TECHNISCHE UNIVERSITÄT WIEN

AUTOMATION & CONTROL INSTITUTE
INSTITUT FÜR AUTOMATISIERUNGS- & REGELUNGSTECHNIK

Identification of the dynamic parameters of the KUKA LBR iiwa R820

BACHELOR THESIS

Conducted in partial fulfillment of the requirements for the degree of a

Bachelor of Science (BSc)

supervised by

Univ.Ass.in Lucia Moser-Lauxmann, MSc
Dr.techn. Minh Nhat Vu
Associate Prof. Dipl.-Ing. Dr.-Ing. Wolfgang Kemmetmüller

submitted at the

TU Wien
Faculty of Electrical Engineering and Information Technology
Automation and Control Institute

by

Simon Halper
Matriculation number 12122101

Vienna, May 2024

Complex Dynamical Systems Group
A-1040 Wien, Gußhausstr. 27–29, Internet: https://www.acin.tuwien.ac.at

Abstract

The dynamic parameters of a robot are a set of parameters that allow the simulation and control of a robot's motion. Although such a set of parameters is crucial for working with any robot, it is not publicly available for the KUKA LBR iiwa R820. Therefore, the dynamic parameters of the KUKA LBR iiwa R820 are estimated in this work. This is done on the basis of the methods proposed in [1] using a nonlinear optimization solver. The results of the parameter identification are verified by simulating the KUKA LBR iiwa R820 with the identified parameters and comparing the simulation results with the robot's measurements.

Kurzzusammenfassung

Die dynamischen Parameter eines Roboters sind eine Menge von Parametern, welche die Simulation und Steuerung der Bewegung eines Roboters ermöglichen. Obwohl ein solcher Satz von Parametern für die Arbeit mit Robotern entscheidend ist, ist er für den KUKA LBR iiwa R820 nicht öffentlich verfügbar. Daher werden die dynamischen Parameter des KUKA LBR iiwa R820 in dieser Arbeit geschätzt. Dies geschieht auf der Grundlage der in [1] vorgeschlagenen Methoden mit Hilfe eines nichtlinearen Optimierungslösers. Die Ergebnisse der Parameteridentifikation werden verifiziert indem der KUKA LBR iiwa R820 mit den identifizierten Parametern simuliert wird und die Simulationsergebnisse mit den Messungen des Roboters verglichen werden.

Contents

List of Figures

List of Tables

1 Introduction

Robots are programmable multi-functional mechanical systems with several degrees of freedom that perform numerous tasks ranging from manipulating objects to autonomous movement to working with humans. A prerequisite for all these tasks to be performed safely and accurately is a profound knowledge of the dynamic behavior of the robot. This knowledge is needed for critical algorithms like simulations, controllers, or collision detection to deliver accurate results.

The dynamic parameters are a set of parameters crucial for computing a robot's dynamic behavior. They can be seen as the physical characteristics of the individual parts of the robot and are usually not published by the robot manufacturers. Furthermore, these parameters must be calibrated periodically depending on several conditions, because of changes in the internal friction of the robot or because the robot uses a different tool. Thus, a framework is needed to update these parameters. There are several methods to compute dynamic parameters, including modeling the individual parts with CAD software or dismantling the robot and measuring the physical properties of the individual parts directly [2]. A more frequently used method is to estimate the dynamical parameters from measurements by exploiting certain properties of the general equations of motion for mechanical systems [1]. This thesis is mainly inspired by [1].

1.1 Related Work

The parameter identification problem for robotic systems is well-studied. For example, Khalil et al. [3] offer a comprehensive overview of the mathematical tools required for parameter identification. Specifically, they describe how a reduced set of dynamical parameters can be identified independently. Furthermore, they provide a broad discussion of how to find a good excitation trajectory, which is crucial for obtaining meaningful results.

For identifying the parameters of the KUKA LBR iiwa R820 specifically, the first framework has been proposed in [1]. There, they define the forward kinematics of the KUKA LBR iiwa R820 in the Modified Denavit-Hartenberg (MDH) convention and, based on that, introduce a reduced set of dynamical parameters that include static and viscous friction terms. They estimate the reduced set of dynamic parameters using the least squares method and independently estimate the physical parameters using local optimization by considering nonlinear constraints. Furthermore, they generate an optimal excitation trajectory based on a finite Fourier series parameterization.

Based on [1], [4] proposes to further reduce the dynamic parameters into the essential parameters by removing poorly identified parameters. Moreover, they propose a different

algorithm for estimating the physical parameters of the robot. Their algorithm contains a stochastic component that randomly selects the starting value for the optimization problem of finding the physical parameters. Additionally, they define the optimization problem in terms of the essential parameters of the KUKA LBR iiwa R820 and solve it with a global optimization solver.

1.2 Contribution and Structure of the thesis

The main contribution of this thesis is to adapt the methods for parameter identification proposed in [1] for the KUKA LBR iiwa R820 at the Automation and Control Institute at TU Wien. Furthermore, excitation trajectories are employed in the real setup to verify the identified parameters.

The thesis is structured as follows. Chapter 2 explains the theoretical background needed for the main part. The main part, Chapter 3, explains all relevant considerations that lead to the parameter identification. Finally, in Chapter 4, the identified parameters are presented, discussed, and verified via accurate measurements.

2 Theoretical Background

2.1 Lagrangian Formulation

The Lagrangian formulation defines a systematic way to derive the equations of motion of any mechanical system independently of the reference coordinate frame [2]. The equations of motion describe how the motion of the system and the torques or forces at the system's actuators relate to each other. The Lagrangian formulation acts on generalized coordinates $\mathbf{q}$, representing the system's degrees of freedom. Therefore, the system already has to be defined in terms of $\mathbf{q}$ for the Lagrangian formulation to be applicable.

First of all, the Lagrangian formulation defines the *Lagrangian* L of the system

$$L = T - U \,, \tag{2.1}$$

which is a function of the generalized coordinates $\mathbf{q}$ [2]. T and U denote the system's total kinetic and potential energy, respectively.

Then, the Lagrange equations are

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right)^{\mathrm{T}} - \left(\frac{\partial L}{\partial \mathbf{q}}\right)^{\mathrm{T}} = \boldsymbol{\tau} \,, \tag{2.2}$$

where $\boldsymbol{\tau}$ is the vector of generalized forces corresponding to the generalized coordinates $\mathbf{q}$. $\boldsymbol{\tau}$ contains all non-conservative forces in the system, i. e. the actuator torques, friction, or forces induced by the contact with the environment. [2]

2.2 Rigid Link Robot Modeling

To properly understand the parameter identification problem, one has to know how robots with rigid links are modeled mathematically. This Section will cover the fundamentals of mathematically describing the state and the motion of any rigid link robot. Furthermore, the robot's parameters will be introduced, and their influence on the model will be outlined.

2.2.1 Pose of a rigid body

Rigid link robots are robots that only consist of rigid bodies. Rigid bodies have the defining property that the distance between two arbitrary points on that body stays the same, even if forces act on that body. For example, they do not deform under the presence

of external forces [2].

The position of a rigid body is completely described by its *position* and *orientation* in reference to a reference frame $(\mathbf{0}, \mathbf{x}, \mathbf{y}, \mathbf{z})$. The vectors comprising the reference frame are elements of $\mathbb{R}^3$. $\mathbf{0}$ specifies the origin of the frame and $\mathbf{x}, \mathbf{y}$ and $\mathbf{z}$ are the orthogonal unit vectors of the frame axis. The position of a point $\mathbf{p}$ can then be expressed in relation to the origin $\mathbf{0}$ as

$$\mathbf{p} = p_x\mathbf{x} + p_y\mathbf{y} + p_z\mathbf{z}\,,$$

where p_x, p_y and p_z are the coordinates of $\mathbf{p}$ [2]. Therefore, $\mathbf{p}$ can be written conveniently as

$$\mathbf{p} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}.$$

To describe a body's orientation, a reference frame $(\mathbf{0}_b, \mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b)$ attached to the body b is considered, as Figure 2.1 illustrates. The values of the vectors $\mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b$ with respect to the inertial frame $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ represent the orientation of the body b with respect to the frame $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ [2].

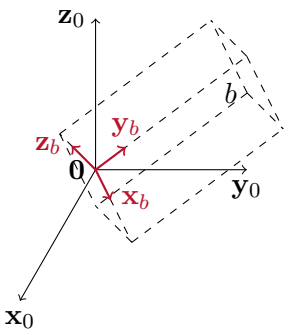

Figure 2.1: Rigid body's orientation

2.2.2 Rotation Matrix

The three vectors, describing the orientation of body b in Figure 2.1, can be combined into matrix $\mathbf{R}_0^b \in \mathbb{R}^{3\times3}$ like

$$\mathbf{R}_0^b = \begin{bmatrix} \mathbf{x}_b & \mathbf{y}_b & \mathbf{z}_b \end{bmatrix} = \begin{bmatrix} x_{bx} & y_{bx} & z_{bx} \\ x_{by} & y_{by} & z_{by} \\ x_{bz} & y_{bz} & z_{bz} \end{bmatrix}, \tag{2.3}$$

where $\mathbf{R}_0^b$ is called *rotation matrix* [2]. The notation $\mathbf{R}_0^b$ means that $\mathbf{R}$ transforms points defined in reference frame $(\mathbf{0}_b, \mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b)$ into the reference frame $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$. Note that because $\mathbf{x}_b, \mathbf{y}_b$ and $\mathbf{z}_b$ are required to be orthogonal unit vectors, $\mathbf{R}$ is an orthogonal matrix. This means that its transpose $\mathbf{R}^{\mathrm{T}}$ is equivalent to the inverse matrix $\mathbf{R}^{-1}$.

Therefore, if the rotation matrix is dependent on a rotation angle θ, its derivative $\dot{\mathbf{R}} = \frac{\mathrm{d}\mathbf{R}(\theta)}{\mathrm{d}\theta}$ must adhere to the following identity [2]

$$\dot{\mathbf{R}}\mathbf{R}^{\mathrm{T}} = -\mathbf{R}\dot{\mathbf{R}}^{\mathrm{T}} \,. \tag{2.4}$$

This shows that $\dot{\mathbf{R}}\mathbf{R}^{\mathrm{T}}$ must be skew-symmetric. Any skew-symmetric matrix $\mathbf{S}$ can be represented by a single vector. In the case of rotation matrices, this vector is called the *vector of angular velocities* and is denoted $\boldsymbol{\omega} = [\omega_x \; \omega_y \; \omega_z]$. $\dot{\mathbf{R}}\mathbf{R}^{\mathrm{T}}$ can then be reconstructed by the formula [2]

$$\dot{\mathbf{R}}\mathbf{R}^{\mathrm{T}} = \mathbf{S}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} . \tag{2.5}$$

There are several methods to construct rotation matrices [2]. In general, it is easier to separate the rotation along a general axis into three consecutive individual rotations around the axis $(\mathbf{x}, \mathbf{y}, \mathbf{z})$. These individual rotations can then be combined mathematically by the matrix multiplication operation. The matrices for rotating around one axis are

$$\begin{aligned} \mathbf{R}_x(\theta) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\theta & -s_\theta \\ 0 & s_\theta & c_\theta \end{bmatrix} , \\ \mathbf{R}_y(\theta) &= \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} , \\ \mathbf{R}_z(\theta) &= \begin{bmatrix} c_\theta & -s_\theta & 0 \\ s_\theta & c_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} , \end{aligned} \tag{2.6}$$

where s_θ and c_θ are shorthand for $\sin(\theta)$ and $\cos(\theta)$ respectively, see [2].

2.2.3 Translation

Besides rotation, translation is another way of transforming a point in space from one frame of reference into another. As depicted in Figure 2.2, translation corresponds to changing the origin of the frame without rotating the frame [2].

Mathematically, translation is described as

$$\mathbf{p}_0 = \mathbf{p}_1 + \mathbf{d}_0^1 \,, \tag{2.7}$$

where $\mathbf{p}_0$ and $\mathbf{p}_1$ is the same arbitrary point, but defined with respect to frame $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ and $(\mathbf{0}_1, \mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1)$ respectively [2].

2.2.4 Homogeneous Transformations

If two reference frames $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ and $(\mathbf{0}_1, \mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1)$ are rotated and translated with respect to each other, the conversion of an arbitrary point $\mathbf{p}_1$ to $\mathbf{p}_0$ is given by

$$\mathbf{p}_0 = \mathbf{R}_0^1 \mathbf{p}_1 + \mathbf{d}_0^1 \,. \tag{2.8}$$

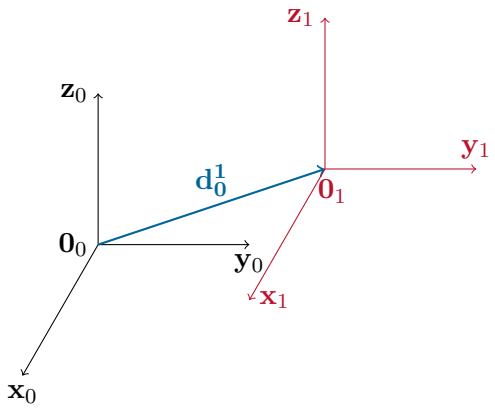

Figure 2.2: Translation of a frame of reference

$\mathbf{P}_0$ is defined as $\mathbf{p}_0$ extended by an additional homogeneous coordinate as

$$\mathbf{P}_0 = \begin{bmatrix} \mathbf{p}_0 \\ 1 \end{bmatrix} . \tag{2.9}$$

Then (2.8) can be rewritten as a single matrix-vector multiplication of $\mathbf{P}_1$ with the *homogeneous transformation* [2]

$$\mathbf{H}_0^1 = \begin{bmatrix} \mathbf{R}_0^1 & \mathbf{d}_0^1 \\ \mathbf{0} & 1 \end{bmatrix} \tag{2.10}$$

as

$$\mathbf{P}_0 = \mathbf{H}_0^1 \mathbf{P}_1 . \tag{2.11}$$

This allows for a more convenient transformation between reference frames, especially when several reference frames are involved because the product of two homogeneous transformations is again a homogeneous transformation. Additionally, the identity

$$\mathbf{H}_1^0 = (\mathbf{H}_0^1)^{-1} \tag{2.12}$$

also holds, which allows for a systematic calculation of the inverse transformation [2].

2.2.5 Robots as Kinematic Chains

In general, any rigid link robot consists of links and joints. Links are the rigid parts of the robot that are connected by joints. Joints perform actions and are responsible for the motion of the robot. A joint can be either revolute or prismatic, creating rotational or translational motion between its two connected links. This structure of links connected by joints is called a *kinematic chain* [2]. A kinematic chain can be opened or closed, depending on whether the structure has a loop. In this thesis, only open kinematic chains are considered. Any joint in an open kinematic chain adds a single degree of freedom to the robot, represented by the variable q_j, where j is the index of

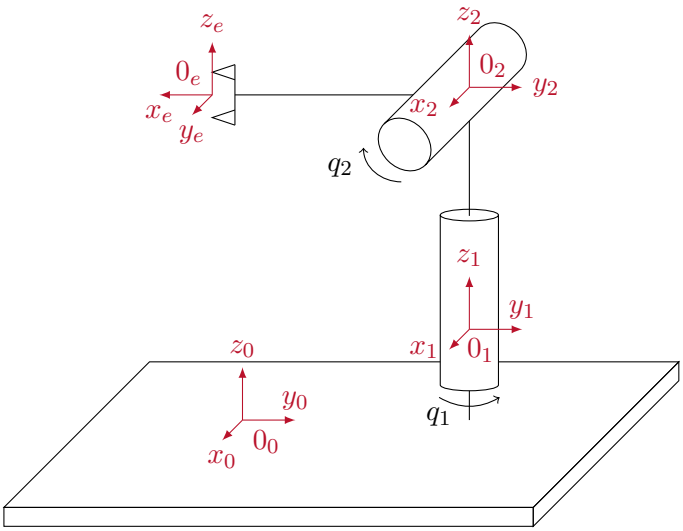

Figure 2.3: Schematic of a 2-DoF robot

the joint. The state of all joints is denoted by the vector of the *generalized coordinates* $\mathbf{q}$ [2].

Consider the robot depicted in Figure 2.3. It consists of two revolute joints and three links, as indicated by q_1 and q_2. Furthermore, several reference frames are already attached. The first reference frame $(\mathbf{0}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ is called the *base frame* and it is assumed to be inertial [2]. The last part of the robot is called *endeffector*, which is why its frame $(\mathbf{0}_e, \mathbf{x}_e, \mathbf{y}_e, \mathbf{z}_e)$ is denoted by a subscript e. Additionally, each joint also has a corresponding reference frame.

Given homogeneous transformations, the position of the end effector with respect to the base frame can be calculated by following the kinematic chain from the end effector to the base in the following

$$\mathbf{H}_0^e = \mathbf{H}_0^1(\mathbf{q})\mathbf{H}_1^2(\mathbf{q})\mathbf{H}_2^e\,. \tag{2.13}$$

Note that $\mathbf{H}_0^1(\mathbf{q})$ and $\mathbf{H}_1^2(\mathbf{q})$ depend on $\mathbf{q}$. This calculation is called the *forward kinematics* of the robot [2]. The space of all possible joint positions is called the *joint space*, and the space of all possible end-effector poses is called the *operational space*. Note that the joint space can have any number of dimensions, but the operational space is always a subset of $\mathbb{R}^6$, see, e.g., [5], [2].

As shown in Section 2.2.4, homogeneous transformations can be split into a rotational and a translational part as

$$\mathbf{H}_k^l(\mathbf{q}) = \begin{bmatrix} \mathbf{R}_k^l(\mathbf{q}) & \mathbf{d}_k^l(\mathbf{q}) \\ \mathbf{0} & 1 \end{bmatrix}. \tag{2.14}$$

Considering the derivatives of $\mathbf{R}_k^l(\mathbf{q})$ and $\mathbf{d}_k^l(\mathbf{q})$ with respect to $\mathbf{q}$, it can be shown that

the vector of angular velocities $\boldsymbol{\omega}_k^l$ of $\dot{\mathbf{R}}_k^l(\mathbf{R}_k^l)^{\mathrm{T}}$ can be represented as

$$\boldsymbol{\omega}_k^l = (\mathbf{J}_{\boldsymbol{\omega}})_k^l(\mathbf{q})\dot{\mathbf{q}}\,. \tag{2.15}$$

Furthermore,

$$\dot{\mathbf{d}}_k^l(\mathbf{q}) = \frac{\partial}{\partial \mathbf{q}}\mathbf{d}_k^l(\mathbf{q}) = (\mathbf{J}_{\mathbf{v}})_k^l(\mathbf{q})\dot{\mathbf{q}}\,. \tag{2.16}$$

This means that the vector of linear velocities $\dot{\mathbf{d}}_k^l$ and the vector of angular velocities $\boldsymbol{\omega}_k^l$ are related to the derivative of the generalized coordinates $\dot{\mathbf{q}}$ by its corresponding $(3 \times n)$ Jacobi matrix $(\mathbf{J}_{\mathbf{v}})_k^l$ and $(\mathbf{J}_{\boldsymbol{\omega}})_k^l$, respectively [2].

2.2.6 Equations of Motion

As described in Section 2.1, a robot's equations of motion can be obtained by applying the Lagrangian formulation to to robot. Because the kinetic and potential energy of a robot are dependent on the joint's position and velocities $\mathbf{q}$ and $\dot{\mathbf{q}}$, the Lagrangian of the system can be written as

$$L(\mathbf{q}, \dot{\mathbf{q}}) = T(\mathbf{q}, \dot{\mathbf{q}}) - U(\mathbf{q})\,. \tag{2.17}$$

It turns out that the kinetic energy $T(\mathbf{q}, \dot{\mathbf{q}})$ can be calculated as follows [2]

$$T(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}\,. \tag{2.18}$$

The $(n \times n)$ matrix $\mathbf{M}$, where n is the number of generalized coordinates in the system, is called the *inertia matrix*. In the case that the robot is modeled as a kinematic chain, it can be broken down into: [2]

$$\mathbf{M}(\mathbf{q}) = \sum_{i=1}^{n}\Big(m_i\Big((\mathbf{J}_{\mathbf{v}})_0^i\Big)^{\mathrm{T}}(\mathbf{J}_{\mathbf{v}})_0^i + \Big((\mathbf{J}_{\boldsymbol{\omega}})_0^i\Big)^{\mathrm{T}}\mathbf{R}_0^i\mathbf{I}_i^i\Big(\mathbf{R}_0^i\Big)^{\mathrm{T}}(\mathbf{J}_{\boldsymbol{\omega}})_0^i\Big)\,. \tag{2.19}$$

Here, m_i refers to the mass of link i, and $\mathbf{R}_0^i$ refers to the rotation matrix from frame i to the base frame 0. Furthermore, $(\mathbf{J}_{\mathbf{v}})_0^i$ and $(\mathbf{J}_{\boldsymbol{\omega}})_0^i$ are the Jacobian matrices relating the contribution of the joint velocities $\dot{\mathbf{q}}$ to the linear velocity of the center of gravity and the angular velocity of link i, respectively. Finally, $\mathbf{I}_i^i$ is the *inertia tensor* of link i with respect to the center of gravity of link i. It is defined as [5]

$$\begin{aligned}\mathbf{I}_i^i &= \begin{bmatrix} I_{i,xx}^i & I_{i,xy}^i & I_{i,xz}^i \\ I_{i,xy}^i & I_{i,yy}^i & I_{i,yz}^i \\ I_{i,xz}^i & I_{i,yz}^i & I_{i,zz}^i \end{bmatrix} \\ &= \begin{bmatrix} \int_{V_i} p_{i,y}^2 + p_{i,z}^2\,\mathrm{d}m & -\int_{V_i} p_{i,x}p_{i,y}\,\mathrm{d}m & -\int_{V_i} p_{i,x}p_{i,z}\,\mathrm{d}m \\ -\int_{V_i} p_{i,x}p_{i,y}\,\mathrm{d}m & \int_{V_i} p_{i,x}^2 + p_{i,z}^2\,\mathrm{d}m & -\int_{V_i} p_{i,y}p_{i,z}\,\mathrm{d}m \\ -\int_{V_i} p_{i,x}p_{i,z}\,\mathrm{d}m & -\int_{V_i} p_{i,y}p_{i,z}\,\mathrm{d}m & \int_{V_i} p_{i,x}^2 + p_{i,y}^2\,\mathrm{d}m \end{bmatrix},\end{aligned} \tag{2.20}$$

where V_i denotes the volume of link i and $\mathbf{p}_i$ is the distance to the center of gravity of link i. Note that this definition implies that $\mathbf{I}_i^i$ is symmetric and that the triangle inequality

$I^i_{i,xx} + I^i_{i,yy} \geq I^i_{i,zz}$ holds for each permutation of the diagonal elements [2], [5]. Moreover, it can be shown that $\mathbf{M}$ has special properties, e.g., symmetric and positive definite [2].

The function $U(\mathbf{q})$ of (2.17) represents the potential energy, which for the setting considered in this thesis simplifies to the contribution of gravity. Therefore,

$$U(\mathbf{q}) = \sum_{i=1}^{n} m_i (\mathbf{g}_0)^{\mathrm{T}} \mathbf{p}_0^{c_i} \,, \tag{2.21}$$

where $\mathbf{p}_0^{c_i}$ denotes the center of gravity of link i and $\mathbf{g}_0$ denotes the direction and magnitude of the gravitational force, both with respect to the base frame [2].

Applying the Lagrangian equation (2.2) to (2.17) yields the dynamic model

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

where $\ddot{\mathbf{q}}$ is the vector of the joints accelerations, the matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ contains centrifugal and Coriolis forces in the system. The vector $\mathbf{g}$ is derivative of U with respect to $\mathbf{q}$, see, e.g., [2], expressed in the form

$$\mathbf{g}(\mathbf{q}) = \frac{\partial U(\mathbf{q})}{\partial \mathbf{q}} \,.$$

2.2.7 Dynamic Parameters

Note that although the formulation of the dynamic model in (2.22) is widely used, it is not the only way to estimate the dynamic model of the robot. More importantly, it is possible to rearrange (2.22) such that it is linear with respect to dynamic parameters that are not dependent on the robot's configuration $\mathbf{q}, \dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. Formally,

$$\boldsymbol{\tau} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi} \,, \tag{2.23}$$

where $\mathbf{Y}$ is called the regressor matrix and $\boldsymbol{\pi}$ is the vector of the dynamic parameters [2], [1].

The derivation of the dynamic parameters requires denoting each variable, its frame of reference, and the link to which it is attributed. Therefore, to avoid confusing the two indices and to stay consistent with the already established homogeneous transformations, the rest of this section will follow the subsequent notation. The corresponding link is denoted in superscript for each variable that has to be expressed with respect to a frame of reference. In contrast, the frame of reference is denoted in subscript. For other variables, e. g. mass or energy, the link is denoted in the subscript [2].

To calculate the dynamic parameters $\boldsymbol{\pi}$, consider the energy T_i of any link i with respect to the base frame. By combining (2.18) and (2.19), T_i becomes [2]

$$T_i = \frac{1}{2}(m_i(\dot{\mathbf{p}}_0^{c_i})^{\mathrm{T}} \dot{\mathbf{p}}_0^{c_i} + (\boldsymbol{\omega}_i^i)^{\mathrm{T}} \mathbf{I}_i^i \boldsymbol{\omega}_i^i) \,. \tag{2.24}$$

Here, the identities

$$\dot{\mathbf{p}}_0^{c_i} = (\mathbf{J_v})_0^i \dot{\mathbf{q}}, \qquad \boldsymbol{\omega}_i^i = \left(\mathbf{R}_0^i\right)^{\mathrm{T}} (\mathbf{J}_\omega)_0^i \dot{\mathbf{q}}$$

have been used. Note that because the robot is assumed to be rigid, the distance between any two points defined in frame i will always stay constant. Therefore, $\mathbf{p}_{c_i}$ can be substituted by

$$\mathbf{p}_0^{c_i} = \mathbf{d}_0^i + \mathbf{R}_0^i \mathbf{r}_i^{c_i} \,, \tag{2.25}$$

where $\mathbf{d}_0^i$ is the vector from the origin of the base frame to the origin of frame i, with respect to the base frame, and $\mathbf{R}_0^i$ is the orientation of frame i with respect to the base frame. Additionally, $\mathbf{r}_i^{c_i}$ is the position of the center of mass of link i in frame i, which is constant. Therefore, by differentiating (2.25), it becomes [2]

$$\dot{\mathbf{p}}_0^{c_i} = \dot{\mathbf{d}}_0^i + \boldsymbol{\omega}_0^i \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right) . \tag{2.26}$$

This transforms (2.24) into

$$\begin{aligned} T_i = \frac{1}{2} (m_i (\dot{\mathbf{d}}_0^i)^{\mathrm{T}} \dot{\mathbf{d}}_0^i + 2 (\dot{\mathbf{d}}_0^i)^{\mathrm{T}} m_i \Big(\boldsymbol{\omega}_0^i \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right)\Big) \\ + m_i \Big(\boldsymbol{\omega}_0^i \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right)\Big)^{\mathrm{T}} \Big(\boldsymbol{\omega}_0^i \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right)\Big) + (\boldsymbol{\omega}_i^i)^{\mathrm{T}} \mathbf{I}_i^i \boldsymbol{\omega}_i^i) \,. \end{aligned} \tag{2.27}$$

Because $\mathbf{R}_0^i$ is a rotation matrix, $\boldsymbol{\omega}_0^i \times (\mathbf{R}_0^i \mathbf{r}_i^{c_i})$ can be transformed as follows

$$\boldsymbol{\omega}_0^i \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right) = \left(\mathbf{R}_0^i \left(\mathbf{R}_0^i\right)^{\mathrm{T}} \boldsymbol{\omega}_0^i\right) \times \left(\mathbf{R}_0^i \mathbf{r}_i^{c_i}\right) = \mathbf{R}_0^i \left(\boldsymbol{\omega}_i^i \times \mathbf{r}_i^{c_i}\right) .$$

Furthermore, by using the facts that

$$\left(\boldsymbol{\omega}_i^i \times \mathbf{r}_i^{c_i}\right) = \mathbf{S}(\boldsymbol{\omega}_i^i) \mathbf{r}_i^{c_i} = -\mathbf{S}(\mathbf{r}_i^{c_i}) \boldsymbol{\omega}_i^i \,.$$

and

$$\left(\mathbf{R}_0^i\right)^{\mathrm{T}} \dot{\mathbf{d}}_0^i = \dot{\mathbf{d}}_i^i$$

(2.27) becomes

$$T_i = \frac{1}{2} (m_i (\dot{\mathbf{d}}_i^i)^{\mathrm{T}} \dot{\mathbf{d}}_i^i + 2 (\dot{\mathbf{d}}_i^i)^{\mathrm{T}} m_i \mathbf{S}\left(\boldsymbol{\omega}_i^i\right) \mathbf{r}_i^{c_i} + m_i (\boldsymbol{\omega}_i^i)^{\mathrm{T}} \mathbf{S}(\mathbf{r}_i^{c_i})^{\mathrm{T}} \mathbf{S}(\mathbf{r}_i^{c_i}) \boldsymbol{\omega}_i^i + (\boldsymbol{\omega}_i^i)^{\mathrm{T}} \mathbf{I}_i^i \boldsymbol{\omega}_i^i) \,. \tag{2.28}$$

Now, by using Steiner's theorem,

$$\bar{\mathbf{I}}_i^i := \mathbf{I}_i^i + m_i \mathbf{S}^{\mathrm{T}}(\mathbf{r}_i^{c_i}) \mathbf{S}(\mathbf{r}_i^{c_i}) \,,$$

the new inertia tensor with respect to the origin of frame i, $\bar{\mathbf{I}}_i^i$, can be defined. Substituting $\bar{\mathbf{I}}_i^i$ yields

$$T_i = \frac{1}{2} (m_i (\dot{\mathbf{d}}_i^i)^{\mathrm{T}} \dot{\mathbf{d}}_i^i + 2 (\dot{\mathbf{d}}_i^i)^{\mathrm{T}} m_i \mathbf{S}(\boldsymbol{\omega}_i^i) \mathbf{r}_i^{c_i} + (\boldsymbol{\omega}_i^i)^{\mathrm{T}} \bar{\mathbf{I}}_i^i \boldsymbol{\omega}_i^i) \,. \tag{2.29}$$

At this point, it is important to note that (2.29) is now linear with respect to $m_i, m_i \mathbf{r}_i^{c_i}$ and the elements of $\bar{\mathbf{I}}_i^i$. Additionally, each of these values is constant, which means the

kinetic energy part of the Lagrangian can be transformed into the form of (2.23) [2].

Moreover, considering the potential energy (2.21) of link i

$$U_i = m_i(\mathbf{g}_0)^{\mathrm{T}}\mathbf{p}_0^{c_i} \tag{2.30}$$

and substituting again $\mathbf{p}_0^{c_i}$ as in (2.25), it becomes clear that the potential energy

$$U_i = (\mathbf{g}_0)^{\mathrm{T}} m_i(\mathbf{d}_0^i + \mathbf{R}_0^i \mathbf{r}_i^{c_i}) \tag{2.31}$$

is also linear with respect to m_i and $m_i\mathbf{r}_i^{c_i}$ [2].

Therefore, the Lagrangian of link i

$$\begin{aligned} L_i &= T_i - U_i \\ &= \frac{1}{2}\Big(m_i(\dot{\mathbf{d}}_i^i)^{\mathrm{T}}\dot{\mathbf{d}}_i^i + 2(\dot{\mathbf{d}}_i^i)^{\mathrm{T}} m_i \mathbf{S}(\boldsymbol{\omega}_i^i)\mathbf{r}_{c_i}^i + (\boldsymbol{\omega}_i^i)^{\mathrm{T}}\bar{\mathbf{I}}_i^i\boldsymbol{\omega}_i^i\Big) - (\mathbf{g}_0)^{\mathrm{T}} m_i(\mathbf{d}_0^i + \mathbf{R}_0^i\mathbf{r}_i^{c_i}) \,, \end{aligned} \tag{2.32}$$

can be rearranged into a multiplication of the two (10×1) vectors $\boldsymbol{\beta}_i$ and $\boldsymbol{\pi}_i'$, as [2]

$$L_i = \boldsymbol{\beta}_i^{\mathrm{T}}\boldsymbol{\pi}_i' = \begin{bmatrix} \frac{1}{2}(\dot{\mathbf{d}}_i^i)^{\mathrm{T}}\dot{\mathbf{d}}_i^i - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^i \\ (\dot{\mathbf{d}}_i^i)^{\mathrm{T}}\mathbf{S}(\boldsymbol{\omega}_i^i) - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{R}_0^i \\ \frac{1}{2}(\omega_{i,x}^i)^2 \\ \omega_{i,x}^i\omega_{i,y}^i \\ \omega_{i,x}^i\omega_{i,z}^i \\ \frac{1}{2}(\omega_{i,y}^i)^2 \\ \omega_{i,y}^i\omega_{i,z}^i \\ \frac{1}{2}(\omega_{i,z}^i)^2 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} m_i \\ m_i\mathbf{r}_i^{c_i} \\ \bar{I}_{i,xx}^i \\ \bar{I}_{i,xy}^i \\ \bar{I}_{i,xz}^i \\ \bar{I}_{i,yy}^i \\ \bar{I}_{i,yz}^i \\ \bar{I}_{i,zz}^i \end{bmatrix} . \tag{2.33}$$

Note that $\boldsymbol{\beta}_i, \boldsymbol{\pi}_i' \in \mathbb{R}^{10\times 1}$, because $\mathbf{r}^{c_i}$ already has three components.

Furthermore, the stacked $(10n \times 1)$ vectors $\boldsymbol{\beta}$ and $\boldsymbol{\pi}'$ can be defined as

$$\boldsymbol{\beta} := \left[\boldsymbol{\beta}_1^{\mathrm{T}}, \boldsymbol{\beta}_2^{\mathrm{T}}, \ldots, \boldsymbol{\beta}_n^{\mathrm{T}}\right]^{\mathrm{T}} \qquad \boldsymbol{\pi}' := \left[\boldsymbol{\pi}_1'^{\mathrm{T}}, \boldsymbol{\pi}_2'^{\mathrm{T}}, \ldots, \boldsymbol{\pi}_n'^{\mathrm{T}}\right]^{\mathrm{T}} , \tag{2.34}$$

such that the Lagrangian of the whole robot simplifies to [2]

$$L = \sum_{i=1}^{n} L_i = \sum_{i=1}^{n} \boldsymbol{\beta}_i^{\mathrm{T}}\boldsymbol{\pi}_i' = \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\pi}' \,. \tag{2.35}$$

Note that in general all components of $\boldsymbol{\beta}_i$ are dependent on $\mathbf{q}$ or $\dot{\mathbf{q}}$, but none of $\boldsymbol{\pi}_i'$. That, combined with the fact that the Lagrange equations (2.2) preserve linearity, is why when the Lagrange equations are applied to (2.35), they simplify to

$$\boldsymbol{\tau} = \left(\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\boldsymbol{\beta}}{\partial\dot{\mathbf{q}}}\right)^{\mathrm{T}} - \left(\frac{\partial\boldsymbol{\beta}}{\partial\mathbf{q}}\right)^{\mathrm{T}}\right)\boldsymbol{\pi}' \,. \tag{2.36}$$

This gives a concrete expression to define the regressor matrix $\mathbf{Y}'$:

$$\mathbf{Y}' := \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \boldsymbol{\beta}}{\partial \dot{\mathbf{q}}}\right)^{\mathrm{T}} - \left(\frac{\partial \boldsymbol{\beta}}{\partial \mathbf{q}}\right)^{\mathrm{T}} . \tag{2.37}$$

Finally, substituting $\mathbf{Y}'$ into (2.36), transforms it into the form of (2.23) [2].

2.3 Trajectories

The *trajectory* of a robot describes the motion of the robot as a function of time. It can be defined either in joint or operational space. The trajectory can be divided into two parts. The *path* purely describes the geometric sequence of points the robot passes by, and the *time parametrization* specifies how fast the robot moves [5].

Typically, the path $\mathbf{q}_p(s)$ is only defined for inputs in range $[0, 1]$, where $\mathbf{q}_p(s)|_{s=0}$ is the starting point and $\mathbf{q}_p(s)|_{s=1}$ is the endpoint. The time parametrization $s(t)$ maps each value between $t = 0$ and $t = T$ to a geometric parameter $s \in [0, 1]$, where T is the trajectory duration. Finally, the trajectory $\mathbf{q}_t$ is the composition of both functions [5]

$$\begin{aligned} \mathbf{q}_p &: [0, 1] \to \mathbb{R}^n , \\ s &: [0, T] \to [0, 1] , \\ \mathbf{q}_t := (\mathbf{q}_p \circ s) &: [0, T] \to \mathbb{R}^n , \end{aligned}$$

where n is the number of degrees of freedom of the robot.

2.4 Static Optimization

Static Optimization is the process of finding a minimum or maximum value of a function $f(\mathbf{x})$, where $\mathbf{x}$ is an element of the finite-dimensional space $\mathbb{R}^n$. This function can generally be of any type, and the input value $\mathbf{x}$ is subject to some constraints. These constraints can either be equality or inequality constraints [6]. Formally, a static optimization problem is presented as follows,

$$\min_{\mathbf{x} \in \mathbb{R}^n} \; f(\mathbf{x}) \tag{2.38a}$$

$$\text{s.t. } g_i(\mathbf{x}) = 0, \; i \in 1, \ldots, p \tag{2.38b}$$

$$h_j(\mathbf{x}) \leq 0, \; j \in 1, \ldots, q \, . \tag{2.38c}$$

As discussed above, $f(\mathbf{x})$ is the function that should be minimized. Additionally, $g_i(\mathbf{x})$ and $h_j(\mathbf{x})$ represent p equality and q inequality constraints on $\mathbf{x}$ respectively. If there are no constraints on $\mathbf{x}$, the problem is called *unconstrained* otherwise it is called *constrained* [6].

2.4.1 Least Squares Method

The least squares method is a method for solving unconstrained optimization problems on linear systems of equations. Such systems of equations can always be written as

$$\mathbf{A}\mathbf{x} = \mathbf{b}\,, \tag{2.39}$$

where the matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ and the vector $\mathbf{b} \in \mathbb{R}^m$ are known and the vector $\mathbf{x} \in \mathbb{R}^n$ is unknown. If $m > n$ and $\mathbf{A}$ has rank m, the system of equations is overdetermined, which means there is no exact solution. If $m < n$ and $\mathbf{A}$ has rank m, the system is underdetermined and there is no unique solution [2]. In this work, only overdetermined systems of equations are of interest.

$\hat{\mathbf{x}}$ is called a least square solution of (2.39), if it solves the following problem

$$\hat{\mathbf{x}} := \min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2\,. \tag{2.40}$$

If $m > n$ and the columns of $\mathbf{A}$ are linearly independent, the solution of (2.40) is given as

$$\hat{\mathbf{x}} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{b} = \mathbf{A}^{\dagger}\mathbf{b}\,, \tag{2.41}$$

where $\mathbf{A}^{\dagger}$ denotes the left pseudo inverse of $\mathbf{A}$ [3].

The least squares method also has the advantage that it allows for the calculation of the relative standard deviation $\sigma_{\hat{\mathbf{x}}_j,r\%}$ for each value $\hat{\mathbf{x}}_j \in \hat{\mathbf{x}}$ under certain assumptions. These assumptions are that the residual error vector $\boldsymbol{\nu}$ of the overdetermined system

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \boldsymbol{\nu} \tag{2.42}$$

is a zero mean additive independent Gaussian noise with standard deviation $\sigma_{\boldsymbol{\nu}}$ and that $\mathbf{A}$ is deterministic. In that case $\sigma_{\hat{\mathbf{x}}_j,r\%}$ can be calculated using the following equations, as in [3].

Because $\boldsymbol{\nu}$ is assumed to be independent and zero mean, its variance matrix $\mathbf{C}_{\boldsymbol{\nu}}$ must be in the form

$$\mathbf{C}_{\boldsymbol{\nu}} = \mathrm{Cov}(\boldsymbol{\nu}) = \mathbb{E}[\boldsymbol{\nu}\boldsymbol{\nu}^{\mathrm{T}}] = \sigma_{\boldsymbol{\nu}}^2\mathbf{E}\,, \tag{2.43}$$

where Cov is the covariance operator and $\mathbb{E}$ is the expectation operator. Additionally, because $\boldsymbol{\nu}$ is assumed to be Gaussian and zero mean, $\sigma_{\boldsymbol{\nu}}^2$ can be estimated using the unbiased estimator [3]

$$\hat{\sigma}_{\boldsymbol{\nu}}^2 := \frac{\|\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\|_2}{m-n}\,. \tag{2.44}$$

Here, the numbers in the denominator $m - n$ are the number of rows and columns of $\mathbf{A} \in \mathbb{R}^{m\times n}$ respectively and the $\hat{}$ sign over $\hat{\sigma}_{\boldsymbol{\nu}}$ indicates that it is an estimator.

Consequential, using that $\mathbf{A}$ is deterministic, the variance matrix $\mathbf{C}_{\hat{\mathbf{x}}}$ of $\hat{\mathbf{x}}$ can be derived the following [3].

$$\mathbf{C}_{\hat{\mathbf{x}}} = \mathrm{Cov}(\hat{\mathbf{x}}) = \mathbf{A}^{\dagger}\mathbb{E}[\mathbf{b}\mathbf{b}^{\mathrm{T}}](\mathbf{A}^{\dagger})^{\mathrm{T}} = \mathbf{A}^{\dagger}\mathrm{Cov}(\boldsymbol{\nu})(\mathbf{A}^{\dagger})^{\mathrm{T}} = \sigma_{\boldsymbol{\nu}}^2(\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1} \tag{2.45}$$

Then, the standard deviation $\sigma_{\hat{x}_j}$ of any parameter $\hat{x}_j \in \hat{\mathbf{x}}$ can be calculated like

$$\sigma_{\hat{x}_j} = \sqrt{\mathbf{C}_{\hat{\mathbf{x}}}(j,j)}\,. \tag{2.46}$$

The notation $\mathbf{C}_{\hat{\mathbf{x}}}(j,j)$ means picking the element in the j^{th} row and column of matrix $\mathbf{C}_{\hat{\mathbf{x}}}$. Furthermore, the relative standard deviation of element $\sigma_{\hat{x}_j,r\%}$ is defined as [3]

$$\sigma_{\hat{x}_j,r\%} := 100\frac{\sigma_{\hat{x}_j}}{|\hat{x}_j|}\,. \tag{2.47}$$

2.5 Error Analysis and Condition Number

Error Analysis is the study of errors in numerical calculations. There are several different error sources. Common examples are modeling errors arising from simplified models, measurement errors, or discretization errors. All these errors occur because computers cannot solve arbitrary complex systems with arbitrary precision in a reasonable amount of time [7].

However, there is also the fact that different mathematical problems are differently sensitive to numerical errors. A measure of how much input perturbations are amplified by a mathematical problem in the worst case is the *condition number*, see [3].

For this thesis, only condition numbers of matrices are relevant. The condition number of a matrix gives an upper bound on how much changes in the input of a system of equations influence its solution. To see this property, consider the system of equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}\,,$$

where $\mathbf{A}$ is an invertable square matrix. Some perturbation $\Delta\mathbf{b}$ in the input will lead to an error $\Delta\mathbf{x}$ in the systems solution,

$$\mathbf{A}(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}\,.$$

Then, the relative error of the solution can be measured in an arbitrary norm $||\cdot||_p$. Any norm $||\cdot||_p$ is defined as [3]

$$||\mathbf{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$

for vectors and matrices as [3]

$$||\mathbf{A}||_p = \max\left\{\frac{||\mathbf{A}\mathbf{x}||_p}{||\mathbf{x}||_p} \mid ||\mathbf{x}||_p \neq \mathbf{0}\right\}.$$

The relative error can be expressed in the following way: [3]

$$\frac{||\Delta\mathbf{x}||_p}{||\mathbf{x}||_p} = \frac{||\mathbf{A}^{-1}\Delta\mathbf{b}||_p}{||\mathbf{A}^{-1}\mathbf{b}||_p} \leq \frac{||\mathbf{A}^{-1}||_p||\Delta\mathbf{b}||_p}{\frac{||\mathbf{b}||_p}{||\mathbf{A}||_p}} = ||\mathbf{A}||_p||\mathbf{A}^{-1}||_p\frac{||\Delta\mathbf{b}||_p}{||\mathbf{b}||_p}\,. \tag{2.48}$$

In this example, the factor relating the perturbation of the input with the error of the solution, and therefore the condition number, is $||\mathbf{A}||_p||\mathbf{A}^{-1}||_p$. In general, the condition number of a matrix relative to the norm $||_p$ is defined as

$$\mathrm{cond}_p(\mathbf{A}) := ||\mathbf{A}||_p||\mathbf{A}^{\dagger}||_p \,, \tag{2.49}$$

where $\mathbf{A}^{\dagger}$ is again the pseudo inverse of $\mathbf{A}$. By convention, if no number is attached to the cond_p function, the condition number relative to the Euclidean norm $||\cdot||_2$ is referred to. In that special case, and if the number of rows of the matrix is greater than the number of columns, the norm of a matrix simplifies to [3]

$$||\mathbf{A}||_2 = \sqrt{\max\{\mathrm{eig}(\mathbf{A}^{\mathrm{T}}\mathbf{A})\}} \,, \tag{2.50}$$

and the condition number is simplified to

$$\mathrm{cond}(\mathbf{A}) = \frac{\sqrt{\max\{\mathrm{eig}(\mathbf{A}^{\mathrm{T}}\mathbf{A})\}}}{\sqrt{\min\{\mathrm{eig}(\mathbf{A}^{\mathrm{T}}\mathbf{A})\}}} \,, \tag{2.51}$$

where $\mathrm{eig}(\mathbf{A})$ denotes all eigenvalues of matrix $\mathbf{A}$, see, e.g. [3].

3 Parameter Identification

This chapter presents the algorithm for the parameter identification of the KUKA LBR iiwa R820. Furthermore, it showcases all the considerations that lead up to the algorithm.

The proposed solution is based on the paper [1] and relies on the fact that the dynamic model of any rigid link robot is linear with respect to a set of dynamic parameters that are independent of the robot configuration. This is shown in detail in Section 3.1. Moreover, Section 3.2 showcases how these parameters are calculated from measurements. The quality of the parameter estimations is highly dependent on the robot's trajectory during the measurements. Therefore, Section 3.3 deals with the problem of finding a good trajectory.

3.1 KUKA LBR iiwa R820

This section introduces the KUKA LBR iiwa R820, including its kinematic and dynamic model. Additionally, the friction model is explained, and the method of base parameter reduction is applied to the robot.

3.1.1 Forward Kinematics

The KUKA LBR iiwa R820 is articulated with seven degrees of freedom (DoF). Given that, the transformation from link n to the previous one $n-1$ is characterized by the five parameters $[d_{n,y}, d_{n,z}, \phi_n, \theta_n, \psi_n]$ as

$$\mathbf{H}_{n-1}^{n} = \mathbf{H}_{Ty,d_{n,y}} \mathbf{H}_{Tz,d_{n,z}} \mathbf{H}_{Rz,\phi_n} \mathbf{H}_{Rx,\theta_n} \mathbf{H}_{Rz,\psi_n} \, . \tag{3.1}$$

The notation $\mathbf{H}_{Ty,d_{y,n}}$ presents a homogeneous transformation that only causes a translation along the y-axis of $d_{y,n}$ units. $\mathbf{H}_{Rz,\phi_n}$ denotes a homogeneous transformation, which only causes a rotation of ϕ_n around the z-axis. Note that these forward kinematics are different to the Modified Denavit-Hartenberg (MDH) convention, which is used in [1]. In this work forward kinematics similar to [8] are preferred to the MDH convention, because they offer a more intuitive description of the structure of the KUKA LBR iiwa R820.

Additionally, because the robot in the experimental setup hangs upside down, the vector of the gravitational force in the base frame $\mathbf{g}_0$ is defined to act in the positive z direction:

$$\mathbf{g}_0 := \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} , \tag{3.2}$$

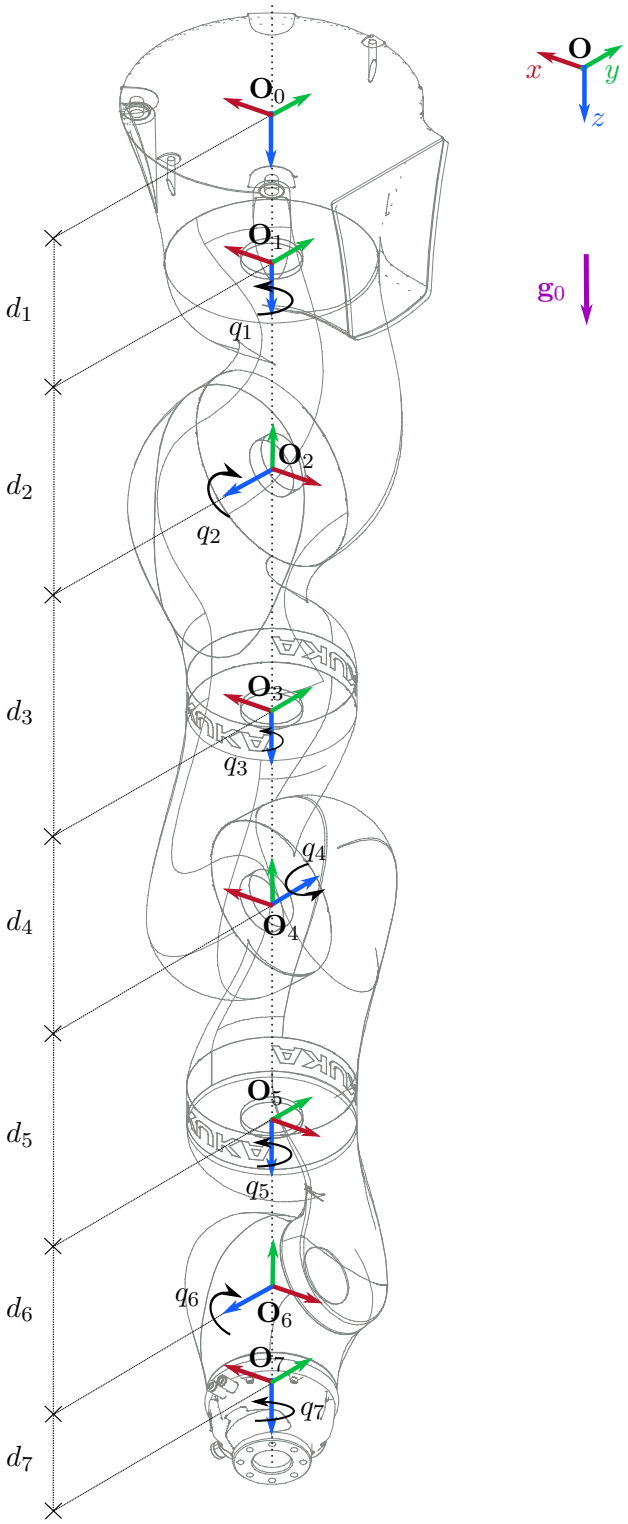

Figure 3.1: Schematic of the KUKA LBR iiwa R820

where g is the acceleration due to the earth's gravity and is assumed to be $9.81\,\mathrm{m/s^2}$. A schematic of the KUKA LBR iiwa R820 with the different coordinate systems attached and the direction of the gravitational force is depicted in Figure 3.1.

The forward kinematics of the KUKA LBR iiwa R820 can be calculated by the parameters shown in Table 3.1.

Link n	$d_{n,y}$	$d_{n,z}$	ϕ_n	θ_n	ψ_n
1	0	d_1	0	0	q_1
2	0	d_2	$-\pi$	$\frac{\pi}{2}$	q_2
3	d_3	0	π	$\frac{\pi}{2}$	q_3
4	0	d_4	0	$\frac{\pi}{2}$	q_4
5	d_5	0	π	$\frac{\pi}{2}$	q_5
6	0	d_6	0	$\frac{\pi}{2}$	q_6
7	d_7	0	π	$\frac{\pi}{2}$	q_7

Table 3.1: Parameters of the KUKA LBR iiwa R820 [8]

3.1.2 Dynamic model

The dynamic model of the KUKA LBR iiwa R820 can be described by

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{F}_s\,\mathrm{sgn}(\dot{\mathbf{q}}) \tag{3.3}$$

according to [1]. As discussed in Section 2.2.6, $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^7$ describe the joint position, velocities, and accelerations. $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ are the matrices describing the Centrifugal and Coriolis forces and $\mathbf{g}(\mathbf{q})$ represents the gravitational force. Note that because the KUKA LBR iiwa R820 has only revolute joints, all elements of $\boldsymbol{\tau}$ are torques.

$\mathbf{F}_v$ and $\mathbf{F}_s \in \mathbb{R}^{7\times7}$ are the viscous and Coulomb frictions respectively. Viscous friction describes the friction that increases proportionally with the velocity. On the other hand, Coulomb friction, also called static friction, does not increase with the velocity of the joint. However, its presence means that a nonzero torque has to be applied to cause the joint to move [5]. Figure 3.2 illustrates the different kinds of friction.

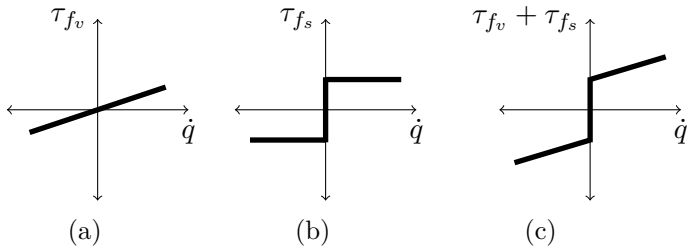

Figure 3.2: Different friction models

Figure 3.2 (a) shows how the torque introduced by viscous friction changes with actuation speed. Figure 3.2 (b) and (c) depict the same relation with Coulomb friction and viscous and Coulomb friction together, respectively. Both kinds of friction are assumed to be independent for different joints. Therefore, $\mathbf{F}_v$ and $\mathbf{F}_s$ are diagonal matrices in the form $\mathrm{diag}(F_{1v}, \ldots, F_{7v})$ and $\mathrm{diag}(F_{1s}, \ldots, F_{7s})$.

3.1.3 Dynamic parameters

Section 2.2.7 shows how the equations of motion can be rearranged into the form of (2.23). In this section, (2.23) is extended to consider the friction model. This is because the equations of motion of the KUKA LBR iiwa R820 (3.3) contain friction terms. Using $\mathbf{Y}'$ and $\boldsymbol{\pi}'$ from (2.34) and (2.37), (3.3) can be transformed into

$$\boldsymbol{\tau} = \mathbf{Y}'(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi}' + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{F}_s \operatorname{sgn}(\dot{\mathbf{q}}) \,. \tag{3.4}$$

Since $\mathbf{F}_v$ and $\mathbf{F}_s$ are constant diagonal matrices, (3.4) can be transformed into (2.23) easily by defining

$$\boldsymbol{\pi}_i := [\boldsymbol{\pi}_i'^{\mathrm{T}}, F_{iv}, F_{is}]^{\mathrm{T}} \tag{3.5a}$$

$$\boldsymbol{\pi} := [\boldsymbol{\pi}_1^{\mathrm{T}}, \boldsymbol{\pi}_2^{\mathrm{T}}, \ldots, \boldsymbol{\pi}_n^{\mathrm{T}}]^{\mathrm{T}} \tag{3.5b}$$

and [1]

$$\mathbf{Y}_i(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) := \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \boldsymbol{\beta}_i}{\partial \dot{\mathbf{q}}}\right)^{\mathrm{T}} - \left(\frac{\partial \boldsymbol{\beta}_i}{\partial \mathbf{q}}\right)^{\mathrm{T}} \tag{3.6a}$$

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) := \begin{bmatrix} & \dot{q}_1 & \operatorname{sgn}(\dot{q}_1) & & 0 & 0 & \ldots & 0 & 0 \\ & 0 & 0 & & \dot{q}_2 & \operatorname{sgn}(\dot{q}_2) & & 0 & 0 \\ \mathbf{Y}_1 & \vdots & \vdots & \mathbf{Y}_2 & \vdots & \vdots & \ddots & \mathbf{Y}_n \vdots & \vdots \\ & 0 & 0 & & 0 & 0 & \ldots & \dot{q}_n & \operatorname{sgn}(\dot{q}_n) \end{bmatrix} . \tag{3.6b}$$

Furthermore, using the substitutions [1]

$$\begin{aligned}
M_i &:= m_i \,, \\
MX_i &:= m_i r_{i,x}^{c_i} \,, \\
MY_i &:= m_i r_{i,y}^{c_i} \,, \\
MZ_i &:= m_i r_{i,z}^{c_i} \,, \\
XX_i &:= \bar{I}_{i,xx}^{i} = I_{i,xx}^{i} + m_i\Big((r_{i,y}^{c_i})^2 + (r_{i,z}^{c_i})^2\Big) \,, \\
XY_i &:= \bar{I}_{i,xy}^{i} = I_{i,xy}^{i} - m_i r_{i,x}^{c_i} r_{i,y}^{c_i} \,, \\
XZ_i &:= \bar{I}_{i,xz}^{i} = I_{i,xz}^{i} - m_i r_{i,x}^{c_i} r_{i,z}^{c_i} \,, \\
YY_i &:= \bar{I}_{i,yy}^{i} = I_{i,yy}^{i} + m_i\Big((r_{i,y}^{c_i})^2 + (r_{i,z}^{c_i})^2\Big) \,, \\
YZ_i &:= \bar{I}_{i,yz}^{i} = I_{i,yz}^{i} - m_i r_{i,y}^{c_i} r_{i,z}^{c_i} \,, \\
ZZ_i &:= \bar{I}_{i,zz}^{i} = I_{i,zz}^{i} + m_i\Big((r_{i,x}^{c_i})^2 + (r_{i,y}^{c_i})^2\Big) \,, \\
FV_i &:= F_{iv} \,, \\
FS_i &:= F_{is} \,,
\end{aligned} \tag{3.7}$$

the vector of dynamic parameters $\boldsymbol{\pi}$ are

$$\boldsymbol{\pi}_i = [M_i, MX_i, MY_i, MZ_i, XX_i, XY_i, XZ_i, YY_i, YZ_i, ZZ_i, FV_i, FS_i]^{\mathrm{T}} \,. \tag{3.8}$$

3.1.4 Base parameters

Section 2.2.7 has shown how the equations of motion can be rearranged into (2.23), where the torque vector $\boldsymbol{\tau}$ only depends on the regressor matrix $\mathbf{Y}$ and the $\boldsymbol{\pi}$ parameter vector. Importantly, the KUKA LBR iiwa R820 is capable of measuring $\boldsymbol{\tau}$ and $\mathbf{q}$. Additionally, $\mathbf{Y}$ can be reconstructed using measurements of $\mathbf{q}$. Therefore, the only unknown in (2.23) is $\boldsymbol{\pi}$. To fully exploit (2.23), $\mathbf{Y}$ must have column maximum rank. However, this is not the case, as this section will show, because there are columns in $\mathbf{Y}$ which are either $\mathbf{0}$ or linearly dependent.

One way to explore why there are columns in $\mathbf{Y}$ that are $\mathbf{0}$ or linearly dependent is to look at the definition of $\boldsymbol{\beta}$ in (2.33) and the relationships between $\dot{\mathbf{d}}_n^n, \boldsymbol{\omega}_n^n$ and $\dot{\mathbf{d}}_{n-1}^{n-1}, \boldsymbol{\omega}_{n-1}^{n-1}$. When examining $\boldsymbol{\beta}$ it will be crucial that these relationships can be described recursively as [3]

$$\begin{aligned}
\dot{\mathbf{d}}_i^i &= \mathbf{R}_i^{i-1}\Big(\dot{\mathbf{d}}_{i-1}^{i-1} + \boldsymbol{\omega}_{i-1}^{i-1} \times \mathbf{d}_{i-1}^i\Big) \,, \\
\boldsymbol{\omega}_i^i &= \mathbf{R}_i^{i-1}\boldsymbol{\omega}_{i-1}^{i-1} + \dot{q}_i \mathbf{a}_i^i \,,
\end{aligned} \tag{3.9}$$

where $\mathbf{d}_{i-1}^i$ is the vector from the origin of frame $i-1$ to the origin of frame i, with respect to frame $i-1$ and $\mathbf{a}_i^i$ represents the axis of actuation of link i, with respect to frame i. Since the base frame is assumed to be fixed, the starting values of $\dot{\mathbf{d}}_0^0$ and $\boldsymbol{\omega}_0^0$ are set to zero, see [3].

Additionally, to simplify the derivations, a new notation is introduced in this section. Recall how $\boldsymbol{\tau}$ is calculated in (2.23). Because of the nature of matrix-vector multiplication, to calculate $\boldsymbol{\tau}$, each column of $\mathbf{Y}$ is multiplied with exactly one element of $\boldsymbol{\pi}$. Therefore, when referring to a single column of $\mathbf{Y}$, it can be identified by the element of $\boldsymbol{\pi}$ it is multiplied within (2.23). This is notated in index notation, e. g. $\mathbf{Y}_{XX_1}$ refers to the column of $\mathbf{Y}$, which is multiplied with XX_1 in (2.23). Similarly, the same notation is also used for $\boldsymbol{\beta}$, as defined in (2.33), which is used to derive $\mathbf{Y}$ in (2.37). However, because $\boldsymbol{\beta}$ is a vector, β_{XX_1} only refers to a single value, the one which is multiplied with XX_1 in (2.35) [3].

Zero Columns in Y

The zero columns are easy to identify. Because $\mathbf{Y}$ has to be calculated for Section 3.2 anyway, one can just check every column if it is zero. However, to provide an understanding why these columns are zero, their corresponding $\boldsymbol{\beta}$ expressions are explicitly calculated in this section. This is expedient, because of the relation between $\mathbf{Y}$ and $\boldsymbol{\beta}$ (2.37). From that, it follows that zero columns in $\mathbf{Y}$ correspond values in $\boldsymbol{\beta}$ that are constant with respect to $\mathbf{q}$ and $\dot{\mathbf{q}}$.

To calculate the values of $\boldsymbol{\beta}$, the first three values of $\boldsymbol{\omega}_n^n$ and the first four values of $\dot{\mathbf{d}}_n^n$ have to be calculated first.

$$
\begin{aligned}
&\boldsymbol{\omega}_0^0 = \mathbf{0} && \dot{\mathbf{d}}_0^0 = \mathbf{0} \\
&\boldsymbol{\omega}_1^1 = \mathbf{R}_1^0\boldsymbol{\omega}_0^0 + \begin{bmatrix}0\\0\\\dot{q}_1\end{bmatrix} = \begin{bmatrix}0\\0\\\dot{q}_1\end{bmatrix} && \dot{\mathbf{d}}_1^1 = \mathbf{R}_1^0\left(\dot{\mathbf{d}}_0^0 + \boldsymbol{\omega}_0^0 \times \mathbf{d}_0^1\right) = \mathbf{0} \\
&\boldsymbol{\omega}_2^2 = \mathbf{R}_2^1\boldsymbol{\omega}_1^1 + \begin{bmatrix}0\\0\\\dot{q}_2\end{bmatrix} = \begin{bmatrix}\dot{q}_1\mathrm{s}_2\\\dot{q}_1\mathrm{c}_2\\\dot{q}_2\end{bmatrix} && \dot{\mathbf{d}}_2^2 = \mathbf{R}_2^1\left(\dot{\mathbf{d}}_1^1 + \boldsymbol{\omega}_1^1 \times \mathbf{d}_1^2\right) = \mathbf{0} \\
& && \dot{\mathbf{d}}_3^3 = \mathbf{R}_3^2\left(\dot{\mathbf{d}}_2^2 + \boldsymbol{\omega}_2^2 \times \mathbf{d}_2^3\right) = \begin{bmatrix}d_3(\dot{q}_1\mathrm{s}_2\mathrm{s}_3 + \dot{q}_2\mathrm{c}_3)\\d_3(\dot{q}_1\mathrm{s}_2\mathrm{c}_3 - \dot{q}_2\mathrm{s}_3)\\0\end{bmatrix}
\end{aligned}
\tag{3.10}
$$

The values of $\mathbf{R}_n^{n-1}$ and $\mathbf{d}_{n-1}^n$ correspond to the rotational and translational parts of $\mathbf{H}_{n-1}^n$ as

$$
\mathbf{H}_{n-1}^n = \begin{bmatrix}(\mathbf{R}_n^{n-1})^{-1} & \mathbf{d}_{n-1}^n\\ \mathbf{0} & 1\end{bmatrix}, \tag{3.11}
$$

where $\mathbf{H}_{n-1}^n$ is defined in (3.1). From this, it can already be seen that $\boldsymbol{\beta}_1$ is mostly empty

$$
\boldsymbol{\beta}_1 = \begin{bmatrix}
\beta_{M_1}\\
\begin{bmatrix}\beta_{MX_1} & \beta_{MY_1} & \beta_{MZ_1}\end{bmatrix}^{\mathrm{T}}\\
\beta_{XX_1}\\
\beta_{XY_1}\\
\beta_{XZ_1}\\
\beta_{YY_1}\\
\beta_{YZ_1}\\
\beta_{ZZ_1}
\end{bmatrix} = \begin{bmatrix}
\frac{1}{2}(\dot{\mathbf{d}}_1^1)^{\mathrm{T}}\dot{\mathbf{d}}_1^1 - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^1\\
(\dot{\mathbf{d}}_1^1)^{\mathrm{T}}\mathbf{S}(\boldsymbol{\omega}_1^1) - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{R}_0^1\\
\frac{1}{2}(\omega_{1,x}^1)^2\\
\omega_{1,x}^1\omega_{1,y}^1\\
\omega_{1,x}^1\omega_{1,z}^1\\
\frac{1}{2}(\omega_{1,y}^1)^2\\
\omega_{1,y}^1\omega_{1,z}^1\\
\frac{1}{2}(\omega_{1,z}^1)^2
\end{bmatrix} = \begin{bmatrix}
-(\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^1\\
-(\mathbf{g}_0)^{\mathrm{T}}\mathbf{R}_0^1\\
0\\0\\0\\0\\0\\\frac{1}{2}\dot{q}_1^2
\end{bmatrix}. \tag{3.12}
$$

In addition, $-(\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^1$ and $-(\mathbf{g}_0)^{\mathrm{T}}\mathbf{R}_0^1$ do not depend on $\mathbf{q}$ or $\dot{\mathbf{q}}$, which means the columns $\mathbf{Y}_{M_1}, \mathbf{Y}_{MX_1}, \mathbf{Y}_{MY_1}, \mathbf{Y}_{MZ_1}, \mathbf{Y}_{XX_1}, \mathbf{Y}_{XY_1}, \mathbf{Y}_{XZ_1}, \mathbf{Y}_{YY_1}$ and $\mathbf{Y}_{YZ_1}$ must be all zero. Furthermore, because

$$\begin{aligned} \beta_{M_2} &= \frac{1}{2}(\dot{\mathbf{d}}_2^2)^{\mathrm{T}}\dot{\mathbf{d}}_2^2 - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^2 = -(\mathbf{g}_0)^{\mathrm{T}}\mathbf{d}_0^2 = -g(d_1 + d_2)\,, \\ \beta_{MZ_2} &= ((\dot{\mathbf{d}}_2^2)^{\mathrm{T}}\mathbf{S}(\boldsymbol{\omega}_2^2) - (\mathbf{g}_0)^{\mathrm{T}}\mathbf{R}_0^2)_z = 0\,, \end{aligned} \tag{3.13}$$

where $*)_z$ means taking the z component of the vector, $\boldsymbol{\beta}_{M_2}$ and $\boldsymbol{\beta}_{MZ_2}$ are also constant with respect to $\mathbf{q}$ and $\dot{\mathbf{q}}$. Therefore $\mathbf{Y}_{M_2}$ and $\mathbf{Y}_{MZ_2}$ are also zero columns.

These are all columns in $\mathbf{Y}$ that are zero because proceeding the calculations for $\boldsymbol{\omega}_n^n$ and $\dot{\mathbf{d}}_n^n$ (3.10) for bigger n only leads to values that depend on $\mathbf{q}$ or $\dot{\mathbf{q}}$. Physically, these columns correspond to $\boldsymbol{\pi}$ parameters that do not contribute to $\boldsymbol{\tau}$. This means that these parameters have no impact on the robot's motion. Therefore, they are also irrelevant for predicting $\boldsymbol{\tau}$ and can be removed from the model. This step will be discussed further in this section [1], [3].

Linear Dependent Columns in Y

It is inefficient to compute the values of $\boldsymbol{\beta}$ and deduce linear dependence in $\mathbf{Y}$. Thus, a systematic approach considering the definitions of β, $\boldsymbol{\omega}_n^n$ and $\dot{\mathbf{d}}_n^n$ must be taken. For the given forward kinematics of the KUKA LBR iiwa R820, three different linear dependence relationships can be identified [1], [3].

The simplest one emerges from two facts about the parameters in Table 3.1. First, ϕ_n is always an integer multiple of π and θ_n is always $\frac{\pi}{2}$, except for link one. Therefore, $\mathbf{R}_n^{n-1}$ can be written as

$$\mathbf{R}_n^{n-1} = \begin{bmatrix} \mp\mathrm{c}_n & 0 & \mathrm{s}_n \\ \pm\mathrm{s}_n & 0 & \mathrm{c}_n \\ 0 & -\mathrm{c}_{\phi_n} & 0 \end{bmatrix}, \tag{3.14}$$

for all $n \in \{2, 3, \ldots, 7\}$, where s_n and c_n are abbreviations for $\sin(q_n)$ and $\cos(q_n)$ respectively.

Secondly, the axis of actuation is always z, which means the term $\dot{q}_n\mathbf{a}_n^n$ of (3.9), can be simplified to

$$\dot{q}_n\mathbf{a}_n^n = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_n \end{bmatrix}. \tag{3.15}$$

Substituting (3.14) and (3.15) from above into the relationship between $\boldsymbol{\omega}_n^n$ and $\boldsymbol{\omega}_{n-1}^{n-1}$ (3.9) yields

$$\boldsymbol{\omega}_n^n = \begin{bmatrix} \mp\mathrm{c}_n & 0 & \mathrm{s}_n \\ \pm\mathrm{s}_n & 0 & \mathrm{c}_n \\ 0 & -\mathrm{c}_{\phi_n} & 0 \end{bmatrix}\boldsymbol{\omega}_{n-1}^{n-1} + \begin{bmatrix} 0 \\ 0 \\ \dot{q}_n \end{bmatrix} = \begin{bmatrix} \mp\mathrm{c}_n\omega_{n-1,x}^{n-1} + \mathrm{s}_n\omega_{n-1,z}^{n-1} \\ \pm\mathrm{s}_n\omega_{n-1,x}^{n-1} + \mathrm{c}_n\omega_{n-1,z}^{n-1} \\ -\mathrm{c}_{\phi_n}\omega_{n-1,y}^{n-1} + \dot{q}_n \end{bmatrix}. \tag{3.16}$$

Using (3.16), and further simplifying

$$-\mathrm{c}_{\phi_n} = \pm 1\,, \tag{3.17}$$

we can calculate the first linear relationship within the $\boldsymbol{\beta}$ vector

$$\begin{aligned}
\beta_{XX_n} + \beta_{YY_n} &= \frac{1}{2}\Big(\big(\omega_{n,x}^n\big)^2 + \big(\omega_{n,y}^n\big)^2\Big) \\
&= \frac{1}{2}\Big(\big(\mp\mathrm{c}_n\omega_{n-1,x}^{n-1} + \mathrm{s}_n\omega_{n-1,z}^{n-1}\big)^2 + \big(\pm\mathrm{s}_n\omega_{n-1,x}^{n-1} + \mathrm{c}_n\omega_{n-1,z}^{n-1}\big)^2\Big) \\
&= \frac{1}{2}\Big(\big(\omega_{n-1,x}^{n-1}\big)^2 + \big(\omega_{n-1,z}^{n-1}\big)^2\Big) \\
&= \beta_{XX_{n-1}} + \beta_{ZZ_{n-1}}\,.
\end{aligned} \tag{3.18}$$

Next, consider the following form of the definition of β_{MZ_n} in (2.33)

$$\beta_{MZ_n} = \Big(-\boldsymbol{\omega}_n^n \times \dot{\mathbf{d}}_n^n - \mathbf{R}_n^0\mathbf{g}_0\Big)_z\,. \tag{3.19}$$

This equation can be rewritten as

$$\beta_{MZ_n} = \Big(-\mathbf{R}_n^{n-1}\Big(\boldsymbol{\omega}_{n-1}^{n-1} \times \Big(\dot{\mathbf{d}}_{n-1}^{n-1} + \boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big)\Big) - \mathbf{R}_n^{n-1}\mathbf{R}_{n-1}^0\mathbf{g}_0\Big)_z\,, \tag{3.20}$$

using the definitions of $\boldsymbol{\omega}_n^n$ and $\dot{\mathbf{d}}_n^n$, (3.9), and the fact that the z coordinates of the input vectors of the cross product do not influence the z coordinate of the output of the cross product. This is important since we can conclude that the $\dot{q}_n\mathbf{a}_n^n$ term of (3.9) can be omitted.

Because $\mathbf{R}_n^{n-1}$ has the form (3.14), it maps the y-coordinate of any vector multiplied with it to the right to the z-coordinate of the product and adds the factor $-\mathrm{c}_{\phi_n}$. Therefore, for any vector $\mathbf{v} \in \mathbb{R}^3$,

$$\Big(\mathbf{R}_n^{n-1}\mathbf{v}\Big)_z = -\mathrm{c}_{\phi_n}(\mathbf{v})_y\,. \tag{3.21}$$

Using this fact, β_{MZ_n} is further simplified to

$$\begin{aligned}
\beta_{MZ_n} &= (-\mathrm{c}_{\phi_n})\Big(-\boldsymbol{\omega}_{n-1}^{n-1} \times \Big(\dot{\mathbf{d}}_{n-1}^{n-1} + \boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big) - \mathbf{R}_{n-1}^0\mathbf{g}_0\Big)_y \\
&= (-\mathrm{c}_{\phi_n})\Big(\omega_{n-1,x}^{n-1}\dot{d}_{n-1,z}^{n-1} - \omega_{n-1,z}^{n-1}\dot{d}_{n-1,x}^{n-1} - \Big(\mathbf{R}_{n-1}^0\mathbf{g}_0\Big)_y \\
&\qquad + \omega_{n-1,x}^{n-1}\Big(\boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big)_z - \omega_{n-1,z}^{n-1}\Big(\boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big)_x\Big) \\
&= (-\mathrm{c}_{\phi_n})\Big(\Big(-\boldsymbol{\omega}_{n-1}^{n-1} \times \dot{\mathbf{d}}_{n-1}^{n-1} - \mathbf{R}_{n-1}^0\mathbf{g}_0\Big)_y \\
&\qquad + \omega_{n-1,x}^{n-1}\Big(\boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big)_z - \omega_{n-1,z}^{n-1}\Big(\boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\Big)_x\Big)\,.
\end{aligned} \tag{3.22}$$

In the last line, there are two important things to note. First, the first term is precisely $\beta_{MY_{n-1}}$. Second, $\mathbf{d}_{n-1}^n$, as defined in Table 3.1, is always constant and its x-coordinate is

always equal to zero. Thus, the two remaining terms in the last line are both products that contain two factors of $\boldsymbol{\omega}_{n-1}^{n-1}$ and the constant d_n. Therefore, by continuing (3.22) the linear dependence of β_{MZ_n} becomes visible as

$$\begin{aligned}\beta_{MZ_n} &= (-\mathrm{c}_{\phi_n})\Big(\beta_{MY_{n-1}} + \left(\omega_{n-1,x}^{n-1}\right)^2 d_{n-1,y}^n + \left(\omega_{n-1,z}^{n-1}\right)^2 d_{n-1,y}^n - \omega_{n-1,y}^{n-1}\omega_{n-1,z}^{n-1} d_{n-1,z}^n\Big) \\ &= (-\mathrm{c}_{\phi_n})\Big(\beta_{MY_{n-1}} + \begin{bmatrix} 0 \\ 2(\beta_{XX_{n-1}} + \beta_{ZZ_{n-1}}) \\ -\beta_{YZ_{n-1}} \end{bmatrix}^{\mathrm{T}} \mathbf{d}_{n-1}^n\Big)\,.\end{aligned} \tag{3.23}$$

Finally, the mass parameters of $\boldsymbol{\beta}$ are also linearly dependent. To show this, let's consider the definition of β_{M_n} in (2.33)

$$\begin{aligned}\beta_{M_n} =& \frac{1}{2}\Big(\left(\dot{\mathbf{d}}_n^n\right)^{\mathrm{T}} \dot{\mathbf{d}}_n^n\Big) - (\mathbf{g}_0)^{\mathrm{T}} \mathbf{d}_0^n \\ =& \frac{1}{2}\Big(\left(\dot{\mathbf{d}}_{n-1}^{n-1} + \boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\right)^{\mathrm{T}} \left(\mathbf{R}_n^{n-1}\right)^{\mathrm{T}} \mathbf{R}_n^{n-1} \left(\dot{\mathbf{d}}_{n-1}^{n-1} + \boldsymbol{\omega}_{n-1}^{n-1} \times \mathbf{d}_{n-1}^n\right)\Big) \\ & - (\mathbf{g}_0)^{\mathrm{T}} \left(\mathbf{d}_0^{n-1} + \mathbf{R}_0^{n-1} \mathbf{d}_{n-1}^n\right).\end{aligned} \tag{3.24}$$

Here, the definition of $\dot{\mathbf{d}}_n^n$ in (3.9) and the definition of homogeneous transformations (2.8) has been used. To this point, there are a few things to observe here. First, because the transpose of a rotation matrix is also its inverse, $\left(\mathbf{R}_n^{n-1}\right)^{\mathrm{T}}$ and $\mathbf{R}_n^{n-1}$ cancel each other out. Second, because according to Table 3.1 $\mathbf{d}_{n-1}^n$ is all zero except either its y- or z-component, the first term can only be evaluated to one of two forms. In the case $\mathbf{d}_{n-1}^n = [0, d_n, 0]^{\mathrm{T}}$, we have

$$\begin{aligned}\beta_{M_n} =& \frac{1}{2}\left(\dot{\mathbf{d}}_{n-1}^{n-1}\right)^{\mathrm{T}} \dot{\mathbf{d}}_{n-1}^{n-1} - (\mathbf{g}_0)^{\mathrm{T}} \mathbf{d}_0^{n-1} \\ & + d_n\left(-\dot{d}_{n-1,x}^{n-1}\omega_{n-1,z}^{n-1} + \dot{d}_{n-1,z}^{n-1}\omega_{n-1,x}^{n-1} - \left((\mathbf{g}_0)^{\mathrm{T}} \mathbf{R}_0^{n-1}\right)_y\right) \\ & + \frac{1}{2}(d_n)^2\left(\left(\omega_{n-1,x}^{n-1}\right)^2 + \left(\omega_{n-1,z}^{n-1}\right)^2\right).\end{aligned} \tag{3.25}$$

Note, that because of the assumption on $\mathbf{d}_{n-1}^n$, the term $(\mathbf{g}_0)^{\mathrm{T}} \mathbf{R}_0^{n-1} \mathbf{d}_{n-1}^n$ is now equivalent with picking the y-component of $(\mathbf{g}_0)^{\mathrm{T}} \mathbf{R}_0^{n-1}$ and multiplying it with d_n. Furthermore, $-\dot{d}_{n-1,x}^{n-1}\omega_{n-1,z}^{n-1} + \dot{d}_{n-1,z}^{n-1}\omega_{n-1,x}^{n-1}$ can be described as the y component of the cross product between $\dot{\mathbf{d}}_{n-1}^{n-1}$ and $\boldsymbol{\omega}_{n-1}^{n-1}$. Thus, (3.25) is simplified

$$\beta_{M_n} = \beta_{M_{n-1}} + d_n \beta_{MY_{n-1}} + (d_n)^2(\beta_{XX_{n-1}} + \beta_{ZZ_{n-1}})\,. \tag{3.26}$$

Analogous, the case $\mathbf{d}_{n-1}^n = [0, 0, d_n]^{\mathrm{T}}$ evaluates to

$$\beta_{M_n} = \beta_{M_{n-1}} + d_n \beta_{MZ_{n-1}} + (d_n)^2(\beta_{XX_{n-1}} + \beta_{YY_{n-1}})\,. \tag{3.27}$$

However, (3.27) can be further simplified. Note that (3.27) consists only of terms that have already been shown to be linearly dependent. Moreover, note that according to Table 3.1 the two cases $\mathbf{d}_{n-1}^{n} = [0, d_n, 0]^{\mathrm{T}}$ and $\mathbf{d}_{n-1}^{n} = [0, 0, d_n]^{\mathrm{T}}$ alternate between consecutive links. Thus,

$$\mathbf{d}_{n-1}^{n} = [0, 0, d_n]^{\mathrm{T}} \implies \mathbf{d}_{n-2}^{n-1} = [0, d_{n-1}, 0]^{\mathrm{T}} . \tag{3.28}$$

Therefore, $\beta_{M_{n-1}}$ in (3.27) can be substituted according to (3.26). Additionally applying (3.18) and (3.23) to (3.27) results in

$$\begin{aligned} \beta_{M_n} =& \beta_{M_{n-2}} + d_{n-1}\beta_{MY_{n-2}} + (d_{n-1})^2(\beta_{XX_{n-2}} + \beta_{ZZ_{n-2}}) \\ &+ d_n(\beta_{MY_{n-2}} + 2d_{n-1}(\beta_{XX_{n-2}} + \beta_{ZZ_{n-2}})) + (d_n)^2(\beta_{XX_{n-2}} + \beta_{YY_{n-2}}) , \end{aligned} \tag{3.29}$$

where (3.28) has also been used to simplify (3.23).

Finally, (3.29) simplifies to

$$\beta_{M_n} = \beta_{M_{n-2}} + (d_{n-1} + d_n)\beta_{MY_{n-2}} + (d_{n-1} + d_n)^2(\beta_{XX_{n-2}} + \beta_{ZZ_{n-2}}) . \tag{3.30}$$

There are three different kinds of linear dependencies within $\boldsymbol{\beta}$ and therefore also in the columns of $\mathbf{Y}$. These are

$$\mathbf{Y}_{YY_n} = -\mathbf{Y}_{XX_n} + \mathbf{Y}_{XX_{n-1}} + \mathbf{Y}_{ZZ_{n-1}} \tag{3.31a}$$

$$\mathbf{Y}_{MZ_n} = (-\mathrm{c}_{\phi_n})\left(\mathbf{Y}_{MY_{n-1}} + \begin{bmatrix} 0 \\ 2(\mathbf{Y}_{XX_{n-1}} + \mathbf{Y}_{ZZ_{n-1}}) \\ -\mathbf{Y}_{YZ_{n-1}} \end{bmatrix}^{\mathrm{T}} \mathbf{d}_{n-1}^{n}\right) \tag{3.31b}$$

$$\mathbf{Y}_{M_n} = \begin{bmatrix} 0 \\ \mathbf{Y}_{M_{n-1}} + d_n\mathbf{Y}_{MY_{n-1}} + (d_n)^2(\mathbf{Y}_{XX_{n-1}} + \mathbf{Y}_{ZZ_{n-1}}) \\ \mathbf{Y}_{M_{n-2}} + (d_{n-1} + d_n)\mathbf{Y}_{MY_{n-2}} + (d_{n-1} + d_n)^2(\mathbf{Y}_{XX_{n-2}} + \mathbf{Y}_{ZZ_{n-2}}) \end{bmatrix}^{\mathrm{T}} \frac{\mathbf{d}_{n-1}^{n}}{||\mathbf{d}_{n-1}^{n}||} . \tag{3.31c}$$

Mathematically, these dependencies reduce the column rank of $\mathbf{Y}$. This complicates future calculations, which is why these dependencies have to be removed [1], [3].

Base Parameter Reduction

Since the zero columns and the linearly dependent columns have been identified in the previous sections, the goal of this section is to reduce $\mathbf{Y}$ and $\boldsymbol{\pi}$ to $\mathbf{Y}_b$ and $\boldsymbol{\pi}_b$, such that

$$\boldsymbol{\tau} = \mathbf{Y}\boldsymbol{\pi} = \mathbf{Y}_b\boldsymbol{\pi}_b , \tag{3.32}$$

but $\mathbf{Y}_b$ has full column rank. The vector $\boldsymbol{\pi}_b$ is referred to as the dynamic base parameter [3]. For that, all columns that are either zero or linearly dependent have to be removed from $\mathbf{Y}_b$. All these columns and their linear relationships are listed in the Appendix (A.1).

Following (A.1), the set $\boldsymbol{\pi}_r$ is defined in (3.33), which contains all dynamic parameters for which their corresponding columns in $\mathbf{Y}$ will be removed. In general, it does not matter which column of a linear relationship is removed as long as this resolves the linear dependency. This thesis follows the convention that the column with the highest index is removed first, as in [3],

$$\begin{aligned}\boldsymbol{\pi}_r := \{&M_1, MX_1, MY_1, MZ_1, XX_1, XY_1, XZ_1, YY_1, YZ_1,\\ &M_2, MZ_2, YY_2, M_3, MZ_3, YY_3, M_4, MZ_4, YY_4,\\ &M_5, MZ_5, YY_5, M_6, MZ_6, YY_6, M_7, MZ_7, YY_7\}\,.\end{aligned} \tag{3.33}$$

Now $\mathbf{Y}_b$ can be defined in set notation as

$$\mathbf{Y}_b = \mathbf{Y} \setminus \{\mathbf{Y}_p \mid p \in \boldsymbol{\pi}_r\}\,, \tag{3.34}$$

meaning that $\mathbf{Y}_b$ only contains columns of $\mathbf{Y}$ that do not correspond to any values of $\boldsymbol{\pi}_r$.

In the next step, $\boldsymbol{\pi}_b$ has to be defined so that its dimensions match those of $\mathbf{Y}_b$ so that multiplication is still possible. For this, all parameters of $\boldsymbol{\pi}_r$ also have to be removed from $\boldsymbol{\pi}$. Since the product of $\mathbf{Y}_b$ and $\boldsymbol{\pi}_b$ must still equal $\boldsymbol{\tau}$, $\boldsymbol{\pi}_b$ has to be modified. To see how to preserve equality, let's consider a matrix with $n-1$ linearly independent columns $[\mathbf{v}_1, \ldots, \mathbf{v}_{n-1}]$ and one linearly dependent column. If it is multiplied with a column vector $[x_1, \ldots, x_n]^{\mathrm{T}}$, the product can be rewritten as follows [3]:

$$\begin{aligned}\begin{bmatrix}\mathbf{v}_1 & \ldots & \mathbf{v}_{n-1} & \sum_{i=1}^{n-1} c_i\mathbf{v}_i\end{bmatrix}\begin{bmatrix}x_1\\ \vdots\\ x_{n-1}\\ x_n\end{bmatrix} &= \begin{bmatrix}\sum_{i=1}^{n-1}\mathbf{v}_i x_i + \sum_{i=1}^{n-1} c_i\mathbf{v}_i x_n\end{bmatrix}\\ &= \begin{bmatrix}\sum_{i=1}^{n-1}\mathbf{v}_i(x_i + c_i x_n)\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{v}_1 & \ldots & \mathbf{v}_{n-1}\end{bmatrix}\begin{bmatrix}x_1 + c_1x_n\\ \vdots\\ x_{n-1} + c_{n-1}x_n\end{bmatrix}.\end{aligned} \tag{3.35}$$

Therefore, if the column corresponding to one of the remaining parameters $\boldsymbol{\pi}_b$ appears in any equation in (A.1), that parameter must be substituted, and the substitution must have the following form.
Given the linear relationship of the column of any parameter $r \in \boldsymbol{\pi}_r$

$$\mathbf{Y}_r = \sum_{p \in \boldsymbol{\pi}_b} c_{p,r}\mathbf{Y}_p\,, \tag{3.36}$$

all parameters $p \in \boldsymbol{\pi}_b$ have to substituted by

$$p_{sub} = p + c_{p,r}r\,. \tag{3.37}$$

$\boldsymbol{\pi}_b$ can be defined as $\boldsymbol{\pi} \setminus \boldsymbol{\pi}_r$, where all elements have been substituted according to (3.37).

Following this procedure for (A.1) yields the following dynamic base parameter vector $\boldsymbol{\pi}_b$:

$$\begin{aligned}\boldsymbol{\pi}_b := [&ZZR_1, FV_1, FS_1,\\ &MX_2, MYR_2, XXR_2, XY_2, XZ_2, YZ_2, ZZR_2, FV_2, FS_2,\\ &MX_3, MYR_3, XXR_3, XY_3, XZ_3, YZR_3, ZZR_3, FV_3, FS_3,\\ &MX_4, MYR_4, XXR_4, XY_4, XZ_4, YZ_4, ZZR_4, FV_4, FS_4,\\ &MX_5, MYR_5, XXR_5, XY_5, XZ_5, YZR_5, ZZR_5, FV_5, FS_5,\\ &MX_6, MYR_6, XXR_6, XY_6, XZ_6, YZ_6, ZZR_6, FV_6, FS_6,\\ &MX_7, MY_7, XXR_7, XY_7, XZ_7, YZ_7, ZZ_7, FV_7, FS_7]^{\mathrm{T}} .\end{aligned} \tag{3.38}$$

To stay consistent with existing literature, e. g., [1] or [3], substitutions are signalized by adding a capital R to the name of the parameter. So ZZR_1 or MYR_2 are substitutions. These substitutions are defined as follows:

$$\begin{aligned}
ZZR_1 &:= ZZ_1 + YY_2 \\ \hline
MYR_2 &:= MY_2 + MZ_3 + d_3M_3 + (d_3 + d_4)(M_4 + M_5 + M_6 + M_7) \\
XXR_2 &:= XX_2 - YY_2 + YY_3 + 2d_3MZ_3 + (d_3)^2M_3 + (d_3 + d_4)^2(M_4 + M_5 + M_6 + M_7) \\
ZZR_2 &:= ZZ_2 + YY_3 + 2d_3MZ_3 + (d_3)^2M_3 + (d_3 + d_4)^2(M_4 + M_5 + M_6 + M_7) \\ \hline
MYR_3 &:= MY_3 - MZ_4 \\
XXR_3 &:= XX_3 - YY_3 + YY_4 \\
YZR_3 &:= YZ_3 + d_4MZ_4 \\
ZZR_3 &:= ZZ_3 + YY_4 \\ \hline
MYR_4 &:= MY_4 + MZ_5 + d_5M_5 + (d_5 + d_6)(M_6 + M_7) \\
XXR_4 &:= XX_4 + YY_5 - YY_4 + 2d_5MZ_5 + (d_5)^2M_5 + (d_5 + d_6)^2(M_6 + M_7) \\
ZZR_4 &:= ZZ_4 + YY_5 + 2d_5MZ_5 + (d_5)^2M_5 + (d_5 + d_6)^2(M_6 + M_7) \\ \hline
MYR_5 &:= MY_5 - MZ_6 \\
XXR_5 &:= XX_5 - YY_5 + YY_6 \\
YZR_5 &:= YZ_5 + d_6MZ_6 \\
ZZR_5 &:= ZZ_5 + YY_6 \\ \hline
MYR_6 &:= MY_6 + MZ_7 + d_7M_7 \\
XXR_6 &:= XX_6 - YY_6 + YY_7 + 2d_7MZ_7 + (d_7)^2M_7 \\
ZZR_6 &:= ZZ_6 + YY_7 + 2d_7MZ_7 + (d_7)^2M_7 \\ \hline
XXR_7 &:= XX_7 - YY_7
\end{aligned} \tag{3.39}$$

3.2 Methods of Parameter Identification

This section describes how and which parameters are calculated. Recalling the problem description, the problem of parameter identification is to calculate the robot's parameters from measurements. In this thesis, the measurements from the rotor angles $\mathbf{q}$ and

torques $\boldsymbol{\tau}$ will be used. Naturally, this is an optimization problem because the amount of data from the measurements will be much larger than the number of parameters. Furthermore, because of measurement noise, it is reasonable to expect the model not to fit the measurements exactly. Therefore, the problem of parameter identification is formulated as an optimization problem that computes the best-fitting parameters for the given measurements.

3.2.1 Model Parameter

The aim of this thesis is to identify the dynamic base parameters (3.38) of the KUKA LBR iiwa R820. However, like in [1], for reference also the physical parameters will be calculated. To avoid confusion between these two sets of parameters, they are explained in detail in this section.

The physical parameters, commonly referred to by the vector $\boldsymbol{\mu}$, are all scalar values used in calculating the motor torques $\boldsymbol{\tau}$ that represent the physical properties of the individual links of the robot. This includes for any link i its mass m_i, as well as its center of gravity $\mathbf{r}_i^{c_i}$, the values of its inertia tensor with respect to its center of gravity $\mathbf{I}_i^i$ and potentially some frictions terms. These values represent inherent properties of the individual link, expressed in their own fixed frame of reference. This is opposed to all values derived from the forward kinematics, like the rotation matrices $\mathbf{R}_{n-1}^n$ or the vector of the velocity of the origin of the individual frames of reference $\dot{\mathbf{d}}_i^n$. These values only depend on how the individual links of any robot are connected, but not on the properties of the links, like their shape, material, etc [3].

Because the physical parameters do not depend on the forward kinematics and thus also not on the configuration $\mathbf{q}$ of the robot, they are constant with respect to the frame they are declared in.

For the equations of motion defined in (3.3), the physical parameters $\boldsymbol{\mu}$ of each link i are [1]

$$\boldsymbol{\mu}_i := [m_i, r_{i,x}^{c_i}, r_{i,y}^{c_i}, r_{i,z}^{c_i}, I_{i,xx}^i, I_{i,xy}^i, I_{i,xz}^i, I_{i,yy}^i, I_{i,yz}^i, I_{i,zz}^i, F_{iv}, F_{is}]^{\mathrm{T}} , \tag{3.40}$$

where, as mentioned above, m_i is the mass of link i and $r_{i,x}^{c_i}, r_{i,y}^{c_i}$ and $r_{i,z}^{c_i}$ the coordinates of its center of gravity. Furthermore, the values $I_{i,ab}^i$ are the entries of the inertia tensor of link i, as defined in (2.20) and F_{iv} and F_{is} are the friction terms according to section 3.1.2.
Finally, all seven $\boldsymbol{\mu}_i$ vectors are stacked together to form the $\boldsymbol{\mu}$ vector

$$\boldsymbol{\mu} := [\boldsymbol{\mu}_1^{\mathrm{T}}, \boldsymbol{\mu}_2^{\mathrm{T}}, \ldots, \boldsymbol{\mu}_7^{\mathrm{T}}]^{\mathrm{T}} . \tag{3.41}$$

Besides the physical parameters $\boldsymbol{\mu}$, there are also the related dynamic parameters $\boldsymbol{\pi}$. The dynamic parameters have already been defined in (3.7) as a nonlinear combination of the physical parameters. Therefore, the dynamic parameters can be viewed as a function of

the physical parameters, i. e. $\boldsymbol{\pi}(\boldsymbol{\mu})$. This relationship is even perfectly reversible if the mass is assumed to be nonzero. This assumption holds for any physical link, showing that the physical and dynamic parameters are different representations of the same concept. Hence, the physical parameters are also constant in their defining frame of reference and also only depend on properties inherent to their corresponding link [2], [3].

The main advantage of the dynamic parameters is their defining feature. They linearly factor the equations of motion (3.3) into the form (2.23). This is the case because they transform the calculations of the Lagrangian from the center of gravity to the origin of the frame of the link, which lies in the axis of actuation of the link.

As Section 2.2.7 shows, this reverses some simplifications and turns (2.24) into the more complex form (2.29). However, this complex form makes it easier to see what is going on. Looking at the definition of $\boldsymbol{\beta}$ in (2.33), the product $\beta_{M_i} \cdot M_i$, encompasses the energy of the linear velocity of the origin of the link as well as its potential energy. Moreover, the terms $\beta_{XX_i} \cdot XX_i$ to $\beta_{ZZ_i} \cdot ZZ_i$ cover the rotational energy around the origin. Finally, $\beta_{MX_i} \cdot MX_i$ to $\beta_{MZ_i} \cdot MZ_i$ represent the translational energy difference between the origin and the center of gravity.

Furthermore, by looking at (2.23), it can be seen that the distinction above between properties of the links and of the forward kinematics of the robot is now represented by the math as well. In that equation, ignoring friction, $\mathbf{Y}$ can be interpreted as the influence of the forward kinematics and $\boldsymbol{\pi}$ as the influence of the properties of the links on the torques experienced by the actuation joints [2], [3].

This interpretation stems from the fact that $\mathbf{Y}$ is defined in terms of $\boldsymbol{\beta}$ in (2.37) Considering the definitions of $\boldsymbol{\beta}$ (2.33), $\dot{\mathbf{d}}_i^i$ and $\boldsymbol{\omega}_i^i$ (3.9), $\mathbf{Y}$ is only dependent on the homogeneous transformations between the links and how the axis of actuation are chosen, all values describing how the robot is assembled.

Another interesting property of $\mathbf{Y}$ is that it is upper-block triangular [3]. This means, by defining

$$\mathbf{y}_{ij} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \boldsymbol{\beta}_i}{\partial q_j}\right)^{\mathrm{T}} - \left(\frac{\partial \boldsymbol{\beta}_i}{\partial q_j}\right)^{\mathrm{T}}, \tag{3.42}$$

$\mathbf{Y}$ can be written as

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_{11} & \mathbf{y}_{12} & \cdots & \mathbf{y}_{1j} \\ \mathbf{0} & \mathbf{y}_{22} & \cdots & \mathbf{y}_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{y}_{jj} \end{bmatrix}. \tag{3.43}$$

This can be explained again by looking at $\boldsymbol{\beta}$ because $\boldsymbol{\beta}_i$ is only dependent on q_j if $j \leq i$. Therefore, all $\mathbf{y}_{ij}$ with $j > i$ must be zero. Intuitively, considering the KUKA LBR iiwa R820, one can think that the moment of inertia experienced by joint one varies heavily if the robot stands upright or at a 90-degree angle. However, the moment of inertia of link

seven is always the same, no matter the state of the other joints.

For the KUKA LBR iiwa R820 and the equations of motion defined in (3.3), $\mathbf{Y}$ turns out to be a (84×7) matrix. This is true because each $\boldsymbol{\pi}_i$ vector has twelve entries, and the KUKA LBR iiwa R820 has seven degrees of freedom, which results in a $(7 \cdot 12 \times 7) = (84 \times 7)$ matrix [1].

Finally, there is also the dynamic base parameters or just base parameters $\boldsymbol{\pi}_b$ and their corresponding matrix $\mathbf{Y}_b$, defined in (3.38) and (3.34). As Section 3.1.4 shows in detail, these are all parameters that independently influence the robot's motion. Therefore, these are also all parameters that can be identified separately. In this thesis, $\boldsymbol{\pi}_b$ has 57 elements in contrast to the 84 elements in $\boldsymbol{\pi}$. Note that this means that $\boldsymbol{\pi}$ cannot be uniquely calculated from $\boldsymbol{\pi}_b$.

3.2.2 Incorporate Measurements

To work out how to use measurements to estimate the parameters, recall (3.32). In both forms of the equation, the torques vector $\boldsymbol{\tau}$ at any given moment is linear with respect to the $\boldsymbol{\pi}$ or $\boldsymbol{\pi}_b$ parameters of the robot and the robots configuration and its derivatives $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ at that moment.
Now, let's consider what happens if two measurements of $\boldsymbol{\tau}, \mathbf{q}, \dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are taken at two different moments in time, denoted by subscript 0 and 1. Then $\boldsymbol{\pi}_b$ appears in both equations

$$\boldsymbol{\tau}_0 = \mathbf{Y}_b(\mathbf{q}_0, \dot{\mathbf{q}}_0, \ddot{\mathbf{q}}_0)\boldsymbol{\pi}_b + \boldsymbol{\nu}_0 \tag{3.44}$$

and

$$\boldsymbol{\tau}_1 = \mathbf{Y}_b(\mathbf{q}_1, \dot{\mathbf{q}}_1, \ddot{\mathbf{q}}_1)\boldsymbol{\pi}_b + \boldsymbol{\nu}_1 \,, \tag{3.45}$$

where $\boldsymbol{\nu}$ denotes the random measurement noise vector.

First of all, this means (3.44) and (3.45) can be combined into one equation

$$\begin{bmatrix} \boldsymbol{\tau}_0 \\ \boldsymbol{\tau}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_b(\mathbf{q}_0, \dot{\mathbf{q}}_0, \ddot{\mathbf{q}}_0) \\ \mathbf{Y}_b(\mathbf{q}_1, \dot{\mathbf{q}}_1, \ddot{\mathbf{q}}_1) \end{bmatrix} \boldsymbol{\pi}_b + \begin{bmatrix} \boldsymbol{\nu}_0 \\ \boldsymbol{\nu}_1 \end{bmatrix} . \tag{3.46}$$

This gives a neat way to include multiple measurements into the optimization problem. Further measurements can just be stacked into the equation, just like $\boldsymbol{\tau}_0$ and $\boldsymbol{\tau}_1$ and $\mathbf{Y}_b(\mathbf{q}_0, \dot{\mathbf{q}}_0, \ddot{\mathbf{q}}_0)$ and $\mathbf{Y}_b(\mathbf{q}_1, \dot{\mathbf{q}}_1, \ddot{\mathbf{q}}_1)$ have been stacked together. Therefore, given N measurements, it is useful to define

$$\overline{\boldsymbol{\tau}} := \begin{bmatrix} \boldsymbol{\tau}_0 \\ \boldsymbol{\tau}_1 \\ \vdots \\ \boldsymbol{\tau}_{N-1} \end{bmatrix}, \quad \overline{\mathbf{Y}}_b := \begin{bmatrix} \mathbf{Y}_b(\mathbf{q}_0, \dot{\mathbf{q}}_0, \ddot{\mathbf{q}}_0) \\ \mathbf{Y}_b(\mathbf{q}_1, \dot{\mathbf{q}}_1, \ddot{\mathbf{q}}_1) \\ \vdots \\ \mathbf{Y}_b(\mathbf{q}_{N-1}, \dot{\mathbf{q}}_{N-1}, \ddot{\mathbf{q}}_{N-1}) \end{bmatrix} \quad \text{and} \quad \overline{\boldsymbol{\nu}} := \begin{bmatrix} \boldsymbol{\nu}_0 \\ \boldsymbol{\nu}_1 \\ \vdots \\ \boldsymbol{\nu}_{N-1} \end{bmatrix} . \tag{3.47}$$

Note that the dimensions of the vector still line up because $\overline{\boldsymbol{\tau}}, \overline{\boldsymbol{\nu}} \in \mathbb{R}^{7N \times 1}$, $\overline{\mathbf{Y}}_b \in \mathbb{R}^{7N \times 57}$ and $\boldsymbol{\pi}_b \in \mathbb{R}^{57 \times 1}$. Thus, given N measurements, (3.46) can be generalized into [1]

$$\overline{\boldsymbol{\tau}} = \overline{\mathbf{Y}}_b \boldsymbol{\pi}_b + \overline{\boldsymbol{\nu}} \,. \tag{3.48}$$

Secondly, because the measurement noise is random and not measurable, (3.48) has almost certainly no exact solution for $\boldsymbol{\pi}_b$. Therefore, $\boldsymbol{\pi}_b$ must be estimated by solving the optimization problem

$$\hat{\boldsymbol{\pi}}_b := \min_{\boldsymbol{\pi}_b} \left\| \overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}}_b \boldsymbol{\pi}_b \right\| . \tag{3.49}$$

Moreover, because the measurements from the KUKA LBR iiwa R820 contain a lot of high-frequency noise, they need to be processed before they can be used. This is done according to [1]. There, the same trajectory is measured K times, producing the raw data matrices $\tilde{\mathbf{q}}_{meas}, \tilde{\boldsymbol{\tau}}_{meas} \in \mathbb{R}^{7N \times K}$. Subsequently, the arithmetic mean of each row of $\tilde{\mathbf{q}}_{meas}$ and $\tilde{\boldsymbol{\tau}}_{meas}$ is calculated as [1]

$$\mathbf{q}_{meas} = \left(\frac{1}{K} \sum_{k=1}^{K} \tilde{\mathbf{q}}_{meas,k} \right) \qquad \boldsymbol{\tau}_{meas} = \left(\frac{1}{K} \sum_{k=1}^{K} \tilde{\boldsymbol{\tau}}_{meas,k} \right) ,$$

where $\tilde{\mathbf{q}}_{meas,k}$ and $\tilde{\boldsymbol{\tau}}_{meas,k}$ denote the k^{th} column of $\tilde{\mathbf{q}}_{meas}$ and $\tilde{\boldsymbol{\tau}}_{meas}$ respectively.

Additionally $\mathbf{q}_{meas}$ and $\boldsymbol{\tau}_{meas}$ are also filtered. The joint positions $\mathbf{q}_{meas}$ are filtered with a non-causal zero-phase low-pass IIR Butterworth filter of order 20 and of a cutoff frequency of 3 Hz. The joint torques $\boldsymbol{\tau}_{meas}$ are also filtered with a non-causal zero-phase low-pass IIR Butterworth filter of order 20 but a cutoff frequency of 3.5 Hz. Finally, the joint velocities $\dot{\mathbf{q}}_{meas}$ and accelerations $\ddot{\mathbf{q}}_{meas}$ are estimated from $\mathbf{q}_{meas}$ using a central difference algorithm [1].

These processed vectors $\mathbf{q}_{meas}, \dot{\mathbf{q}}_{meas}, \ddot{\mathbf{q}}_{meas}$ and $\boldsymbol{\tau}_{meas}$ are then used to construct $\overline{\mathbf{Y}}$ and $\overline{\boldsymbol{\tau}}$.

3.2.3 Calculation of the Dynamic Base Parameters

As introduced in Section 2.4.1, the least squares method solves an optimization problem uniquely if the given model is linear and the matrix has full column rank. Fortunately, (3.48) fulfills these requirements, as the column rank of $\overline{\mathbf{Y}}_b$ is the same as the column rank of $\mathbf{Y}_b$, which has full column rank by definition. Therefore, the $\boldsymbol{\pi}_b$ parameters can be calculated using the least squares method. Formally, the identification problem can be formulated as a least squares problem like

$$\hat{\boldsymbol{\pi}}_{b,LS} := \min_{\boldsymbol{\pi}_b} \left\| \overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}}_b \boldsymbol{\pi}_b \right\|_2^2 . \tag{3.50}$$

The solution of (3.49) is

$$\hat{\boldsymbol{\pi}}_{b,LS} = (\overline{\mathbf{Y}}_b^{\mathrm{T}} \overline{\mathbf{Y}}_b)^{-1} \overline{\mathbf{Y}}_b^{\mathrm{T}} \overline{\boldsymbol{\tau}} = \overline{\mathbf{Y}}_b^{\dagger} \overline{\boldsymbol{\tau}} . \tag{3.51}$$

Section 2.4.1 also shows how to calculate the relative standard deviations $\sigma_{\hat{\pi}_j,r\%}$ of the estimated parameters $\hat{\boldsymbol{\pi}}_{b,LS}$. This allows for a rough estimate of how well a given parameter is identified. For example, experience has shown that a parameter can be considered poorly identified if its relative standard deviation is greater than ten times the minimum relative standard deviation [3].

3.2.4 Calculation of the Physical Parameters

Although the least squares method has the advantage of calculating the relative standard deviation of the estimated parameters, it also has some disadvantages. First, it can identify the $\boldsymbol{\pi}_b$ parameter. These parameters can describe the robot's motion perfectly. However, they are very abstract and thus complex, so it is challenging to verify intuitively if there are any problems. Therefore, they might be hard to work with. Moreover, the least squares method does not offer a good way to take constraints into account. One example of a constraint that can be useful in certain situations is the condition of physical feasibility [1].

Therefore, the physical parameters $\boldsymbol{\mu}$ are also explicitly calculated in this thesis. One approach to calculate $\boldsymbol{\mu}$ is to take the already identified base parameters $\hat{\boldsymbol{\pi}}_b$ and to search for a consistent set of $\boldsymbol{\mu}$ parameters using (3.7) and (3.39). However, this approach is not so easy to implement and it is not certain that a solution to that problem exits [9].

This is why another approach is taken. Instead of calculating $\boldsymbol{\mu}$ from $\hat{\boldsymbol{\pi}}_b$, $\boldsymbol{\mu}$ is calculated directly from the measurements $\overline{\boldsymbol{\tau}}$ and $\overline{\mathbf{Y}}$. This is done by modifying the optimization problem (3.49) to be dependent on $\boldsymbol{\mu}$ and adding several boundary conditions:

$$\hat{\boldsymbol{\mu}} := \min_{\boldsymbol{\mu}} f(\boldsymbol{\mu}) = \min_{\boldsymbol{\mu}} \left\| \overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}} \boldsymbol{\pi}(\boldsymbol{\mu}) \right\|_2^2 \tag{3.52a}$$

$$\text{s.t. } \boldsymbol{\mu}_{lb} \leq \boldsymbol{\mu} \leq \boldsymbol{\mu}_{ub} \,, \tag{3.52b}$$

$$m_i > 0 \,, \tag{3.52c}$$

$$\text{eig}(\mathbf{I}_i^i) > 0 \,, \tag{3.52d}$$

$$\mathbf{h}(\boldsymbol{\mu}) \geq 0 \,, \tag{3.52e}$$

$$\forall i \in \{1, \ldots, 7\}$$

as in [1], where $\boldsymbol{\pi}$ is now considered a function with $\boldsymbol{\mu}$ as input, according to (3.7). The lower and upper bound $\boldsymbol{\mu}_{lb}$ and $\boldsymbol{\mu}_{ub}$ in (3.52b) ensure that the estimated parameters $\hat{\boldsymbol{\mu}}$ end up at somewhat realistic values. The values for $\boldsymbol{\mu}_{lb}$ and $\boldsymbol{\mu}_{ub}$ are calculated from already existing estimates of the $\boldsymbol{\mu}$ parameters provided by the starter package for using the KUKA LBR iiwa R820 at ACIN, TU Wien. The boundary conditions (3.52c) and (3.52d) enforce physical feasibility, as they restrain $\hat{\boldsymbol{\mu}}$ only to contain positive masses and positive definite inertia tensors.

The additional inequality boundary conditions $\mathbf{h}$ from (3.52e) restrain the solution $\hat{\boldsymbol{\mu}}$ in such a way that it is physically consistent and that simulations based on $\hat{\boldsymbol{\mu}}$ are numerically stable. The inequalities are mentioned as sanity checks in [10], but their exact form have been set heuristically by [1]. Equation (3.53a) imposes the triangle inequality on the diagonal elements of inertia tensors $\mathbf{I}_i^i$, as this inequality must be satisfied by the inertia tensor of any rigid body, see Section 2.2.5. Additionally, (3.53b) ensures that the differences between the diagonal elements of each inertia tensor are not getting too large. Moreover, (3.53c) restrains the non-diagonal elements of the inertia tensors to stay small compared to the main diagonal elements [1]. Furthermore, (3.53d) impose lower bounds on the diagonal inertia tensor elements.

Finally, (3.53e) and (3.53f) enforce that the smallest diagonal inertia tensor values for links one to five correspond to the axis parallel to the link length [1]. For example, by looking at the schematic of the KUKA LBR iiwa R820, Figure 3.1, it can be seen that link 1 extends along the z-axis of reference frame $\mathbf{O}_1$. Therefore, the diagonal value of $\mathbf{I}_1^1$ corresponding to the z-axis, $I_{1,zz}^1$, should be smaller than $I_{1,xx}^1$ and $I_{1,yy}^1$.

All inequality boundary conditions $\mathbf{h}(\boldsymbol{\mu})$ together are [1]:

$$\forall i \in \{1, \dots, 7\} :$$

$$[I_{i,xx}^i + I_{i,yy}^i, I_{i,xx}^i + I_{i,zz}^i, I_{i,yy}^i + I_{i,zz}^i]^{\mathrm{T}} \geq [I_{i,zz}^i, I_{i,yy}^i, I_{i,xx}^i]^{\mathrm{T}} \,, \tag{3.53a}$$

$$100 \, \min(I_{i,xx}^i, I_{i,yy}^i, I_{i,zz}^i) \geq \max(I_{i,xx}^i, I_{i,yy}^i, I_{i,zz}^i) \,, \tag{3.53b}$$

$$0.1 \, |\min(I_{i,xx}^i, I_{i,yy}^i, I_{i,zz}^i)| \geq \max(|I_{i,xy}^i|, |I_{i,xz}^i|, |I_{i,yz}^i|) \,, \tag{3.53c}$$

$$[I_{i,zz}^i, I_{i,yy}^i, I_{i,xx}^i]^{\mathrm{T}} \geq 10^{-4}[1, 1, 1]^{\mathrm{T}} \,, \tag{3.53d}$$

$$\forall j \in \{1, 3, 5\} :$$

$$\min(I_{j,xx}^j, I_{j,yy}^j) \geq 3 \, I_{j,zz}^j \,, \tag{3.53e}$$

$$\forall k \in \{2, 4\} :$$

$$\min(I_{k,xx}^k, I_{k,zz}^k) \geq 3 \, I_{k,yy}^k \,. \tag{3.53f}$$

Now that the problem is stated, it can be solved by a constrained nonlinear optimization solver. For this thesis, the *fmincon* MATLAB function with the *interior-point* solver has been used. Some solvers work more efficiently if the gradient of the function to be optimized can be calculated. Indeed, for (3.52a) this is possible, because given the observation that

$$\forall \mathbf{x} \in \mathbb{R}^{n \times 1} : \|\mathbf{x}\|_2^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} \,,$$

$f(\boldsymbol{\mu})$, from (3.52a), can be rewritten to

$$f(\boldsymbol{\mu}) = (\overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}}\boldsymbol{\pi}(\boldsymbol{\mu}))^{\mathrm{T}}(\overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}}\boldsymbol{\pi}(\boldsymbol{\mu})) \,,$$

from which the gradient of $f(\boldsymbol{\mu})$ can be calculated as

$$\mathrm{grad}_f(\boldsymbol{\mu}) = (-2)(\overline{\boldsymbol{\tau}} - \overline{\mathbf{Y}}\boldsymbol{\pi}(\boldsymbol{\mu}))^{\mathrm{T}}\overline{\mathbf{Y}}\frac{\partial \boldsymbol{\pi}(\boldsymbol{\mu}')}{\partial \boldsymbol{\mu}'}\Big|_{\boldsymbol{\mu}'=\boldsymbol{\mu}} \,.$$

3.2.5 Calculations

All calculations done in the supplied Maple and MATLAB files can be grouped into three parts.

Part one refers to Section 3.1. It encompasses the definition of the forward kinematics and the equations of motion, as well as the calculation and verification of the matrices $\mathbf{Y}$ and $\mathbf{Y}_b$ and the parameter vector $\boldsymbol{\pi}_b$. An overview of the most essential calculations is depicted in Algorithm 1.

Algorithm 1 [MAPLE] Part 1: Dynamics and Parameters of the KUKA LBR iiwa R820

1: calculate forward kinematics (Section 3.1.1)
2: calculate $\boldsymbol{\beta}$ and $\mathbf{Y}$ (Sections 2.2.7 and 3.1.3)

3: apply and verify base parameter reduction (Section 3.1.4)

4: calculate $\boldsymbol{\tau}(\boldsymbol{\mu})$ from $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{g}$ (Section 3.1.2)
5: transform $\boldsymbol{\tau}(\boldsymbol{\mu}) \rightarrow \boldsymbol{\tau}(\boldsymbol{\pi})$
6: verify $\boldsymbol{\tau}(\boldsymbol{\pi}) = \mathbf{Y}_b\boldsymbol{\pi}_b(\boldsymbol{\pi})$

7: produce MATLAB function func_Y$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$
8: produce MATLAB function func_Y_b$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$
9: produce MATLAB function func_pi$(\boldsymbol{\mu})$
10: produce MATLAB function func_pi_b$(\boldsymbol{\pi})$

Part two is concerned with finding a good excitation trajectory, as described in Section 3.3. Here also the *interior-point* solver of the *fmincon* function in MATLAB is used. However, because the *interior-point* solver only converges to local minima, it is invoked several times to ensure that a sufficiently good local minimum is found. Additionally, each trajectory is tested in simulation for viability.

Algorithm 2 [MATLAB/SIMULINK] Part 2: find suitable excitation trajectory

Require: func_Y_b$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$

1: $\mathbf{x}_0 \leftarrow \text{uniform}\Big([-1.5, 1.5]^{2L}\Big)$
2: find $\mathbf{a}$ and $\mathbf{b}$ (Section 3.3)

3: test trajectory parameters $\mathbf{a}, \mathbf{b}$ in simulation
4: evaluate simulation measurements with algorithm 3
5: return $\mathbf{a}, \mathbf{b}$ and evaluation results

The third and final part of the calculations is the implementation of Section 3.2. It covers the estimation of the base and physical parameters and their verification. Algorithm 3 gives a crude outline of the calculations done.

Algorithm 3 [MATLAB] Part 3: evaluate measurements

Require: func_Y_b($\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$)
Require: func_Y($\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$)
Require: func_pi_b($\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$)
Require: func_pi($\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$)
Require: measurement data

1: process measurement data (Section 3.2.2)
2: calculate $\overline{\mathbf{Y}}$ and $\overline{\boldsymbol{\tau}}$ (Section 3.2.2)

3: estimate $\boldsymbol{\pi}_b$ with least squares (Section 3.2.3)
4: estimate $\boldsymbol{\mu}$ (Section 3.2.4)

5: verify estimations (Section 4.2.1)
6: return $\hat{\boldsymbol{\pi}}_b, \hat{\boldsymbol{\mu}}$ and verification results

3.3 Excitation Trajectory

This section is concerned with finding a good excitation trajectory. Intuitively, it should make sense that the excitation trajectory is very important for the quality of the estimated parameters. That is, if the robot just uses the rotor in link one during the measurement period, probably only the moment of inertia in the direction of actuation of link one will be identified reliably. All other parameters are not excited, which means they do not contribute much to the robot's motion. Therefore, a good trajectory should excite all parameters equally and also consistently. If parameters are only activated for a short period of time, lots of uncertainty will be added by the remaining time, which will also decrease the quality of the estimation [3].

Formally, this notion of a *good trajectory* is hard to define. However, a good way to characterize the quality of a trajectory is to use the condition number of the matrix $\overline{\mathbf{Y}}$ generated from the measurements from that trajectory. That is because the condition number, as introduced in Section 2.5, bounds the maximum error in the solution for a given measurement uncertainty. This means a low condition number trajectory is less sensitive to measurement uncertainties than a high condition number. Naturally, this behavior is expected from a good trajectory. Therefore, finding a good excitation trajectory can again be formulated as the optimization problem in the following [1]

$$\begin{aligned} &\text{out of all trajectories} \quad \mathbf{p} : [0, T] \to \mathbb{R}^n \\ &\qquad\qquad \text{find} \quad \mathbf{p}_{opt} = \min_{\mathbf{p}} \operatorname{cond}(\overline{\mathbf{Y}}([\mathbf{p}(0), \dot{\mathbf{p}}(0), \ddot{\mathbf{p}}(0)], \ldots, [\mathbf{p}(T), \dot{\mathbf{p}}(T), \ddot{\mathbf{p}}(T)])) \,, \end{aligned} \tag{3.54}$$

where $[\mathbf{p}(j), \dot{\mathbf{p}}(j), \ddot{\mathbf{p}}(j)]$ represents the vectors $[\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}]$ according to trajectory $\mathbf{p}$ at the point in time j.

3.3.1 Parameterization of the Excitation trajectory

To solve the optimization problem defined in (3.54), the trajectory is defined as the first $L = 10$ terms of a Fourier series with fundamental angular frequency $\omega_f = \frac{2\pi}{20\,\mathrm{s}}$. This results in a band-limited excitation signal that should not excite any unmodeled aspects in a major way [11]. Because of the value of ω_f, the trajectory will repeat every 20 s. Therefore, to construct $\overline{\mathbf{Y}}_b$, only the first 20 s of the trajectory is considered, and it is sampled with a frequency of 100 Hz. The optimization problem is then formally defined by [1]

$$\min_{a_{i,j}\ b_{i,j}} \ \mathrm{cond}(\overline{\mathbf{Y}}_b) \tag{3.55a}$$

$$\text{s.t. } \forall i \in \{1,\dots,7\},\ \forall l \in \{1,\dots,L\}$$

$$q_i(t) := \sum_{l=1}^{L} \frac{a_{i,l}}{\omega_f l}\sin(\omega_f l t) - \frac{b_{i,l}}{\omega_f l}\cos(\omega_f l t)\,, \tag{3.55b}$$

$$\left[\sum_{l=0}^{L}\frac{b_{i,l}}{l}, \sum_{l=0}^{L} a_{i,l}, \sum_{l=0}^{L} b_{i,l} l\right]^{\mathrm{T}} = \mathbf{0}\,, \tag{3.55c}$$

$$\sum_{l=1}^{L}\frac{1}{l}\sqrt{a_{i,l}^2 + b_{i,l}^2} \le \omega_f q_{i,max}\,, \tag{3.55d}$$

$$\sum_{l=1}^{L}\sqrt{a_{i,l}^2 + b_{i,l}^2} \le \dot{q}_{i,max}\,, \tag{3.55e}$$

$$|[a_{i,l}, b_{i,l}]| \le \min\left(l\frac{\omega_f}{L} q_{i,max}, \dot{q}_{i,max}\right). \tag{3.55f}$$

Here $q_{i,max}$ and $\dot{q}_{i,max}$ define the maximal joint actuator displacement and velocities respectively, according to [12]. (3.55c) forces the trajectory to start with zero acceleration and velocity at the joint position $\mathbf{0}$. Additionally, (3.55d) and (3.55e) ensure that the trajectory does not exceed the robot's motion range and maximal velocity, respectively. Finally, (3.55f) imposes restrictions on the individual Fourier coefficients, such that the trajectory is feasible [1].

A numerical result for (3.55) is obtained by using the fmincon solver in MATLAB. For better variation, the solver is invoked multiple times using different starting values $\mathbf{x}_0 = [\mathbf{a}_0, \mathbf{b}_0]$. These starting values are uniformly sampled from the interval $[-1.5, 1.5]$. The resulting trajectories are then simulated and compared against each other. The trajectory with the best simulation results that is also feasible to use in the lab is then selected.

4 Evaluation

The proposed algorithm is evaluated in two steps. First, a good excitation trajectory is computed according to Algorithm 2. Then, this trajectory is performed in the lab by the KUKA LBR iiwa R820 and $\mathbf{q}$ and $\boldsymbol{\tau}$ are sampled with a frequency of 8 kHz. Subsequently, this measurement data is used to calculate $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ according to Algorithm 3. Finally, the estimated parameters $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ are used in simulation to calculate $\boldsymbol{\tau}$. The simulation results are then compared to the measurements of $\boldsymbol{\tau}$.

4.1 Excitation Trajectory

The trajectory solving (3.55) and giving the best results is depicted in Figure 4.1. It was initially designed for a duration of 20 s, but in reality, the KUKA LBR iiwa R820 was only able to perform it in 27 s without reaching its limits. Therefore, the x-axis of the figure goes from 0 s to 27 s. $\overline{\mathbf{Y}}_b$ constructed from this trajectory has a condition number of 111.7, which is not as low as the condition number obtained in [1], but still remarkably small, considering that $\overline{\mathbf{Y}}_b$ is a $(7 \cdot 8000 \cdot 27 \times 57) = (392000 \times 57)$ matrix.

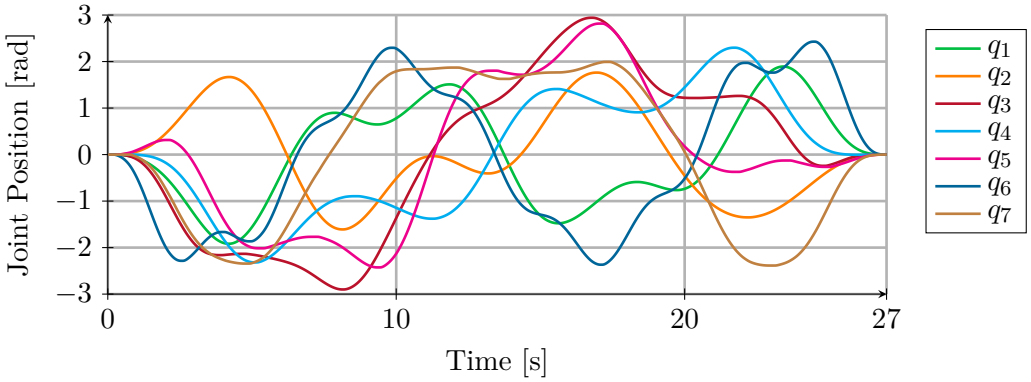

Figure 4.1: Joint Positions for excitation trajectory

Furthermore, the individual Fourier series coefficients are listed in the appendix in Table A.1.

4.2 Parameter Identification

The estimated $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ parameter are listed in Tables 4.1 and 4.2 respectively.

By just looking at the numbers, a few things can be noticed. First of all, the relative standard deviation $\sigma_{\hat{\pi}_j,r\%}$ of $\hat{\boldsymbol{\pi}}_b$ can give a sense of which parameters were excited the most. The only parameters besides MYR_2, ZZR_2 and MYR_4 with a $\sigma_{\hat{\pi}_j,r\%}$ smaller than one are friction terms. This indicates that friction greatly impacts the movement of the KUKA LBR iiwa R820. This also explains the sudden jumps in the torque measurements, Figures 4.2 and 4.3. The friction parameter of $\hat{\boldsymbol{\pi}}_b$ also roughly equal the friction parameter of $\hat{\boldsymbol{\mu}}$.

Secondly the relative standard deviations $\sigma_{\hat{\pi}_j,r\%}$ of most $\boldsymbol{\pi}_b$ parameters are relatively low, $< 10\%$, which means they are in fact identified accurately.

However, the estimated $\boldsymbol{\mu}$ parameters have very odd values for the coordinates of the center of gravity of each link. The outliers are especially significant along the axis towards the following link, e. g., the z-axis for links 1, 3, 5, and 7 and the y-axis for links 2, 4, and 6. This might indicate that the center of gravity of the different links are not sufficiently excited by the trajectory.

Name	Value	$\sigma_{\hat{\pi}_j,r}$ [%]	Name	Value	$\sigma_{\hat{\pi}_j,r}$ [%]	Name	Value	$\sigma_{\hat{\pi}_j,r}$ [%]
ZZR_1	4.276	1.17	FV_3	11.133	0.49	FS_5	6.951	0.3
FV_1	28.404	0.19	FS_3	6.656	0.35	MX_6	-0.03	5.54
FS_1	10.506	0.26	MX_4	-0.037	6.05	MYR_6	-0.12	1.36
MX_2	0.06	4.29	MYR_4	-2.254	0.1	XXR_6	0.352	6.88
MYR_2	-5.8	0.07	XXR_4	2.67	1.66	XY_6	0.028	27.65
XXR_2	4.143	1.73	XY_4	0.001	1819.16	XZ_6	0.056	14.86
XY_2	-0.925	3.74	XZ_4	-0.041	48.4	YZ_6	-0.087	8.46
XZ_2	1.443	3.29	YZ_4	0.146	17.29	ZZR_6	0.48	2.07
YZ_2	0.444	8.69	ZZR_4	1.909	1.64	FV_6	2.75	1.55
ZZR_2	9.572	0.93	FV_4	17.658	0.39	FS_6	3.111	0.85
FV_2	29.136	0.22	FS_4	7.437	0.34	MX_7	-0.003	53.1
FS_2	13.207	0.22	MX_5	0.006	33.01	MY_7	0.048	3.55
MX_3	0.033	9.61	MYR_5	-0.084	2.91	XXR_7	-0.005	206.21
MYR_3	-0.071	4.31	XXR_5	0.079	36.68	XY_7	0.053	10.99
XXR_3	0.669	10.76	XY_5	-0.159	7.59	XZ_7	-0.057	11.52
XY_3	-1.261	2.87	XZ_5	0.022	63.88	YZ_7	-0.154	4
XZ_3	-0.346	7.49	YZ_5	-0.165	8.82	ZZ_7	0.27	5.2
YZ_3	-0.392	7.38	ZZR_5	0.061	49.28	FV_7	2.814	1.59
ZZR_3	0.628	6.36	FV_5	5.953	0.74	FS_7	3.03	0.76

Table 4.1: $\hat{\boldsymbol{\pi}}_b$ parameter

Par.	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	Link 7
m_i	6.714	9.806	3.349	3.38	1.58	0.502	0.327
$r_{i,x}^{c_i}$	−0.581	0.008	0.02	−0.015	−0.005	−0.04	0.05
$r_{i,y}^{c_i}$	−0.201	−0.72	−0.375	−0.476	−0.112	−0.612	0.157
$r_{i,z}^{c_i}$	0	−0.012	−0.572	−0.348	−0.823	−0.174	0.478
$I_{i,xx}^{i}$	0.88	0.748	0.185	0.898	0.03	0.022	0.067
$I_{i,xy}^{i}$	0.254	0	0	0.109	0	0.028	0.007
$I_{i,xz}^{i}$	0.209	0.036	0.016	0.109	0.012	0.012	0
$I_{i,yy}^{i}$	0.883	0.249	0.139	0.224	0.03	0.02	0.114
$I_{i,yz}^{i}$	0.257	0.032	0.024	0.108	0.012	0	0
$I_{i,zz}^{i}$	0.235	0.996	0.046	0.673	0.001	0.041	0.153
F_{iv}	28.791	29.352	10.824	17.643	5.629	2.656	2.884
F_{is}	10.282	13.344	6.738	7.375	7.035	3.169	3.033

Table 4.2: $\hat{\boldsymbol{\mu}}$ parameter

4.2.1 Validation of the identified parameters

Now that $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ have been calculated, the quality of the estimations has to be verified. This is done in the following two sections.

Verifying properties of the Inertia Matrix

As mentioned in Section 2.2.6, the inertia matrix $\mathbf{M}(\mathbf{q})$ is always positive definite for any real robot. In fact, from its definition, if the inertia matrix were not positive definite, that would mean that the robot could have negative kinetic energy. Therefore, any feasible set of parameters must always yield a positive definite inertia matrix.

That this is automatically the case for $\hat{\boldsymbol{\mu}}$ can be seen by looking at (2.24) and the constraints (3.52c) and (3.52d) that $\hat{\boldsymbol{\mu}}$ is subject to. Since $\hat{\boldsymbol{\mu}}$ must have positive masses and positive definite inertia tensors, (2.24) must be positive. However, this is not the case for $\hat{\boldsymbol{\pi}}_b$ because these parameters are calculated without constraints. Therefore $\hat{\boldsymbol{\pi}}_b$ is tested with this method.

The actual expression of the inertia matrix is very complex and depends on the robot configuration $\mathbf{q}$. Therefore, its positive definiteness is not proven mathematically. Instead, it is tested empirically. So, if there is a configuration $\mathbf{q}$, for which the inertia matrix with the estimated $\hat{\boldsymbol{\pi}}_b$ parameters $\mathbf{M}_{\hat{\boldsymbol{\pi}}_b}(\mathbf{q})$ is not positive definite, the estimated $\hat{\boldsymbol{\pi}}_b$ parameters are not feasible.

It has been observed that if the inertia matrix was positive definite along one trajectory solving the trajectory optimization problem (3.55a), it was also reliably positive definite along other trajectories. Therefore, the $\hat{\boldsymbol{\pi}}_b$ parameters of Table 4.1 where tested along the trajectory expressed in Figure A.1 and Table A.2. The inertia matrix evaluated along this trajectory yielded only positive definite matrices, which creates some confidence in $\hat{\boldsymbol{\pi}}_b$.

Comparing simulation results with measurements

The real test for $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ is, of course, how accurate they allow to simulate the motion of the KUKA LBR iiwa R820. The following figures show how well the simulation results match the measured data. The trajectory from Section 4.1 is used for the comparison. The blue line represents the measured torque values processed as described in Section 3.2.2. Of course it is not sufficient to test the obtained parameters only on against the measurements with which they have been calculated with. However, due to a lack of time and resources the parameters are not tested against a different trajectory in this thesis.

To begin with, it can be seen in Figure 4.2 that $\hat{\boldsymbol{\pi}}_b$ and $\hat{\boldsymbol{\mu}}$ produce very similar results. Comparing the torques predicted by the simulation with the measurements, it is noticeable that the graphs do not line up exactly. However, overall the result is quite promising, as the simulation results show behavior similar to the measurements. Additionally, it is noticeable that the simulation results almost precisely predict the jumps due to friction but consistently overestimate oscillations in torques between them.

Figure 4.3 shows the difference between the measurements and the simulation. The blue dotted lines represent $\pm 10\%$ of the difference between the maximal and minimal torque measured for each joint. It can be seen from the plots that the simulation results perform similarly for each joint. However, looking at these plots it is also evident, that although the overall results are promising, there is still room for improvement. The jumps in the torques due to friction are not predicted consistently and the $\pm 10\%$ line is crossed occasionally.

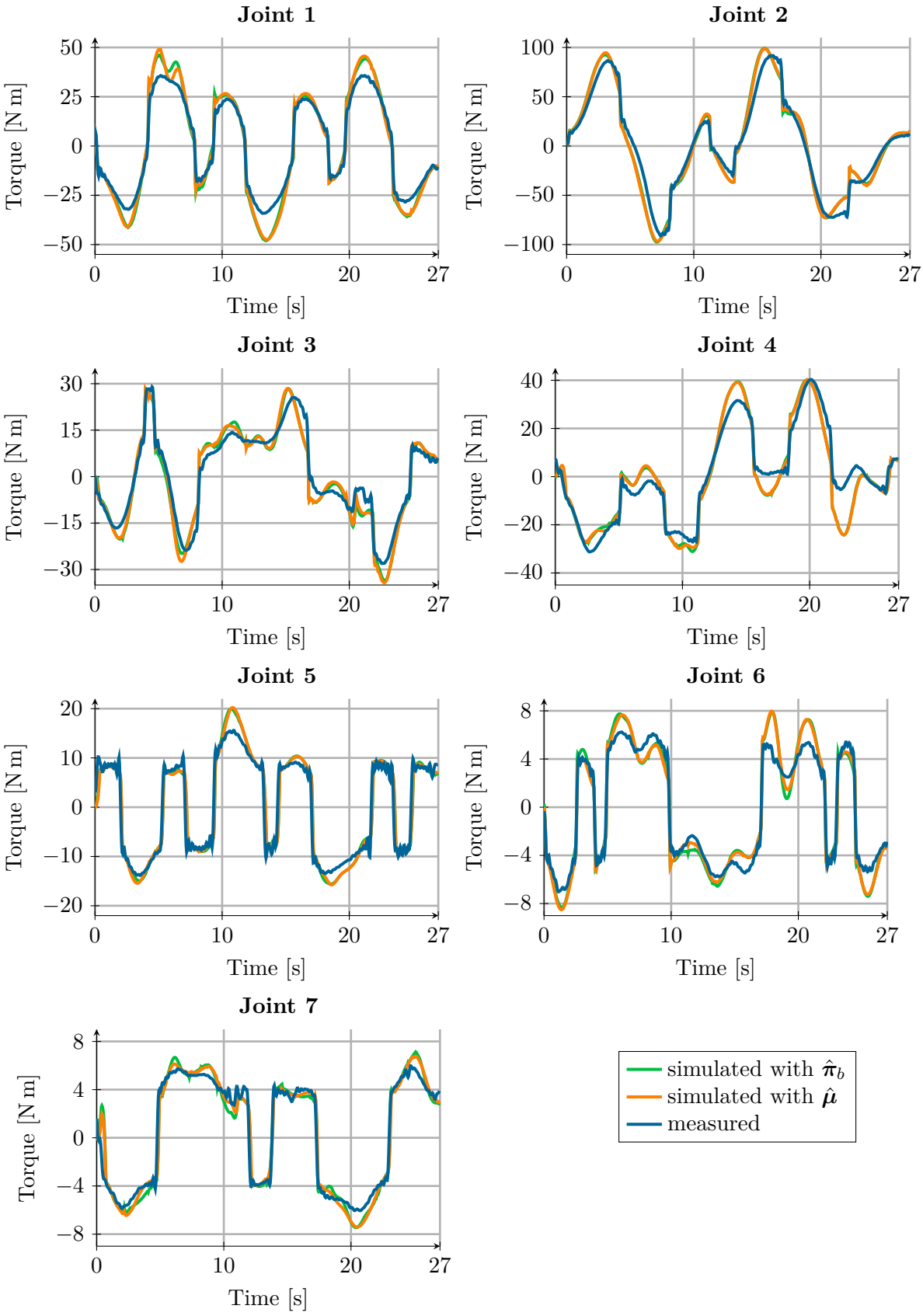

Figure 4.2: Time evolution of measurement and simulation data

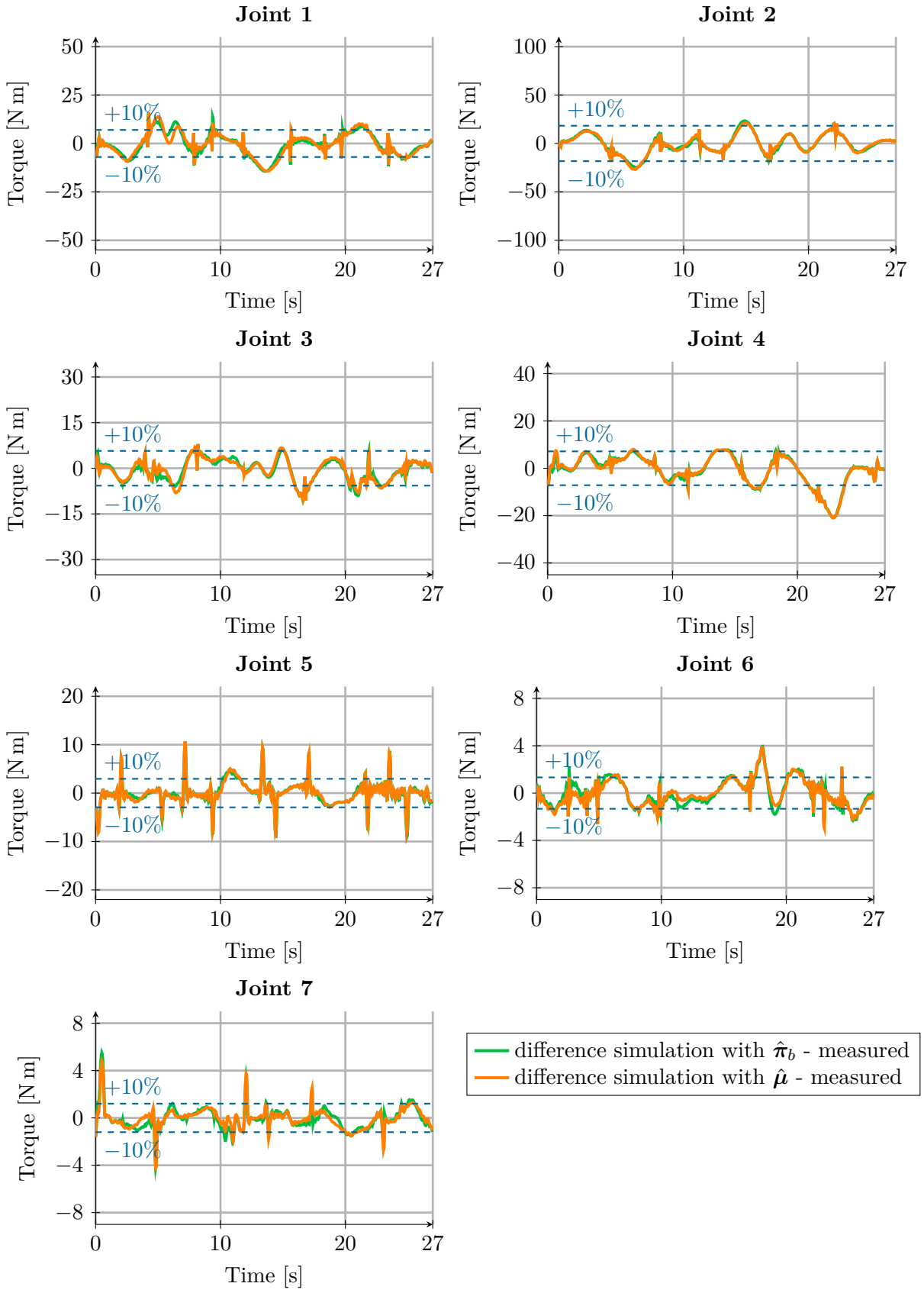

Figure 4.3: Differences between simulation and measurement results

5 Conclusion

In this thesis, the physical and dynamical base parameters of the KUKA LBR iiwa R820 have been estimated. These estimations rely on the key fact that the equations of motion of any robot are linear concerning the dynamical parameters. Based on that, the thesis shows that not all dynamic parameters independently influence the torques acting on the robot's joints. However, from these considerations, a set of parameters that independently influence the robot's motion, called the dynamic base parameters, is derived. Furthermore, a physical interpretation of the different sets of parameters is given.

Moreover, based on [1], it is explained how these parameters can be estimated. This is done by searching for an appropriate trajectory and then formulating a static optimization problem that already existing methods can solve.

The results are promising and can be further improved. They are promising because the estimated parameters allow a rough prediction of the torques acting on the robot's joints, but reliable exact predictions are not possible. One approach to resolve these issues might be to consider a more detailed friction model because friction seems to significantly influence the required torques for the robot used in the experiment. Furthermore, it might be worthwhile to explore how different trajectories influence the quality of the estimations. Maybe a different parameterization of the trajectory yields better results.

Additionally, it should be considered that the estimated parameters have been tested for a very dynamic trajectory close to the limits of the KUKA LBR iiwa R820. Therefore, the estimated parameters might perform well for use cases with slower movements.

A Appendix

The following equation lists all columns of $\mathbf{Y}$, that will be removed in Section 3.1.4.

$$
\begin{aligned}
\mathbf{Y}_{M_1} &= \mathbf{0} \\
\mathbf{Y}_{MX_1} &= \mathbf{0} \\
\mathbf{Y}_{MY_1} &= \mathbf{0} \\
\mathbf{Y}_{MZ_1} &= \mathbf{0} \\
\mathbf{Y}_{XX_1} &= \mathbf{0} \\
\mathbf{Y}_{XY_1} &= \mathbf{0} \\
\mathbf{Y}_{XZ_1} &= \mathbf{0} \\
\mathbf{Y}_{YY_1} &= \mathbf{0} \\
\mathbf{Y}_{YZ_1} &= \mathbf{0} \\
\hline
\mathbf{Y}_{M_2} &= \mathbf{0} \\
\mathbf{Y}_{MZ_2} &= \mathbf{0} \\
\mathbf{Y}_{YY_2} &= \mathbf{Y}_{ZZ_1} - \mathbf{Y}_{XX_2} \\
\hline
\mathbf{Y}_{M_3} &= d_3\mathbf{Y}_{MY_2} + (d_3)^2(\mathbf{Y}_{XX_2} + \mathbf{Y}_{ZZ_2}) \\
\mathbf{Y}_{MZ_3} &= \mathbf{Y}_{MY_2} + 2d_3(\mathbf{Y}_{XX_2} + \mathbf{Y}_{ZZ_2}) \\
\mathbf{Y}_{YY_3} &= \mathbf{Y}_{XX_2} + \mathbf{Y}_{ZZ_2} - \mathbf{Y}_{XX_3} \\
\hline
\mathbf{Y}_{M_4} &= (d_3 + d_4)\mathbf{Y}_{MY_2} + (d_3 + d_4)^2(\mathbf{Y}_{XX_2} + \mathbf{Y}_{ZZ_2}) \\
\mathbf{Y}_{MZ_4} &= -\mathbf{Y}_{MY_3} + d_4\mathbf{Y}_{YZ_3} \\
\mathbf{Y}_{YY_4} &= \mathbf{Y}_{XX_3} + \mathbf{Y}_{ZZ_3} - \mathbf{Y}_{XX_4} \\
\hline
\mathbf{Y}_{M_5} &= \mathbf{Y}_{M_4} + d_5\mathbf{Y}_{MY_4} + (d_5)^2(\mathbf{Y}_{XX_4} + \mathbf{Y}_{ZZ_4}) \\
\mathbf{Y}_{MZ_5} &= \mathbf{Y}_{MY_4} + 2d_5(\mathbf{Y}_{XX_4} + \mathbf{Y}_{ZZ_4}) \\
\mathbf{Y}_{YY_5} &= \mathbf{Y}_{XX_4} + \mathbf{Y}_{ZZ_4} - \mathbf{Y}_{XX_5} \\
\hline
\mathbf{Y}_{M_6} &= \mathbf{Y}_{M_4} + (d_5 + d_6)\mathbf{Y}_{MY_4} + (d_5 + d_6)^2(\mathbf{Y}_{XX_4} + \mathbf{Y}_{ZZ_4}) \\
\mathbf{Y}_{MZ_6} &= -\mathbf{Y}_{MY_5} + d_6\mathbf{Y}_{YZ_5} \\
\mathbf{Y}_{YY_6} &= \mathbf{Y}_{XX_5} + \mathbf{Y}_{ZZ_5} - \mathbf{Y}_{XX_6} \\
\hline
\mathbf{Y}_{M_7} &= \mathbf{Y}_{M_6} + d_7\mathbf{Y}_{MY_6} + (d_7)^2(\mathbf{Y}_{XX_6} + \mathbf{Y}_{ZZ_6}) \\
\mathbf{Y}_{MZ_7} &= \mathbf{Y}_{MY_6} + 2d_7(\mathbf{Y}_{XX_6} + \mathbf{Y}_{ZZ_6}) \\
\mathbf{Y}_{YY_7} &= \mathbf{Y}_{XX_6} + \mathbf{Y}_{ZZ_6} - \mathbf{Y}_{XX_7}
\end{aligned}
\tag{A.1}
$$

	q_1	q_2	q_3	q_4	q_5	q_6	q_7
a_1	-0.00437	0.0045	-0.48108	-0.38535	-0.3115	-0.01783	-0.03516
a_2	-0.62028	0.5327	0.20123	0.00163	0.39644	-0.9791	0.05008
a_3	0.01104	-0.01662	0.01383	-0.01261	-0.03152	0.00268	0.12333
a_4	0.00621	-0.43165	0.01188	0.61809	-0.06543	0.03933	-0.04052
a_5	0.65089	-0.0904	0.30624	-0.00794	0.51263	-0.00515	-0.01269
a_6	-0.0074	-0.00206	-0.00066	-0.06088	-0.30772	0.0109	-0.00299
a_7	-0.01111	-0.00327	-0.00206	-0.14301	-0.00729	0.04044	-0.01498
a_8	-0.00142	0.00137	0.00002	-0.0073	-0.06771	0.16152	-0.01108
a_9	-0.00244	0.00523	-0.02635	-0.00264	-0.11498	0.71862	-0.01839
a_{10}	-0.02111	0.00021	-0.02305	0	-0.00291	0.02859	-0.0376

	q_1	q_2	q_3	q_4	q_5	q_6	q_7
b_1	0.0056	0.01689	0.19887	0.01563	0.17841	-0.01337	0.4156
b_2	-0.06165	-0.11657	-0.30741	-0.00663	-0.44345	0.01059	-0.36439
b_3	-0.01263	-0.01822	0.01997	-0.0001	0.0028	0.01044	-0.72843
b_4	0.00086	0.40579	0.00432	-0.07209	0.00087	0.02059	-0.12088
b_5	0.20832	-0.27007	-0.41027	0.00076	0.18801	-0.00293	0.02384
b_6	0.00879	0.00209	0.00581	0.0001	0.09186	0.00116	0.00527
b_7	-0.04198	-0.00198	0.02527	0.03555	-0.0092	0.01342	0.04237
b_8	-0.02403	-0.00112	0.01892	0.0034	-0.01634	0.02934	0.02719
b_9	-0.01637	0.00135	0.09176	0.00065	-0.06588	-0.06231	0.06407
b_{10}	-0.03087	-0.00038	0.12011	0	-0.00066	0.01184	0.17404

Table A.1: Fourier coefficients of the used trajectory

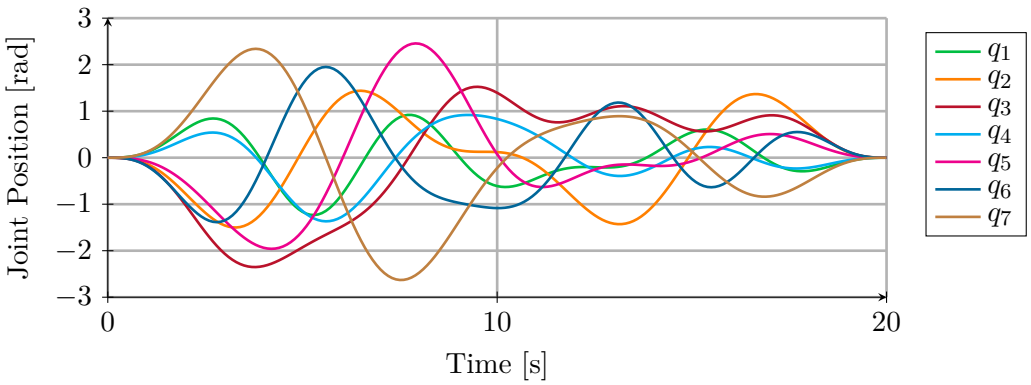

Figure A.1: Joint Positions for the feasibility test trajectory

	q_1	q_2	q_3	q_4	q_5	q_6	q_7
a_1	−0.01473	0.0016	−0.40337	−0.04114	−0.05262	0.00011	0.00855
a_2	0.00018	−0.69745	−0.18612	0.05967	−0.62028	−0.12766	0.87478
a_3	0.51786	−0.00037	0.22333	0.36807	0.59427	−0.59839	−0.48185
a_4	−0.09658	0.67202	−0.04078	−0.1675	0.00431	0.12273	−0.39893
a_5	−0.40672	0.0242	0.40695	−0.2191	0.07432	0.60321	−0.00254

	q_1	q_2	q_3	q_4	q_5	q_6	q_7
b_1	−0.04209	0.00182	0.18905	0.03977	0.14574	−0.0001	−0.15416
b_2	−0.00097	0.02406	−0.42849	−0.28353	−0.27149	0.34643	0.24615
b_3	0.02858	−0.00341	0.00112	0.36068	−0.3306	−0.67652	0.35796
b_4	0.38574	−0.12365	−0.01963	0.04395	0.49598	−0.01205	−0.35288
b_5	−0.31694	0.09097	0.14861	−0.14611	−0.11898	0.27699	−0.0001

Table A.2: Fourier coefficients of the feasibility test trajectory

Bibliography

[1] Y. R. Stürz, L. M. Affolter, and R. S. Smith, "Parameter identification of the kuka lbr iiwa robot including constraints on physical feasibility," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6863–6868, 2017.

[2] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics - Modelling, Planning and Control*. London, UK: Springer-Verlag, 2009.

[3] W. Khalil and E. Dombre, *Modeling, Identification & Control of Robots*. Florida, US: CRC Press, 2012.

[4] T. Xu *et al.*, "Dynamic identification of the KUKA LBR iiwa robot with retrieval of physical parameters using global optimization," *IEEE Access*, vol. 8, pp. 108 018–108 031, 2020.

[5] K. M. Lynch and F. C. Park, *Modern Robotics - Mechanics, Planning, and Control*. Cambridge, UK: Cambridge University Press, 2017, ISBN: 9781107156302.

[6] B. Chachuat, "Nonlinear and dynamic optimization: From theory to practice," 2007, Laboratoired'Automatique, École Polytechnique Fédérale de Lausanne, (accessed 02.05.2024). [Online]. Available: `http://infoscience.epfl.ch/record/111939`.

[7] M. Drmota, B. Gittenberger, G. Karigl, and A. Panholzer, *Mathematik für Informatik*. Berlin, Germany: Heldermann Verlag, 2007, ISBN: 9783885381174.

[8] Fast Swing-Up Trajectory Optimization for a Spherical Pendulum on a 7-DoF Collaborative Robot, (visited on 07.05.2024). [Online]. Available: `https://ieeexplore.ieee.org/abstract/document/9561093`.

[9] N. F. Vicente Mata Francesc Benimeli and A. Valera, "Dynamic parameter identification in industrial robots considering physical feasibility," *Advanced Robotics*, vol. 19, no. 1, pp. 101–119, 2005.

[10] gazebosim.org, (visited on 07.05.2024). [Online]. Available: `https://classic.gazebosim.org/tutorials?tut=inertia&cat=build_robot`.

[11] V. Bargsten, P. Zometa, and R. Findeisen, "Modeling, parameter identification and model-based control of a lightweight robotic manipulator," in *Proceedings of the International Conference on Control Applications (CCA)*, 2013, pp. 134–139.

[12] KUKA LBR iiwa 14 R820, Basic Data, (visited on 09.05.2024). [Online]. Available: `https://www.kuka.com/-/media/kuka-downloads/imported/8350ff3ca11642998dbdc81dcc2ed44c/0000246833_en.pdf`.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt wurde. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Wien, Mai 2024

Simon Halper