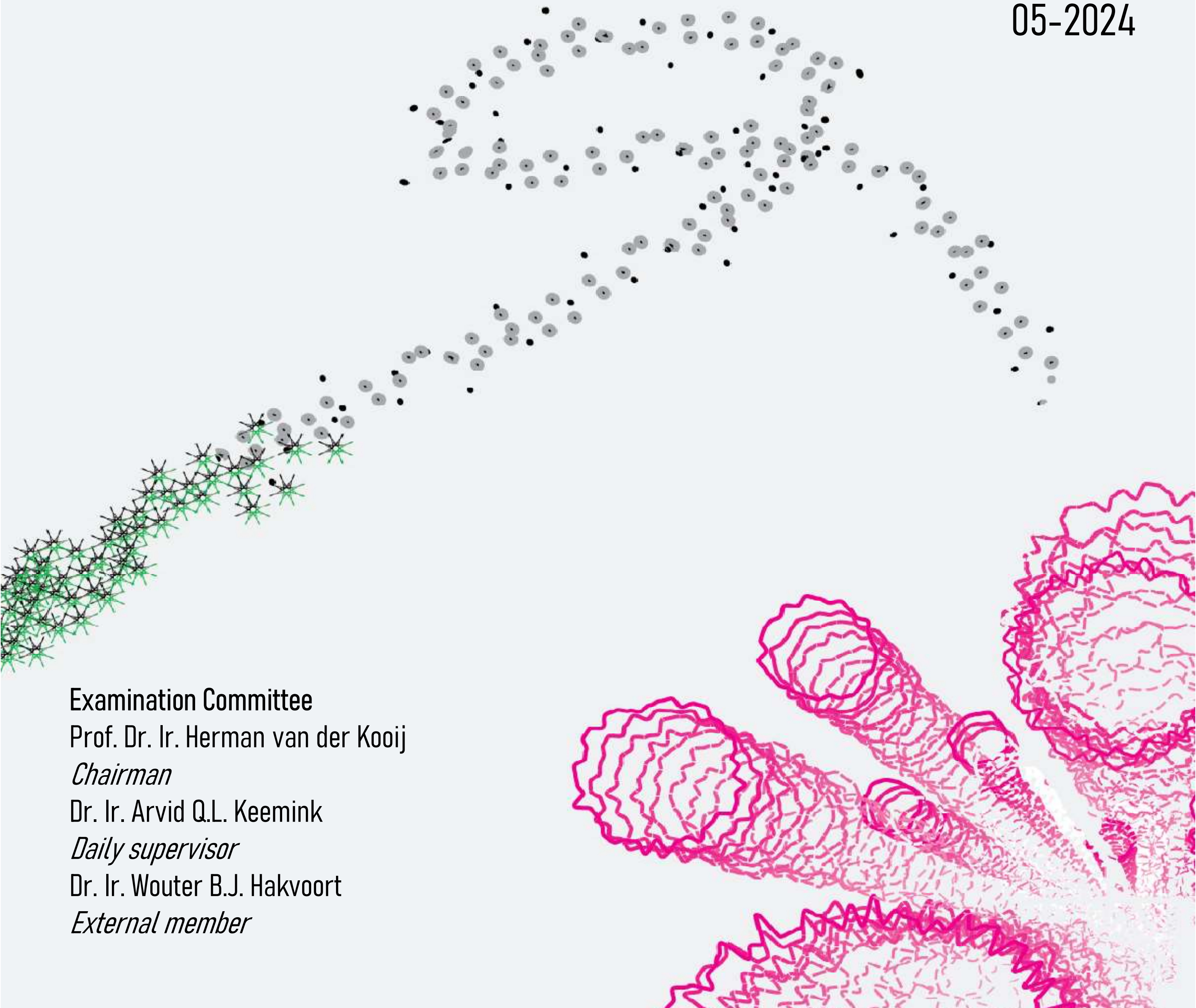
UNIVERSITY OF TWENTE.

| Faculty of Engineering Technology | Department of Mechanical Engineering |

INERTIAL PARAMETER IDENTIFICATION OF BIPEDAL SYSTEMS: A SIMULATION-BASED COMPARISON

Nils van der Gaag
Master's Thesis
05-2024



Examination Committee
Prof. Dr. Ir. Herman van der Kooij
Chairman
Dr. Ir. Arvid Q.L. Keemink
Daily supervisor
Dr. Ir. Wouter B.J. Hakvoort
External member

Preface

Dear Reader,

This document presents my master's thesis 'Inertial parameter identification of bipedal systems: a simulation-based comparison'. It was written to complete my master's mechanical engineering at the University of Twente. After a challenging but educational journey of about a year and two months, I can finally say it is complete. I hope this thesis contributes to breaking barriers for paraplegic people. I am optimistic about the research that is done at the university to take exoskeletons to the next level, and hopefully we will soon notice the contributions in our daily lives.

I would like to express gratitude to my daily supervisor, Arvid Keemink, who pushed me to my limits but also helped me for hours when I couldn't figure it out myself. I want to thank the people from the scrum project as well, since they brought structure and clear goals to my day-to-day work.
I think it is safe to say that I wouldn't be finished writing this thesis if it wasn't for my family and girlfriend, supporting my every move and spending hours listening to my technical stories.

Have fun reading this thesis.

Nils van der Gaag
Tilburg, May 2024

Abstract

This thesis presents the implementation of inertial parameter identification using least-squares estimation, comparing four methods to solve the resulting equations. In the first chapters the equation of motion is written linearly in the inertial parameters, which leads to a least squares problem that cannot be solved directly.

In the included paper, the least squares problem is scaled using measurement uncertainty in the joints and a prior estimation, and solved using a singular value decomposition. The effects of scaling are examined by comparing the results of the unscaled, and multiple scaled problems. Using simulation data of the Symbitron lower-limb exoskeleton, it is shown that the row- and column-scaled version performs best, despite being sensitive to a prior estimation.

Contents

Chapter 1

Introduction

1.1 Research goal

The primary goal of this research is to develop an effective method for estimating the inertial parameters of bipedal systems, focusing particularly on the Symbitron lower-limb exoskeleton (LLE), which is shown in Fig. 1.1. These devices can enhance the quality of life for people with paraplegia by enhancing their mobility and autonomy [2, 3], as well as mitigating health issues associated with prolonged immobility such as muscle deterioration and cardiovascular complications [2, 4, 5]. Despite these benefits, the widespread adoption of LLEs is hindered by their limitations in robust balance control, often requiring users to rely on additional supports like crutches.

Current research efforts predominantly aim to refine gait replication through pre-programmed joint trajectories, possibly adjusting to unexpected disturbances [6–8]. However, these strategies do not fully replicate the stable locomotion observable in humans. Therefore, advancements in robust balance control are essential.

The University of Twente is currently performing research in this field, using the Symbitron exoskeleton to explore advanced control strategies. One promising strategy is controlling the Center of Mass (CoM) and Whole Body Angular Momentum (WBAM), called centroidal dynamics control. This strategy effectively reduces the complex, high-dimensional problem of individually controlling each joint into a more manageable, low-dimensional task focused on broader system dynamics [1].

This thesis specifically targets the development of a method for the identification of inertial parameters crucial for model-based control strategies like centroidal dynamics control. These parameters are fundamental for accurately predicting the necessary actuation forces to achieve desired movements. Given that manual estimation methods are labor-intensive and computer-aided design (CAD) models offer limited accuracy, this research aims to use statistical learning algorithms on measurement data from integrated or external sensors to improve the repeatability and precision of parameter estimation.

The objectives of this thesis are outlined as follows:

- Provide a proof of concept of parameter identification for the Symbitron LLE, that uses its existing sensors with the potential addition of two force-measuring platforms
- Outline the implementation of the method
- Evaluate the effectiveness of the estimated parameters within a centroidal dynamics control framework

This research does not extend to estimating human inertial parameters, nor does it delve deeply into the theories underpinning centroidal dynamics control.

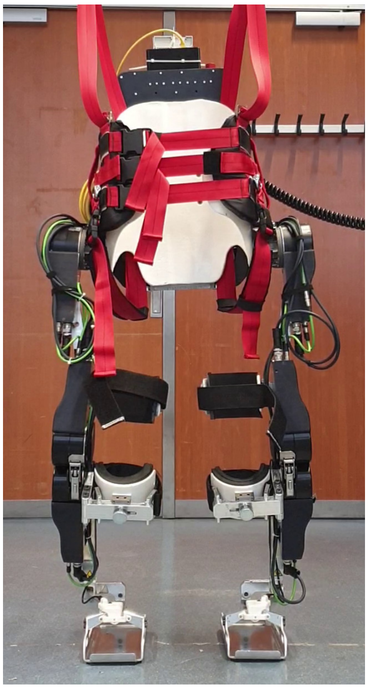


Figure 1.1: The symbitron exoskeleton [1].

1.2 Thesis structure

The main part of this thesis is presented in the paper in chapter 3. chapter 2 is intended to introduce concepts that are used, and expand on topics that are only briefly addressed in the paper. chapter 4 contains additional results and analysis, and reasons on why some choices are made in the analysis of the paper. Finally chapter 5 concludes on the combination of the paper and additional results.

Chapter 2

Background knowledge

This chapter will provide a brief overview of the concepts that are needed to understand the theory in the paper in chapter 3. For full derivations or definitions, the reader is referred to the sources.

2.1 Floating base rigid body dynamics

The LLE is able to move with respect to the inertial frame. In the context of robot dynamics, such system is said to have a floating base. The additional degrees of freedom (DOFs) introduced by a floating base add some degree of complexity to the dynamics of the system. Furthermore, the links can be assumed to be rigid, because of their thickness and the relatively stiff materials that are used. This assumption is always inaccurate to some extend, so if it appears that the used method produces flawed results, the validity of this assumption should be tested.

This section is by no means intended to be a complete overview of the dynamics, but it will introduce some concepts associated with modelling rigid floating base systems, that are needed to understand the following chapters.

2.1.1 Homogeneous transform and Spatial vectors

The position of the origin of a coordinate frame Ψ_l with respect to another Ψ_i can be represented as

$$ {}^{i}\mathbf{p}_l = \begin{bmatrix} {}^{i}\mathbf{p}_{l,x} \\ {}^{i}\mathbf{p}_{l,y} \\ {}^{i}\mathbf{p}_{l,z} \end{bmatrix}, \tag{2.1} $$

where subscript $_x$, $_y$ and $_z$ denote the x, y and z components, respectively. The orientation of Ψ_l w.r.t. Ψ_i is represented by the 3×3 rotation matrix

$$ {}^{i}\mathbf{R}_l = \begin{pmatrix} \hat{\mathbf{x}}_i \cdot \hat{\mathbf{x}}_l & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{x}}_l & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{x}}_l \\ \hat{\mathbf{x}}_i \cdot \hat{\mathbf{y}}_l & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{y}}_l & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{y}}_l \\ \hat{\mathbf{x}}_i \cdot \hat{\mathbf{z}}_l & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{z}}_l & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{z}}_l \end{pmatrix} \tag{2.2} $$

where $\hat{\mathbf{x}}_i$, $\hat{\mathbf{y}}_i$ and $\hat{\mathbf{z}}_i$ denote the x-, y- and z-coordinate unit vector of Ψ_i, respectively, and $(\cdot)$ the dot product between two vectors.

The homogeneous transformation matrix ${}^{i}\mathbf{H}_l$ transforms the position representation of a point expressed in Ψ_l to a representation in Ψ_i:

$$ \begin{bmatrix} {}^{i}\mathbf{r} \\ 1 \end{bmatrix} = {}^{i}\mathbf{H}_l \begin{bmatrix} {}^{l}\mathbf{r} \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{i}\mathbf{R}_l & {}^{i}\mathbf{p}_l \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} {}^{l}\mathbf{r} \\ 1 \end{bmatrix} \tag{2.3} $$

where ${}^{i}\mathbf{r}$ is a position represented in Ψ_i coordinates. The position of a reference frame origin and a point are assigned different symbols for clarity.

An important property of $\mathbf{H}$ is that it can be chained, i.e.

$$ {}^{i}\mathbf{H}_l = {}^{i}\mathbf{H}_k\, {}^{k}\mathbf{H}_l. \tag{2.4} $$

The spatial motion vector is a generalization of linear and angular velocity which is useful to describe the motion of a rigid body. A motion vector can be a velocity as well as an acceleration. The spatial velocity and acceleration vectors ${}^{i}\mathbf{v}_l^k$ and ${}^{i}\mathbf{a}_l^k$ describe the motion of rigid body l with respect to body k, and are expressed in Ψ_i coordinates. If all quantities are expressed in the same frame and describe the motion w.r.t. the same frame, the upper-left and upper-right index can be omitted. The motion vectors are defined as

$$ \mathbf{v}_l = \begin{pmatrix} \boldsymbol{\omega}_l \\ \dot{\mathbf{r}}_{l,o} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\omega}_l \\ \dot{\mathbf{r}}_{l,p} - \boldsymbol{\omega}_l \times \mathbf{r}_{l,p} \end{pmatrix}, \qquad \mathbf{a}_l = \begin{pmatrix} \dot{\boldsymbol{\omega}}_l \\ \ddot{\mathbf{r}}_{l,o} \end{pmatrix} = \begin{pmatrix} \dot{\boldsymbol{\omega}}_l \\ \ddot{\mathbf{r}}_{l,p} - \boldsymbol{\omega}_l \times \dot{\mathbf{r}}_{l,p} \end{pmatrix} \tag{2.5} $$

where $\mathbf{r}_{l,p}$, $\dot{\mathbf{r}}_{l,p}$ and $\ddot{\mathbf{r}}_{l,p}$ are the position, linear velocity and linear acceleration of a point p attached to body l with respect to the frame in which the motion is expressed, respectively. The quantities $\dot{\mathbf{r}}_{l,o}$ and $\ddot{\mathbf{r}}_{l,o}$ can be seen as the linear velocity and acceleration of a point attached to body l, coinciding with the origin of the frame in which $\mathbf{v}_l$ is expressed. Symbols $\boldsymbol{\omega}_l$ and $\dot{\boldsymbol{\omega}}_l$ represent the rotational velocity and acceleration of body l, respectively. In some literature $\mathbf{v}$ is called a 'twist'.

A similar vector can be defined for forces, called a spatial force vector. The spatial force acting on body l, expressed in Ψ_i coordinates is expressed as

$$ {}^{i}\mathbf{f}_l = \begin{bmatrix} {}^{i}\mathbf{w}_{l,o} \\ {}^{i}\mathbf{n}_l \end{bmatrix}, \tag{2.6} $$

where ${}^{i}\mathbf{w}_{l,o}$ is the total moment acting on body l around the origin of Ψ_i, and ${}^{i}\mathbf{n}_l$ the force acting on body l, both expressed in Ψ_i coordinates.

While $\mathbf{v}_l$ and $\mathbf{a}_l$ describe different motion quantities, they are both motion vectors and therefore abide by the same mathematical laws. The same can be said for any force vector.

Changing coordinates is done using the spatial transform and the spatial force transform, $\mathbf{X}$ and $\mathbf{X}^F$, respectively, such that

$$ {}^{i}\mathbf{m} = {}^{i}\mathbf{X}_l\, {}^{l}\mathbf{m}, \qquad {}^{i}\mathbf{f} = {}^{i}\mathbf{X}_l^F\, {}^{l}\mathbf{f}. \tag{2.7} $$

where $\mathbf{m}$ denotes any motion vector.

The transformations are defined as

$$ {}^{i}\mathbf{X}_l = \begin{bmatrix} {}^{i}\mathbf{R}_l & \mathbf{0} \\ \mathbf{S}({}^{i}\mathbf{p}_l)\, {}^{i}\mathbf{R}_l & {}^{i}\mathbf{R}_l \end{bmatrix}, \qquad {}^{i}\mathbf{X}_l^F = {}^{i}\mathbf{X}_l^{-T} = \begin{bmatrix} {}^{i}\mathbf{R}_l & \mathbf{S}({}^{i}\mathbf{p}_l)\, {}^{i}\mathbf{R}_l \\ \mathbf{0} & {}^{i}\mathbf{R}_l \end{bmatrix}, \tag{2.8} $$

where $\mathbf{S}(\mathbf{x})$ is the skew operator, defined as

$$ \mathbf{S}(\mathbf{x}) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \tag{2.9} $$

which can be seen as the matrix-vector form of a cross-product, in the sense that $\mathbf{x} \times \mathbf{y} = \mathbf{S}(\mathbf{x})\mathbf{y}$. Both $\mathbf{X}$ and $\mathbf{X}^F$ can be chained, similar to the homogeneous transformation in Eq. (2.4).

The following cross-products are defined for spatial vectors:

$$ \mathbf{m}_i \times \mathbf{m}_k = \begin{bmatrix} \bar{\mathbf{m}}_i \\ \bar{\mathbf{m}}_{i,o} \end{bmatrix} \times \begin{bmatrix} \bar{\mathbf{m}}_k \\ \bar{\mathbf{m}}_{k,o} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{m}}_i \times \bar{\mathbf{m}}_k \\ \bar{\mathbf{m}}_i \times \bar{\mathbf{m}}_{k,o} + \bar{\mathbf{m}}_{i,o} \times \bar{\mathbf{m}}_k \end{bmatrix}, \tag{2.10} $$

and

$$\mathbf{m} \times \mathbf{f} = \begin{bmatrix} \bar{\mathbf{m}} \\ \bar{\mathbf{m}}_o \end{bmatrix} \times \begin{bmatrix} \bar{\mathbf{f}}_o \\ \bar{\mathbf{f}} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{m}} \times \bar{\mathbf{f}} + \bar{\mathbf{m}}_o \times \boldsymbol{f} \\ \bar{\mathbf{m}} \times \bar{\mathbf{f}} \end{bmatrix}. \tag{2.11}$$

where the bar indicates that it is a 3 × 1 part of the spatial vector.

Similar to Eq. (2.9), a matrix form can be defined for these products as well, which is defined as

$$\mathbf{S}(\mathbf{m}) = \begin{bmatrix} \mathbf{S}(\bar{\mathbf{m}}) & \mathbf{0} \\ \mathbf{S}(\bar{\mathbf{m}}_o) & \mathbf{S}(\bar{\mathbf{m}}) \end{bmatrix}, \tag{2.12}$$

so that

$$\mathbf{m}_i \times \mathbf{m}_k = \mathbf{S}(\mathbf{m}_i)\mathbf{m}_k, \qquad \mathbf{m} \times \mathbf{f} = -\mathbf{S}^T(\mathbf{m})\mathbf{f}. \tag{2.13}$$

The linear and angular momentum of a rigid body expressed in its center of mass $\mathbf{c}$ are $\bar{\mathbf{h}} = m\dot{\mathbf{r}}_c$ and $\bar{\mathbf{h}}_c = \bar{\mathbf{I}}_c\boldsymbol{\omega}$, respectively, where $\bar{\mathbf{I}}_c$ is the 3 × 3 inertia matrix and m is the mass. Expressing the latter about another point p yields $\bar{\mathbf{h}}_p = \bar{\mathbf{h}}_c + \mathbf{c}^p \times \bar{\mathbf{h}}$, where $\mathbf{c}^p$ is the center of mass position relative to point p. In spatial vector form this yields

$$\mathbf{h}_c = \begin{bmatrix} \bar{\mathbf{h}}_c \\ \bar{\mathbf{h}} \end{bmatrix}, \qquad \mathbf{h}_p = \begin{bmatrix} \boldsymbol{I} & \mathbf{S}(\mathbf{c}^p) \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} \mathbf{h}_c \tag{2.14}$$

where $\boldsymbol{I}$ is the identity matrix.

The spatial inertia expressed in the center of mass $\mathbf{I}_c$ can be obtained from its spatial momentum $\mathbf{h}_c = \mathbf{I}_c\mathbf{v}_c$, which implies

$$\mathbf{I}_c = \begin{bmatrix} \bar{\mathbf{I}}_c & \mathbf{0} \\ \mathbf{0} & m\boldsymbol{I} \end{bmatrix}. \tag{2.15}$$

If this is the spatial inertia of link i can be expressed in the link i origin using Eq. (2.14), i.e.

$$\begin{aligned} \mathbf{h}_i &= \begin{bmatrix} \boldsymbol{I} & \mathbf{S}(\mathbf{c}^i) \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} \mathbf{I}_c\mathbf{v}_c = \begin{bmatrix} \boldsymbol{I} & \mathbf{S}(\mathbf{c}^i) \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{I}}_c & \mathbf{0} \\ \mathbf{0} & m\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ \mathbf{S}(\mathbf{c}^i)^T & \boldsymbol{I} \end{bmatrix} \mathbf{v}_i \\ &= \begin{bmatrix} \bar{\mathbf{I}}_c + m\mathbf{S}(\mathbf{c}^i)\mathbf{S}(\mathbf{c}^i)^T & m\mathbf{S}(\mathbf{c}^i) \\ m\mathbf{S}(\mathbf{c}^i)^T & m\boldsymbol{I} \end{bmatrix} \mathbf{v}_i, \end{aligned} \tag{2.16}$$

in which $\mathbf{I}_i$ can be identified as

$$\mathbf{I}_i = \begin{bmatrix} \bar{\mathbf{I}}_c + m\mathbf{S}(\mathbf{c}^i)\mathbf{S}(\mathbf{c}^i)^T & m\mathbf{S}(\mathbf{c}^i) \\ m\mathbf{S}(\mathbf{c}^i)^T & m\boldsymbol{I} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{I}}_i & m\mathbf{S}(\mathbf{c}^i) \\ m\mathbf{S}(\mathbf{c}^i)^T & m\boldsymbol{I} \end{bmatrix} \tag{2.17}$$

The equation of motion of rigid link i is given by differentiating its spatial momentum in time, i.e.

$$\mathbf{f}_i = \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{I}_i\mathbf{v}_i) = \mathbf{I}_i\mathbf{a}_i + \dot{\mathbf{I}}_i\mathbf{v}_i = \mathbf{I}_i\mathbf{a}_i + \mathbf{v}_i \times \mathbf{I}_i\mathbf{v}_i \tag{2.18}$$

as shown by [9].

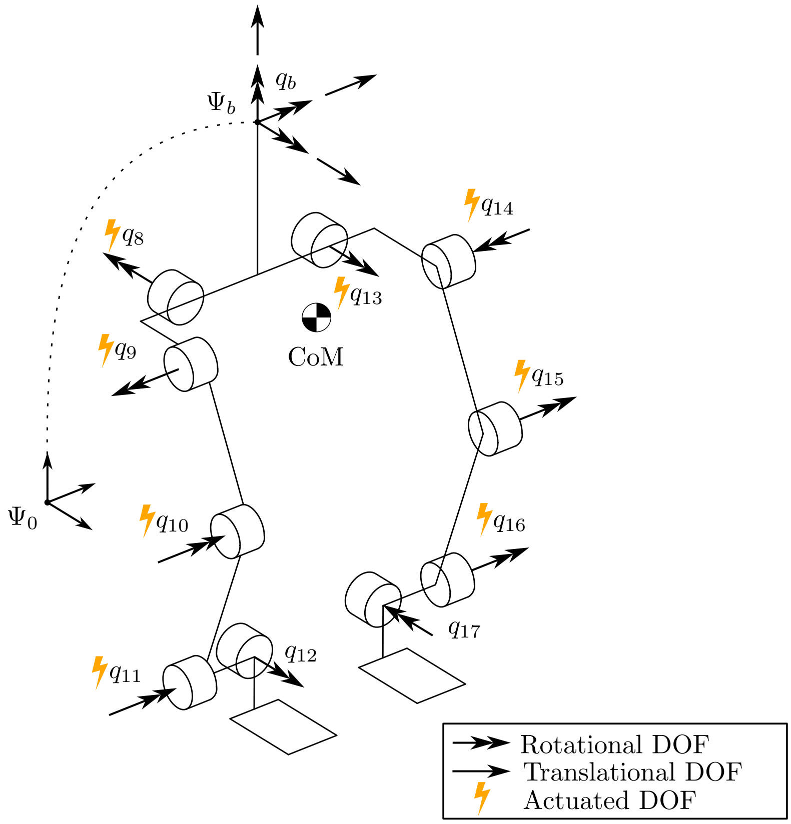


Figure 2.1: A schematic of the LLE.

2.1.2 Joint space representation

In the context of rigid body modelling, the LLE can be represented as a kinematic tree with each leg being a branch and the body link connected to the world by a ‘free’ or ‘6 degree-of-freedom’ (6-DOF) joint. With the $j = 10$ rotational joints, this makes a total of $n = 16$ degrees of freedom.
The configuration of a rotational joint i can be represented by its joint angle θ with respect to a reference $q_i = \theta_i$. The configuration of the 6-DOF joint $\mathbf{q}_b$ can be represented by the position and orientation of the floating base origin with respect to an inertial frame Ψ_0. There are multiple methods for this, but a straightforward way to describe the position is the position expressed in the inertial frame. The orientation can conveniently be described by a unit quaternion, since this is a concise and unique representation of an orientation. Therefore, the joint configuration has $n + 1 = 17$ elements: four for the base orientation, three for the base position, and $j = 10$ for the rotational joint configurations:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_b^T & \mathbf{q}_r^T \end{bmatrix}^T \in \mathbb{R}^{n+1}, \tag{2.19}$$

where $\mathbf{q}_b^T = \begin{bmatrix} \boldsymbol{\eta}^T & {}^0\mathbf{p}_b^T \end{bmatrix}^T \in \mathbb{R}^7$ the concatenated unit quaternion and position of the floating base origin, and $\mathbf{q}_r = \begin{bmatrix} \theta_1 & \theta_2 & \dots & \theta_j \end{bmatrix}^T \in \mathbb{R}^j$ the concatenated joint angles. A schematic of the LLE is shown in Fig. 2.1

The first and second time derivatives of the joint configuration $\dot{\mathbf{q}}_r$ and $\ddot{\mathbf{q}}_r$, can be described by the joint velocities and accelerations, respectively:

$$\dot{\mathbf{q}}_r = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dots & \dot{\theta}_j \end{bmatrix}^T, \qquad \ddot{\mathbf{q}}_r = \begin{bmatrix} \ddot{\theta}_1 & \ddot{\theta}_2 & \dots & \ddot{\theta}_j \end{bmatrix}^T \tag{2.20}$$

The first and second time derivatives of the 6-DOF joint configuration can be described by the angular and linear velocity and acceleration in body frame, respectively. I.e.

$$\dot{\mathbf{q}}_b = \begin{bmatrix} {}^b\boldsymbol{\omega}_b^0 \\ {}^b\dot{\mathbf{p}}_b^0 \end{bmatrix} \in \mathbb{R}^6, \qquad \ddot{\mathbf{q}}_b = \begin{bmatrix} {}^b\dot{\boldsymbol{\omega}}_b^0 \\ {}^b\ddot{\mathbf{p}}_b^0 \end{bmatrix} \in \mathbb{R}^6, \tag{2.21}$$

where all quantities are w.r.t. the inertial frame. Therefore, $\dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$.

2.1.3 Dynamic model of the LLE

A dynamic model relates the motion of a system to the forces acting on the system. One way to describe this relation is using the joint-space dynamics of the system, which can be expressed in joint-space using the Euler-Lagrange formulation as

$$\mathbf{M}(\boldsymbol{\phi}, \mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\boldsymbol{\phi}, \mathbf{q}) + \boldsymbol{\xi}(\boldsymbol{\phi}, \dot{\mathbf{q}}) = \boldsymbol{\tau}, \tag{2.22}$$

where the left-hand side only contains motion-induced forces, and the right-hand side all other forces. The elements of this are as follows:

- $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$: the first and second time derivative of the system configuration $\mathbf{q}$, respectively
- $\boldsymbol{\phi}$: the inertial and frictional parameters
- $\mathbf{M}(\boldsymbol{\phi}, \mathbf{q}) \in \mathbb{R}^{n \times n}$: the mass matrix
- $\mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$: the centripetal and Coriolis effects matrix
- $\mathbf{g}(\boldsymbol{\phi}, \mathbf{q}) \in \mathbb{R}^n$: the gravitational force vector
- $\boldsymbol{\xi}(\boldsymbol{\phi}, \dot{\mathbf{q}}) \in \mathbb{R}^n$: the friction torque vector
- $\boldsymbol{\tau} \in \mathbb{R}^n$: the applied force vector

The gravity, friction, Coriolis and centripetal effects are often summed in the bias torque vector $\mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$

$$\mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\boldsymbol{\phi}, \mathbf{q}) + \boldsymbol{\xi}(\boldsymbol{\phi}, \dot{\mathbf{q}}), \tag{2.23}$$

and the applied forces are often expanded as a sum of external forces $\mathbf{f}_e \in \mathbb{R}^e$ and joint actuation torques $\mathbf{f}_a \in \mathbb{R}^a$ as

$$\boldsymbol{\tau} = \mathbf{S}^T\mathbf{f}_a + \mathbf{J}_g^T(\mathbf{q})\mathbf{f}_e, \tag{2.24}$$

where $\mathbf{S} \in \mathbb{F}_2^{n \times a}$ is the actuated joint selection matrix and $\mathbf{J}(\mathbf{q})_g^T \in \mathbb{R}^{n \times e}$ the transpose of the geometric Jacobian that maps $\mathbf{f}_e$ to joint space. Using Eq. (2.24) and Eq. (2.23), Eq. (2.22) can be written as:

$$\mathbf{M}(\boldsymbol{\phi}, \mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}^T\mathbf{f}_a + \mathbf{J}_g^T(\mathbf{q})\mathbf{f}_e. \tag{2.25}$$

2.2 Least squares for parameter identification

This section introduces the concept of linear least squares (LS) estimation for the purpose of parameter identification.

2.2.1 Least squares parameter estimation

As shown by [10], the system dynamics from eq. Eq. (2.25) can be expressed linearly w.r.t. the inertial parameters $\boldsymbol{\phi}$, i.e.

$$\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\phi} = \boldsymbol{\tau} \tag{2.26}$$

where $\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times p}$ is a matrix representing the systems current motion, which will be constructed in section Sec. 2.2.2. In more detail, the inertial parameter vector $\boldsymbol{\phi}$ is the concatenated vector of inertial and frictional parameters of the body and links, ie. $\boldsymbol{\phi} = \begin{bmatrix}\boldsymbol{\phi}_b^T & \boldsymbol{\phi}_1^T & \boldsymbol{\phi}_2^T & ... & \boldsymbol{\phi}_j^T & \mathbf{d}_v^T & \mathbf{d}_c^T\end{bmatrix}^T$, with

$$\boldsymbol{\phi}_i = \begin{bmatrix}m_i & \mathbf{c}_i^T m_i & l(\mathbf{I}_i)^T\end{bmatrix}^T, \qquad \forall i \in b, 1, 2, \ldots, j \tag{2.27}$$

and

$$\mathbf{d}_v = \begin{bmatrix}d_{v_1} & d_{v_2} & \ldots & d_{v_j}\end{bmatrix}^T, \qquad \mathbf{d}_c = \begin{bmatrix}d_{c_1} & d_{c_2} & \ldots & d_{c_j}\end{bmatrix}^T \tag{2.28}$$

where m_i is the mass of link i, $\mathbf{c}_i m_i$ the first moment of inertia, i.e. $\mathbf{c}_i = \begin{bmatrix}c_{x_i} & c_{y_i} & c_{z_i}\end{bmatrix}^T$ is the center of mass position as seen from the link origin, $l(\mathbf{I}_i) = \begin{bmatrix}I_{xx_i} & I_{xy_i} & I_{xz_i} & I_{yy_i} & I_{yz_i} & I_{zz_i}\end{bmatrix}^T$ are the second moments and the products of inertia, and d_{v_i} and d_{c_i} are the viscous and coulomb friction coefficients, respectively. This makes $\boldsymbol{\phi}_i \in \mathbb{R}^{10}$ and $\mathbf{d}_v, \mathbf{d}_c \in \mathbb{R}^j$ and therefore $p = 10 + 12j$. These are referred to as the 'standard inertial parameters'.
Note that eq. Eq. (2.26) only holds under the assumption that friction appears linearly w.r.t. $\mathbf{K}$, all quantities are noise-free, the robot links are rigid and there is no backlash in the system [11].

If N samples are recorded during a motion of the joints, $\mathbf{K}$ and $\boldsymbol{\tau}$ can be constructed for each of the samples and stacked, such that for the complete trajectory their relation is described by

$$\mathbf{W}\boldsymbol{\phi} = \mathbf{y}, \tag{2.29}$$

with

$$\mathbf{W} = \begin{bmatrix}\mathbf{K}_1 \\ \mathbf{K}_2 \\ \vdots \\ \mathbf{K}_N\end{bmatrix} \in \mathbb{R}^{(nN) \times p}, \qquad \mathbf{y} = \begin{bmatrix}\boldsymbol{\tau}_1 \\ \boldsymbol{\tau}_2 \\ \vdots \\ \boldsymbol{\tau}_N\end{bmatrix} \in \mathbb{R}^{nN}.$$

In practice, measurement noise and modelling errors cause $\mathbf{y}$ to have some residual component $\boldsymbol{\epsilon} \in \mathbb{R}^{nN}$ such that

$$\mathbf{W}\boldsymbol{\phi} + \boldsymbol{\epsilon} = \mathbf{y}. \tag{2.30}$$

If $N > \frac{p}{n}$, eq. Eq. (2.30) will be an over-determined system of equations, which generally has no exact solution. However, a solution can be approximated by minimizing the squared l_2-norm of the residual $||\boldsymbol{\epsilon}||_2^2 = \epsilon_1^2 + \epsilon_2^2 + \ldots + \epsilon_{nN}^2$ to find an estimation $\hat{\boldsymbol{\phi}}$ of $\boldsymbol{\phi}$, as

$$\hat{\boldsymbol{\phi}} = \arg\min_{\hat{\boldsymbol{\phi}}} ||\boldsymbol{\epsilon}||_2^2 = \arg\min_{\hat{\boldsymbol{\phi}}} (\mathbf{y} - \mathbf{W}\hat{\boldsymbol{\phi}})^T(\mathbf{y} - \mathbf{W}\hat{\boldsymbol{\phi}}). \tag{2.31}$$

This is referred to as the Ordinary Least Squares (OLS) solution. We can find it by setting the partial derivatives w.r.t. $\boldsymbol{\phi}$ to zero, finally arriving at

$$\hat{\boldsymbol{\phi}}_{OLS} = \mathbf{W}^{+}\mathbf{y} = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{y}, \tag{2.32}$$

involving the Moore-Penrose pseudoinverse of $\mathbf{W}$, $\mathbf{W}^{+}$. This results in a solution where also $||\hat{\boldsymbol{\phi}}||_2^2$ is minimized [12]. Therefore, the result is in general not physically consistent. Note that this solution is only valid if we assume that $\boldsymbol{\epsilon}$ has zero-mean, is heteroskedastic and serially uncorrelated [11].

The regressor matrix $\mathbf{W}$ is in general not full column rank, and therefore $(\mathbf{W}^T\mathbf{W})$ will not be invertible. Furthermore, the regressor matrix will usually be scaled by left or right multiplication with a matrix. Both of these topics will be addressed in the paper of chapter 3.

2.2.2 Building the regressor matrix

In most literature about parameter identification, the matter of building the regressor matrix is not addressed. However, this is a complicated task and crucial for parameter identification. Most information from this section is gathered from [13], where the process is explained for fixed-base systems. In this section, the method is explained and adapted to floating-base kinematic-tree systems, starting from the linearity in parameters. For the full derivation, the reader is referred to above-mentioned source.

The EoM of the last link j of a kinematic chain, can be expressed using Eq. (2.18) in Ψ_j coordinates as

$$\mathbf{f}_j = \mathbf{I}_j\mathbf{a}_j^0 + \mathbf{v}_j^0 \times \mathbf{I}_j\mathbf{v}_j^0 \tag{2.33}$$

where $\mathbf{v}_j^0$ and $\mathbf{a}_j^0$ are the spatial velocity and acceleration, respectively, and $\mathbf{I}_j$ the spatial inertia expressed in the link j origin. All motion quantities are w.r.t. the inertial frame, so the superscript ^{0} will be omitted from now on.

Using Eq. (2.17) and substituting the frame j origin velocity and acceleration in Eq. (2.5), the first term can be expanded as

$$\mathbf{I}_j\mathbf{a}_j = \begin{bmatrix} \bar{\mathbf{I}}_j & m_j\mathbf{S}(\mathbf{c}_j) \\ \mathbf{S}(\mathbf{c}_j)^T & m_j\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\omega}}_j \\ \ddot{\mathbf{p}}_j - \boldsymbol{\omega}_j \times \dot{\mathbf{p}}_j \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{I}}_j\dot{\boldsymbol{\omega}}_j + m_j\mathbf{S}(\mathbf{c}_j)(\ddot{\mathbf{p}}_j - \boldsymbol{\omega}_j \times \dot{\mathbf{p}}_j) \\ \mathbf{S}(\mathbf{c}_j)^T\dot{\boldsymbol{\omega}}_j + m_j(\ddot{\mathbf{p}}_j - \boldsymbol{\omega}_j \times \dot{\mathbf{p}}_j) \end{bmatrix} \tag{2.34}$$

The second term of Eq. (2.33) can be expanded using Eq. (2.13) and Eq. (2.5) as

$$\begin{aligned} \mathbf{v}_j \times \mathbf{I}_j\mathbf{v}_j = -\mathbf{S}(\mathbf{v}_j)^T\mathbf{I}_j\mathbf{v}_j &= \begin{bmatrix} \mathbf{S}(\boldsymbol{\omega}_j) & \mathbf{S}(\dot{\mathbf{p}}_j) \\ \mathbf{0} & \mathbf{S}(\boldsymbol{\omega}_j) \end{bmatrix} \begin{bmatrix} \bar{\mathbf{I}}_j & m_j\mathbf{S}(\mathbf{c}_j) \\ \mathbf{S}(\mathbf{c}_j)^T & m_j\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_j \\ \dot{\mathbf{p}}_j \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{S}(\boldsymbol{\omega}_j)\bar{\mathbf{I}}_j\boldsymbol{\omega}_j + m_j\mathbf{S}(\mathbf{c}_j)\mathbf{S}(\boldsymbol{\omega}_j)\dot{\mathbf{p}}_j \\ \mathbf{S}(\boldsymbol{\omega}_j)m_j\mathbf{S}(\mathbf{c}_j)^T\boldsymbol{\omega}_j + \mathbf{S}(\boldsymbol{\omega}_j)m_j\dot{\mathbf{p}}_j \end{bmatrix}. \end{aligned} \tag{2.35}$$

Combining Eq. (2.34) and Eq. (2.35) and simplifying results in the expanded form of Eq. (2.33)

$$\mathbf{f}_j = \begin{bmatrix} \bar{\mathbf{I}}_j\dot{\boldsymbol{\omega}}_j + \mathbf{S}(\boldsymbol{\omega}_j)\bar{\mathbf{I}}_j\boldsymbol{\omega}_j - \mathbf{S}(\ddot{\mathbf{p}}_j)m_j\mathbf{c}_j \\ m_j\ddot{\mathbf{p}}_j + \mathbf{S}(\dot{\boldsymbol{\omega}}_j)m_j\mathbf{c}_j + \mathbf{S}(\boldsymbol{\omega}_j)\mathbf{S}(\boldsymbol{\omega}_j)m_j\mathbf{c}_j \end{bmatrix} \tag{2.36}$$

This equation can be written linearly in the inertial parameters as

$$\mathbf{f}_j = \mathbf{A}_j(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\phi}_j, \tag{2.37}$$

where $\boldsymbol{\phi}_j$ is the vector of inertial parameters of link j, and $\mathbf{A}_j(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is a matrix capturing its current motion expressed in Ψ_j coordinates, such that

$$\mathbf{A}_j = \begin{bmatrix} \mathbf{0}_{3\times 1} & -\mathbf{S}(\ddot{\mathbf{p}}_j) & \mathbf{L}(\dot{\boldsymbol{\omega}}_j) + \mathbf{S}(\boldsymbol{\omega}_j)\mathbf{L}(\boldsymbol{\omega}_j) \\ \ddot{\mathbf{p}}_j & \mathbf{S}(\dot{\boldsymbol{\omega}}_j) + \mathbf{S}(\boldsymbol{\omega}_j)\mathbf{S}(\boldsymbol{\omega}_j) & \mathbf{0}_{3\times 6} \end{bmatrix}. \tag{2.38}$$

The $\mathbf{L}(\cdot)$ operator is defined as

$$\mathbf{L}(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & 0 & 0 & 0 \\ 0 & x_1 & 0 & x_2 & x_3 & 0 \\ 0 & 0 & x_1 & 0 & x_2 & x_3 \end{bmatrix} \tag{2.39}$$

Generalizing Eq. (2.37) to all links, the force 'felt' at joint i is the result of the forces induced by the motion of all distal links, such that

$${}^{i}\mathbf{f}_i = \sum_{l=i}^{j} {}^{i}\mathbf{f}_{il} = \sum_{l=i}^{j} {}^{i}\mathbf{X}_l^F \, {}^{l}\mathbf{A}_l \boldsymbol{\phi}_l, \tag{2.40}$$

where ${}^{i}\mathbf{f}_{il}$ is the force on joint i as a result of the motion of link l. If we put this in matrix-vector form, we get

$$\begin{bmatrix} {}^{i}\mathbf{f}_i \\ {}^{i+1}\mathbf{f}_{i+1} \\ \vdots \\ {}^{j}\mathbf{f}_j \end{bmatrix} = \begin{bmatrix} \mathbf{A}_i & {}^{i}\mathbf{X}_{i+1}^F \mathbf{A}_{i+1} & \dots & {}^{i}\mathbf{X}_j^F \mathbf{A}_j \\ \mathbf{0} & \mathbf{A}_{i+1} & \dots & {}^{i+1}\mathbf{X}_j^F \mathbf{A}_j \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_j \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_{i+1} \\ \vdots \\ \boldsymbol{\phi}_j \end{bmatrix}, \tag{2.41}$$

where $\mathbf{A}_i = {}^{i}\mathbf{A}_i$ for readability.
The formulation can be adapted to be valid for kinematic trees as well by including a function δ, such that

$$\begin{bmatrix} {}^{i}\mathbf{f}_i \\ {}^{i+1}\mathbf{f}_{i+1} \\ \vdots \\ {}^{j}\mathbf{f}_j \end{bmatrix} = \begin{bmatrix} \mathbf{A}_i & \delta_i(i+1)\, {}^{i}\mathbf{X}_{i+1}^F \mathbf{A}_{i+1} & \dots & \delta_i(j)\, {}^{i}\mathbf{X}_j^F \mathbf{A}_j \\ \mathbf{0} & \mathbf{A}_{i+1} & \dots & \delta_{i+1}(j)\, {}^{i+1}\mathbf{X}_j^F \mathbf{A}_j \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_j \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_{i+1} \\ \vdots \\ \boldsymbol{\phi}_j \end{bmatrix}, \tag{2.42}$$

where

$$\delta_i(l) = \begin{cases} \boldsymbol{I} & \text{if joint } l \text{ is distal to joint } i, \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{2.43}$$

A link i is said to be distal to another link j, if it is located farther along the kinematic chain from the base or root of the system, such that link j lies between the base and link i in the sequence of connected joints and links.

Generally only the joint torques can be measured, so not the full left-hand side is known. Therefore, $\mathbf{f}$ must be projected on the measured axes of the joint $\boldsymbol{\beta}$, reducing the system of equations to

$$\boldsymbol{\tau} = \mathbf{K}_\phi \boldsymbol{\phi}_\phi, \tag{2.44}$$

with

$$\boldsymbol{\tau}_i = \boldsymbol{\beta} \cdot {}^{i}\mathbf{f}_i, \qquad \mathbf{K}_{\phi,il} = \boldsymbol{\beta} \cdot \delta_i(l)\, {}^{i}\mathbf{X}_l^F \mathbf{A}_l \tag{2.45}$$

If DH-parameters are used for the kinematic model, a rotational joint will always turn around its z-axis. Therefore, the most relevant realizations of $\boldsymbol{\beta}$ are $\boldsymbol{\beta}_r = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$ and $\boldsymbol{\beta}_f = \mathbf{1}_{6\times 1}$ for a rotational joint and a floating base, respectively.

Until now the friction has been ignored. If we only assume friction in the joints modelled as a combination of dry and viscous friction, Eq. (2.44) changes to

$$\boldsymbol{\tau} - \boldsymbol{\tau}_f = \mathbf{K}_\phi \boldsymbol{\phi}_\phi, \tag{2.46}$$

where $\boldsymbol{\tau}_f$ is the friction torque vector with only entries at rotational joint indices, such that $\boldsymbol{\tau}_f = \begin{bmatrix} \mathbf{0}_{1\times 6} & f_{f,1} & f_{f,2} & \dots & f_{f,j} \end{bmatrix}^T$ and

$$f_{f,i} = d_{v,i}\,\dot{\theta}_i + d_{c,i}\,\mathrm{sign}(\dot{\theta}_i), \tag{2.47}$$

where $\dot{\theta}_i$ is the rotational velocity of the joint, d_v the viscous friction coefficient and d_c the dry (or Coulomb) friction coefficient. The friction can be integrated into $\mathbf{K}$ by adding columns and adding all viscous and dry friction coefficients to $\boldsymbol{\phi}$, i.e.

$$\boldsymbol{\tau} = \mathbf{K}_\phi \boldsymbol{\phi} + \boldsymbol{\tau}_f = \begin{bmatrix} \mathbf{K}_\phi & \mathbf{K}_d \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_\phi \\ \mathbf{d}_v \\ \mathbf{d}_c \end{bmatrix} = \mathbf{K}\boldsymbol{\phi}, \tag{2.48}$$

where $\mathbf{d}_v$ and $\mathbf{d}_c$ are vectors with the viscous and dry friction coefficients of each joint, respectively, and

$$\mathbf{K}_d = \begin{bmatrix} \mathbf{0}_{6\times j} & \mathbf{0}_{6\times j} \\ \mathrm{diag}(\dot{\boldsymbol{\theta}}) & \mathrm{diag}(\mathrm{sign}(\dot{\boldsymbol{\theta}})) \end{bmatrix}, \tag{2.49}$$

where $\dot{\boldsymbol{\theta}} = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dots & \dot{\theta}_j \end{bmatrix}$. Note that zeros are added in Eq. (2.49) to account for the fact that the 6-DOF joint of the floating base has no joint friction.

Finally, we arrived at Eq. (2.26), which can be stacked to form Eq. (2.29). It must be noted that the friction can be modelled in any way, as long as it is linear in its parameters, i.e. the system can be written as $\boldsymbol{\tau} = \mathbf{K}\boldsymbol{\phi}$.

Chapter 3

Paper

Least Squares Inertial Parameter Identification of Bipedal Systems: A Simulation-Based Comparison

N. Van der Gaag

***Abstract*—This work presents an evaluation of a least squares approach to inertial parameter estimation of a bipedal system. Four different methods for scaling the regressor matrix are implemented and compared on their usability for centroidal dynamics control. The comparison is done using simulation data of an under-actuated lower-limb exoskeleton with limited range of motion. The results show that scaling the rows of the regressor matrix with the uncertainty in the joints, and columns with a prior estimate before solving the system, increases the accuracy of predictions of generalized forces, joint torque, and center of mass position and momentum. Additionally, scaling the columns has a large positive influence on the estimated parameter accuracy, but this depends heavily on the accuracy of the prior estimation.**

***Index Terms*—System identification, inertial parameter estimation, centroidal dynamics control**

I. Introduction

Lower-limb exoskeletons (LLEs) can play a pivotal role in enhancing the autonomy of paraplegic individuals by restoring their ability to walk and engage in daily activities [1], [2]. By moving the paralyzed legs, LLEs are not only facilitating mobility but are also helping to mitigate secondary health issues associated with immobility [1], [3], [4].

Despite their potential, LLEs are less commonly employed than wheelchairs. An important limitation is their current inability to provide robust balance support. Challenges arise particularly on slippery or uneven terrain, where the exoskeleton's response to disturbances is inadequate. Consequently, paraplegic users must rely on crutches and external assistance to maintain or regain balance, diminishing the exoskeleton's appeal as a tool for regaining independence.

Current research primarily focuses on the pre-calculation or recording of joint trajectories that replicate a natural gait, adjusting these in response to unexpected perturbations [5]–[7]. While these approaches are promising, they fall short in ensuring balance in some conditions.

A. Centroidal dynamics control

Authors [8], [9] and more recently [10] have shown that controlling the dynamics of the Center of Mass (CoM) and Whole Body Angular Momentum (WBAM) of bipedal systems is effective in maintaining balance under destabilizing conditions. Controlling the CoM and WBAM dynamics can be beneficial, because it simplifies the high-dimensional task of controlling all joints to a high-level, low-dimensional task. As a result, it can clear the path for more complex tasks, such as walking, where the CoM plays a large role in maintaining stability.

B. Dynamic model identification

Controlling the centroidal dynamics of a system requires an accurate description of its CoM. More generally, model based approaches for control benefit from having an accurate dynamic model available. Therefore, gaining knowledge of the Body Segment Inertial Parameters (BSIPs) is a crucial step in the design of such controllers. The BSIPs can be acquired from Computer Aided Design (CAD) data or found by performing measurements on the individual parts of the robot. However, these methods can be inaccurate and cumbersome if the robot must be disassembled, so BSIPs are often estimated from movement data using statistical learning algorithms [11].

Roughly two implementations of these algorithms can be distinguished: offline and online identification. The first focusses on processing pre-collected data to derive the system's characteristics. On the other hand, online identification involves continuously updating the model parameters in real-time as new data becomes available. Since each method has its strengths and weaknesses, exploiting the benefits of both using a hybrid implementation could be a step in the direction of robust balance control for LLEs.

Another topic of interest for an accurate dynamic model, is finding the kinematic parameters. Although this work will focus on finding the inertial parameters, the reader should be warned that the availability of an accurate kinematic model is crucial for estimating the inertial parameters, as well as controlling the centroidal dynamics. If interested, for example [12] can provide useful methods for estimating kinematic parameters.

C. Related work

The subject of system identification of bipedal systems is a heavily researched topic. Perhaps the most widely used technique is to make use of the (under certain assumptions) linearity in the inertial parameters (Sec. II-B) to perform a least squares (LS) optimization. In [13], the inertial parameters were estimated using only joint torques, only ground reaction forces (GRFs) and both measurements combined. While joint torques only performed best in some cases, the combined measurements performed better in general. In [14], the inertial parameters are estimated using only the dynamics of the floating base, to eliminate the need for joint torque estimates.

Some authors make use of constrained optimization techniques to enforce the physical consistency of the estimated parameters. In [15] a hierarchical optimization is performed to find estimates of the inertial parameters under physicality constraints. In [16] the inertial parameters of an LLE are estimated using a constrained quadratic program (QP) to enforce physical consistency. Notable is that in this work only affordable sensors are used for kinematic and inertial parameter identification, and inertial parameters of both the LLE and human in it.

While it seems desirable to have physically consistent estimates, [17] showed that the constrained problem is ill-posed, and estimates will often lie on the user-specified boundaries of physical consistency [18] [19]. Moreover, [17] provides an alternative using differential geometry. While the method was proven to be effective in finding physically consistent parameters, it is computationally much more expensive and thus not suitable for online identification. Furthermore, the optimization suffers because it is non-convex and has multiple local minima. Some of these issues are addressed in their more recent work [20].

Although a variety of other methods exist, most remain evaluated only on fixed-base systems. A broad overview of methods is given by [11], in which each method is evaluated on multiple fixed-base systems, based on noise immunity, estimation accuracy, convergence and numerical complexity. The author also presents a decision process which gives the desired identification procedure for a specific scenario.

D. System under evaluation

The system used in simulation for this work is the Symbitron exoskeleton from the University of Twente [1] as shown in Fig. 1, with modifications made by [10]. It consists of eleven links, interconnected by ten rotational joints. Therefore, it has a total of $n = 16$ degrees of freedom (DOFs): three body position, three body rotation and ten rotational. The five joints per leg allow the following movements: hip ab-/adduction and flexion/extension, knee flexion/extension, ankle plantar-/dorsi-flexion and ankle in-/eversion. Each joint is equipped with a position measuring device and, except for the ankle in-/eversion, are torque-actuated. The body contains an inertial measurement unit (IMU) and a gyroscope at the body link to sense its linear acceleration, angular velocity and orientation with respect to the world. For experiments, the ground reaction forces (GRFs) can be measured using two force-sensing platforms, one for each foot.

E. Contribution of this work

The aim of this research is to evaluate different implementations of least-squares inertial parameter identification, based on the usability of their results in a centroidal dynamics controller. Two sub-goals can be differentiated:

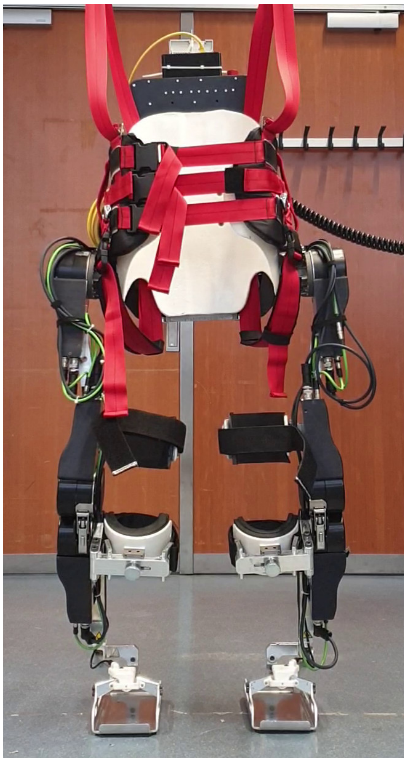


Fig. 1: The symbitron exoskeleton [10].

- Providing a proof of concept of parameter identification of the evaluated system, given its limited range of motion
- Compare different approaches in the context of centroidal dynamics control

F. Outline of this paper

Sec. II lays some foundations to explain the used methods in Sec. III. Sec. IV-A describes the performed simulation experiments, of which the results are presented in Sec. VI. Finally, conclusions are drawn in Sec. VII.

II. Preliminaries

This section will introduce some concepts that will lead to the used solution techniques in Sec. III.

A. System model

A schematic of the LLE is shown in Fig. 2. The links and body are considered rigid, so the dynamics can be modelled as those of a kinematic tree with a floating base, connected to the world by a 6-degree-of-freedom (DOF) joint. Therefore its joint configuration can be described by the 6-DOF joint coordinates $\mathbf{q}_b \in \mathbb{R}^7$ (a unit quaternion and position w.r.t. the world) for the body, with the joint configuration vector $\mathbf{q}_r \in \mathbb{R}^j$. Concatenating those gives the full system configuration

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_b^T & \mathbf{q}_r^T \end{bmatrix}^T \in \mathbb{R}^{n+1}. \quad (1)$$

The joint space formulation of the equation of motion (EOM) relates the joint motion quantities $\mathbf{q}$ and $\dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$

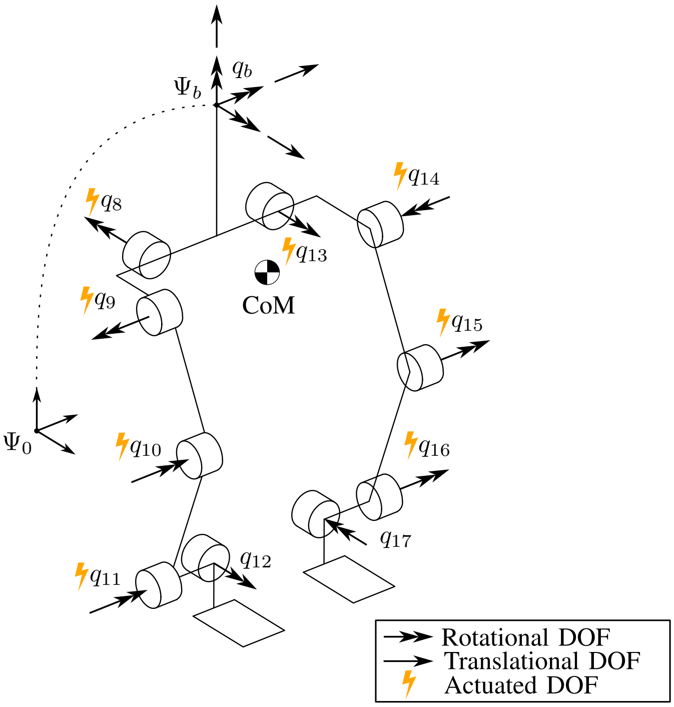


Fig. 2: A schematic of the LLE. The symbols Ψ_0 and Ψ_b denote the inertial and body coordinate frame, respectively.

to the applied forces $\boldsymbol{\tau}$ in the presence of inertial parameters $\boldsymbol{\phi}$. Using the Euler-Lagrange formalism, it can be expressed as [21]

$$\mathbf{M}(\boldsymbol{\phi}, \mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\boldsymbol{\phi}, \mathbf{q}) = \boldsymbol{\tau}, \tag{2}$$

with the variables as follows:

- $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$: the first and second time derivative of the system configuration $\mathbf{q}$, respectively
- $\mathbf{M}(\boldsymbol{\phi}, \mathbf{q}) \in \mathbb{R}^{n \times n}$: the mass matrix
- $\mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$: the centripetal and Coriolis effects matrix
- $\mathbf{g}(\boldsymbol{\phi}, \mathbf{q}) \in \mathbb{R}^{n}$: the gravitational torque vector
- $\boldsymbol{\tau} \in \mathbb{R}^{n}$: the applied force vector

In the context of rigid body dynamics, the gravity, Coriolis and centripetal effects are often summed in the bias torque vector $\mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n}$

$$\mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\boldsymbol{\phi}, \mathbf{q}), \tag{3}$$

and the applied forces are often expanded as a sum of external forces $\mathbf{f}_e \in \mathbb{R}^{e}$ and joint actuation torques $\mathbf{f}_a \in \mathbb{R}^{a}$ as

$$\boldsymbol{\tau} = \mathbf{S}^T \mathbf{f}_a + \mathbf{J}_g^T(\mathbf{q})\mathbf{f}_e, \tag{4}$$

where $\mathbf{S} \in \mathbb{F}_2^{n \times a}$ is the actuated joint selection matrix and $\mathbf{J}(\mathbf{q})_g^T \in \mathbb{R}^{n \times e}$ the transpose of the geometric Jacobian that maps $\mathbf{f}_e$ to joint space. Substitution of Eq. (4) and Eq. (3) in Eq. (2) gives:

$$\mathbf{M}(\boldsymbol{\phi}, \mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}^T \mathbf{f}_a + \mathbf{J}_g^T(\mathbf{q})\mathbf{f}_e. \tag{5}$$

For simplicity the dependencies of matrices and vectors $\mathbf{M}, \mathbf{C}, \mathbf{g}, \mathbf{S}, \mathbf{J}_g$ on the dynamic model and joint motion quantities $\boldsymbol{\phi}, \mathbf{q}, \dot{\mathbf{q}}$ are omitted in further evaluation.

B. Linearity in inertial parameters

As shown by [22], the system dynamics from Eq. (5) can be expressed linearly w.r.t. the inertial parameters, i.e.

$$\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\phi} = \boldsymbol{\tau} \tag{6}$$

where $\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times p}$ is a matrix representing the system's current motion, which is often called the model regressor. In more detail, $\boldsymbol{\phi}$ is a vector containing all inertial parameters of the body and links, ie. $\boldsymbol{\phi} = \begin{bmatrix} \boldsymbol{\phi}_b^T & \boldsymbol{\phi}_1^T & \boldsymbol{\phi}_2^T & \dots & \boldsymbol{\phi}_j^T \end{bmatrix}^T \in \mathbb{R}^p$, with

$$\boldsymbol{\phi}_i = \begin{bmatrix} m_i & \mathbf{c}_i^T m_i & \mathbf{I}_i^T \end{bmatrix}^T, \qquad \forall i \in b, 1, 2, \dots, j \tag{7}$$

where m_i is the mass of link i, $\mathbf{c}_i m_i$ the first moment of inertia, i.e. $\mathbf{c}_i = \begin{bmatrix} c_{x_i} & c_{y_i} & c_{z_i} \end{bmatrix}^T$ is the center of mass position as seen from the link origin, and $\mathbf{I}_i = \begin{bmatrix} I_{xx_i} & I_{xy_i} & I_{xz_i} & I_{yy_i} & I_{yz_i} & I_{zz_i} \end{bmatrix}^T$ are the second moments and the products of inertia. This makes $\boldsymbol{\phi}_i \in \mathbb{R}^{10}$, and therefore $p = 10(j+1)$. These are referred to as the 'standard inertial parameters'.

Note that Eq. (6) only holds under the assumption that all quantities are noise-free, the robot links are rigid and there is no backlash in the system [11].

C. Least squares estimation

If N samples are recorded during a motion of the joints, $\mathbf{K}$ and $\boldsymbol{\tau}$ can be constructed for each of the samples and stacked, such that for the complete trajectory their relation is described by

$$\mathbf{W}\boldsymbol{\phi} + \boldsymbol{\epsilon} = \mathbf{y}. \tag{8}$$

with residual $\boldsymbol{\epsilon} \in \mathbb{R}^{nN}$, and

$$\mathbf{W} = \begin{bmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \\ \vdots \\ \mathbf{K}_N \end{bmatrix} \in \mathbb{R}^{(nN) \times p}, \qquad \mathbf{y} = \begin{bmatrix} \boldsymbol{\tau}_1 \\ \boldsymbol{\tau}_2 \\ \vdots \\ \boldsymbol{\tau}_N \end{bmatrix} \in \mathbb{R}^{nN}.$$

If $N > \frac{p}{n}$, Eq. (8) will be an overdetermined system of equations, which generally has no exact solution. However, a solution can be approximated by minimizing the squared l_2-norm of the residual $||\epsilon||_2^2$, which results in

$$\hat{\phi}_{OLS} = \mathbf{W}^+ \mathbf{y} = (\mathbf{W}^T \mathbf{W})^{-1} \mathbf{W}^T \mathbf{y}, \tag{9}$$

involving the Moore-Penrose pseudoinverse of $\mathbf{W}$, $\mathbf{W}^+$. This solution is referred to as the Ordinary Least Squares (OLS) solution, which minimizes $||\hat{\boldsymbol{\phi}}||_2^2$ as well [15]. Therefore, the result is in general not physically consistent.

If $\mathbf{W}$ does not have full column rank, $(\mathbf{W}^T\mathbf{W})$ will not be invertible, which will be solved in Sec. III-A. Note that this solution is only valid if we assume that $\boldsymbol{\epsilon}$ has zero-mean, is heteroskedastic and serially uncorrelated [11].

D. Building the regressor matrix

$\mathbf{K}$ can be constructed from inertial forces of each link on another. The entry at the i^{th} row and l^{th} column, $\mathbf{K}_{il}$, is the generalized force on joint i as a result of the movement of link l, which can be expressed as [12]

$$\mathbf{K}_{il} = \boldsymbol{\beta} \cdot \delta_i(l)\, {}^i\mathbf{X}_l^F \mathbf{A}_l(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}), \tag{10}$$

where $(\cdot)$ denotes the dot product, $\boldsymbol{\beta}$ is the measured DOF, which for rotational joints and 6-DOF joint is $\boldsymbol{\beta}_r = \begin{bmatrix}0 & 0 & 1 & 0 & 0 & 0\end{bmatrix}^T$ and $\boldsymbol{\beta}_f = \mathbf{1}_{6\times1}$, respectively. The function $\delta_i(l)$ results in the identity matrix if l is a child link of i, and zeros otherwise. The matrix $\mathbf{A}_l(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ captures the current motion of link l, such that

$$\mathbf{A}_l = \begin{bmatrix} \mathbf{0}_{3\times1} & -\mathbf{S}(\ddot{\mathbf{p}}_l^0) & \mathbf{L}(\dot{\boldsymbol{\omega}}_l^0) + \mathbf{S}(\boldsymbol{\omega}_l^0)\mathbf{L}(\boldsymbol{\omega}_l^0) \\ \ddot{\mathbf{p}}_l^0 & \mathbf{S}(\dot{\boldsymbol{\omega}}_j^0) + \mathbf{S}(\boldsymbol{\omega}_l^0)\mathbf{S}(\boldsymbol{\omega}_l^0) & \mathbf{0}_{3\times6} \end{bmatrix} \tag{11}$$

where all variables are expressed in link l coordinates, and $\ddot{\mathbf{p}}_l^0$ is the acceleration of the origin of the link l frame with respect to the inertial frame, $\boldsymbol{\omega}_l^0$ and $\dot{\boldsymbol{\omega}}_l^0$ the angular velocity and acceleration, respectively, and the functions $\mathbf{S}(\cdot)$ and $\mathbf{L}(\cdot)$ are given by

$$\mathbf{S}(\mathbf{x}) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \tag{12}$$

$$\mathbf{L}(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & 0 & 0 & 0 \\ 0 & x_1 & 0 & x_2 & x_3 & 0 \\ 0 & 0 & x_1 & 0 & x_2 & x_3 \end{bmatrix}. \tag{13}$$

In Eq. (10), the motion matrix $\mathbf{A}_l$ is transformed from link j to link i coordinates by the so-called ‘spatial force transform’ ${}^i\mathbf{X}_l^F$, which is given by

$${}^i\mathbf{X}_l^F = \begin{bmatrix} {}^i\mathbf{R}_l & \mathbf{S}({}^i\mathbf{p}_l)\,{}^i\mathbf{R}_l \\ \mathbf{0} & {}^i\mathbf{R}_l \end{bmatrix} \tag{14}$$

with ${}^i\mathbf{R}_l$ the rotation matrix from l to i coordinates, and ${}^i\mathbf{p}_l$ the origin of the link l frame expressed in link i coordinates.

Now $\mathbf{K}$ can be constructed for each time step and stacked to form $\mathbf{W}$, as in Eq. (8).

E. Base parameters

As mentioned in [23] and [22], the column rank of $\mathbf{W}$ is determined by the geometric structure of the robot as well as the motion that is captured by $\mathbf{W}$. This makes that the regressor matrix will generally be column rank deficient. As a result, some inertial parameters will only appear in linear combinations in $\mathbf{y}$ and some will not be present at all. Therefore, a minimum number of necessary parameters m exists to reconstruct the joint torque for a given regressor matrix $\mathbf{W}$. These are often called ‘base parameters’ in literature. In contrast to the standard inertial parameters, base parameters do in general not have any physical meaning.

While analytical tools exist to determine the base parameters for a given system [24], literature usually resorts to numerical methods such as the QR-decomposition or Singular Value Decomposition (SVD) of the regressor matrix [23]. In this work, only the SVD is considered, since it gives some additional insights.

F. Singular value decomposition

Using the SVD, the regressor matrix can be decomposed as $\mathbf{W} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, where $\mathbf{U}$ and $\mathbf{V}$ are both orthogonal matrices, and $\boldsymbol{\Sigma} = \begin{bmatrix}\mathbf{S}^T & \mathbf{0}\end{bmatrix}^T$ is diagonal with $\mathbf{S}_{ii} = s_i$ the singular values of $\mathbf{W}$, where $s_1 \geq s_2 \geq ... \geq s_p \geq 0$.

Since $\mathbf{W}$ is generally rank deficient, there will only be $m < p$ non-zero singular values and it is only necessary to identify the m base parameters to completely reconstruct the joint torque for the given $\mathbf{W}$. Therefore, it is useful to divide $\mathbf{U}$, $\mathbf{S}$ and $\mathbf{V}$ in the part essential for reconstructing the joint torque, and in a part corresponding to the nullspace:

$$\mathbf{U} = \begin{bmatrix}\mathbf{U}_e & \mathbf{U}_n\end{bmatrix}, \mathbf{S} = \begin{bmatrix}\mathbf{S}_e & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_n\end{bmatrix}, \mathbf{V} = \begin{bmatrix}\mathbf{V}_e & \mathbf{V}_n\end{bmatrix} \tag{15}$$

with $\mathbf{U}_e \in \mathbb{R}^{N\cdot n\times m}$, $\mathbf{S}_e \in \mathbb{R}^{m\times m}$, $\mathbf{V}_e \in \mathbb{R}^{p\times m}$ and subscript e and n denote the essential and nullspace parts, respectively. The values on the main diagonal of $\mathbf{S}_n$ will be (close to) zero.

III. Solution methods

This section will provide methods to solve for the inertial parameters without taking the inverse $(\mathbf{W}^T\mathbf{W})^{-1}$ directly, as in Eq. (9).

A. Solving the LS problem using the SVD

The SVD offers a solution to the least squares problem in which $(\mathbf{W}^T\mathbf{W})^{-1}$ does not have to be calculated explicitly. Multiplying Eq. (8) with $\mathbf{U}^T$ and using the SVD on $\mathbf{W}$ yields

$$\mathbf{U}^T\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T\boldsymbol{\phi} + \mathbf{U}^T\boldsymbol{\epsilon} = \boldsymbol{\Sigma}\mathbf{V}^T\boldsymbol{\phi} + \mathbf{U}^T\boldsymbol{\epsilon} = \mathbf{U}^T\mathbf{y}. \tag{16}$$

Where the orthogonality of $\mathbf{U}$ is used. Substitution of Eq. (21) and using $\mathbf{V}^T\mathbf{V} = \boldsymbol{I}$ (identity) yields

$$\boldsymbol{\Sigma}\boldsymbol{\alpha} + \mathbf{U}^T\boldsymbol{\epsilon} = \mathbf{U}^T\mathbf{y}. \tag{17}$$

Since the orthogonality of $\mathbf{U}$ implies that $\mathbf{U}^T\boldsymbol{\epsilon}$ preserves the Euclidean length of $\boldsymbol{\epsilon}$, it is possible to minimize $||\mathbf{U}^T\boldsymbol{\epsilon}||_2^2$ instead:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} ||\mathbf{U}^T\boldsymbol{\epsilon}||_2^2 = \arg\min_{\boldsymbol{\alpha}} ||\mathbf{U}^T\mathbf{y} - \boldsymbol{\Sigma}\boldsymbol{\alpha}||_2^2 \tag{18}$$

Substituting Eq. (15) and $\mathbf{d} = \mathbf{U}^T\mathbf{y}$ yields

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left|\left| \mathbf{d} - \begin{bmatrix} \mathbf{S}_e & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_n \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_e \\ \boldsymbol{\alpha}_n \end{bmatrix} \right|\right|_2^2 . \tag{19}$$

From this equation can be seen that the choice of $\hat{\boldsymbol{\alpha}}_n$ does not matter, since $\mathbf{S}_n = \mathbf{0}$, and that $||\mathbf{U}^T\boldsymbol{\epsilon}||_2^2$ is minimized, if

$$\hat{\alpha}_{e,i} = \frac{d_i}{s_i}, \qquad \forall i = 1, 2, \dots m. \tag{20}$$

The orthogonal matrix $\mathbf{V}^T$ can be used to rotate the base parameters to standard parameters space using

$$\hat{\boldsymbol{\phi}} = \mathbf{V}\hat{\boldsymbol{\alpha}} = \mathbf{V}_e\hat{\boldsymbol{\alpha}}_e + \mathbf{V}_n\hat{\boldsymbol{\alpha}}_n. \tag{21}$$

B. Exploiting the nullspace

The OLS solution of Eq. (20) minimizes $||\boldsymbol{\epsilon}||_2^2$, without providing any guarantees to the correctness of $\hat{\boldsymbol{\phi}}$. However, the estimated parameters will be used for control of the LLE, possibly without using the full EoM of Eq. (5), but rather its components. Therefore, it is preferred to find a $\hat{\boldsymbol{\phi}}$ that is close to the real physical values.

This can be achieved by changing the nullspace parameters $\hat{\boldsymbol{\alpha}}_n$, as this will not impact the residual $\boldsymbol{\epsilon}$. In [25] the nullspace parameters are altered so that the l_2-norm of the difference between the estimated parameters and a prior estimate $\bar{\boldsymbol{\phi}}$ is minimized:

$$\hat{\boldsymbol{\alpha}}_n = \arg\min_{\boldsymbol{\alpha}_n} ||\hat{\boldsymbol{\phi}} - \bar{\boldsymbol{\phi}}||_2^2 \tag{22}$$

which results in the solution for the nullspace parameters

$$\hat{\boldsymbol{\alpha}}_n = \mathbf{V}_n^+(\bar{\boldsymbol{\phi}} - \mathbf{V}_e\hat{\boldsymbol{\alpha}}_e) = \mathbf{V}_n^T(\bar{\boldsymbol{\phi}} - \mathbf{V}_e\hat{\boldsymbol{\alpha}}_e). \tag{23}$$

The standard parameters can again be found using Eq. (21). The prior estimate $\bar{\boldsymbol{\phi}}$ can be found from CAD data or previously collected measurements.

C. Scaling the least squares problem

Scaling the least squares problem is good practice in general, since it can solve some issues related to the OLS solution [26].

Overfitting can be reduced by left multiplication of $\mathbf{W}$ by a weight matrix $\mathbf{G}$ with the relative uncertainties. If $\mathbf{G}$ is diagonal, left multiplication amounts to scaling the elements of $\boldsymbol{\epsilon}$, as it becomes

$$\tilde{\boldsymbol{\epsilon}} = \mathbf{G}\boldsymbol{\epsilon} = \mathbf{G}\mathbf{y} - \mathbf{G}\mathbf{W}\boldsymbol{\phi}, \tag{24}$$

effectively changing the norm with which the estimation is evaluated. The new minimization problem yields,

$$\hat{\boldsymbol{\phi}} = \arg\min_{\boldsymbol{\phi}} (\mathbf{y} - \mathbf{W}\boldsymbol{\phi})^T(\mathbf{G}^T\mathbf{G})(\mathbf{y} - \mathbf{W}\boldsymbol{\phi}) \tag{25}$$

with solution

$$\hat{\boldsymbol{\phi}} = (\mathbf{G}\mathbf{W})^+\mathbf{G}\mathbf{y}. \tag{26}$$

Note that in most cases it is numerically inefficient to go through the multiplication of $\mathbf{GW}$, since $\mathbf{G} \in \mathbb{R}^{(nN)\times(nN)}$ will mostly consist of zeros. It is much more efficient to scale the i^{th} row of $\mathbf{W}$ with $\mathbf{G}_{ii}$.

Additionally, the problem can be scaled by right multiplication of $\mathbf{W}$ by a diagonal matrix $\mathbf{H}$. This requires changing variables, such that

$$\tilde{\boldsymbol{\epsilon}} = \tilde{\mathbf{y}} - \tilde{\mathbf{W}}\tilde{\boldsymbol{\phi}} \tag{27}$$

with

$$\tilde{\mathbf{W}} = \mathbf{W}\mathbf{H}, \qquad \tilde{\mathbf{y}} = \mathbf{y} - \mathbf{W}\boldsymbol{\gamma}, \qquad \boldsymbol{\phi} = \mathbf{H}\tilde{\boldsymbol{\phi}} + \boldsymbol{\gamma}$$

and amounts to scaling the columns of $\mathbf{W}$. A solution for $\tilde{\boldsymbol{\phi}}$ with $\tilde{\mathbf{W}}$ and $\tilde{\mathbf{y}}$ is found similarly as for $\hat{\phi}$ in Eq. (9).

Using right multiplication does not change the problem mathematically, i.e. $\hat{\boldsymbol{\phi}}$ minimizing $||\tilde{\mathbf{y}} - \tilde{\mathbf{W}}\tilde{\boldsymbol{\phi}}||$ is the same as when $||\mathbf{y} - \mathbf{W}\hat{\boldsymbol{\phi}}||$ is minimized. However, if $\mathbf{H}$ is not orthogonal, the singular values of $\tilde{\mathbf{W}}$ will be different from $\mathbf{W}$, which can have implications on selecting the base parameters, as will be explained in Sec. II-F [26].

There is not one best procedure for selecting $\mathbf{G}$ and $\mathbf{H}$. However, since scaling with $\mathbf{G}$ alters the norm with which $\hat{\boldsymbol{\phi}}$ is selected, it can be used to balance the uncertainty in the joint measurements. Loosely speaking, a row of $\mathbf{W}$ and corresponding $\mathbf{y}$ value is 'trusted' more when a larger weight is assigned to it. Therefore, one possibility is to scale the rows with the standard deviation of the corresponding joint torque sensor, such that for joint i,

$$\mathbf{G}_{ii} = \frac{1}{\sigma_{\tau,i}}, \tag{28}$$

with $\sigma_{\tau,i}$ is the standard deviation of the noise of the torque sensor in joint i. However, this does account for the fact that the regressor matrix $\mathbf{W}$ contains noisy measurements. Therefore, [27] proposed weighting every row with the standard deviation of the residual of the OLS solution, i.e.

$$\mathbf{G}_{ii} = \frac{1}{\hat{\sigma}_{\epsilon,i}}, \tag{29}$$

with $\hat{\sigma}_{\epsilon,i}$ the standard deviation of the residual in joint i of the OLS solution, such that

$$\hat{\sigma}^2_{\epsilon,i} = \frac{||\hat{\epsilon}_{OLS,i}||_2^2}{N-p} = \frac{||\mathbf{y}_i - \mathbf{W}_i\hat{\phi}_{OLS}||_2^2}{N-p}, \tag{30}$$

where $||\hat{\epsilon}_{OLS,i}||_2^2$ is the l^2-norm of the residual of joint i, and $\mathbf{y}_i$ and $\mathbf{W}_i$ the values of $\mathbf{y}$ and $\mathbf{W}$ corresponding to joint i. The resulting matrix $\mathbf{G}$ will have repeating values on its main diagonal:

$$\mathbf{G} = \text{diag}(\begin{bmatrix}\mathbf{B}_1 & \mathbf{B}_2 & \dots & \mathbf{B}_N\end{bmatrix}) \tag{31}$$

with

$$\mathbf{B}_i = \begin{bmatrix}\frac{1}{\hat{\sigma}_{\epsilon,1}} & \frac{1}{\hat{\sigma}_{\epsilon,2}} & \cdots & \frac{1}{\hat{\sigma}_{\epsilon,n}}\end{bmatrix}, \qquad i = 1, 2, \dots, N. \tag{32}$$

Right multiplication of $\mathbf{W}$ by $\mathbf{H}$ can be beneficial if the uncertainty in the initial guess of inertial parameters is known, or can be used to balance the singular values of $\mathbf{W}$ to have a condition number close to 1 [26]. However, knowing the uncertainty of the initial guess is only useful if the relative uncertainty is known, which is generally not the case. Scaling the condition number would have as effect that the solution $\tilde{\phi}$ is less susceptible to changes in $\tilde{\mathbf{W}}$ and $\tilde{\mathbf{y}}$, which is desirable. However, this requires $\mathbf{H} = \mathbf{T}^{-1}$ where $\mathbf{T}$ is the triangular matrix from the Householder triangularization, which is generally unknown.

In [25], right multiplication of $\mathbf{W}$ with the prior estimation $\bar{\phi}$ is proposed. This has as advantage that the singular values are scaled according to the influence of the corresponding base parameter on the torque prediction, which will be explained in more detail in Sec. III-D.

D. Truncated SVD method

The uncertainty in the estimated parameters can be found from the residual variance. If $\mathbf{W}$ is scaled as in Eq. (30), the residual variance in each joint should the same size. Therefore the variance can be determined by [25]

$$\sigma^2_\epsilon = \frac{||\boldsymbol{\epsilon}||_2^2}{N-m} = \frac{||\mathbf{U}^T\boldsymbol{\epsilon}||_2^2}{N-m}, \tag{33}$$

because of the preservation of length under multiplication of $\mathbf{U}^T$. Therefore, the variance of the i^{th} base parameter estimate is inversely related to the corresponding singular value:

$$\text{var}(\hat{\alpha}_i) = \frac{\sigma^2_\epsilon}{s_i^2}. \tag{34}$$

This shows that for a small singular value, the uncertainty in the corresponding base parameter will be large.

A solution to this problem is the truncated SVD method [26]. This method uses only $z < m$ singular values and base parameters to keep the uncertainty in the base parameters small. A similar analysis as in [25] can be done to determine the amount of singular values that should be taken into account to keep the residual small. If the amount of singular values z is too large, the estimation will result in an over-fit, which means that the estimated parameters are too specific for the data. If too few singular values are used, it will result in a under-fit, making the residual larger than necessary.

For the selection of base parameters, scaling of the columns of $\mathbf{W}$ is useful. The unscaled regressor matrix only contains data about the movement, but the selection should contain the base parameters with the most influence on the generalized forces. Therefore, scaling the columns of $\mathbf{W}$ with $\mathbf{H} = \text{diag}(\bar{\phi})$ before performing the SVD, could have a positive impact on this selection, as it should scale the singular values according to the expected influence on the generalized forces. Looking at Eq. (34), it will also make sure that the base parameters with the most expected influence on the generalized forces, will have the most accurate estimation.

E. Low-pass filtering

Various authors showed that low-pass filtering the noisy measurements has a positive impact on the estimation [11]. Therefore, this will be included in the comparison as well. The joint positions $\mathbf{q}_r$ will be filtered before differentiating to joint velocity and acceleration $\dot{\mathbf{q}}_r$ and $\ddot{\mathbf{q}}_r$, respectively. Additionally, the body rotational velocity is filtered before differentiating to rotational acceleration. The joint velocities, acceleration and body rotational acceleration are filtered again after the differentiation.

IV. Simulation experiments

The methods described in Sec. III are tested and compared based on their performance. This is done by applying them to simulation data of the LLE, and evaluating their output based on some important quantities for centroidal dynamics control, and inertial parameter identification in general.

A. Simulations

Separate data sets are obtained from simulation; training and validation data. For each data set, the LLE is tasked to perform 20 straight-lined movements of the CoM within its base of support (BoS), standing on both feet. A cubic trajectory is used as position reference of the CoM, which gradually increases and then decreases in velocity, to prevent the LLE from falling.

For controlling the CoM of the LLE, the controller from [10] is used, without the torso stabilization task to maximize torso movement. The controller runs on Simulink, and sends the torque signals via an UDP port to a Mujoco program, which simulates the dynamics and sends the measurements back to Simulink. Ideal measurements are taken of the joint positions and torques during each movement. Additionally, the body rotational velocity, linear acceleration and orientation w.r.t. the world is measured.

During post-processing, wrenches on both feet are estimated using a least squares estimation, to simulate force-plate measurements. Noise with the same variance as measurement data from the real LLE is added to the measurements of the training data. The validation data will only contain ideal measurements, to make the analysis unaffected by noise.

B. Comparison

The solution techniques of the least squares problem, established in this work are evaluated using the simulation data. An overview of the assessed techniques and if the nullspace (Eq. (21)) is used, is shown in Table I. Because the CoM position and momentum prediction is important for centroidal dynamics control, additional attention is given to the estimation of these quantities. The methods are evaluated based on the following attributes:

1) Generalized force prediction
2) Joint torque prediction
3) CoM position prediction
4) CoM momentum prediction
5) Estimation accuracy of the inertial parameters

Attributes 1 to 4 are all measured by their root-mean-squared error (RMSE). Attribute 5 is quantified by the average estimation error: $|\frac{\hat{\phi}-\phi}{\phi}| \times 100\%$.

Because the estimations partly rely on prior estimate $\bar{\phi}$, the identification process is repeated several times with different randomly generated prior estimates. Results are compared with $\bar{\phi}$ generated within ± 12.5% and ± 25% of the real ϕ value.

In each iteration, the optimal amount of singular values z is found for each weighting scheme to minimize the RMSE of the generalized force prediction $\hat{\mathbf{y}}$. The z singular values are used to estimate parameters. After 20 iterations, all metrics are calculated over all iterations. To provide a reference, the prior estimate $\bar{\phi}$ will be evaluated on all qualities as well.

The process is repeated with data that has been filtered with four different cut-off frequencies: 30, 50, 70 and 100 Hz. The applied filter is a second order low-pass Butterworth filter, which is applied phase-less using the `filtfilt()` command in Matlab.

TABLE I: Overview of evaluated solution techniques

Name	Solution technique	Nullspace used
Ref.	- (prior estimate $\bar{\phi}$)	-
OLS	OLS	Yes
WLS-R	WLS (rows scaled with $\frac{1}{\hat{\sigma}_{\epsilon,i}}$ as Eq. (31))	Yes
WLS-C	WLS (columns scaled with $\bar{\phi}$)	No
WLS-RC	WLS (combining WLS-R and WLS-C)	No

V. Results

Table II shows the best metric values for each solution method. Given that the metrics vary across different filter frequencies, only the best results are presented here. Therefore each value is still calculated over 20 iterations, but the filter frequency may not be the same for each metric.

Fig. 3 shows a CoM position prediction, Fig. 4 a joint torque prediction and Fig. 5 a CoM momentum prediction. All have been made using a WLS-RC estimation with prior estimation accuracy of ±25%, trained on filtered data with a cut-off frequencies of 70, 100 and 50 Hz, respectively. As a reference, the same predictions are done using $\bar{\phi}$, presented as 'Ref.'. The sample time in these figures is 0.02 s, but some data points are filtered out for reasons explained in Sec. VI. Therefore, the x-axis shows the sample number and not the time.

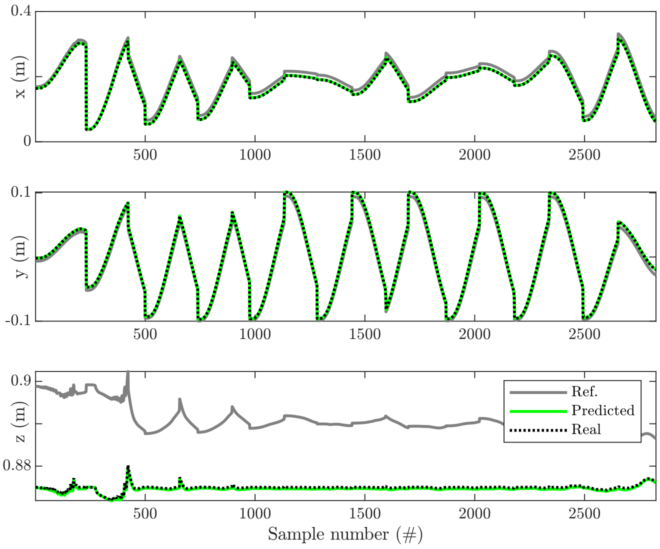


Fig. 3: CoM position prediction using an WLS-RC estimation.

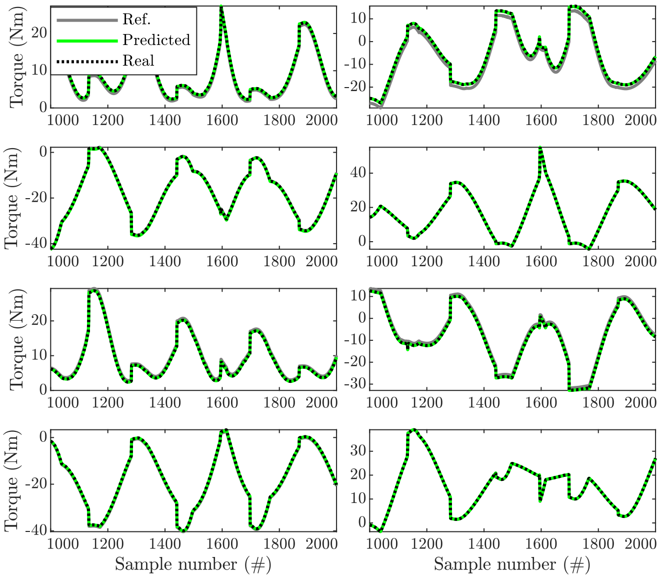


Fig. 4: Joint torque prediction using an WLS-RC estimation. Note that this prediction is made using perfect knowledge of the external forces to make the results independent of the measurement noise.

Finally, Fig. 6 shows the average estimation error with prior estimation accuracy of ±25%, for the each link separate. Link 1 corresponds to the body, followed by the right and left leg links, respectively. The leg links are numbered from proximal to distal, with link 2 and 7 corresponding to right and left first hip links, respectively.

VI. Discussion

This section presents a more elaborate analysis of the results, and possible implications of the methods on the results are discussed.

TABLE II: Best results after 20 iterations with different prior estimates.

Prior est. uncertainty	±12.5%					±25%				
Solution Method	OLS	WLS-R	WLS-C	WLS-RC	Ref.	OLS	WLS-R	WLS-C	WLS-RC	Ref.
Gen. force RMSE (N & Nm)	5.34	9.69	3.17	**1.45**	18.3	9.21	13.5	4.23	**1.85**	26
Torque RMSE (Nm)	2.47	5.18	0.784	**0.641**	1.99	4.11	7.91	1.03	**0.712**	3.61
CoM position RMSE (mm)	12	14	3.5	**1.0**	12	20	18	3.3	**1.0**	26
CoM momentum RMSE (Ns & Nms)	0.094	0.099	0.045	**0.022**	0.13	0.18	0.17	0.064	**0.029**	0.27
Avg. estimation error (%)	85.2	616	5.87	**5.74**	6.2	283	3480	11.5	**11.2**	12.5
Avg. amount of singular values (#)	19.9	19.7	5.81	10.3	-	23.2	33.1	8.35	13.3	-

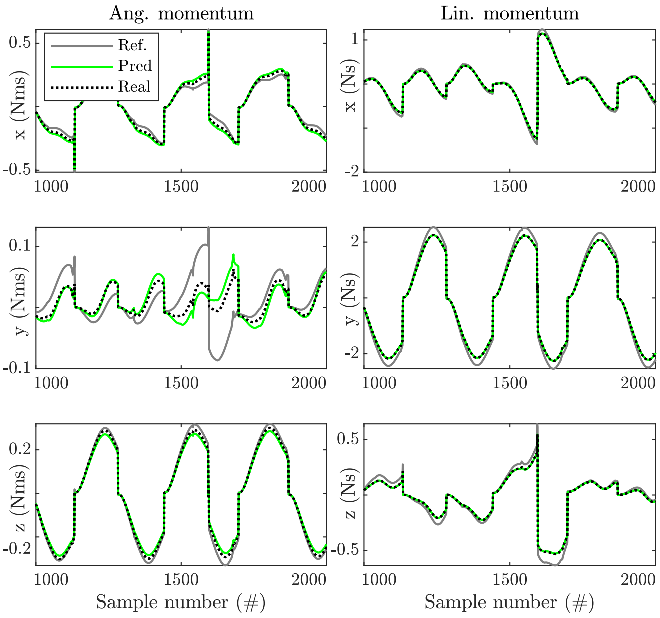


Fig. 5: CoM momentum prediction using an WLS-RC estimation over the validation data.

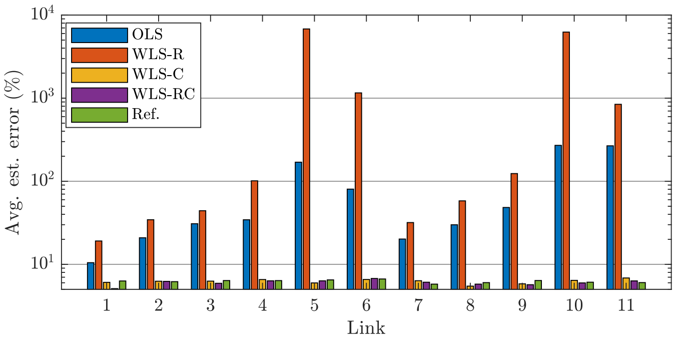


Fig. 6: The average estimation error over 20 iterations for each link.

A. Analysis of results

While the generalized force RMSE improves using all methods, the joint torque prediction does not. One explanation is that the least squares estimation minimizes the generalized force residual, so no direct attention is given to any of the other evaluation attributes. In additional analysis the prior estimation accuracy was decreased to ±50%, which showed that the gap between the performance of the reference and least squares estimations increases as the prior estimations become less accurate. However, these results are not included since prior estimations of most parameters are thought to be more accurate.

Table II the WLS-C and WLS-RC perform better on all attributes and therefore show that scaling the columns improves the estimations. The impact is especially large on the estimation accuracy. While scaling the columns should not mathematically change the minimization, it does change which singular values are taken into account with the truncated SVD method, and therefore the resulting estimations.

The table also shows that scaling the rows is beneficial. The row-scaled methods use more singular values and decrease the prediction errors, indicating that more information from the regressor matrix can be used for the estimation without overfitting on the data.

The effect of using the nullspace for more accurate estimations (Eq. (21)) can be observed through the difference in estimation error between OLS and WLS-R estimations. OLS uses fewer singular values, which leaves a larger nullspace. While OLS has better estimation accuracy, it is still low compared to the column-scaled methods, and therefore indicates that using the nullspace is not sufficient to produce accurate estimations.

Table II shows that the combination of row and column scaling is more effective for most metrics than the effects separately. This can be caused by the larger amount of singular values the WLS-RC is using, which is up to four times as high as WLS-C. Since the singular values are sorted on relevance, being able to use more results in much better predictions. The small influence of row scaling compared to no scaling at all accentuates the conclusion that minimizing that generalized force residual does not necessarily result in improved performance on the other metrics.

The column scaled methods show a large dependency on the prior estimation accuracy, looking at the estimation error. In tests with ±50% prior estimation accuracy, this trend was continued, indicating that the estimations will only be a little more accurate than the prior estimations.

Finally, Fig. 6 shows that there are large differences in the estimation accuracy over the links. This is caused by the fact that the distal links are almost stationary during all movements, which results in unexcited inertial parameters. This could be improved by including movement on one leg, such as walking. However, for that advancements are needed in the controller software.

B. Evaluation of methods

The analysis in this work is done using simulation data. Therefore, some real-world disturbances are omitted, which makes the conclustions drawn from this experiment less reliable than testing on a real LLE.

To begin with, it is assumed that all kinematic parameters are known very accurately. This makes that the used model could in theory be exactly identified. In the real world, such accurate measurements of kinematic parameters are hard to obtain. This could potentially have a large effect on the CoM prediction.

Also, prior estimations are generated within a percentage of the real values. In the real world, their accuracy would depend on measurement errors, or parts added after measurements or that are not modelled in CAD software. Therefore the prior estimations could be inaccurate in many ways, not depending on the 'real' parameter value.

Another issue in a real-world implementation is related to the measurement of the GRFs. Measurements would be obtained using force-sensing platforms that are not rigidly connected to the LLE, since no force sensors are present in the feet. However, an accurate mapping of the GRFs to joint space via $\mathbf{J}_g^T$ requires the position of the force platforms w.r.t. the LLE to be known, which could be hard to achieve.

Additionally, the noise is now assumed to be additive. However, especially for the force measurements this assumption is not valid, since in real life these are constructed from a set of strain gauges which are multiplied and added. Therefore, the total force measurement will contain some combination of the strain gauge noise, unknown to the author.

Furthermore, the amount of used singular values is determined using the validation data, which is also used to perform the analysis. In a real-world scenario, the identification could be done in a similar fashion, but the performance could not be determined using ideal measurements. One way to assess the performance would be using the parameters in a feed-forward controller and measuring the relative amount that it contributes to the total control signal. However, only the joint-torque prediction accuracy is measured with this method.

In the real LLE it would be hard to implement the movements used in simulation, since its inertial and kinematic parameters are not yet accurately known and might cause the LLE to fall. To mitigate this issue, in real experiments the identification could be done iteratively, increasing the amplitude of the movements closer to the edge of the BoS as more accurate inertial parameters are estimated.

To conclude, the IMU data is now taken as directly in the origin of the body frame. However, in the real LLE there will be a mapping from the IMU acceleration to body acceleration, involving multiplication of noisy measurements. Furthermore, in the simulation the orientation of the body is taken as ideal, since it is represented as a unit quaternion, which is in real-life measured from a combination of sensors. Also, the effects of different sampling of different sensors and their resolution are omitted.

One issue that could not be resolved were oscillations in the joints with much higher frequency than the movements, possibly leading to unreliable filtering of the data. Attempts to solve this issue involved adding damping to the control torques, but were ineffective. Therefore, an appropriate filter must be found when implementing the techniques into the real LLE.

While this work uses an LLE to evaluate identification methods, no attention has been given to human interaction. While [16] showed that human BSIPs can be estimated in a similar fashion, humans will introduce all kinds of disturbances, such as reflexes to maintain balance.

In this research, not much attention has been given to trajectory optimization, although multiple sources mention it as essential step in the identification [28] [MORE SOURCES]. Although it may not be applicable to this work, [25] showed that trajectory optimization mostly affects the excluded singular values and therefore has only little influence on the results. Trajectory optimization could be more relevant to systems with limited range of motion, such as the LLE in this work.

C. Future research

A natural extension of this work is to resolve the issues stated in Sec. VI-B, by including frictional parameters, and the parameters of the LLE user. Additionally, better results could be obtained using trajectory optimization and other methods for filtering the training data, possibly filtering in a way that preserves consistency of the EoM. Furthermore, the methods should be implemented in the real LLE to evaluate the methods in a real situation where modelling errors are present and less ideal signals can be used, and human parameters should also be estimated.

The amount of used singular values z is in this work chosen to minimize the generalized force RMSE. However, often other metrics, such as the torque RMSE, are more important and might be more interesting to minimize.

This research only implements one of many possible identification techniques. For example [11] gives an overview of some techniques, with a flow chart to choose one applicable to the situation. If the controller structure is known, other techniques might be more relevant than the one implemented in this work.

Next steps could include online methods, which if implemented correctly, could have benefits for example when a user is picking up an object. Also, looping the methods as described in this work, using an estimation as 'prior estimation' for a next estimation procedure on other data, could be investigated. Since one of the limiting factors in this work is that noisy joint velocities and accelerations are used, it could potentially benefit from using IMUs on the legs to obtain more accurate accelerations. Finally, to account for nonlinearities or other unmodelled dynamics, a combination with more advanced machine learning methods such as neural networks could be used.

VII. Conclusion

Feasibility of least squares estimation of the BSIPs for the evaluated system is demonstrated in this work. While all metrics can be improved by using WLS-RC compared to the reference, the estimation accuracy shows a large dependency on the prior estimates. Moreover, some techniques even showed decreased performance on some metrics. Therefore, experiments should evaluate the usability of the methods in the real world.

References

[1] C. Meijneke, G. Van Oort, V. Sluiter, E. Van Asseldonk, N. L. Tagliamonte, F. Tamburella, I. Pisotta, M. Masciullo, M. Arquilla, M. Molinari, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and H. Van Der Kooij, "Symbitron Exoskeleton: Design, Control, and Evaluation of a Modular Exoskeleton for Incomplete and Complete Spinal Cord Injured Individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 330–339, 2021.

[2] R. B. van Dijsseldonk, I. J. W. van Nes, A. C. H. Geurts, and N. L. W. Keijsers, "Exoskeleton home and community use in people with complete spinal cord injury," *Scientific Reports*, vol. 10, p. 15600, 9 2020.

[3] F. Tamburella, N. L. Tagliamonte, I. Pisotta, M. Masciullo, M. Arquilla, E. H. F. van Asseldonk, H. van der Kooij, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and M. Molinari, "Neuromuscular Controller Embedded in a Powered Ankle Exoskeleton: Effects on Gait, Clinical Features and Subjective Perspective of Incomplete Spinal Cord Injured Subjects," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 5, pp. 1157–1167, 2020.

[4] C. Baunsgaard, U. Nissen, A. Brust, A. Frotzler, C. Ribeill, Y. Kalke, N. León, B. Gómez, K. Samuelsson, W. Antepohl, U. Holmström, N. Marklund, T. Glott, A. Opheim, J. Penalva, N. Murillo, J. Nachtegaal, W. Faber, and F. Biering-Sørensen, "Exoskeleton gait training after spinal cord injury: An exploratory study on secondary health conditions," *Journal of Rehabilitation Medicine*, vol. 50, no. 9, pp. 806–813, 2018.

[5] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. van der Kooij, "Design and Control of the MINDWALKER Exoskeleton," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, pp. 277–286, 3 2015.

[6] T. Gurriet, S. Finet, G. Boeris, A. Duburcq, A. Hereid, O. Harib, M. Masselin, J. Grizzle, and A. D. Ames, "Towards Restoring Locomotion for Paraplegics: Realizing Dynamically Stable Walking on Exoskeletons," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2804–2811, IEEE, 5 2018.

[7] Q. Chen, H. Cheng, C. Yue, R. Huang, and H. Guo, "Dynamic Balance Gait for Walking Assistance Exoskeleton," *Applied Bionics and Biomechanics*, vol. 2018, pp. 1–10, 7 2018.

[8] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Autonomous Robots*, vol. 40, no. 3, pp. 473–491, 2016.

[9] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a Momentum-Based Control Framework and Application to the Humanoid Robot Atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.

[10] A. Vallinas, A. Keemink, C. Bayón, E. van Asseldonk, and H. van der Kooij, "Momentum-Based Balance Control of a Lower-Limb Exoskeleton During Stance," in *2023 International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–6, IEEE, 9 2023.

[11] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, "Inertial Parameter Identification in Robotics: A Survey," *Applied Sciences*, vol. 11, p. 4303, 5 2021.

[12] J. Hollerbach, W. Khalil, and M. Gautier, "Model Identification," *Springer Handbook of Robotics*, pp. 321–344, 2008.

[13] M. Mistry, S. Schaal, and K. Yamane, "Inertial parameter estimation of floating base humanoid systems using partial force sensing," *9th IEEE-RAS International Conference on Humanoid Robots*, pp. 492–497, 2009.

[14] K. Ayusawa, G. Venture, and Y. Nakamura, "Identification of the inertial parameters of a humanoid robot using unactuated dynamics of the base link," in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*, pp. 1–7, IEEE, 12 2008.

[15] J. Jovic, A. Escande, K. Ayusawa, E. Yoshida, A. Kheddar, and G. Venture, "Humanoid and Human Inertia Parameter Identification Using Hierarchical Optimization," *IEEE Transactions on Robotics*, vol. 32, pp. 726–735, 6 2016.

[16] R. Mallat, V. Bonnet, W. Huo, P. Karasinski, Y. Amirat, M. Khalil, and S. Mohammed, "Human-Exoskeleton System Dynamics Identification Using Affordable Sensors," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6759–6765, 2018.

[17] T. Lee and F. C. Park, "A Geometric Algorithm for Robust Multibody Inertial Parameter Identification," *IEEE Robotics and Automation Letters*, vol. 3, pp. 2455–2462, 7 2018.

[18] 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, 2018.

[19] S. Traversaro, S. Brossette, A. Escande, and F. Nori, "Identification of Fully Physical Consistent Inertial Parameters using Optimization on Manifolds," 4 2016.

[20] T. Lee, P. M. Wensing, and F. C. Park, "Geometric Robot Dynamic Identification: A Convex Programming Approach," *IEEE Transactions on Robotics*, vol. 36, pp. 348–365, 4 2020.

[21] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Dynamics and Control*. 2004.

[22] C. H. An, C. G. Atkeson, and J. M. Hollerbach, "Estimation of Inertial Parameters of Rigid Body Links of Manipulators," *Proceedings of the IEEE Conference on Decision and Control*, pp. 990–995, 1985.

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

[24] W. Khalil and E. Dombre, "Chapter 9 - Dynamic modeling of serial robots," in *Modeling, Identification and Control of Robots*, pp. 191–233, Oxford: Butterworth-Heinemann, 2002.

[25] R. Waiboer, R. Aarts, and B. Jonker, "Modelling and Identification of a Six Axes Industrial Robot," *ASME Proceedings — 5th International Conference on Multibody Systems, Nonlinear Dynamics, and Control*, 4 2005.

[26] C. L. Lawson and R. J. Hanson, "25. Practical Analysis of Least Squares Problems," in *Solving Least Squares Problems*, pp. 180–198.

[27] M. Gautier, "Dynamic identification of robots with power model," in *Proceedings of International Conference on Robotics and Automation*, vol. 3, pp. 1922–1927, IEEE, 1997.

[28] T. Lee, B. D. Lee, and F. C. Park, "Optimal excitation trajectories for mechanical systems identification," *Automatica*, vol. 131, p. 109773, 9 2021.

Chapter 4

Additions to paper

In this chapter, additional results are presented, as well as reasoning behind certain choices that are made. It also expands on some discussion points in the paper, about the methods used in the analysis.

4.1 Additional results

In chapter 3 some predictions of joint torques and CoM position and momentum are shown. These are made with ideal data which would be impossible in the real LLE. Therefore, to get an idea of the real performance, the same predictions are shown here, but using noisy measurement data as would be done in the real LLE.

First of all, the joint torque prediction in the real LLE is made using a prediction of the GRFs on the foot, and noisy position and velocity data, of which the latter is filtered using a second-order butterworth low-pass filter in real time. The GRFs can be predicted from the regressor matrix and noisy joint torques using a least squares optimization as follows:

$$\mathbf{f}_e^{i+1} = (\mathbf{J}_g(\mathbf{q}_d^{i+1})^T)^+(\mathbf{K}(\mathbf{q}_d^{i+1}, \dot{\mathbf{q}}_d^{i+1}, \ddot{\mathbf{q}}_d^{i+1})\hat{\boldsymbol{\phi}} - \mathbf{S}^T\mathbf{f}_a^i), \tag{4.1}$$

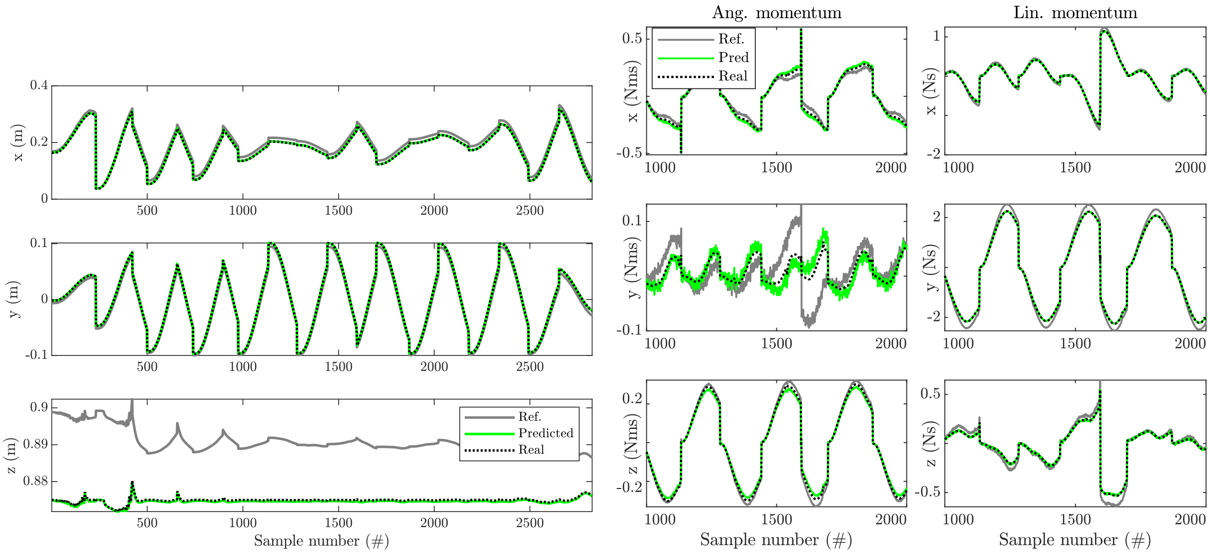
where subscript d indicates a desired joint motion quantity and superscript i and $i+1$ the current and next time step, respectively. The pseudo-inverse of the geometric Jacobian $(\mathbf{J}_g^T)^+$ is taken for a least-squares estimation, $\mathbf{K}$ is the current motion matrix, $\mathbf{S}^T$ the actuated joint selection matrix and $\mathbf{f}_a$ the actuation torques. Additionally, the torque prediction should give the necessary joint torques as a function of the desired accelerations, which are therefore put in as ideal values. The torque predictions over the same data as in chapter 3 are shown in Fig. 4.2.

The CoM position prediction only uses joint position data, which can be measured accurately (with a standard deviation of $9 \cdot 10^{-6}$ rad). Its performance with noisy data will therefore not vary visibly from chapter 3. Additionally, the real-time CoM momentum calculation in the LLE is done using the same joint velocities as for the torque predictions, and will therefore also contain some noise. Both the CoM position and momentum predictions using realistic data, are shown in Fig. 4.1.

In addition to the results in Table II in chapter 3, the estimations are done using prior estimations with an absolute offset from the real values, to see what will happen if the manual measurements would be off by an absolute number, instead of a percentage. A similar table as Table II from chapter 3, is shown in Table D.1. For these results, prior estimations are generated within ±0.125 of the real parameters (unit depends on the specific parameter). Note that these results are obtained from 20 iterations as well.
In addition to Table II in chapter 3, the full table of results is shown in appendix C. Finally, appendix D shows a table with full estimations using each solution method.

Table 4.1: Best results after 20 iterations with different prior estimates.

Solution Method	OLS	WLS-R	WLS-C	WLS-RC	Ref.
Gen. force RMSE (N & Nm)	24	12.9	22.6	**12.8**	29.4
Torque RMSE (Nm)	15	**7.05**	13.9	7.25	18.9
CoM position RMSE (mm)	6.5	4.8	8.5	**3.7**	11
CoM momentum RMSE (Ns & Nms)	0.084	**0.076**	0.11	0.094	0.13
Avg. estimation error (%)	22300	**8540**	17200	13200	22600
Avg. amount of singular values (#)	26.7	27.5	21.5	33.4	0



(a) Position.

(b) Momentum.

Figure 4.1: Realistic CoM predictions.

4.2 Additional discussion

4.2.1 Analysis of results

The reason that the torque predictions from Fig. 4.2 do not contain noise is twofold: the prediction is done using desired ‘ideal’ accelerations, which would otherwise be the largest source of noise. Additionally, the predictions are done using predicted GRFs that are obtained by an optimization. Therefore, the GRFs will contain noise to compensate for noise in the regressor matrix.

From a comparison of Fig. 4.2 and Fig. 3 from chapter 3 can be concluded that the torque prediction benefits more from the parameter estimations in the real situation than in the idealized simulation. The CoM predictions of Fig. 4.1 are compared to Fig. 4 of chapter 3 only affected by additional noise.

Table D.1 shows the large dependency of the column-scaled methods on the prior estimations. The WLS-R method does not have this dependency as much, which could explain why it performs better on some metrics as compared to the paper. However, WLS-RC still performs better on CoM position prediction and only marginally worse on CoM momentum prediction and torque prediction. Therefore, it can be concluded that WLS-RC is generally the best choice of these methods, since the error in the prior estimations could be some combination of absolute and relative.

4.2.2 Evaluation of methods

As mentioned in the discussion of chapter 3, the data contained some vibration of the legs in most movements, with high frequency compared to the prescribed motion. This leads to issues when filtering

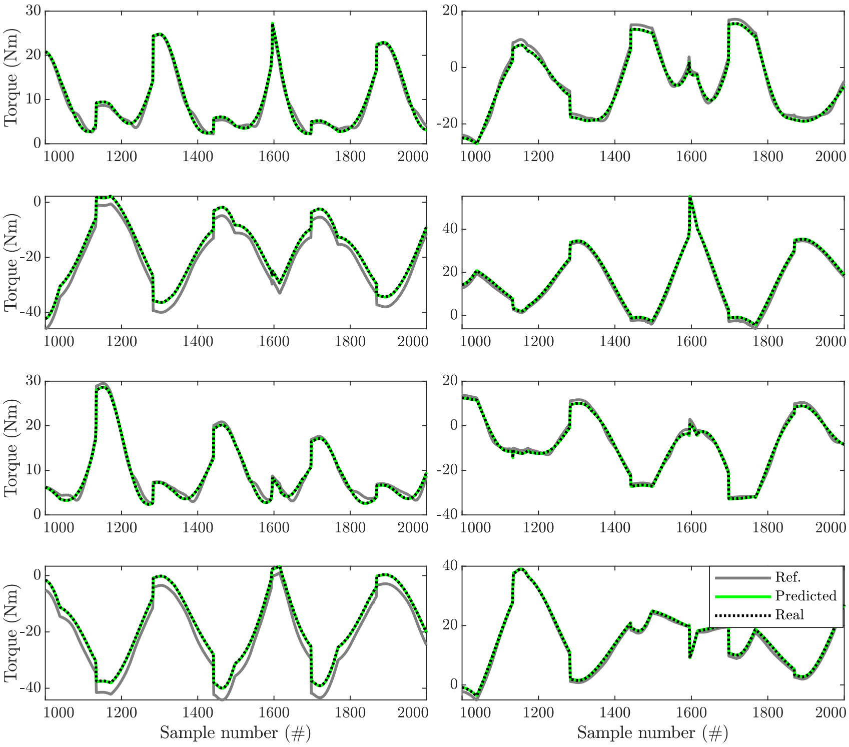


Figure 4.2: Realistic torque predictions.

the data, since the high-frequency vibrations are filtered out, which results in large differences between the ideal signals and the filtered signals.

Fig. 4.3 shows the ideal joint accelerations of two data sets: the training data used in the analysis of chapter 3, and data of the LLE ‘floating in space’, without the vibration in the legs. The spikes in Fig. 4.3a correspond to moments when the legs are vibrating.

Calculating the difference between the filtered and ideal joint accelerations, yields

$$\mathbf{e}_a = \ddot{\mathbf{q}}_{j,f} - \ddot{\mathbf{q}}_j, \tag{4.2}$$

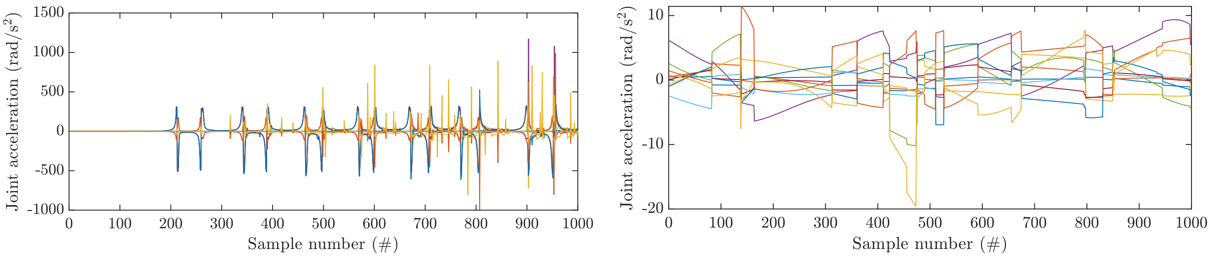
where $\ddot{\mathbf{q}}_{j,f}$ is the filtered joint acceleration and $\ddot{\mathbf{q}}_j$ the real joint acceleration.

Taking the RMSE of Eq. (4.2) for both signals in Fig. 4.3, results in an RMSE of 57 Nm and 5.0 Nm for the training and floating data, respectively, when filtered at 70 Hz. The large $\mathbf{e}_a$ for the training data set could decrease the estimation accuracy, because the EoM using the filtered data would be inconsistent.

Additionally, the legs have very limited movement, since the LLE is not yet able to perform movements standing on one leg. Since the goal of the LLE is to assist people with walking, it is expected that the identification procedure will have such movement data available in the future.

Because of these issues and those mentioned in chapter 3, the presented analysis will have different results than a real implementation in the LLE.

Finally, appendix B provides an explanation of why friction is mentioned in chapter 2, but was not taken into account in the analysis of chapter 3.



(a) With vibrating legs.

(b) Floating in space.

Figure 4.3: Ideal joint acceleration data.

Chapter 5

Conclusion

This thesis investigates if least-squares parameter estimation can be used for a floating-base, bipedal system. chapter 2 explains how the regressor matrix can be built so that the equation of motion can be expressed linearly in the inertial parameters. In chapter 3, multiple methods of solving this system are compared on their performance in general, and for implementation in the controller that is currently used.

The paper shows that appropriate scaling of the regressor matrix is essential for finding parameters that perform well on the evaluated metrics. A combination of scaling the rows with the measurement uncertainty, and the columns with a prior estimation, leads to the best results. However, without proper scaling, least-squares estimations do not generally lead to better performance on torque and CoM position prediction. Furthermore, performance relies partly on the accuracy of the prior estimations, which is assumed to be a percentage of the real values.

Therefore, this assumption was changed in additional analysis where instead of a percentage, an absolute error was added to the prior estimations. This showed that scaling only the rows might be better in this situation, but that scaling both the rows and columns might be a safe choice. Finally, it was found that predictions using noisy data also benefit from parameter estimations.

Appendix A

Statement of use of large language models

During the preparation of this work the author used ChatGPT 4 [14] in order to restructure text and provide linguistic suggestions. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work

Appendix B

Motivation for excluding friction in analysis

The analysis of chapter 3 excludes friction parameters. However, they are still mentioned in the thesis, because friction parameters can be hard to measure manually and cannot be obtained from CAD data. Furthermore, they could potentially change quickly over time, due to lubrication, dirt or wear. Therefore, it is important that they will be included in the real LLE. The following will explain why friction is excluded in chapter 3.

The regressor matrix was tested on correctness during its construction using the EoM error

$$\boldsymbol{\epsilon}_e = \mathbf{y}_e - \mathbf{W}_e \boldsymbol{\phi}, \tag{B.1}$$

where $\mathbf{y}_e$ and $\mathbf{W}_e$ are the noiseless versions of $\mathbf{y}$ and $\mathbf{W}$, respectively, and $\boldsymbol{\phi}$ the inertial parameters that are used in the simulation. This was done using simulation data from Simulink and Simscape, including joint friction as stated in Sec. 2.2.2 and GRFs. With that data it resulted in an EoM RMSE of order 10^{-14}. However, during preliminary testing, the EoM error using the Mujoco environment increased to order 10^{0}, and the presence of friction was found to be the largest source.

At that time the GRFs were directly saved from Mujoco as simulation output. In a later stage of the thesis, the GRFs were reconstructed from the regressor matrix and joint torques instead of saved from Mujoco, since the GRFs obtained from Mujoco produced large errors in the EoM as well.

After the analysis was finished, it appeared that reconstructing the GRFs using Eq. (4.1) decreased the EoM RMSE including friction to order 10^{-5}, which is still more than twice as large as without friction (RMSE of order $5 \cdot 10^{-6}$), but small enough for analysis.

While it would be interesting to perform additional analysis including friction, the software for the analysis does currently not allow for that.

Appendix C

Full table of results

Table C.1 shows all the results that were found during the comparison of chapter 3.

Table C.1: Results over 20 iterations with different prior estimates.

Prior estitmation accuracy	Solution Method	Filter Frequency (Hz)	Gen. force RMSE (N & Nm)	Torque RMSE (Nm)	CoM position RMSE (mm)	CoM mom. RMSE (Ns & Nms)	Avg. estimation error (%)	Avg. amount of used sing. val. (#)
±12.5%	Ref.	-	18.3	1.99	12.3	0.128	6.2	-
	OLS	0	31	18.4	173	0.628	21700	67
		30	8.29	**2.47**	**12.4**	0.113	**85.2**	6.85
		50	8.44	2.81	14.2	0.107	195	6.4
		70	5.94	2.51	16	0.096	153	9.45
		100	**5.34**	2.64	15.4	**0.0938**	203	9.55
	WLS-C	0	8.75	2.07	9.65	0.122	6.06	1.45
		30	7.81	2	9.33	0.102	6.38	6.15
		50	8.03	1.9	10.5	0.106	6.16	4.3
		70	3.89	**0.784**	**3.52**	0.0498	**5.87**	10.9
		100	**3.17**	1.73	6.73	**0.0446**	5.96	6.3
	WLS-R	0	15.3	6.62	39.7	0.307	2240	18.3
		30	10.7	**5.18**	14.7	0.101	**616**	19.6
		50	12.9	8.14	**13.7**	**0.099**5	1060	17.6
		70	10.6	6.58	16.3	0.103	1620	21.1
		100	**9.69**	5.27	16.2	0.109	1940	22
	WLS-RC	0	10.3	2.08	11.1	0.135	6.09	1.15
		30	**1.45**	1.1	**1.02**	**0.0221**	5.8	13.6
		50	1.83	1.09	1.72	0.0298	**5.74**	11.9
		70	2.16	**0.641**	3.01	0.044	5.82	18.5
		100	3.13	1.52	5.55	0.0458	5.95	6.65
±25%	Ref.	-	26	3.61	25.6	0.268	12.5	-
	OLS	0	31	18.4	173	0.624	22100	66.9
		30	11.2	4.58	28.9	0.191	**283**	13.5
		50	11.9	**4.11**	**20.4**	0.199	481	14.4
		70	**9.21**	5.02	27.7	**0.177**	392	10.8
		100	10	5.59	29	0.198	492	10.7
	WLS-C	0	13.1	3.23	16.2	0.182	12.4	2.1
		30	8.59	2.42	9.73	0.113	12.1	9.55
		50	9.25	2.65	11	0.122	12.1	9
		70	**4.23**	**1.03**	**3.28**	**0.0643**	**11.5**	12.4
		100	4.35	2.46	11.9	0.0724	11.8	8.7
	WLS-R	0	18.8	9.22	57.5	0.465	5810	36.5
		30	17.2	11.1	23.1	0.186	24800	37.3
		50	18.1	12.7	**18.5**	**0.168**	8460	29.4
		70	14.4	8.77	27	0.177	**3480**	27.6
		100	**13.5**	**7.91**	20.5	0.173	4640	35
	WLS-RC	0	15.9	3.66	26	0.217	12.6	2.4
		30	**1.85**	1.37	1.15	0.0335	11.4	15.2
		50	2.18	1.41	**0.996**	**0.0292**	11.3	17.6
		70	2.26	**0.712**	3.33	0.0444	**11.2**	19.3
		100	4.4	2.32	8.99	0.0733	11.7	12.1

Appendix D

Full table with estimations

As requested, a table with estimations. Note that these are specific for the training data and the (within a percentage of the real value) randomly generated used prior estimation

Table D.1: Estimations with prior estimation accuracy of ±25%, trained on data filtered at 50 Hz.

Parameter	Link	Real	OLS	WLS-R	WLS-C	WLS-RC
Mass	1	13.5	15	15	13.2	13.7
	2	4.35	3.85	3.9	4.7	4.11
	3	5.18	5.94	6.01	5.11	5.04
	4	3.5	3.29	3.29	3.92	3.82
	5	0.198	-0.0569	-0.0631	0.234	0.244
	6	0.547	0.211	0.204	0.53	0.557
	7	4.35	3.75	3.8	3.24	3.84
	8	5.18	4.94	5.01	5.66	5.62
	9	3.5	4.03	4.03	3.62	3.41
	10	0.198	-0.101	-0.103	0.189	0.183
	11	0.547	0.226	0.223	0.614	0.522
CoM (x)	1	-0.773	-0.863	-0.799	-0.842	-0.829
	2	0.36	0.355	0.347	0.294	0.347
	3	1.04	0.915	0.75	0.962	0.944
	4	0.39	0.371	0.362	0.357	0.342
	5	0.000492	-0.00208	-0.0249	0.000494	0.000495
	6	0.0268	0.0213	0.0361	0.0221	0.0222
	7	0.36	0.292	0.298	0.322	0.316
	8	1.04	1.08	0.924	1.07	1.04
	9	0.39	0.334	0.366	0.411	0.392
	10	0.000492	-0.00286	-0.0212	0.000403	0.000403
	11	0.0268	0.0293	0.0496	0.0303	0.0303
CoM (y)	1	0.0715	0.108	0.105	0.085	0.0884
	2	0.0594	0.00734	-0.0917	0.0639	0.0634
	3	0.0425	-0.0721	-0.0538	0.0392	0.0405
	4	-0.0299	-0.0329	0.0518	-0.0307	-0.0298
	5	-0.00409	0.00683	-0.00222	-0.00487	-0.0049
	6	-0.000725	0.0124	-0.000685	-0.000689	-0.00069
	7	0.0594	-0.00221	-0.115	0.0595	0.0597
	8	0.0425	-0.035	0.047	0.0489	0.0526
	9	-0.0299	-0.0122	0.0887	-0.0221	-0.0224
	10	-0.00409	0.0193	0.000175	-0.00339	-0.00343
	11	-0.000725	-0.0161	-0.00134	-0.000681	-0.000681
CoM (z)	1	4.94	5.91	5.76	5.1	4.9
	2	-0.328	-0.289	-0.348	-0.35	-0.355

	3	-0.0258	-0.047	-0.0533	-0.0295	-0.0294
	4	-0.0423	-0.0382	-0.0286	-0.0476	-0.048
	5	-0.00768	0.00721	0.0127	-0.00667	-0.00668
	6	0.0676	0.0493	0.0492	0.0586	0.0646
	7	0.328	0.187	0.249	0.247	0.248
	8	0.0258	0.00815	0.00153	0.0248	0.025
	9	0.0423	0.049	0.0486	0.0377	0.0375
	10	0.00768	0.0205	0.0252	0.00593	0.00593
	11	-0.0676	-0.0391	-0.0526	-0.0535	-0.0664
I_{xx}	1	2.29	2.7	2.71	2.19	2.18
	2	0.0808	0.155	0.209	0.0762	0.0758
	3	0.0155	0.0185	0.0167	0.0186	0.0186
	4	0.00868	0.00795	0.00584	0.00795	0.00795
	5	0.00174	0.0019	0.000392	0.00201	0.00201
	6	0.0131	0.00999	0.01	0.01	0.01
	7	0.0808	0.165	0.218	0.0862	0.0857
	8	0.0155	0.0192	0.0178	0.0192	0.0192
	9	0.00868	0.00861	0.00611	0.00854	0.00854
	10	0.00174	0.002	0.000869	0.00202	0.00202
	11	0.0131	0.0151	0.0153	0.0152	0.0152
I_{xy}	1	0.00621	0.00149	-0.0183	0.00663	0.00664
	2	-0.00483	-0.0115	-0.0143	-0.00374	-0.00374
	3	-0.00747	-0.00699	-0.0058	-0.00679	-0.00679
	4	0.00161	0.00217	0.00204	0.00192	0.00192
	5	3.97e-05	0.000286	-0.00187	4.55e-05	4.55e-05
	6	5.17e-05	-4.71e-05	-0.0223	3.92e-05	3.92e-05
	7	-0.00483	0.00386	0.00711	-0.00478	-0.00478
	8	-0.00747	-0.00812	-0.00548	-0.00802	-0.00802
	9	0.00161	0.00165	0.000178	0.00145	0.00145
	10	3.97e-05	0.000186	-0.00262	3.49e-05	3.49e-05
	11	5.17e-05	-0.00134	0.0169	4.72e-05	4.72e-05
I_{xz}	1	0.312	0.307	0.312	0.314	0.318
	2	0.0545	0.0559	0.0635	0.0656	0.0625
	3	0.0175	0.00388	0.0119	0.0213	0.0214
	4	0.00258	-0.00477	0.00437	0.00297	0.00297
	5	8.46e-06	-0.000561	-0.000112	9.37e-06	9.37e-06
	6	-0.00333	-0.00375	-0.00284	-0.00371	-0.00371
	7	-0.0545	-0.0551	-0.0613	-0.0605	-0.0576
	8	-0.0175	-0.0273	-0.0443	-0.0156	-0.0155
	9	-0.00258	-0.00851	-0.0194	-0.00261	-0.00261
	10	-8.46e-06	0.000914	-0.0036	-6.88e-06	-6.88e-06
	11	0.00333	0.00463	0.0125	0.00414	0.00414
I_{yy}	1	2.31	1.95	2.01	2.52	2.44
	2	0.128	0.127	0.125	0.127	0.127
	3	0.326	0.385	0.386	0.387	0.385
	4	0.0834	0.0762	0.0775	0.0764	0.0763
	5	0.000446	0.00052	0.00119	0.000489	0.000489
	6	0.0145	0.0151	0.0937	0.0169	0.0169
	7	0.128	0.105	0.103	0.105	0.105
	8	0.326	0.325	0.325	0.327	0.328
	9	0.0834	0.0652	0.0657	0.0652	0.0652
	10	0.000446	0.000504	-0.000405	0.000368	0.000368
	11	0.0145	0.00969	0.0576	0.0123	0.0123
I_{yz}	1	-0.029	-0.00516	0.0199	-0.0356	-0.0357
	2	0.00249	0.00178	0.00292	0.00202	0.00202

	3	0.00328	-0.0132	-0.00433	0.00282	0.00281
	4	0.00137	0.0129	0.0116	0.00115	0.00115
	5	-4.94e-05	0.000356	0.00302	-6e-05	-6e-05
	6	9.73e-05	-0.000393	-0.0481	0.000121	0.000121
	7	-0.00249	-0.0023	-0.00498	-0.00245	-0.00245
	8	-0.00328	0.00522	0.00324	-0.00269	-0.00268
	9	-0.00137	-0.00475	-0.00403	-0.00159	-0.00159
	10	4.94e-05	-0.00139	-0.00855	6.11e-05	6.11e-05
	11	-9.73e-05	0.00371	-0.0481	-0.00012	-0.00012
I_{zz}	1	0.0951	0.103	0.0975	0.0991	0.0985
	2	0.0587	0.0694	0.0707	0.0695	0.0695
	3	0.323	0.232	0.215	0.287	0.293
	4	0.0822	0.0578	0.0605	0.0622	0.0683
	5	0.00132	-0.000146	0.08	0.00159	0.00159
	6	0.00229	0.0024	0.00174	0.00242	0.00242
	7	0.0587	0.0549	0.055	0.0549	0.0549
	8	0.323	0.205	0.214	0.361	0.357
	9	0.0822	0.141	0.033	0.0658	0.064
	10	0.00132	-0.00169	0.0475	0.001	0.001
	11	0.00229	0.00267	0.00123	0.0022	0.0022

Bibliography

[1] A. Vallinas, A. Keemink, C. Bayón, E. van Asseldonk, and H. van der Kooij, "Momentum-Based Balance Control of a Lower-Limb Exoskeleton During Stance," in *2023 International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–6, IEEE, 9 2023.

[2] C. Meijneke, G. Van Oort, V. Sluiter, E. Van Asseldonk, N. L. Tagliamonte, F. Tamburella, I. Pisotta, M. Masciullo, M. Arquilla, M. Molinari, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and H. Van Der Kooij, "Symbitron Exoskeleton: Design, Control, and Evaluation of a Modular Exoskeleton for Incomplete and Complete Spinal Cord Injured Individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 330–339, 2021.

[3] R. B. van Dijsseldonk, I. J. W. van Nes, A. C. H. Geurts, and N. L. W. Keijsers, "Exoskeleton home and community use in people with complete spinal cord injury," *Scientific Reports*, vol. 10, p. 15600, 9 2020.

[4] F. Tamburella, N. L. Tagliamonte, I. Pisotta, M. Masciullo, M. Arquilla, E. H. F. van Asseldonk, H. van der Kooij, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and M. Molinari, "Neuromuscular Controller Embedded in a Powered Ankle Exoskeleton: Effects on Gait, Clinical Features and Subjective Perspective of Incomplete Spinal Cord Injured Subjects," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 5, pp. 1157–1167, 2020.

[5] C. Baunsgaard, U. Nissen, A. Brust, A. Frotzler, C. Ribeill, Y. Kalke, N. León, B. Gómez, K. Samuelsson, W. Antepohl, U. Holmström, N. Marklund, T. Glott, A. Opheim, J. Penalva, N. Murillo, J. Nachtegaal, W. Faber, and F. Biering-Sørensen, "Exoskeleton gait training after spinal cord injury: An exploratory study on secondary health conditions," *Journal of Rehabilitation Medicine*, vol. 50, no. 9, pp. 806–813, 2018.

[6] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. van der Kooij, "Design and Control of the MINDWALKER Exoskeleton," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, pp. 277–286, 3 2015.

[7] T. Gurriet, S. Finet, G. Boeris, A. Duburcq, A. Hereid, O. Harib, M. Masselin, J. Grizzle, and A. D. Ames, "Towards Restoring Locomotion for Paraplegics: Realizing Dynamically Stable Walking on Exoskeletons," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2804–2811, IEEE, 5 2018.

[8] Q. Chen, H. Cheng, C. Yue, R. Huang, and H. Guo, "Dynamic Balance Gait for Walking Assistance Exoskeleton," *Applied Bionics and Biomechanics*, vol. 2018, pp. 1–10, 7 2018.

[9] R. Featherstone, "Dynamics of Rigid Body Systems," in *Rigid Body Dynamics Algorithms*, pp. 39–64, Boston, MA: Springer US, 2008.

[10] C. H. An, C. G. Atkeson, and J. M. Hollerbach, "Estimation of Inertial Parameters of Rigid Body Links of Manipulators," *Proceedings of the IEEE Conference on Decision and Control*, pp. 990–995, 1985.

[11] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, "Inertial Parameter Identification in Robotics: A Survey," *Applied Sciences*, vol. 11, p. 4303, 5 2021.

[12] J. Jovic, A. Escande, K. Ayusawa, E. Yoshida, A. Kheddar, and G. Venture, “Humanoid and Human Inertia Parameter Identification Using Hierarchical Optimization,” *IEEE Transactions on Robotics*, vol. 32, pp. 726–735, 6 2016.

[13] J. Hollerbach, W. Khalil, and M. Gautier, “Model Identification,” *Springer Handbook of Robotics*, pp. 321–344, 2008.

[14] OpenAI, “ChatGPT,” *Available at: https://chat.openai.com/.*