FIGAROH: a Python toolbox for dynamic identification and geometric calibration of robots and humans

Dinh Vinh Thanh Nguyen, Vincent Bonnet, Sabbah Maxime, Maxime Gautier, Pierre Fernbach, Florent Lamiraux

FIGAROH: a Python toolbox for dynamic identification and geometric calibration of robots and humans

Thanh D. V. Nguyen[1,2], Vincent Bonnet[1,*], Maxime Sabbah[1], Maxime Gautier[3], Pierre Fernbach[2], Florent Lamiraux[1]

***Abstract*— The accuracy of the geometric and dynamic models for robots and humans is crucial for simulation, control, and motion analysis. For example, joint torque, which is a function of geometric and dynamic parameters, is a critical variable that heavily impacts the performance of model-based control, or that can motivate a clinical decision after a biomechanical analysis. Fortunately, these models can be identified using extensive works from literature. However, for a non-expert, building an identification model and designing an experimentation plan, which should not require long hours and/or lead to poor results, is not a trivial task, especially for anthropometric structures such as humanoids or humans that need frequent update. In this work, we propose a unified framework for geometric calibration and dynamic identification in the form of a Python open-source toolbox. Besides identification model building and data processing, the toolbox can automatically generate exciting postures and motions to minimize the experimental burden from the robot, measurements, and environment description. The possibilities of this toolbox are exemplified with several datasets of human, humanoid, and serial robots.**

I. INTRODUCTION

Robot unit specific geometric and dynamic parameters can be provided by robot manufacturer or accounted for in the controller, but often they use simplistic models while requesting extra fees. Geometric parameters refer to the 3D slight position and orientation offsets of each joint, while dynamic parameters refer to the Segment Inertial Parameters (SIP), joint and drive chain parameters. SIP are often provided by manufacturer. However, joint and drive chain parameters are usually unknown despite they can have a dramatic effect on the joint torque estimate. For example, the current gain drive provided by a robot motor manufacturer is known to have an inaccuracy of 10 to 15% [1]. In human motion analysis, it is necessary to perform a geometric calibration as the location of motion sensors changes at each new experiment. The human SIP can be estimated from population-averaged Anthropometric Tables (AT) [2], but they are notoriously inaccurate for people with atypical mass distribution, such as infants, obese individuals, or athletes.

Fortunately, all these parameters can be identified, as identification of rigid multi-body systems is a mature research field that has been used with numerous robots [3], [4] and in biomechanics to analyse human motion [5]. Although few recent toolboxes were proposed [6], [7], [8], [9], there is still no unified framework for geometric and dynamic identification. Pybotics [6] and the Robot Calibration package for ROS [9] are specifically designed for geometric calibration of robots, including 6D robot frame offsets and camera parameters using 3D pose provided by an external device or a closed-loop constraint, respectively. The toolbox Multirobot-calibration [7] provides different frameworks for robot geometric calibration including self-contained, self-touch methods. The BIRDy software [8], on the other hand, presents an exhaustive benchmark of different robot dynamic identification algorithms. However, no toolbox has handled the automatic generation of exciting data depending on specific robot structure, measurement type and location and environment.

A geometrical calibration process requires the measurement of kinematic data collected over calibration postures. For robots, kinematic measurements refer to encoder data and external pose measurements coming from a Laser tracker for instance [10], or using closed-loop constraints [11]. For human, they can be collected from a Stereophotogrammetric System (SS) for example. Dynamic identification process requires the measurement of kinematic and dynamometric quantities. The type and accuracy of dynamometric quantities vary from one system to another. They can be the motor torques or currents of a robot, the joint torques or the external wrench applied to the system measured with a force sensor or a force-plate at different locations [5]. Depending on the type of dynamometric inputs, different parameter sets can be identified, and thus different identification model should be devised. Once data are collected, they can be inputted into a Least-Square (LS) optimization problem minimizing the difference between measured and estimated entities, and the kinematic and/or dynamic parameters to be identified can be then retrieved. Kinematic and dynamometric data should be collected over exciting static postures for geometrical calibration and exciting motions for dynamic identification. Generally, exciting postures and motions are generated manually in a sequential way or even randomly [12]. This is acceptable for simple serial chains in unconstrained environment. However, for complex robots such as humanoids due to their kinematic complexity, intrinsic instability, and self-collision risks such methods cannot be used. Also, if the postures/motions are randomly designed, then the time required to perform the system identification will be longer. In this case, it is better to use a set of Optimal Exciting

[1] LAAS-CNRS, Université Paul Sabatier, CNRS, Toulouse, France
[2] TOWARD S.A.S, Toulouse, France
[3] LS2N, Université de Nantes, Nantes, France
* corresponding author: vincent.bonnet@laas.fr
This work was partially supported by PAL Robotics

Calibration Postures (OEP) and Optimal Exciting Motions (OEM) specifically designed to excite the parameters to be identified while taking into account mechanical constraints. Literature regarding serial manipulators proposes to generate OEP and OEM using optimization approaches based on the minimization of a criterion related to the condition number or the determinant of the so-called regressor matrix [13], [4], [14], which relates parameters to be identified and measurements. Finally, since regressor matrices are not full-rank due to the fact that some parameters are not identifiable, only the so-called Base Parameters (BP) [15] can be identified with a classical LS approach. However, most of the newly designed robots and simulators are based on the URDF modelling convention that requires each individual parameter. Recently, several studies have proposed solutions to obtain physically consistent SIP using constrained optimization [16]. By doing so, each individual parameter to be identified can be retrieved.

In this context, the paper introduces a practical open-source Python toolbox for the Free Identification of Geometrical and dynAmic parameters of RObots and Humans (FIGAROH). FIGAROH is the first comprehensive toolbox for modeling of identification models, generating OEP and OEM automatically, and processing data of rigid multi-body systems. Several example datasets are described in section V, including the geometric calibration and dynamic identification of a TIAGo collaborative robot, a TALOS humanoid robot, a human, a Staubli TX40 and a UR10 robot.

II. MAIN FEATURES OF FIGAROH

FIGAROH is written in Python for easy data processing and fast prototyping. Geometric and dynamic computations mainly rely on Pinocchio robot modeling library, one of the most efficient libraries for this purpose [17]. Fig. 1 shows an overview of FIGAROH main features. The toolbox takes as input the multi-modal measurements, their type and location on the model, the nominal mechanical model described with a URDF file, and elements of the robot environment such as the location of possible obstacles. The outputs are the identified geometric and/or dynamic parameters, a statistical analysis of the identified parameters, and an updated URDF file of the robot.

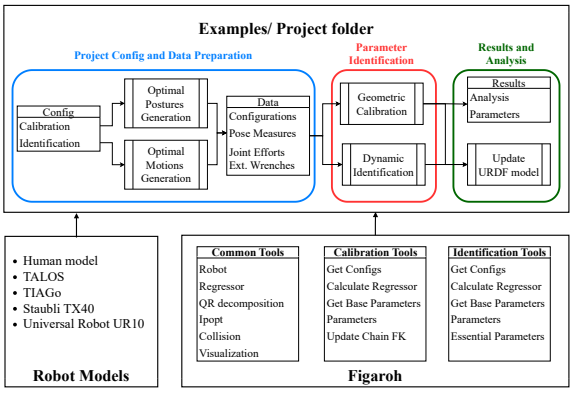

Fig. 1. Overview of FIGAROH main features and organization.

A. Robot modelling

A robot model is composed of one or more kinematic chains with N_L links and N_J joints. The absolute pose $\mathbf{Y}$ of any frame on the robot can be calculated as a function of the joint configuration vector $\mathbf{q}$ and of the local joint pose parameter vector $\mathbf{p}$ using the forward kinematic model:

$$\mathbf{P} = f(\mathbf{q}, \mathbf{p}, \mathbf{\Phi}_g) = {}^{base}\mathbf{T}_1 *^1 \mathbf{T}_2 * ... *^n \mathbf{T}_{end} \quad (1)$$

where $\mathbf{p} = \begin{bmatrix}\mathbf{p}_1 \cdots \mathbf{p}_{N_J}\end{bmatrix}$ are vectors of the nominal values of local joint poses in parent frames with $\mathbf{p}_j = \begin{bmatrix}p_{xj} & p_{yj} & p_{zj} & \phi_{xj} & \phi_{yj} & \phi_{zj}\end{bmatrix}$; and $\mathbf{\Phi}_g = \begin{bmatrix}\mathbf{\Phi}_{g1} \cdots \mathbf{\Phi}_{gN_J}\end{bmatrix}$ are vectors of the local joint pose variations that are to be identified, with $\mathbf{\Phi}_{gj} = \begin{bmatrix}\delta p_{xj} & \delta p_{yj} & \delta p_{zj} & \delta\phi_{xj} & \delta\phi_{yj} & \delta\phi_{zj}\end{bmatrix}$.

Dynamic identification in robots refers to the identification of:

- 10 Segment Inertial Parameters (SIP) for each segment j that are expressed in the joint frame $\mathbf{\Phi}_{dj} = \begin{bmatrix}M_j & \mathbf{MS}_j & \mathbf{TI}_j\end{bmatrix}$, where M_j is the mass, $\mathbf{MS}_j = \begin{bmatrix}MSX_j & MSY_j & MSZ_j\end{bmatrix}$ is the 3-dimensional vector of the first moment of inertia, and the 6-dimensional vector $\mathbf{TI}_j = \begin{bmatrix}XX_j & YY_j & ZZ_j & XY_j & XZ_j & YZ_j\end{bmatrix}$ gathers the components of the 3×3 tensor of inertia.
- Joint parameters for each segment j the static F_{sj} and viscous F_{vj} friction parameters and the joint torque sensor offset off_j.
- Drive chain parameters including for each joint j, $g_{\tau j}$ the motor's input of the current control loop to torque conversion gain, I_{mj} the rotor inertia of the motor, F_{vmj} and F_{smj} the motor viscous and dry frictions.
- Gain K_j associated to joint torque sensors or force plate.

In the general case of a floating base multi-body system, the inverse dynamic model can be calculated using the recursive Newton-Euler method as follows [18], [19]:

$$\begin{bmatrix}\mathbf{H}_{ww} & \mathbf{H}_{wc} \\ \mathbf{H}_{cw} & \mathbf{H}_{cc}\end{bmatrix}\begin{bmatrix}\ddot{\mathbf{q}}_w \\ \ddot{\mathbf{q}}\end{bmatrix} + \begin{bmatrix}\mathbf{b}_w \\ \mathbf{b}_c\end{bmatrix} = \begin{bmatrix}\mathbf{0} \\ \tau\end{bmatrix} + \sum_{k=1}^{N_c}\begin{bmatrix}\mathbf{J}_{w_k}^T \\ \mathbf{J}_{c_k}^T\end{bmatrix}\mathbf{F}_k \quad (2)$$

where the upper part of the equation represents the root-link dynamics, and the lower part accounts for the other chain segment dynamics.

- $\mathbf{H}_{ww} \in \mathbb{R}^{6\times6}$ and $\mathbf{H}_{wc} \in \mathbb{R}^{6\times N_J}$ are the root-link inertia matrices; $\mathbf{H}_{cw} \in \mathbb{R}^{N_J\times6}$, $\mathbf{H}_{cc} \in \mathbb{R}^{N_J\times N_J}$ are the chains segments inertia matrices;
- $\ddot{\mathbf{q}}_w \in \mathbb{R}^6$ denotes the linear and angular acceleration vector of the root-link in the global system of reference;
- $\ddot{\mathbf{q}} \in \mathbb{R}^{N_J}$ and $\tau \in \mathbb{R}^{N_J}$ are the joint accelerations and torques vectors, respectively;
- $\mathbf{b}_w \in \mathbb{R}^6$ and $\mathbf{b}_c \in \mathbb{R}^{N_J}$ are the bias force vectors describing centrifugal, Coriolis, and gravity forces of the root-link and of the chain segments, respectively;
- N_c is the number of contact points;
- $\mathbf{J}_{w_k}$ and $\mathbf{J}_{c_k}$ are the Jacobian matrices expressed at contact point k that map external wrenches $\mathbf{F}_k = \begin{bmatrix}F_{X_k} & F_{Y_k} & F_{Z_k} & M_{X_k} & M_{Y_k} & M_{Z_k}\end{bmatrix}^T$ to the base-link and chains segments, respectively.

The drive chain model between the motor torques τ_{mj} and the joint torque τ_j is as follows:

$$\begin{aligned} \tau_{mj} &= I_{mj}\ddot{q}_j + H_{ccj}\ddot{q}_j + b_{cj} + \tau_{fmj} + \tau_{fj} \\ \tau_{mj} &= g_{\tau j} I_j \\ \tau_{mj} - \tau_j &= I_{mj}\ddot{q}_j + \tau_{fmj} \\ \text{with : } \tau_{fmj} &= F_{vm}\dot{q}_j + F_{sm}\text{sign}(\dot{q}_j) + off_{mj} \\ \tau_{fj} &= F_{vj}\dot{q}_j + F_{sj}\text{sign}(\dot{q}_j) + off_j \end{aligned} \tag{3}$$

Using the difference between τ_{mj} and τ_j, it is possible to identify the drive chain parameters specifically.

B. Identification models

Both kinematic and dynamic models can be approximated in linear form to relate measurements and parameters that need to be identified using their regressor matrix. In the case of the geometric parameters, this can be achieved by using the first-order Taylor development of Eq. 1:

$$\Delta\mathbf{P} = \mathbf{R}_g(\mathbf{q}, \mathbf{p})\mathbf{\Phi}_g \tag{4}$$

where $\mathbf{R}_g \in \mathbb{R}^{N_m \times N_p}$ is the geometric regressor matrix corresponding to the Jacobian matrix relating the N_p parameters to be identified with the N_m measurements.

The dynamic regressor $\mathbf{R}_d \in \mathbb{R}^{N_m \times N_p}$ is directly obtained thanks to the fact that Eq. 2 and Eq. 3 are linear with respect to the inertial parameters:

$$\mathbf{D} = \mathbf{R}_d(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{\Phi}_d \tag{5}$$

where $\mathbf{D} \in \mathbb{R}^{N_m}$ represents any dynamometric measurements.

The regressors are rank-deficient as some parameters cannot be estimated separately in an LS sense. For the geometric calibration, the maximum number of identifiable parameters is at most 4 out of 6 geometric parameters for a revolute joint, and 2 out 6 geometric parameters for a prismatic joint [20]. Furthermore, the identifiability of these parameters is influenced by the placement of joints in the kinematic tree. For example, if two consecutive joints are co-linear, it is impossible for both joint axis offsets parameters to be identified separately as they are linearly dependent [20]. In this case, a so-called Base Parameter (BP) is created as a linear combination of the two or more co-linear parameters.

Similarly, for dynamic identification, if two segments are linked by a single hinge following z axis, the position of the centers of mass along x and y axes cannot be dissociated. Thus, a BP is created as a linear combination of the centers of mass along x and y axes.

To cope with these dependency issues, it is necessary to ensure that the regressor matrix is full rank. To do so, there are two common approaches, namely symbolic and numeric calculation, to correct for rank-deficient matrix by identifying a set of BPs [15]. The calculation of the BP involves finding the equivalent regressor $\mathbf{R}_b \in \mathbb{R}^{N_m \times N_B}$ that is a full column rank matrix by combining the columns that are not linearly independent. This results in the elimination and regrouping of the parameters to form the vector of base parameters $\mathbf{\Phi}_b \in \mathbb{R}^{N_B}$.

In FIGAROH, the numerical approach based on QR decomposition is chosen for its robustness and ability to easily sort parameters, and thus determine linear expression of grouped parameters [21]. These expressions obtained through this approach can be used to verify the physical meaning of the identification model and understand parameter identifiability. The numerical approach is also well-suited when different static or dynamic BP sets need to be selected. For example, when working in static, the velocities and accelerations are set to 0, therefore the obtained BP are solely functions of the masses and of the center of masses. The remaining static parameters are then regrouped depending on the kinematics of the system. Static $\mathbf{R}_b^S$ and dynamic $\mathbf{R}_b^D$ sub-regressor matrices can be created from the BP for the mass, center of mass and inertia tensor, respectively. As listed in Table I, FIGAROH builds the identification models depending on the available measurements to identify all or some of the parameters [22].

TABLE I

LIST OF PARAMETERS THAT CAN BE IDENTIFIED FROM MEASUREMENTS.

Measurements	Identifiable parameters
Joint torques	SIP and joint parameters
External wrench	SIP
Joint torques, external wrench	SIP, joint parameters
Joint torques, motor current	SIP, joint and drive chain parameters
External wrench, motor current	SIP and drive chain
Full pose	Kinematic parameters
Position	Kinematic parameters
Orientation	Joint angle offsets
Kinematics, joint torques	Kinematic parameters, joint elasticity

C. Parameters determination

Numerous least-square methods exist in the literature, which can be used to solve the parameter estimation problem. FIGAROH re-implements or uses the following approaches:

- Ordinary Least-Square (OLS).
- Weighted Least-Square (WLS).
- Total Least-Square (TLS) [23].
- Quadratic Constrained Programming (QCP) [16].
- Iterative Least-Square (ILS) [15].
- Levenberg-Marquardt (LM) from SciPy [24].

OLS can be used to solve the following problem with $\mathbf{Y} \in \mathbb{R}^{N_m}$ referring to the measurement vector:

$$\mathbf{\Phi}_b^* = (\mathbf{R}_b^T\mathbf{R}_b)^{-1}\mathbf{R}_b^T\mathbf{Y} \tag{6}$$

When dealing with elements of $\mathbf{Y}$ with different units and order of magnitude, it is preferable to use a WLS method:

$$\mathbf{\Phi}_b^* = (\mathbf{R}_b^T\mathbf{W}\mathbf{R}_b)^{-1}\mathbf{Y}\mathbf{R}_b^T\mathbf{Y} \tag{7}$$

where $\mathbf{W}$ is a weight matrix based on the calculation of the relative standard deviation of the identified parameters for each element of $\mathbf{Y}$ [1], [22].

For dynamic identification, if the dynamometric sensor such as the joint torque sensors or a force-plate needs to

be calibrated and/or the drive chain parameters need to be identified, then a TLS approach [23], [1] can be used. It will require to provide data of two experiments, one with the nominal system and one with an additionnal mass. The data is then gathered into unloaded $\mathbf{R}_{du}$ and loaded $\mathbf{R}_{dl}$ regressor matrices and Eq. 3 is reformulated as:

$$\begin{bmatrix} -\mathbf{R}_{du} & i_m & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{R}_{du} & \mathbf{0} & \tau_j & \mathbf{0} & \mathbf{0} \\ -\mathbf{R}_{dl} & i_m & \mathbf{0} & -\mathbf{R}_{dup} & -\mathbf{R}_{dkp} \\ -\mathbf{R}_{dl} & \mathbf{0} & \tau_j & -\mathbf{R}_{dup} & -\mathbf{R}_{dkp} \end{bmatrix} \begin{bmatrix} \phi \\ g_\tau \\ K_j \\ \phi_{up} \\ \phi_{kp} \end{bmatrix} = 0 \quad (8)$$

$$\mathbf{R}_{dtot}\mathbf{\Phi}_{tot} = 0 \quad (9)$$

where $\mathbf{R}_{dup}$ and $\mathbf{R}_{dkp}$ are the observation matrices corresponding respectively to the unknown and known payload inertial parameters.

As noted by Gautier et al. [22], $\mathbf{R}_{dtot}$ is a full rank matrix because of the measurement perturbations. Therefore, the system described in Eq. 9 is modified to the closest compatible one with respect to the Frobenius norm:

$$\hat{\mathbf{R}}_{dtot}\hat{\mathbf{\Phi}}_{tot} = 0 \quad (10)$$

where $\hat{\mathbf{R}}_{dtot}$ is the closest rank deficient matrix from $\mathbf{R}_{dtot}$ and is calculated using the singular value decomposition of $\mathbf{R}_{dtot} = USV^T$.

The solution $\hat{\mathbf{\Phi}}_{tot}$ of Eq. 10 is given by the last column of V. However, V must be scaled. Thus, it is required to use a known mass value and the corresponding measurements in the TLS to scale the solution vector $\hat{\mathbf{\Phi}}_{tot}$ [22].

The QCP algorithm allows for the separate retrieval of the value of each individual parameter using physical consistency constraints at contrary to the BP. The following QCP problem aims to determine the inertial parameters that are physically plausible while fitting the dynamometric measurements and not deviating too much from their nominal value:

$$\begin{aligned} \text{Find} \quad & \mathbf{\Phi}^* \text{ solution of } \min_{\mathbf{\Phi}} \|\mathbf{D} - \mathbf{R}_d\mathbf{\Phi}\|^2 + \alpha\|\mathbf{\Phi}_r - \mathbf{\Phi}\|^2 \\ \text{st.} \quad & M_j > 0 \\ & \sum_{j=1}^{N_L} M_j = M_{tot} \\ & 0 \leq \mathbf{MS_j} - M_j\mathbf{CoM_j}^- \\ & \mathbf{MS_j} - M_j\mathbf{CoM_j}^+ \leq 0 \\ & \text{for } v \neq 0, v^T\mathbf{TI}_j v > \epsilon \end{aligned} \quad (11)$$

where $\mathbf{\Phi}_r$ is the nominal vector of inertial parameters either given by anthropomorphic tables for human [2] or by CAD data for robot. M_{tot} is the total system mass that is easily identifiable. $\mathbf{CoM_j}^+$ and $\mathbf{CoM_j}^-$ are respectively the 3D upper and lower boundaries on the 3D position of the center of mass for link j. $v \in \mathbb{R}^{2000\times 3}$ is a set of 2000 non zero vectors uniformly distributed over the unit sphere used to constrain the inertia matrix to be positive defined [16], $\epsilon = 10^-3$ is a tolerance parameter.

For geometrical calibration, the kinematic equations are non-linear. The two common approaches to solve the LS problem are linearizing the kinematic model and solving Eq. 4 using an ILS method or using the LM algorithm. ILS updates the parameters based on the gradient of the error function until the estimation converges to a stable solution with a specified tolerance. The LM algorithm combines features of the Gauss-Newton algorithm and of the gradient descent method. The solution of parameters is found by iteratively updating the estimates of parameters via the dynamic adjustment of two damping coefficients that controls the influence of the two methods until convergence is reached.

Regardless of the LS method chosen, the relative standard deviation $\sigma_\%$ [1], [22] of identified parameters can be used to give an image of the accuracy of the identified base parameter values.

D. Generation of optimal postures and motions

Extensive studies on criterion and methods to generate OEP and OEM in order to avoid an ill-conditioned regressor matrix are provided in robotics literature [13]. FIGAROH uses some of the most popular criteria from the literature. In the following, $\Psi()$ denotes one of the criteria defined in Table II. $[\sigma_1, \sigma_2, ..., \sigma_{N_p}]$ are the singular values of the regressor matrix.

TABLE II
EXCITATION CRITERIA USED FOR OEP AND OEM GENERATION [4] .

Index	Expression
O_1	$\frac{\sqrt[N_p]{\sigma_1\sigma_2...\sigma_{N_p}}}{\sqrt{N_m}}$
O_2	$\frac{\sigma_{max}}{\sigma_{min}}$
O_3	σ_{min}

However, setting up an optimization problem to generate dataset of hundreds of samples while ensuring the respect of mechanical constraints and avoiding convergence issues requires expert skills. FIGAROH proposes to automatically generate this dataset from the robot, measurement types and environment description. To reduce calculation time and avoid convergence issues, several smaller optimization problems are solved iteratively. If the system has a tree-structured kinematics, separate optimization problems are solved for each kinematic chain.

The first step is to solve the optimization problem aimed at determining the postures that excite the static parameters. Although the implementation of the method for generating OEP for geometric calibration and for the identification of mass and center of mass are conceptually similar, they are different due to the availability of measurements. Geometric calibration has fewer measurements for a given posture

compared to dynamic identification, which typically relies on joint torque measurements. For geometric calibration, a recent algorithm [25], based on evaluating and ranking the informativeness of each posture available in a large pool of N feasible candidate postures, is re-implemented in FIGAROH. Determining the most informative postures is often regarded as solving a combinatorial optimization problem in the experimental design field [26]. FIGAROH proposes to transform this combinatorial problem in a continuous one by assigning for each candidate posture i, an information matrix $\mathbf{\Sigma_i}$ and a weight ω_i. The algorithm seeks the best weighting vector $\boldsymbol{\omega} \in \mathbb{R}^N$ to optimize an excitation criterion as follows:

$$\begin{aligned} \text{Find } \boldsymbol{\omega}^* \quad & \text{solution of } \max_{\boldsymbol{\omega}} \ \Psi(\sum_{i=1}^{N} \omega_i \mathbf{\Sigma_i})) \\ \text{st.} \quad & \sum_{i=1}^{N} \omega_i = 1 \end{aligned} \tag{12}$$

Finally, after a simple ranking of the weights in $\boldsymbol{\omega}$, it is possible to determine the minimal number of required OEP $\mathbf{q}_S^*$. In fact, the criterion value will augment with the number of postures until it reaches a plateau or a peak, depending of the retained cost function, that is easily detectable. Fig. 2 exemplifies the evolution of O_1 criterion when postures are classified using Eq. 12. A clear peak is observable at 40 exciting postures. This is a real novelty as the number of calibration postures is usually set prior to the calibration process by the user.

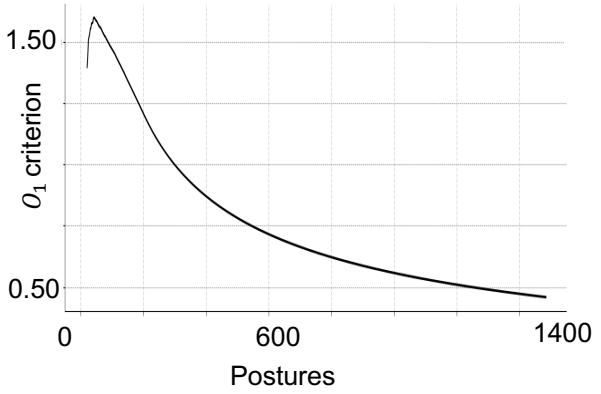

Fig. 2. Representative evolution of the criterion 0_1 when classifying exciting postures using Eq. 12.

OEP related to the mass and center of mass are determined iteratively by block of N_p postures that ensure mechanical constraints. N_p is set as the minimal number of postures required to have at least a determined system of equations. The first N_p OEP are determined thanks to Eq. 13. Then, the following N_p OEP take into account the previous postures by including them in the optimization process through the stacking of the static BP regressor (increasing the number of rows). N_p OEP are iteratively calculated and $\mathbf{R}_b^S$ is augmented until variation of the criterion is below a given threshold [19], [27].

$$\begin{aligned} \text{Find } \mathbf{q}_S^* \quad & \text{solution of } \min_{\mathbf{qs}} \ \Psi(\mathbf{R}_b^S) \\ \text{st.} \quad & q_j^- \leq q_{jS} \leq q_j^+ \\ & |\tau_j| \leq \tau_j^+ \\ & \mathbf{P}_E^- \leq \mathbf{P}_E(\mathbf{q}) \leq \mathbf{P}_E^+ \\ & 0 \leq \mathbf{d} \end{aligned} \tag{13}$$

where q_i^- and q_i^+ are the joint boundaries, τ_j^+ is the maximal joint torque. $\mathbf{P}_E^-$ and $\mathbf{P}_E^+$ constrain the end-effector position and orientation to remain in a subset of their workspace to avoid collision with the environment. $\mathbf{d}$ is the vector of distances between the different vertices of robot that should be kept positive to avoid auto-collision. It is calculated using hpp-fcl library [17].

Once feasible OEP are obtained, a new optimization process is used to determine the OEM $\mathbf{q}_D \in \mathbb{R}^{N_J \times N_s}$ that excite the inertia parameters. N_s is the arbitrarily set number of time samples between two consecutive OEP p and $p+1$. Eq. 14 is solved separately between each consecutive OEP, and previous calculated trajectories are taken into account in the optimization process by stacking the dynamic BP regressor $\mathbf{R_b^D}$. Cubic spline functions $CS()$ [28] are used to interpolate the trajectory. Thus, the problem of determining OEM boils down to determining the $\mathbf{N_w} \in \mathbb{R}^{N_J \times N_k}$ waypoints of cubic spline functions, with $N_k = 5$ set by default, as follows [19]:

$$\begin{aligned} \text{Find } \mathbf{q}_D^* \quad & \text{solution of } \min_{\mathbf{N_w}} \ \Psi(\mathbf{R}_b^D) \\ \text{st.} \quad & CS(0, q_{jD}) = \mathbf{q}_{Sj}^*(p) \\ & CS(N_s t_s, q_{jD}) = \mathbf{q}_{Sj}^*(p+1) \\ & \dot{CS}(0, q_{jD}) = \dot{CS}(N_s t_s, q_{jD}) = 0 \\ & q_j^- \leq CS(k t_s, q_{jD}) \leq q_j^+ \\ & \left|\dot{CS}(k t_s, q_{jD})\right| \leq \dot{q}_j^+ \\ & |\tau_j| \leq \tau_j^+ \\ & \mathbf{P}_E^- \leq \mathbf{P}_E(\mathbf{q}) \leq \mathbf{P}_E^+ \\ & 0 \leq \mathbf{d} \\ & \text{with} \quad k = 1, ..., (N_s - 1) \end{aligned} \tag{14}$$

where t_s is the sampling time.

Even though using several iterative optimization processes is sub-optimal, it has the advantage of considerably reducing calculation time and convergence issues, while globally converging toward a global minimum [27], [19]. All the optimization processes are implemented using IPOPT solver, and numerical gradients are used as analytical gradients of the cost functions listed in Table II are not obtainable.

III. TOOLBOX DETAILS

The toolbox FIGAROH is made open source and now accessible at https://gitlab.laas.fr/gepetto/figaroh. As described in Fig. 1, the main functional tools of FIGAROH are located in `/src/figaroh` which can be installed from the source code, and it includes a `models` folder containing 3D and URDF models of considered systems and an `examples` folder containing projects in separate sub-folders.

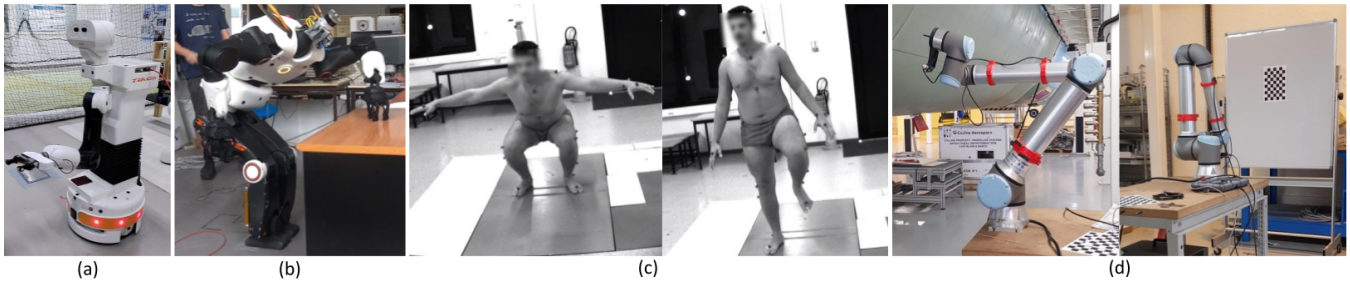

Fig. 3. Experimental setup of the datasets used in FIGAROH. (a) TIAGo geometric calibration using a SS, (b) TALOS geometric calibration using a 3 points planar contact, (c) human dynamic identification, (d) UR10 geometric calibration using an embedded camera and a chessboard.

A. Project configuration and data preparation

1) Project configuration: Each project requires a project configuration file in `yaml` file format that specifies the requirements for the calibration or identification procedure. For geometric calibration, the required configurations should contain the level of calibration (either only joint offset or full-set kinematic parameters), as well as the inclusion of non-geometric parameters (such as joint elasticity) in the geometric model. The starting and ending frames of calibrated kinematic chains, the usage of a floating base, and the parent joint frame closest to where measurements are taken, should also be declared in this configuration file. Additionally, a list of 6 boolean elements can be set to indicate the location and number of observable Degrees of Freedom (DoF) of the measurements. An example of such a file is provided here. For dynamic identification, notional values of additional parameters for the considered robot that are not specified in URDF file should be added in `robot_params` field such as friction parameters, actuator gear ratio. In `problem_params` field, several options of dynamic effects, such as joint frictions, actuator inertias, joint effort offsets can be chosen to be included in the dynamic model. Furthermore, the type of effort measurements (joint torque sensor or external wrench measurements), must also be specified here. Lastly, data pre-processing and LS method-related parameters can be specified.

2) Data preparation: The input data for geometric calibration consists of a set of static postures obtained from encoder readings, along with the corresponding complete or partial pose measurements. These measurements can be obtained either from external sensors or from geometrically constrained postures. The user can provide the static postures, or FIGAROH can generate OEP launching a dedicated script `optimal_config.py`.

Similarly, exciting motions for dynamic identification can be provided by the user or automatically generated by FIGAROH from the script `optimal_trajectory.py`. For dynamic identification, if only joint configurations are made available, joint velocities and accelerations are numerically calculated, which may introduce some noise. FIGAROH has re-implemented all the numerical recipes spread across numerous reference papers related to dynamic identification published by Gautier et al. [21], [1], [22], [29]. For example, dedicated Butterworth filtering and decimating methods or automatic removal of near-zero velocity data for friction coefficients can be applied using the functions implemented in `low_pass_filter_data`.

B. Parameters identification

After gathering necessary input data and defining project configurations, both geometric calibration and dynamic identification procedures share a similar scheme including 3 main steps:

- 1/ Load robot model and initialize a config parameter dictionary object: The class `Robot()` inherits all attributes and functions from `robot` object in Pinocchio library. It plays an essential role throughout the procedure in storing data and using computing algorithms provided by Pinocchio such as `rnea`, `computeJointTorqueRegressor`, `forwardKinematics`, `computeFrameKinematicRegressor` [17]. A function `get_param_from_yaml` retrieves information in project config file and creates a dictionary that can be accessed at any point of the procedure.
- 2/ Determine the BP and construct corresponding base regressor matrix automatically: Expressions of BP and column-specified structure of BP can be numerically identified using `regressor` and `qrdecomposition` tools, (see section II.B).
- 3/ Perform the estimation with one of the LS techniques available in FIGAROH (see Section II.C).

More details of these steps are shown in the following section with usecases.

C. Results and analysis

After determining the identified parameters, statistical analysis such as relative standard deviation [30] of identified parameters and Root-Mean-Square Errors (RMSE) are computed for statistical evaluations and visualization. These evaluations are critical to assess the quality of parameter identification. Besides, if cross validation is requested, additional calculation and plotting are also provided. Finally, the identified parameters can separately be saved into a `xacro` or `yaml` file which can be used to update the robot model.

IV. USECASES WITH DATASET

A. Human model dynamic identification

Dataset for human calibration and identification was collected with one healthy subject (male, 26 years old, 97kg) using a 20 cameras SS and a forceplate (Vero v2.2 Vicon , AMTI OR6 Series) sampled at 100Hz. The popular plug-in gait-template [31] based on 35 markers was used to estimate human motion. Human geometric calibration uses markers positions gathered from a static "T-pose". Using the Vicon chord function [31] re-implemented in FIGAROH, joint center positions were computed from the marker measurements and fitted for static calibration purposes. After human geometric calibration was performed, inverse kinematics of a 43 DoF human model was performed. This model follows the joint axes definition proposed by the International Society of Biomechanics [32], [33]. The nominal inertial parameters were based on an AT [2]. For dynamic identification, the subject was asked to perform several movements such as squats, or arms and legs rapid flexion/extensions as shown in Fig. 3.c. Fig. 4 shows a comparison between the measured external wrench and its estimate when using a model based on the AT and the identified one. The corresponding averaged RMSE were 19.47N and 33.22N.m for the model based on AT and 16.84N and 16.77N.m for the identified model.

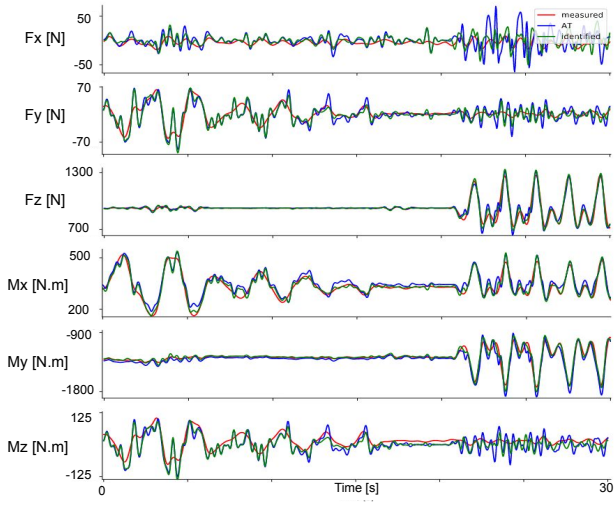

Fig. 4. Human external wrench comparison between measurements (red) and its estimation using AT model (blue) and the identified model (green).

B. Floating base robots

1) TALOS and TIAGo geometric calibration: Floating base robots have a large workspace, and the motion of their base should be monitored. Thus, a SS appears as a good choice for their geometric calibration [27]. The experimental setup for the torso-arm chain geometric calibration of TALOS humanoid robot and of TIAGo mobile collaborative arm can be seen in Fig. 3.a. Two clusters of three markers attached to the base and to the end-effector were monitored using a 20 cameras SS (Miqus M3 cameras, Qualisys). The measurements included synchronized robot joint configurations and absolute marker coordinates taken at designed static OEP. FIGAROH then calculated numerically BP and identified geometric parameters. Using cross validation postures shown in Fig. 5, when comparing to the absolute position measurement of the end-effector, the calibrated models showed an RMSE of 0.3mm for TALOS torso-arm chain and of 1.5mm for TIAGo arm. It was 14.1mm and 16.9mm for the uncalibrated models of the TALOS torso-arm chain and of the TIAGo arm, respectively.

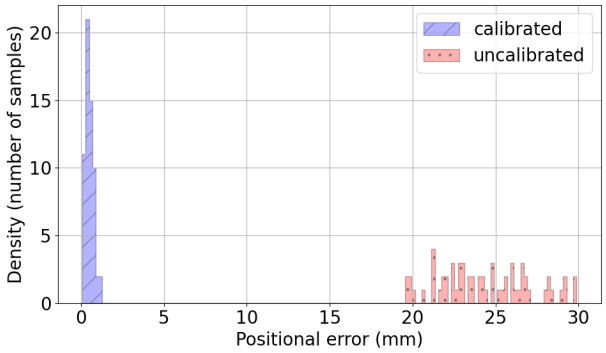

Fig. 5. Residual errors of the absolute position of end-effector before and after calibration of the right arm and torso of TALOS with motion capture.

2) TALOS whole-body calibration using planar constraints: Fig. 3.b shows how geometric parameters of the TALOS humanoid robot can be used using 3-point contacts with a single plane. The TALOS was modeled with two kinematic chains of 15 DoF from foot to hand on the same side. Measurements of joint configurations were recorded for 31 OEP, which were automatically generated using FIGAROH to maximize the O_1 criterion. During cross validation postures, the RMSE was reduced from 10.4mm to 3.3mm when using the calibrated model.

3) TIAGo torso-arm dynamic identification: This dataset presents the dynamic identification, including drive chain parameters of the TIAGo robot and its differential transmission at the wrist level. It is composed of the joint configurations and motor currents collected during OEM that were calculated by FIGAROH as described in section II.D. The identified model was capable of estimating measured motor currents with only 1% of difference.

C. Serial manipulators

1) Stäubli TX40 dynamic identification: The Stäubli TX40 is a 6 DoF industrial serial manipulator. Thanks to Gautier et al. [1], FIGAROH dataset includes the previously published data used for the dynamic identification of the TX40. Exciting motions and motor currents are provided. Using the same input data FIGAROH builds the identification model, reproduces the data elaboration pipeline, implements a WLS and finally identifies successfully the published reference inertial parameters including drive chain ones. Moreover, the so-called essential parameters [30], [1] that are a subset of the actually identifiable BP, ie with a

relative standard deviation inferior to 5%, are also retrieved automatically by FIGAROH.

2) Universal Robots UR10: Eye-in-hand calibration was performed with a UR10 robot. As shown in Fig. 3.d, a camera looking at a calibration chess board was located to the gripper. The generation of OEP and transition motions between two postures were done thanks to HPP software. The camera was calibrated before carrying out the calibration experimentation. The full-set of kinematic parameters was identified. The absolute position of end-effector was then estimated with a RMSE of 1.76mm.

V. CONCLUSIONS

FIGAROH is an open-source toolbox for unified geometric calibration and dynamic identification of robots and humans which is made accessible to non-experts with state-of-the-art methods. Besides making modeling and data processing simple, one of the key assets of FIGAROH is its ability to automatically generate OEP and OEM. FIGAROH functionalities were demonstrated over several dataset including serial manipulators and anthropomorphic structures.

Future works will include kinodynamic identification, i.e the identification of geometric and dynamic parameters in one integrated procedure, and modelling of non linear flexibilities that have preponderant influence in light weight collaborative robots.

REFERENCES

[1] M. Gautier and S. Briot, "Global identification of drive gains parameters of robots using a known payload," in *2012 IEEE Int. Conf. Robot. Autom.*, 2012, pp. 2812–2817.

[2] R. Dumas, L. Chèze, and J. P. Verriest, "Adjustments to McConville et al. and Young et al. body segment inertial parameters," *J Biomech*, vol. 40, pp. 543–553, 2007.

[3] J. Hollerbach, W. Khalil, and M. Gautier, *Model Identification*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 321–344.

[4] Y. Sun and J. M. Hollerbach, "Observability index selection for robot calibration," in *2008 IEEE Int. Conf. Robot. Autom.*, 2008, pp. 831–836.

[5] G. Venture, K. Ayusawa, and Y. Nakamura, "Motion capture based identification of the human body inertial parameters," in *2008 30th Annu Int Conf IEEE Eng Med Biol Soc*, 2008, pp. 4575–4578.

[6] N. Nadeau, "Pybotics: Python toolbox for robotics," *J. Open Source Softw.*, vol. 4, p. 1738, Sep. 2019.

[7] J. Rozlivek, L. Rustler, K. Stepanova, and M. Hoffmann, "Multisensorial robot calibration framework and toolbox," in *2020 IEEE-RAS 20th Int. J. Humanoid Robot. (Humanoids)*, 2021, pp. 459–466.

[8] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, "Inertial parameter identification in robotics: A survey," *Appl. Sci.*, vol. 11, 2021.

[9] M. Ferguson, "Robot calibration," https://github.com/mikeferguson/robot_calibration, 2022.

[10] A. Joubair and I. A. Bonev, "Comparison of the efficiency of five observability indices for robot calibration," *Mech Mach Theory*, vol. 70, pp. 254–265, 2013.

[11] M. Ikits and J. Hollerbach, "Kinematic calibration using a plane constraint," in *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 4, 1997, pp. 3191–3196 vol.4.

[12] J. Baelemans, P. van Zutven, and H. Nijmeijer, "Model parameter estimation of humanoid robots using static contact force measurements," in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2013, pp. 1–6.

[13] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robot. Comput.-Integr. Manuf.*, vol. 31, pp. 21–29, 2015.

[14] J. Swevers, C. Ganseman, D. B. Tukel, J. de Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE Transactions on Robotics and Automation*, vol. 13, pp. 730–740, 1997.

[15] W. Khalil and E. Dombre, *Modeling, identification & control of robots*. London : Hermes Penton Ltd., 2002.

[16] J. Jovic, A. Escande, K. Ayusawa, E. Yoshida, A. Kheddar, and G. Venture, "Humanoid and human inertia parameter identification using hierarchical optimization," *IEEE Trans. Robot.*, vol. 32, pp. 726–735, 2016.

[17] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *IEEE International Symposium on System Integrations (SII)*, 2019.

[18] Y. Fujimoto and A. Kawamura, "Robust biped walking with active interaction control between robot and environment," in *Proceedings of 4th IEEE International Workshop on Advanced Motion Control - AMC '96 - MIE*, vol. 1, 1996, pp. 247–252 vol.1.

[19] V. Bonnet, P. Fraisse, A. Crosnier, M. Gautier, A. González, and G. Venture, "Optimal exciting dance for identifying inertial parameters of an anthropomorphic structure," *IEEE Trans. Robot.*, vol. 32, pp. 823–836, 2016.

[20] L. J. Everett and T.-W. Hsu, "The Theory of Kinematic Parameter Identification for Industrial Robots," *J. Dyn. Sys., Meas., Control.*, vol. 110, pp. 96–100, 03 1988.

[21] M. Gautier, "Numerical calculation of the base inertial parameters of robots," in *Proceedings., IEEE Int. Conf. Robot. Autom.*, 1990, pp. 1020–1025 vol.2.

[22] M. Gautier and A. Jubien, "Force calibration of kuka lwr-like robots including embedded joint torque sensors and robot structure," in *2014 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 416–421.

[23] M. Gautier, P. Vandanjon, and C. Presse, "Identification of inertial and drive gain parameters of robots," in *Proc. IEEE Conf. Decis. Control*, vol. 4, 1994, pp. 3764–3769 vol.4.

[24] J. J. Moré, "The levenberg-marquardt algorithm: Implementation and theory," in *Numerical Analysis*, G. A. Watson, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1978, pp. 105–116.

[25] K. Kamali and I. A. Bonev, "Optimal experiment design for elasto-geometrical calibration of industrial robots," *IEEE/ASME Trans. Mechatron.*, vol. 24, pp. 2733–2744, 2019.

[26] D. Daney, Y. Papegay, and B. Madeline, "Choosing measurement poses for robot calibration with the local convergencemethod and tabu search," *Int. J. Rob. Res.*, vol. 24, pp. 501–518, 2005.

[27] V. Bonnet, J. Mirabel, D. Daney, F. Lamiraux, M. Gautier, and O. Stasse, "Practical whole-body elasto-geometric calibration of a humanoid robot: Application to the talos robot," *Robot. Auton. Syst.*, p. 104365, 2023.

[28] S. Tonneau, J. Chemin, P. Fernbach, and G. Saurel, "ndcurves," 2013. [Online]. Available: https://github.com/loco-3d/ndcurves

[29] W. Khalil, M. Gautier, and C. Enguehard, "Identifiable parameters and optimum configurations for robots calibration," *Robotica*, vol. 9, p. 63–70, 1991.

[30] C. M. Pham and M. Gautier, "Essential parameters of robots," in *Proc. IEEE Conf. Decis. Control*, 1991, pp. 2769–2774 vol.3.

[31] M. Fellinger, J. Passler, and W. Seggl, "Plug-in Gait Reference Guide," *Human and Nonhuman Bone Identification*, pp. 227–246, 2010.

[32] G. Wu, S. Siegler, P. Allard, C. Kirtley, A. Leardini, D. Rosenbaum, M. Whittle, D. D. D'Lima, L. Cristofolini, H. Witte, O. Schmid, and I. Stokes, "Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion—part i: ankle, hip, and spine," *J. Biomechs*, pp. 543–548, 2002.

[33] G. Wu, F. C. van der Helm, H. (DirkJan) Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, F. W. Werner, and B. Buchholz, "Isb recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion—part ii: shoulder, elbow, wrist and hand," *J. Biomechs.*, pp. 981–992, 2005.