Online Identification of Payload Inertial Parameters Using Ensemble Learning for Collaborative Robots

Wael Taie, Khaled ElGeneidy, Ali AL-Yacoub, and Sun Ronglei

***Abstract*—Collaborative robots (Cobots) are essential in flexible automation solutions, enabling fast and easy reconfiguration to adapt to varying task requirements in dynamic environments. This requires the ability to safely handle different payloads with varying inertial parameters, which may not be known in advance. Hence, online identification of the payload's inertial parameters becomes essential for safe interactions, accurate path following, and stable grasping. Most existing methods require additional sensors, calibration procedures, or custom filtering, which increases the complexity and estimation time. In this letter, we propose a novel online identification method that employs a bagging ensemble machine learning approach to identify the payload inertial parameters without external sensors or additional filtering and calibration steps. The method uses available joint position, velocity, and torque measurements from the Cobot to train neural networks and decision trees as weak learners. The method is tested in simulation and validated using the Franka Emika Panda Cobot. The results showed that our method outperforms the state-of-the-art recursive least square methods reducing prediction errors by 75%–78% for mass and 49.5%–60% for the center of mass, while estimating accurate payload parameters within the first time step.**

***Index Terms*—In-Hand manipulation, online identification, payload dynamics.**

I. Introduction

FORECASTS for the collaborative robots (Cobots) market show strong expansion across many industries. According to Statista Research Department [1], The expected compound annual growth rate of the worldwide Cobot market is anticipated to exceed 12% during the period from 2020 to 2030, reaching up to $2.1 billion by 2026. Cobots are often favoured in dynamic environments, particularly for small-batch production, where their tasks can frequently change. The task variation can include following different trajectories, as well as carrying varying payloads. Adapting the cobot to execute new tasks can be achieved through the robot interface by programming a new trajectory or through direct human-robot interaction during the teaching process. Neglecting changes in the dynamic properties (mass, center of mass, and inertia matrix) of new payloads, can result in safety and performance issues such as trajectory tracking errors, object slippage, or false collision detection due to torques from uncompensated loads. Those issues become more evident in high-speed applications and model-based control strategies. The benefits of accurate estimation of a payload's inertial parameters can be leveraged in many applications such as dynamic motion planning algorithms to prevent collisions with moving obstacles [2], improving sensor-less collision detection [3], and minimizing trajectory tracking errors [4]. Additionally, it plays a crucial role in optimizing grasping pose determination, as explored in [5] and [6]. Hence, for cobots to safely and reliably manipulate new loads, fast and accurate identification of the payload's inertial parameters is essential to allow the controller to compensate for the variations in the payload and maintain task performance.

In this letter, we propose a novel bagging ensemble machine learning method for fast online identification of the inertial parameters of unknown loads, which does not require additional external sensors or task-specific filtering and calibration steps ahead of each new task. This can save reconfiguration time in flexible automation systems, as the Cobot can quickly estimate the new payload's inertial parameters online as part of the teaching stage, without delaying the production cycle. The key contribution of this method is the exclusion of the noisy joint acceleration data from the required measurements, which lifts the need for any additional filtering steps. A comparative study between the accuracy of ensemble machine learning-based neural networks (NN) and ensemble machine learning-based decision trees (DT) is presented and benchmarked against recent work presented in [7] utilizing RLS with additional filters and calibration steps. It is worth noting that this work utilizes the same robot platform and excitation trajectory for a fair comparison.

This letter's subsequent sections are structured as follows: Section II offers an overview of current payload identification methodologies, while Section III defines the mathematical formulation of the problem. The implementation of the proposed method is illustrated in Section IV, followed by performance evaluations through experiments in Section V. Finally, Section VI concludes the study and outlines future work.

Manuscript received 13 August 2023; accepted 5 December 2023. Date of publication 22 December 2023; date of current version 2 January 2024. This letter was recommended for publication by Associate Editor G. Bao and Editor H. Liu upon evaluation of the reviewers' comments. This work was supported by the National Natural Science Foundation of China under Grant U21A20121. *(Corresponding author: Wael Taie.)*

Wael Taie is with the State Key Laboratory of Intelligent Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China, and with the Department of Mechatronics Systems Engineering, October University for Modern Sciences and Arts, Cairo 12451, Egypt (e-mail: I201922098@hust.edu.cn).

Khaled ElGeneidy is with the School of Engineering, Coventry University, Cairo 11837, Egypt (e-mail: ad5316@coventry.ac.uk).

Ali AL-Yacoub is with the EPSRC Centre for Innovative Manufacturing in Intelligent Automation, Loughborough University, LE11 3TU Loughborough, U.K. (e-mail: a.al-yacoub@lboro.ac.uk).

Sun Ronglei is with the State Key Laboratory of Intelligent Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: ronglei@hust.edu.cn).

Digital Object Identifier 10.1109/LRA.2023.3346268

II. The State of the art

Objects' inertial parameters can be determined using three distinct approaches based on the level and method of interaction with the object as classified in [8]. The first approach relies solely on visual information like object geometry, RGB images, and depth images to establish links between visual and inertial parameters. This approach requires a camera and either prior object geometry knowledge or extensive datasets. The Exploratory method, the second approach, involves direct interaction with the object, measuring applied forces and motion to estimate parameters based on physical laws or learning models. It provides accurate estimates but relies on analytical models and controlled settings. In contrast, the fixed object approach involves attaching the payload to the end-effector of the robot. This method estimates parameters by using the measured end-effector wrench and joint motions. This approach proves more suitable for our study because it does not demand additional sensors. It can provide a precise estimation of the inertial parameters by using the object's 3D movement capability. Furthermore, the integration of visual and force-torque measurements is often used to enhance accuracy, as shown in [9].

The Studies in the third approach are inspired by classical dynamic model identification techniques in robotics, which are compared in [10]. This work indicates that only methods based on least square identification and machine learning regression can work online effectively. Least-squares (LS) fitting presented in [11] has been the standard procedure for the identification of payloads for several decades. This methodology resembles the identification of robot dynamics proposed in [12]. The LS fitting equations can be solved using offline classical LS or online Recursive Least-Squares Method (RLS). Depending on the system's hardware capabilities and specific requirements, more advanced adaptations of the LS method are proposed, such as the Recursive Total Least-Squares Method (RTLS) as shown in [13], or quaternion-based filtering as presented in [14].

The utilization of RLS and RTLS approaches necessitates the measurement of payload forces and acceleration. These methodologies are divided into four categories. The initial group always needs additional sensors to evaluate the wrench between the end-effector and payload through a wrist-mounted 6D force/torque sensor coupled with acceleration data from an inertial measurement unit (IMU) [15]. The second group does not need additional sensors; it employs joint torque sensing and the robot's established dynamic model to estimate external torque, subsequently deducing payload forces [7], [16]. The third cluster also does not need additional sensors, it utilizes variations in measured joint torque during motions with and without the payload to compute payload inertial parameters during repeated identical motions [17], [18]. The fourth cluster integrates payload and robot dynamics identification, resulting in more complex calculations and parameter identification challenges. It also relies on measurements of the joints and does not need additional sensors [3].

The main problem with using the LS estimation approach with these groups is the assumption that all the measurement errors come only from the torque measurements and that the data matrix is error-free as illustrated in [8] and [13]. The measurements of the joint's position, velocity, and acceleration are included in the data matrix. It is well recognized that double-time differentiation of the encoder data to calculate the acceleration of the joints results in noisy acceleration data with multiple outliers and a low signal-to-noise ratio. The presence of a noisy signal in the data matrix decreases the accuracy of least squares prediction as indicated in [10]. To mitigate the lack of prediction accuracy, robust filtering and calibration techniques are imperative to ensure accurate LS outcomes. Yet, using additional filtering or calibration processes increases the computational complexity of the estimation process and consumes additional time.

Advanced system identification methods, including artificial neural networks (ANN) and intelligent algorithms, have evolved to handle intricate nonlinear systems. An improved chaotic sparrow search algorithm proposed in [19], combines Kent chaotic mapping, Student's t-distribution, and the Lévy flight strategy to identify manipulator dynamic parameters. this method gives accurate results, but it requires more computational resources and time than traditional algorithms due to the complex operations involved in the chaotic map, the t-distribution, and the Lévy fight. [20] utilizes unsupervised classification to identify the payload variation using the joint toque signal. Meanwhile, [21] employs support vector machine regression to estimate payload parameters in excavator equipment. The results of this work only show the mass parameter estimation in a simulation environment. it gives higher accuracy than the traditional least square identification method. the support vector machine model is a singular predictive model.

Evidence, as illustrated in [22] indicates that depending exclusively on one singular predictive model, such as an individual decision tree (DT) or one neural network (NN), may not produce optimal outcomes. Ensemble techniques have been proven more effective, whether they involve aggregating models of the same type, such as using a group of NN models or a group of DT models, or a blend of different model types, such as using a group of NN and DT models together, as corroborated by [23]. This ensemble approach contributes to reducing bias, variance, or both. In addition, a single model's limitations in capturing intricate data structures are countered by combining models, thus enhancing prediction accuracy, as supported by [24]. Hence, an ensemble machine learning method is suitable for this task. It can build robust models against the torque sensor noise and provide fast, accurate estimation for the inertial parameters.

III. Mathematical Formulation

A. Payload Dynamics

Identifying the inertial parameters of payloads attached to manipulators as shown in Fig. 1 requires establishing relationships between the inertia matrix $I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix}$, mass($m$),center of mass $r = \begin{bmatrix} r_x & r_y & r_z \end{bmatrix}^T$, measured forces ($F$), torques ($\tau$), kinematic parameters (linear acceleration (a) and rotational acceleration(α), angular velocity(ω)), and gravity vector (g_0). To accomplish this, the Newton-Euler equations for the object are formulated as the following:

$$\vec{F}_{EE} = m\vec{a}_{EE} + \vec{\alpha} \times m\vec{r} + \vec{\omega} \times (\vec{\omega} \times m\vec{r}) - m\vec{g} \tag{1}$$

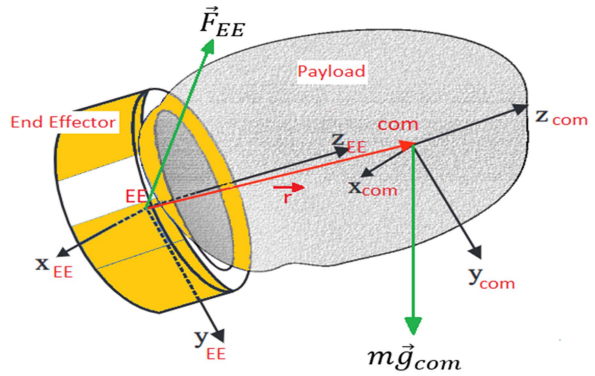


Fig. 1. Forces exerted on the payload.

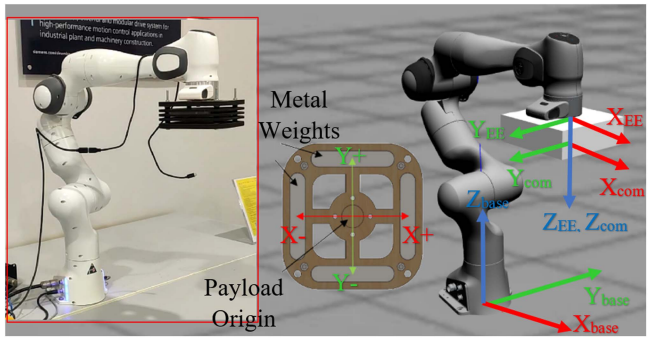


Fig. 2. Simulation of the Franka Emika system mimics the real-world experimental configuration while holding the payload. In the middle of the figure, the top view of the payload's design is depicted, revealing its coordinate frame.

$$\vec{\tau}_{EE} = I_{EE}\vec{\alpha} + \vec{\omega} \times (I_{EE}\vec{\omega}) + m\vec{r} \times \vec{a}_{EE} - m\vec{r} \times \vec{g} \quad (2)$$

Assuming that all parameters are measured relative to the end-effector frame, the inertia matrix I is aligned with the end-effector frame's axes using the parallel axis theorem. These parameters are combined into the $\phi = \left[m, mr_x, mr_y, mr_z, I_{xx}, I_{xy}, I_{xz}, I_{yy}, I_{yz}, I_{zz}\right]$ in order to solve (1) and (2) for the dynamic parameters. Moreover, the resultant wrench force F can be expressed as follows:

$$F = \begin{bmatrix} \vec{F}_{EE} \\ \vec{\tau}_{EE} \end{bmatrix} = V(\omega, \alpha, a, g)\phi \quad (3)$$

Where $V(\omega, \alpha, a, g)$ is the data matrix which includes all kinematic data. Note that the quantities introduced thus far have been measured in Cartesian space, as a payload-centered perspective is employed. Nonetheless, a joint-space perspective, which offers benefits in robotics, can be achieved by contemplating the following:

$$J^T(q)F = J^T(q)V(\omega, \alpha, a, g)\phi$$
$$\tau = B(a, \alpha, \omega, g, q)\phi \quad (4)$$

Where $J(q) \in R^{6\times n}$ is the manipulator Jacobian matrix at the end-effector frame. From this model, we conclude that the inertial parameters identification depends on the joint torque values and the robot trajectory. So, when the data-driven approach is used it is important to have a dataset that contains these crucial parameters.

B. The Excitation Trajectory

The use of sinusoidal trajectories in joint space is a popular method for designing excitation trajectories for parameter estimation [7], [13], [15], [25]. The trajectory of a joint is composed by combining a given number of N-weighted sine and cosine functions, respectively. According to this method, the trajectory of the joint is provided as:

$$q_i(t) = q_{i,0} + \gamma_i(t) \quad (5)$$

$$\gamma_i(t) = \sum_{k=1}^{n} \rho_{i,k} \sin(2\pi k f t) + \delta_{i,k} \cos(2\pi k f t) \quad (6)$$

$\gamma_i(t)$ signifies the sinusoidal component.$q_{i,0}$ represents a constant offset angle; $\rho_{i,k}$ and $\delta_{i,k}$ are the coefficients of the sine and cosine parts, respectively. To achieve periodicity, the frequency f of the joint trajectories is identical. It should be noted that trajectories based on (5) are jerk-limited (since the derivative of a sinusoidal function is sinusoidal again), and so reduce unwanted excitation of the manipulator's mechanical structure. In [13], the weights $\rho_{i,k}$ and $\delta_{i,k}$ are optimized by an appropriate criterion that analyzes the effect of the selected trajectory on the noise and bias sensitivity of the estimates. Nevertheless, in [7], only hand-tuned motions are used. In this approach, all robot joints are used to generate the excitation trajectory regardless of whether it affects the process of estimation or not.

IV. Ensemble Machine Learning Implementation

For our experiments, Franka Emika panda Cobot is used. It has 7 degrees of freedom; it provides 7 joint position, velocity, and torque measurements. Its maximum payload is 3 kg. A Franka Emika cobot simulator has been developed to gather a dataset for training a learning model. Subsequently, the trained model will undergo validation using the physical Franka Emika cobot. The Gazebo platform was utilized to do simulations of the cobot, including a ROS interface. The simulator imitates the behavior of the actual robot, providing accessible controls and real-time feedback on the robot's status. This simulator mirrors the experience of using the Franka ROS Interface package. The Franka Emika controller is used without compensating the inertial parameters of the payload. The payload is fixed at the end-effector of the cobot where the payload frame axes are aligned with the end-effector frame axes as shown in Fig. 2.

The payload is designed to allow changing the mass (m) and center of mass components (r_x, r_y and r_z) independently. The payload shown in Fig. 2 looks like a container that has 4 slots; each slot can contain 3 metal weights; each metal weight mass is 0.275 kg. The mass of the empty container is 0.790 kg. Changing the location of the metal weights between the four slots represents changing the center of mass components r_x or r_y or both. Changing the weight position inside the same slot represents changing r_z. Changing the number of metal weights represents changing the mass of the payload. The payload is designed by Solidworks Software where the complete inertial parameters

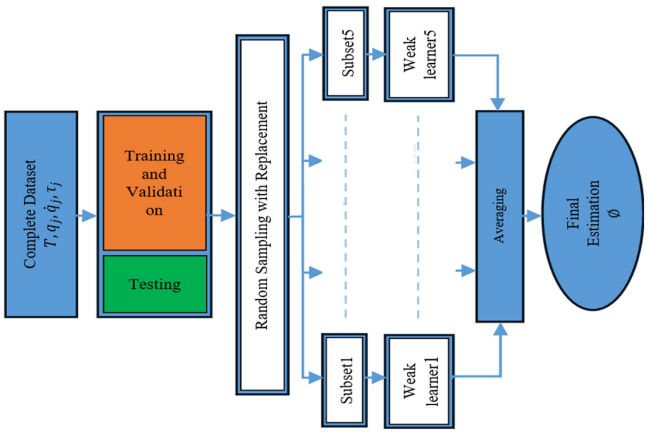


Fig. 3. Proposed Bagging Ensemble Method.

data $(m, mr_x, mr_y, mr_z, I_{xx}, I_{xy}, I_{xz}, I_{yy}, I_{yz}, I_{zz})$ can be extracted. This design helps in collecting datasets with a diverse range of payloads. The dataset collection process is performed by setting up the new payload with new inertial parameters and moving the cobot through a specific excitation trajectory.

Every different payload affects the position, velocity, and torque measurements for the 7 joints of the cobot while executing the excitation trajectory. So, every data set row has 32 variables (time stamp, 10 payload inertial parameters, 7 joint position measurements, 7 joint velocity measurements, and 7 torque measurements). The masses used in the dataset start with the mass of the empty container, then it changed by adding metal weights one by one till 5 metal weights. The total mass of the container with 5 metal weights is 2.163 kg, which is lower than the max payload. However, it is used to ensure the safety of the process and to avoid the reflex error in the cobot controller while executing the variable velocity and acceleration trajectory. The payload has two slots in the X direction (+X and -X), two slots in the Y direction (+Y and -Y), and two positions in every slot in the Z direction (top and bottom). The objects in the dataset have been chosen according to the available positions in the four slots that allow the change of one component of the center of mass while fixing the others for every mass. Seventy-seven different objects have been used in the database to ensure diversity. The total number of samples in this dataset is 249788 samples.

Our results were benchmarked against the state-of-the-art work employing RLS without using any external sensor. Hence, the excitation trajectory is adopted from [7] and tested using the same robot. The adopted excitation trajectory is a sinusoidal trajectory generated from (5). The weights of the sinusoidal function are matched with the benchmarked work.

The bagging ensemble machine learning technique is illustrated in Fig. 3. The input dataset is divided into 70% for training and 15% each for validation and testing. Training and validation data are randomly sampled with replacement into subsets used to train weak learners. In this study, simple NN and DT serve as weak learners. A single hidden layer NN with 500 ReLU-activated neurons is employed, trained for up to 1000 epochs. DT lacks depth and leaf node restrictions, needing 2 samples per internal node split and 1 for leaf node creation; splitting disregards minimum impurity decrease, using all features for optimal splits. The bagging ensemble comprises five weak learners, trained twice: one with a group of five NNs and another with a group of five DTs, hyperparameters determined via trial and error.

The training process followed two approaches to train the weak learners. The first approach can be called "single stage training", which uses all training data to identify all the 10 inertial parameters at once using 5 weak learners The output of every weak leaner is ϕ_i which represent the inertial parameters of the payload $(m_i, mr_{x_i}, mr_{y_i}, mr_{z_i}, I_{xx_i}, I_{xy_i}, I_{xz_i}, I_{yy_i}, I_{yz_i}, I_{zz_i})$ that estimated by the weak learner i. The second approach can be called "two-stage training", which splits the training into two stages. the first trains only five weak learners using mass and center of mass parameters only $(m_i, mr_{x_i}, mr_{y_i}, mr_{z_i})$. In the second step, another five weak learners are trained solely using the moments of inertia parameters $(I_{xx_i}, I_{xy_i}, I_{xz_i}, I_{yy_i}, I_{yz_i}, I_{zz_i})$. The output of the bagging ensemble model is ϕ, which includes the average value of every inertial parameter estimated from the 5 weak learners as shown in (7), (8) :

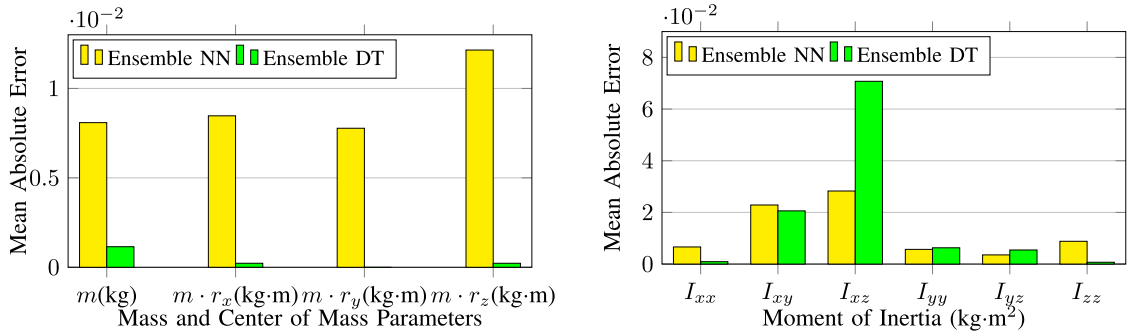
$$\phi = (m_{\text{avg}},\ mr_{x_{\text{avg}}},\ mr_{y_{\text{avg}}},\ mr_{z_{\text{avg}}},\ I_{xx_{\text{avg}}},\ I_{xy_{\text{avg}}},\ I_{xz_{\text{avg}}},\ I_{yy_{\text{avg}}},\ I_{yz_{\text{avg}}},\ I_{zz_{\text{avg}}}) \tag{7}$$

$$K_{\text{avg}} = \frac{\sum_{i=1}^{5} K_i}{5} \tag{8}$$

Where K_{avg} is the average value. and K_i is the inertial parameter estimated from weak learner i. The transition of the mathematical model in (4) to the bagging ensemble model is depicted in Fig. 3. The model's input encompasses the time step T, joint angle q_j, joint velocity$\dot{q}$, and measured joint torque τ_j. Joint acceleration is not utilized due to its' noise characteristics. This noisy signal poses accuracy challenges for the model's outcomes. The model's output represents the payload's inertial parameters.

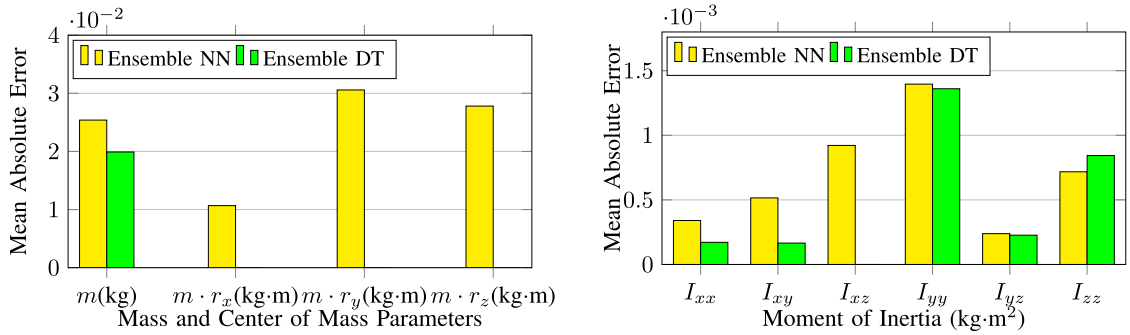
V. Results and Discussion

The results presented below demonstrate the capability of identifying payload inertial parameters through cobot joint sensor measurements, eliminating the need for external sensors or additional calibration procedures. In Fig. 4, the outcomes of the bagging ensemble from single-stage training are depicted. The comparison involves mean absolute errors of bagging ensembles employing Neural Network (NN) weak learners and Decision Tree (DT) weak learners. The ensemble DT surpasses the ensemble NN in estimating m, $m \cdot r_x$, $m \cdot r_y$, $m \cdot r_z$, I_{xx}, I_{xy}, and I_{zz} by 85.8%, 97.3%, 99.85%, 98.2%, 86%, 10%, and 92%, respectively. Conversely, the ensemble NN outperforms the ensemble DT in estimating I_{xz}, I_{yy}, and I_{yz} by 60%, 9.7%, and 34.8%, respectively. In the two-stage training results depicted in Fig. 5, the ensemble DT outperforms the ensemble NN by 21.6%, 49.7%, 67.8%, 2.5%, and 4.5% for the estimations of $m, I_{xx}, I_{xy}, I_{yy}$, and I_{yz}, respectively. Additionally, it exhibits an improvement of about 99.5% for the estimations of $m \cdot r_x$, $m \cdot r_y$, $m \cdot r_z$, and I_{xz}. Generally, the Decision Tree demonstrates superior performance compared to the Neural Network in this process while validating the model with the simulation data.



(a) Estimation of mass and center of mass parameters (b) Estimation of moment of inertia parameters

Fig. 4. Mean absolute error of bagging ensemble model obtained from single-stage training.



(a) Estimation of mass and center of mass parameters (b) Estimation of moment of inertia parameters

Fig. 5. Mean absolute error of bagging ensemble model obtained from two-stage training.

The comparison between one-stage and two-stage methods can be concluded from the results presented in Figs. 4 and 5. The single-stage method demonstrates superior performance over the two-stage method when employing an ensemble NN model, showcasing improvements of 68%, 20.6%, 74.5%, and 56% in the estimation of parameters $m, m \cdot r_x, m \cdot r_y$ and $m \cdot r_z$ respectively. Furthermore, the single-stage method outperforms the two-stage method with an ensemble DT model by 94.2% and 84.80% in the estimation of parameters m and $m \cdot r_y$. Conversely, the two-stage method excels over the single-stage method with an ensemble NN model, exhibiting significant enhancements of 94.8%, 97.7%, 96.7%, 75.5%, 93.3%, and 91.8% in the estimation of I_{xx}, I_{xy}, I_{xz}, I_{yy}, I_{yz} and I_{zz} respectively. Additionally, the two-stage method outperforms the single-stage method using an ensemble DT model by 81.5%, 99.2%, 99.8%, 78.4%, and 95.8% in the estimation of I_{xx}, I_{xy}, I_{xz}, I_{yy} and I_{yz} respectively.

A. Experimental Validation

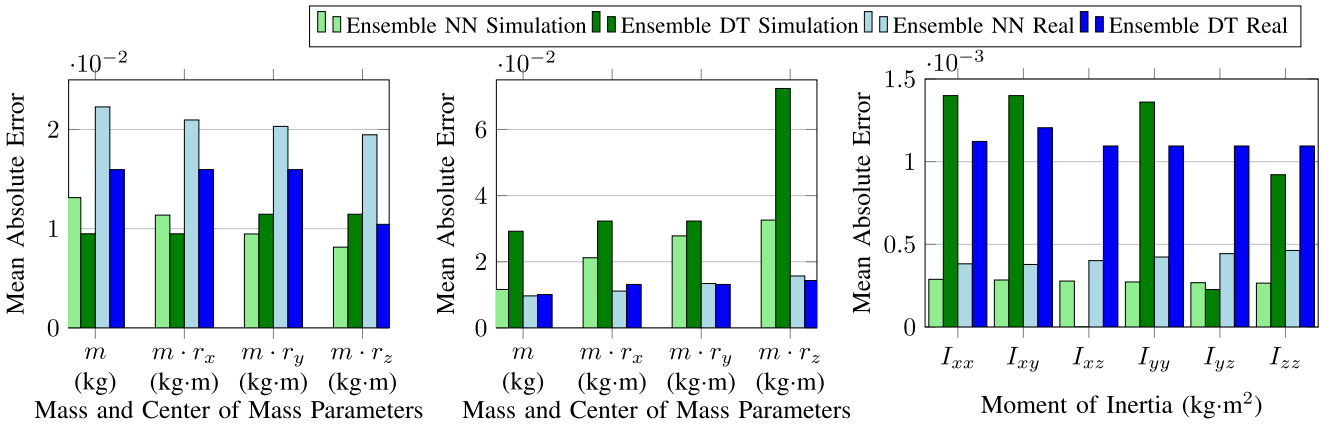
In this step, we validate the model in a real-world environment. The real Franka Emika cobot platform was carefully prepared to replicate the conditions of the simulation environment, as depicted in Fig. 2. The testing payload was securely affixed to the end effector of the cobot for the validation process. Table I illustrates the actual payload inertial parameters. The testing payload parameters used during this validation process are entirely new and were not part of the training dataset. These unseen parameters are crucial in assessing the model's generalization ability to previously unseen data. The cobot executed the excitation trajectory during the validation process, and the resulting identification errors were recorded.

TABLE I
REAL TESTING PAYLOAD INERTIA PARAMETERS

m (kg)	r_x (m)	r_y (m)	r_z (m)		
1.340	-0.016	0.016	0.019		
I_{xx} (kg.m^2)	I_{xy} (kg.m^2)	I_{xz} (kg.m^2)	I_{yy} (kg.m^2)	I_{yz} (kg.m^2)	I_{zz} (kg.m^2)
0.00607	0	-0.00015	0.00607	0.00015	0.00982

A comparative analysis of the trained model's performance in identifying the inertial parameters using input data from both real and simulation environments is provided in Fig. 6. The real and simulation data results are closely aligned, with only slight and acceptable differences. Notably, Fig. 6(b) exhibits improved estimation results for real-world data compared to the simulation data and even surpasses the performance of the model shown Fig. 6(a). On the other hand, Fig. 6(c) compares the model's performance in identifying the moment of inertia parameters using input data from both real and simulation environments. The model demonstrates highly accurate results when utilizing data from the real cobot input, with only slight and acceptable differences observed between the simulation and real-world results.

The results of the proposed method in a real-world environment are compared with the benchmark work presented

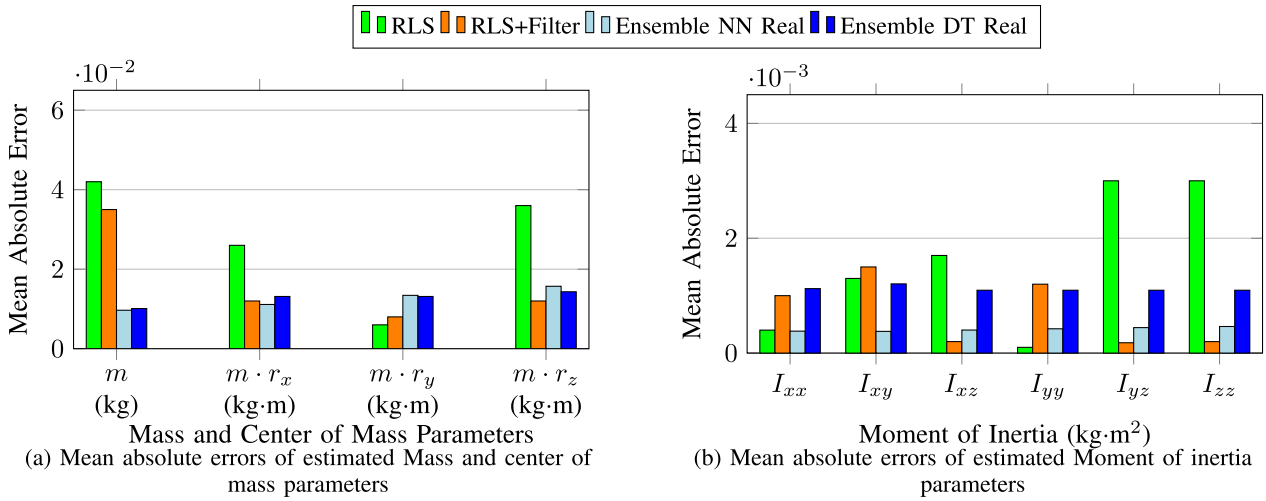


(a) Mass and center of mass results estimated by the model obtained from single stage training

(b) Mass and center of mass results estimated by the model obtained from two-stage training

(c) Moment of inertia results estimated by the model obtained from two-stage training

Fig. 6. Comparison among the results of ensemble algorithm estimation of the real cobot and simulation.



(a) Mean absolute errors of estimated Mass and center of mass parameters

(b) Mean absolute errors of estimated Moment of inertia parameters

Fig. 7. Comparison among the results of ensemble learning algorithm on the real cobot and the benchmark results.

in [7]. The following figures compare the mean absolute error of RLS method results and RLS with special filter results of the benchmark work with the proposed method results. As depicted in Fig. 7, the ensemble Neural Network (NN) excels over the RLS method in estimating m, $m \cdot r_x$, $m \cdot r_z$, I_{xx}, I_{xy}, I_{xz}, I_{yz}, and I_{zz} by 78%, 57.2%, 56.4%, 4.5%, 70.9%, 76.4%, 85.2%, and 84.6% respectively. It also outperforms the RLS method with a filter in estimating m, $m \cdot r_x$, I_{xx}, I_{xy}, and I_{yy} by 72.4%, 7.3%, 61.8%, 74.8%, and 64.7%, respectively.

Furthermore, the Ensemble Decision Tree (DT) surpasses the RLS method in estimating m, $m \cdot r_x$, $m \cdot r_z$, I_{xy}, I_{xz}, I_{yz}, and I_{zz} by 75.9%, 49.5%, 60.2%, 7.3%, 35.6%, 63.5%, and 63.4% respectively. Additionally, it outperforms the RLS with the filter method in estimating m, I_{xy}, and I_{yy} by 71.15%, 19.6%, and 8.7%, respectively. In conclusion, the proposed algorithm, using NN and DT as weak learners, achieves highly competitive results with the accuracy of RLS and RLS with a special filter. Notably, the proposed algorithm eliminates the need for calibration or special filtering on the real cobot platform to identify the payload's inertial parameters. This feature consistently saves the operating time of the cobot, making the proposed algorithm an efficient and reliable approach for real-world applications.

Comparing the estimation time of the proposed approach with the valid estimation time data of I_{zz} of the benchmark work, The proposed algorithm shows a very fast estimation response as shown in Fig. 8. It can estimate an accurate value for the I_{zz} from the first time step. So, the cumulative average value of the I_{zz} can be calculated from the first time step. The red line represents the cumulative average. It reaches approximately no variation after 2.85 seconds, meaning that the excitation trajectory's second half does not affect the estimated value. Unlike I_{zz} resulting from RLS in [7], it has been reported that the average can be calculated starting from the second half of the identification process time, which means after 3 seconds. In the first half, RLS results exhibit significant overshoots, which negatively impact the accuracy of the cumulative average value.

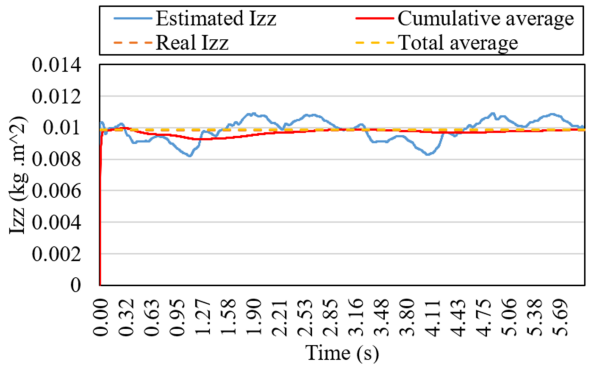


Fig. 8. I_{zz} estimation with Time.

VI. Conclusion

This letter has introduced a fast online method for identifying the inertial parameters of unknown payloads attached to cobots using a bagging ensemble machine learning approach employing NN and DT as weak learners. The proposed approach not only lifts the need for external sensors or custom calibration procedures, but also does not require custom filters as noisy acceleration data is not used. Notably the method has shown a rapid response and has sustained the accuracy of the estimation while transitioning from simulated data to real cobot data. The proposed method was benchmarked against the state-of-the-art method employing the recursive least squares using the widely used Franka Emika collaborative robot. The results showed that our proposed method using ensemble NN has significantly reduced the prediction error compared to RLS by 78%, 57.2% and 56.4% for the estimations of m, $m \cdot r_x$ and $m \cdot r_z$ respectively. Moreover, the proposed method using ensemble DT has reduced the estimation error with respect to RLS by 75.9%, 49.5% and 60.2% for the estimation of m, $m \cdot r_x$ and $m \cdot r_z$ respectively. Additionally, our proposed method estimated the payload parameters within the first time step, compared to up to 3 s for the benchmarked method to converge to a stable estimation, which makes it well suited for fast online identification. Future work will focus on optimizing the excitation method to enhance the identification process accuracy across variable paths with the objective of integrating online identification within task planning. In addition, a hybrid ensemble learning approach can be proposed, where NN and DT will be utilized together as weak learners in one ensemble model to further improve the accuracy of the estimation.

References

[1] S. R. Department, Mar. 2023, "Size of the collaborative (cobot) robot market worldwide in 2020 and 2021, with a forecast for 2022 to 2030," S. R. Department, [Online]. Available: https://www.statista.com/statistics/748234/global-market-size-collaborative-robots/

[2] S. Farsoni and M. Bonfè, "Complete and consistent payload identification during human-robot collaboration: A safety-oriented procedure," in *Proc. Hum.-Friendly Robot. : HFR: 14th Int. Workshop Hum.-Friendly Robot.*, 2022, pp. 15–28.

[3] C. Gaz and A. De Luca, "Payload estimation based on identified coefficients of robot dynamics—With an application to collision detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 3033–3040.

[4] T. Winiarski, S. Jarocki, and D. Seredyński, "Grasped object weight compensation in reference to impedance controlled robots," *Energies*, vol. 14, no. 20, 2021, Art. no. 6693.

[5] D. Buchholz, D. Kubus, I. Weidauer, A. Scholz, and F. M. Wahl, "Combining visual and inertial features for efficient grasping and bin-picking," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 875–882.

[6] N. Mavrakis, A. M. Ghalamzan, R. Stolkin, L. Baronti, M. Kopicki, and M. Castellani, "Analysis of the inertia and dynamics of grasped objects, for choosing optimal grasps to enable torque-efficient post-grasp manipulations," in *Proc. IEEE-RAS 16th Int. Conf. Humanoid Robots*, 2016, pp. 171–178.

[7] A. Kurdas, M. Hamad, J. Vorndamme, N. Mansfeld, S. Abdolshah, and S. Haddadin, "Online payload identification for tactile robots using the momentum observer," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 5953–5959.

[8] N. Mavrakis and R. Stolkin, "Estimation and exploitation of objects' inertial parameters in robotic grasping and manipulation: A survey," *Robot. Auton. Syst.*, vol. 124, 2020, Art. no. 103374.

[9] P. Nadeau, M. Giamou, and J. Kelly, "The sum of its parts: Visual part segmentation for inertial parameter identification of manipulated objects," in *Proc. IEEE Int. Conf. Robot. Automat.*, London, U.K., 2023, pp. 3779–3785, doi: 10.1109/ICRA48891.2023.10160394.

[10] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, "Inertial parameter identification in robotics: A survey," *Appl. Sci.*, vol. 11, no. 9, 2021, Art. no. 4303.

[11] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Rigid body load identification for manipulators," in *Proc. IEEE 24th Conf. Decis. Control*, 1985, pp. 996–1002.

[12] A. Bahloul, S. Tliba, and Y. Chitour, "Dynamic parameters identification of an industrial robot with and without payload," *IFAC-Papersonline*, vol. 51, no. 15, pp. 443–448, 2018.

[13] D. Kubus, T. Kroger, and F. M. Wahl, "On-line estimation of inertial parameters using a recursive total least-squares approach," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3845–3852.

[14] S. Farsoni, C. T. Landi, F. Ferraguti, C. Secchi, and M. Bonfè, "Real-time identification of robot payload using a multirate quaternion-based kalman filter and recursive total least-squares," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 2103–2109.

[15] D. Kubus, T. Kroger, and F. M. Wahl, "On-line rigid object recognition and pose estimation based on inertial parameters," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2007, pp. 1402–1408.

[16] G. Garofalo, N. Mansfeld, J. Jankowski, and C. Ott, "Sliding mode momentum observers for estimation of external torques and joint acceleration," in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 6117–6123.

[17] S. Salah, J. Sandoval, M. Ghiss, M. A. Laribi, and S. Zeghloul, "Online payload identification of a Franka Emika robot for medical applications," in *Advances in Service and Industrial Robotics: Results of RAAD*. Berlin, Germany: Springer, 2020, pp. 130–136.

[18] J. Sandoval and M. A. Laribi, "Tool compensation for a medical cobot-assistant," in *Gravity Compensation in Robotics*. Berlin, Germany: Springer, 2022, pp. 147–163.

[19] X. Li, J. Gu, X. Sun, J. Li, and S. Tang, "Parameter identification of robot manipulators with unknown payloads using an improved chaotic sparrow search algorithm," *Appl. Intell.*, vol. 52, pp. 10341–10351, 2022.

[20] L. Angel and J. Viola, "Payload estimation for a robotic system using unsupervised classification," in *Proc. IEEE 21st Symp. Signal Process., Images, Artif. Vis.*, 2016, pp. 1–5.

[21] A. Walawalkar, S. Heep, M. Frank, and C. Schindler, "Payload estimation in excavators using a machine learning based parameter identification method," in *Proc. Adv. Dyn. Veh. Roads Tracks: Proc. 26th Symp. Int. Assoc. Veh. Syst. Dyn.*, 2020, pp. 1670–1680.

[22] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Int. Workshop Mult. Classifier Syst.*, 2000, pp. 1–15.

[23] G. Brown, *Ensemble Learning*. Boston, MA, USA: Springer, 2010, pp. 312–320, doi: 10.1007/978-0-387-30164-8_252.

[24] G. Brown, J. Wyatt, R. Harris, and X. Yao, "Diversity creation methods: A survey and categorisation," *Inf. Fusion*, vol. 6, no. 1, pp. 5–20, 2005.

[25] J. Swevers, C. Ganseman, D. B. Tukel, J. De Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE Trans. Robot. Automat.*, vol. 13, no. 5, pp. 730–740, Oct. 1997.