Parameter Identification of a 6-DoF Serial Manipulator with Coupled Joints and Load-Assisting Springs for Industrial Applications

Matteo Nini
DISMI
UNIMORE, Reggio Emilia, Italia
matteo.nini@unimore.it

Federica Ferraguti
DISMI
UNIMORE, Reggio Emilia, Italia
federica.ferraguti@unimore.it

Matteo Ragaglia
Gaiotto Automation
SACMI Group, Piacenza, Italy
matteo.ragaglia@sacmigroup.com

Mattia Bertuletti
Gaiotto Automation
SACMI Group, Piacenza, Italy
mattia.bertuletti@sacmigroup.com

Simone Di Napoli
Gaiotto Automation
SACMI Group, Piacenza, Italy
simone.dinapoli@sacmigroup.com

Cesare Fantuzzi
DISMI
UNIMORE, Reggio Emilia, Italy
cesare.fantuzzi@unimore.it

***Abstract*—This paper presents a novel approach for identifying the dynamic parameters of a 6 DoF serial manipulator characterized by coupling and springs, which is a common mechanics for industrial robots. The proposed method consists of two steps: at first, a static identification process for estimating the masses and centers of gravity (CoGs) of the links is performed; then, a dynamic identification process for determining the inertias, motor inertias, and frictions is executed. In the dynamic identification process, a trajectory is used to generate the required dynamic response of the system, and a regression matrix is employed to combine the identified parameters. Finally, a constrained optimization method is utilized to extract the parameters. The proposed method has been validated through simulations and experiments, showing high accuracy and reliability. This research contributes to the advancement of robot modeling and control, and has potential applications in various industrial fields.**

***Index Terms*—dynamic parameter identification, serial manipulator, constrained optimization, industrial robots**

I. INTRODUCTION

Robotic manipulators are widely used across various industries, including manufacturing, healthcare, and automation. Accurate dynamic modeling and parameter identification are essential for effective simulation, control, and teleoperation of these systems [1]. However, obtaining reliable dynamic parameters is challenging due to factors such as system noise, unmodeled dynamics, and limited manufacturer data [2]. This paper addresses these challenges by presenting a novel approach for dynamic parameter identification in a robotic manipulator characterized by coupled joints and load-assisting springs. Our approach integrates static and dynamic trajectories with constrained optimization methods to improve parameter estimation accuracy.

Typically, the process of identifying a robot involves several key steps: creating a model, planning experiments, gathering data, processing signals, estimating parameters, and confirming the model's accuracy [3]. For the initial step, one must decide between two modeling options. The first option is the Euler-Lagrange (E-L) robotic model, which has garnered considerable attention from numerous researchers. For instance, Gaz et al. [4] employed this model to extract feasible parameters using penalty-based optimization methods. Similarly, Jubien et al. [5] used the E-L model to compare the actual and confidential parameters of a Kuka robot, and Liu et al. [6] applied it to enhance the identification method for redundant manipulators.
The second option is the Newton-Euler (NE) model, chosen by Atkeson et al. [7] for estimating the inertial parameters of robot links, and also by this research. The preference for the NE model is due to its iterative nature, which facilitates fast computation, especially beneficial when dealing with complex manipulators. Additionally, this method simplifies the formulation of the regression matrix, which is crucial for the identification phase.

The literature presents a crucial distinction in the methods used for parameter identification, showcasing a diverse range of estimation techniques [8]. Traditional least-squares methods are a popular choice, as demonstrated by Gautier et al. [9], who employed both direct and inverse dynamic identification models to derive parameters from torques, and by Jia et al. [10], who refined these methods using proprioception data. The Inverse Dynamic Identification Model with Least-Squares (IDIM-OLS) [11] and its variants, including the Weighted Least-Squares (IDIM-WLS) and Total Least-Squares (IDIM-TLS) methods [12], while effective, encounter challenges such as noise sensitivity and dependency on the trajectory's conditioning, as shown in [13] and [14]. These issues can lead to biased estimates, including negative masses. Alternative approaches have been investigated, such as the integration of the cuckoo search algorithm with the least-squares method by Ding et al. [15] for dynamic parameter estimation, and the application of Lie theory-based identification by Fu et al. [16] to formulate the dynamic model as a closed-form ma-

Copyright notice: 979-8-3503-7636-4/24/$31.00 ©2024 IEEE

trix equation. Recent developments suggest using constrained optimization to ensure physical consistency in parameter estimates, with some approaches reformulating identification as a Semi-Definite Programming problem as illustrated in [17], [18], [19] and [20]. Although Deep Neural Networks have been explored for dynamics learning, as in the study by Su et al. [21], this work prioritizes approaches that offer a more transparent link between parameters and robot dynamics.

The existing literature on dynamic parameter identification of robotic manipulators focuses on industrial robots with uncoupled joints and no additional components like springs [3]. In contrast, our work examines robots with coupled joints and springs, adding complexity to the dynamic identification challenge. Springs introduce external forces converted into torques on the robot's joints, influenced by the robot's position and requiring accurate accounting during identification. Joint coupling further complicates the structure, such as using a closed-kinematic chain to redistribute the robot's weight [22].

Identifying dynamic coefficients typically suffices for robot control and simulation tasks. However, deriving practical numerical values for the original dynamic parameters is crucial for dynamic simulations in CAD-based systems like CoppeliaSim [23], where users must input specific dynamic parameters for each link, and for model-based control strategies relying on the recursive Newton-Euler algorithm [24], which require knowledge of each link's dynamic parameters. Gaz et al. [25] addressed this issue by finding a full set of parameters using nonlinear optimization methods.

We selected the least-squares approach for its simplicity, interpretability, and computational efficiency, and developed a novel regression matrix that effectively captures the dynamics of manipulators with coupled joints and springs. By integrating the Newton-Euler method with least-squares-based parameter identification, our methodology employs a two-step process involving static and dynamic trajectories to estimate dynamic parameters such as masses, centers of gravity (CoG) positions, inertias, motor inertias, and friction coefficients. However, the resulting system of equations is under-determined, leading to infinite solutions. To address this issue, an optimization phase is introduced to identify the optimal set of parameters and minimize errors. Considering the nonlinear nature of the problem, we further impose constraints on the parameters and link masses to eliminate infeasible solutions and ensure physical relevance. Springs are modeled based on the robot's position to introduce additional torque accurately. The influence of coupled joints is integrated into the analysis, recalibrating torques to ensure a refined parameter identification process.

Our method starts with multiple static poses to estimate link masses and CoG positions, laying the groundwork for dynamic parameter estimation. The second phase involves dynamic trajectories to determine inertias, motor inertias, and friction coefficients.

This paper demonstrates the effectiveness of the proposed methodology in accurately identifying the dynamic parameters of robotic manipulators with coupled joints and springs, enhancing the accuracy of dynamic models for real-time control and simulation.

The contributions of this paper are as follows:

- Introduction of a new methodology for identifying dynamic parameters in robotic manipulators with coupled joints and springs, employing a combination of the Newton-Euler model, static and dynamic trajectories, and constrained optimization.
- Addressing a research gap by focusing on a complex and under-studied category of robotic manipulators, and presenting results that validate the approach's effectiveness.
- Experimental validation using the Gaiotto GA-OL robot to demonstrate the proposed methodology's applicability in real-world scenarios.

The remainder of this paper is organized as follows: Section II presents the theoretical foundations of the proposed approach, covering static and dynamic identification along with optimization techniques. Section III provides the experimental validation, including a description of the robot, an ideal case study in MATLAB, and the results using the real robot. Finally, Section IV concludes the paper and outlines potential future research directions.

II. METHODOLOGY

The proposed method exploits the Least Squares (LS) method [26] to identify parameters for a 6-DOF industrial serial manipulator. Grounded in Newton-Euler equations, our methodology comprises two phases, depicted in Fig. 1:

- Static Identification: Utilizes the robot's static configuration to ascertain arm masses and centers of gravity (CoGs).
- Dynamic Identification: Aims to identify motion-related parameters, including inertia and friction.

Data for both phases are collected from experiments and simulations. All identified parameters undergo validation against benchmarks. The identification phase results in a system of equations that intertwines the identifiable parameters. This complexity makes isolating individual parameter values challenging without further analysis. Thus, an under-determined system emerges, necessitating additional steps for resolution. Consequently, an optimization phase was introduced to pinpoint the parameter set that most accurately represents the system model.

A. Robot Information

The method presented herein aims to identify the dynamic parameters of serial manipulators that are characterized by specific features, including load springs and mechanical couplings. These distinctive features necessitate a meticulous and precise parameter identification process to accurately characterize the dynamics of the manipulator.

To achieve this, we first articulate the dynamic parameters in symbolic form. For the purpose of identification, vectors of dynamic parameters are established, denoted as $\pi_{s_i} \in \mathbb{R}^{p_s \times 1}$ for static identification and $\pi_{d_i} \in \mathbb{R}^{p_d \times 1}$ for dynamic identification. Here, i specifies the link number, while p_s and p_d

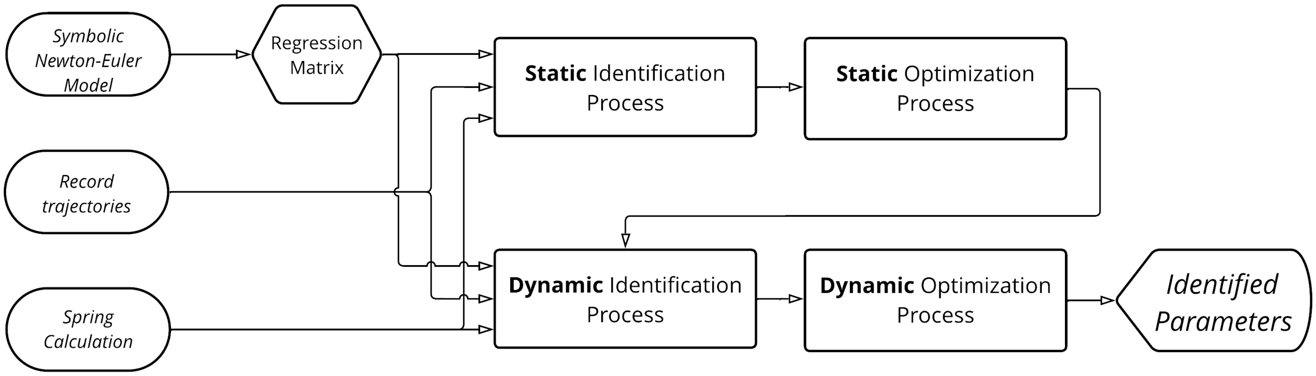


Fig. 1: Parameter Identification Workflow. It explains the general design and depicts the most important steps of the procedure.

represent the number of unknown parameters for each joint during the static and dynamic identification, respectively.

$$\pi_{s_i} = [m_i, m_i l_{ix}, m_i l_{iy}, m_i l_{iz}]^T \tag{1}$$

$$\pi_{d_i} = [I_{ixx}, I_{iyy}, I_{izz}, I_{ixy}, I_{ixz}, I_{iyz}, I_{mi}, F_{si}, F_{vi}]^T \tag{2}$$

Where $m_i \in \mathbb{R}$ represents the mass, (l_{ix}, l_{iy}, l_{iz}) the three components of the CoG, (I_{ixx}, I_{iyy}, I_{izz}, I_{ixy}, I_{ixz}, I_{iyz}) the tensor inertia components of the i-th link; meanwhile $I_{mi} \in \mathbb{R}$ represents the motor inertia, $F_{si} \in \mathbb{R}$ the static friction and $F_{vi} \in \mathbb{R}$ the viscous friction of the i-th joint.

B. Parameter Static Identification

1) Computation of the regression matrix: In order to identify the dynamic parameters, we exploited the Newton-Euler model. The algorithm involves a forward recursion, starting from the base frame and proceeding to the end-effector, and a backward recursion, starting from the end-effector and proceeding to the base. Once the forward recursion values are computed, the backward recursion is executed with modifications from the standard algorithm. Specifically, we aim to obtain the regression matrix $Y \in \mathbb{R}^{n \times n \cdot p_s}$ (where n is the number of the manipulator's degrees of freedom), which, when multiplied by the vector of dynamic parameters $\pi_{s_{tot}} \in \mathbb{R}^{n \cdot p_s \times 1}$, yields the joint torque vector $\tau \in \mathbb{R}^n$. Here, $\pi_{s_{tot}}$ is an aggregation of all the vectors $\pi_{si} \in \mathbb{R}^{p_s}$, each corresponding to an individual joint/link.

$$\tau = Y(q, \dot{q}, \ddot{q})\pi_{s_{tot}} \tag{3}$$

In order to find the regression matrix, the following steps are performed:

- The expressions for force $f_i \in \mathbb{R}^3$ and moment $\mu_i \in \mathbb{R}^{3\times 1}$ are combined into a single column vector $\gamma_i \in \mathbb{R}^6$:

$$\gamma_i = \begin{bmatrix} f_i & \mu_i \end{bmatrix}^T = D_{i+1}\gamma_{i+1} + \Gamma_i \tag{4}$$

  Where $\gamma_{i+1} \in \mathbb{R}^6$ groups the parameters of force and moment which depend on the value at the next, while $\Gamma_i \in \mathbb{R}^2$ contains the remaining parameters. The matrix $D_{i+1} \in \mathbb{R}^{6\times 6}$ is a pseudo-rotation matrix that contains the multiplicative factors and is defined as:

$$D_{i+1} = \begin{bmatrix} R^i_{i+1} & 0 \\ S(\bar{r}^i_{i-1,i})R^i_{i+1} & R^i_{i+1} \end{bmatrix} \tag{5}$$

  where the operator $S(\cdot)$ represents the skew-symmetric matrix, $R^i_{i+1} \in \mathbb{R}^{3\times 3}$ is the rotational matrix between the link $i+1$ and i, and $\bar{r}^i_{i-1,i} \in \mathbb{R}^3$ represents the distance between the $i-th$ and the $(i-1)-th$ reference systems (RFs).
- The actuated torque on each joint can be expressed in terms of γ_i, starting from the definition of torque in the Newton-Euler equations:

$$\tau_i = \begin{bmatrix} 0 & (R^i_{i-1})^T \bar{z}_0 \end{bmatrix} \begin{bmatrix} f^i_i \\ \mu^i_i \end{bmatrix} \tag{6}$$

  Where $\bar{z}_0 \in \mathbb{R}^3$ is the z-axis coordinates of the first RF. The vector that applies this transformation is called the transformation vector:

$$Tr_i = \begin{bmatrix} 0 & (R^i_{i-1})^T \bar{z}_0 \end{bmatrix} \tag{7}$$

- The vector γ_i is composed of two components, $D_{i+1}\gamma_{i+1}$, which is a function of the dynamic parameters, and Γ_i. Γ_i needs to be expressed as a function of the vector π_{si}. To perform this operation, Γ_i is decomposed into the product of a matrix $K_i \in \mathbb{R}^{6\times p_s}$, which is a function of the kinematic quantities calculated with Newton-Euler equations, and the vector π_{si}:

$$\Gamma_i = K_i \pi_{si} = \begin{bmatrix} K_{i11} & K_{i12} \\ K_{i21} & K_{i22} \end{bmatrix} \begin{bmatrix} m_i \\ m_i \bar{r}^i_{i,C_i} \end{bmatrix} \tag{8}$$

  Where $\bar{r}^i_{i,C_i} \in \mathbb{R}^3$ is the distance between the $i-th$ CoG and the $i-th$ RF. All the values of $K_{ijk} \in \mathbb{R}$ can be defined to compose the matrix K_i for each link:

$$K_i = \begin{bmatrix} \ddot{P}_i & 0 \\ S(\bar{r}^i_{i-1,i})\ddot{P}_i & -S(\ddot{P}^i_i) \end{bmatrix} \tag{9}$$

Where $\ddot{P}_i \in \mathbb{R}^3$ represent the linear acceleration of the joint i. Based on these calculations, the vector γ_i can be rewritten as:

$$\gamma_i = D_{i+1}\gamma_{i+1} + K_i\pi_{si} \tag{10}$$

- Upon computing all the D_{i+1} and K_i matrices, we proceed to formulate the composite matrix $K_{\text{tot}} \in \mathbb{R}^{6\cdot n \times n\cdot p_s}$, which incorporates all of them. K_{tot} will then be used to compute the column vector $\gamma_{tot} \in \mathbb{R}^{6\cdot n\times 1}$ that group all γ_i vectors.

$$\gamma_{tot} = K_{tot}\pi_{s_{tot}} \tag{11}$$

Where:

$$K_{tot} = \begin{bmatrix} K_1 & D_2^1 K_2 & \dots & D_6^1 K_6 \\ 0 & K_2 & \dots & D_6^2 K_6 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & K_6 \end{bmatrix} \tag{12}$$

And:

$$D_j^i = D_{i+1}D_{i+2}\dots D_j \tag{13}$$

In the computational implementation, all individual $Tr_i \in \mathbb{R}^{6\times 1}$ matrices are computed and then incorporated into the diagonal of the $Tr_{tot} \in \mathbb{R}^{6\cdot n\times n}$ matrix. To obtain the output matrix Y, the transformation matrix Tr_{tot} is multiplied by the matrix K_{tot}.
Ultimately, the output matrix Y can be represented as:

$$Y = Tr_{tot}K_{tot} \tag{14}$$

2) Reduction of the regression matrix: The regression matrix lacks full rank due to zero-norm columns and linear dependencies. Dynamic parameters fall into three categories:

- Non-Identifiable: Negligible effect on manipulator dynamics.
- Linear Combinations: Dependent on other parameters.
- Singularly Identifiable: Independently identifiable.

A multi-step procedure refines the matrix and discriminates between parameter types.

1) Initialization: $N > 10$ sets of positions $q(k) \in \mathbb{R}^n$, velocities $\dot{q}(k) \in \mathbb{R}^n$, and accelerations $\ddot{q}(k) \in \mathbb{R}^n$ are selected.
2) Total Regression Matrix: $Y_{tot} \in \mathbb{R}^{n\cdot N\times n\cdot p_s}$ is formed by concatenating individual matrices corresponding to the N sets. Each $i-th$ column is denoted as $f_{ci} \in \mathbb{R}^{n\cdot N}$:

$$Y_{tot} = \begin{bmatrix} Y(q(t_1), \dot{q}(t_1), \ddot{q}(t_1)) \\ Y(q(t_2), \dot{q}(t_2), \ddot{q}(t_2)) \\ \vdots \\ Y(q(t_N), \dot{q}(t_N), \ddot{q}(t_N)) \end{bmatrix} \tag{15}$$

3) Rank: The rank of Y_{tot} defines the dimension of the minimum parameter vector.
4) Non-Identifiables: Columns with zero norm are removed.
5) Identifiability: The $i-th$ column of the Y_{tot} matrix is removed, and the rank of the new matrix is computed. If the rank is reduced, the $i-th$ parameter is considered singularly identifiable, and its corresponding column in Y_{tot} is added to a new matrix Y_{ident}.
6) Linear Combination: If a linearly combined parameter corresponding to the current parameter already exists in the minimum parameter vector, then it is not included; otherwise, it is added, and its corresponding column is included in Y_{ident}.
7) Coefficients: Coefficients $\alpha_i \in \mathbb{R}$ are computed for previously excluded parameters

$$\alpha_i = (Y_{ident}^T Y_{ident})Y_{ident}^T f_{ci} \tag{16}$$

where f_{ci} is the $i-th$ column of Y_{tot} corresponding to the considered parameter.
8) Minimal Parameter Vector: The minimum identifiable parameter vector π_{min} is obtained by adding the product of the coefficients α_i and the excluded parameters (from step 5) that were inserted into π_e to the identifiable parameter vector π_{ident}:

$$\pi_{min} = \pi_{ident} + \sum_{i=1}^{len(\pi_e)} \alpha_i \pi_e(i) \tag{17}$$

3) Parameter Identification: In this phase, the minimum parameter vector's elements are identified, usually comprising linear combinations of multiple dynamic parameters. Datasets of joint positions $q \in \mathbb{R}^n$, velocities $\dot{q} \in \mathbb{R}^n$, accelerations $\ddot{q} \in \mathbb{R}^n$, and motor torques $\tau_{mot} \in \mathbb{R}^n$ are loaded. Torques are mapped back to joints via the mechanical coupling matrix. The regression matrix is numerically evaluated, resulting in A_{tot} of dimensions $(N * n) \times (length(\pi_{min}))$, a numerical estimate of Y_{tot}. A_{tot} is formed by stacking evaluations of the reduced matrix A, using values from the N static poses:

$$A_{tot} = \begin{bmatrix} A(q(t_1), \dot{q}(t_1), \ddot{q}(t_1)) \\ A(q(t_2), \dot{q}(t_2), \ddot{q}(t_2)) \\ \vdots \\ A(q(t_N), \dot{q}(t_N), \ddot{q}(t_N)) \end{bmatrix} \tag{18}$$

The process of parameter identification inherently involves the resolution of the following linear equation system:

$$\tau_{tot} = \begin{bmatrix} \tau(t_1) \\ \tau(t_2) \\ \vdots \\ \tau(t_N) \end{bmatrix} = A_{tot}\Pi_{min} \tag{19}$$

Where Π_{min} represents the numerical evaluation of the symbolic vector π_{min}.

To solve for the minimum parameters, the matrix A_{tot} is pseudo-inverted using the following formula:

$$\Pi_{min} = (A_{tot}^T A_{tot})^{-1} A_{tot}^T \tau_{tot} \tag{20}$$

Through this mathematical manipulation, one derives the vector encapsulating the estimated minimal dynamic parameters.

C. Parameter Static Optimization

Minimal parameters often represent linear combinations of actual dynamic parameters. The under-determined system thus necessitates global optimization techniques.

An optimization algorithm typically involves defining an objective function, constraints, and bounds, mathematically expressed as:

$$\begin{cases} \min f(x) \\ s.t. g(x) = 0 \end{cases} \tag{21}$$

Here, $f(x)$ represents the objective function, and $g(x)$ signifies the constraint function.

This method optimizes relevant parameters under set constraints, offering a robust solution to parameter identification. Single-variable equations were removed, as they don't require optimization. The right-hand terms were shifted to the left, resulting in:

$$ax + b = c \rightarrow ax + b - c = 0 \tag{22}$$

These terms formed a vector eq_{sol}. The objective function for static optimization was:

$$f(x) = eq_{sol} eq_{sol}^T \tag{23}$$

This setup minimizes the square of the terms, aiming for accurate and low-error solutions.

The parameters obtained from the static identification are finally grouped inside the vector Π_s.

D. Parameter Dynamic Identification

In dynamic identification, remaining parameters dependent on joint velocities and accelerations are ascertained through operational trajectories. Target parameters include motor inertias Im_i, inertia tensor elements, and frictions F_s, F_v. The friction model is both simple and effective, consisting of a viscous term proportional to the joint velocity and a static (Coulomb) component. This static part has a constant absolute value and depends on the direction of the joint velocity. [27]

The methodology resembles that of static identification, but with modifications. Kinematic and dynamic data are included in the regression matrix for parameter estimation. The focus is on parameters like motor inertia and frictions, requiring tailored optimization techniques.

1) Computing of the regression matrix: Typically, joint torques are calculated using:

$$\tau = K_r \tau_{mot} - K_r^2 I_m \ddot{q} \tag{24}$$

Where $K_r \in \mathbb{R}^{n \times n}$ is the transmission ratio matrix and $I_m \in \mathbb{R}^{3\times 3}$ the diagonal matrix of motor inertias. The first term indicates motor-to-joint torque, the second accounts for resistant torque due to motor inertia.

Lacking motor inertias, a pseudo-joint torque $\tau^* \in \mathbb{R}^n$ is used:

$$\tau^* = K_r \tau_{mot} \tag{25}$$

This τ^* also accounts for resistant torques. The equation becomes:

$$\tau^* = Y^*(q, \dot{q}, \ddot{q}) \pi_{d_{tot}} \tag{26}$$

Motor inertias are treated as unknown dynamic parameters in $\pi_{d_{tot}}$.

The matrix Γ_i is defined through the sum of two terms, similar to the static case. The first is $K_i \pi_{d_i}$, and the second is a constant $C_i \in \mathbb{R}^{6\times p_s}$, dependent on previously identified dynamic parameters.

$$\Gamma_i = K_i \pi_{d_i} + C_i \pi_{s_i} = K_i \begin{bmatrix} \hat{I}' \\ I_m^{i+1} \\ Fs \\ Fv \end{bmatrix} + C_i \begin{bmatrix} m_i \\ m_i \bar{r}_{i,C_i}^i \end{bmatrix} \tag{27}$$

I_m^{i+1}, F_s, and F_v denote motor inertia and friction coefficients. The vector $\hat{I}'$ represents inertia tensor terms, transformed via a Huygens-Steiner transformation:

$$\hat{I}_i^i = I_i^i + m_i S^T(r_{i,C_i}^i) S(r_{i,Ci}^i) \tag{28}$$

The creation of the regression matrix K_{tot} follows the same steps as previously outlined, and a similar approach is used for C_i, treated as another regression matrix. The overall expression becomes:

$$\gamma_{tot} = K_{tot} \pi_{d_{tot}} + C_{tot} \pi_{s_{tot}} \tag{29}$$

Y_{tot} is then obtained as the sum of these transformed regression matrices:

$$Y_{tot} = Tr_{tot}(K_{tot} \pi_{d_{tot}} + C_{tot} \pi_{s_{tot}}) \tag{30}$$

Components related to friction and motor inertia are integrated into $Tr_{tot} K_{tot}$, finally obtaining the regression matrix Y^*.

2) Parameter Identification: The procedures for regression matrix reduction and parameter identification are consistent with the static case.

Only K_{tot}'s regression matrix is reduced; C_{tot} remains constant. In parameter identification, the mapped motor torques yield joint torques incorporating inertial resistant torques. Additionally, C_{tot} becomes fully defined when joint kinematic parameters are substituted, and is treated as a subtractive term in the total torque:

$$\tau_{tot} = A_{tot} \Pi_{min} + C_{tot} \Pi_s \tag{31}$$

Consequently:

$$\Pi_{min} = (A_{tot}^T A_{tot})^{-1} A_{tot}^T (\tau_{tot} - C_{tot} \Pi_s) \tag{32}$$

III. EXPERIMENTAL VALIDATION

The proposed methodology has been experimentally validated using the Gaiotto GA-OL robot[1], as shown in Fig.2. This 6-DOF industrial serial manipulator features two springs positioned between the first and third joints to enhance its load handling capabilities, a detail depicted in Fig.3. Additionally, it includes mechanical couplings between the second and third joints (shown in the diagram in Fig.3), and between the fourth and fifth joints (using a bevel pair). These couplings result in a non-diagonal gear reduction matrix.

[1]www.gaiotto.com

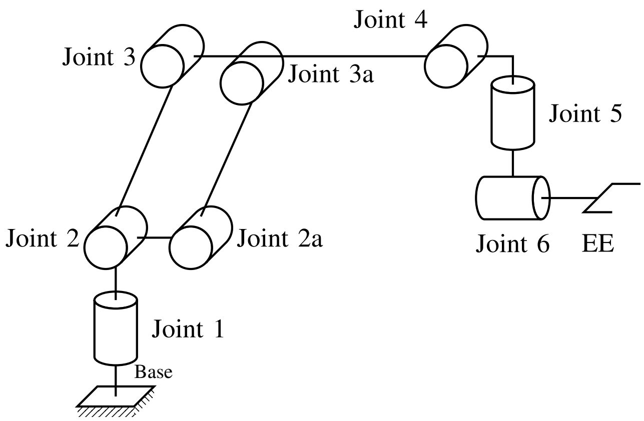


Fig. 2: The Gaiotto GA-OL diagram illustrates the positioning of joints and highlights the coupling between joints 2 and 3 through a closed-kinematic chain (Joint 2a and 3a). The coupling between Joints 4 and 5 (a bevel pair) and Springs are not depicted in this diagram.

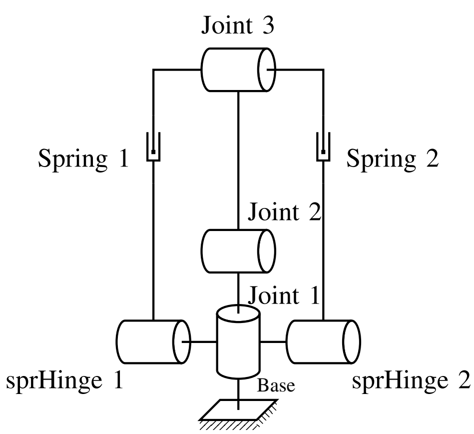


Fig. 3: The Gaiotto GA-OL Spring Layout details the placement of springs within the robot's structure, specifically between joints 1 and 3. These springs extend as joint 2 moves, exerting torque along joint 2.

A. Ideal Case Study

Three different MATLAB models of an ideal manipulator were generated to validate the code, using parameters similar to those of the Gaiotto GA-OL robot. The unknown dynamic parameters were derived from the manipulator's CAD drawings. The three idealized models evaluated include: i) a basic serial manipulator, ii) a serial manipulator equipped with mechanical couplings, and iii) a serial manipulator that incorporates both mechanical couplings and compensation springs.

In the static identification phase, we used a dataset consisting of around 1600 points arranged in a 3D grid, covering most of the robot's workspace. This comprehensive dataset allowed us to thoroughly analyze the spatial distribution of joint torques. Utilizing the methods outlined in previous chapters, static identification and optimization processes were applied to these ideal models. Data sets were generated using the actual joint positions at rest, which then informed the computing of joint torques for the specific poses imposed on the models.

Static Identification			Static Optimization		
	MSE	SDE		MSE	SDE
Link 1	5.08e-33	7.00e-15	Link 1	5.08e-33	7.00e-17
Link 2	1.41e-04	1.19e-02	Link 2	1.64e-04	1.19e-02
Link 3	6.66e-04	2.58e-02	Link 3	6.82e-04	2.58e-02
Link 4	2.09e-05	4.56e-03	Link 4	2.93e-05	4.56e-03
Link 5	6.02e-05	7.75e-03	Link 5	6.02e-05	7.75e-03
Link 6	1.43e-11	3.77e-06	Link 6	1.36e-10	8.18e-06

TABLE I: Static identification and optimization results of the Ideal MATLAB Model. MSE: Mean Square Error, SDE: Standard Deviation of Error

Table I presents the validation results for the static identification and optimization of the third model, incorporating an added noise of 5% of the torque range to reflect its idealized nature. These results were obtained by comparing the torques measured on the ideal robot model in various poses to those inferred by the static identification model in the same positions. The low Mean Squared Error (MSE) values indicate a high precision level in the identification process.

Regarding the dynamic identification phase, to create a coherent data set for the ideal robot, data on real-time joint position, velocity, and acceleration were utilized. The dataset was constructed from seven distinct trajectories, each consisting of approximately 70,000 samples at varying speeds, to extensively cover the robot's workspace. Similar to the static case, this data informed the recalculation of torques for the imposed trajectories.

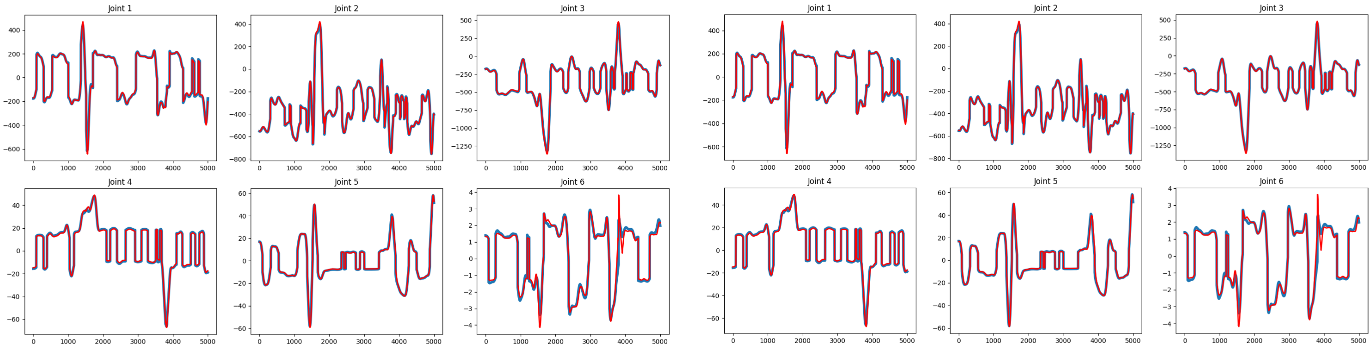
In Fig.4, the measured and predicted joint torques for the six joints during the model validation phase are presented. The identification process demonstrates a close match between the actual and predicted data, accurately reflecting the robot's behavior. However, the validation phase leads to an underdetermined system of equations without a single definitive solution. Consequently, the optimization phase seeks to at least match the results of the identification phase, without necessarily improving the model's accuracy.

B. Gaiotto GA-OL Static Identification

The static identification procedure yields the previously mentioned under-determined system of equations for the actual manipulator. To validate these parameters, the torques measured on the robot's motors were compared to those predicted by the derived system. The outcomes, showcased in Table II, illustrate the model's precision through the calculation of the Mean Squared Error (MSE) and the Standard Deviation of the errors for each joint.

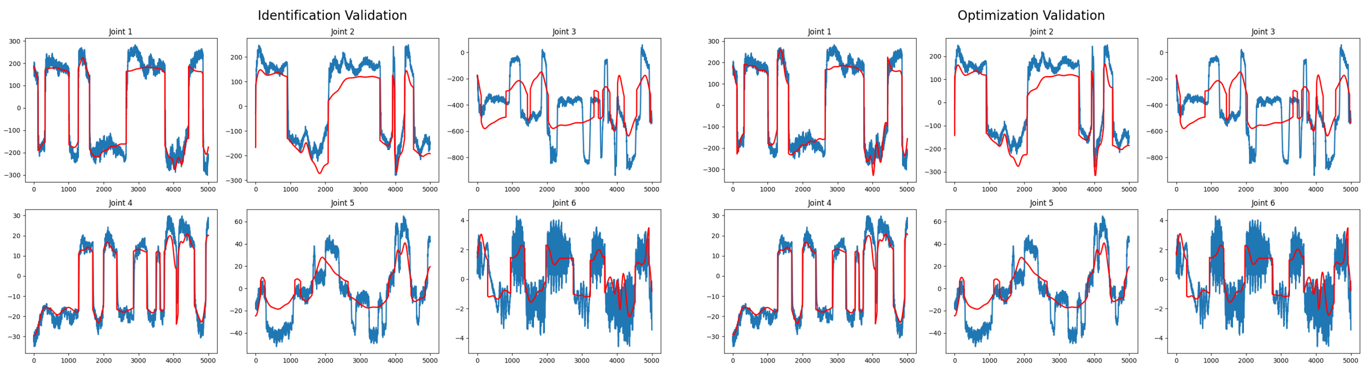
It is important to note that the optimization phase may reduce the precision of the inferred parameters. Therefore, establishing an accurate system of equations during the identification phase is crucial.

The model accurately predicts most of the joint torques but shows noticeable inaccuracies for joint 3, likely attributable to insufficient real-world testing for this joint. Additionally, a



(a) Dynamic identification results (b) Dynamic optimization results

Fig. 4: MATLAB Ideal Model results. Blue solid lines: measured torque of the ideal model; Red solid lines: predicted torque from the identified model.



(a) Gaiotto GA-OL dynamic identification results (b) Gaiotto GA-OL dynamic optimization results

Fig. 5: Gaiotto GA-OL dynamic results. Blue solid lines: measured torque of the real robot; Red solid lines: predicted torque from the identified model.

Static Identification			Static Optimization		
	MSE	SDE		MSE	SDE
Link 1	169.69	12.54	Link 1	169.67	12.54
Link 2	803.85	24.19	Link 2	1071.41	25.59
Link 3	6497.74	74.50	Link 3	6821.87	74.14
Link 4	15.73	2.89	Link 4	16.89	3.07
Link 5	14.02	3.67	Link 5	14.04	3.73
Link 6	0.53	0.48	Link 6	0.56	0.52

TABLE II: Static identification and optimization results of the Gaiotto GA-OL. MSE: Mean Square Error, SDE: Standard Deviation of Error

minor inaccuracy for joint 2 suggests a need for refinement in the spring model. During the static optimization phase, limits for individual parameters were established based on CAD estimates and geometric constraints [25]. The optimization commenced with initial parameters derived from CAD data, with results detailed in Table II. The obtained Mean Squared Error (MSE) reflects the same challenges identified during the identification phase.

C. Gaiotto GA-OL Dynamic Identification

The dynamic identification phase yielded a comparison between model-predicted and actual joint torques. Fig. 5a illustrates the high precision of the model predictions, although the resulting system of equations for defining the robot's dynamic parameters is under-defined. Similar to the static identification phase, the model accurately predicts most joint torques but shows inaccuracies for joint 3. To enhance the results, a deeper analysis of the closed-kinematic nature of joints 2 and 3, along with the spring model, is necessary. In the dynamic optimization phase, initial constraints based on the physical limits were applied to the dynamic parameters. These constraints were deliberately broad to allow an extensive algorithmic exploration of the solution space. Before optimization, initial parameters were established using CAD estimates for the links. Unlike in the static case, the dynamic scenario requires that inertia tensors be positively defined, ensuring that the inertia matrix is positive-definite, denoted as $\lambda(I_i) > 0$. The results of the optimization, shown in 5b, closely align with the identification results, indicating that the optimization phase successfully identified the optimal set of parameters for an accurate representation of the real model's system of equations.

D. Models obtained

For future model-based control, two modeling approaches are considered:

- Newton-Euler Model: Uses Newton-Euler equations with substituted dynamic parameters from optimization to find joint torques based on position, velocity, and acceleration.

$$\tau_i^i = \mu_i^{iT} {R_i^{i-1}}^T \bar{z}_0 + K_r^i I_m^i \dot{\omega}_m^{iT} \bar{z}_m^i \tag{33}$$

- Regression Matrix: Utilizes the regression matrix to recalculate minimum parameters without optimization, sacrificing individual parameter knowledge but gaining model precision.

$$\tau_{tot} = A_{tot} \Pi_{min} + C_{tot} \Pi_s \tag{34}$$

  Both cases offer different dynamic parameter sets for model-based control testing.

IV. CONCLUSIONS

In conclusion, this paper offers a significant contribution to the field of robotics and automation, particularly for industrial applications requiring 6-DoF serial manipulators. The authors introduce a novel, two-step approach for the identification of dynamic parameters, which is critical for improving the performance and reliability of these systems. The paper paves the way for more accurate and efficient manipulators by addressing the complexities introduced by coupled joints and load-assisting springs. Future work could build on these results by conducting a more detailed study of the closed-kinematic model to more precisely evaluate weight distribution during the robot's operation. Additionally, the proposed methodology could be applied to a wider variety of manipulator configurations, or these algorithms could be integrated into real-time control systems.

V. ACKNOWLEDGMENT

This work is partly supported by Regione Emilia-Romagna under the agreement "Innovazione della value proposition di Gaiotto Automation 2021-2025" (grant number / CUP: E32C21001090009)."

REFERENCES

[1] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo *et al.*, *Robotica–Modellistica, Pianificazione e Controllo*. McGraw-Hill libri Italia, 2008.

[2] J. Swevers, W. Verdonck, and J. De Schutter, "Dynamic model identification for industrial robots," *IEEE control systems magazine*, vol. 27, no. 5, pp. 58–71, 2007.

[3] J. Wu, J. Wang, and Z. You, "An overview of dynamic parameter identification of robots," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 5, pp. 414–419, Oct. 2010.

[4] C. Gaz, M. Cognetti, A. Oliva, P. Robuffo Giordano, and A. De Luca, "Dynamic Identification of the Franka Emika Panda Robot With Retrieval of Feasible Parameters Using Penalty-Based Optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, Oct. 2019.

[5] A. Jubien, M. Gautier, and A. Janot, "Dynamic identification of the Kuka LightWeight robot: Comparison between actual and confidential Kuka's parameters," in *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Jul. 2014, pp. 483–488.

[6] S.-P. Liu, "An improved parameter identification method of redundant manipulator," https://journals.sagepub.com/doi/epub/10.1177/17298814211002118, 2021.

[7] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[8] J. Hollerbach, W. Khalil, and M. Gautier, "Model identification," *Springer handbook of robotics*, pp. 113–138, 2016.

[9] M. Gautier, A. Janot, and P.-O. Vandanjon, "Didim: A new method for the dynamic identification of robots from only torque data," in *2008 IEEE international conference on robotics and automation*. IEEE, 2008, pp. 2122–2127.

[10] J. Jia, M. Zhang, C. Li, C. Gao, X. Zang, and J. Zhao, "Improved dynamic parameter identification method relying on proprioception for manipulators," *Nonlinear Dynamics*, vol. 105, no. 2, pp. 1373–1388, Jul. 2021.

[11] M. Gautier, "Dynamic identification of robots with power model," in *Proceedings of international conference on robotics and automation*, vol. 3. IEEE, 1997, pp. 1922–1927.

[12] S. Van Huffel and J. Vandewalle, *The total least squares problem: computational aspects and analysis*. Siam Philadelphia, 1987, vol. 11.

[13] A. Janot, M. Gautier, A. Jubien, and P. O. Vandanjon, "Comparison between the cloe method and the didim method for robots identification," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 5, pp. 1935–1941, 2014.

[14] A. Janot, P.-O. Vandanjon, and M. Gautier, "A generic instrumental variable approach for industrial robot identification," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 1, pp. 132–145, 2013.

[15] L. Ding, X. Li, Q. Li, and Y. Chao, "Nonlinear Friction and Dynamical Identification for a Robot Manipulator with Improved Cuckoo Search Algorithm," *Journal of Robotics*, vol. 2018, 2018.

[16] Z. Fu, J. Pan, E. Spyrakos-Papastavridis, Y.-H. Lin, X. Zhou, X. Chen, and J. S. Dai, "A Lie-Theory-Based Dynamic Parameter Identification Methodology for Serial Manipulators," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 5, pp. 2688–2699, Oct. 2021.

[17] C. D. Sousa and R. Cortesao, "Physical feasibility of robot base inertial parameter identification: A linear matrix inequality approach," *The International Journal of Robotics Research*, vol. 33, no. 6, pp. 931–944, 2014.

[18] ——, "Inertia tensor properties in robot dynamics identification: A linear matrix inequality approach," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 1, pp. 406–411, 2019.

[19] P. M. Wensing, S. Kim, and J.-J. E. Slotine, "Linear matrix inequalities for physically consistent inertial parameter identification: A statistical perspective on the mass distribution," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 60–67, 2017.

[20] Y. Han, J. Wu, C. Liu, and Z. Xiong, "An iterative approach for accurate dynamic model identification of industrial robots," *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1577–1594, 2020.

[21] H. Su, W. Qi, C. Yang, J. Sandoval, G. Ferrigno, and E. D. Momi, "Deep Neural Network Approach in Robot Tool Dynamics Identification for Bilateral Teleoperation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2943–2949, Apr. 2020.

[22] C. Tan, H. Zhao, and H. Ding, "Identification of dynamic parameters of closed-chain industrial robots considering motor couplings," *Computers and Electrical Engineering*, vol. 99, p. 107740, Apr. 2022.

[23] E. Rohmer, S. P. N. Singh, and M. Freese, "Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013, www.coppeliarobotics.com.

[24] J. Y. Luh, M. W. Walker, and R. P. Paul, "On-line computational scheme for mechanical manipulators," 1980.

[25] C. Gaz, F. Flacco, and A. De Luca, "Extracting feasible robot parameters from dynamic coefficients using nonlinear optimization methods," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2075–2081.

[26] Å. Björck, "Least squares methods," *Handbook of numerical analysis*, vol. 1, pp. 465–652, 1990.

[27] M. Ragaglia, A. M. Zanchettin, L. Bascetta, and P. Rocco, "Accurate sensorless lead-through programming for lightweight robots in structured environments," *Robotics and Computer-Integrated Manufacturing*, vol. 39, pp. 9–21, Jun. 2016.