

Kinematic calibration of a lightweight manipulator for medical applications

Krzysztof ARENT^{1*}, Adam KURNICKI², Piotr PORTASIAK³, and Bartłomiej STAŃCZYK⁴

¹ Department of Cybernetics and Robotics, Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

² Department of Automation and Metrology, Electrical Engineering and Computer Science Faculty, Lublin University of Technology, ul. Nadbystrzycka 38A, 20-618 Lublin, Poland

³ Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

⁴ Accrea Medical Robotics, ul. Hiacyntowa 20, 20-143 Lublin, Poland

Abstract. The paper presents a kinematics calibration procedure for a lightweight manipulator designed for medical applications. They comprise improving the dexterity of a dysfunctional arm of a handicapped patient in an electric wheelchair as well as supporting biopsies and surgeries. Consequently, there are several manipulator distinguishing features of the manipulator design that are relevant to kinematics calibration. In particular, these are: a small area in the workspace within which the end-effector operates, affordability for non-commercial users, a delicate, dexterous design. In this context we propose a specialized procedure that features a low cost calibration tool enabling the end-effector to reach the correct positions for data acquisition. The key parameters of the calibration tool were obtained by applying two techniques of numerical analysis, workspace clustering and arbitrary choice, and subsequent experimental verification. The procedure exploits classical results concerning the kinematics calibration and is empirically verified by comprehensive simulation and experimental studies.

Keywords: manipulator; kinematics; calibration; medical robots.

1. INTRODUCTION

The intensive development of collaborative robots and their non-industrial applications can be observed over recent years. The ability of a manipulator to interact directly with humans opens new area of its applications, in particular in medicine and assistive technology. What is more, the manufacturing of robotic technology has become affordable not only for large companies but also for smaller ones. This facilitates the spread of robotic technology in niche but socially important areas of life. In these circumstances, a few new aspects have become important, in particular: small/medium enterprises cannot use expensive measurement equipment; the device has to be affordable for the end-user; in the case of medical or assistive applications robots have to meet complex user requirements and fulfill restrictive safety regulations.

These conditions are apparently contradictory and require individual approach to each robot. In other words, the established evaluations criteria for applied algorithms and techniques need to be revised and they should take into account the context.

To be more specific, we refer to ACCREA company¹ and its ARIA segment of manipulators (ARIA stands for Assistive

robotic arm). What can be noticed concerning these manipulators is that due to the usage their effectors operate inside small regions of their workspaces; due to imperfections in the manufacturing process their kinematics must be calibrated; due to the necessity of the cost reduction the calibration procedure must not use expensive measuring instruments. This puts the problem of kinematics calibration in a slightly different light.

The kinematics calibration problem has attracted considerable attention since the 1980s. A good summary of results obtained to date is included in [1]. It follows, that a typical kinematics calibration procedure decomposes into four tasks: error modeling, measurement, identification, validation. Usually, a model error is the difference between the actual and the estimated end-effector position and orientation. The actual ones are functions of Denavit-Hartenberg or Hayati's parameters. Measurement is carried out for manipulator configurations that are optimal with respect to an observation index for the identification Jacobian. Typical identification procedures of kinematic parameters rely on least squares estimation. Validation usually comes down to parameters correction and analysis of the associated estimation errors. There are also other approaches to the calibration problem. For example, the article [2] is concerned with the calibration problem of kinematics subject to both geometric and non-geometric variations. The author proposed calibration by diffeomorphisms, and then he formulated its solvability conditions.

The successful application of kinematic calibration theory in practice depends, in particular, on the selection of robot poses

*e-mail: krzysztof.arent@pwr.edu.pl

Manuscript submitted 2024-04-26, revised 2024-07-09, initially accepted for publication 2024-07-30, published in November 2024.

¹<http://accrea.com/>

and the accuracy of measurements of the corresponding end-effector position and orientation. The poses should be informative and insensitive to noises. This issue has been addressed for a long time and is still active, see e.g. [3–6]. The end-effector position and orientation measurements are made using complex commercial photogrammetric [7–9], laser based [10, 11], probing [12] and portable [13] CMM systems (coordinate measuring machine). They provide high accuracy of measurement but they are expensive. For these reasons, measurement is an important issue addressed in publications on the calibration of the kinematics of real manipulators.

The authors in [6] carry out the open-loop geometric calibration of a redundant *KUKA LWR4+*. The postures for the identification process were selected with the help of two observability indices (the inverse condition number and one proposed by Born-Meng) that in turn were applied in three optimization procedures. The associated 6D end-effector poses were measured using Nikon K610 CMM. The Denavit-Hartenberg (DH) parameters were identified with the acquired data by solving numerically a nonlinear optimization problem. The resulted DH parameters significantly improve the accuracy of the 6D end-effector pose but none of applied algorithms proved to be noticeably better than the others.

In [12] the authors improve the accuracy of FANUC LR Mate 200iC inside a pre-selected small area inside the workspace by applying a new, low-cost technique for measuring the 6D end-effector pose. This technique relies on a Renishaw precision touch probe mounted on the flange of a manipulator and a special triangular plate with three datum 2-in spheres 300 mm apart. The accuracy has been improved ca. 10 times.

While much attention was initially paid to the kinematics calibration of serial, open chain rigid-link manipulators, the research work in this field now covers a much wider class of them. It includes hybrid robots [14], robots with continuum mechanisms [15], modular robots [16], parallel manipulators [17], space manipulators [18]. The general scheme of a calibration process is similar in all these cases. The main differences are in the data acquisition and the model error design stages that are specific for particular robots kinematic structures. Regardless of the manipulator structure, the problem of a simple, effective, low-cost specialized method of data acquisition for calibration arises when a particular type of manipulator reaches the product stage. This issue is the central focus of this article. A similar problem is discussed in [19].

This work refers to the ideas in [6, 20] and [12] just outlined. It proposes a kinematic calibration procedure for ARIA manipulators. Their postures are enforced by a custom low-cost calibration tool at the measurement stage. The key parameters of the calibration tool are obtained through systematic and comprehensive simulation studies. Workspace clustering and arbitrary choice techniques have been specifically designed and applied for this purpose. The tool is simple, in particular it allows the determination of ten postures for which the end-effector stays within a specified small region of the workspace. It is justified by the envisioned application of ARIA manipulators. Three identification algorithms are launched in parallel to determine kinematic parameters at the identification stage. This redundancy

improves the reliability of the resulting parameters values. The validation process is based on simulation and empirical studies and shows that the 6D end-effector pose accuracy is significantly improved, especially in the region around the calibration tool. The entire calibration procedure has been implemented in *Matlab* and *C++*, allowing the calibration process to be automated in practice and to apply it under small and medium-sized enterprises (SMEs) settings, where financial and technological constraints are greater than those of large companies.

This paper is organized as follows. The design, properties and expected use cases of an ARIA manipulator are discussed in Section 2. The measurement and identification processes complemented by the ideas underlying the calibration tool are presented in Section 3. Implementation of the calibration procedure, simulation analysis of the ARIA kinematics calibration procedure and simulation-based comparative analysis of gauge prototypes are contained in Section 4. The experimental setup and the results of experimental studies are discussed in Section 5. The conclusions are drawn in Section 6.

2. ARIA MANIPULATOR

The assistive robotic arm (ARIA) was designed by Accrea Engineering company as a lightweight, modular and relatively inexpensive manipulator for human-robot interaction in medical applications [21].

2.1. Manipulator construction / kinematics

The ARIA manipulator is a human-sized robotic arm with six revolute joints and an end-effector, specific to the intended use, see Fig. 1.

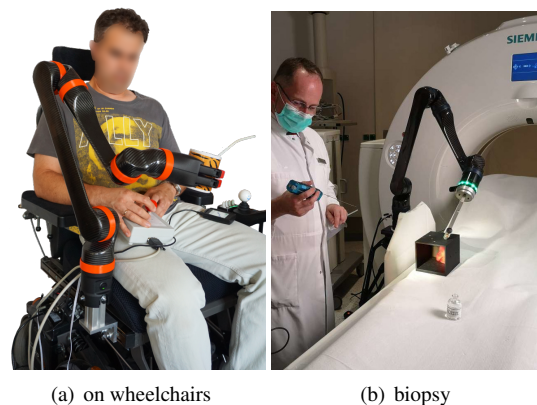


Fig. 1. ARIA manipulator and its areas of use

Each joint is constructed with a Maxon BLDC motor, a harmonic gearbox and multiple sensors that provide signals for axis control, such as position, speed and torque control. The structural components of the arm are made of rigid materials. The manipulator links are manufactured from carbon fibre and the base from aluminum.

The Denavit-Hartenberg convention is applied to define the relative position and orientation of two consecutive links of

Kinematic calibration of a lightweight manipulator

ARIA [1]. The kinematic chain of an ARIA manipulator is shown in Fig. 2. The associated DH parameters are collected in Table 1. The rotation θ_i about Z_{i-1} axis is represented by the sum

$$\theta_i := q_i + \theta_i^0, \quad (1)$$

where q_i is a joint variable and θ_i^0 is an angular offset. The parameters d_i and a_i denote offsets along Z_{i-1} and X_i axes, respectively. In turn, α_i stands for an axis misalignment between Z_{i-1} and Z_i axes. The joints variables $q_1 \div q_6$ take values in radians from the set

$$\mathcal{Q} = \{(q_1, \dots, q_6) \in \mathbb{R}^6 \mid -2\pi \leq q_1 \leq 2\pi, \\ -0.97 \leq q_2 \leq 3.615, 2\pi \leq q_3 \leq 1.69, \\ -2\pi \leq q_4 \leq 2\pi, -\pi \leq q_5 \leq \pi, -2\pi \leq q_6 \leq 2\pi\}. \quad (2)$$

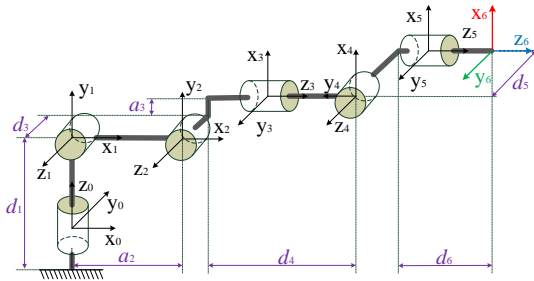


Fig. 2. Kinematic chain of the ARIA manipulator

Table 1

Nominal Denavit-Hartenberg parameters values of an ARIA manipulator

i	θ_i^0 [rad]	d_i [m]	a_i [m]	α_i [rad]
1	0	0.205	0.0	$\pi/2$
2	0	0.0	0.41	0.0
3	$\pi/2$	-0.0048	0.03	$\pi/2$
4	0	0.336	0.0	$-\pi/2$
5	0	-0.049	0.0	$\pi/2$
6	0	0.202	0.0	0.0

Denavit-Hartenberg convention modified by Hayati [22] is used in the subsequent part of this article. The nominal values of the Hayati parameters β_i are equal to zero for all i .

2.2. Application to increase the manual dexterity of disabled people

ARIA manipulators have two versions: BATEO and BONITO. BATEO is a lightweight arm designed for integration onto powered wheelchairs, as shown in Fig. 1a. It is used by persons with severe physical disabilities, especially by people who have no upper limb or its usage is strongly limited. It helps them in performing activities of daily living. Here kinematic calibration is not critical although advisable, in particular when a disabled

person is required to use several wheelchairs with manipulators (e.g. in different locations). Then the manual control skills acquired on one device are sufficient to use another effectively and comfortably. The BONITO version of ARIA manipulator has the same kinematics as BATEO but these manipulators differ in their manufacturing technology. This system is dedicated to biopsy of neoplastic lesions in abdominal and thoracic tissues (e.g. liver, lungs).

2.3. Biopsy application

BONITO version of ARIA manipulator is a cobot designed for integration with a BONITO system, developed within the frame of the project: “Configurable arm for stabilization of the biopsy needle insertion trajectory”, supported by the Polish National Centre for Research and Development (see Fig. 1b). Biopsy procedure, performed in the traditional way, relies on manual insertion of the needle by the doctor. Needle insertion is time-consuming and most of the time the target tissues cannot be reached due to their small sizes. This problem can be considerably reduced with the aid of a manipulator. The idea of a robotic approach is presented in Fig. 3. Here, the role of the doctor is limited to two main tasks:

- 1) indicating the skin puncture point and cancer tissue sampling point on a volumetric model of the patient’s body, obtained on the basis of DICOM image processing from a previously performed computerized tomography (CT) scan;
- 2) inserting the needle into a given depth along the trajectory determined by the end-effector device of the guiding arm.

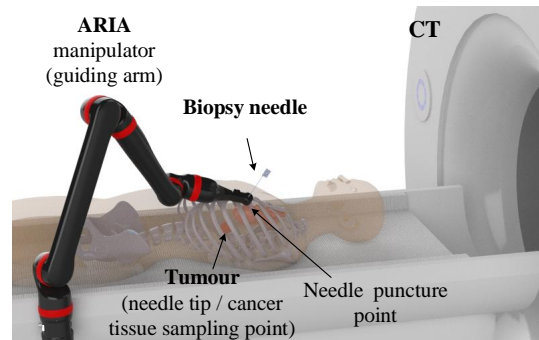


Fig. 3. Biopsy with ARIA guiding arm

A manipulator provides greater stiffness and precision to needle handling compared to human hands.

It can be noticed, by comparing Fig. 1 and Fig. 3, that the end-effector for biopsy consists of an extension arm and a needle attached to its end. The axis of the extension arm coincides with Z_6 and the axis of the needle is perpendicular to Z_6 and parallel to X_6 . Thus the end-effector coordinate frame, $X_{6e}Y_{6e}Z_{6e}$, originates at the point of intersection of the extension arm axis and the needle axis. The axes X_6 and X_{6e} are mutually parallel and have the same directions. The same applies to the pairs: Y_6, Y_{6e} and Z_6, Z_{6e} . The needle is inserted into the patient’s body along the X_{6e} axis, in the direction opposite to the X_{6e} axis direction.

2.4. Accuracy study

The guiding manipulator for biopsy must demonstrate high accuracy and resolution. The inaccurate needle positioning can destroy cells and organs. What is more, this may imply the need to repeat the tests.

If the doctor's/operator's error is excluded then the needle positioning error includes:

- (i) errors in calibration of the CT image coordinate frame with the base coordinate frame of the table and thus the guiding arm,
- (ii) manipulator positioning errors.

The required needle placement accuracy during biopsy is determined by the clinically significant size of cancer foci. There is no general agreement on this value. The doctors from the Lublin Medical University in Poland, in a survey, proposed 0.5 mL tumour volume as the limit for the significant cancer foci. A 0.5 mL spherical shaped tumour has radius r_t of almost 5 mm which means the needle placement error should be smaller than 5 mm. This accuracy is affected by two factors that can be clarified with the help of Fig. 4.

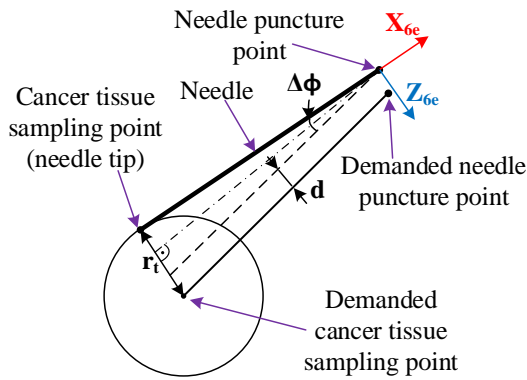


Fig. 4. Positioning error of biopsy needle

The *needle puncture point* in Fig. 4 is the location of the origin of $X_{6e}Y_{6e}Z_{6e}$ coordinate frame for some inaccurate DH parameters. The *demedanded needle puncture point* in Fig. 4 is the location of the origin of $X_{6e}Y_{6e}Z_{6e}$ coordinate frame for actual but unknown DH parameters. Let the position of *demedanded needle puncture point* in $X_{6e}Y_{6e}Z_{6e}$ be denoted by $\vec{d} = [d_x, d_y, d_z]$ and $d := \|\vec{d}\|_2$. \vec{d} can be interpreted as the arm's end-effector positioning error at the skin puncture point. The variable $\Delta\phi$ denotes a needle orientation error in terms of RPY angles. Let $\Delta\phi = [\Delta\phi_X, \Delta\phi_Y, \Delta\phi_Z]$. By construction $\Delta\phi_X = 0$. It is easy to show that

$$\Delta\phi_Y = \Delta\phi_Z = 2 \arcsin\left(\frac{r_t - d}{2\sqrt{2}l_{\max}}\right), \quad (3)$$

where: $l_{\max} = 0.2$ m is the maximum needle insertion length. Taking into account the worst-case scenario shown in Fig. 4 and assuming $d_x = d_y = d_z = 1$ mm, the orientation error $\Delta\phi_Y$ and $\Delta\phi_Z$ should not exceed 0.66° .

Since the manipulator links are made with use of low-cost technology, its kinematic parameters differ from the nominal values due to manufacturing inaccuracies. It follows from some empirical studies, that manufacturing errors can cause translational inaccuracies (axes displacements) of up to 5 mm and angular inaccuracies (axes twist) of up to 5° for each link.

2.5. Remarks

The ARIA robotic control system must be precise enough in order a physician could perform biopsy needle activities properly. This highlights the importance of kinematics calibration for the ARIA manipulator. The use of good values for the Denavit-Hartenberg parameters in the control system undoubtedly facilitates manual control of the manipulator in the task space by users with physical disabilities, especially when they need to operate several manipulators in different locations.

The intended applications of the ARIA manipulator imply that its end-effector will mainly operate in a small part of the workspace. For example, the end-effector will operate right above a human torso in the case of biopsy application, see Fig. 4. Or the end-effector will operate above a table aside or in front of a wheelchair to which the manipulator would be mounted. These areas are particularly important from the point of view of kinematics calibration.

The above analysis of the envisaged applications of ARIA justifies focusing attention on local calibration. Given the imperfections of the manipulator implementation, local calibration should lead to parameters values that provide better positioning precision of the end-effector than calibration based on measurements taken from the entire workspace.

3. CALIBRATION PROCEDURE

3.1. Kinematic model of the robot

The joint space of ARIA, \mathbb{Q} , has been defined in (2). Since ARIA is a non-redundant manipulator, there is exactly one the end-effector pose

$$y_e = \begin{bmatrix} p_e^T & \phi_e^T \end{bmatrix}^T \quad (4)$$

in the workspace for every $q \in \mathbb{Q}$ where p_e describes the end-effector position and ϕ_e its orientation [23]. The forward kinematics k can be written in the form

$$y_e = k(q). \quad (5)$$

With reference to Section 2, the modified Denavit-Hartenberg (mDH) convention is used to determine kinematics representation in terms of a homogeneous transformation matrix $T_e^b(q)$. It describes the position and orientation of the end-effector frame e with respect to the base coordinate frame b . This matrix can be obtained as

$$T_e^b(q) = T_0^b T_n^0(q) T_e^n, \quad (6)$$

where

$$T_0^n(q) = A_1^0(q_1) A_2^1(q_2) \cdots A_n^{n-1}(q_{n-1}). \quad (7)$$

Kinematic calibration of a lightweight manipulator

Due to the Denavit-Hartenberg convention modified by Hayati [22], the transformation of the link i is ([20]):

$$A_i^{i-1}(q_i) = \text{Rot}(z, q_i + \theta_i^0) \text{Trans}(z, d_i) \text{Trans}(x, a_i) \text{Rot}(x, \alpha_i) \text{Rot}(y, \beta_i). \quad (8)$$

Note that for a manipulator with revolute joints, the parameters d_i and β_i never occur together. If the two consecutive joint axes i and $i+1$ are parallel in a manipulator with revolute joints, the value of d_i is zero while if the two consecutive joint axes are perpendicular, the value of β_i is zero. Therefore the vector of mDH parameters corresponding to the link i ,

$$\Phi_i = \begin{bmatrix} \theta_i^0 & d_i & a_i & \alpha_i & \beta_i \end{bmatrix}^T, \quad (9)$$

in fact has four coordinates.

The vectors assembly $\Phi = \begin{bmatrix} \Phi_1^T & \dots & \Phi_n^T \end{bmatrix}^T$ forms an mDH parameters vector associated to the manipulator.

It should be noted that Φ does not include the end-effector parameters. During data acquisition, the end-effector is part of the measurement setup and hence T_e^n is known. The T_e^n matrix for the target end-effector (for the biopsy) should be determined separately.

3.2. Kinematic error

In fact, the exact value Φ^* of mDH parameters associated with a real manipulator is not known, but only its approximate $\hat{\Phi}$. In consequence there is a mismatch between the real pose y_e and its estimate $\hat{y}_e := k(q, \hat{\Phi})$. The mismatch can be represented by a kinematic error defined as follows:

$$\Delta y_e(\hat{\Phi}) := {}^b y_e - {}^b \hat{y}_e. \quad (10)$$

There is a linear relationship between $\Delta y_e(\hat{\Phi})$ and the estimation error $\Delta \Phi := \Phi^* - \hat{\Phi}$:

$$\Delta y_e(\hat{\Phi}) = J(\hat{\Phi}, q) \Delta \Phi, \quad (11)$$

where

$$J(\hat{\Phi}, q) = \frac{\partial k}{\partial \Phi}(\hat{\Phi}, q). \quad (12)$$

If ϕ_e in (5) is formed from Euler ZYX angles then the individual columns of J in (12) can be expressed as follows [20]:

$$\frac{\partial k}{\partial \theta_i^0} = \begin{bmatrix} {}^b z_{i-1} \times {}^b d_{i-1,e} \\ {}^b z_{i-1} \end{bmatrix}_{6 \times 1}, \quad (13a)$$

$$\frac{\partial k}{\partial d_i} = \begin{bmatrix} {}^b z_{i-1} \\ 0 \end{bmatrix}_{6 \times 1}, \quad (13b)$$

$$\frac{\partial k}{\partial a_i} = \begin{bmatrix} {}^b x_i \\ 0 \end{bmatrix}_{6 \times 1}, \quad (13c)$$

$$\frac{\partial k}{\partial \alpha_i} = \begin{bmatrix} {}^b x_i \times {}^b d_{i,e} \\ {}^b x_i \end{bmatrix}_{6 \times 1}, \quad (13d)$$

$$\frac{\partial k}{\partial \beta_i} = \begin{bmatrix} {}^b y_i \times {}^b d_{i,e} \\ {}^b y_i \end{bmatrix}_{6 \times 1}. \quad (13e)$$

${}^b x_i, {}^b y_i, {}^b z_i$ are versors of the i -th coordinate frame in the base coordinate frame and

$${}^b d_{i,e} = {}^b R_i^i p_e = ({}^b p_e - {}^b p_i), \quad (14)$$

where ${}^b p_i$ is a vector pointing the origin O_i of the i -th coordinate frame in the base coordinate frame and ${}^b R_i$ is a transformation matrix of i -th coordinate frame with respect to the base coordinate frame.

In turn, the kinematic error $\Delta y_e = \begin{bmatrix} \Delta {}^b p_e^T & \Delta {}^b \phi_e^T \end{bmatrix}^T$ (10) can be calculated for each of the components separately [1]. Let

${}^b T_e(q, \hat{\Phi}) = \begin{bmatrix} {}^b \hat{R}_e & {}^b \hat{p}_e \\ 0 & 1 \end{bmatrix}$ be a homogeneous transformation matrix associated to $k(\hat{\Phi}, q)$. Then

$$\Delta {}^b p_e = {}^b p_e - {}^b \hat{p}_e, \quad (15)$$

while $\Delta {}^b \phi_e$ is resulted from the following equation:

$$S(\Delta {}^b \phi_e) = ({}^b R_e - {}^b \hat{R}_e) ({}^b \hat{R}_e)^T. \quad (16)$$

S in (16) is a skew-symmetric matrix. Its elements specify $\Delta {}^b \phi_e$.

Once $\Delta y_e(\hat{\Phi})$ and $J(\hat{\Phi}, q)$ are known the equation (11) can be used by an optimization algorithm to determine $\hat{\Phi}$ [23]. The right-hand sides in expressions (13) can be calculated using the measurement of q and the current estimates vector $\hat{\Phi}$ while the calculations in (14), (16) additionally request the measurement of ${}^b y_e$.

3.3. Parameter identification

The performance of the identification algorithm that is built upon the kinematic error (10) and the relationship (11) strongly relies on the good conditioning of $J(\hat{\Phi}, q)$. To address this problem, extensive and carefully designed data sets of $({}^b y_e^l, q^l)$ are used to drive identification algorithms [23]. The superscript $l = 1, \dots, m$ labels the single measurement of a pair consisting of the end-effector pose and the associated joints angles for configuration in the data set. Here $m \gg n \cdot N_{mDH} / N_y$ where n is a size of q , N_{mDH} is a size of Φ and N_y is a size of ${}^b y_e$. Following [6], for an open-loop kinematic serial link manipulator the value of q^l can be obtained by maximizing an observability index $O(q)$ of the Jacobian (12) defined e.g. using the inverse condition number of J :

$$O(q) = \frac{\sigma_{\min}(J(\Phi_{\text{nom}}, q))}{\sigma_{\max}(J(\Phi_{\text{nom}}, q))}. \quad (17)$$

Φ_{nom} stands for the nominal value of Φ and $\sigma_{\min}, \sigma_{\max}$ are Jacobian minimal and maximum singular values, respectively. The higher value of the index (17), the more all mDH parameters

are equally identifiable. Other observability indices are gathered and discussed in [1]. Consequently

$$q^l = \underset{q}{\operatorname{argmax}} O(q), \text{ subject to} \quad (18)$$

$$q_{i,\text{init}}, q_i \in (q_{i,\text{min}}, q_{i,\text{max}}), \quad i = 1, \dots, n.$$

The value of ${}^b y_e^l$ associated to q^l is obtained from the measurement provided it is technically feasible.

With the kinematic error (10) and well designed data set of $({}^b y_e^l, q^l)$ the cost criterion E for the kinematics calibration task can be formulated as follows:

$$E(\Delta\Phi) := \sum_{l=1}^m \|\Delta y_e^l\|_2^2 = \sum_{l=1}^m \|J(\hat{\Phi}, q^l)\Delta\Phi\|_2^2. \quad (19)$$

The sought parameters vector $\hat{\Phi}$ can be found by solving the following optimization problem: determine the value $\hat{\Phi}$, such that

$$\hat{\Phi} = \Phi_{\text{init}} + \Delta\hat{\Phi}, \quad (20)$$

where

$$\Delta\hat{\Phi} = \underset{\Delta\Phi}{\operatorname{argmin}} E(\Delta\Phi). \quad (21)$$

The value of $\Delta\hat{\Phi}$ in (21) is determined numerically using non-linear least square optimization or other non-linear optimization algorithms available in numerical computing environments such as Matlab.

4. POSE SELECTION FOR KINEMATIC CALIBRATION

As noted in Section 3.3 the performance of a kinematics calibration algorithm strongly relies on applied data sets composed of $({}^b y_e^l, q^l)$, where $l = 1 \dots m$. It is recommended, e.g. in [1, 6], to use observability indices, such as (17), for generation of q^l . A silent assumption in this case is to have a coordinate measuring machine capable of determining the end-effector pose, ${}^b y_{e,\text{ref}}^l$, associated to q^l .

As indicated in Section 1 and Section 2.5 we are focused on a specialized low-cost solution for an ARIA manipulator. In the considered case the end-effector operates within a small area of the workspace resulting from an analysis of users' requirements that are outlined in Section 2.2 and Section 2.3.

We propose a calibration tool that allows the end-effector of an ARIA manipulator to be placed in predefined ${}^b y_{e,\text{ref}}^l$ poses, where $l = 1, \dots, 10$. The associated q^l configuration is a resultant value. The individual values for ${}^b y_{e,\text{ref}}^l$ are results of systematic and extensive numerical studies discussed in subsequent sections: Section 4.1 ÷ Section 4.3.

The value $m = 10$ is a compromise between a recommended value for m and the efficiency of data acquisition. The numerical and empirical results presented in the following sections show that this value is correct.

4.1. Kinematics Calibration Toolbox

Kinematics Calibration Toolbox is a custom software framework developed in Matlab for the maintenance of a kinematic calibration process. The toolbox supports activities at each of the key stages of calibration. Matlab is one of the best and most reliable environments for the implementation of complex numerical procedures operating on measurement data. Currently offered Matlab toolboxes for robotics do not support the calibration process.

4.1.1. Modelling

Mathematical objects, discussed in Section 2.1, Section 3.1 and Section 3.2, are represented by several Matlab global variables and functions. The kinematic chain of ARIA from Fig. 2 is implemented as an Matlab object of the Robotics Toolbox *Serial Link* class. The nominal values of Denavit-Hartenberg parameters of ARIA in Table 1 and the limit values for specific joint angles (2) are stored in Matlab tables. The variables: p_e, ϕ_e, y_e and q in (5), Φ, Φ_i in (9), Δy_e in (10), ${}^b \hat{p}_e$ in (15), ${}^b R_e, {}^b \hat{R}_e$ in (16) are represented by several Matlab variables each. Basically, they are designed to store initial, actual, and ultimate values. The J jacobian in (12) is represented by a Matlab function which is the result of the implementation of (13), (14). It is also one of the toolbox utilities. Consequently, this toolbox is very flexible from the viewpoint of the experiment design and implementation.

4.1.2. Data acquisition

There are three logically interrelated methods of the data acquisition that are supported by the Kinematics Calibration Toolbox. The first method refers to standard techniques reported e.g. in [1, 6]. Randomly chosen $q_{\text{init}}^l, l = 1, \dots, m = 10$ are optimized according to (17), (18) using the Matlab `fmincon` function from the Optimization Toolbox that allows to take into account the joints constraints in (2). Then the resulting q^l are used to determine ${}^b y_e^l$ based on (5)÷(8) with the help of the Matlab object representing the ARIA kinematic chain. This object is constructed for Φ pretending to store the true values of the ARIA mDH parameters, such that $\Phi \neq \Phi_{\text{nom}}$. The resulted data set of $(q^l, {}^b y_e^l)$ is designed for validation of identification procedures in Section 4.1.3 and as a reference for the data sets generated using the subsequent methods.

In the second method, ${}^b y_e^l$ for $l = 1, \dots, m = 10$ are determined by *calibration tools*, discussed in Section 4.2 and Section 4.3. The values of the associated q^l are determined using an inverse kinematics method, being a part of the Matlab object representing ARIA kinematics with Φ parameters. The results obtained here are compared with the outcomes of the first method. The value of the kinematic error (10) associated with the resulted $\hat{\Phi}$ is a comparison criterion. This acquisition method is basically intended for iterative design and numerical evaluation of a calibration tool.

The above discussed acquisition methods acquire data from the virtual ARIA manipulator. The last one is designed for data acquisition from measurements on the real manipulator. This is a two step procedure. First, the measurements are stored in a

computer memory in a CSV format. Next, the data are imported to a Matlab workspace by a suitable utility of the toolbox. The measurement of q^l is taken using encoders in the joints of an ARIA manipulator. The values of ${}^b y_e^l$ are measured in two ways: using a motion capture system or a *calibration tool*. In both cases, dedicated fixtures are mounted to the manipulator wrist. The role of the acquired data is twofold. First they are used for identification and evaluation of $\hat{\Phi}$. Second, they allow for comparison of a virtual and real *calibration tool* and then evaluation of the real tool.

4.1.3. Identification

There are three identification algorithms referring to (19), (20), (21) that are available in the Kinematics Calibration Toolbox. This algorithmic redundancy is intended to increase the reliability of numerical studies. One of them is based on a custom implementation of a Gauss-Newton's algorithm. At some stage it uses an ordinary least square algorithm implemented in a Matlab `lsqminnorm` function. A second identification algorithm is based on the Matlab `fmincon` function with default settings. It allows us to take into account a prior knowledge on intervals boundaries of identified parameters. The last one is based on the Matlab `lsqnonlin` function and can be perceived as a build-in Matlab counterpart of the first algorithm. The mentioned above algorithms are labelled by *OLS*, *OPT1*, *OPT2* (see below in Fig. 7 and Fig. 11).

4.1.4. Validation

There are three main Matlab scripts in the Kinematics Calibration Toolbox that are designed for conducting the entire kinematics calibration process. Their roles can be summarized as follows:

- 1) calibration based on data sets obtained using virtual kinematic chain of ARIA;
- 2) calibration tool design and analysis with data sets based on a virtual kinematic chain of ARIA;
- 3) calibration based on real data sets.

The outcome of each script includes:

- 1) the values of estimates of $\hat{\Phi}$ mDH parameters returned by each identification routine (see e.g. Table 1, Table 3 and Table 4);
- 2) compact plots with position and orientation errors: $\Delta^b p_e$, $\Delta^b \phi_e$ (see Fig. 7 and Fig. 11). These plots are arranged in the form of 4×4 arrays. The X , Y , Z coordinates of $\Delta^b p_e$ ($\Delta^b \phi_e$) and the total error $\|\Delta^b p_e\|_2$ ($\|\Delta^b \phi_e\|_2$) are exposed in each column of such an array. The first column is associated to mDH parameters having nominal values (*NOM*). In the subsequent columns, mDH parameters have values identified using the *OLS*, *OPT1* and *OPT2* algorithms, respectively.

The aforementioned data are the basis for evaluating calibration results, comparative analysis, and evaluation of calibration tool designs. In order to highlight the dependence of position and orientation errors on $\hat{\Phi}$, we will use the following notations below: $e_p(\hat{\Phi}) := \Delta^b p_e$, $e_\rho(\hat{\Phi}) := \Delta^b \phi_e$.

4.2. Workspace clustering technique

The idea of workspace clustering is depicted in Fig. 5. First, a region of the ARIA workspace is selected, see a gray cuboid in Fig. 5a. Notice, that the selected region in Fig. 5a corresponds to an operational region of the end-effector of the manipulator in Fig. 3. Then the area is divided into clusters in the form of cubes of side length L . Finally, q^l , $l = 1, \dots, m = 10$ configurations are searched such that they have the same values of the observability index $O(q^l)$, see (17). The associated with them ${}^b p_{e,\text{ref}}^l$ positions must belong to mutually different clusters, see Fig. 5b. The resulted ${}^b y_{e,\text{ref}}^l$ in this way constitutes a design of a *calibration tool*.

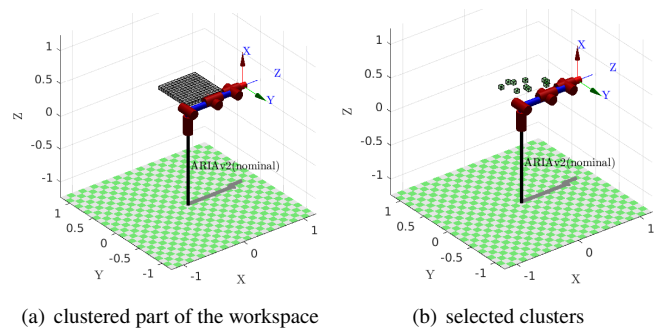


Fig. 5. Workspace clustering technique

The results of numerical studies of ARIA kinematics calibration based on data sets obtained using the workspace clustering technique are included in Fig. 7. They are discussed in Section 4.4.

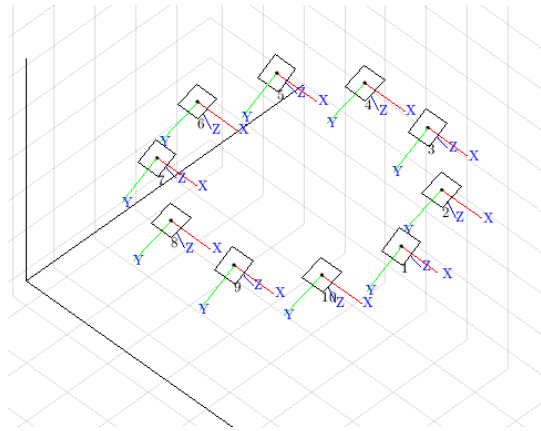
4.3. Arbitrary choice technique

The set of requested end-effector poses $y_{e,\text{ref}}^l$ for a *calibration tool* was specified arbitrarily, based on empirical analysis of an operational region of the ARIA end-effector in Fig. 3. The specific values are collected in Table 2 and they are visualized in Fig. 6a. Notice, that this proposition of a *calibration tool*

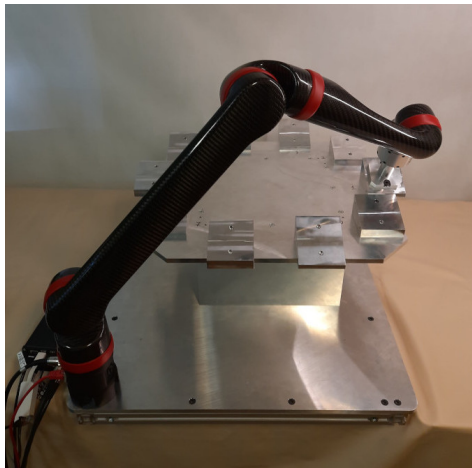
Table 2
Calibration tool specification

l	${}^b p_{e,\text{ref}}^l$			${}^b \phi_{e,\text{ref}}^l$ (RPY/zyx)		
	X [m]	Y [m]	Z [m]	X [deg]	Y [deg]	Z [deg]
1	0.566	0.354	0.235	-150	0	0
2	0.518	0.501	0.235	-160	0	0
3	0.393	0.592	0.235	-150	0	0
4	0.238	0.592	0.235	-160	0	0
5	0.113	0.501	0.235	-150	0	0
6	0.066	0.354	0.235	-160	0	0
7	0.113	0.207	0.235	-150	0	0
8	0.238	0.116	0.235	-160	0	0
9	0.393	0.116	0.235	-150	0	0
10	0.518	0.207	0.235	-160	0	0

is an implementation friendly. All the suggested end-effector orientations ${}^b\phi_{e,\text{ref}}^l$ are the same while end-effector positions, ${}^b p_{e,\text{ref}}^l$, are evenly distributed over a circle of radius 0.25 m.



(a) design



(b) implementation and usage

Fig. 6. Arbitrary choice technique

The results of numerical studies of ARIA kinematics calibration based on data sets obtained using the arbitrary choice technique are included in Fig. 7. They are discussed in Section 4.4.

The implementation of the calibration tool is shown in Fig. 6b. The truncated square prisms in Fig. 6b, located at the top aluminium plate, are counterparts of the rectangles in Fig. 6a. The base of the manipulator is fixed rigidly to the bottom aluminium plate in the left corner. The centre of fastening determines the origin of the coordinate frame, which can be seen at the picture in Fig. 6a. This coordinate frame can be identified with the base coordinate frame or with $X_0Y_0Z_0$ in Fig. 2. The element with a cylindrical form visible at the upper right corner of the picture in Fig. 6b is mounted to the last link of the manipulator instead of a gripper. If this element is in correct contact with the l -th truncated square prism in the calibration tool, then ${}^b p_{e,\text{ref}}^l$ and ${}^b\phi_{e,\text{ref}}^l$ take a value from Table 2 and the measurements of q^l can be made for data acquisition.

Table 3

Identified mDH parameters values of a virtual ARIA manipulator: an example resulting from numerical analysis (the case when the OLS algorithm was used)

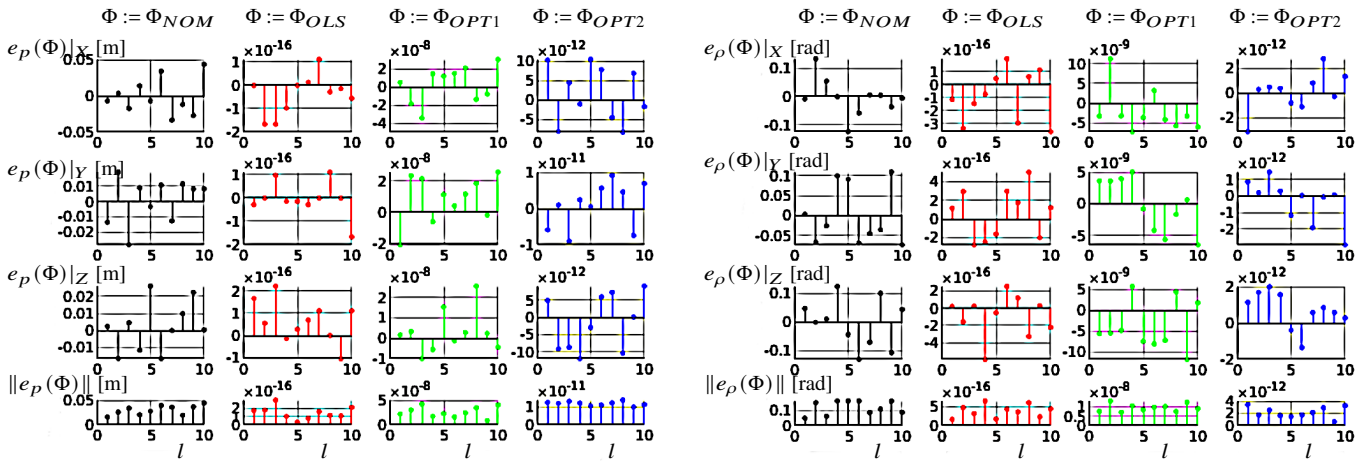
i	θ_i^0 [rad]	d_i [m]	a_i [m]	α_i [rad]	β_i [rad]
1	1.67e-7	0.205	1.28e-6	1.57	*
2	1.17e-6	*	0.41	0.1	-1.41e-7
3	1.57	-0.0048	0.03	1.57	*
4	-0.8	0.34	-3.25e-7	-1.57	*
5	3.16e-7	-0.049	-2.00e-8	1.57	*
6	-3.97e-7	0.205	-7.38e-8	5.39e-8	*

4.4. Comparative analysis

The analysis of plots with position and orientation errors in Fig. 7 leads to the following conclusions.

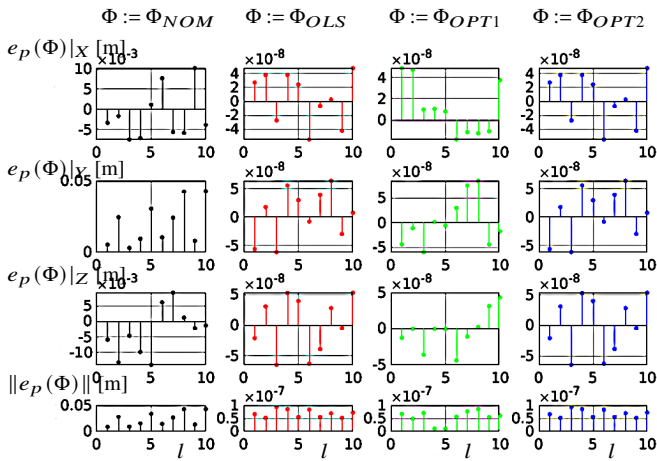
1. The data sets obtained using standard techniques as well as the two techniques proposed in this article, i.e. the clustering workspace technique and the arbitrary choice technique leads to mDH estimates $\hat{\Phi}$ which guarantee significant reduction of position and orientation errors. In each case the values of these errors are smaller than the requested precision (i.e. $e_p(\hat{\Phi}) < 1$ mm and $e_\rho(\hat{\Phi}) < 0.66^\circ$ for each entry, see Section 2.4).
2. When comparing the plots in Fig. 7a, b with the corresponding plots in Fig. 7(c,d) and Fig. 7(e,f), it can be seen that the position and orientation errors $e_p(\hat{\Phi})$, $e_\rho(\hat{\Phi})$ associated to standard data acquisition methods are much smaller than the one associated to the data acquisition techniques proposed in this section. This is because in the first case the data $(q^l, {}^b y_e^l)$ is acquired from the entire workspace whereas in the other cases these data is acquired from a small region of the manipulator workspace. In simulations, when simplified manipulator models are taken into account, we know that there exists a single value of $\hat{\Phi}$ (e.g. $\hat{\Phi} = \Phi^*$) such that $e_p(\hat{\Phi}) \equiv 0$, $e_\rho(\hat{\Phi}) \equiv 0$ for any $(q^l, {}^b y_e^l)$. Therefore the more diverse data acquired from the entire workspace the better results of calibration. In the case of a real robot we are faced with implementation imperfections. They cause that there is no single value of $\hat{\Phi}$ for which $e_p(\hat{\Phi})$ and $e_\rho(\hat{\Phi})$ are equal to zero or close to zero across the entire workspace. Therefore it is better to take data $(q^l, {}^b y_e^l)$ from the intended working region. The resulting value of $\hat{\Phi}$ should reduce calibration errors $e_p(\hat{\Phi})$ and $e_\rho(\hat{\Phi})$ inside this region. Measurement data from outside this region can negatively affect the error.
3. The workspace clustering technique provides strongly differentiated values for the variables $(q^l, {}^b y_e^l)$, $l = 1, \dots, m$ that share the same observability index $O(q^l)$. This is an advantage from a computational perspective but a disadvantage from an implementation perspective. On the other hand, the opposite is true for the arbitrary choice technique. It is relatively easy to obtain implementation-friendly variables $(q^l, {}^b y_e^l)$ without guaranteeing good conditioning of the calibration algorithm. We considered the first technique

Kinematic calibration of a lightweight manipulator

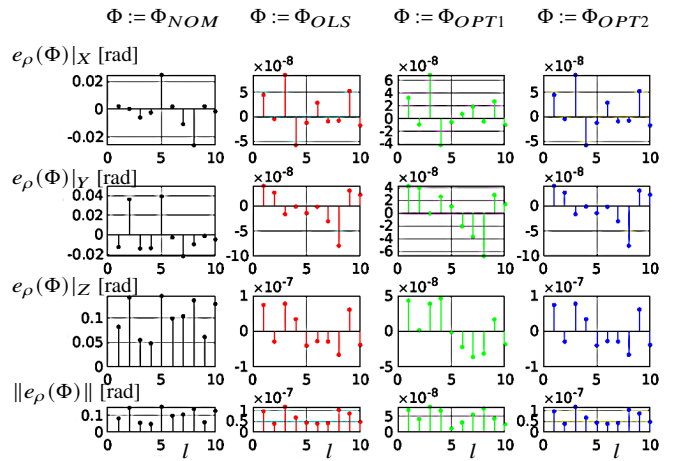


(a) position ($q^l, l = 1 \dots, 10$, chosen using (18))

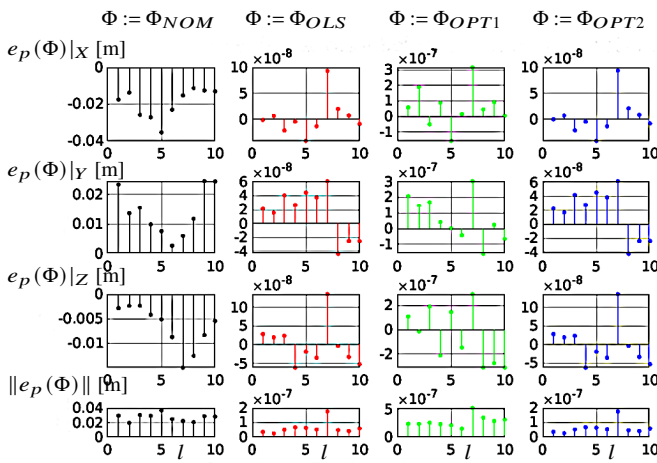
(b) orientation ($q^l, l = 1 \dots, 10$, chosen using (18))



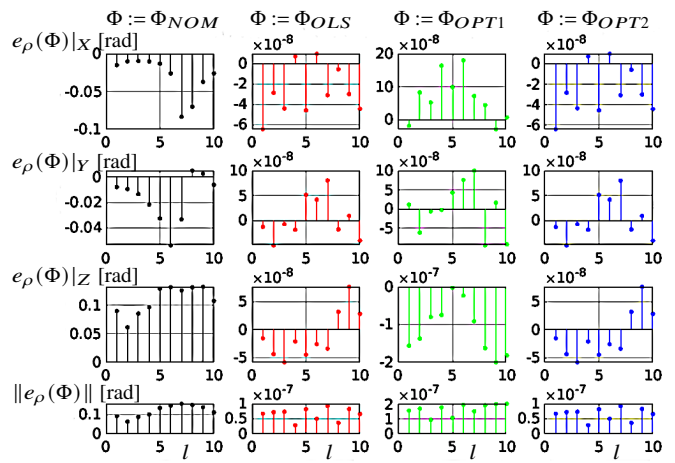
(c) position (workspace clustering)



(d) orientation (workspace clustering)



(e) position (arbitrary choice)



(f) orientation (arbitrary choice)

Fig. 7. Summary of plots with $e_p(\Phi)$ and $e_\rho(\Phi)$ for different data set acquisition methods

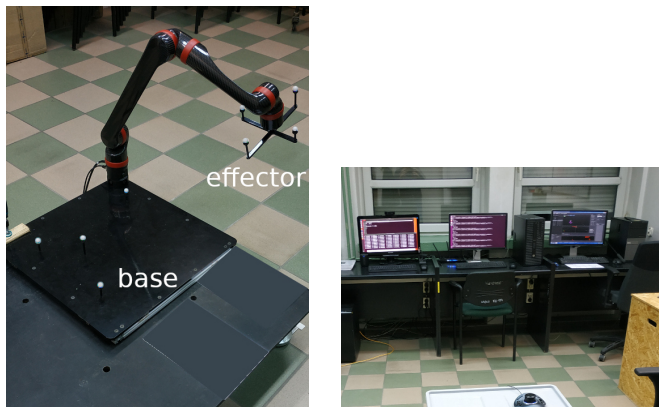
as a benchmark for the second. If the calibration errors $e_p(\hat{\Phi})$ and $e_\rho(\hat{\Phi})$ for the measurement data obtained by the workspace clustering are larger than ones from the arbitrary choice techniques we choose the latter. Otherwise, we again use the arbitrary choice technique to find better values for $(q^l, b y_e^l)$. The calibration tool specification in Table 2 is the result of the above procedure. To see it compare the corresponding plots in Fig. 7c, d) and Fig. 7e, f. Note that the calibration errors presented in these figures associated to the workspace clustering and arbitrary choice techniques are of the same order of magnitude.

5. EXPERIMENTAL STUDY

The calibration results of a real ARIA manipulator are presented and discussed in this section. The data acquisition is performed using the calibration tool from Section 4.3. The identification and validation processes are carried out with the use of Kinematics Calibration Toolbox from Section 4.1. In what follows an experimental setup is discussed in Section 5.1. Experimental results and their discussion are included in Section 5.2 and Section 5.3, respectively.

5.1. Experimental setup

The core hardware components of the experimental setup include: ARIA manipulator and three PC workstations, shown in Fig. 8, complemented by six OptiTrack Prime 17W cameras. All of them are integrated using a laboratory computer network.



(a) ARIA manipulator with an end-effector probe and motion capture markers (b) PC workstations for the data acquisition and processing

Fig. 8. The experimental setup core components

The ARIA manipulator is mounted to a solid base made of sheet steel. Besides, there are four motion capture markers mounted to the base that are visible in Fig. 8. They are aimed to synchronize the optical global coordinate frame of the motion capture system being a part of the experimental setup with the manipulator base coordinate frame. A cross-shaped probe is mounted to the end of the last link of the manipulator. A motion capture marker is attached to each probe tip, that can be seen in

Fig. 8. The probe is designed to determine the pose of the end-effector coordinate frame using an OptiTrack motion capture system. Coordinate frames associated with the markers fixed to the base and the probe, respectively are defined in a Motive software, as it illustrated in Fig. 9. Their poses in the optical global coordinate frame of the motion capture system are returned by Motive and then they are used to determine the position ${}^b p_e$ and the orientation ${}^b \phi_e$ of the end-effector coordinate frame in the base coordinate frame as the positions of each marker in the base and end-effector coordinate frames are known from the design.

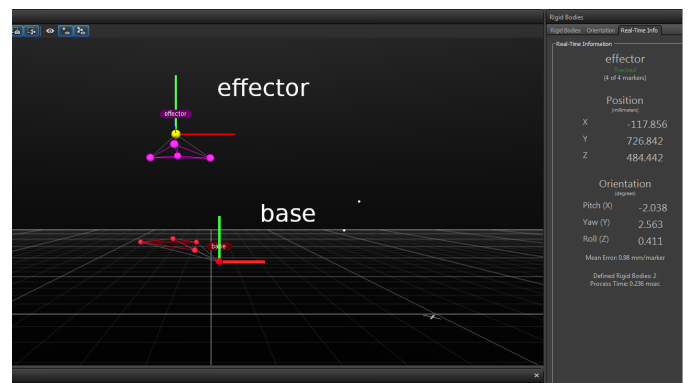


Fig. 9. The base and probe coordinate frames in Motive – an OptiTrack motion capture user's interface

Significant spacing between markers in the cross-shaped probe ensures sufficient accuracy for the motion capture system to measure the position and orientation of the end-effector.

The software integration of the experimental setup for the data acquisition was implemented using Robot Operating System (ROS). A general software architecture in terms of ROS nodes and topics is shown in Fig. 10. /aria_v2_ros_driver is a ROS interface to ARIA, developed by the manufacturer. /mocap_node is a ROS interface to OptiTrack motion capture system that is available in a ROS repository. The other ROS nodes are parts of ManipKinCal_ROS software developed for the needs of the integration and to ensure the effectiveness

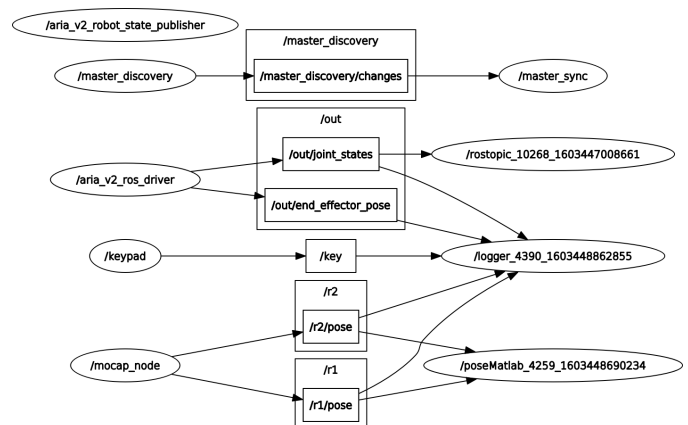


Fig. 10. ROS nodes and topics of the experimental setup software for the data acquisition

of conducting the data acquisition using the experimental setup. The /aria_v2_ros_driver node runs on the left PC in Fig. 8b along with the nodes from ManipKinCal_ROS. The /mocap_node and roscore run on the middle PC while the Motive program runs on the right one in the figure under consideration.

A file in a CSV format is the final result of an experiment. It includes records of data having the sens of q^l , ${}^b p_e^l$, ${}^b \phi_e^l$, $l = 1, \dots, 10$, identically as in Section 4.1.2. The data stored in this file is further processed using the Kinematics Calibration Toolbox.

The experimental procedure consisted in placing the probe in the position ${}^b p_e^l$ and orientation ${}^b \phi_e^l$ specified in Table 2 and then reading and storing the resulted joint configuration q^l . The probe is moved manually, using a 3DConnection space mouse, until the moment when the desired value of the end-effector pose appears on a computer screen, some in Fig. 8b. Then the actual value of a joint configuration is stored on a file in response to a keypress that is handled by a ManipKinCal_ROS software component.

5.2. Results

The results of calibrating of mDH parameters of ARIA obtained using the custom implementation of a Gauss-Newton's algorithm are collected in Table 4. The results calculated by two other identification algorithms available in Kinematics Calibration Toolbox are similar. The nominal values of these parameters, collected in Table 1 were taken as the initial parameters for optimization procedures from the Kinematics Calibration Toolbox. The exception is the link No 2, where $\beta_2 = 0$ was used instead d_2 .

As in Section 4.1.4, position and orientation errors associated to the identified mDH parameters are determined and visualized in Fig. 11 for the assessment of these parameters.

Table 4

Identified mDH parameters values of the ARIA manipulator: an example resulting from experimental studies (the case when the OLS algorithm was used)

i	θ_i^0 [rad]	d_i [m]	a_i [m]	α_i [rad]	β_i [rad]
1	0.0068	0.21	-0.0016	1.58	*
2	-0.0024	*	0.41	-0.011	0.0063
3	1.54	-0.0066	0.036	1.59	*
4	-0.015	0.34	-0.00081	-1.59	*
5	0.026	-0.049	0.0018	1.56	*
6	0.0093	0.216	-0.0011	-0.0087	*

5.3. Discussion

Comparison of Table 4 and Table 1 leads to the observation that the identified mDH parameters are comparable with the nominal ones but they are clearly different. The same observation can be made when comparing Table 4 with Table 3. At this point, it should be taken into account that kinematics of virtual and real ARIAs are slightly different.

Figure 11 allows the comparison of the end-effector position and orientation errors for nominal mDH parameters and the identified ones. Errors are reduced by about five times. The improvement is clear but not as great as in the numerical analysis results in Fig. 7. This is most likely due to nongeometric variations that occur in the real ARIA manipulator.

These observations apply to the part of the manipulator workspace around the calibration tool specified in Table 2. This is acceptable given the envisaged use of the manipulator, outlined in Section 2.3. It follows from Fig. 11 that $e_p(\Phi) \approx 0.5$ mm and $e_p(\Phi) \approx 0.43^\circ$ which meets the accuracy requirements set out in Section 2.4.

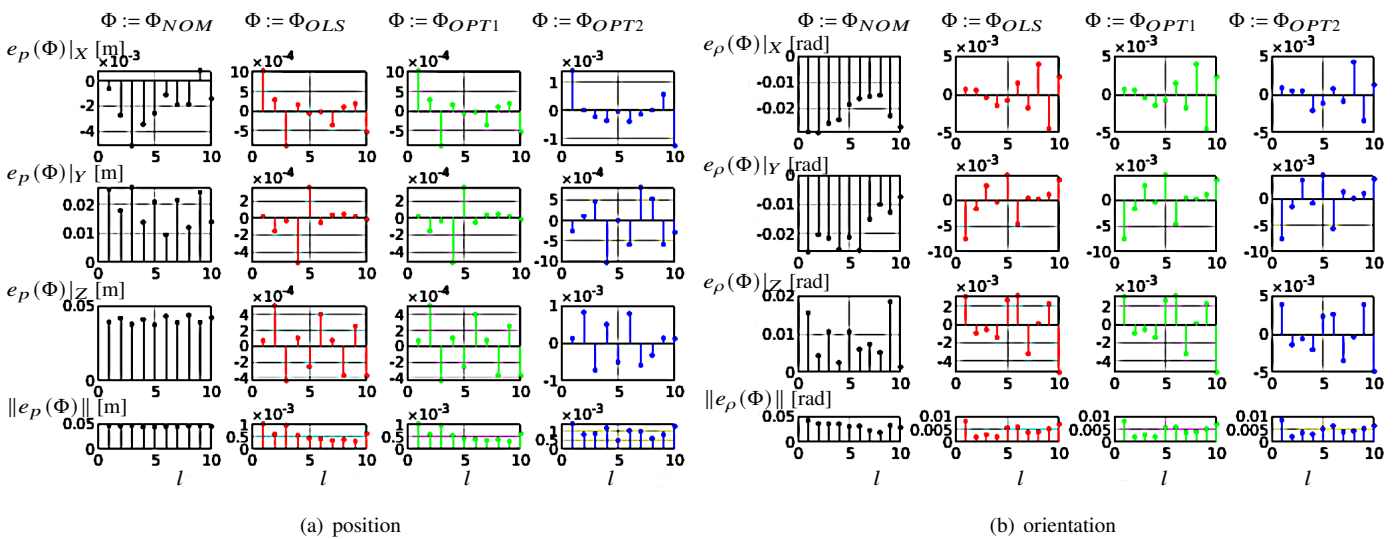


Fig. 11. Plots with $e_p(\Phi)$ and $e_p(\Phi)$ for data set acquired from experimental studies

The manipulator stiffness deficit is certainly a major component of the causes of the aforementioned nongeometric variations. A formal description of the stiffness deficit is difficult and therefore it is treated as an uncertainty in the kinematics model. Fortunately, this uncertainty is small enough that calibration of the kinematics parameter values makes it possible to reduce the position and orientation error of the end-effector to an acceptable value.

The OptiTrack motion capture system was used to empirically verify the results of the simulation studies reported in Fig. 7e, f (Section 4.4). A low-cost calibration tool shown in Fig. 6b was manufactured after the aforementioned results were found to be acceptable. This methodology ensures that the cost of manufacturing the calibration tool is not incurred without ensuring that it is justified. The OptiTrack motion capture system is not used to validate the Kinematics Calibration Tool. The precision of the tool is to be a consequence of proper design and appropriate manufacturing technology.

6. CONCLUSIONS

A kinematic calibration procedure for ARIA manipulators has been proposed in this article. Empirical results show a clear reduction in the end-effector pose error, by several times, after applying the identified parameters to the manipulator kinematics. It is comparable to the accuracy reported in [12].

Special attention has been paid to lowering the cost of a calibration procedure so that it could be affordable for a small/medium enterprise (SME). Activities in this direction are justified because SMEs often develop niche robots that meet specific and important human needs and at the same time SMEs face with limited budgets. Because the end-effector position and orientation measurement system is the most expensive part of the setup for kinematics calibration we proposed a low-cost calibration tool that eliminates the need for such a costly equipment in a specific application.

The design of the calibration tool is simple, based on a circular plan. This is the result of the manipulator usage analysis and extensive numerical studies. The new Workspace Clustering Technique and the Arbitrary Choice Technique played a significant role in these studies. The subsequent procedures of identification and verification in the calibration process are standard. All steps of the calibration process are supported by a specially developed toolbox in Matlab.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support of the National Centre for Research and Development as part of the “Smart Growth 2014–2020” project (activity 1.1), co-financed by the European Regional Development Fund, for their funding of the L2R project, which provided the framework for the work presented in this article.

REFERENCES

- [1] J.M. Hollerbach, W. Khalil, and M. Gautier, “Model identification,” in *Springer Handbook of Robotics*. Springer, 2016, pp. 113–138, doi: [10.1007/978-3-319-32552-1_6](https://doi.org/10.1007/978-3-319-32552-1_6).
- [2] K. Tchoń, “Calibration of manipulator kinematics,” *IEEE Trans. Robot. Automat.*, vol. 8, no. 5, pp. 671–678, 1992, doi: [10.1109/70.163792](https://doi.org/10.1109/70.163792).
- [3] A. Goswami, A. Quaid, and M. Peshkin, “Identifying robot parameters using partial pose information,” *IEEE Contr. Syst. Mag.*, vol. 13, no. 5, pp. 6–14, 1993, doi: [10.1109/37.236317](https://doi.org/10.1109/37.236317).
- [4] M.R. Driels and W.E. Swayze, “Automated partial pose measurement system for manipulator calibration experiments,” *IEEE Tran. Robot. Automat.*, vol. 10, no. 4, pp. 430–440, 1994, doi: [10.1109/70.313094](https://doi.org/10.1109/70.313094).
- [5] A. Horne and L. Notash, “Pose selection for the kinematic calibration of a prototyped 4 degrees of freedom manipulator,” *Trans. Can. Soc. Mech. Eng.*, vol. 33, pp. 619–632, 2009, doi: [10.1139/tcsme-2009-0043](https://doi.org/10.1139/tcsme-2009-0043).
- [6] S. Kolyubin, L. Paramonov, and A. Shiriaev, “Robot kinematics identification: KUKA LWR4+ redundant manipulator example,” *J. Phys.-Conf. Ser.*, vol. 659, no. 1, p. 012011, 2015, doi: [10.1088/1742-6596/659/1/012011](https://doi.org/10.1088/1742-6596/659/1/012011).
- [7] S. Marie, E. Courteille, and P. Maurine, “Elasto-geometrical modeling and calibration of robot manipulators: Application to machining and forming applications,” *Mech. Mach. Theory*, vol. 69, pp. 13–43, 2013, doi: [10.1016/j.mechmachtheory.2013.05.003](https://doi.org/10.1016/j.mechmachtheory.2013.05.003).
- [8] P. Li, R. Zeng, W. Xie, and X. Zhang, “Relative posture-based kinematic calibration of a 6-rss parallel robot by optical coordinate measurement machine,” *Int. J. Adv. Robot. Syst.*, vol. 15, pp. 1–14, 2018, doi: [10.1177/1729881418765861](https://doi.org/10.1177/1729881418765861).
- [9] B. Curtis, “Robotic arm kinematics and calibration 6-dof powerball Iwa 4p,” U.S. Army Combat Capabilities Development Command Armaments Center, Benét Laboratories, Tech. Rep., Feb 2019.
- [10] G. Tang and L. Liu, “Robot calibration using a single laser displacement meter,” *Mechatronics*, vol. 3, no. 4, pp. 503–516, 1993, doi: [10.1016/0957-4158\(93\)90020-3](https://doi.org/10.1016/0957-4158(93)90020-3).
- [11] A. Nubiola and I.A. Bonev, “Absolute calibration of an abb irb 1600 robot using a laser tracker,” *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 1, pp. 236–245, 2013.
- [12] A. Joubair and I. Bonev, “Kinematic calibration of a six-axis serial robot using distance and sphere constraints,” *Int. J. Adv. Manuf. Technol.*, vol. 77, p. 515–523, 2014, doi: [10.1007/s00170-014-6448-5](https://doi.org/10.1007/s00170-014-6448-5).
- [13] J. Li, L.-D. Yu, J.-Q. Sun, and H.-J. Xia, “A Kinematic Model for Parallel-Joint Coordinate Measuring Machine,” *J. Mech. Robot.*, vol. 5, no. 4, p. 044501, 2013, doi: [10.1115/1.4025121](https://doi.org/10.1115/1.4025121).
- [14] N. Shen *et al.*, “Efficient Model-Free Calibration of a 5-Degree of Freedom Hybrid Robot,” *J. Mech. Robot.*, vol. 14, no. 5, p. 051011, 2022, doi: [10.1115/1.4053824](https://doi.org/10.1115/1.4053824).
- [15] Z. Wang, S. Bao, B. Zi, Z. Jia, and X. Yu, “Development of a Novel 4-DOF Flexible Endoscopic Robot Using Cable-driven Multi-segment Continuum Mechanisms,” *J. Mech. Robot.*, vol. 16, p. 031011, 2023, doi: [10.1115/1.4057075](https://doi.org/10.1115/1.4057075).
- [16] R. Ju *et al.*, “Design, Modeling, and Kinematics Analysis of a Modular Cable-Driven Manipulator,” *J. Mech. Robot.*, vol. 14, no. 6, p. 060903, 2022, doi: [10.1115/1.4054206](https://doi.org/10.1115/1.4054206).

Kinematic calibration of a lightweight manipulator

- [17] C. Huang, F. Xie, X.-J. Liu, and Q. Meng, "Measurement configuration optimization and kinematic calibration of a parallel robot," *J. Mech. Robot.*, vol. 14, pp. 1–11, 2021, doi: [10.1115/1.4053012](https://doi.org/10.1115/1.4053012).
- [18] Z. Wang, B. Cao, Z. Xie, B. Ma, K. Sun, and Y. Liu, "Kinematic calibration of a space manipulator based on visual measurement system with extended Kalman filter," *Machines*, vol. 11, no. 3, p. 409, 2023, doi: [10.3390/machines11030409](https://doi.org/10.3390/machines11030409).
- [19] G. Boschetti and T. Sinico, "A novel step-by-step procedure for the kinematic calibration of robots using a single draw-wire encoder," *Int. J. Adv. Manuf. Technol.*, vol. 131, pp. 4129–4147, 2024, doi: [10.1007/s00170-024-13219-1](https://doi.org/10.1007/s00170-024-13219-1).
- [20] J. Peng, Y. Ding, G. Zhang, and H. Ding, "An enhanced kinematic model for calibration of robotic machining systems with parallel mechanisms," *Robot. Comput.-Integr. Manuf.*, vol. 59, pp. 92–103, 2019, doi: [10.1016/j.rcim.2019.03.008](https://doi.org/10.1016/j.rcim.2019.03.008).
- [21] A. Kurnicki and B. Stańczyk, "Development of a modular light-weight manipulator for human-robot interaction in medical applications," *Informatyka, Automatyka, Pomiar w Gospodarce i Ochronie Środowiska*, vol. 10, pp. 33–37, 2020, doi: [10.35784/iapgos.2066](https://doi.org/10.35784/iapgos.2066).
- [22] S. Hayati and M. Mirmirani, "Improving the absolute positioning accuracy of robot manipulators," *J. Robot. Syst.*, vol. 2, no. 4, pp. 397–413, 1985, doi: [10.1002/rob.4620020406](https://doi.org/10.1002/rob.4620020406).
- [23] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, ser. Advanced Textbooks in Control and Signal Processing. Springer London, 2010.