

Real-time Inertial Parameter Identification of Floating-Base Robots Through Iterative Primitive Shape Division

Jiafeng Xu¹, Yu Zheng^{1†}, Xinyang Jiang¹, Sicheng Yang¹, Lingzhu Xiang¹, Zhengyou Zhang¹, *Fellow, IEEE*

Abstract—Dynamic models play a key role in robot motion generation and control and the identification of inertial parameters is a critical component for obtaining an accurate dynamic model of a robot. This paper presents a novel iterative primitive shape division method for the inertia parameter identification of floating-base robots. Describing a robot by a set of primitive shapes with uniform mass distributions, the method iteratively divides the primitive shapes into smaller ones and refines their masses, which quickly converges to yielding the true inertia parameters of the robot. This method guarantees the physical consistency of the obtained parameters, possesses a high computational efficiency for online deployment, and works with no contact force measurement. Furthermore, it can be used to estimate the position and magnitude of an external load applied to the robot. Simulations and experiments on a quadruped robot have been conducted to verify the effectiveness and efficiency of the proposed method.

I. INTRODUCTION

Dynamic models are of great importance for robot motion generation and control. An accurate dynamic model can improve the performance of a robot. Commonly, the model parameters of a robot can be obtained from its computer-aided design (CAD) model. For robots containing non-standard parts such as industrial computers, circuit boards, cameras, etc., however, CAD parameters often cannot reflect the true mass distribution characteristics. Moreover, when there are unknown loads on the robot, the inertia parameters from CAD or identified offline do not include those information and need to be identified in real time.

A. Related Work

Mayeda *et al.* [1] proposed an inertial parameter identification method based on the characteristics of specific motion, which is easily performed without decomposing the manipulator into parts. Atkeson *et al.* [2] recast the nonlinear Newton-Euler equations into a form linearly related to the inertial parameters and completed the inertia parameter identification of links and unknown load on MIT Serial Link Direct Drive Arm and PUMA 600 robots based on the classic least-squares method. Ayuzawa *et al.* [3] developed a method to identify the mass, center of mass (CoM), and inertia tensor of human body segments from a short-time measurement using the motion capture system and force plates. Nevertheless, the obtained results are not necessarily physically consistent and there can exist negative masses and non-positive definite inertia matrices. To solve this issue, Ayuzawa and his colleagues [4], [5] continued to improve

this and propose a real-time method to identify the physically consistent mass characteristics of various parts of the human body. The new method approximates a rigid body by finite mass points and represents physically consistent conditions as the positivity of mass of each point. However, this method often requires a large number of points to obtain relatively accurate inertia parameters of a rigid body, leading to a significant amount of computation time for a multi-link robot. Yamane [6] reorganized the Lagrangian dynamic equations into a linear expressions through regression matrix calculated by numerical derivative and realized the kinematic and dynamic parameters identification of the humanoid robot. This method needs joint torque and foot force sensors. Mistry *et al.* [7] developed a general framework for least-squares fitting of full-body inertial parameters of floating-base systems with only one set of partial force/torque sensors. The solution of the least-squares method is reprojected to meet the physical consistency constraints, which affects the accuracy of identified parameters and increases the computational load.

Lee and Park [8] proposed a geometric algorithm based on a coordinate-invariant definition of distance. It significantly improves the accuracy and robustness of parameter estimation, even for high-dimensional systems that are affected by measurement noises and ill-conditioned reference trajectories. However, the global solution can hardly be guaranteed because of the non-convexity of the problem and the computational burden is somewhat heavy. Subsequently, they [9] remedied the non-convexity through a convex formulation of the regularization term based on a second-order approximation of the Riemannian distance and optimized the computational complexity. Kwon *et al.* [10] introduced a unified kinodynamic model identification framework.

Other methods based on linear matrix inequalities [11], [12], Bayesian estimation [13], and maximum likelihood estimation [14] have also been developed for the inertial parameter identification of robots. Bayesian and maximum likelihood methods are robust to measurement noises and ill-conditioned data, but the models usually contain significant deterministic structural errors, which cannot be explained by random variables [15]. Olson [16] pointed out that the maximum likelihood method should be considered only when the measurement position and torque are noisy.

B. Our Work

In this paper, we propose a real-time method for the inertia parameter identification of floating-base robots. First, we represent every robot's link with a set of primitive shapes, where each primitive shape has a known size and position in

[†] denotes the corresponding author. e-mail: petezheng@tencent.com

¹The authors are with Tencent Robotics X, Shenzhen, Guangdong, China.

the link but an unknown mass to be determined. Then, the inertia parameters of every link can be calculated from its primitive shapes with all the shape's masses as unknowns and the parameter's physical consistency can be imposed simply by restricting the masses to the positive. In this way, the identification of the robot's inertia parameters is reduced to the identification of these unknown masses, which can be written as a simple quadratic programming (QP) problem. To refine the identification result, we present an iterative procedure to adaptively and gradually divide some of the primitive shapes into smaller ones, which guarantees that the new identified parameters at every iteration are better. As the QP problem to be solved at every iteration is very simple, the iterative procedure converges quickly, ensuring the high efficiency of the proposed method.

The main contributions of this paper include

- An iterative primitive shape division method is proposed to identify the physically consistent inertia parameters of a floating-base robot (Sections III-C and III-D);
- The proposed method can work with no contact force measurement and use only the joint angle and torque measurements and the IMU data, which are usually available on a floating-base robot (Section III-B);
- The proposed method has been tested on a quadruped robot in both simulation and real experiments, showing a superior performance in computational accuracy and efficiency over some of the existing methods, and it has also been applied to the unknown load identification on the robot (Section IV).

II. DYNAMIC MODEL

The floating-base framework provides a general representation of the rigid-body system unattached to the world and it can be attached to an inertial frame with six degrees of freedom (DoFs). The dynamics of a floating-base robot, having M rigid bodies and N_G DoFs, can be described as

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}^T(\mathbf{q})\mathbf{f} \quad (1)$$

where \mathbf{q} , $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}} \in \mathbb{R}^{N_G}$ denote the generalized position, velocity, and acceleration, respectively, $\mathbf{H}(\mathbf{q}) \in \mathbb{R}^{N_G \times N_G}$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{N_G \times N_G}$ is the centrifugal and Coriolis matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^{N_G}$ is the gravitational force vector, $\mathbf{S} = [\mathbf{0}_{(N_G-6) \times 6} \quad \mathbf{I}_{(N_G-6) \times (N_G-6)}]$ is the actuated joint selection matrix, $\boldsymbol{\tau}$ is the joint torque vector, $\mathbf{J} \in \mathbb{R}^{N_f \times N_G}$ is the Jacobian of N_f constraints, and $\mathbf{f} \in \mathbb{R}^{N_f}$ is the constraint force vector.

The equation (1) of motion can be rewritten in a form that is linear to the inertia parameters [2], i.e.,

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\Phi = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}^T(\mathbf{q})\mathbf{f} \quad (2)$$

where $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{N_G \times 10M}$ is the regression matrix, $\Phi = [\Phi_1^T \quad \Phi_2^T \quad \dots \quad \Phi_M^T]^T \in \mathbb{R}^{10M}$ is the vector of inertial parameters of M links and each link has 10 inertia parameters constituting Φ_i for $i = 1, 2, \dots, M$ as

$$\Phi_i = [m_i \quad m_i r_{xi} \quad m_i r_{yi} \quad m_i r_{zi} \quad I_{xxi} \quad I_{xyi} \quad I_{xzi} \quad I_{yyi} \quad I_{yzi} \quad I_{zzi}]^T \quad (3)$$

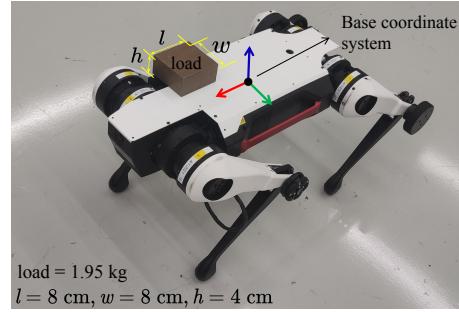


Fig. 1. A quadruped robot developed by Tencent Robotics X carrying an external load.

where m_i is the mass, r_{xi} , r_{yi} , r_{zi} are the coordinates of the CoM, and I_{xxi} , I_{xyi} , I_{xzi} , I_{yyi} , I_{yzi} , I_{zzi} are the components of the inertia tensor of link i with respect to the coordinate frame at the joint associated with link i .

III. INERTIA PARAMETER IDENTIFICATION

A. General Identification Problem Formulation

In the general identification problem, the generalized coordinates, joint torques, and external forces acted on the robot need to be measured by sensors such that the regression matrix \mathbf{Y} can be calculated through numerical differentiation [6] or chain derivation [17] methods in every control cycle. Collecting the measurement data over excitation trajectories and assembling (2) for all the measurements, we obtain

$$\tilde{\mathbf{Y}}\Phi = \tilde{\mathbf{F}} \quad (4)$$

where $\tilde{\mathbf{Y}}$ and $\tilde{\mathbf{F}}$ are the concatenations of $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ and $\mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}^T(\mathbf{q})\mathbf{f}$ for all the measurements of \mathbf{q} , $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, $\boldsymbol{\tau}$, and \mathbf{f} over the excitation trajectories, respectively. Then, the inertial parameter identification problem can be written as

$$\arg \min_{\Phi} (\tilde{\mathbf{Y}}\Phi - \tilde{\mathbf{F}})^T \mathbf{W} (\tilde{\mathbf{Y}}\Phi - \tilde{\mathbf{F}}) \quad (5)$$

where \mathbf{W} is the weight matrix to regulate different components of $\tilde{\mathbf{Y}}\Phi - \tilde{\mathbf{F}}$ as they have different units. Furthermore, to ensure the physical consistency of the identified parameters, the inertia parameters of each link must satisfy

$$m_i > 0, \quad \mathbf{I}_i^B \succ 0 \quad (6)$$

where \mathbf{I}_i^B is the body inertia tensor matrix of link i .

However, not all the floating-base robots are equipped with full force/torque sensors and the full construction of (4) is sometimes infeasible. Also, solving (5) subject to (6) is a sequential QP problem [18], whose computation time would not allow the identification to be performed in real time. The rest of this section is to solve these issues.

B. Identification with Only Joint Torque Sensing

When the direct contact force measurement is unavailable, as on our quadruped robot shown in Fig. 1, it is still possible to identify the inertial parameters with only the measurement

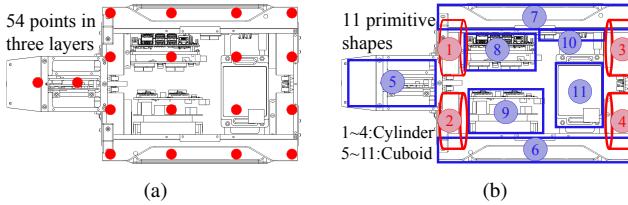


Fig. 2. Representation of the robot base using (a) 54 mass points distributed in three layers or (b) 11 primitive shapes including 4 cylinders and 7 cuboids.

of joint torques [7]. Following the work [7], we first conduct the QR decomposition of \mathbf{J}^T as

$$\mathbf{J}^T = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{0}_{N_G - N_f} \end{bmatrix} \quad (7)$$

where $\mathbf{Q} \in \mathbb{R}^{N_G \times N_G}$ is an orthogonal matrix and $\mathbf{R} \in \mathbb{R}^{N_f \times N_f}$ is an upper triangular matrix of full rank. Premultiplying (2) by \mathbf{Q}^T and combining (7), we then obtain

$$\mathbf{Q}^T \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \Phi = \mathbf{Q}^T \mathbf{S}^T \boldsymbol{\tau} + \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} \mathbf{f} \quad (8)$$

Extracting the last $N_G - N_f$ rows of (8), which does not depend on the contact forces \mathbf{f} , yields

$$\mathbf{N}_u \mathbf{Q}^T \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \Phi = \mathbf{N}_u \mathbf{Q}^T \mathbf{S}^T \boldsymbol{\tau} \quad (9)$$

where $\mathbf{N}_u = [\mathbf{0}_{(N_G - N_f) \times N_f} \quad \mathbf{I}_{(N_G - N_f) \times (N_G - N_f)}]$ is the row selection matrix.

Using (9) instead of (2) and following the same procedure, we can establish a similar equation to (4) and optimization problem to (5) for the inertia parameter identification with only joint torque sensing. Without introducing new notations, we continue to the notations $\tilde{\mathbf{Y}}$ and $\tilde{\mathbf{F}}$ in the following discussion while they are now derived from $\mathbf{N}_u \mathbf{Q}^T \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ and $\mathbf{N}_u \mathbf{Q}^T \mathbf{S}^T \boldsymbol{\tau}$ in (9) instead of the counterparts in (2).

C. Primitive Shape Representation

The physical consistency condition (6) imposes nonlinear constraints on and makes the optimization problem (5) more complex. By approximating a rigid body with finite mass points, as illustrated in Fig. 2a, the problem can be reduced to a QP problem with only the boundary constraints on the mass of every point [5]. However, a mass point does not contain the volume and inertia tensor information, and to accurately approximate the inertia of a rigid body, a number of mass points are needed, which results in a large-scale QP problem. Inspired by [5], in this paper we propose using a set of primitive shapes (e.g., ellipsoids, cuboids, cylinders, polyhedra, and etc.) to represent a rigid body, as depicted in Fig. 2b, where every primitive shape has a known size and position in the rigid body and the mass distribution within a primitive shape is assumed to be uniform. One can refer to the CAD model of a robot to set the initial types, numbers, sizes, and positions of primitive shapes for every robot link, as we do for the base of our quadruped robot in Fig. 2b.

To make it easier for readers to understand the proposed method, without loss of generality, we take cuboids as an example hereinafter and assume that link i of a robot is

represented by n_i cuboids and the principal axes of every cuboid are aligned with the coordinate frame of link i . For cuboid j on link i , let $m_{i,j}$ denote its mass, $l_{i,j}, w_{i,j}, h_{i,j}$ its side lengths, and $\mathbf{c}_{i,j}$ its CoM position in the link coordinate frame. Here it should be noted that only $m_{i,j}$'s are unknowns to be determined in the identification while $l_{i,j}, w_{i,j}, h_{i,j}$, and $\mathbf{c}_{i,j}$'s are all known when we set the cuboids. Then, the inertial parameters of cuboid j expressed in the coordinate frame of link i can be written as

$$\Phi_{i,j} = [m_{i,j} \quad m_{i,j} \mathbf{c}_{i,j}^T \quad \text{vec}(\mathbf{I}_{i,j}^B - m_{i,j} \hat{\mathbf{c}}_{i,j}^2)]^T \quad (10)$$

where $\hat{\mathbf{c}}_{i,j}$ is the skew-symmetric matrix generated by $\mathbf{c}_{i,j}$, $\text{vec}(\mathbf{I}_{i,j}^B - m_{i,j} \hat{\mathbf{c}}_{i,j}^2)$ converts the symmetric inertia tensor matrix $\mathbf{I}_{i,j}^B - m_{i,j} \hat{\mathbf{c}}_{i,j}^2$ into a 6-D vector as in (3), and $\mathbf{I}_{i,j}^B$ is the body inertia tensor of cuboid j , i.e.,

$$\mathbf{I}_{i,j}^B = \frac{m_{i,j}}{12} \begin{bmatrix} w_{i,j}^2 + h_{i,j}^2 & 0 & 0 \\ 0 & l_{i,j}^2 + h_{i,j}^2 & 0 \\ 0 & 0 & l_{i,j}^2 + w_{i,j}^2 \end{bmatrix}. \quad (11)$$

Extracting the common factor m_j , we can rewrite (10) as

$$\Phi_{i,j} = m_{i,j} \mathbf{U}_{i,j} \quad (12)$$

where $\mathbf{U}_{i,j} = [1 \quad \mathbf{c}_{i,j}^T \quad \text{vec}(\mathbf{I}_{i,j}^B / m_{i,j} - \hat{\mathbf{c}}_{i,j}^2)]^T$. From (11) we see that $\mathbf{U}_{i,j}$ depends only on known parameters $l_{i,j}, w_{i,j}, h_{i,j}, \mathbf{c}_{i,j}$ and is known. By adding all n_i cuboids for link i together, the inertia parameters of link i can be expressed as

$$\Phi_i = \sum_{j=1}^{n_i} m_{i,j} \mathbf{U}_{i,j} = \mathbf{U}_i \mathbf{m}_i \quad (13)$$

where $\mathbf{U}_i = [\mathbf{u}_{i,1} \quad \mathbf{u}_{i,2} \quad \dots \quad \mathbf{u}_{i,n_i}]$ and $\mathbf{m}_i = [m_{i,1} \quad m_{i,2} \quad \dots \quad m_{i,n_i}]^T$. We can further write the vector Φ of inertia parameters of the robot as

$$\Phi = \mathbf{U} \mathbf{m} \quad (14)$$

where $\mathbf{U} = \text{diag}(\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_M) \in \mathbb{R}^{10M \times N_m}$, $\mathbf{m} = [\mathbf{m}_1^T \quad \mathbf{m}_2^T \quad \dots \quad \mathbf{m}_M^T]^T \in \mathbb{R}^{N_m}$, and $N_m = \sum_{i=1}^M n_i$ is the number of unknown parameters.

Substituting (14) into (5) leads to a new identification form

$$\arg \min_{\mathbf{m}} (\tilde{\mathbf{Y}} \mathbf{U} \mathbf{m} - \tilde{\mathbf{F}})^T \mathbf{W} (\tilde{\mathbf{Y}} \mathbf{U} \mathbf{m} - \tilde{\mathbf{F}}) \quad (15)$$

where the physical consistency constraints are simplified to $m_{i,j} > 0$. The problem (15) is a simple QP problem with N_m variables, for which several off-the-shelf solvers are available and we use qpOASES [19] in this paper.

D. Adaptive Iterative Division

Since every primitive shape has a uniform mass distribution, the initial setting of primitive shapes may not be able to accurately reflect the mass distribution of the robot. Instead of starting with a large set of primitive shapes, which will induce a large-scale QP problem, we propose an iterative procedure to adaptively divide primitive shapes into smaller ones and refine their masses such that the iteration can quickly converge to yielding the accurate inertia parameters of the robot and the number of primitive shapes and the sizes of QP problems can be kept small, as illustrated in Fig. 3.

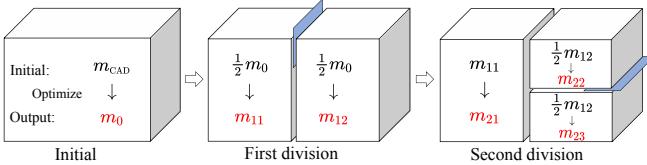


Fig. 3. Illustration of iterative shape division. At every iteration (division), the masses of remaining and divided primitive shapes are recomputed by solving a QP problem as (15). The initial condition of the QP problem is taken to be the result of the previous iteration to expedite the computation.

Algorithm 1 Iterative Shape Division

```

1: Set an initial set of primitive shapes
2: Let  $\mathbf{U}_0$  be the matrix  $\mathbf{U}$  as used in (14)
3: Let  $\mathbf{m}_0$  be the optimal solution of (15)
4:  $k \leftarrow 0$ 
5: repeat
6:   Calculate  $v_{i,j}$ 's by (16) for every primitive shape
7:   Divide the primitive shape with the maximal  $v_{i,j}$ 
8:    $k \leftarrow k + 1$ 
9:   Recompute  $\mathbf{U}_k$  and  $\mathbf{m}_k$ 
10: until  $\|\mathbf{U}_k \mathbf{m}_k - \mathbf{U}_{k-1} \mathbf{m}_{k-1}\| < \epsilon$ 
Output:  $\mathbf{U}_k$  and  $\mathbf{m}_k$ 
```

In this paper, we choose one primitive shape and divide it into two at every iteration, which can be extended to including more complex operations such as dividing more shapes into multiple parts or merging small shapes. Considering that it would be more beneficial to divide a primitive shape with large mass or volume, we define a division criterion as

$$v_{i,j} = k_1 m_{i,j} V_{i,j} + k_2 m_{i,j} / V_{i,j} \quad (16)$$

where $V_{i,j}$ is the volume of primitive shape j on robot link i and k_1 and k_2 are the weights. At every iteration, we simply choose the primitive shape with the maximal $v_{i,j}$ to divide.

After every division, a new QP problem in the form of (15) with one more mass variable will be solved to recalculate the optimal mass values of all the primitive shapes. The initial mass values as well as the sizes and positions of newly-generated shapes from division can be calculated from their original shape to expedite the solution of the new QP problem, as marked in Fig. 3. With this initial condition, we can also prove that the initial objective value of the new QP problem equals the optimal objective value of the QP problem at the previous iteration. Then, the optimal objective value of the new QP problem will not be greater than the previous one, which implies that the optimal objective value is monotonically decreasing as the division procedure iterates. As the optimal objective value is bounded below by zero, the procedure is guaranteed to converge. **Algorithm 1** describes the pseudo-code of this procedure.

IV. EXPERIMENTS AND DISCUSSIONS

A. Experimental Setup

The proposed method has been implemented in C++ and experiments have been conducted on our quadruped robot

(Fig. 1). The robot has 18 DoFs and is equipped with an IMU, joint encoders, and a Pico-ITX form factor x86 single-board computer with the Intel i7-8665UE processor running the Linux 5.4 NOHZ_FULL kernel and Ubuntu 20.04. An on-line Butterworth low-pass filter is implemented to eliminate the high-frequency noise in the measured joint angles prior to the calculation of joint velocities and accelerations. A state estimator similar to the work [20] and an MPC controller for the robot to execute reference motions are also implemented and run at 1 kHz. The joint torques are estimated from motor currents during the experiments. The measurement data for the inertia parameter identification are collected every 2 ms.

We conducted both simulation and real experiments on the proposed method in two cases:

- 1) *Base Link Identification:* We let the robot trot for 5 s and identify the inertia parameters of the base link every 20 ms with 10 sets of measurement data collected within the 20 ms. During the 5 s, therefore, the identification is performed 250 successive times. To deal with the noise and abnormal measurement data, we take a weighted average of the current and last identified parameters as the final result of the current identification, which acts as a low-pass filter and can be written as

$$\Phi_t = \alpha \Phi_t + (1 - \alpha) \Phi_{t-1} \quad (17)$$

where α is the filter coefficient and $\alpha = 0.5$ in the experiments. In this case, we set the primitive shapes as shown in Fig. 2b and do not further divide them. The inertia parameters of the other links are taken to be the CAD values permanently as they are light-weight.

- 2) *Unknown Load identification:* We place a block weighing 1 kg with a size of $4 \times 4 \times 4$ cm³ in simulation and 1.95 kg with a size of $8 \times 8 \times 4$ cm³ in the real experiment at a specific position within an area of 0.4×0.2 m² on the robot's base, as shown in Fig. 1, while the robot is trotting. The load information is unknown to the robot. Starting with a cuboid with a height of 4 cm covering the area, we use the proposed method to identify the inertia parameters of the load with respect to the robot's base. In this case, a chosen cuboid is divided from its longest side and the inertia parameters of the robot are kept constant.

We compare the proposed method with three existing ones:

- Nonlinear: solving (5) subject to the nonlinear physical consistency constraints (6) by Ceres Solver [21];
- Linear: solving (5) with only the positive-mass constraint as a QP problem, sacrificing the physical correctness of the result for faster computation;
- Points: solving the QP problem formulated in [5] with the points assigned as in Fig. 2a.

B. Simulation Results

- 1) *Base Link Identification:* Fig. 4 plots the identified inertia parameters of the base link by the four methods in comparison with the CAD values, which are the ground truth since the CAD model is used in simulation. It can be seen

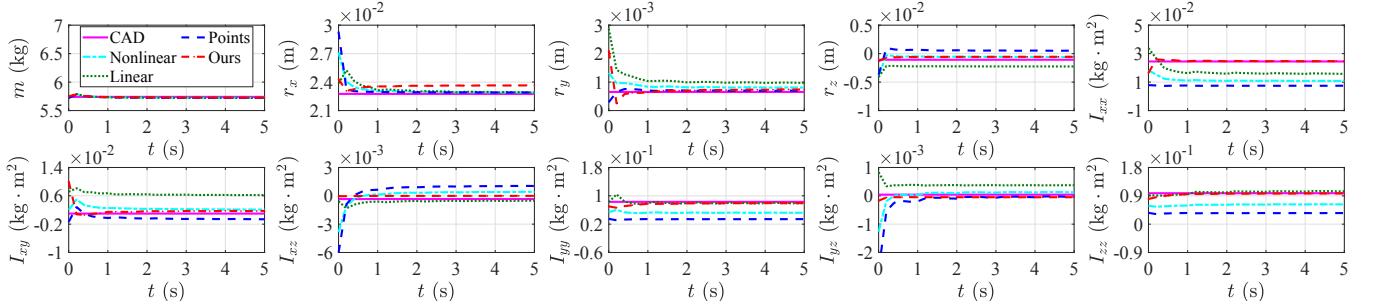


Fig. 4. Comparison of the inertia parameters obtained on the simulated robot by four methods and the CAD parameters as the ground truth.

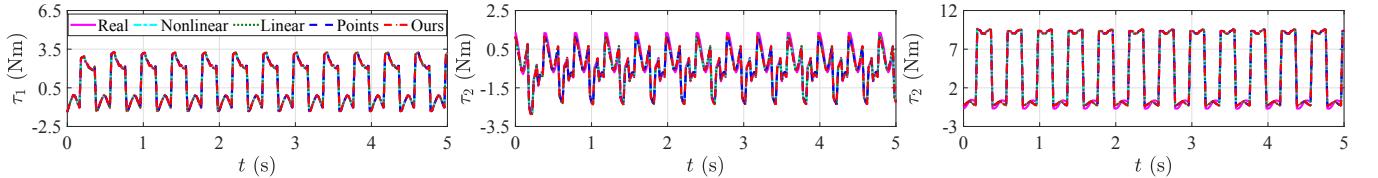


Fig. 5. Three projected joint torques in the null space of contact Jacobian calculated with the four sets of identified parameters in comparison with the real values obtained directly from the measured joint torques on the simulated robot.

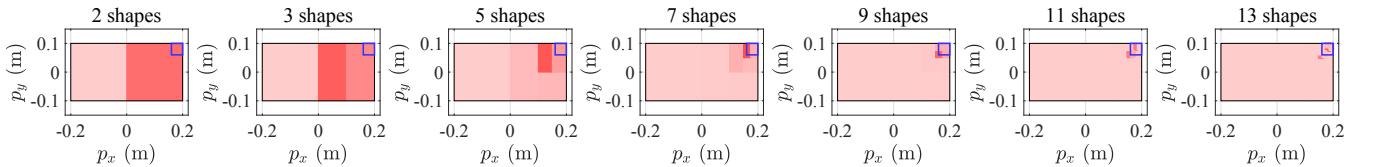
Fig. 6. Iterative division of a $0.4 \times 0.2 \text{ m}^2$ area to identify an external load weighting 1 kg with a size of $4 \times 4 \times 4 \text{ cm}^3$ (marked as the blue square) on the simulated robot. The shaded rectangles with different intensities of red color represent the primitive shapes (cuboids in this example) during the iteration and the color intensity is proportional to the cuboid's mass.

TABLE I. RESULTS OF BASE LINK IDENTIFICATION

Algorithm	Simulation			Experiment		
	t_c	v_{obj}	e_p	t_c	v_{obj}	e_p
Nonlinear	3.086	0.6542	0.0181	3.541	2.2988	1.4258
Linear	0.032	0.6566	0.0218	0.048	2.8716	1.8833
Points	0.483	0.6956	0.0838	0.436	2.3685	1.5029
Ours	0.045	0.6518	0.0147	0.052	2.5781	1.4950

 t_c —average computation time (ms) v_{obj} —average optimal objective value of (15) e_p —average error between the identified and CAD parameters

that the result of our method is very close to ground truth. Fig. 5 shows the projected joint torques calculated with the identified parameters, which cross-validate the correctness of the identification. The average computation time, optimal objective function, and difference from the CAD parameters evaluated by the 2-norm are collected in the left half of Table I, which show that the proposed method outperforms the other methods in both accuracy and efficiency.

2) *Unknown Load Identification*: Fig. 6 shows the iteration of the proposed division procedure used to identify the external load, by which the initial cuboid keeps being divided into smaller cuboids and the mass distribution over the cuboids is gradually concentrated on the true location of the load. The optimal objective value, the computation time, and the error in the identified inertia parameters of the load compared with the ground truth versus the times of division are plotted in Fig. 7, which shows that the proposed

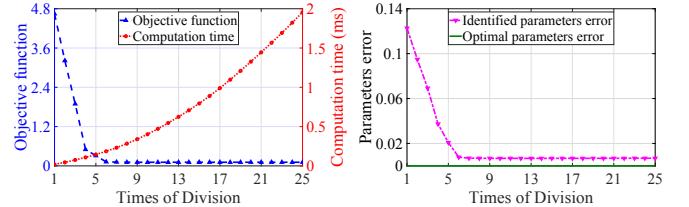


Fig. 7. Optimal objective value, computation time, and error in the identified parameters evaluated by the 2-norm versus the times of division in simulation. The iteration has converged after the 7-th division.

procedure quickly converges after several iterations.

C. Experiment Results

1) *Base Link Identification*: Fig. 8 shows the identification results on the real robot. Although the CAD parameters can no longer reflect the true values, we still take them as a common reference in comparing all the results of the four methods, since we do not have a better reference on hand. It can be seen that the result of the proposed method is comparable to the others, which is cross-validated by the fitting of the projected joint torques with the identified parameters, as shown in Fig. 9. Some statistics results are provided in the right half of Table I, which again show the superior performance of the proposed method.

2) *Unknown Load Identification*: In the load identification, we use the inertia parameters of the robot's base link

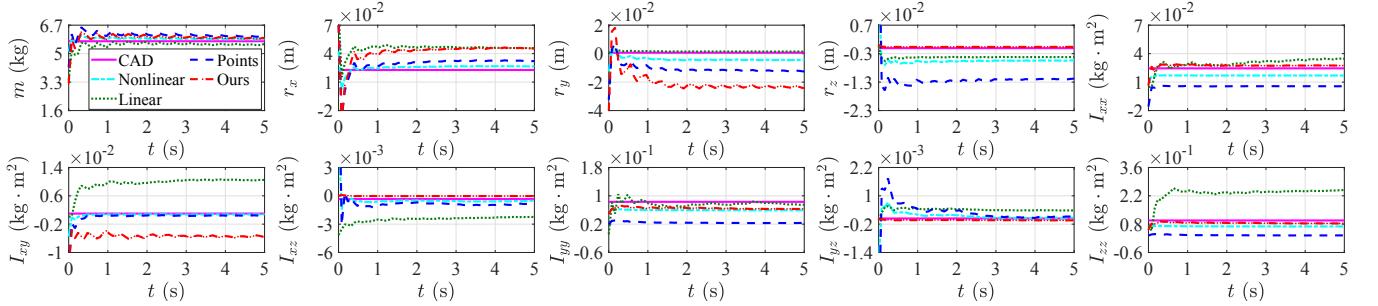


Fig. 8. Comparison of the inertia parameters obtained on the real robot by four methods and the CAD parameters as a common reference.

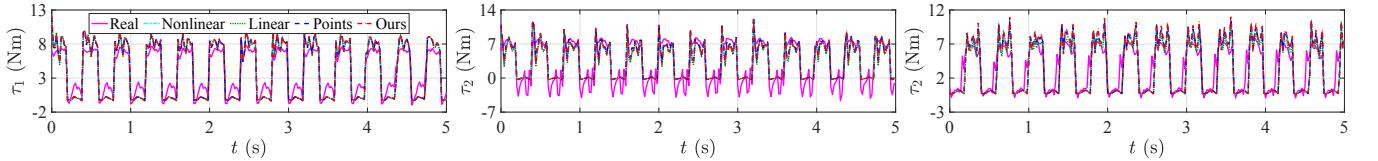


Fig. 9. Three projected joint torques in the null space of contact Jacobian calculated with the four sets of identified parameters in comparison with the real values obtained directly from the measured joint torques on the real robot.

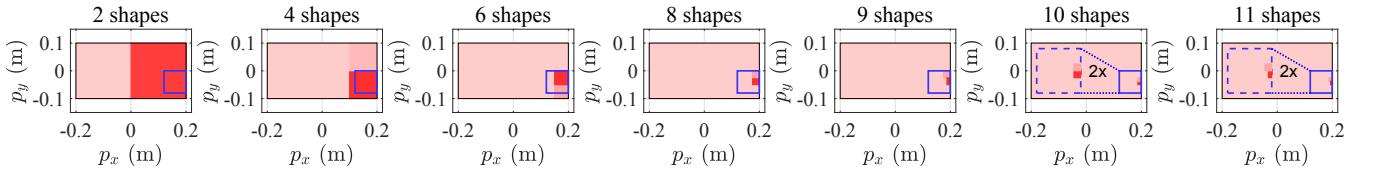
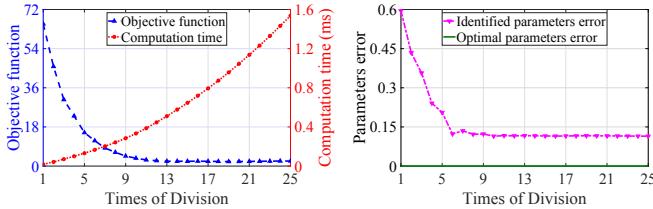
Fig. 10. Iterative division of a $0.4 \times 0.2 \text{ m}^2$ area to identify an external load weighting 1.95 kg with a size of $8 \times 8 \times 4 \text{ cm}^3$ (marked as the blue square) on the real robot. The shaded rectangles with different intensities of red color represent the primitive shapes (cuboids in this example) during the iteration and the color intensity is proportional to the cuboid's mass.

Fig. 11. Optimal objective value, computation time, and error in the identified parameters evaluated by the 2-norm versus the times of division in the real experiment. The iteration has converged after the 11-th division.

identified above. Fig. 10 reveals the iterative process of identifying the inertia parameters of the load with the proposed division procedure, which again shows the convergence of the mass distribution to the load. The performance of the procedure versus the times of division is shown in Fig. 11.

D. Discussions

From Fig. 5 and 9, we notice that the difference between the joint torques calculated with the identified parameters and the actual ones in the real experiment is greater than that in simulation. This is mainly because the actual measurement data for the identification contain more noises and errors. For example, with the absence of joint torque sensors, the actual joint torques are currently calculated from motor currents and can be affected by friction or other time-varying or unmodeled factors. Also, the joint accelerations are obtained by numerical difference, which has numerical errors. In the

unknown load identification, similarly, the parameter error in the real case (Fig. 11) is bigger than that in the simulated case (Fig. 7). In addition to the errors in the measurement data, another reason here is that the robot model used in the real experiment has errors, which will propagate into the identified load. Despite errors, the proposed method can still reasonably identify the inertia parameters and is very fast.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a method for the real-time inertia parameter identification of floating-base robots based on an iterative primitive shape division procedure. Approximating each link of a robot with primitive shapes, the identification problem can be reduced to the identification of masses of the shapes, which can be quickly computed as a QP problem and guarantees the physical consistency of the obtained inertia parameters. Through the adaptive iterative division of shapes, the identified parameters can be refined and quickly converge to the true values. The effectiveness and efficiency of the proposed method have been verified on a quadruped robot in both simulation and real experiments.

Currently, we only consider dividing the primitive shapes. In terms of the future work, we will try more sophisticated operations on the primitive shapes (e.g., adding, deleting, and merging) as well as other strategies and criteria for choosing appropriate operations at every iteration such that the method can be more accurate and efficient. Moreover, we will try the method with more motion trajectories on more robots.

REFERENCES

- [1] H. Mayeda, K. Osuka, and A. Kangawa, "A new identification method for serial manipulator arms," *IFAC Proceedings Volumes*, vol. 17, no. 2, pp. 2429–2434, 1984.
- [2] 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.
- [3] K. Ayusawa, Y. Nakamura, and G. Venture, "Optimal estimation of human body segments dynamics using realtime visual feedback," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 1627–1632.
- [4] K. Ayusawa and Y. Nakamura, "Identification of base parameters for large-scale kinematic chains based on physical consistency approximation by polyhedral convex cones," in *ROMANSY 18 Robot Design, Dynamics and Control*. Springer, 2010, pp. 91–98.
- [5] K. Ayusawa, G. Venture, and Y. Nakamura, "Real-time implementation of physically consistent identification of human body segments," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 6282–6287.
- [6] K. Yamane, "Practical kinematic and dynamic calibration methods for force-controlled humanoid robots," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2011, pp. 269–275.
- [7] M. Mistry, S. Schaal, and K. Yamane, "Inertial parameter estimation of floating base humanoid systems using partial force sensing," in *2009 9th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2009, pp. 492–497.
- [8] T. Lee and F. C. Park, "A geometric algorithm for robust multibody inertial parameter identification," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2455–2462, 2018.
- [9] T. Lee, P. M. Wensing, and F. C. Park, "Geometric robot dynamic identification: A convex programming approach," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 348–365, 2019.
- [10] J. Kwon, K. Choi, and F. C. Park, "Kinodynamic model identification: A unified geometric approach," *IEEE Transactions on Robotics*, 2021.
- [11] 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.
- [12] 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.
- [13] J.-A. Ting, M. N. Mistry, J. Peters, S. Schaal, and J. Nakanishi, "A bayesian approach to nonlinear parameter identification for rigid body dynamics," in *Robotics: Science and systems*. Philadelphia, USA, 2006, pp. 32–39.
- [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, no. 5, pp. 730–740, 1997.
- [15] N. Ramdani and P. Poignet, "Robust dynamic experimental identification of robots with set membership uncertainty," *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 2, pp. 253–256, 2005.
- [16] M. M. Olsen, J. Swevers, and W. Verdonck, "Maximum likelihood identification of a dynamic robot model: Implementation issues," *The international Journal of robotics research*, vol. 21, no. 2, pp. 89–96, 2002.
- [17] M. Gabicci, A. Bracci, D. De Carli, M. Fredianelli, and A. Bicchi, "Explicit lagrangian formulation of the dynamic regressors for serial manipulators," in *Proceedings of the XIX Aimeta Congress*, 2009.
- [18] V. Mata, F. Benimeli, N. Farhat, and A. Valera, "Dynamic parameter identification in industrial robots considering physical feasibility," *Advanced Robotics*, vol. 19, no. 1, pp. 101–119, 2005.
- [19] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: a parametric active-set algorithm for quadratic programming," *Math. Prog. Comp.*, vol. 6, no. 4, pp. 327–363, 2014. [Online]. Available: <https://github.com/coin-or/qpOASES>
- [20] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots: consistent fusion of leg kinematics and imu," *Robotics*, vol. 17, pp. 17–24, 2013.
- [21] S. Agarwal, K. Mierle, and Others, "Ceres solver," <http://ceres-solver.org>.