	-110	240	-90	0	-90	0	0	0	0	0	0	0.52	0	0	0
0	-110	245	-90	0	-90	0	0	0	0	0	0	0.6	0	0	0
0	-110	250	-90	0	-90	0	0	0	0	0	0	0.68	0	0	0
0	-110	255	-90	0	-90	0	0	0	0	0	0	0.77	0	0	0
Step 8: identification of $\delta\theta_3$															
0	-110	175	-90	0	-90	0	0.88	0	0	-0.28	0.88	0.03	-1.63	0	0
0	-110	180	-90	0	-90	0	0.8	0	0	-0.28	0.8	0.02	-1.63	0	0
0	-110	185	-90	0	-90	0	0.72	0	0	-0.27	0.72	0.02	-1.63	0	0
0	-120	190	-90	0	-90	0	0.76	0	0	-0.38	0.76	0.06	-2.42	0	0
0	-120	195	-90	0	-90	0	0.67	0	0	-0.37	0.67	0.06	-2.42	0	0
0	-120	185	-90	0	-90	0	0.84	0	0	-0.39	0.84	0.06	-2.42	0	0
Step 9: identification of $\delta\theta_2$															
-20	-95	210	-90	-90	0	0.39	0.8	-1.93	0.95	-0.08	0.04	0.05	0.02	0.39	0.01
-20	-90	210	-90	-90	0	0.36	0.77	-2.06	0.85	-0.01	0.01	0	0.34	0.41	0.01
-20	-85	210	-90	-90	0	0.32	0.74	-2.21	0.72	0.07	-0.01	-0.05	0.63	0.42	0.02

Table 3 continued

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	$\Psi_{i\theta_6}$	Ψ_{id_6}	$\Psi_{i\theta_5}$	$\Psi_{i\theta_4}$	Ψ_{ia_3}	Ψ_{id_4}	Ψ_{ia_2}	$\Psi_{i\theta_3}$	$\Psi_{i\theta_2}$	Ψ_{ia_1}
-20	0	15	-90	-90	0	0.71	0.77	1.49	1.62	-0.31	1.55	-1.38	1.65	0.41	0.01
-20	-5	15	-90	-90	0	0.65	0.79	1.66	1.52	-0.3	1.6	-1.31	1.67	0.4	0
-20	5	15	-90	-90	0	0.78	0.74	1.29	1.73	-0.33	1.5	-1.43	1.58	0.43	0.02
Step 10: identification of δa_1															
-160	-125	5	-90	-90	0	0.97	0.54	-2.14	1.85	-1.44	-0.01	-0.49	-8.99	-1.68	1.89
-160	-120	5	-90	-90	0	0.8	0.63	-2.48	1.47	-1.17	-0.06	-0.47	-7.72	0.46	1.94
-160	-115	5	-90	-90	0	0.6	0.71	-2.73	1.03	-0.86	-0.09	-0.42	-6.22	2.58	1.97
-160	-110	5	-90	-90	0	0.39	0.77	-2.85	0.58	-0.55	-0.09	-0.34	-4.62	4.49	1.96
-160	-105	5	-90	-90	0	0.19	0.81	-2.88	0.14	-0.25	-0.06	-0.26	-3.08	6.08	1.92

perturbed values. This step was necessary not only to validate the proposed approach but also because, for some parameters, we could not find points where $\Psi_{it} = 0 (\forall t \neq j)$ was exactly zero. For example, from Table 3 it can be seen that the distance error at the calibration points used for the identification of $\delta\theta_4$ also depends slightly on δa_3 and δd_4 (both of which will be identified after $\delta\theta_4$). Similarly, the distance error at the calibration points used for the identification of δa_3 depends slightly on δd_4 (which is identified in the following step). Nonetheless, simulations carried out with different values of the error parameters showed that the identification process converged to the correct values.

3.5 Experimental setup of the calibration procedure

Given the effects of temperature changes on the geometrical properties of the manipulator, all tests were performed while monitoring both the room temperature and the temperature of the motor driving circuits. In fact, experimental tests only started when the motor amplifier temperature stabilized to its steady state value after the initial transient that starts when enabling robot power. More in detail, the room temperature was kept constant at 25°, while the temperature of the motor driving circuits stabilized between 57° and 59°. The kinematic calibration of the Adept Viper S650 was then carried out following the proposed step-by-step procedure (described in the flowchart in Fig. 2), moving the robot through the sets of calibration points reported in Table 3. More in detail, at the beginning of each step of the procedure, the robot was moved to the configuration specified by the joint coordinates θ_0 : at each step we checked whether the measured distance was still zero, to ensure that the encoder was working properly and no wire deformation occurred. The robot was then moved to the set of calibration points; at each calibration point, we waited three seconds before recording the wire length, to allow any dynamic oscillations to subside. In addition, moving the robot between a set of calibration points without twisting the wire required the definition of a

number of intermediate path points (via points). Finally, once the calibration procedure ended and we estimated all error parameters, we implemented error compensation by modifying the nominal parameters embedded in the robot controller.

Figure 3 shows the manipulator during the calibration procedure in the configuration described by θ_0 (where the encoder is mounted so that the measured distance is zero) and in a generic configuration θ_i .

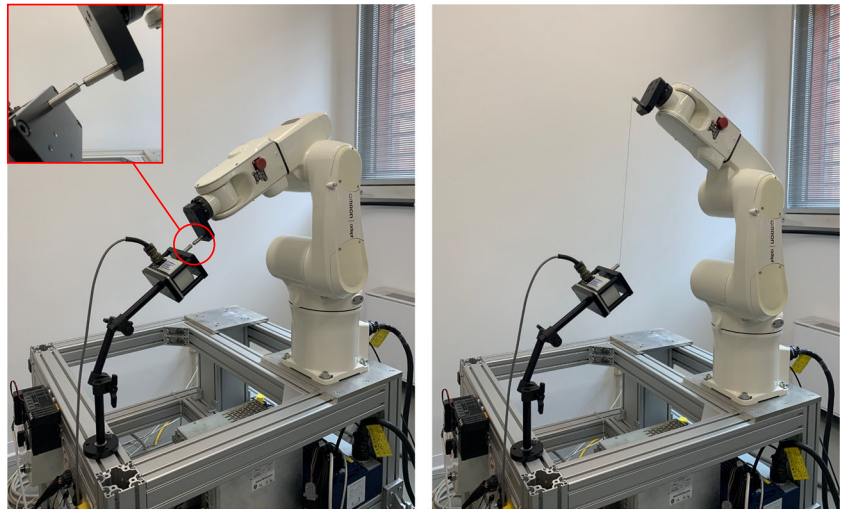
3.6 Validation procedure

Validation is often carried out using the same end-effector and measurement system used to acquire measurement data for calibration: this could produce biased results and an over-estimation of the resulting accuracy after calibration. For this reason, we carried out the validation of our kinematic calibration procedure following two different approaches: the first uses the draw-wire encoder, while the second uses a touch probe and a granite surface plate.

3.6.1 Validation procedure using the draw-wire encoder

When moving a robot within its workspace, the target pose can be specified in the Cartesian space (i.e., with respect to a reference frame) or in the joint space (i.e., specifying each joint variable). If a pose in the robot’s workspace is specified in the Cartesian space, the controller of the robot computes the joint angles necessary to reach that pose through the inverse kinematics. For a specific pose in the robot’s workspace, there could be multiple solutions to the inverse kinematics and they are usually referred to as robot configurations [50]. For a 6 DOF anthropomorphic arm with a spherical wrist, the same pose can be reached with up to 8 different configurations. In particular, there are two possible solutions for θ_1 (typically referred to as “righty” or “lefty”), two possible solutions for θ_3 (“elbow above” or “elbow below”) and two possible solutions for θ_4 (“nonflip” or “flip”). Given a well-calibrated robot, if the configuration of the manipulator

Fig. 3 Experimental setup during the calibration procedure: system at the configuration specified by the joint coordinates θ_0 (left) and at the i th calibration configuration, specified by the joint coordinates θ_i (right)



is changed while the commanded pose remains the same, the Cartesian coordinates should remain unchanged, as well as the distance between the end-effector and a fixed location. Therefore, the variation of this distance can be measured with the draw encoder and used as a metric to evaluate the robot's accuracy.

To assess the robot's accuracy before and after calibration, we first placed the draw-wire encoder outside of the manipulator's workspace, in order not to hinder the robot's movements. We then moved the robot through a set of 50 different poses within the entire robot's workspace, and at each pose the robot's configuration was changed between "elbow above"/"elbow below" and "nonflip/flip" (four possible combinations). At each robot pose and configuration, the distance measured by the draw-wire encoder was recorded. The distances corresponding to the same end-effector pose were compared, and the maximum discrepancy between them was used as a performance metric.

3.6.2 Validation procedure using a touch probe and a granite surface plate

To evaluate the accuracy of the robot before and after calibration, we also used a touch probe mounted on the robot's flange and a granite surface plate placed on the table on which the robot stands. The faces of a granite surface plate are machined with a strict flatness tolerance: all points on the faces lie on the same plane following the planar equation

$$z_i = ax_i + by_i + c, \quad (23)$$

where a , b and c are the constant coefficients that characterize the plane. By using a touch probe, we can move the robot's end-effector until it touches the plate and record the corresponding end-effector coordinates computed by the robot controller. When a suitable number of data points have been

acquired, we can perform the best surface fit in the least-squares sense. The fit of the plane (i.e., the values of the residuals) reflects the accuracy of the robot itself. In fact, this quantity has been effectively used as a cost function for robot calibration in [31]; here, we only use it to assess the robot's accuracy before and after calibration. For this purpose, the recorded coordinates of acquisition points can be grouped in the following column vectors

$$\mathbf{X} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad \mathbf{Z} = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}, \quad (24)$$

and Eq. 23 can be rewritten in matrix form as

$$\mathbf{Z} = a\mathbf{X} + b\mathbf{Y} + c. \quad (25)$$

This is an overconstrained linear system: the coefficients a , b and c can be calculated by least linear squares using Moore-Penrose pseudoinverse. Let us first define

$$\mathbf{\Upsilon} = [\mathbf{X} \ \mathbf{Y} \ \mathbf{1}], \quad (26)$$

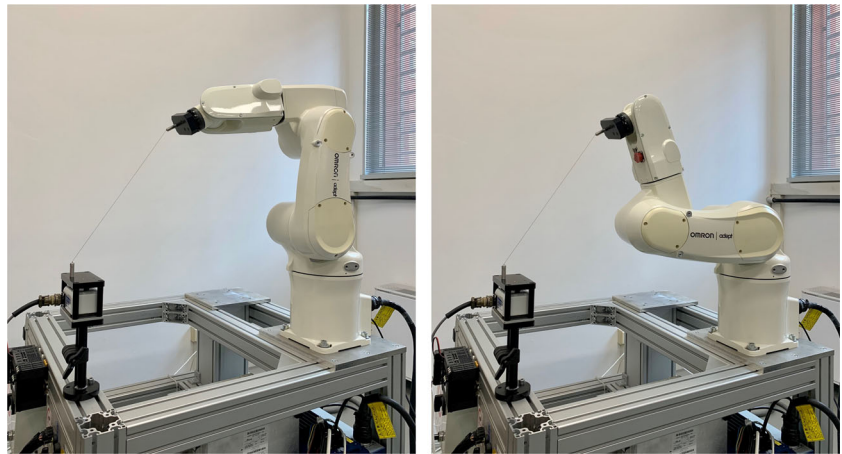
where $\mathbf{1}$ is an all-ones column vector of size n . Then, we can approximately solve for the coefficients as

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = (\mathbf{\Upsilon}^T \mathbf{\Upsilon})^{-1} \mathbf{\Upsilon}^T \mathbf{Z}. \quad (27)$$

For each data acquisition point, the residual from fitting the plane can be calculated as

$$\Delta_i = ax_i + by_i + c - z_i, \quad (28)$$

Fig. 4 Experimental setup during the first validation procedure, where the robot is moved through a set of poses in the workspace, changing its configuration: “above” and “noflip” (left), “below” and “flip” (right)



and this quantity can be used as a performance metric to evaluate the robot’s accuracy before and after calibration (Figs. 4 and 5).

To carry out this validation procedure, we used a 3D touch probe directly connected to the robot controller; its specifications are a resolution of 1mm and a maximum error of 4mm. The granite surface plate that we used is machined with grade 0 accuracy according to the DIN 876 standard and has a flatness tolerance of 6mm; its dimensions are 400

x 250 x 70mm. To acquire the end-effector coordinates of points that lie on the surface plate, we defined a 19×11 grid, for a total of 209 acquisition points. The robot was first moved above each acquisition point and then moved along the z direction until the touch probe detected a contact and the end-effector coordinates were recorded.

4 Results

4.1 Kinematic calibration results

The kinematic calibration procedure described in Sect. 3.5 was fully automated and took approximately 45 min to perform. The estimated error parameters are reported in Table 4. The identified joint offset values appear significantly higher than the geometric parameter offset values: this suggests that most of the robot’s inaccuracy before calibration is due to incorrect offset values used to describe the manipulator’s home position.

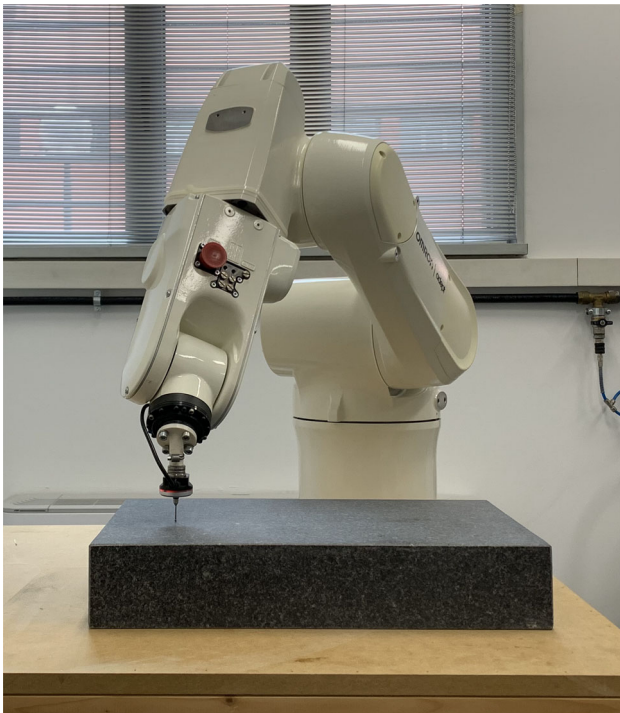
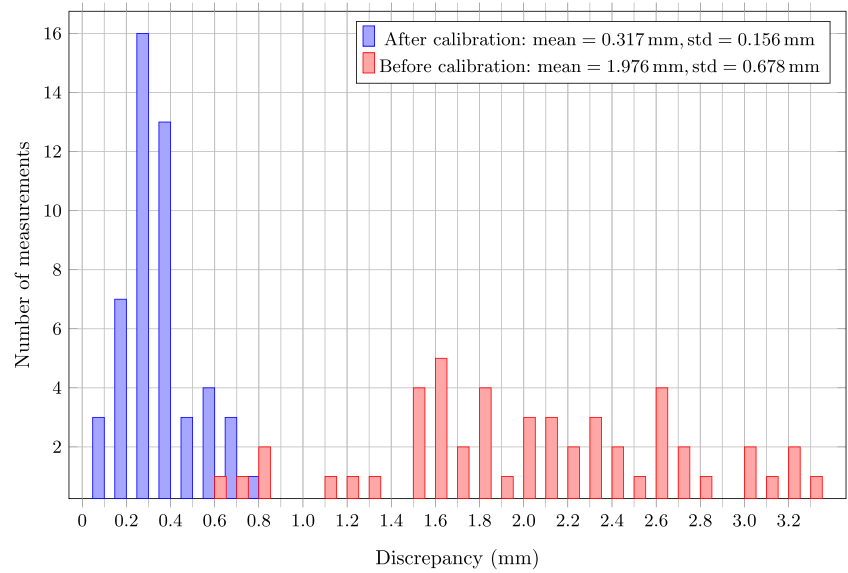


Fig. 5 Experimental setup during the second validation procedure: the touch probe detects the contact with the granite surface plate (the LED light turns red) and the corresponding end-effector coordinates are recorded

Table 4 Kinematic calibration results

Parameter	Nominal value	Estimated offset value
θ_2 (°)	0.000	0.675
θ_3 (°)	0.000	−0.485
θ_4 (°)	0.000	0.245
θ_5 (°)	0.000	−0.575
θ_6 (°)	0.000	−1.215
a_1 (mm)	75.000	−0.005
a_2 (mm)	270.000	0.105
a_3 (mm)	−90.000	0.025
d_4 (mm)	295.000	−0.105
d_6 (mm)	80.000	0.115

Fig. 6 Maximum discrepancy among the recorded distances at the same end-effector pose but different configurations, observed in 50 different poses within the robot's workspace, before and after calibration



4.2 Validation results

4.2.1 Validation using the draw-wire encoder

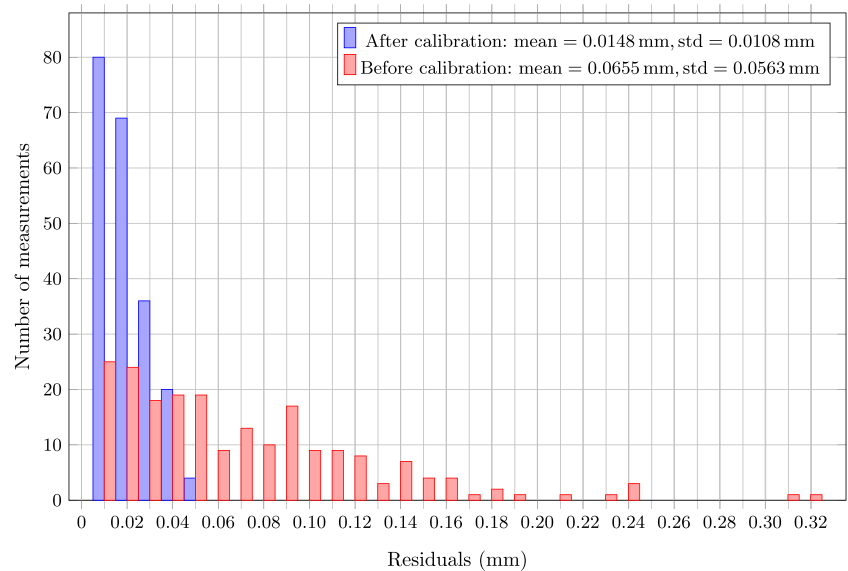
The validation procedure described in Sect. 3.6.1 was carried out before and after calibration: its results are reported in Fig. 6 in the form of a histogram. Before calibration, the discrepancy among the recorded distances (at the same end-effector pose but different configuration) assumes values in the order of some millimeters and presents a high variability between measurements taken in different poses within the robot's workspace. After calibration, the discrepancy assumes values in the order of tenths of millimeters and the variation between different measures is rather small. In fact, after calibration, the mean value of the maximum discrepancy

is reduced by 84% while the standard deviation is reduced by 77%.

4.2.2 Validation using the touch probe and granite surface plate

The second validation procedure, described in Sect. 3.6.2, was carried out before and after calibration: its results are presented in Fig. 7 in the form of a histogram. Figure 7 shows that the residuals of the plane fit are greatly reduced after calibration: before calibration, the residuals are below 0.32 mm, while the maximum deviation obtained after calibration is less than 0.05 mm. In particular, after calibration the mean value of the residuals is decreased by 77% while the standard deviation is decreased by 80%.

Fig. 7 Residuals of the plane fit given a set of 209 points that lies on granite surface plate, before and after calibration



5 Discussion

The results of the two different validation procedures showed a clear improvement in the robot accuracy after calibration, proving the effectiveness of the proposed step-by-step calibration procedure. More in detail, after calibration, the maximum discrepancy among the recorded distances in the same end-effector pose but with different configurations decreased from 3.3mm to 0.8mm. A similar performance metric was used in [40], where distances corresponding to the same end-effector pose but different orientations were recorded and compared: if this deviation is below 2mm, it is safe to assume that the robot is fairly well calibrated for most applications [40]. Since changing the robot configuration while keeping its end-effector pose constant requires a greater movement of the joints than changing its orientation, we can assume that our robot is well calibrated (while before calibration it was not). On the other hand, we cannot compare the results of the second validation procedure with other works, as in [31] the plane fit was used as a cost function during calibration, but validation was carried out using a CMM.

The proposed calibration approach offers several advantages over standard calibration approaches. First, measurement data for the calibration of robots are typically obtained through laser trackers because of their high accuracy. However, they are very expensive, require special training to use, and their calibration is time consuming. Using a single draw-wire encoder, which is much less expensive than laser trackers, reduces the cost of calibration while effectively improving the robot's positioning accuracy. However, since we did not repeat the calibration procedure using laser trackers, it is impossible to claim with certainty that our procedure produces the same (or better) resulting accuracy after calibration. In addition, using standard approaches and measurement systems, the coordinates of the robot's end-effector are acquired in the coordinate system of the instrument and have to be transformed into the base coordinate system of the robot. This transformation typically requires the estimation of six additional parameters, increasing the computational burden of the identification process and potentially introducing errors that could propagate to the final results. Conversely, our approach does not require the explicit knowledge of a reference frame and identification of additional parameters, making calibration more robust.

Our work also differs significantly from previous studies that used a single draw-wire encoder [40–45]. By addressing two aspects that were not addressed in previous works — the optimal encoder location and set of calibration points — we found that it is possible to define a set of calibration points for each unknown parameter: in these points, the difference between measured and predicted distance depends only on that parameter. As a result, we proposed a step-by-step cal-

ibration procedure where the parameters are subsequently identified, one at a time: each parameter is estimated faster and more accurately than with standard approaches. Following standard robot calibration procedures, the robot is moved to a number of calibration points where the measured error depends on all of the unknown error parameters; the error parameters are then estimated simultaneously and often with complex optimization algorithms (which cannot be implemented directly into the robot controllers due to their limited computational power). However, the optimization algorithm may converge to a set of error values that minimize the cost function but do not represent the actual error parameter values. Following the proposed procedure, only one of the unknown parameters is estimated at a time, and fewer measurements are processed at the same time: the identification process is more robust and, since it does not require complex optimization algorithms, it can be implemented directly into the robot controller, without the need for extra hardware to process the data. This is particularly advantageous in industrial environments where computing platforms such as Matlab may not be available.

Nevertheless, the proposed calibration procedure has some potential limitations. The analytical approach used to find the sets of calibration points is based on the linearization of direct kinematics, which is effective only if the errors between actual and nominal parameters are relatively small. However, this may not be the case when calibration is performed after mechanical parts or batteries are replaced: in these instances a coarse calibration should be performed first. In addition, our calibration approach does not allow to find the error related to the first joint angle.

Future studies should aim to compare the resulting accuracy after calibration obtained by the proposed step-by-step procedure with that obtained using standard optimization algorithms and more expensive measurement systems, such as laser trackers. In addition, a higher quality displacement sensor should be considered, in order to evaluate the peak performance of the presented calibration procedure.

6 Conclusions

In this paper, we presented a cost-effective and practical step-by-step kinematic calibration procedure for industrial robots using 1D measurement data obtained through a single draw-wire encoder. The heart of our approach lies in the proper choice of encoder location and calibration points. In particular, we positioned the draw-wire encoder in a location specified in the joint space and developed a novel analytical approach to find different sets of calibration points where the distance error depends only on one of the unknown error parameters. As a result, we proposed a step-by-step calibration procedure through which each error parameter

is subsequently identified: this improves the identification accuracy while reducing the computational burden of the identification process. In fact, since our calibration procedure does not require complex optimization algorithms, it can be implemented directly into the robot controllers without the need for extra hardware to process measurement data. Numerical simulations and calibration experiments on a 6 DOF anthropomorphic arm proved the effectiveness and robustness of the proposed method, which represents a cost-effective and computationally less demanding alternative to standard calibration approaches.

Appendix A: Proof of the proposed analytical approach

The proposed approach to find different sets of calibration points that only depend on one unknown parameter is here demonstrated in the case of a 2 degrees-of-freedom planar arm, depicted in Fig. 8. In this case, the joint coordinates are $\theta = [\theta_1, \theta_2]$ while the geometric parameters are the link lengths $p = [l_1, l_2]$. Without loss of generality, we assume that we only wish to estimate the joint offset: therefore we define $e = [\theta_1, \theta_2]$ and $\delta e = [\delta\theta_1, \delta\theta_2]$. During the calibration procedure, the planar arm is first moved to the configuration specified by the joint coordinates $\theta_0 = [\theta_{10}, \theta_{20}]$ and the encoder is mounted so that the two tips touch when the robot is in this configuration (i.e., the measured distance is null). The planar arm is then moved through a set of N calibration points. Let us consider the i th calibration point,

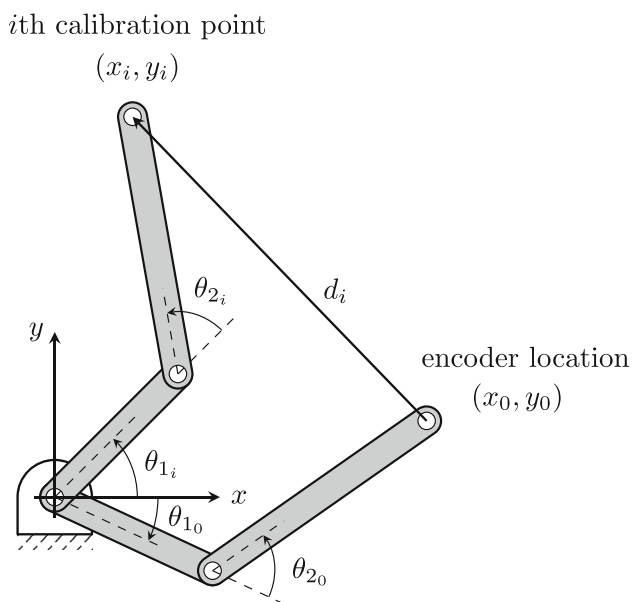


Fig. 8 Planar 2 degrees-of-freedom arm

specified by the joint coordinates $\theta_i = [\theta_{1i}, \theta_{2i}]$ as depicted in Fig. 8. The predicted distance can be easily calculated as

$$d_i = \sqrt{\Delta x_{i-0}^2 + \Delta y_{i-0}^2} = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}. \quad (\text{A1})$$

This quantity is compared with the distance \tilde{d}_i measured by the draw-wire encoder and the discrepancy between the two is used as a cost function. The coordinates of the i th calibration point can be computed through the direct kinematics

$$\begin{cases} x_i = f_x(\theta_i, p_n) \\ y_i = f_y(\theta_i, p_n) \end{cases}, \quad (\text{A2})$$

where p_n are the nominal geometric parameters (i.e., nominal link lengths). Since the fixed location of the encoder is specified by the joint coordinates θ_0 , the coordinates of the encoder location can also be computed through the direct kinematics:

$$\begin{cases} x_0 = f_x(\theta_0, p_n) \\ y_0 = f_y(\theta_0, p_n) \end{cases}. \quad (\text{A3})$$

However, the joint coordinates are affected by errors: these errors influence both the coordinates of the i th calibration point and the encoder coordinates. To evaluate the contribution of the errors on θ_1 and θ_2 on the i th calibration point and encoder coordinates, Eqs. A2 and A3 can be expressed as a Taylor expansions truncated at the first order:

$$\begin{cases} x_i = f_x(\theta_i, p_n) + \frac{\partial f_x(\theta, p)}{\partial \theta_1} \Big|_{\substack{\theta=\theta_i \\ p=p_n}} \delta\theta_1 + \frac{\partial f_x(\theta, p)}{\partial \theta_2} \Big|_{\substack{\theta=\theta_i \\ p=p_n}} \delta\theta_2 \\ y_i = f_y(\theta_i, p_n) + \frac{\partial f_y(\theta, p)}{\partial \theta_1} \Big|_{\substack{\theta=\theta_i \\ p=p_n}} \delta\theta_1 + \frac{\partial f_y(\theta, p)}{\partial \theta_2} \Big|_{\substack{\theta=\theta_i \\ p=p_n}} \delta\theta_2 \end{cases}. \quad (\text{A4})$$

$$\begin{cases} x_0 = f_x(\theta_0, p_n) + \frac{\partial f_x(\theta, p)}{\partial \theta_1} \Big|_{\substack{\theta=\theta_0 \\ p=p_n}} \delta\theta_1 + \frac{\partial f_x(\theta, p)}{\partial \theta_2} \Big|_{\substack{\theta=\theta_0 \\ p=p_n}} \delta\theta_2 \\ y_0 = f_y(\theta_0, p_n) + \frac{\partial f_y(\theta, p)}{\partial \theta_1} \Big|_{\substack{\theta=\theta_0 \\ p=p_n}} \delta\theta_1 + \frac{\partial f_y(\theta, p)}{\partial \theta_2} \Big|_{\substack{\theta=\theta_0 \\ p=p_n}} \delta\theta_2 \end{cases}. \quad (\text{A5})$$

To simplify the notation, let us call $(x_{i,th}, y_{i,th}) = (f_x(\theta_i, p_n), f_y(\theta_i, p_n))$ and $(x_{0,th}, y_{0,th}) = (f_x(\theta_0, p_n), f_y(\theta_0, p_n))$ the theoretical coordinates of the two points (i.e., evaluated with nominal values of the parameters). Let us also define $\Delta x_{i-0,th} = x_{i,th} - x_{0,th}$, $\Delta y_{i-0,th} = y_{i,th} - y_{0,th}$ and the three following matrices:

$$\Phi_i = \begin{bmatrix} \partial f_x / \partial e \\ \partial f_y / \partial e \end{bmatrix}_{\substack{\theta=\theta_i \\ p=p_n}} = \begin{bmatrix} \frac{\partial f_x(\theta, p)}{\partial \theta_1} & \frac{\partial f_x(\theta, p)}{\partial \theta_2} \\ \frac{\partial f_y(\theta, p)}{\partial \theta_1} & \frac{\partial f_y(\theta, p)}{\partial \theta_2} \end{bmatrix}_{\substack{\theta=\theta_i \\ p=p_n}}$$

$$= \begin{bmatrix} \Phi_{i11} & \Phi_{i12} \\ \Phi_{i21} & \Phi_{i22} \end{bmatrix} \tag{A6}$$

$$\Phi_0 = \left[\frac{\partial f_x / \partial \mathbf{e}}{\partial f_y / \partial \mathbf{e}} \right]_{\substack{\theta = \theta_0 \\ p = p_n}} = \left[\frac{\frac{\partial f_x(\theta, \mathbf{p})}{\partial \theta_1}}{\frac{\partial f_y(\theta, \mathbf{p})}{\partial \theta_1}} \quad \frac{\frac{\partial f_x(\theta, \mathbf{p})}{\partial \theta_2}}{\frac{\partial f_y(\theta, \mathbf{p})}{\partial \theta_2}} \right]_{\substack{\theta = \theta_0 \\ p = p_n}} \tag{A7}$$

$$\Phi_{i-0} = \begin{bmatrix} \Phi_{011} & \Phi_{012} \\ \Phi_{021} & \Phi_{022} \end{bmatrix} \tag{A8}$$

With these definitions, Eqs. A4 and A5 can be rewritten as follows:

$$\begin{cases} x_i = x_{i,th} + \Phi_{i11} \delta\theta_1 + \Phi_{i12} \delta\theta_2 \\ y_i = y_{i,th} + \Phi_{i21} \delta\theta_1 + \Phi_{i22} \delta\theta_2 \end{cases} \tag{A9}$$

$$\begin{cases} x_0 = x_{0,th} + \Phi_{011} \delta\theta_1 + \Phi_{012} \delta\theta_2 \\ y_0 = y_{0,th} + \Phi_{021} \delta\theta_1 + \Phi_{022} \delta\theta_2 \end{cases} \tag{A10}$$

The two quantities Δx_{i-0} and Δy_{i-0} in Eq. A1 can be expressed as:

$$\begin{cases} \Delta x_{i-0} = \Delta x_{i-0,th} + (\Phi_{i11} - \Phi_{011}) \delta\theta_1 + (\Phi_{i12} - \Phi_{012}) \delta\theta_2 \\ \Delta y_{i-0} = \Delta y_{i-0,th} + (\Phi_{i21} - \Phi_{021}) \delta\theta_1 + (\Phi_{i22} - \Phi_{022}) \delta\theta_2 \end{cases} \tag{A11}$$

The partial Jacobians that appear subtracted from each other can be rewritten as a single matrix. The previous expression can be rewritten as:

$$\begin{cases} \Delta x_{i-0} = \Delta x_{i-0,th} + \Phi_{i-011} \delta\theta_1 + \Phi_{i-012} \delta\theta_2 \\ \Delta y_{i-0} = \Delta y_{i-0,th} + \Phi_{i-021} \delta\theta_1 + \Phi_{i-022} \delta\theta_2 \end{cases} \tag{A12}$$

Those two quantities appear squared when computing the predicted distance d_i between the robot’s end-effector and the encoder. Neglecting the infinitesimal of higher order, we obtain

$$\begin{cases} \Delta x_{i-0}^2 = \Delta x_{i-0,th}^2 + 2\Delta x_{i-0,th}(\Phi_{i-011} \delta\theta_1 + \Phi_{i-012} \delta\theta_2) \\ \Delta y_{i-0}^2 = \Delta y_{i-0,th}^2 + 2\Delta y_{i-0,th}(\Phi_{i-021} \delta\theta_1 + \Phi_{i-022} \delta\theta_2) \end{cases} \tag{A13}$$

In Eq. A1, Δx_{i-0}^2 and Δy_{i-0}^2 are then summed to obtain d_i , which can be rewritten as

$$d_i = \sqrt{\Delta x_{i-0,th}^2 + \Delta y_{i-0,th}^2 + \delta d_{i,\delta\theta_1} + \delta d_{i,\delta\theta_2}} \tag{A14}$$

where $\delta d_{i,\delta\theta_1}$ and $\delta d_{i,\delta\theta_2}$ are the errors in the predicted distance due to the discrepancy between the nominal and actual values of θ_1 and θ_2 . These two quantities can be computed as

$$\delta d_{i,\delta\theta_1} = 2(\Delta x_{i-0,th} \Phi_{i-011} + \Delta y_{i-0,th} \Phi_{i-021}) \delta\theta_1 \tag{A15}$$

$$\delta d_{i,\delta\theta_2} = 2(\Delta x_{i-0,th} \Phi_{i-012} + \Delta y_{i-0,th} \Phi_{i-022}) \delta\theta_2. \tag{A16}$$

By observing Eqs. A15 and A16, it becomes clear that the contribution of the error parameters on the predicted distance may also become null. For instance, if the term $(\Delta x_{i-0,th} \Phi_{i-011} + \Delta y_{i-0,th} \Phi_{i-021})$ in Eq. A15 is null, an error on θ_1 does not influence the distance d_i : in this case, at the i th calibration point, the quantity $d_i - \tilde{d}_i$ depends only on the error on θ_2 . Equations A15 and A16 can be rewritten in matrix form:

$$\delta d_{i,\delta\theta_1} = 2[\Delta x_{i-0,th} \ \Delta y_{i-0,th}] \Phi_{i-01} \delta\theta_1 \tag{A17}$$

$$\delta d_{i,\delta\theta_2} = 2[\Delta x_{i-0,th} \ \Delta y_{i-0,th}] \Phi_{i-02} \delta\theta_2, \tag{A18}$$

where Φ_{i-01} and Φ_{i-02} are the first and second column of matrix Φ_{i-0} . In both cases, the first row is multiplied by $\Delta x_{i-0,th}$ and the second row is multiplied by $\Delta y_{i-0,th}$. Let us define the vector

$$\mathbf{v}_i = [\Delta x_{i-0,th} \ \Delta y_{i-0,th}], \tag{A19}$$

and the corresponding unit vector $\hat{\mathbf{v}}_i$, representing the theoretical direction of the wire

$$\hat{\mathbf{v}}_i = \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}. \tag{A20}$$

Equations A17 and A18 can be rewritten as

$$\delta d_{i,\delta\theta_1} = 2\|\mathbf{v}_i\| \langle \Phi_{i-01}, \hat{\mathbf{v}}_i \rangle \delta\theta_1 = 2\|\mathbf{v}_i\| \Psi_{i1} \delta\theta_1 \tag{A21}$$

$$\delta d_{i,\delta\theta_2} = 2\|\mathbf{v}_i\| \langle \Phi_{i-02}, \hat{\mathbf{v}}_i \rangle \delta\theta_2 = 2\|\mathbf{v}_i\| \Psi_{i2} \delta\theta_2. \tag{A22}$$

The two scalars Ψ_{i1} and Ψ_{i2} , when compared, quantify the reciprocal influence of $\delta\theta_1$ and $\delta\theta_2$ on the distance d_i . They can also be grouped into a single matrix:

$$\Psi_i = [\Psi_{i1} \ \Psi_{i2}], \tag{A23}$$

which is the same matrix as in Eq. 15, obtained in the case of a 2 degrees-of-freedom arm where the only unknown error parameters are the joint offsets. This demonstrates the validity of our approach, which can be extended to manipulators with higher degrees of freedom and with more unknown parameters.

Acknowledgements The authors would like to show their gratitude to Mr. Alessandro D’Amico who provided insight, materials, and expertise that greatly assisted the research.

Funding Open access funding provided by Università degli Studi di Padova within the CRUI-CARE Agreement.

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Zhu D, Feng X, Xu X, Yang Z, Li W, Yan S, Ding H (2020) Robotic grinding of complex components: a step towards efficient and intelligent machining - challenges, solutions, and applications. *Robotics and Computer-Integrated Manufacturing* 65. <https://doi.org/10.1016/j.rcim.2019.101908>
- Li J, Li B, Shen NY, Qian H, Guo ZM (2017) Effect of the cutter path and the workpiece clamping position on the stability of the robotic milling system. *Int J Adv Manuf Technol* 89(9–12):2919–2933. <https://doi.org/10.1007/s00170-016-9759-x>
- Zeng Y, Tian W, Li D, He X, Liao W (2017) An error-similarity-based robot positional accuracy improvement method for a robotic drilling and riveting system. *Int J Adv Manuf Technol* 88(9–12):2745–2755. <https://doi.org/10.1007/s00170-016-8975-8>
- Schneider U, Drust M, Ansaloni M, Lehmann C, Pellicciari M, Leali F, Gunnink JW, Verl A (2016) Improving robotic machining accuracy through experimental error investigation and modular compensation. *Int J Adv Manuf Technol* 85(1–4):3–15. <https://doi.org/10.1007/s00170-014-6021-2>
- Cordes M, Hintze W (2017) Offline simulation of path deviation due to joint compliance and hysteresis for robot machining. *Int J Adv Manuf Technol* 90(1–4):1075–1083. <https://doi.org/10.1007/s00170-016-9461-z>
- Yin S, Guo Y, Ren Y, Zhu J, Yang S, Ye S (2014) Real-time thermal error compensation method for robotic visual inspection system. *Int J Adv Manuf Technol* 75(5–8):933–946. <https://doi.org/10.1007/s00170-014-6196-6>
- Schröder K, Albright SL, Grethlein M (1997) Complete, minimal and model-continuous kinematic models for robot calibration. *Robot Comput -Integr Manuf* 13(1):73–85. [https://doi.org/10.1016/S0736-5845\(96\)00025-7](https://doi.org/10.1016/S0736-5845(96)00025-7)
- Denavit J, Hartenberg RS (1955) A kinematic notation for lower-pair mechanisms based on matrices. *J Appl Mech Trans ASME* 22(2):215–221. <https://doi.org/10.1115/1.4011045>
- Nubiola A, Bonev IA (2013) Absolute calibration of an ABB IRB 1600 robot using a laser tracker. *Robot Comput -Integr Manuf* 29(1):236–245. <https://doi.org/10.1016/j.rcim.2012.06.004>
- Nubiola A, Bonev IA (2014) Absolute robot calibration with a single telescoping ballbar. *Precision Eng* 38(3):472–480. <https://doi.org/10.1016/j.precisioneng.2014.01.001>
- Hayati SA (1983) Robot arm geometric link parameter estimation. In: *The 22nd IEEE conference on decision and control*, pp 1477–1483. <https://doi.org/10.1109/cdc.1983.269783>
- Stone HW, Sanderson AC (1988) Statistical performance evaluation of the S-model arc signature identification technique. In: *Proceedings. 1988 IEEE international conference on robotics and automation*, pp 939–9462. <https://doi.org/10.1109/ROBOT.1988.12180>
- Zhuang H, Roth ZS, Hamano F (1992) A complete and parametrically continuous kinematic model for robot manipulators. *IEEE Trans Robot Autom* 8(4):451–463. <https://doi.org/10.1109/70.149944>
- Zhuang H, Wang LK, Roth ZS (1993) Error-model-based robot calibration using a modified CPC model. *Robot Comput Integr Manuf* 10(4):287–299. [https://doi.org/10.1016/0736-5845\(93\)90042-1](https://doi.org/10.1016/0736-5845(93)90042-1)
- He R, Zhao Y, Yang S, Yang S (2010) Kinematic-parameter identification for serial-robot calibration based on POE formula. *IEEE Trans Robot* 26(3):411–423. <https://doi.org/10.1109/TRO.2010.2047529>
- Chen I-M, Yang G, Tan CT, Yeo SH (2001) Local POE model for robot kinematic calibration. *Mech Mach Theory* 36(11–12):1215–1239. [https://doi.org/10.1016/S0094-114X\(01\)00048-9](https://doi.org/10.1016/S0094-114X(01)00048-9)
- 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. *Robot Comput -Integr Manuf* 35:151–168. <https://doi.org/10.1016/j.rcim.2015.03.007>
- Santolaria J, Conte J, Ginés M (2013) Laser tracker-based kinematic parameter calibration of industrial robots by improved CPA method and active retroreflector. *Int J Adv Manuf Technol* 66(9–12):2087–2106. <https://doi.org/10.1007/s00170-012-4484-6>
- Yu C, Chen X, Xi J (2018) Determination of optimal measurement configurations for self-calibrating a robotic visual inspection system with multiple point constraints. *Int J Adv Manuf Technol* 96(9–12):3365–3375. <https://doi.org/10.1007/s00170-018-1739-x>
- Motta JMST, De Carvalho GC, McMaster RS (2001) Robot calibration using a 3D vision-based measurement system with a single camera. *Robot Comput -Integr Manuf* 17(6):487–497. [https://doi.org/10.1016/S0736-5845\(01\)00024-2](https://doi.org/10.1016/S0736-5845(01)00024-2)
- Švaco M, Šekoranja B, Šuligoj F, Jerbić B (2014) Calibration of an industrial robot using a stereo vision system. *Proc Eng* 69:459–463. <https://doi.org/10.1016/j.proeng.2014.03.012>
- Yang P, Guo Z, Kong Y (2020) Plane kinematic calibration method for industrial robot based on dynamic measurement of double ball bar. *Precision Eng* 62:265–272. <https://doi.org/10.1016/j.precisioneng.2019.12.010>
- Whitney DE, Lozinski CA, Rourke JM (1986) Industrial robot forward calibration method and results. *J Dyn Syst Meas Control Trans ASME* 108(1):1–8. <https://doi.org/10.1115/1.3143737>
- Judd RP, Knasinski AB (1990) A technique to calibrate industrial robots with experimental verification. *IEEE Trans Robot Autom* 6(1):20–30. <https://doi.org/10.1109/70.88114>
- Driels MR, Pathre US (1994) Robot calibration using an automatic theodolite. *Int J Adv Manuf Technol* 9(2):114–125. <https://doi.org/10.1007/BF01750418>
- Nubiola A, Slamani M, Joubair A, Bonev IA (2014) Comparison of two calibration methods for a small industrial robot based on an optical CMM and a laser tracker. *Robotica* 32(3):447–466. <https://doi.org/10.1017/S0263574713000714>
- Driels MR, Swayze W, Potter S (1993) Full-pose calibration of a robot manipulator using a coordinate-measuring machine. *Int J Adv Manuf Technol* 8(1):34–41. <https://doi.org/10.1007/BF01756635>
- Luo G, Zou L, Wang Z, Lv C, Ou J, Huang Y (2021) A novel kinematic parameters calibration method for industrial robot based on Levenberg-Marquardt and differential evolution hybrid algorithm. *Robot Comput -Integr Manuf* 71:102165. <https://doi.org/10.1016/j.rcim.2021.102165>
- Nguyen H-N, Zhou J, Kang H-J (2015) A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network. *Neurocomputing* 151:996–1005. <https://doi.org/10.1016/j.neucom.2014.03.085>
- Cao HQ, Nguyen HX, Nguyen TT, Nguyen VQ, Jeon JW (2021) Robot calibration method based on extended Kalman filter-dual

- quantum behaved particle swarm optimization and adaptive neuro-fuzzy inference system. *IEEE Access* 9:132558–132568. <https://doi.org/10.1109/ACCESS.2021.3115120>
31. Messay-Kebede T, Sutton G, Djaneye-Boundjou O (2018) Geometry based self kinematic calibration method for industrial robots. In: 2018 IEEE international conference on robotics and automation (ICRA), pp 4921–4926. <https://doi.org/10.1109/ICRA.2018.8460764>
 32. Fang L, Dang P (2016) Kinematic calibration method of robots based on quantum-behaved particle swarm optimization. *J Mech Eng* 52(7):23–30. <https://doi.org/10.3901/JME.2016.07.023>
 33. Klimchik A, Pashkevich A, Chablat D, Hovland G (2013) Compliance error compensation technique for parallel robots composed of non-perfect serial chains. *Robot Comput -Integr Manuf* 29(2):385–393. <https://doi.org/10.1016/j.rcim.2012.09.008>
 34. Dépincé P, Hascoët J-Y (2006) Active integration of tool deflection effects in end milling. part 2. compensation of tool deflection. *Int J Mach Tools Manuf* 46(9):945–956. <https://doi.org/10.1016/j.ijmachtools.2005.08.014>
 35. Thomas F, Ottaviano E, Ros L, Ceccarelli M (2005) Performance analysis of a 3–2–1 pose estimation device. *IEEE Trans Robot* 21(3):288–297. <https://doi.org/10.1109/TRO.2004.840894>
 36. Legnani G, Tiboni M (2014) Optimal design and application of a low-cost wire-sensor system for the kinematic calibration of industrial manipulators. *Mech Mach Theory* 73:25–48. <https://doi.org/10.1016/j.mechmachtheory.2013.09.005>
 37. Jeong JW, Kim SH, Kwak YK, Smith CC (1998) Development of a parallel wire mechanism for measuring position and orientation of a robot end-effector. *Mechatronics* 8(8):845–861. [https://doi.org/10.1016/S0957-4158\(98\)00034-8](https://doi.org/10.1016/S0957-4158(98)00034-8)
 38. Muralidhara Jain NR, Sushith K (2022) Experimental investigations on robotic linear stage traverse error using draw wire sensor. *Mater Today: Pro* 52:1521–1529. <https://doi.org/10.1016/j.matpr.2021.11.229>
 39. Gan Y, Duan J, Dai X (2019) A calibration method of robot kinematic parameters by drawstring displacement sensor. *International Journal of Advanced Robotic Systems* 16(5). <https://doi.org/10.1177/1729881419883072>
 40. Messay T, Ordóñez R, Marcil E (2016) Computationally efficient and robust kinematic calibration methodologies and their application to industrial robots. *Robot Comput -Integr Manuf* 37:33–48. <https://doi.org/10.1016/j.rcim.2015.06.003>
 41. Gao G, Li Y, Liu F, Han S (2021) Kinematic calibration of industrial robots based on distance information using a hybrid identification method. *Complexity* 2021. <https://doi.org/10.1155/2021/8874226>
 42. Yun H, Jeon H, Yang S, Jun MBG (2022) Calibration of industrial robots with spherical joint using single wire encoder. *Manuf Lett* 33:46–50. <https://doi.org/10.1016/j.mfglet.2022.07.120>
 43. Driels MR, Swayze WE (1994) Automated partial pose measurement system for manipulator calibration experiments. *IEEE Trans Robot Autom* 10(4):430–440. <https://doi.org/10.1109/70.313094>
 44. Li Z, Li S, Wu H (2022) A new variable step-size Levenberg-Marquardt algorithm for industrial robot calibration. In: 2022 IEEE international conference on networking, sensing and control (ICNSC), pp 1–6. <https://doi.org/10.1109/ICNSC55942.2022.10004134>
 45. Li Z, Li S, Luo X (2023) Efficient industrial robot calibration via a novel unscented Kalman filter-incorporated variable step-size Levenberg-Marquardt algorithm. *IEEE Trans Instrum Meas* 72:1–12. <https://doi.org/10.1109/TIM.2023.3265744>
 46. Roth Z, Mooring B, Ravani B (1987) An overview of robot calibration. *IEEE J Robot Autom* 3(5):377–385. <https://doi.org/10.1109/JRA.1987.1087124>
 47. Driels MR, Pathre US (1990) Significance of observation strategy on the design of robot calibration experiments. *J Robot Syst* 7(2):197–223. <https://doi.org/10.1002/rob.4620070206>
 48. Borm J-H, Menq C-H (1991) Determination of optimal measurement configurations for robot calibration based on observability measure. *Int J Rob Res* 10(1):51–63. <https://doi.org/10.1177/027836499101000106>
 49. Sun Y, Hollerbach JM (2008) Observability index selection for robot calibration. In: 2008 IEEE international conference on robotics and automation, pp 831–836. <https://doi.org/10.1109/ROBOT.2008.4543308>
 50. Craig JJ (1989) *Introduction to robotics: mechanics and control*, 2nd edn. Addison-Wesley Longman Publishing Co. Inc, USA

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.