Dynamics Calibration of a Redundant Flexible Joint Robot based on Gyroscopes and Encoders

Dennis Schüthe[1], Felix Wenk[2] and Udo Frese[1]
[1]*Multi-Sensor Interactive Systems Group, University of Bremen, Enrique-Schmidt-Str. 5, Bremen, Germany*
[2]*CPS, German Research Center for Artificial Intelligence (DFKI), Enrique-Schmidt-Str. 5, Bremen, Germany*

Keywords: Parameter Identification, Dynamics Identification, Model Fitting, Flexible Joints, Least-square, Entertainment Robot.

Abstract: In this paper we show the identification of the dynamic parameters of a redundant flexible joint robot with a flexible bearing in the first joint. The bearing leads to distortions of the link velocities measured by gyroscopes. The bearing flexibility is not modeled explicitly, resulting in a very simple model. For the joint positions encoders are used. We show how to pack the calibration problem into the sparse least-squares on manifolds toolkit (SLOM) and present the results. For verification we compare the measured data with the predicted data computed using the identified dynamic parameters.

1 INTRODUCTION

For model predictive control it is essential to have a good model. One question is the qualitative structure of the model, i.e. the decision which of the countless effects happening in reality are modeled and which of them are ignored. This is usually answered by expert knowledge, i.e. experience and experiment. The other question is to determine the quantitative parameters governing these modeled effects, which need to be determined by a calibration procedure.

In this paper we are concerned with a specialized ball playing 3-DOF entertainment robot, which has on purpose a very low-cost design (Figure 1) being built in large parts from laser cut acrylic. This design comes with some mechanical problems, most notably a high friction and high elasticity almost everywhere in the structure and actuator system.

We want to illustrate in this paper the calibration of an elastic joint robot model to this machine using motor encoders and gyroscopes. The encoders are placed on every motor, i.e. every joint. But, not every joint has a gyroscope. Thus, the data has to be extracted from the sensors for each joint.

We also show that the obtained model is reasonably precise even though the robot's elasticity is not only located in the joints, as assumed by the model, but also in the whole structure. Most notably the effect is recognized in the first bearing connecting the robot's body to the base.

1.1 System

The system under calibration and parameter finding is a 2.1 m tall ball playing entertainment robot named "Doggy" (Figure 1). Its task is to hit balls with its head. The head is made of a 40 cm Styrofoam sphere and holds one of our two IMU sensors. Moreover, the heads orientation does not matter when playing a ball – as it is a sphere, so only the position and velocity matter. The head is connected to the robot's three revolute jointed system via a carbon rod. The first axis acts like a hip and gives the robot a redundant degree of freedom (DOF). The redundancy is a result of the common intersection point of all axes (Figure 1). Therefore, the end-effector (EOF) moves on a partial sphere (due to joint limits) and the two DOF are controlled by three redundant joints. A stereo camera system for ball tracking is placed after the first joint to turn with the robot. Additionally, our circuit board with a microcontroller, motor drivers and another IMU sensor is placed on the first body.

Each joint is driven by a DC motor using a tooth belt as coupling. As the tooth belt is not totally stiff, we get elasticity between the motor and the link. Each motor has an encoder for the positioning, read by the microcontroller.

The material used to build our robot is mostly acrylic glass (roughly 80%) and the base plate is made of steel. The rest is made of aluminum (Figure 1).

Schüthe, D., Wenk, F. and Frese, U.
Dynamics Calibration of a Redundant Flexible Joint Robot based on Gyroscopes and Encoders.
DOI: 10.5220/0005976603350346
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 335-346
ISBN: 978-989-758-198-4

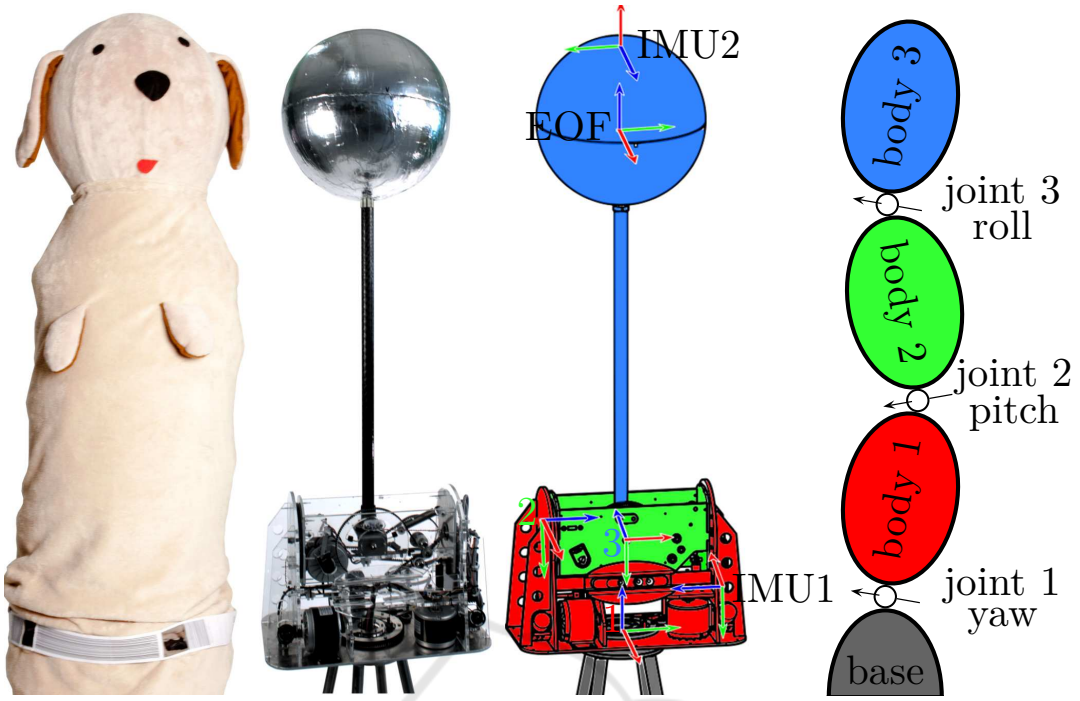


Figure 1: "Doggy" the ball playing entertainment robot. From left to right: The costume of the robot, its interior, the coordinate systems with color encoding (the world coordinate frame is on the ground in the same orientation as 1 in its initial position), and the kinematic chain. The video (http://www.informatik.uni-bremen.de/agebv2/downloads/videos/schuetheIcinco16_calibrationMotion.mp4) shows the movement of the calibration.

1.2 Problem Statement

Our goal is to calibrate the parameters of the robot's dynamics to use them in a following work for model predictive control (MPC) which we already developed in simulation (Schüthe and Frese, 2015). For such an MPC we need the state $\boldsymbol{x}$ of our robotic system and the dynamic function with known parameters. The states are not measurable directly. Thus, we will have to build an observer, e.g. a Kalman-Filter, to estimate the states using the dynamic model. Moreover, the aim is to control the robot using an optimal controller which needs the model of the controlled system as well. For both we need the dynamics including its parameters.

Some of the dynamic parameters are known a priori, such as the inertia of the robot, which is pretty well known from the CAD model. Other parameters, like the motor inertia b, the motors mutual induction constant k_{mi} and the voltage to torque constant k_{pwm} are known only approximately, as there is only few information on the used low-cost motor. However, some parameters are not known and can only be guessed. These are the motor friction $\boldsymbol{\tau}_{\text{fm}}$, the spring constant $\boldsymbol{k}_{\text{s}}$, and spring damping constant $\boldsymbol{d}_{\text{s}}$.

A major challenge in the calibration is the first bearing, between base and first body. It is a ball bearing with some kind of flexibility, especially when moving joint 2 and 3 (see link of Figure 1). So the IMUs will measure the movements of the motors added with – we could say – "noise", which is the result of the flexible bearing.

To solve the calibration problem we use the motor encoders' and the gyroscopes' data, with a 1 kHz sampling frequency. We explicitly relinquish of the accelerometer data to show that the robots parameters can be calibrated without accelerometers. Moreover, our upcoming estimator should work on gyroscope data only and we want to see if it works also for the calibration.

1.3 Contribution

We show that the robots' parameters can be obtained by using encoders and only gyroscopes for measurements. We prove that it is also possible to do this if there is a large error on the gyroscopes' data. To our knowledge the estimation of the link position and velocity has never been done by using gyroscope data only with additional information of encoders. Moreover, we explain how to do such a calibration by using the sparse least-squares on manifold (SLOM) toolkit and that it is very straightforward to implement.

2 RELATED WORK

The options for identifying dynamic parameters are summarized in the overview paper by (Wu et al., 2010), where the authors distinguish on-line and off-line identification. The former deals with adaptive control algorithms and neural networks. In adaptive control the parameters are adjusted and estimated on-line for using them in the controller. Neural networks are effectively identifying nonlinear systems. The parameters are used as weights for the network and are approached by training the weights in real-time (Narendra and Parthasarathy, 1990). The latter, off-line identification, has three methods to name: (1.) Physical experiments, such as isolating each link and measure its mass, the inertia and center of gravity (COG). (2.) Computer aided design (CAD) techniques can be used for inertia and COG measuring. But other parts, such as the identification of friction or the motor parameters, are not possible. (3.) The third possibility is the identification by analyzing the input/output behavior on a well known motion and minimizing the difference of the real data and the estimated data of the model. In our identification we make use of (2.) and (3.) in combination.

In (Grotjahn et al., 2001) and (Vuong and Ang Jr, 2009) the identification of dynamic parameters of industrial robots is presented. In (Grotjahn et al., 2001) the identification does not require the friction model a priori. Vuong et al. have to deal with highly nonlinear friction and build a model which overcomes this problem. They also demonstrate the identification using an off-line method with a least-squares algorithm, which is enhanced by constraints to adjust the least-squares result (Mata et al., 2005). Both have a model for a rigid body robot, so there is no flexibility.

The identification with flexible joints is discussed and demonstrated in (Moberg, 2010), where also the friction is estimated. Additionally, the model is enhanced by flexible links. To identify the parameters a least-squares off-line method is used.

How accelerometers and gyroscopes can be utilized to obtain the state of flexible robots and use them for control was shown in (Staufer and Gattringer, 2012; Cheng and Oelmann, 2010). However, there is a combination of both IMU sensors to estimate the state. In our case we only want to use the gyroscope.

In the identification process the least-squares solver takes a big role. Also for our identification process we make use of a least-squares solver packed into a framework called SLOM (Hertzberg, 2008), which can be used for mapping problems, but also for calibration and sensor fusion (Hertzberg et al., 2012).

3 MODEL

To identify the parameters for the dynamics we need an appropriate dynamic model. It is also necessary to know the kinematics for transformations from one coordinate system to the other. This is needed for the measurement functions, which will be explained later.

3.1 Kinematics

With the Kinematics we describe the relations of coordinate systems. Each joint and each IMU has one coordinate system. Additionally, there are the world coordinate system and the end-effector (EOF) coordinate system. We can transform from one system to another using translations and rotations. In our case we only need to transform velocities to create a relation between angular motor velocities $\dot{\boldsymbol{\theta}}$ and gyroscope measurements $\boldsymbol{\omega}$. Only rotations of the form ${}^{to}_{from}\boldsymbol{R}$ are needed. A rotation of the z-axes moves the correlated body together with its coordinate system (Figure 1). The rotations – ${}^{IMU1}_{1}\boldsymbol{R}$ and ${}^{IMU2}_{EOF}\boldsymbol{R}$ – should also be identified by our calibration, as they could be misaligned by a few degrees.

3.2 Dynamics - Elastic Joint Model

As mentioned before, we need to take into account the elasticity between motor and link due to the tooth belt coupling. This dynamic model is well known and described in (Siciliano and Khatib, 2008; Albu-Schaeffer et al., 2007) as

$$\begin{aligned}\mathbf{0} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{\tau}_g(\boldsymbol{q}) + \\ \boldsymbol{K}_\mathrm{s}\left(\boldsymbol{q}-\boldsymbol{\theta}\right) + \boldsymbol{D}_\mathrm{s}\left(\dot{\boldsymbol{q}}-\dot{\boldsymbol{\theta}}\right) + \boldsymbol{\tau}_\mathrm{fl}\end{aligned} \tag{1}$$

$$\boldsymbol{\tau}_\mathrm{m} = \boldsymbol{B}\ddot{\boldsymbol{\theta}} + \boldsymbol{K}_\mathrm{s}\left(\boldsymbol{\theta}-\boldsymbol{q}\right) + \boldsymbol{D}_\mathrm{s}\left(\dot{\boldsymbol{\theta}}-\dot{\boldsymbol{q}}\right) + \boldsymbol{\tau}_\mathrm{fm}\ , \tag{2}$$

where motor angle, velocity and acceleration are $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, and $\ddot{\boldsymbol{\theta}}$ respectively. $\boldsymbol{M}(\boldsymbol{q})$ is the link inertia, the Coriolis and centrifugal terms are represented by $\boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}})$, $\boldsymbol{\tau}_g(\boldsymbol{q})$ holds the gravitational terms. The coupling between motor and link is approximated by a spring with stiffness $\boldsymbol{K}_\mathrm{s} = \mathrm{diag}(\boldsymbol{k}_\mathrm{s})$ and damping $\boldsymbol{D}_\mathrm{s} = \mathrm{diag}(\boldsymbol{d}_\mathrm{s})$. The friction on link and motor side is denoted as $\boldsymbol{\tau}_\mathrm{fl}$ and $\boldsymbol{\tau}_\mathrm{fm}$. The motor torque $\boldsymbol{\tau}_\mathrm{m}$ and the motor inertia $\boldsymbol{B} = b\boldsymbol{I}$ ($\boldsymbol{I}$ is the unit matrix) have to be transformed to the link side by the gear ratio $n_G = 42.092$.

$$\boldsymbol{B} = n_G^2\mathbf{B}_{motor} \tag{3}$$

$$\boldsymbol{\tau}_\mathrm{m} = n_G\tau_{motor} \tag{4}$$

A common way to simplify (2) is to neglect the Coriolis and centrifugal terms, which are hardly noticeable in our system. We can also neglect the gravitational terms $\boldsymbol{\tau}_g(\boldsymbol{q})$ as we inserted springs that counteract the gravitational force on the pitch and yaw

axes, i.e. axes 2 and 3. Thus, we can position the robot anywhere in the working space and it holds its position without any torque input. Additionally, we neglect the link friction $\boldsymbol{\tau}_{\mathrm{fl}}$ like in (Moberg, 2010). In some test cases we have seen that the link friction is close to zero. We assume that the motor parameters k_{pwm}, k_{mi}, and b are identical, because the motors are of the same type. For the spring parameters each element i of the vectors $\boldsymbol{k}_{\mathrm{s}}$ and $\boldsymbol{d}_{\mathrm{s}}$ represents the specific stiffness and damping of the i-th joint. The simplified dynamics

$$\mathbf{0} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{\tau}_{\mathrm{c}} \tag{5}$$

$$\boldsymbol{\tau}_{\mathrm{c}} = \mathrm{diag}(\boldsymbol{k}_{\mathrm{s}})\left(\boldsymbol{q} - \boldsymbol{\theta}\right) + \mathrm{diag}(\boldsymbol{d}_{\mathrm{s}})\left(\dot{\boldsymbol{q}} - \dot{\boldsymbol{\theta}}\right) \tag{6}$$

$$\mathbf{0} = b\ddot{\boldsymbol{\theta}} - \boldsymbol{\tau}_{\mathrm{c}} + \boldsymbol{\tau}_{\mathrm{fm}} - \boldsymbol{\tau}_{\mathrm{m}}(\boldsymbol{u}, \dot{\boldsymbol{\theta}}) \tag{7}$$

can be rearranged to formulate a dynamical state space representation, where the highest derivative is placed on the left hand: $\dot{\boldsymbol{x}} = f_{\mathrm{dyn}}(\boldsymbol{x}, \boldsymbol{u})$, with the state

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{q}^T & \dot{\boldsymbol{q}}^T & \boldsymbol{\theta}^T & \dot{\boldsymbol{\theta}}^T \end{pmatrix}^T \quad . \tag{8}$$

The input vector defines a pwm (pulse width modulation) signal of the voltage passed to the DC motor. $\boldsymbol{\tau}_{\mathrm{m}}$ is the torque to the system as result of the voltage. The transfer function $f_{\mathrm{dyn}}(\boldsymbol{x}, \boldsymbol{u})$ is

$$f_{\mathrm{dyn}}(\boldsymbol{x}, \boldsymbol{u}) = \begin{pmatrix} \dot{\boldsymbol{q}} \\ -\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{\tau}_{\mathrm{c}} \\ \dot{\boldsymbol{\theta}} \\ (b\boldsymbol{I})^{-1}\left(\boldsymbol{\tau}_{\mathrm{c}} + \boldsymbol{\tau}_{\mathrm{m}}(\boldsymbol{u}, \dot{\boldsymbol{\theta}}) - \boldsymbol{\tau}_{\mathrm{fm}}\right) \end{pmatrix} \quad . \tag{9}$$

Link Inertia. The link moment of inertia $\boldsymbol{M}(\boldsymbol{q})$ is the computation result of Matlab's Spatial_V2 library (Featherstone, 2012). We fed it with CAD model information of each joint, i.e. its COG inertia, its mass, and its displacement between axis coordinate system and COG.

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{1,2} & m_{2,2} & m_{2,3} \\ m_{1,3} & m_{2,3} & m_{3,3} \end{pmatrix} \tag{10}$$

with the matrix entries

$$\begin{aligned} m_{1,1} &= 1.824\mathrm{c}_2\mathrm{s}_2 - 2.975\mathrm{c}_2^2 - 503.760\mathrm{c}_2^2\mathrm{c}_3^2 + \\ &\quad 0.106\mathrm{c}_2^2\mathrm{c}_3\mathrm{s}_3 + 0.183\mathrm{c}_2\mathrm{c}_3\mathrm{s}_2 - 0.138\mathrm{c}_2\mathrm{s}_2\mathrm{s}_3 + \\ &\quad 1012.973 \\ m_{1,2} &= 0.8192\mathrm{s}_2 - 8.5760\mathrm{c}_2 + 0.0691\mathrm{c}_3\mathrm{s}_2 + \\ &\quad 0.0916\mathrm{s}_2\mathrm{s}_3 - 0.1062\mathrm{c}_2\mathrm{c}_3^2 - 503.7604\mathrm{c}_2\mathrm{c}_3\mathrm{s}_3 \\ m_{1,3} &= 514.2253\mathrm{s}_2 + 0.0916\mathrm{c}_2\mathrm{c}_3 - 0.0691\mathrm{c}_2\mathrm{s}_3 \\ m_{2,2} &= 503.7604\mathrm{c}_3^2 - 0.0531\sin(2 * q_3) + 40.0367 \\ m_{2,3} &= 0.0691\mathrm{c}_3 + 0.0916\mathrm{s}_3 \\ m_{3,3} &= 514.2253 \quad . \end{aligned}$$

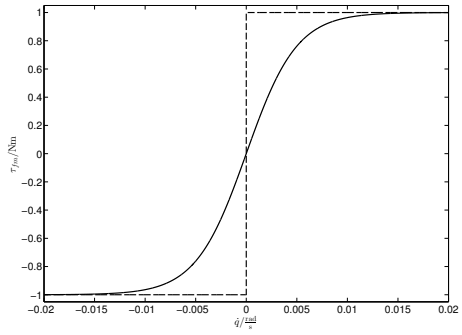


Figure 2: Coulomb friction as Signum function (dashed line) and approximated Sigmoid function (solid line).

The inertia is given in gm^2. The inertia depends on the i-th link position of the time step k with $\mathrm{c}_i = \cos(q_{k,i})$ and $\mathrm{s}_i = \sin(q_{k,i})$.

Motor Torque $\boldsymbol{\tau}_{\mathrm{m}}$. The torque is given by the pwm Signal $\boldsymbol{u}$, the maximum voltage of the power supply $\mathrm{U}_{\mathrm{max}} = 32\,\mathrm{V}$, and a gain k_{pwm} which translates the voltage to a torque. This torque is countered by a mutual induction (value of k_{mi}) and depends on the motor velocity $\dot{\boldsymbol{\theta}}$, which is the second part of the equation.

$$\boldsymbol{\tau}_{\mathrm{m}}(\boldsymbol{u}, \dot{\boldsymbol{\theta}}) = k_{\mathrm{pwm}}U_{\mathrm{max}}\boldsymbol{I}\boldsymbol{u} - k_{\mathrm{mi}}\boldsymbol{I}\dot{\boldsymbol{\theta}} \tag{11}$$

Motor Friction $\boldsymbol{\tau}_{\mathrm{fm}}$. The motor friction consists of different parts (Olsson et al., 1998). First, the static friction which holds the joint at a position until a force higher than this friction is produced. As our goal is to control the robot mostly in motion, we neglect this friction. Secondly, there is viscous friction, which increases proportionally to the velocity. We can also neglect that as we use a tooth belt. Finally, there is the kinetic or Coulomb friction acting at a constant rate over all velocities. This is typically described as $\boldsymbol{\tau}_{\mathrm{fm}} = \mathrm{diag}(\boldsymbol{\mu}_{\mathrm{fm}})\,\mathrm{sgn}(\dot{\boldsymbol{q}})$, where $\boldsymbol{\tau}_{\mathrm{fm}}$ is the friction torque and $\boldsymbol{\mu}_{\mathrm{fm}}$ the kinetic friction coefficient. The Signum function (sgn) translates the velocity $\dot{\boldsymbol{q}}$ to a constant value over all velocities depending on its sign. The disadvantage of this model is the discontinuity of the sgn-function – this is unwanted in simulations, e.g. when using an ODE-Solver. We approximated the Signum function by a Sigmoid function

$$\boldsymbol{\tau}_{\mathrm{fm}} = 2\,\mathrm{diag}(\boldsymbol{\mu}_{\mathrm{fm}}) \begin{pmatrix} \frac{1}{1+\exp(-400\dot{q}_1)} - 0.5 \\ \frac{1}{1+\exp(-400\dot{q}_2)} - 0.5 \\ \frac{1}{1+\exp(-400\dot{q}_3)} - 0.5 \end{pmatrix} \quad . \tag{12}$$

This differs from the sgn function only close to zero velocity (Figure 2). Our calibration goal is to get $\boldsymbol{\mu}_{\mathrm{fm}}$.

Not Modeled. In our model we leave out the vibration and movement of the first axis around its x- and y-axes, which is normally the case in the joint's turning definition. But in our case the bearing of Axis 1 allows little movements around those axes due to its flexibility. Leading to a deviation between model and real world. We want to see, if it is possible to find parameters that still fit the model to the real behavior. Therefore, we expect that movements around x_1- and y_1-axes will be put into the springs and dampings of axes 2 and 3.

The motivation for this simplification is: First, that in general the simplest model that works is the best model. Second, that incorporating the elasticity in the Axis 1 bearing directly would require four additional states, namely position and velocity for 2-DOF of elastic movement.

4 MEASUREMENT FUNCTIONS

To determine the parameters governing the behavior of the robot, we search for the parameters which best explain sensor measurements taken while exercising the degrees of freedom of the robot during a calibration motion. I.e. we search for the parameters $\boldsymbol{\vartheta}$ which result in the least squared difference of the actual measurements and the measurements predicted from the parameters. Formally, we search the least-squares estimate $\hat{\boldsymbol{\vartheta}}$

$$\hat{\boldsymbol{\vartheta}} = \arg\min_{\boldsymbol{\vartheta}} \frac{1}{2} \|F(\boldsymbol{Z}, \boldsymbol{\vartheta})\|_{\Sigma}^2 \ , \tag{13}$$

where $\boldsymbol{Z}$ is the vector of all stacked measurements and Σ the covariance of the measurement error calculated by F. F computes the difference between actual and predicted (using the parameters) measurements.

4.1 Calibration Parameters

$\boldsymbol{\vartheta}$ consists of the time-invariant parameters $\boldsymbol{\vartheta}_{\text{calib}}$, in which we are interested primarily, and the parameters $\boldsymbol{\vartheta}_{\text{state}}$ describing the state of the robot during the calibration motion at each sensor sampling time t_k. For N samples we have

$$\boldsymbol{\vartheta} = \begin{pmatrix} \boldsymbol{\vartheta}_{\text{calib}} \\ \boldsymbol{\vartheta}_{\text{state}} \end{pmatrix} \text{ with } \boldsymbol{\vartheta}_{\text{state}} = \begin{pmatrix} \boldsymbol{\vartheta}_{\text{state}\,1} \\ \vdots \\ \boldsymbol{\vartheta}_{\text{state}\,N} \end{pmatrix} \ . \tag{14}$$

$\boldsymbol{\vartheta}_{\text{state}\,k}$ is the state vector as defined in (8) at time t_k and the concrete parameters to be calibrated are

$$\begin{aligned} \boldsymbol{\vartheta}_{\text{calib}} = (&k_{\text{pwm}} \quad k_{\text{mi}} \quad b \quad \boldsymbol{\mu}_{\text{fm}}^T \quad \boldsymbol{k}_{\text{s}}^T \quad \boldsymbol{d}_{\text{s}}^T \\ &\boldsymbol{\omega}_{1,0}^T \quad \boldsymbol{\omega}_{2,0}^T \quad {}^{IMU1}_{1}\boldsymbol{R}^T \quad {}^{IMU2}_{EOF}\boldsymbol{R}^T)^T \end{aligned} \tag{15}$$

In addition to model quantities, $\boldsymbol{\vartheta}_{\text{calib}}$ contains parameters pertaining to the sensors used. These are the rotation of the end-effector frame relative to the IMU mounted on the end-effector (${}^{IMU2}_{EOF}\boldsymbol{R}$), the rotation of the coordinate frame of the first axis relative to the IMU mounted after that axis (${}^{IMU1}_{1}\boldsymbol{R}$, see Figure 1), and biases of the gyroscopes ($\boldsymbol{\omega}_{1,0}$ and $\boldsymbol{\omega}_{2,0}$). While not really time-invariant, the gyroscope biases vary so slowly that we consider them to be constant over the duration of the calibration motion.

The two rotation matrices of $\boldsymbol{\vartheta}_{\text{calib}}$ have only three DOF each (${}^{IMU1}_{1}\boldsymbol{R}, {}^{IMU2}_{EOF}\boldsymbol{R} \in \mathbb{SO}(3)$), but do not have a 3-dimensional, singularity-free parametrization. The solver we use, SLOM, can perform least squares optimization on elements of manifolds meeting certain criteria, which are called $\boxplus$-manifolds and include $\mathbb{SO}(3)$, by parameterizing changes to the manifold elements vectorially. The $\boxplus$-theory is treated very thoroughly by (Hertzberg et al., 2013). For the purpose of this particular calibration problem, we have two operators to apply a vectorially represented change $\boldsymbol{\delta}$ to a rotation matrix $\boldsymbol{R}_1$ and to obtain $\boldsymbol{\delta}$ from two rotation matrices $\boldsymbol{R}_1, \boldsymbol{R}_2$, namely

$$\begin{aligned} &\boxplus : \mathbb{SO}(3) \times \mathbb{R}^3 \to \mathbb{SO}(3) \\ &\quad \text{with } \boldsymbol{R}_2 = \boldsymbol{R}_1 \boxplus \boldsymbol{\delta} = \boldsymbol{R}\,\text{Rot}(\boldsymbol{\delta}) \end{aligned} \tag{16}$$

$$\begin{aligned} &\boxminus : \mathbb{SO}(3) \times \mathbb{SO}(3) \to \mathbb{R}^3 \\ &\quad \text{with } \boldsymbol{\delta} = \boldsymbol{R}_2 \boxminus \boldsymbol{R}_1 = \text{aRot}(\boldsymbol{R}_1^{-1}\boldsymbol{R}_2). \end{aligned} \tag{17}$$

$\text{Rot}(\boldsymbol{v}) = \exp \boldsymbol{v}_\times$ with $\boldsymbol{v}_\times \boldsymbol{u} = \boldsymbol{v} \times \boldsymbol{u}$ for any $\boldsymbol{u} \in \mathbb{R}^3$ is the rotation matrix whose scaled axis representation is $\boldsymbol{v}$ and aRot is its inverse.

These operators are best thought of as '$+$' and '$-$' operations for rotations, which encapsulate the structure of the rotation representation. They are easily extended to the parameter vector by using ordinary '$+$' and '$-$' for the vectorial components and '$\boxplus$' and '$\boxminus$' for the rotation components.

4.2 Measurements

Sensor Measurements. The components related to measurements of (13) are filled in here. There are two types of measurements. Sensor measurements and pseudo-measurements derived from properties we know about the robot.

The sensors are sampled at N time points t_k, $1 \leq k \leq N$ over the duration of the calibration motion. Each sensor is sampled at each time t_k (i.e. when a measurement is available the time of the microcontroller is used as time stamp), so for time t_k, there is a stack of measurements $\boldsymbol{Z}_k$, which contains a sample of each sensor,

$$\boldsymbol{Z}_k = \left(\boldsymbol{\theta}_{\text{meas},k}^T \quad {}^{IMU1}\boldsymbol{\omega}_{1,k}^T \quad {}^{IMU2}\boldsymbol{\omega}_{2,k}^T\right)^T. \tag{18}$$

In (18), $\boldsymbol{\theta}_{\text{meas},k} \in \mathbb{R}^3$ are rotary encoder measurements of each axis and ${}^{IMU1}\boldsymbol{\omega}_{1,k}$, ${}^{IMU2}\boldsymbol{\omega}_{2,k}$ the gyroscope measurements in their sensor coordinate frame.

All these measurements build the measurement vector

$$\boldsymbol{Z} = \left[\boldsymbol{Z}_k\right]_{k=1}^{N} \quad . \tag{19}$$

Error Function. The error function F of (13) consists of following components. For each time t_k there is a component pertaining to the motor

$$F_{\text{motor},k} = \boldsymbol{\theta}_{\text{meas},k} - \boldsymbol{\theta}_k - \boldsymbol{\theta}_0 \quad , \tag{20}$$

where $\boldsymbol{\theta}_0$ is the known zero-position of the motor and $\boldsymbol{\theta}_k$ the respective component of the parameters $\boldsymbol{\vartheta}$ as defined in (14) and (8), respectively.

Similarly, there are error components considering the gyroscopes ($F_{\text{gyro1},k}, F_{\text{gyro2},k}$), which complete

$$F_k = \begin{pmatrix} F_{\text{motor},k}^T & F_{\text{gyro1},k} & F_{\text{gyro2},k}^T \end{pmatrix}^T \quad . \tag{21}$$

The error considering only the gyroscope at the first axis is

$$F_{\text{gyro1},k} = \dot{q}_{k,1} - {}^{world}\boldsymbol{\omega}_{1,k,2} \quad , \tag{22}$$

with $\dot{q}_{k,i}$ the i-th link angle velocity, ${}^{world}\boldsymbol{\omega}_{1,k,2}$ the y-axis of gyroscope one, and

$${}^{world}\boldsymbol{\omega}_{1,k} = {}^{1}_{world}\boldsymbol{R}^{-1}\, {}^{IMU1}_{1}\boldsymbol{R}^{-1}\left({}^{IMU1}\boldsymbol{\omega}_{1,k} - \boldsymbol{\omega}_{1,0}\right), \tag{23}$$

with the rotation from world to first frame

$${}^{1}_{world}\boldsymbol{R} = \begin{pmatrix} \cos(q_{k,1}) & \sin(q_{k,1}) & 0 \\ -\sin(q_{k,1}) & \cos(q_{k,1}) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad . \tag{24}$$

I.e. it is the difference in the angular velocity of the axis as stored in $\boldsymbol{\vartheta}$ and the vertical component of measured angular velocity corrected by the gyroscope bias and rotated into the world reference frame.

The error function component for the end-effector gyroscope is

$$F_{\text{gyro2},k} = {}^{IMU2}\hat{\boldsymbol{\omega}}_2 - {}^{IMU2}\boldsymbol{\omega}_{2,k} \tag{25}$$

with

$$\begin{aligned} {}^{IMU2}\hat{\boldsymbol{\omega}}_2 = {}^{IMU2}_{EOF}\boldsymbol{R} \Big({}^{EOF}_{1}\boldsymbol{R}\, {}^{IMU1}_{1}\boldsymbol{R}^{-1}\, {}^{IMU1}\boldsymbol{\omega}_{1,k} \\ + {}^{EOF}\boldsymbol{\omega}_{\dot{q}_2} + {}^{EOF}\boldsymbol{\omega}_{\dot{q}_3} \Big) + \boldsymbol{\omega}_{2,0} \quad , \end{aligned} \tag{26}$$

where

$${}^{EOF}\boldsymbol{\omega}_{\dot{q}_2} = \begin{pmatrix} 0 & \sin(q_{k,3}) & 0 \\ 0 & -\cos(q_{k,3}) & 0 \\ 0 & 0 & 0 \end{pmatrix} \dot{\boldsymbol{q}}_k \quad , \tag{27}$$

$${}^{EOF}\boldsymbol{\omega}_{\dot{q}_3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \dot{\boldsymbol{q}}_k \quad , \tag{28}$$

$${}^{EOF}_{1}\boldsymbol{R} = \begin{pmatrix} c_2 & 0 & -s_2 \\ -s_2 s_3 & -c_3 & -c_2 s_3 \\ s_2 c_3 & s_3 & c_2 c_3 \end{pmatrix} \quad . \tag{29}$$

This looks more complicated than it is. $F_{\text{gyro2},k}$ is the difference between the sensor measurement of the gyroscope at the end-effector and the expected angular velocity at the location of the end-effector gyroscope, which is the measurement of the gyroscope at the first axis plus the angular velocity caused by the rotations of the second and third axes.

In addition to the sensor measurement, we know that initially, at $t_k = t_1$, all angular velocities of the links and the motors are zero and that the links have approximately the same angles as the motors. The corresponding component of the error function is

$$F_{\text{initial}} = \begin{pmatrix} (\boldsymbol{q}_1 - \boldsymbol{\theta}_{\text{meas},1})^T & \dot{\boldsymbol{q}}_1^T & \dot{\boldsymbol{\theta}}_1^T \end{pmatrix} \quad . \tag{30}$$

We also know that the system behaves approximately according to the dynamics function f_{dyn} in (9), which relates the states of two successive time points t_k and t_{k+1}. This contributes to the error function $N-1$ times, with $T_s = t_{k+1} - t_k$ and the pwm $\boldsymbol{u}_k$,

$$\begin{aligned} F_{\text{dyn},k} = \boldsymbol{\vartheta}_{\text{state},k} + T_s f_{\text{dyn}}(\boldsymbol{\vartheta}_{\text{state},k}, \boldsymbol{u}_k) \\ - \boldsymbol{\vartheta}_{\text{state},k+1} \quad . \end{aligned} \tag{31}$$

With these error functions, the bias concerning the horizontal axes of the gyroscope at the first axis is unobservable. To fix this, we keep the commands of the first $M \leq N$ time steps zero, i.e. $\boldsymbol{u}_k = \boldsymbol{0}$ for $1 \leq k \leq M$. To let the solver exploit this knowledge, we define the error function components only valid until t_M:

$$F_{\text{bias},k} = \begin{pmatrix} (\boldsymbol{\omega}_{1,0} - \boldsymbol{\omega}_{1,k})^T & (\boldsymbol{\omega}_{2,0} - \boldsymbol{\omega}_{2,k})^T \end{pmatrix}^T \tag{32}$$

That is, as long as there is no command to move, the gyroscopes should measure their bias.

In summary, the error function leads to

$$F(\boldsymbol{Z}, \boldsymbol{\vartheta}) = \begin{pmatrix} F_{\text{initial}} \\ \left[F_k\right]_{k=1}^{N} \\ \left[F_{\text{dyn},k}\right]_{k=1}^{N-1} \\ \left[F_{\text{bias},k}\right]_{k=1}^{M} \end{pmatrix} \quad . \tag{33}$$

The corresponding covariance is a block-diagonal matrix. We assumed the same covariance for all errors of the same kind, so with A^P denoting P repetitions of A, the covariance is

$$\Sigma = \text{blkdiag}(\Sigma_{\text{initial}}, \Sigma_{\text{meas}}^N, \Sigma_{\text{dyn}}^{N-1}, \Sigma_{\text{bias}}^M). \tag{34}$$

5 EVALUATION

Our identification uses an off-line method based on a trajectory the robot is following. Here we implemented a simple P-controller to move to different positions. The maximum torque set to the motors by the controller can also be adjusted. For our identification

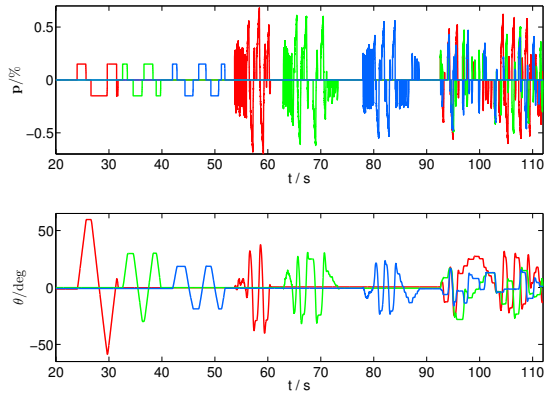


Figure 3: On top we see the pwm signal $\boldsymbol{u}$ and on bottom the measured motor angle $\boldsymbol{\theta}$ transformed to the link side. The color encoding is red, green, and blue for yaw (1), pitch (2), and roll (3) axis, respectively.

we move the robot in two pwm stages, the first is slow with large displacement ($\boldsymbol{u} = 0.15$). This should help the identification of the friction coefficient $\boldsymbol{\mu}_{\text{fm}}$. In the second stage we move very fast ($\boldsymbol{u} = 0.90$). In the fast stage we implemented a kind of a chirp signal, e.g. between 53 s and 56 s in Figure 3. The chirp should be useful to classify the parameters of the spring ($\boldsymbol{d}_{\text{s}}$ and $\boldsymbol{k}_{\text{s}}$). The parameters for the motor, i.e. k_{pwm}, k_{mi}, and b, should be detectable at any time the robot moves. The rotations ${}^{IMU1}_{1}\boldsymbol{R}$ and ${}^{IMU2}_{EOF}\boldsymbol{R}$ should be close to the known values from the CAD.

Figure 3 shows the pwm signal together with the measured angles of the encoders. The movement with the distortion of the bearing can be seen in the video mentioned in Figure 1. The effect of the flexible bearing is best seen on the first gyroscope's x- and z-axis, due to an indirect effect by movements of joints 2 or 3 (Figure 4).

Joints 2 and 3 indirectly affect the first gyroscope's x- and z-axis, due to the flexibility of the bearing.

5.1 Initial Guess

The data-sheet of encoders and IMU sensors give us the standard deviations $\sigma_{\text{encoder}} = 6.9259 \times 10^{-5}$ rad and $\sigma_{\text{gyro}} = 1.745 \times 10^{-3}$ rad/s, which remains the same for all measurements. The measurement covariance is given by

$$\Sigma_{\text{meas}} = \text{blkdiag}(\sigma^2_{\text{encoder}} \boldsymbol{I}^{3\times3}, \sigma^2_{\text{gyro}} \boldsymbol{I}^{6\times6}) \quad . \tag{35}$$

For the initial covariance Σ_{initial} only the component regarding the displacement of link and motor angle changes. This is due to the influence of the spring. Σ_{dyn} is set to a change in the motor velocity in 1000 steps of 15°/s. The link velocity is set to

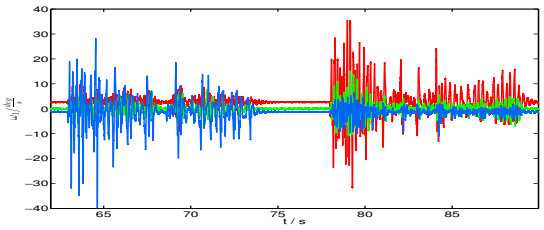


Figure 4: Gyroscope 1 measurement influenced by the bearings flexibility. The effect on the x- and z-axis is shown, which should be 0 in theory by a movement of joints 2 or 3. Red, green, and blue denotes x, y, and z axis, respectively.

change 20°/s. For motor and link position covariance is chosen small, as the position error is mostly given by the velocity error. The dynamic covariance is constant for all time steps. It emphasizes only the main effects and neglects cross-correlations. I.e. elements on the diagonal are chose conservatively and are zero off diagonal.

$$\begin{aligned}\Sigma_{\text{dyn}} = \text{blkdiag}(&10^{-12}\boldsymbol{I}^{3\times3}, 1.21\times10^{-4}\boldsymbol{I}^{3\times3},\\ &10^{-12}\boldsymbol{I}^{3\times3}, 6.889\times10^{-5}\boldsymbol{I}^{3\times3})\end{aligned} \tag{36}$$

$$\Sigma_{\text{initial}} = \text{blkdiag}(0.0076\boldsymbol{I}^{3\times3}, \sigma^2_{\text{encoder}}\boldsymbol{I}^{6\times6}) \tag{37}$$

For the identification we need to set an initial guess of the parameters. For the motor we can compute the k_{pwm} and k_{mi} from basic motor data – in our case this information is limited to the rated current and speed, the voltage supplied, and the output torque. We approximate the motor inertia by its weight and radius. The gyroscopes offset $\boldsymbol{\omega}_{1,0}$ and $\boldsymbol{\omega}_{2,0}$ can be measured beforehand and used as initial guess. All other parameters were hand tuned within a simulation of the dynamics to get close to the measurement for each axis separately. This was only done for the first slow movements. We took the parameters which fit quite well as initial guess. The initial guess and the identified parameters are given in Table 1. We can see only a slight difference for k_{pwm}, b, $\boldsymbol{\mu}_{\text{fm}}$, bias $\boldsymbol{\omega}_{1,0}$ and $\boldsymbol{\omega}_{2,0}$, and for the rotation matrices. This result is, except for $\boldsymbol{\mu}_{\text{fm}}$, expectable, as we can measure or compute the parameters with a small tolerance. For $\boldsymbol{\mu}_{\text{fm}}$ we just had a good hand tuning with the simulation. The divergence of parameter k_{mi} is not a surprise, as we could only roughly compute this a priori. For the spring parameters we also expected a low value for the stiffness, but for Axis 3 we expected a higher value than for Axis 2. As one can feel that Axis 3 is more "stiff" than Axis 2 in its spring. On the other hand, we had the idea that the calibration algorithm will pack the bearings flexibility into the spring stiffness, which we can see here.

Table 1: Initial guess and identified parameters.

	Initial guess	Identified
k_{pwm}/ NmV^{-1}	3.487	4.696
k_{mi}/ Nmsrad^{-1}	15.563	31.328
b/ kgm^2	2.681	2.428
$\boldsymbol{\mu}_{\text{fm}}$/ Nm	$(2.0\ 2.0\ 2.0)^T$	$(1.599\ 2.565\ 4.855)^T$
$\boldsymbol{k}_{\text{s}}$/ Nmrad^{-1}	$(400\ 400\ 600)^T$	$(187.555\ 120.070\ 103.056)^T$
$\boldsymbol{d}_{\text{s}}$/ Nmsrad^{-1}	$(1.0\ 1.0\ 1.0)^T$	$(5.335\ 4.143\ 8.043)^T$
$\boldsymbol{\omega}_{1,0}$ / deg	$(2.615\ 0.014\ -1.273)^T$	$(2.615\ 0.014\ -1.273)^T$
$\boldsymbol{\omega}_{2,0}$ / deg	$(2.723\ -0.102\ -0.652)^T$	$(2.723\ -0.103\ -0.652)^T$
${}^{IMU1}_{axis1}\boldsymbol{R}$	$\begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}^T$	$\begin{pmatrix} -0.999 & -0.0439 & -0.0199 \\ 0.0189 & 0.0219 & -0.999 \\ 0.044 & -0.999 & -0.020 \end{pmatrix}^T$
${}^{IMU2}_{EOF}\boldsymbol{R}$	$\begin{pmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{pmatrix}^T$	$\begin{pmatrix} -0.009 & 0.027 & 0.999 \\ 0.030 & -0.999 & 0.027 \\ 0.999 & 0.030 & 0.008 \end{pmatrix}^T$

5.2 Verification

To verify our results we compare the measured data with predicted data of a model. (1.) We take the first state SLOM optimized as our initial state $\boldsymbol{x}_0$. (2.) We call our dynamics recursively to compute the new state from the old $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + T_{\text{s}} f_{dyn}(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1})$ like in (31), with $T_{\text{s}} = 1$ ms. Thus we get all states and can compare the measured position and velocity with the predicted position $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}$. (3.) The predicted value for $\boldsymbol{\omega}_1$ based on the state can be extracted from (23). This means a rotation of the first axis leads to a change in $\boldsymbol{\omega}_1$. (4.) With (26) we get the predicted $\boldsymbol{\omega}_2$ using the predicted $\boldsymbol{\omega}_1$, $\boldsymbol{q}$, and $\dot{\boldsymbol{q}}$. We compare the predicted and measured gyroscope values.

The comparison is splitted into five different regions of the measurement for better visualization. We start with the slow behavior for all three axes in Figure 5(a). We see that the predicted data fits well to the measured data, except for the motor positions. The reason is an accumulated error on the velocity will affect the joint's position. Also small velocity errors are leading to drift. This is a well known behavior. For the velocities there should be no drift visible. The measurements are approximated by our prediction with little divergence for velocities.

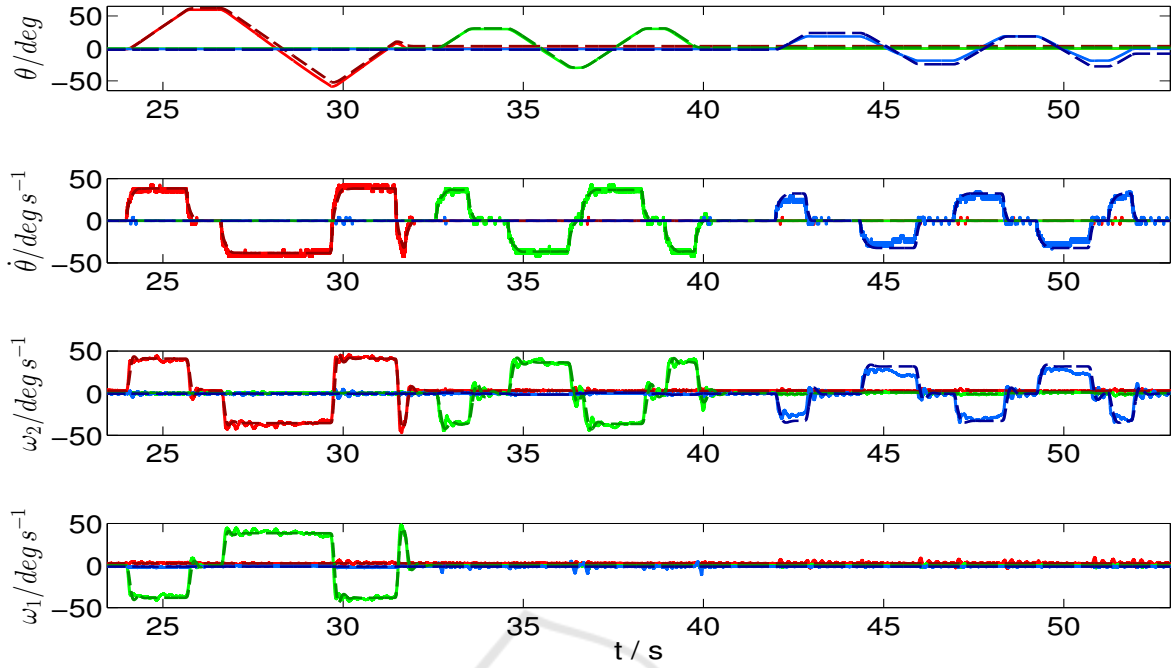
The second region shows the yaw axis movement (Figure 5(b)). There is a difference between the part where the joint's velocity changes fast and the part where it moves more constantly in one direction. The motor velocity is slightly better fitted than the link velocity transferred to the gyroscope motion. As expected, there is no influence of the flexible bearing.

Movements of the flexible bearing are provoked when the motor turning the pitch axis changes direction (Figure 6(a)). Here, the flexible bearing gets stimulated. After the shock, the joint performs very little force on the bearing, so it can recover from the impact and return to its initial state, i.e. zero position and velocity. While recovering it oscillates. Although the effect is clearly visible in the gyroscopes, the motion of the turning axis can still be detected. For the motor velocity it fits again to the measured data, but our predicted second gyroscope values fit not as well as for the first axis. As we neglect the flexible bearing in the model, the predicted measurement of the first gyroscope is close to zero. Adjusting the spring stiffness is the solver's only chance to account for the movements of the flexible bearing. Thus, it finds a compromise between spring stiffness and flexibility of the bearing to fit the data as well as possible.

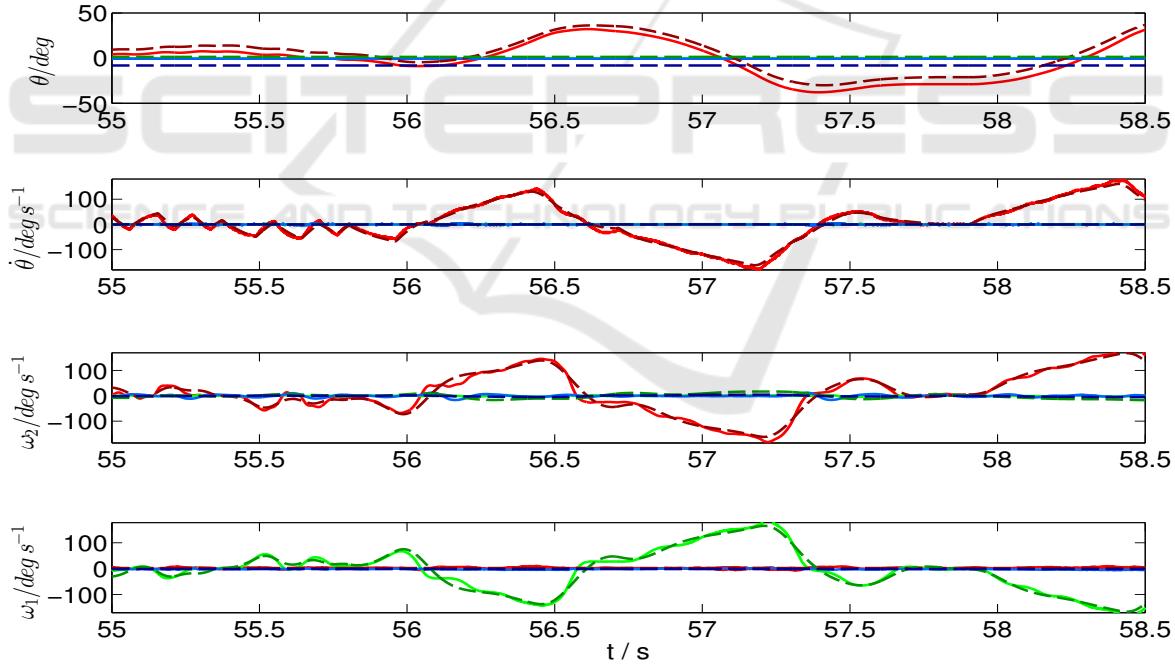
This effect is much better seen on the roll axis (Figure 6(b)). Note the delay between measurement and predicted data. This is caused by the spring stiffness. If we put a rod instead of a tooth belt ($\boldsymbol{k}_{\text{s}} \to \infty$), the link would follow the motor immediately. The opposite would be a very flexible tooth belt, i.e. $\boldsymbol{k}_{\text{s}} \to 0$. Then the link would follow eventually. The compromise SLOM does is to take motions from the bearing and the joints into account. From the view of SLOM, there is no difference between these two motions. Thus, it lowers the stiffness.

In Figure 7 all axes move together. It is hard to make a statement here, because the drift of the angles affects the other joints velocity as they are coupled through the link inertia $\boldsymbol{M}(\boldsymbol{q})$. However, in principle the motor velocities are predicted well, whereas the gyroscope values have higher variations to the measured values, due to the under-determined spring, as result of the bearings flexibility.

Cross Verification To cross check our identified parameters we took another motion sample and compare measured and predicted data for a new pwm



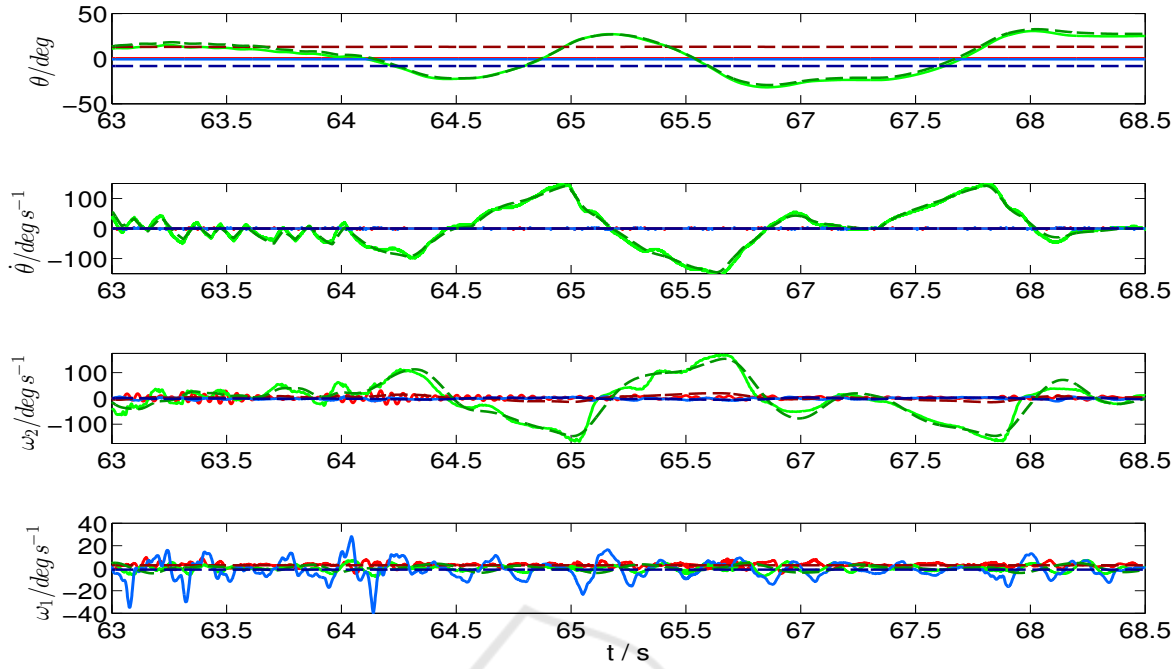
(a) Slow motion.



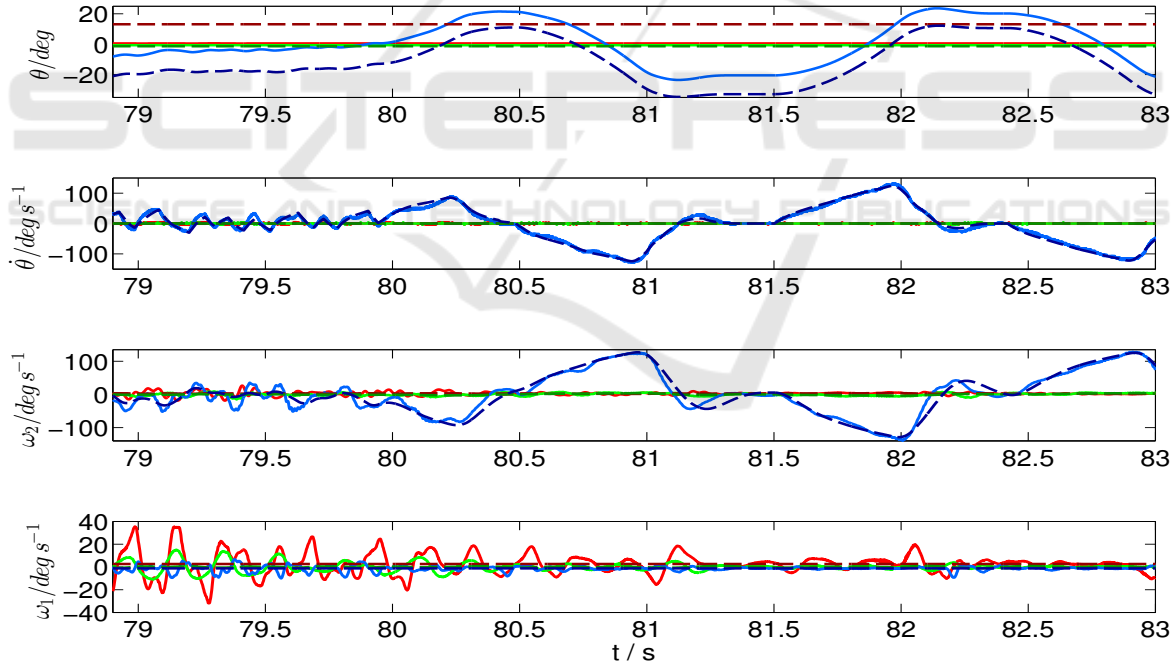
(b) Joint 1 moving (yaw).

Figure 5: Comparison of measured (solid) and predicted data (dashed). For each figure we see from top to bottom the motor position $\boldsymbol{\theta}$, the motor velocity $\dot{\boldsymbol{\theta}}$, the second gyroscope ${}^{IMU2}\boldsymbol{\omega}_2$, and the first ${}^{IMU1}\boldsymbol{\omega}_1$. For motor angle and velocity the red, green, and blue denote yaw, pitch, roll axis respectively. For the gyroscopes it denotes x, y, z axis based on its coordinate system.

signal. Figure 8 shows this new recorded motion. The predicted motion (dashed line) approximately fits the measurement as described in the previous section. Additionally, there is no chirp in this movement, which simulates the normal behavior we want to achieve. It shows that the parameters found are a



(a) Joint 2 moving (pitch).



(b) Joint 3 moving (roll).

Figure 6: Comparison of measured (solid) and predicted data (dashed). For each figure we see from top to bottom the motor position $\boldsymbol{\theta}$, the motor velocity $\dot{\boldsymbol{\theta}}$, the second gyroscope ${}^{IMU2}\boldsymbol{\omega}_2$, and the first ${}^{IMU1}\boldsymbol{\omega}_1$. For motor angle and velocity the red, green, and blue denote yaw, pitch, roll axis respectively. For the gyroscopes it denotes x, y, z axis based on its coordinate system.

good fit for other motions. Thus, we are able to predict the velocities quite well, but need to have in mind the drift of the positions.

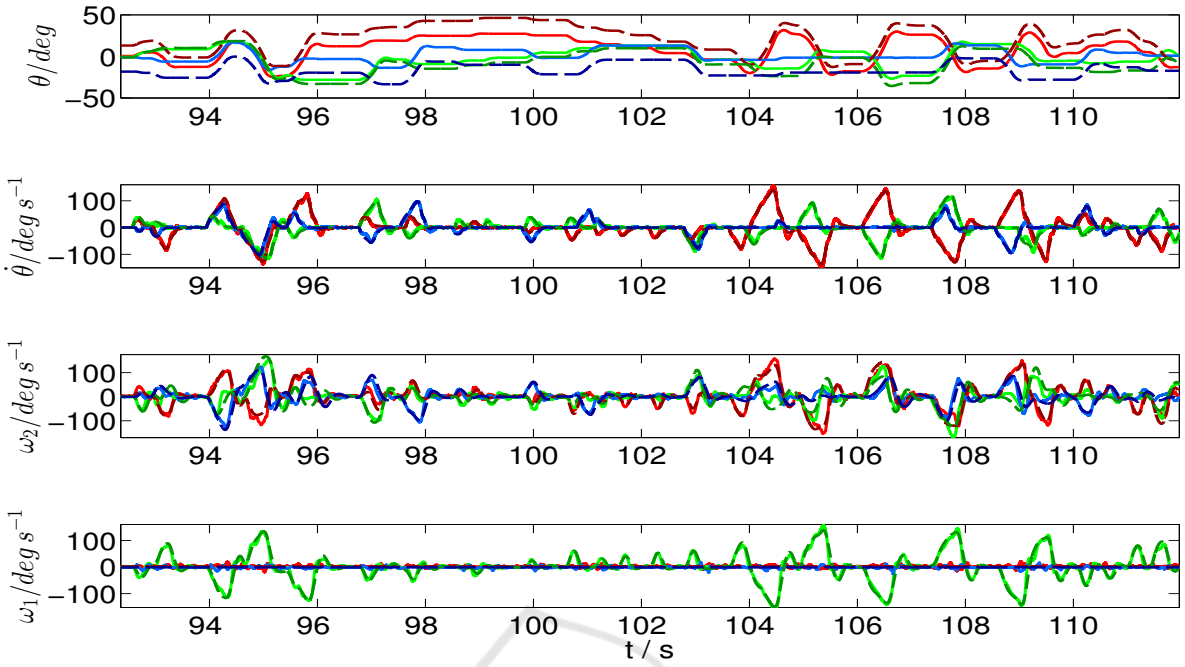


Figure 7: Comparison of measured (solid) and predicted data (dashed). For motor angle $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}$ the red, green, and blue denote yaw, pitch, roll axis respectively. For the gyroscopes $\boldsymbol{\omega}$ it denotes x, y, z axis based on its coordinate system.

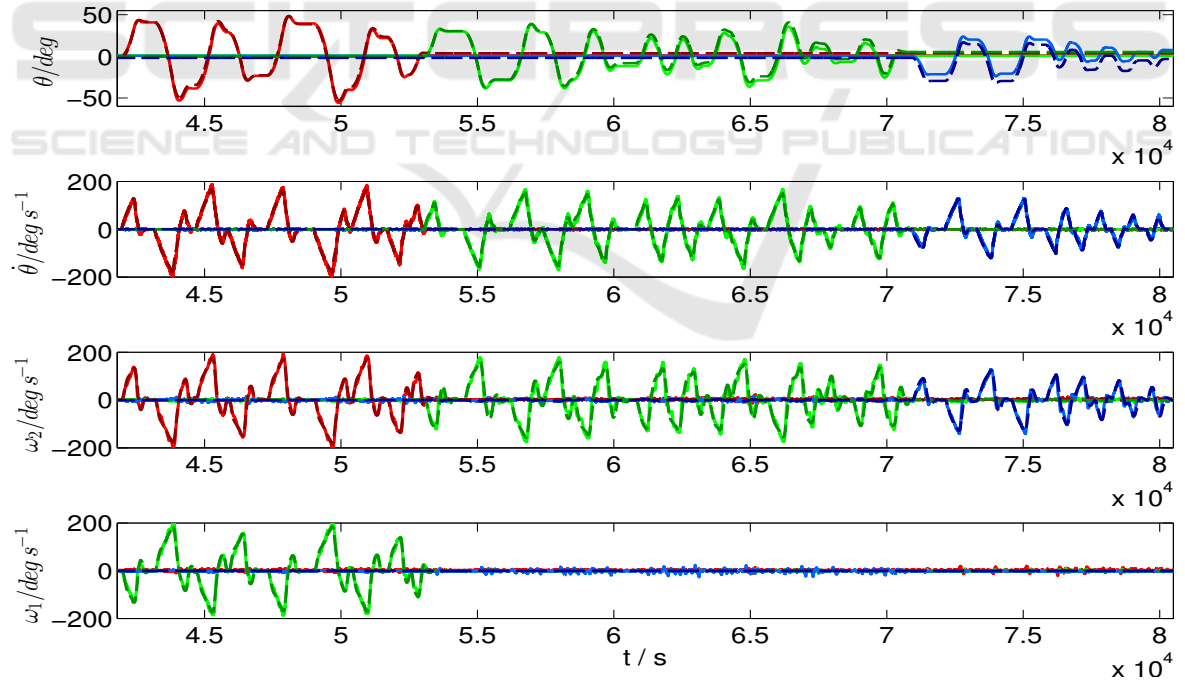


Figure 8: Comparison of measured (solid) and predicted data (dashed). For motor angle $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}$ the red, green, and blue denote yaw, pitch, roll axis respectively. For the gyroscopes $\boldsymbol{\omega}$ it denotes x, y, z axis based on its coordinate system.

6 CONCLUSION

We have been shown that it is possible to find parameters for the dynamics even if the model takes no flexibility of the bearing into account, which moves the whole body. To get more accurate results the solver packs the bearing's flexibility into the flexibility between joints and links. The predicted behavior of the robot based on the identified model parameters has deviations to the measured data. These deviations are larger whenever the velocities of the second and third bodies change direction, because the bearing's flexibility gets stimulated. The overall approximation is fitting to the measurements well. However, while the model is reasonably good and, at the same time, very simple, we suspect it will not be good enough for our controller.

By now, we are working on an extension of the model presented herein, where the flexibility of the bearing is part of the dynamic model. We believe that this can be achieved by putting two new joints before the first one, acting directly on the same coordinate system as joint one. The bodies of the two inserted joints will have no mass and no inertia. Their rotation is about the x and y axes of joint one. The flexibility is given by a spring acting between the origin and the joints position. We hope that this will give us a better model and separate the behavior of joint two and three from the behavior of the bearing.

Additionally, we have to calibrate the cameras for the ball tracking, too. We want to add this calibration into the calibration we were stated herein. Moreover, we can use the cameras to examine in some positions the position of the links 2 and 3.

ACKNOWLEDGEMENT

This work has been supported by the Graduate School SyDe, funded by the German Excellence Initiative within the University of Bremen's institutional strategy.

REFERENCES

Albu-Schaeffer, A., Ott, C., and Hirzinger, G. (2007). A unified passivity-based control framework for position, torque and impedance control of flexible joint robots. *The International Journal of Robotics Research*, 26(1):23–39.

Cheng, P. and Oelmann, B. (2010). Joint-angle measurement using accelerometers and gyroscopes — a survey. *IEEE Transactions on Instrumentation and Measurement*, 59(2):404–414.

Featherstone, R. (2012). Spatial_v2 (version 2). http://royfeatherstone.org/spatial/v2/notice.html.

Grotjahn, M., Daemi, M., and Heimann, B. (2001). Friction and rigid body identification of robot dynamics. *International Journal of Solids and Structures*, 38(1013):1889 – 1902.

Hertzberg, C. (2008). A framework for sparse, non-linear least squares problems on manifolds. Diploma thesis, Universität Bremen.

Hertzberg, C., Wagner, R., and Frese, U. (2012). Tutorial on quick and easy model fitting using the slom framework. In Stachniss, C., Schill, K., and Uttal, D., editors, *Spatial Cognition VIII*, volume 7463 of *Lecture Notes in Computer Science*, pages 128–142. Springer-Verlag Berlin Heidelberg.

Hertzberg, C., Wagner, R., Frese, U., and Schröder, L. (2013). Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Information Fusion*, 14(1):57–77.

Mata, V., Benimeli, F., Farhat, N., and Valera, A. (2005). Dynamic parameter identification in industrial robots considering physical feasibility. *Advanced Robotics*, 19(1):101–119.

Moberg, S. (2010). *Modeling and Control of Flexible Manipulators*. PhD thesis, Linköping University, Automatic Control, The Institute of Technology.

Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27.

Olsson, H., strm, K., de Wit, C. C., Gfvert, M., and Lischinsky, P. (1998). Friction models and friction compensation. *European Journal of Control*, 4(3):176 – 195.

Schüthe, D. and Frese, U. (2015). Optimal control with state and command limits for a simulated ball batting task. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015*, pages 3988–3994.

Siciliano, B. and Khatib, O. (2008). *Springer handbook of robotics*. Springer.

Staufer, P. and Gattringer, H. (2012). State estimation on flexible robots using accelerometers and angular rate sensors. *Mechatronics*, 22(8):1043 – 1049.

Vuong, N. D. and Ang Jr, M. H. (2009). Dynamic model identification for industrial robots. *Acta Polytechnica Hungarica*, 6(5):51–68.

Wu, J., Wang, J., and You, Z. (2010). An overview of dynamic parameter identification of robots. *Robotics and Computer-Integrated Manufacturing*, 26(5):414 – 419.