2011 11th IEEE-RAS International Conference on Humanoid Robots
Bled, Slovenia, October 26-28, 2011

Practical Kinematic and Dynamic Calibration Methods for Force-Controlled Humanoid Robots

Katsu Yamane

Abstract— **This paper presents methods and experimental results regarding the identification of kinematic and dynamic parameters of force-controlled biped humanoid robots. We first describe a kinematic calibration method to estimate joint angle sensor offsets. The method is practical in the sense that it only uses joint angle and link orientation sensors, which most humanoid robots are equipped with. The basic idea is to solve an optimization problem that represents a kinematic constraint that can be easily enforced, such as placing both feet flat on floor. We then present two methods to identify physically consistent mass and local center of mass parameters even when obtaining enough excitation is difficult, as is always the case in humanoid robots. We demonstrate by experiment that these methods give good identification results even when the regressor has a large condition number. Moreover, we show that gradient-based optimization performs better than the least-square method in many cases.**

I. Introduction

Compared to traditional manipulators, identification of kinematic and dynamic parameters of floating-base humanoid robots can be difficult for a number of reasons. In particular, it is often difficult to get a well-conditioned regressor not only because there are many parameters to identify, but also because the robot has to maintain balance throughout the data collection process. Furthermore, global position and orientation measurements may be inaccurate or even unavailable.

This paper addresses the problem of identifying kinematic and dynamic parameters of force-controlled humanoid robots. Section II describes a method for identifying joint angle sensor offsets that does not require external measurements such as motion capture. In Section III, we present two methods for inertial parameter identification that is easy to implement and robust against poor excitation data.

We use the Carnegie Mellon University/Sarcos Humanoid Robot (Fig. 1) as a test case. The robot has 31 joints in the arms, legs and trunk. Each joint is actuated by a hydraulic actuator with a force sensor to measure the actuator force, which can be converted to the joint torque by multiplying the moment arm. Joint angles are measured by potentiometers. There is also a six-axis force sensor at each foot to measure the ground contact force. An inertial measurement unit (IMU) is attached to the pelvis link.

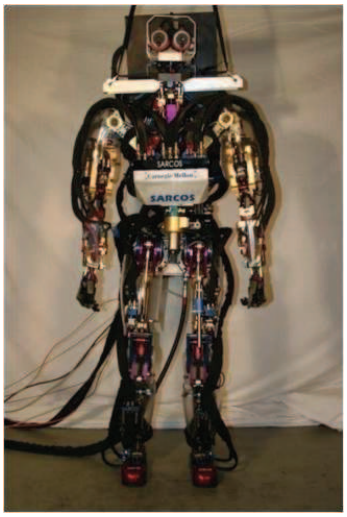

Fig. 1. CMU/Sarcos humanoid robot used for the experiments.

II. Kinematic Calibration

A. Motivation

The purpose of kinematic calibration described in this section is to estimate the offsets in joint angle sensors. In our robot, it is known that the potentiometers may slide with respect to the link to which they are supposed to be fixed when a large torque is applied, especially at the four ankle joints (flexion/extension and adduction/abduction) that do not have enough space to obtain good fixture. Identifying the joint angle sensor offsets usually requires external measurement devices such as motion capture systems. However, setting up such systems is tedious if the recalibration needs to be done frequently.

This section describes an algorithm that can identify the joint angle offsets by only measuring the orientations of one or more links in addition to the joint angles. Link orientations can be measured using IMUs, which most humanoid robots are equipped with for balance control. The algorithm therefore does not require external measurements for calibration.

The key idea is to move the joints randomly under simple kinematic constraints such as placing the feet flat on the floor. If the joints are backdrivable with reasonable magnitude of external force, we can collect the data by moving the robot's joints by hand. After collecting sufficient variation of poses, we can compute the offsets by solving an optimization problem with a cost function that decreases as the kinematic constraints are better satisfied. For example, if the feet are in flat contact with the ground, the cost function would be the height variation of the sole vertices. Note that we cannot use the height itself because IMUs do not give accurate height information.

Similar methods can be found for autonomous kinematic

K. Yamane is with Disney Research, Pittsburgh, 4720 Forbes Ave. Suite 110, Pittsburgh, PA 15213. kyamane@disneyresearch.com

calibration of fixed-base manipulators where the joint angles are measured while the endpoint moves on the surface of a fixed plane [1]–[3]. We have to use a different formulation because the Cartesian position of the root joint is unknown, and therefore we cannot assume co-planar contact points across frames. Instead, we determine the kinematic parameters so that multiple contact points in each frame form a single plane.

In our experiment, the we keep both feet in flat contact with the floor during data collection and the optimization problem is summarized as follows:

- Inputs (measurements): orientations of one or more links and joint angles at various sample poses.
- Variables: orientation of the root joint at each sample pose and joint angle sensor offset of each joint.
- Cost function: orientation error and variation of sole vertex heights.

B. Formulation

Consider the case where we can obtain orientation data of M $(M \geq 1)$ links, and there are N rotational joints whose angles affect the orientations of the feet and/or the links with IMUs. In the following discussion, we assume that we want to identify the offsets of all N joints for simplicity of representation, although it is trivial to exclude some of the joints from offset identification while using their measurements.

Let us assume that we have collected K sample poses that satisfy the kinematic constraint. The measurements available for the k-th sample are the orientation data from the IMUs $\hat{\boldsymbol{R}}_k^m$ $(m = 1, 2, \ldots, M)$ and joint angles $\hat{\boldsymbol{\theta}}_k \in \boldsymbol{R}^N$. The parameters to be identified are the joint offsets $\Delta\boldsymbol{\theta}$ and the root orientation at each sample $\boldsymbol{q}_k$.

By solving the optimization problem, we obtain $\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_K$ and $\Delta\boldsymbol{\theta}$ that minimizes the cost function Z, which is chosen as

$$Z = \frac{1}{2}\sum_k (L_k + C_k) \tag{1}$$

where L_k and C_k represent the link orientation error and constraint error, respectively, at the k-th sample. Each term will be described in detail below.

The first term L_k is computed by

$$L_k = \sum_m \sum_i (\boldsymbol{p}_i^m - \hat{\boldsymbol{p}}_i^m)^T (\boldsymbol{p}_i^m - \hat{\boldsymbol{p}}_i^m) \tag{2}$$

$$\boldsymbol{p}_i^m = \boldsymbol{R}^m(\boldsymbol{q}_k, \hat{\boldsymbol{\theta}}_k + \Delta\boldsymbol{\theta})\boldsymbol{s}_i \tag{3}$$

$$\hat{\boldsymbol{p}}_i^m = \hat{\boldsymbol{R}}_k^m \boldsymbol{s}_i \tag{4}$$

where $\boldsymbol{R}^m(*,*)$ denote the forward kinematics function to compute the orientation of the m-th link with IMU from root orientation and joint angles. We evaluate the orientation error by the squared position error of the predefined three points fixed to each link with orientation measurement, and denote their relative position by $\boldsymbol{s}_i$ $(i = 1, 2, 3)$. For example, we can choose $\boldsymbol{s}_i$ as $(d, 0, 0)^T$, $(0, d, 0)^T$ and $(0, 0, d)^T$ where d is a user-defined constant. Using larger d corresponds to using larger weight for the L_k term because the position error will be larger for the same orientation error.

The second term C_k represents the kinematic constraint enforced during the data collection. If the feet are supposed to be in flat contact with a horizontal floor, for example, we can define C_k as

$$C_k = \sigma_z^2(\boldsymbol{q}_k, \hat{\boldsymbol{\theta}}_k + \Delta\boldsymbol{\theta}) \tag{5}$$

where $\sigma_z^2(*,*)$ is the function to compute the variation of the heights of the sole vertices obtained from the given root orientation and joint angles. The sole vertices are typically chosen as the four corners of each sole.

In our implementation, we apply the conjugate gradient method [4] to obtain the solution. The initial values for $\Delta\boldsymbol{\theta}$ are set to zero. The initial $\boldsymbol{q}_k$ is chosen by taking the average of root orientations computed using all IMU measurements individually.

C. Experimental Results

We used one IMU attached the pelvis link to measure its orientation ($M = 1$). Because this orientation is not affected by any joint, only the offsets of the leg joints can be identified ($N = 14$). We used four corners of the sole (eight points in total) to compute the height variance.

The sample pose data were collected at 500 Hz while hanging the robot from a gantry at several different heights and leg configuration. The robot can be moved with a reasonable force with the pump turned off. The data collection process resulted in eight motion sequences about 4 minutes long in total.

Because the pose changes slowly during the data collection, we can significantly downsample the data. It is also desirable to use as few samples as possible to minimize the computation time.

Figure 2 shows how the cost function value and computation time relate to the interval of data used for the calibration. The cost function value is averaged over frames taken at 0.2 s interval, regardless of the sample interval used for calibration. This result therefore includes cross-validations at samples not used for the optimization. As shown here, the cost function value after calibration maintains reasonably small value up to 1.8 s interval, which takes only about a minute for computation.

Figure 3 shows several examples of poses before and after the calibration. Note that the root position is fixed because we do not have that information. It can be observed that the left and right feet are flat at the same height after the calibration, while they are clearly not before. This fact is also obvious in Fig. 4, which shows the height of eight sole corners at sample poses with the average height in each sample subtracted.

III. Dynamic Calibration

A. Motivation

Accurate inertial parameters are crucial for model-based control and their identification for robot manipulators has been actively studied [5]–[7]. These methods utilize the fact that the joint torques can be represented as a linear equation

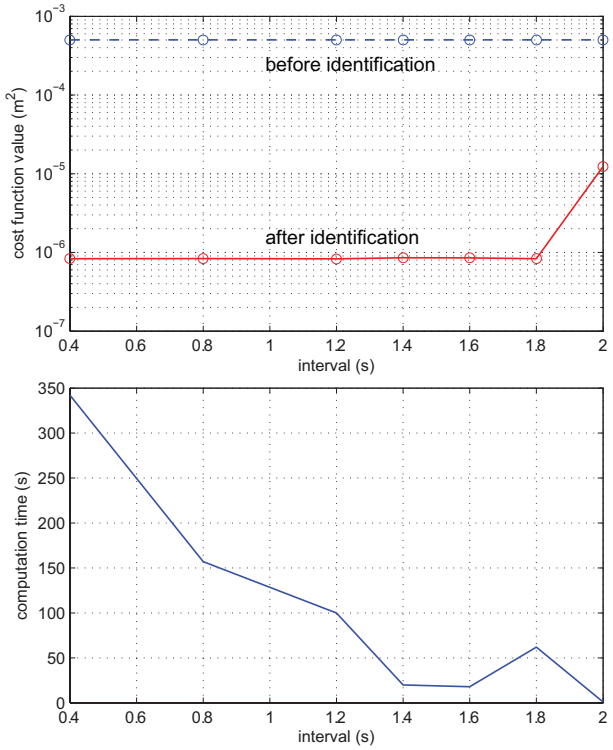

Fig. 2. Cost function value (top) and computation time (bottom) using samples at various intervals.

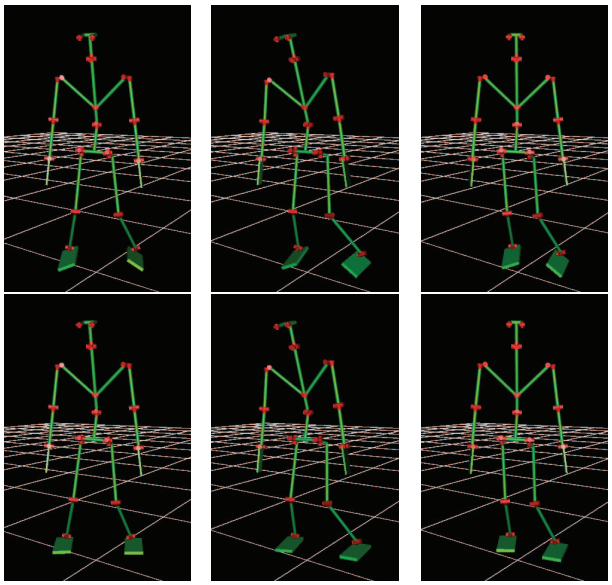

Fig. 3. Robot poses before (top row) and after (bottom row) the calibration.

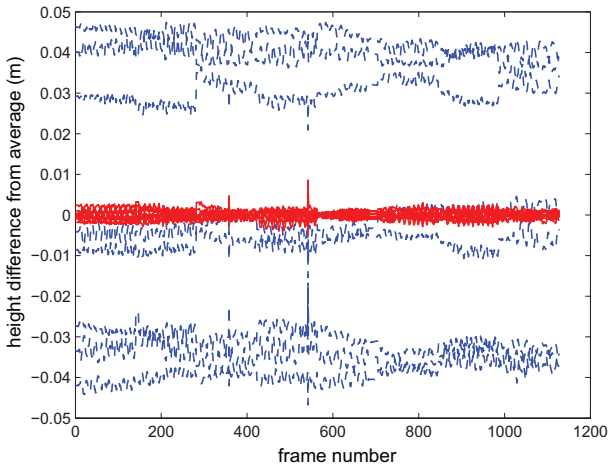

Fig. 4. Foot vertex heights from the average in each frame. Blue (dashed): before calibration, red (solid): after calibration.

of the inertial parameters. In particular, the set of parameters that affect the dynamics and therefore are identifiable with this process is called *base parameters*. Given the joint torque and motion data, the base parameters that give the minimum error can be obtained by the pseudo-inverse of the regressor.

The data should have enough excitation in order to identify all base parameters reliably and methods for optimizing the excitation have been developed [8]. With poor excitation, the regressor will be ill-conditioned and the identification may result in physically inconsistent inertial parameters such as a negative mass. Possible solutions for this issue include grouping and eliminating non-identifiable parameters using the singular values of the regressor [9] or a statistical model [10], and constraining the inertial parameters within the consistent parameter space [10], [11].

Unfortunately, applying these methods to floating-base humanoid robots is not straightforward. The main issue is how to realize accelerations large enough to perform reliable identification while maintaining the robot's balance. Ayusawa et al. [12] proved that, in theory, the set of base parameters identified by using only the floating-base dynamics is the same as that using the joint dynamics. It is therefore possible to identify the base parameters with motion and six-axis external force data without joint torque measurements [13]. Using the same technique, Venture et al. [14] developed a method for finding the combination of available data that optimizes the identification quality. Mistry et al. [15] generalized the idea of [12] to allow different combinations of joint torque and contact force measurements for identification.

Another practical issue is the complexity of the humanoid robot dynamics and additional codes required to perform the identification and to use its results for robot control and simulation. Computing the regressor usually requires reorganization of the rigid-body dynamics equations, which is difficult for complex mechanisms. The identified base parameters have to be converted to the parameters used in standard inverse dynamics formulation [16].

In this section, we first present a practical method for computing the regressor only using an inverse dynamics function that is usually available for model-based control. We then present two methods that directly computes physically consistent inertial parameters even with poor excitation: 1) consider only the major singular vectors of the regressor to obtain the least-square solution, and 2) apply a gradient-based method to solve an optimization problem with lower

bounds on the mass parameters. Experimental results suggest that the former method can yield reasonable least-square solutions if appropriate condition number threshold is used. However, the latter method works well even with the original regressor and appears to give better cross-validation results.

The second method is closely related to Ayusawa and his colleagues' work on inertial parameter identification of human body segments [11] in that they also apply gradient-based optimization. One of the differences is that they represent a link by a set of point masses placed on its surface to obtain consistent parameters. We do not use this representation because the hydraulic hoses typically run outside the link geometry in the robot's CAD model. Also it would be easier to obtain good excitation in their application because humans can do much more aggressive motions than humanoid robots.

Because we do not expect to obtain excitation good enough to identify the moments of inertia of the links, we focus on identifying the mass and local center of mass (CoM) parameters and use the CAD model to compute the link moments of inertia. Except for the poor excitation issue, however, including the moments of inertia is straightforward.

B. Problem Formulation

The dynamics of humanoid robots with a floating base is described as

$$\boldsymbol{M}(\boldsymbol{\theta},\boldsymbol{\phi})\ddot{\boldsymbol{\theta}} + \boldsymbol{c}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}},\boldsymbol{\phi}) = \boldsymbol{S}^T\boldsymbol{\tau} + \boldsymbol{J}^T(\boldsymbol{\theta})\boldsymbol{f} \tag{6}$$

where

$\boldsymbol{\theta}$	:	the generalized coordinates
$\boldsymbol{\phi}$	:	the N inertial parameters to be identified
$\boldsymbol{M}$	:	joint-space inertia matrix
$\boldsymbol{c}$	:	the centrifugal, Coriolis and gravitational forces
$\boldsymbol{\tau}$	:	active joint torques
$\boldsymbol{J}$	:	the contact Jacobian matrix
$\boldsymbol{f}$	:	contact forces

and $\boldsymbol{S}^T$ is the matrix to map joint torques to generalized forces. In particular, if the robot has M degrees of freedom (DoF) and the first six elements of the generalized coordinates correspond to the six DoF of the floating base, $\boldsymbol{S}^T$ has the form

$$\boldsymbol{S}^T = \begin{pmatrix} \boldsymbol{0}_{6\times(M-6)} \\ \boldsymbol{1}_{(M-6)\times(M-6)} \end{pmatrix} \tag{7}$$

where $\boldsymbol{0}_*$ and $\boldsymbol{1}_*$ represent the zero and identity matrix of appropriate sizes. In the rest of the paper, we represent the left-hand side of Eq.(6) by $\boldsymbol{F}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}},\ddot{\boldsymbol{\theta}},\boldsymbol{\phi})$.

Assume that we have an K-frame long sequence of experimental data comprising the joint angles $\boldsymbol{\theta}_k$, joint torques $\boldsymbol{\tau}_k$, and contact forces $\boldsymbol{f}_k$ at frame k $(k = 1, 2, \ldots, K)$. Using an inverse dynamics algorithm, we can compute $\boldsymbol{F}(*)$ from $\boldsymbol{\theta}_k$ and its velocity and acceleration at frame k:

$$\boldsymbol{F}_k(\boldsymbol{\phi}) = \boldsymbol{F}(\boldsymbol{\theta}_k,\dot{\boldsymbol{\theta}}_k,\ddot{\boldsymbol{\theta}}_k,\boldsymbol{\phi}). \tag{8}$$

It is known that $\boldsymbol{F}_k(\boldsymbol{\phi})$ becomes a linear function of $\boldsymbol{\phi}$ [5], [6]. Note that instead of the local CoM position of the i-th link $\boldsymbol{s}_i$, $\boldsymbol{\phi}$ should include the first moment of inertia $\boldsymbol{ms}_i = (ms_{ix}\ ms_{iy}\ ms_{iz})^T = m_i\boldsymbol{s}_i$ to make the equation linear, where m_i is the mass.

Let $\boldsymbol{A}_k$ denote its coefficient matrix (regressor) at frame k, i.e.,

$$\boldsymbol{F}_k(\boldsymbol{\phi}) = \boldsymbol{A}_k\boldsymbol{\phi}. \tag{9}$$

By concatenating Eq.(9) for all frames, we obtain

$$\bar{\boldsymbol{F}}(\boldsymbol{\phi}) = \bar{\boldsymbol{A}}\boldsymbol{\phi}. \tag{10}$$

On the other hand, we can compute the right-hand side of Eq.(6) from the force measurements at frame k as

$$\hat{\boldsymbol{F}}_k = \boldsymbol{S}^T\boldsymbol{\tau}_k + \boldsymbol{J}^T(\boldsymbol{\theta}_k)\boldsymbol{f}_k. \tag{11}$$

Concatenating Eq.(11) for all frames, we obtain

$$\bar{\hat{\boldsymbol{F}}} = \bar{\boldsymbol{S}}^T\bar{\boldsymbol{\tau}} + \bar{\boldsymbol{J}}^T\bar{\boldsymbol{f}}. \tag{12}$$

The goal of the parameter identification process is to compute the parameters $\boldsymbol{\phi}$ that satisfy

$$\bar{\boldsymbol{A}}\boldsymbol{\phi} = \bar{\hat{\boldsymbol{F}}}. \tag{13}$$

However, we may not be able to compute $\bar{\hat{\boldsymbol{F}}}$ due to sensor limitations. For example, joint torque measurement may be not be available as is the case with most humanoid robots driven by electric motors. Also, some contact forces may be unavailable or less accurate due to limitations in the on-board force-torque sensors.

In general, we can divide $\bar{\boldsymbol{\tau}}$ and $\bar{\boldsymbol{f}}$ into available and unavailable components and formulate the optimization problem only using the available measurements [15]. Let us divide $\bar{\boldsymbol{\tau}}$ and $\bar{\boldsymbol{f}}$ into components whose measurements are available and unavailable, $\bar{\boldsymbol{\tau}}_a$, $\bar{\boldsymbol{\tau}}_u$ and $\bar{\boldsymbol{f}}_a$, $\bar{\boldsymbol{f}}_u$ respectively. Then Eq.(12) can be rewritten as

$$\begin{aligned} \bar{\hat{\boldsymbol{F}}} &= \begin{pmatrix} \bar{\boldsymbol{S}}_a^T & \bar{\boldsymbol{J}}_a^T \end{pmatrix}\begin{pmatrix} \bar{\boldsymbol{\tau}}_a \\ \bar{\boldsymbol{f}}_a \end{pmatrix} + \begin{pmatrix} \bar{\boldsymbol{S}}_u^T & \bar{\boldsymbol{J}}_u^T \end{pmatrix}\begin{pmatrix} \bar{\boldsymbol{\tau}}_u \\ \bar{\boldsymbol{f}}_u \end{pmatrix} \\ &= \bar{\boldsymbol{H}}_a^T\begin{pmatrix} \bar{\boldsymbol{\tau}}_a \\ \bar{\boldsymbol{f}}_a \end{pmatrix} + \bar{\boldsymbol{H}}_u^T\begin{pmatrix} \bar{\boldsymbol{\tau}}_u \\ \bar{\boldsymbol{f}}_u \end{pmatrix}. \end{aligned} \tag{14}$$

Let $\bar{\boldsymbol{N}}_u$ represent the null space basis of $\bar{\boldsymbol{H}}_u^T$. By left-multiplying $\bar{\boldsymbol{N}}_u$ to the both sides of Eq.(14), we obtain

$$\bar{\boldsymbol{N}}_u\bar{\hat{\boldsymbol{F}}} = \bar{\boldsymbol{N}}_u\bar{\boldsymbol{S}}_a^T\bar{\boldsymbol{\tau}}_a + \bar{\boldsymbol{N}}_u\bar{\boldsymbol{J}}_a^T\bar{\boldsymbol{f}}_a \tag{15}$$

because $\bar{\boldsymbol{N}}_u\bar{\boldsymbol{H}}_u^T = \boldsymbol{0}$.

Multiplying both sides of Eq.(13) by $\bar{\boldsymbol{N}}_u$, we obtain

$$\bar{\boldsymbol{A}}_N\boldsymbol{\phi} = \bar{\hat{\boldsymbol{F}}}_N \tag{16}$$

where $\bar{\boldsymbol{A}}_N = \bar{\boldsymbol{N}}_u\bar{\boldsymbol{A}}$ and $\bar{\hat{\boldsymbol{F}}}_N = \bar{\boldsymbol{N}}_u\bar{\hat{\boldsymbol{F}}}$. Because of Eq.(15), Eq.(16) only includes the measurable forces.

The base parameters are generally a linear combination of a subset of the standard inertial parameters $\boldsymbol{\phi}$. Therefore, computing the regressor requires the symbolic representation of the robot dynamics and a dedicated code for evaluating the symbolic equations with given motion data. These processes can be difficult for complex mechanisms.

In this paper, we choose to use the formulation with standard inertial parameters instead of the base parameters. The drawback is that, because not all parameters are identifiable in principle, it may be difficult to obtain physically consistent parameters even with the optimal excitation. On the other hand, this approach gives us an opportunity to explore the following two advantages:

- The regressor can be computed easily using an inverse dynamics function as shown in Section III-C.
- We can directly obtain the inertial parameters.

Instead of directly solving Eq.(16), we attempt to compute a set of inertial parameters that is as close as possible to known nominal parameters $\boldsymbol{\phi}_0$ that can be obtained from, for example, the robot's CAD model. Accordingly, we divide $\boldsymbol{\phi}$ as $\boldsymbol{\phi} = \boldsymbol{\phi}_0 + \Delta\boldsymbol{\phi}$. Also, Eq.(16) does not have an exact solution in most cases due to measurement noise.

Taking the above two points into account, our problem is to find $\Delta\boldsymbol{\phi}$ that minimizes

$$Z = ||\Delta\bar{\hat{\boldsymbol{F}}}_N - \bar{\boldsymbol{A}}_N\Delta\boldsymbol{\phi}||^2 \tag{17}$$

where $\Delta\bar{\hat{\boldsymbol{F}}}_N = \bar{\hat{\boldsymbol{F}}}_N - \bar{\boldsymbol{A}}_N\boldsymbol{\phi}_0$.

C. Minimizing Eq.(17)

A straightforward method to obtain $\Delta\boldsymbol{\phi}$ is to use the pseudo-inverse of $\bar{\boldsymbol{A}}_N$:

$$\bar{\boldsymbol{A}}_N^\sharp = (\bar{\boldsymbol{A}}_N^T\bar{\boldsymbol{A}}_N)^{-1}\bar{\boldsymbol{A}}_N^T \tag{18}$$

which gives the minimum-norm $\Delta\boldsymbol{\phi}$ that minimizes Eq.(17) as $\Delta\boldsymbol{\phi} = \bar{\boldsymbol{A}}_N^\sharp\Delta\bar{\hat{\boldsymbol{F}}}_N$. However, if the condition number of $\bar{\boldsymbol{A}}_N$ is large, the resulting inertial parameters $\boldsymbol{\phi}_0 + \Delta\boldsymbol{\phi}$ may include inconsistent values.

A workaround for this problem is to omit small singular values of $\bar{\boldsymbol{A}}_N$. Let the singular-value decomposition of $\bar{\boldsymbol{A}}_N$ be

$$\bar{\boldsymbol{A}}_N = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{19}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices and $\boldsymbol{\Sigma}$ is a diagonal matrix whose diagonal elements are the singular values of $\bar{\boldsymbol{A}}_N$, σ_i $(i = 1, 2, \ldots, N,\ \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_N)$. The pseudo-inverse of $\bar{\boldsymbol{A}}_N$ can be written as

$$\bar{\boldsymbol{A}}_N^\sharp = \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T. \tag{20}$$

Instead of the regular $\boldsymbol{\Sigma}^{-1} = \mathrm{diag}\{1/\sigma_i\}$, we use $\hat{\boldsymbol{\Sigma}}^{-1}$ whose elements are given by

$$\hat{\Sigma}_{ii}^{-1} = \begin{cases} 1/\sigma_i & \text{if } \sigma_1/\sigma_i < c_{max} \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

where c_{max} is a user-defined maximum condition number. We refer to this method as the *least-square* method.

Another way to minimize Eq.(17) is to apply gradient-based numerical optimization using $\Delta\boldsymbol{\phi} = \boldsymbol{0}$ as the initial guess. The advantage of this method is that we can explicitly set lower and/or upper bounds for each parameter. In our implementation, we use the conjugate-gradient method [4] with lower bounds set to prevent negative mass parameters. We also replace $\bar{\boldsymbol{A}}_N$ by

$$\bar{\boldsymbol{A}}'_N = \boldsymbol{U}\hat{\boldsymbol{\Sigma}}\boldsymbol{V}^T \tag{22}$$

where $\hat{\boldsymbol{\Sigma}}$ is a diagonal matrix whose elements are given by

$$\hat{\Sigma}_{ii} = \begin{cases} \sigma_i & \text{if } \sigma_1/\sigma_i < c_{max} \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

This method will be referred to as the *gradient-based* method.

D. Computing the Regressor

Computing the regressor $\boldsymbol{A}_k$ in our formulation is trivial because we use the standard inertial parameters. This subsection describes a method for computing the i-th column of $\boldsymbol{A}_k$, $\boldsymbol{a}_{ki}$. This method is partly inspired by the unit vector method for computing joint-space inertia matrix for robot manipulators [17].

First we compute the generalized forces $\boldsymbol{F}_k^0$ with the nominal inertial parameters using the inverse dynamics function:

$$\boldsymbol{F}_k^0 = \boldsymbol{F}(\boldsymbol{\theta}_k, \dot{\boldsymbol{\theta}}_k, \ddot{\boldsymbol{\theta}}_k, \boldsymbol{\phi}_0). \tag{24}$$

We then make a unit change to the i-th element of $\boldsymbol{\phi}_0$ to obtain $\boldsymbol{\phi}'_i$, and compute the new generalized forces corresponding to the new inertial parameters by $\boldsymbol{F}_k^i = \boldsymbol{F}(\boldsymbol{\theta}_k, \dot{\boldsymbol{\theta}}_k, \ddot{\boldsymbol{\theta}}_k, \boldsymbol{\phi}'_i)$. Finally, $\boldsymbol{a}_{ki}$ can be computed by

$$\boldsymbol{a}_{ki} = \boldsymbol{F}_k^i - \boldsymbol{F}_k^0. \tag{25}$$

We now describe how to modify the mass and local CoM parameters according to a unit change to the i-th element of $\boldsymbol{\phi}$. If the i-th element of $\boldsymbol{\phi}$ corresponds to the mass of link j, we modify the inertial parameters of link j as

$$m_j \leftarrow m_j + 1 \tag{26}$$

$$\boldsymbol{s}_j \leftarrow \frac{m_j}{m_j + 1}\boldsymbol{s}_j \tag{27}$$

where m_j has been increased by a unit mass but $\boldsymbol{ms}_j$ remains the same. If the i-th element of $\boldsymbol{\phi}$ corresponds to ms_{j*} $(* = x, y, z)$, we modify the CoM as

$$s_{j*} \leftarrow s_{j*} + \frac{1}{m_j} \tag{28}$$

so that ms_{j*} increases by 1.

E. Experimental Results

We collected three sequences of data using the controller described in [18], where the robot is controlled to track a human motion capture sequence while maintaining balance. We use one sequence for identification and the other two for cross validation, each of which is about 30 s long. During the motion, no contact state change occurs and both feet keep flat contact with the ground. The motion contains significant amount of upper body movement, while the lower body is relatively stationary but does have small amount of motion mostly due to the errors in the dynamics model used for the experiment. The arm joints are controlled by a proportional-derivative position controller. A model-based controller for maintaining balance and tracking a reference motion is used to compute the joint torque commands of other joints.

We measure the joint angles, joint torques, and ground contact forces during the experiments. The root orientation

is measured by an IMU attached to the pelvis link. The root position is estimated as described in [18] using the reference motion and leg joint angles. Although each joint has a joint torque sensor, we cannot obtain joint torque data for position-controlled joints due to the specification of the control software. Also, we are aware that the horizontal force components of the six-axis force-torque sensors in the ankles are not well calibrated.

The original data have been recorded at 2 ms interval, and downsampled to 100 ms interval for the identification. To compute joint velocities and accelerations, the joint angle measurements are interpolated by piece-wise third-order polynomials whose coefficients are determined considering 125 data points before and after each frame. This interpolation therefore also serves as a low-pass filter without delay.

We compare the following parameter sets for identification:

1) Individual links (L): mass and local CoM parameters of individual links are identified.
2) Individual links with symmetry constraint (LS): mass and local CoM parameters of the links on the left side of the robot are identified, and the links on the right side are assumed to have the same parameters with mirroring.

The second and third rows of Table I summarize the basic properties of each parameter set with the data we used. As expected, the regressors in both parameter sets have large condition numbers.

The rest of Table I shows the cost function values Eq.(17) using the two methods. We tested with three values of c_{max}. Setting $c_{max} = 1 \times 10^6$ essentially uses all singular values because it is larger the condition numbers of both regressors. The cost function values using the original inertial parameters (obtained from CAD data) are in the range of 2.11×10^7 to 2.29×10^7 for the three motions.

Because we do not know the real inertial parameters, the only failure we can detect is negative mass parameters, which is indicated by "n/a" in Table I. Resulting in negative mass parameters is not surprising because individual mass parameters are not identifiable even theoretically. However, the two methods are able to compute positive mass parameters except when the condition number threshold is too large in the least square method.

The following observations can be made from these results:

- Small singular values must be omitted to obtain reasonable results using pseudo-inverse, while the gradient-based method gives consistent results regardless of c_{max}.
- Enforcing symmetry generally gives better cross-validation results with slightly worse direct-validation results.
- In most of the cases where both pseudo-inverse and gradient-based methods give reasonable results, the gradient-based method gives significantly better cross-validation results, although the direct-validation results are similar.

Figure 5 shows representative direct- and cross-validation results for the $c_{max} = 1 \times 10^6$ case obtained by the gradient-based method without symmetry constraint. While the identified parameters show better match with the measurements in general, they have larger error than the nominal parameters in the cross-validation of the root front-aft force. This mismatch may be due to the hydraulic hose bundle that applies unknown external force to the rear side of the root. The hose forces may be different in the two sets of data because the cross-validation data sequence was obtained a few months before the identification data.

IV. Discussion

In this paper, we have presented methods and experiments regarding the identification of kinematic and dynamic parameters force-controlled biped humanoid robots.

The first method allows us to estimate joint angle sensor offsets only using the sensors that are usually available on most humanoid robots. The key idea is to utilize a constraint enforced by the environment instead of the global position and orientation measurements commonly used for this purpose. In our experiment, we manually moved the robot by hand while keeping both feet in flat contact with the floor, and demonstrated that the constraint is enough to estimate the joint angle sensor offsets. A limitation of this method is that it cannot identify offsets of joints that are not related to the constraint or the IMUs.

In the second part of the paper, we presented two methods for inertial parameter identification of humanoid robots, with focus on easy implementation and robustness against poor excitation data. The former feature is realized by identifying the standard inertial parameters instead of the base parameters, while the latter is handled by either omitting small singular values of the regressor in the least-square method, or by solving an optimization problem using a gradient-based method. We demonstrated by experiments that the least-square approach can yield consistent inertial parameters if the condition number threshold is appropriate. We also showed that the gradient-based method gives reasonable results even with ill-conditioned regressors, and that the cross-validation results are better than the least-square method.

Acknowledgement

The author would like to thank Prof. Gentiane Venture for her comments on the inertial parameter identification section of the manuscript.

References

[1] X. Zhong and J. Lewis, "A new method for autonomous robot calibration," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1995, pp. 1790–1795.
[2] H. Zhuang, S. Motaghedi, and Z. Roth, "Robot calibration with planar constraints," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1999, pp. 805–810.
[3] M. Ikits and J. Hollerbach, "Kinematic calibration using a plane constraint," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1997, pp. 3191–3196.
[4] M. Renouf and P. Alart, "Conjugate gradient type algorithms for fricional multi-contact problems: Applications to granular materials," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, pp. 2019–2041, 2004.

TABLE I

COST FUNCTION VALUES FOR OPTIMIZATION AND CROSS-VALIDATION.

parameter set		L			LS		
number of parameters		92			56		
condition number of regressor		1.38×10^5			3.81×10^4		
maximum condition number		1×10^6	1×10^3	1×10^2	1×10^6	1×10^3	1×10^2
least square	direct validation	n/a	n/a	1.56×10^5	n/a	n/a	1.97×10^5
	cross validation (1)	n/a	n/a	7.13×10^5	n/a	n/a	5.77×10^5
	cross validation (2)	n/a	n/a	7.93×10^5	n/a	n/a	9.47×10^5
gradient	direct validation	1.52×10^5	1.52×10^5	1.52×10^5	1.99×10^5	1.99×10^5	1.99×10^5
	cross validation (1)	5.28×10^5	5.28×10^5	5.29×10^5	4.51×10^5	4.51×10^5	4.50×10^5
	cross validation (2)	7.29×10^5	7.29×10^5	7.30×10^5	5.81×10^5	5.81×10^5	5.84×10^5

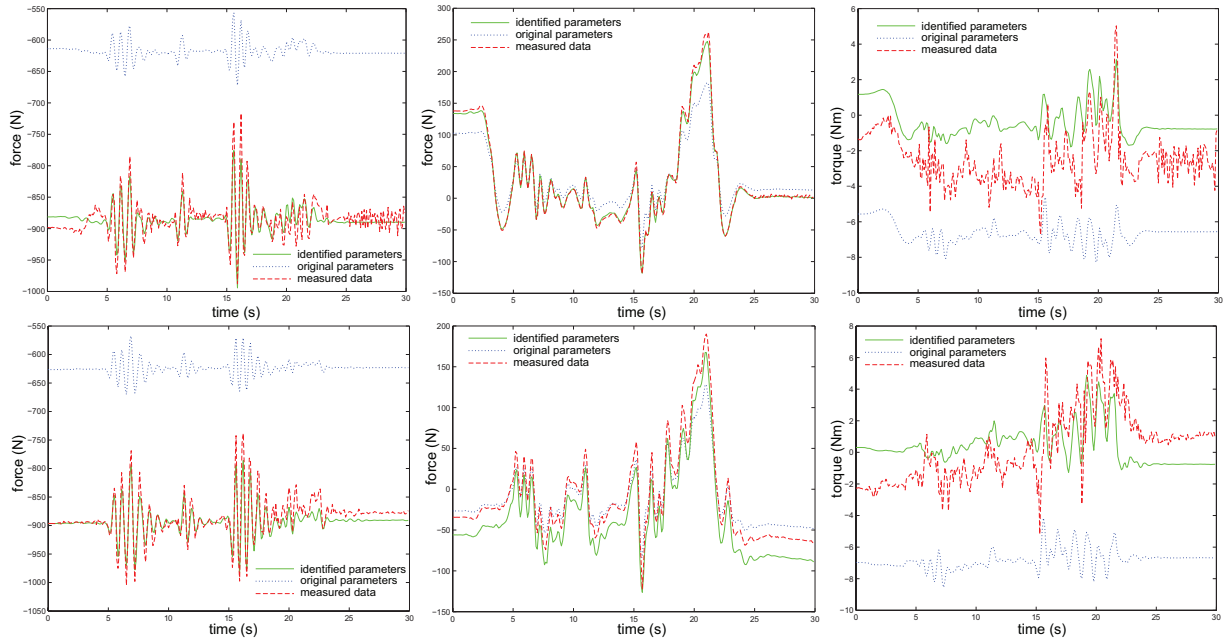

Fig. 5. Direct- and cross-validation results for $c_{max} = 1 \times 10^6$, gradient-based method, without symmetry constraint. Top row: self validation, bottom row: cross validation (1). From left to right: root vertical force (downward is positive), root front-aft force (backward is positive), and left knee joint torque.

[5] H. Mayeda, K. Osuka, and A. Kanagawa, "A new identification method for serial manipulator arms," in *Proceedings of the IFAC 9th World Congress*, vol. 2, 1984, pp. 74–79.

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

[7] W. Khalil and E. Dombre, *Modeling, identification and control of robots*. London, U.K.: Hermès Penton, 2002.

[8] M. Gautier and W. Khalil, "Exciting trajectories for the identification of base inertial parameters of robots," *International Journal of Robotics Research*, vol. 4, no. 11, pp. 363–375, 1992.

[9] M. Gautier, "Numerical calculation of the base inertial parameters of robots," in *IEEE International Conference on Robotics and Automation*, 1990, pp. 1020–1025.

[10] J. Ting, M. Mistry, J. Peters, S. Schaal, and J. Nakanishi, "A Bayesian approach to nonlinear parameter identification for rigid body dynamics," in *Robotcs: Science and Systems*, 2006.

[11] K. Ayusawa, G. Venture, and Y. Nakamura, "Real-time implementation of physically consistent identification of human body segments," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011, pp. 6282–6287.

[12] ——, "Symbolic proof of inertia-parameter identifiability of legged mechanism using base-link dynamics," in *Proceedings of the IFAC International Conference on System Identification*, 2009, pp. 693–698.

[13] ——, "Identification of humanoid robots dynamics using floating-base motion dynamics," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 2854–2859.

[14] G. Venture, K. Ayusawa, and Y. Nakamura, "A numerical method for choosing motions with optimal excitation properties for identification of biped dynamics—an application to human," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 1226–1231.

[15] M. Mistry, S. Schaal, and K. Yamane, "Inertial parameter estimation of floating base humanoid systems using partial force sensing," in *IEEE-RAS International Conference on Humanoid Robots*, 2009, pp. 492–497.

[16] G. Venture, K. Ayusawa, and Y. Nakamura, "Real-time identification and visualization of human segment parameters," in *Proceedings of the IEEE International Conference on Engineering in Medicine and Biology*, 2009, pp. 3983–3986.

[17] M. Walker and D. Orin, "Efficient Dynamic Computer Simulation of Robot Manipulators," *ASME Journal on Dynamic Systems, Measurement and Control*, vol. 104, pp. 205–211, 1982.

[18] K. Yamane, S. Anderson, and J. Hodgins, "Controlling humanoid robots with human motion data: Experimental validation," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, Nashville, TN, December 2010, pp. 504–510.