Controller implementation on the base and torso of SERGIO

T.M.C. Peters

CST 2015.061

Master's thesis

Coach(es): ir. J.J.M. Lunenburg
dr. ir. M.J.G. van de Molengraft

Supervisor: prof.dr.ir. M. Steinbuch

Eindhoven University of Technology
Department of Mechanical Engineering
Control Systems Technology

Eindhoven, May, 2015

Abstract

In order to facilitate the demand in domestic service robots AMIGO is used at the TU/e for developing and testing both hardware and software. The gain experience with AMIGO let to the development of a new robot, SERGIO. A modular design method is used for SERGIO giving it interchangeable parts consisting of a base, torso, two arms and a head. In this thesis the identification, controller design and implementation of the low level control for both base and torso are presented.

The base platform is driven by four individual actuated mecanum wheels making it full holonomic. The goal of the base controller is to track the supplied reference velocities as accurate as possible. With the reference velocities in the robot space and the measured position in wheel orientation the control problem led to two main choices in the feed back control structure. Decentralized control on individual wheels is compared to decentralized control on the Degrees Of Freedom (DOF) of the robot with use of Multiple Input Multiple Outputs (MIMO) Frequency Response Function (FRF) tools. Since the reference is on velocity level and the sensor data on position level, control is compared on the position- and velocity error. In addition to the feedback feed forward, compensating for both acceleration and friction, is investigated. With the rollers on the outside of the wheels the friction becomes direction dependent and a method is proposed to compensate for this. To validate the design improvements tests are preformed with an external measurement unit that show cancellation of wheel slip. Similar tests are performed with AMIGO that clearly show the improvements in the design of SERGIO.

The torso is designed as a human like mechanism consisting of a ankle-, knee- and hip joint. To reduce costs only the ankle- and hip joint are actuated and the knee joint is coupled to the ankle joint by a four bar guiding mechanism. The joints are actuated by lead screws. Gas springs are used to compensate gravity forces and to reduce the required motor force. A multiple joint dynamic model of the torso is made of which the parameters are estimated by use of experiments. In simulation the arms are added to the identified model to calculate the required drive train torques. This resulted in the advise on the gas spring load and replacement of the gearboxes to increase the reachable space of the shoulders. On the torso mechanism without arms and gas springs a controller is designed, and implemented, which copes with all non-linear dynamics induced by the linear actuated revolute joints and the coupled joint by the four bar guiding mechanism. Future work, to make the torso ready to use, include minor adjustments in the controller parameters to work with the changed hardware.

Nomenclature

AMIGO	Autonomous Mate for IntelliGent Operations
DOF	Degree Of Freedom
FRF	Frequency Response Function
IMU	Inertial Measurement Unit
MIMO	Multiple Input and Multiple output
Orocos	Open Robot Control Software
PD	Proportional and Derivative action
PID	Proportional, Integral and Derivative action
RGA	Relative Gain Array
ROS	Robot Operating System
SERGIO	Second Edition Robot for Generic Indoor Operations
SISO	Single Input Single Output
TU/e	Eindhoven University of Technology

Contents

Abstract i

Nomenclature iii

1 Introduction 1
- 1.1 Background . . . 1
- 1.2 Motivation . . . 2
- 1.3 Problem statement . . . 3
- 1.4 Outline . . . 4

2 Base 5
- 2.1 Design . . . 6
- 2.2 Related work . . . 7
  - 2.2.1 Transfer robot and wheel frame . . . 7
  - 2.2.2 Slip . . . 8
  - 2.2.3 Control methods . . . 9
  - 2.2.4 Contribution . . . 9
- 2.3 Identification . . . 9
  - 2.3.1 Method . . . 10
  - 2.3.2 Measurement FRF in the air . . . 10
  - 2.3.3 Measurement FRF on the floor . . . 11
- 2.4 Feed back control . . . 12
  - 2.4.1 Coupled versus decoupled feedback control . . . 13
  - 2.4.2 Position versus velocity feedback control . . . 15
  - 2.4.3 Stability . . . 16
- 2.5 Feed forward . . . 18
- 2.6 Performance . . . 20
  - 2.6.1 Data processing . . . 20
  - 2.6.2 Results corridor experiment . . . 21
  - 2.6.3 Results bump experiment . . . 23
  - 2.6.4 Design validation . . . 23
- 2.7 Final implementation . . . 24

3 Torso 27
- 3.1 Design . . . 27
- 3.2 Related work . . . 28
  - 3.2.1 Independent joint control . . . 29
  - 3.2.2 Multivariable joint control . . . 32

3.2.3 Contributions . . . 35
3.3 Model . . . 35
3.3.1 Kinematic relations . . . 36
3.3.2 Equations of motion . . . 37
3.4 Identification . . . 38
3.4.1 Simulation . . . 38
3.4.2 Model parameters . . . 39
3.4.3 Trunk identification . . . 41
3.4.4 Leg identification . . . 43
3.4.5 Identified model . . . 45
3.5 Spring stiffness . . . 45
3.5.1 Specifications and requirements . . . 46
3.5.2 Trunk analysis . . . 48
3.5.3 Leg analysis . . . 50
3.5.4 Spring and drive train choice . . . 51
3.5.5 Comparison AMIGO and SERGIO . . . 52
3.6 Torso controller design . . . 53
3.6.1 Feedback . . . 53
3.6.2 Feed forward . . . 55
3.6.3 Performance . . . 56
3.7 Final implementation . . . 58

4 Conclusions & Recommendations **61**
4.1 Conclusions . . . 61
4.2 Recommendations . . . 63

Bibliography **66**

A Base **67**
A.1 Base drive train specifications . . . 67
A.2 FRF measurements . . . 67
A.3 Feed forward . . . 71
A.4 Performance . . . 71

B Torso **75**
B.1 Specifications . . . 75
B.2 State space representation of a DC motor driven joint . . . 75
B.3 Kinematic relations. . . . 76
B.4 Locations and velocities of the centres of mass . . . 77
B.5 Simulation results. . . . 77
B.6 Identification results . . . 80
B.7 Spring load analysis . . . 81
B.8 FRF measurements . . . 84
B.9 Controller comparison . . . 87

Chapter 1

Introduction

1.1 Background

Over the last few decades robots made their introduction in the domestic environment. Examples are autonomous vacuum cleaners and lawn mowers. These robots are developed to provide one specific service, namely cleaning the floor and mowing the lawn respectively. A more recent development in household robots is the domestic service robot. Ageing of the world population results in the need of extra care for the elderly. A domestic service robot can provide this care. The main goal of such a robot is the ability to perform multiple household tasks and assist elderly and disabled people in their daily lives. Preferably these robots work fully autonomous meaning no extra person is needed to control the robot. Examples of domestic service robots are shown in Figure 1.1. What the robots have in common is a movable base to transfer itself in a household, sensors to scan the environment they are operating in and a multi degree of freedom manipulator to manipulate objects.

In order to facilitate the demand for domestic service robots AMIGO (Autonomous Mate for IntelliGent Operations) shown in Figure 1.1a is developed [8] at the department of Mechanical Engineering at the TU/e (Eindhoven University of Technology). The development of this robot took part in several projects [1, 3] aiming to increase knowledge an experience in the field of service robots. At this moment the team of Tech United is still participating in the RoboCup@Home competition with AMIGO, where it finished second place at the world cup of 2014[4]. In this competition the robots compete in all kinds of challenges related to tasks the robots should be able to perform in a household. An example of such a challenge is the navigation test where the robot should drive to different way points while encountering objects like humans and doors blocking its path. Although there are promising results shown with AMIGO there is still room for improvement before the service robot can be deployed in the domestic environment.

(a) AMIGO (b) Rose (c) Care-O-bot

Figure 1.1: Domestic service robots.

1.2 Motivation

In order to continue progress in the field of autonomous service robots AMIGOs successor SERGIO (Second Edition Robot for Generic Indoor Operations), Figure 1.2, has been designed and build, as described in de Roest [9] and Derks [10]. The positive aspects in the design of AMIGO are also implemented in SERGIO, like a holonomic base and a 7-DOF arm. A holonomic base enables the robot to immediately drive in any desired direction which is useful to accurately position the robot. The 7-DOF arm result in the freedom to orient and position the gripper which is useful when manipulating objects. The main design improvements in SERGIO are:

- **Cost:** The aim is to get the domestic service robot in mass production. To realise this the costs should stay low and therefore many off the shelve products are chosen.
- **Modular:** A modular design makes it easy to interchange parts. Interchangeable parts lead to reduction in production costs. It also results in easier maintenance, because all parts are better accessible. SERGIO consists of the following modular parts: base, torso, head, two arms and two grippers.
- **Mecanum wheels:** Mecanum wheels are chosen over omni wheels because of better orientation and placement of the wheels. Its orientation should make it easier to overcome doorsteps and cables driving in forward direction. The placement of mechanum wheels increases stability and reduces the width of the base.
- **Suspension:** Suspension in the base makes it more compliant to the floor. A stiff base design

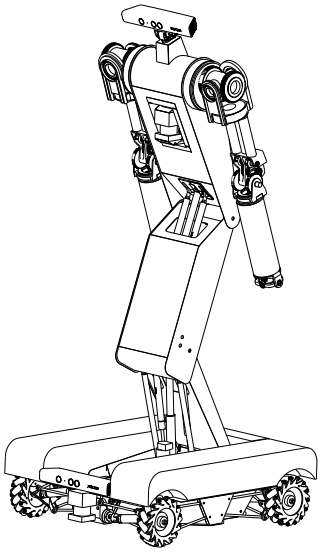


Figure 1.2: SERGIO design.

as used in AMIGO results in wheel slip, especially on a non compliant floor. Slip in an omni wheel driven base leads to unexpected behaviour and bad localization of the robot.

- **Torso workspace:** The torso workspace is increased with the use of an ankle-, knee- and hip joint which enables a human like motion for the torso. This enables the robot to pick up objects from the floor and increases the reach when picking up objects from the table. The ankle- and knee joint are mechanically coupled by a four bar mechanism to reduce the amount of actuators needed.
- **Rigid torso:** To reduce the error in the position and orientation transfer from base to gripper the torso is designed as a rigid structure with use of sheet metal box structures.
- **Arm stiffness and play**: Backlash and flexibility in the arms cause a large error in positioning the gripper. The arm is designed to keep the reachable space but reduce this play and flexibility.

1.3 Problem statement

AMIGO is performing well with the current low level control structure, but the changes in the design of its successor require an adjusted implementation of the low level control. This leads to the problem statement of this thesis:

> 'Design and implement the low level control for the base and torso of SERGIO and compare the performance with its predecessor.'

For the base controller small adjustments in the decoupling matrix and recalibration of the controller could suffice. The improved base design however gives the opportunity to apply other and

maybe better control methods resulting in the contribution of this thesis:

- Find the most suitable controller for a holonomic base, based on literature and experience with AMIGO

The torso mechanism exists of three revolute joints driven by two prismatic actuators. Two of the revolute joints are coupled by a four bar guiding mechanism. The control of this mechanism results in the second contribution:

- Design and implement a controller for a torso mechanism, coping with non-linearities caused by the lead screw driven revolute joints and the joint coupled by a four bar guiding mechanism.

In the control of both the base and the torso, safety is an important aspect, because the robot is placed in an environment where it moves around and interacts with people. In operation the robot should never harm a person and to guarantee this a safe implementation of the controller is of the largest importance. In addition to its safety the implementation should be able to communicate with the higher level software that is not robot specific, which leads to the third and last contribution:

- Implement the base and torso controllers on the service robot incorporating safety measures and using the present interface.

1.4 Outline

In Chapter 2 the base controller design and implementation is presented, starting in the first section with an introduction on the base design. On the basis of the design a short literature study is presented on dynamics and control of holonomic platforms in Section 2.2. In Section 2.3 the identification of the system with use of FRF (Frequency Response Function) measurement. The model and literature study are used to analyse and compare several feedback control structures and to choose the most suitable one, in Section 2.4. In addition to the feedback design some results of adding feed forward are discussed in 2.5. In order to validate design goals experiments on both SERGIO and AMIGO are performed and compared in Section 2.6. Section 2.7 presents the final implementation in the used software architecture, including the safety measures.

In Chapter 3 the torso mechanism of SERGIO is addressed. First the design is presented in Section 3.1. In Section 3.2 a literature study is performed on the modelling- and control of multiple joint systems and how to cope with the non-linearities. The modelling methods are used to set up the equations of motion of the system in Section 3.3. Based on results from the model in simulation, experiments are performed to identify unknown model parameters in Section 3.4. In Section 3.5 the identified model is used to calculate the loads of the gas springs that will be used in the torso to compensate for gravity forces induced by the torso and arms. With use of the literature study and the identified model a controller is designed and tested. In Section 3.7 the final implementation of the controller on the robot is presented including safety measures. In the end the results are summarized and recommendations are given in Chapter 4.

Chapter 2

Base

The base of SERGIO is shown in Figure 2.1. In this chapter a suitable controller for the base is designed tested and implemented. First the base design is presented followed by a literature study on possible control methods and problems encountered in controlling a holonomic base. After that the system is identified by measuring a MIMO FRF. With use of the model, literature and experiments several control structures are analysed and a control structure for the feedback is chosen. In addition to the feedback feed forward is investigated to improve performance. Tests are performed with both the base of SERGIO and AMIGO to compare performance and validate the design goals. At the end the final implementation in the current software architecture is presented which incorporates safety measures.

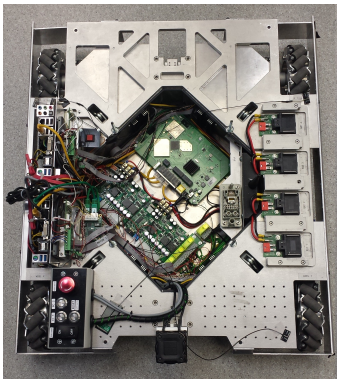
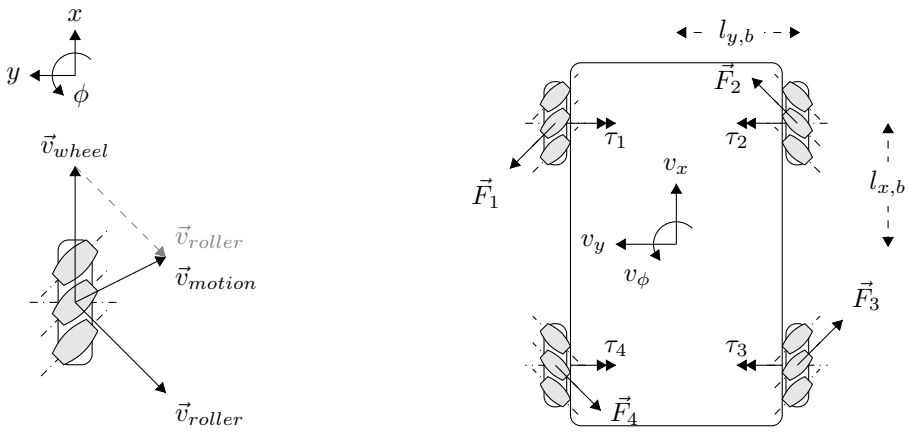


Figure 2.1: Base of SERGIO.

2.1 Design

The four mecanum wheels make the base a full holonomic platform, meaning it can rotate and move in any direction immediately (x, y and ϕ). This can be illustrated for one wheel with use of Figure 2.2a. If the movement of the wheel in the horizontal plane is forced in a desired direction, there is always a combination of roller and wheel velocities that ensures smooth motion in that direction without slipping. This also implies that moving the wheel in the direction of the rollers result in zero angular velocity of the wheel and vice versa. By combining four individual actuated wheels in a rigid platform (in a correct orientation) motion in any direction can be realized. Each actuator generates a wheel torque that results into a force working on the platform in the direction of the roller axis, as illustrated in Figure 2.2b. The combination of the four wheel forces result in a force in x and y direction and a torque in ϕ direction on the platform. These forces and torque combined determine the rotation and direction of movement of the base platform.



(a) Velocity decoupling mecanum wheel.

(b) Forces on the base generated by the wheels.

Figure 2.2

In order to realise these base motions the lower level software should be implemented. The lower level software in this thesis is referred to as the software that controls the actuator outputs to follow desired references gain by the higher level software. The higher level software is the software part that can be used on multiple robots. It can be seen as the brain of the robot that determines which and how tasks should be performed by the robot. In case of the base the higher level software determines a reference velocity for the base and the lower level software should ensure this reference is followed by controlling the actuators. The lower level software should be implemented in Orocos [7] (Open Robot Control Software) running in the ROS (Robot Operating System) environment on a PC of SERGIO. The Orocos Project is an open source project which makes it possible for other people to apply the same controllers.

The hardware components used in the base control loop are shown in Figure 2.3. Through ethercat the PC can communicate with a custom I/O board that is programmed to handle all in and outputs of the base. The board retrieves the encoder counts from the encoder connected to the DC motor. A voltage signal is send by the I/O board to the amplifier which is a current controller that controls the motor. The DC motor (Maxon [5]) has a maximum nominal torque of 0.101 Nm

and a maximum torque of 1.2 Nm which is transferred to the wheels through a gearbox. A list of specifications of the hardware can be found in Appendix A.1. The challenge in the controller design is finding the best structure that combines the sensor data and reference velocities to generate the actuator commands that ensure tracking of the reference velocities.

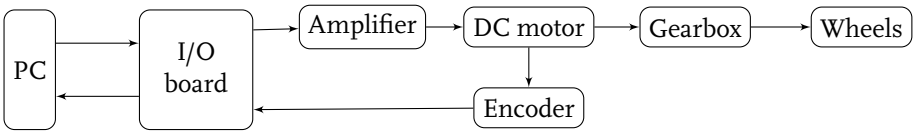


Figure 2.3: Hardware components in the control loop of the base.

2.2 Related work

In literature holonomic driven platforms are widely discussed. A distinction can be made between quasi and full holonomic platforms. Where a full holonomic platform can move in any direction immediately a quasi holonomic platform has to orient its wheels first. In this section the force and torque transfers to move the base in the desired direction are extracted form literature. These transfers can be used in a method to detect wheel slip which can be tested on the robot. Several control methods are found in literature and shortly elaborated.

2.2.1 Transfer robot and wheel frame

In Muir and Neuman [16] the kinematic relation between wheel and robot velocities for several base designs are described, as for the mecanum wheeled base,

$$\vec{\omega}_w = \boldsymbol{T}_{Rw}\vec{v}_R, \tag{2.1}$$

$$\begin{bmatrix}\omega_1\\ \omega_2\\ \omega_3\\ \omega_4\end{bmatrix} = \frac{1}{r_w}\begin{bmatrix}-1 & 1 & l_{x,b}+l_{y,b}\\ 1 & 1 & l_{x,b}+l_{y,b}\\ 1 & -1 & l_{x,b}+l_{y,b}\\ -1 & -1 & l_{x,b}+l_{y,b}\end{bmatrix}\begin{bmatrix}v_x\\ v_y\\ \omega_\phi\end{bmatrix}, \tag{2.2}$$

with $\vec{\omega}_w$ the rotational wheel velocities, $\vec{v}_R$ the robot velocities per direction, $\boldsymbol{T}_{Rw}$ the transfer matrix from robot to wheel velocities, r_w the wheel radius and the other parameters as shown in Figure 2.2. By taking the pseudo inverse of the transfer matrix the relation from wheel velocities to robot velocities is obtained,

$$\vec{v}_R = (\boldsymbol{T}_{Rw})^\dagger\vec{\omega}_w. \tag{2.3}$$

In a similar matter the dynamic relation between wheel torques and the forces, and moment

working on the robot platform are,

$$\vec{\tau}_w = \boldsymbol{T}_{\tau,Rw}\vec{F}_R, \tag{2.4}$$

$$\begin{bmatrix}\tau_{w,1}\\ \tau_{w,2}\\ \tau_{w,3}\\ \tau_{w,4}\end{bmatrix} = \frac{r_w}{4}\begin{bmatrix}-1 & 1 & \frac{1}{l_{x,b}+l_{y,b}}\\ 1 & 1 & \frac{1}{l_{x,b}+l_{y,b}}\\ 1 & -1 & \frac{1}{l_{x,b}+l_{y,b}}\\ -1 & -1 & \frac{1}{l_{x,b}+l_{y,b}}\end{bmatrix}\begin{bmatrix}F_{R,x}\\ F_{R,y}\\ M_{R,\phi}\end{bmatrix}, \tag{2.5}$$

with $\vec{\tau}_w$ the wheel torques, $\vec{F}_R$ the forces and moment working on the robot and $\boldsymbol{T}_{\tau,Rw}$ the transfer matrix from robot forces to wheel torques. Again in the pseudo inverse can be used to calculate the robot forces, $\vec{F}_R = \boldsymbol{T}^{\dagger}_{\tau,Rw}\vec{\tau}_w$. Note that this system is overdetermined because there are four actuators and only three degrees of freedom.

2.2.2 Slip

The transformation matrix is based on the fact no slip occurs and the wheels stay in contact with the floor at al times. A method to detect slipping of wheels is presented in Rojas and Förster [18]. The measured wheel velocities $\vec{\omega}_w$ result in the calculated robot velocities $\vec{v}_R = \boldsymbol{T}^{\dagger}_{Rw}\vec{\omega}_w$ and the following estimate of the wheel velocities

$$\hat{\omega}_w = \boldsymbol{T}_{Rw}\boldsymbol{T}^{\dagger}_{Rw}\vec{\omega}_w. \tag{2.6}$$

The estimated wheel velocity is always equal to the actual wheel velocity if the product of the transformation matrix with its pseudo inverse is equal to the identity matrix $\boldsymbol{I}$. For a four wheeled holonomic base this does not hold,

$$I - \boldsymbol{T}_{Rw}\boldsymbol{T}^{\dagger}_{Rw} = \frac{1}{4}\begin{bmatrix}1 & -1 & 1 & -1\\ -1 & 1 & -1 & 1\\ 1 & -1 & 1 & -1\\ -1 & 1 & -1 & 1\end{bmatrix}. \tag{2.7}$$

The way to detect slip is by checking the following condition,

$$\omega_{w,1} - \omega_{w,2} + \omega_{w,3} - \omega_{w,4} = 0. \tag{2.8}$$

If this condition does not hold it means that either the base platform is deforming or slip occurred in the base. It is assumed a base platform is design as a rigid body structure and therefore deformation is negligible and this condition can be used to detect slip. Note that slip can still occur unnoticed if traction is lost in one of the directions of motion and all wheels slip the same amount in that direction, which is unlikely to occur.

To prevent or suppress wheel slip several methods are found in literature. One way is to mechanically couple the four wheels with differential drives and 3 actuators to drive the system[6]. Another way is to modify the measured wheel velocity vector with a factor dependent on the slip detection factor[18]. In a suspended base this might be obsolete and therefore the slip condition in (2.8) will be used to investigate the level of slip.

2.2.3 Control methods

The overall control problem of a holonomic base is dependent on the goal it is used for. In every case there is a reference velocity or position expressed in the world or euclidean space that needs to be tracked. In this thesis the focus is on the low level control, meaning the only available information of the robot position is gain from sensors that track the wheel orientation. The two main control structures are controlling the wheels individually or control the three DOF of the robot. The main methods used is PID (Proportional, Integral and Derivative) control. The controller parameters can be manually tuned [12, 18]. In Gloye et al. [11] the parameters are calibrated with the use of a gradient reinforcement learning method and in Han et al. [13] a the parameters are tuned using a fuzzy system. Another control method is using a non linear model of the complete drive train including motor dynamics and use optimal[14] or adaptive[15] control based on the Lyapunov stability theorem.

What all methods mentioned here have in common is none of them use tuning methods in the frequency domain, which is used on AMIGO[8]. The control structure chosen on AMIGO is a position controller that controls each wheel separately. Three reasons to prefer position over velocity control are wheel slip leading to instability of the robot, noise on the feedback signal due to differentiation of the encoder signal and the static error that remains on the velocity without using integral action. The decoupled control (control per wheel) is chosen over coupled control because of instability in coupled control caused by wheel slip. In addition to the feedback controller a velocity feed forward is added on the three input velocities, meaning in the coupled form. Note that this control structure is implemented in 2011 and the drive train of AMIGO is changed a little over the years and with it the controller.

2.2.4 Contribution

In the case of the suspended mecanum wheeled base of SERGIO it is interesting to see if the effect of wheel slip is excluded or reduced to a minimal level such that it can be neglected. In this case other control structures can be implemented, tested and compared. Based on experimental results the most suited of the investigated control structures is chosen and will be implemented on SERGIO.

2.3 Identification

In order to design a decent controller for the platform it should be identified first. As discussed before this can be done with the use of a parametric model of the system. The disadvantage is there are always modelling errors and unknown parameters, such that experiments are necessary to identify and possibly adjust the model. Another identification method is with the use of a FRF measurement which result in a decent MIMO model of the system to design a controller. The advantage is that there is now need of a parametric model but the input to output transfers are used.

2.3.1 Method

To measure the MIMO FRF a three points method is used. The three point method allows for a controller to be used during the measurements such that they can be performed safely without damaging the system or the environment. Besides safety a reference velocity can be set to overcome the non-linearities caused by friction and play resulting in a better FRF estimate.

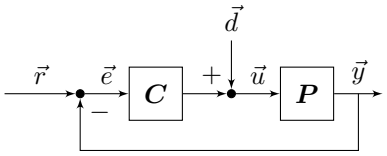


Figure 2.4: FRF measurement scheme.

During the FRF measurement a white noise $\vec{d}$ is added to the plant input, as shown in Figure 2.4. The input signal $\vec{u}$, error $\vec{e}$ and disturbance $\vec{d}$ are logged to calculate the FRF. From the signals $\vec{d}$ and $\vec{u}$ the sensitivity of the plant can be determined with use of the function `tfestimate` implemented in MATLAB[2]. This function computes the power spectral density of $\vec{d}$ and the cross power spectral density of $\vec{d}$ and $\vec{u}$ to compute the transfer function which is the sensitivity,

$$\mathcal{S}_{d\to u} = (I + \boldsymbol{PC})^{-1}. \tag{2.9}$$

Using the same method the process sensitivity can be determined as it is the transfer from disturbance to error,

$$\mathcal{PS}_{d\to e} = -(I + \boldsymbol{CP})^{-1}\boldsymbol{P}, \tag{2.10}$$

resulting in the MIMO model of the plant,

$$\boldsymbol{P} = -\mathcal{PS}(\mathcal{S})^{-1}, \tag{2.11}$$

Note that for a model with four inputs it requires at least four measurements to determine the MIMO FRF. Applying noise to all four wheels in the same measurements causes correlation between the different signals resulting in a bad estimation of the transfer functions. By applying a disturbance to only one of the wheels the transfer from this disturbance to to all the inputs and errors can be computed, resulting in one row of the sensitivity functions. Repeating this measurement for each wheel results in the complete sensitivity matrices.

2.3.2 Measurement FRF in the air

In order to start measuring an FRF on the floor a low gain controller should be implemented. Because of safety reasons the controller is first measured with the base platform placed on blocks, such that the wheels can freely rotate in the air. With the wheels in the air the system can be seen as four SISO (Single Input Single Output) systems and the FRF can be measured for each wheel individually.

The four resulting frequency responses are shown in Figure 2.5 and their corresponding sensitivity and process sensitivity with coherence. From both coherence results it can be concluded that the FRF of the plant is a good estimate in the range between 2 Hz and 100 Hz. The bode plot shows that

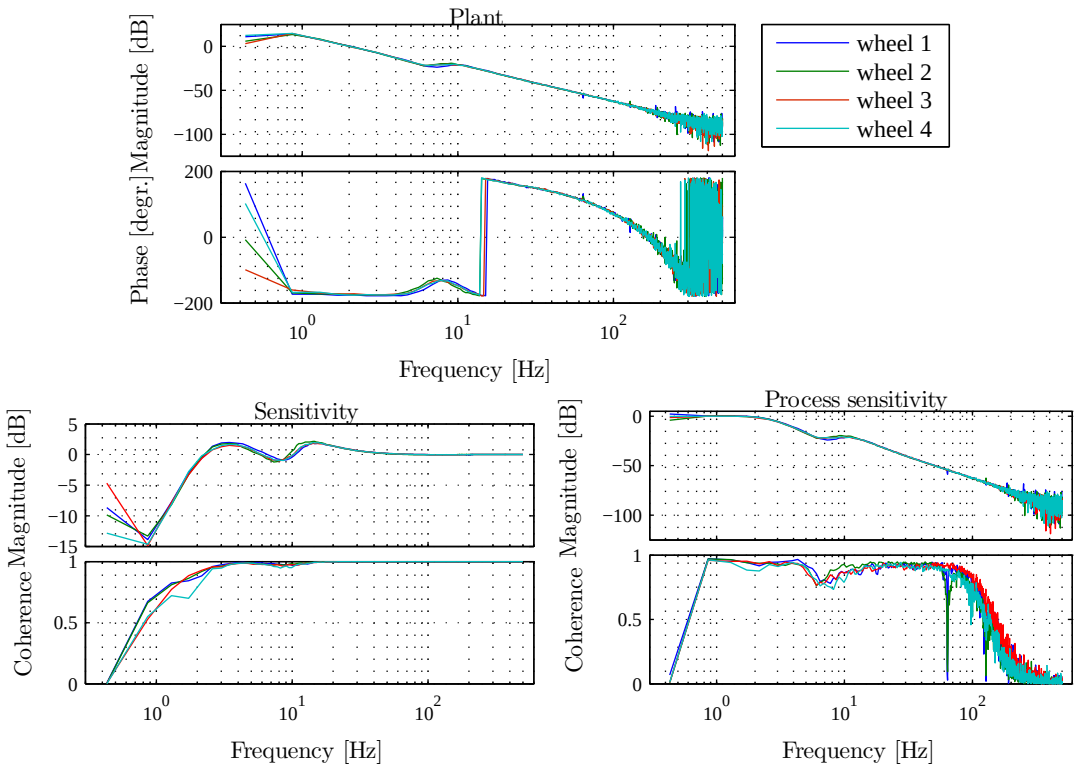


Figure 2.5: Results measuring an FRF per wheel with the wheels hanging in the air such that they could freely rotate (resolution of 0.431 Hz). The top plot shows the transfer from wheel torque to wheel angle. The bottom plots show the sensitivity and process sensitivity and their corresponding coherence.

all four wheels result in identical plant dynamics as expected by the similar design. The FRF show a compliance in the drive train at a frequency of 8 Hz. This decoupling is probably caused by the play in the drive train from motor to wheel. It can either be in the gearbox or the drive shaft which together connect the wheel to the motor. One of the reasons for the phase drop in the higher frequency ranges is the delay caused by communication between the PC and the I/O board through EtherCAT. This is one of the downsides of using a PC instead of a dedicated chip to control the hardware.

2.3.3 Measurement FRF on the floor

In operation the base will drive around on the floor which will bring some extra dynamics to the system by the moving mass of the robot, friction forces by contact with the floor and cross relations from one wheel to another. Therefore the base is placed on the floor and the MIMO FRF is measured. Based on the air measurements a controller is implemented that controls the individual wheel orientations. A rotational reference velocity is applied such that both wheels and rollers have a constant velocity and long measurements can be taken because the robot stays in place. Experimentally the rotational speed and injected noise level are manually tuned by looking at the coherence of both sensitivity and process sensitivity. The resulting velocity is 0.14 $\frac{\text{rad}}{\text{s}}$ corresponding to a wheel speed

of 1.0 $\frac{\text{rad}}{\text{s}}$ and the noise is set to 1.0 Nm wheel torque.

The results of the measurements are shown in Figure 2.6 and more detailed results can be found in Appendix A.2. Note that only the sensitivities and coherence of the diagonal term of one wheel is shown, because these are similar for the other wheels as can be found in the appendix. The Coherence show good results in the frequency range from 20 Hz up to 200 Hz. In the frequency range from 2 up to 20 Hz the coherence of the process sensitivity is low which is probably caused by non-linearities in the system, like the friction of the floor. Attempts are made to improve the coherence by reducing the control effort, changing the noise level, velocity direction and speed, but none of them result in a better coherence over the complete range of frequencies. For the off diagonal therms the coherence is really small which is probably caused by the damping of the floor and plastic rollers and the suspension of the base platform, which result in a bad signal to noise ratio. In order to improve the accuracy of the FRF other methods should be investigated and tested. From this point on however the FRF measured will be used as the model of the base platform.

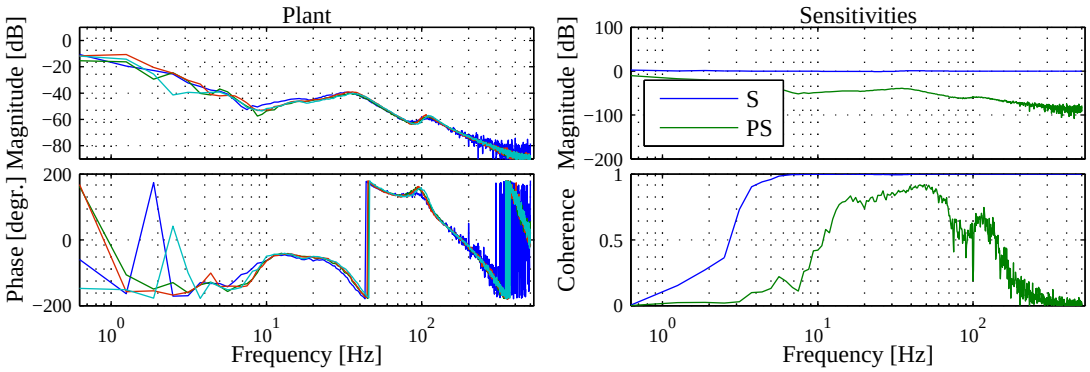


Figure 2.6: Results measuring an MIMO FRF on the floor (resolution of 0.625 Hz). The diagonal terms of the Plant, wheel torque to wheel orientation are shown in the left. On the right the sensitivities of one of the diagonal terms is shown with their coherence.

The diagonal terms of the FRF measured on the floor shows two differences with the FRF measured in the air. First the change of mass line at a frequency of 11 Hz is larger because of the larger moving mass when rotating on the floor. Secondly it shows a compliance at a frequency of 90 Hz. A FRF measurement with a forward reference velocity did not show this compliance effect at 90 Hz, as shown in the appendix, meaning it is probably caused by the rollers as they do not rotate in forward direction. Another remarkable effect showed when measuring a FRF on a non compliant floor instead of a compliant floor. A non compliant floor causes the change in mass line at 20 Hz to shift to the left. This is something that should be accounted for in the controller design such that the base is stabilized on both compliant as non compliant floors.

2.4 Feed back control

The low level controller should be implemented using the interface as shown in Figure 2.7, where the subscript R is in the robot space, subscript w is in the wheel space, $\vec{v}$ are velocities, $\vec{\tau}$ are torques and $\vec{q}$ contains positions and orientations relative to start-up. Within this interface the control can

freely be chosen with the goal to track the reference velocities $\vec{v}_R$ as good as possible. In this section coupled control is compared with decoupled control and velocity control is compared with position control. At the end the final control structure is presented and stability tests are shown.

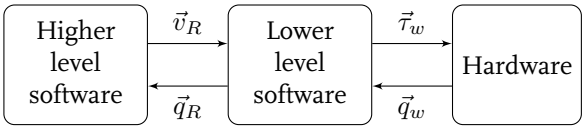
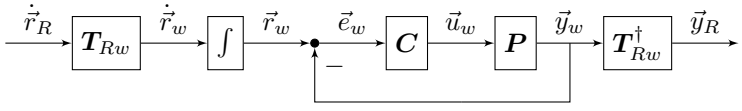


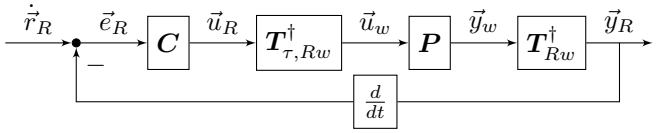
Figure 2.7: Interface of the low level software.

2.4.1 Coupled versus decoupled feedback control

In the current interface the reference that should be tracked are the velocities of the robot in the x, y and ϕ direction, which is a commonly used method. From the reference side the velocity error of these three directions should be controlled leading to the control structure shown in Figure 2.8b. The available hardware however allows to set the torque of the individual wheels and it returns the orientation of each wheel. This means that from the hardware side controlling the individual wheel errors seems the obvious way to go, see Figure 2.8a.



(a) Coupled control on position error.



(b) Decoupled control on velocity error.

Figure 2.8: Two possible control structures for the base platform.

The presented control structures assume a decentralized controller can be used on the system. Decentralized control means that the dominant transfers are placed on the diagonal elements of the plants by use of transfer matrices. A different controller for each diagonal element is then applied. In such a structure the transfer of the off-diagonal elements must be low such that they can be neglected or seen as a disturbance. A method to test if a decentralized controller is possible is the RGA (Relative Gain Array) analysis presented in Skogestad and Postlethwaite [19].

The RGA is a measure of interactions and is defined as,

$$\mathrm{RGA}(\boldsymbol{P}) = \Lambda(\boldsymbol{P}) \triangleq \boldsymbol{P} \times (\boldsymbol{P}^{-1})^T, \tag{2.12}$$

where $\times$ denotes element-by-element multiplication. A plant is likely to be controllable with a decentralized controller if the diagonal elements of the RGA are close to one and the off-diagonal elements are close to zero at frequencies around the closed-loop bandwidth. A simple measure for selecting

parings is the RGA number,

$$\text{RGA number}(\boldsymbol{P}) \triangleq ||\Lambda(\boldsymbol{P}) - \boldsymbol{I}||_{sum}. \tag{2.13}$$

From this equation it can be seen that an RGA number close to zero is desired because the RGA matrix should be close to identity. The RGA number for the measured FRF from wheel to wheel is shown in Figure 2.9. It shows a low RGA number in the frequency range from 20 Hz up to 300 Hz, meaning decentralized control in this region is possible. However in the frequency region bellow 10 Hz the RGA number is large meaning decentralized control is difficult in this region.

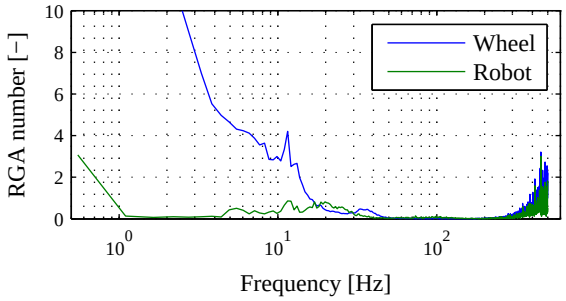


Figure 2.9: RGA number of the system in wheel and robot space.

The plant can be decoupled to get the three directions of freedom on the diagonal with the decoupling matrices presented in Section 2.2,

$$\boldsymbol{P}_R = \boldsymbol{T}_{Rw}^{\dagger} \boldsymbol{P} \boldsymbol{T}_{\tau,Rw}. \tag{2.14}$$

The diagonal elements of the FRF from the decoupled system can be found in Appendix A.2. The RGA analysis of this system results in the RGA numbers presented by the green line in Figure 2.9. The RGA numbers are close to zero for almost the complete measured frequency range which means decentralized control can be used in this range. It clearly shows that for the model gain from the measurements the decoupled system is the most suitable system for decentralized feedback control. This does not mean that the decentralized control in coupled form cannot be used to stabilize and control the plant.

A holonomic base with four wheels results in an overdetermined plant, because of the four actuators controlling three degrees of freedom. The most likely reason why decoupled control is the preferred way to go by RGA analysis can be found with use of the slip detection theorem presented in Section 2.2. If the system is controlled on the wheel positions and the error of all but one wheel is zero and the controller gains are high this will result in an internal tension. The controller on the wheel with the larger error wants to reduce it and is therefore sending out a motor torque. The only way to bring the system to a resting state where all errors converge to zero is by slip or lifting the wheel from the ground. On AMIGO this control structure is successfully used because of two reasons, first the wheels of AMIGO slip easily because of the stiff design and secondly there are no integrators used which means no increasing control output on a static error. So it should be kept in mind that controlling individual wheel positions high gains at low frequencies should be avoided.

Besides the model based preference for decoupled control there is also a safety matter that should be taken into account. The FRF is measured on a flat floor. So what happens when one of the wheels

loses contact with the floor. In case of coupled control each wheel has its own controller and stability can still be guaranteed if the designed controller also stabilizes the wheel in the air. For a decoupled controller the wheel that is losing contact with the ground will still get the same controller output as the other wheels while the resistance is reduced drastically resulting in fast wheel spin. So even though decoupled control is preferred by the model it does not necessary mean it is the best choice for each application. Because SERGIO is operating in a household environment where the floor is flat and there are only small obstacles present it is assumed the wheels will stay in contract with the floor at all times. Note that driving over an obstacle can lead to a split second of losing contact but it is assumed this will not cause any problems and therefore the decoupled control is the best way to go.

2.4.2 Position versus velocity feedback control

Dependent on integrating the reference signal or an differentiating the feedback signal the controller could work on the position or velocity error respectively. In this subsection some of the up and downsides of position and velocity control are given and a time domain comparison is shown.

Depending on the controller implementation there could be no real difference between controlling the position and velocity error. By using a high gain controller that ensures good reference tracking and applying a smooth reference trajectory as input, both position and velocity controllers will follow this input smoothly. Differences show up when the system has to work at its limits. If for instance a reference velocity should be tracked that the actuators cannot follow the position controller will result in an increasing error over time while the velocity error is always limited by the applied reference. In case of a service robot safety is important and it should be ensured there are no references applied that the robot cannot handle. A downside in velocity control is differentiating the sensor signal. This leads to the amplification of sensor noise. However this can be accounted for in the controller by a low pass filter.

Because there are no real preferences between position or velocity control in this application both versions are tested on the robot by applying a reference trajectory and measuring the velocity error. As a reference trajectory all possible combinations of the three directions of freedom are used. This means applying a velocity in the three directions separately, all combinations of two directions and all three directions at once. In all directions both a positive and a negative velocity is applied resulting in a set of 14 reference velocities. The linear velocities are set to 0.75 m/s and the rotational velocities to 0.8 rad/s and each reference trajectory has a constant acceleration and deceleration of 1 s with a constant velocity in between for 3 s. The reference velocities are chosen such that they are in the application range of the domestic service robot.

The resulting error of one of the measurements is shown in Figure 2.10. The error referred to here is the difference between the reference velocities obtained from the higher level software and the velocities calculated out of the measured wheel positions,

$$\vec{e}_R(t) = \vec{r}_R(t) - \frac{d}{dt}\boldsymbol{T}_{Rw}^{\dagger}\vec{y}_w(t) \tag{2.15}$$

The figure shows that both position and velocity control can follow the reference pretty accurately. However the position controller has a longer settling time and a larger error at the acceleration and deceleration phase. This is probably caused by the fact that the controllers have a different bandwidth

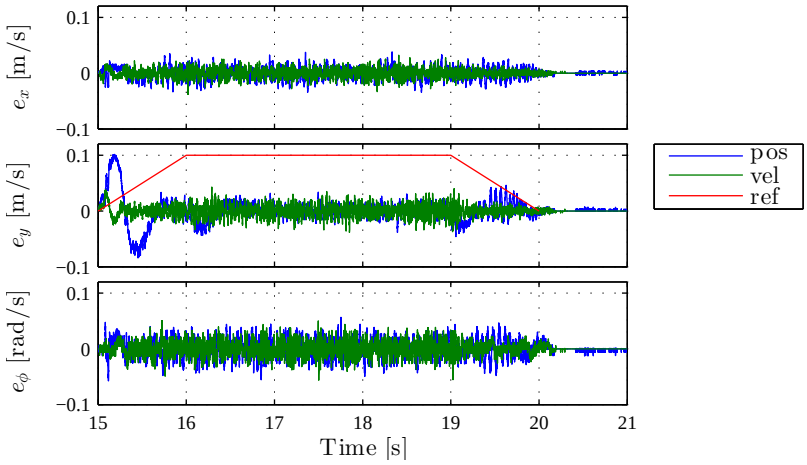


Figure 2.10: Velocity error in all three directions of both position and velocity decoupled controller when following a reference velocity of 0.75 m/s in y-direction of which the shape is shown in red.

because modulus and phase margins should be met. An nice effect of the velocity controller is that with a zero reference and the robot standing still the error nicely converges to zero as will the control outputs. For the position control there is always a small remaining offset.

To get an idea about the overall error difference between both controllers the Euclidean norm of the error for all three directions is calculated,

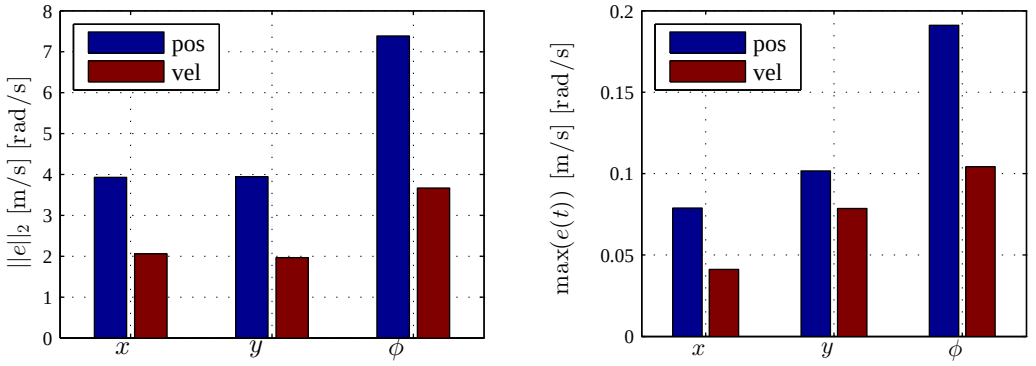
$$||e_{direction}||_2 = \sqrt{\sum_{k=0}^{k=end} e^2_{direction}(k)}, \tag{2.16}$$

where k represents one sample. From the results shown in Figure 2.11a it can be concluded that for the implemented controllers the one based on velocity control results in an overall smaller error. In addition to the norm also the maximum error over the complete data set is computed as shown in Figure 2.11b which is also lower for velocity control in all three directions. Therefore velocity control will be chosen over position control for SERGIO.

2.4.3 Stability

With the chosen control structure the last task is tuning the controller and ensuring stability of the MIMO system. A method to ensure stability of decentralized controlled MIMO systems is described in Skogestad and Postlethwaite [19]. There are three requirements that should be met in this theorem. The first criteria states that the plant $\boldsymbol{P}$ is stable. However this criterion is only required such that the system remains stable when one of the loops is taken out of service. In case of the base of SERGIO the controller is only active with all loops in service, meaning this requirement is not necessary. However the plant response as show in Appendix A.2 shows that the system is stable without controllers.

The second requirement states that each individual controlled loop by itself is stable. This means



(a) Euclidean norm of the error per direction.

(b) Maximum error per direction.

Figure 2.11: Comparison between velocity and position controller.

SISO stability for the diagonal components of the decentralized plant. The controllers designed to meet this requirement consist of a gain, an integrator and a low pass filter with settings shown in Table 2.1. The open loop response and Nyquist are shown in Figure 2.12 that show a stable response of the system. Note that no notch is used to increase performance by suppressing the high peak at 35 Hz because the exact location of this peak can change when driving over different floors and in different directions discussed in Appendix A.2.

	x	y	ϕ
Gain	180	180	80
Integrator zero frequency	30 Hz	30 Hz	7 Hz
Low pass pole frequency	200 Hz	200 Hz	200 Hz
Low pass damping	0.7	0.7	0.7
Bandwidth	3.3 Hz	3.1 Hz	3.9 Hz
Modulus margin	4.8	5.2	4.8
Phase margin	40°	32°	33°

Table 2.1: Controller settings and corresponding bandwidth and margins for all three directions.

With the first two checks fulfilled there are two theorems to finally ensure stability of the complete MIMO system. One theorem is used for a plant with diagonal dominance at low frequencies and the other for a plant with diagonal dominance at crossover frequency. Figure 2.9 shows that at cross over frequency the diagonal dominance is a little less than in the other frequency regions and therefore the first theorem is used to test stability. This theorem states that the following criterion should be met,

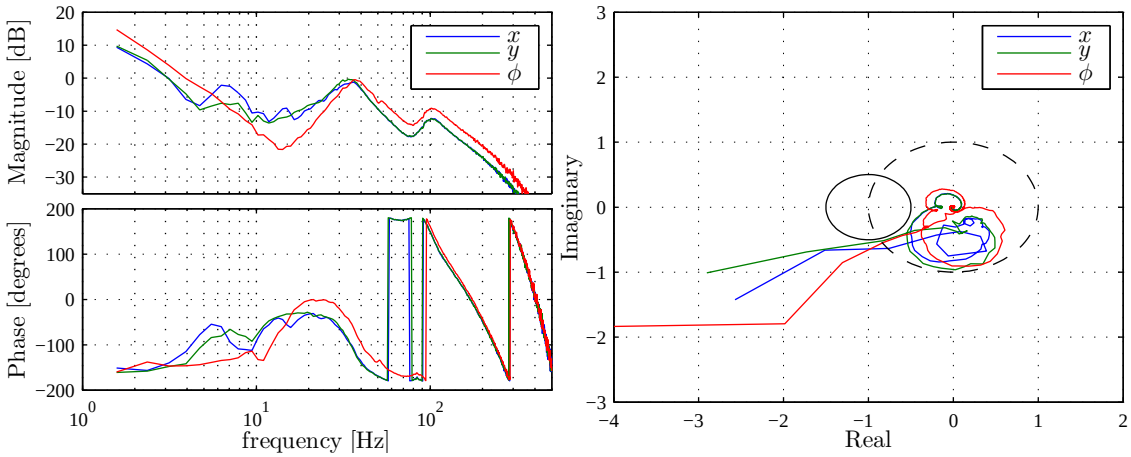
$$\bar{\sigma}(\tilde{\boldsymbol{T}}) < \frac{1}{\mu(\boldsymbol{E}_s)}, \quad \forall\omega, \tag{2.17}$$

with

$$\tilde{\boldsymbol{T}} = (\boldsymbol{I} + \tilde{\boldsymbol{P}}\boldsymbol{C})^{-1}, \tag{2.18}$$

$$\boldsymbol{E}_T = (\boldsymbol{P} - \tilde{\boldsymbol{P}})\tilde{\boldsymbol{P}}^{-1}, \tag{2.19}$$

where $\tilde{\boldsymbol{T}}$ is the closed loop containing only the diagonal components (zero on the off-diagonal terms),



(a) Bode. (b) Nyquist.

Figure 2.12: Stability of the diagonal terms.

$\boldsymbol{E}_T$ is an 'interaction measure', $\tilde{\boldsymbol{P}}$ contains only the diagonal elements of the plant $\boldsymbol{P}$, $\bar{\sigma}$ is the maximum singular value and $\mu()$ is the structured singular value with respect to the diagonal structure of $\tilde{\boldsymbol{T}}$. Figure 2.13 shows that this criteria is met for the complete frequency range. It can now be concluded that the MIMO system with the implemented controllers is stable, under the assumption that the identified model in Section 2.3 is a good representation of the system.

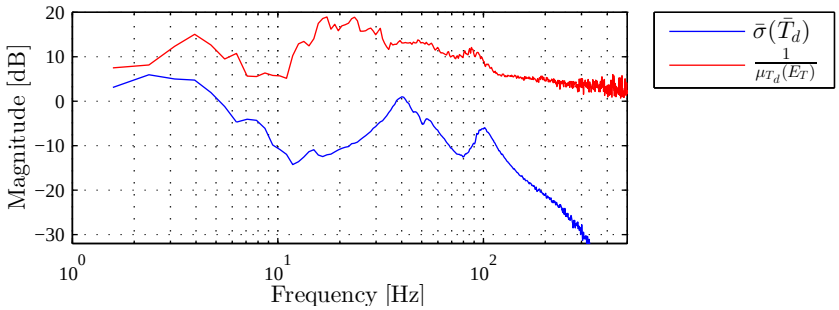


Figure 2.13: Stability of the MIMO system.

2.5 Feed forward

Feed forward is a powerful tool to improve reference tracking of a system. A commonly used feed forward compensation used in motion control includes friction compensation and an acceleration term to compensate for the mass inertia. A possible feed forward for the service robot can be of the following form,

$$\vec{u}_R = \boldsymbol{K}_a \ddot{\vec{r}}_R + \boldsymbol{K}_v \dot{\vec{r}}_R + \boldsymbol{K}_c \operatorname{sign}(\dot{\vec{r}}_R), \tag{2.20}$$

where $\boldsymbol{K}$ are diagonal matrices containing the gains per direction and $\vec{r}$ the reference trajectory in the three DOF of the robot. These parameters can be tuned per direction by applying a reference trajectory containing constant acceleration and velocity parts. The form of the acceleration reference in is shown in Figure A.8 by the black line, note that the magnitude is scaled to show the effect of the feed forward. The top plot shows the tuning results of the feed forward in y direction with use of the position error. The effect of viscous friction is not visible in the error and therefore not included. In the bottom plot the velocity error is shown. It shows that the effect of the applied feed forward on the velocity error is minimal.

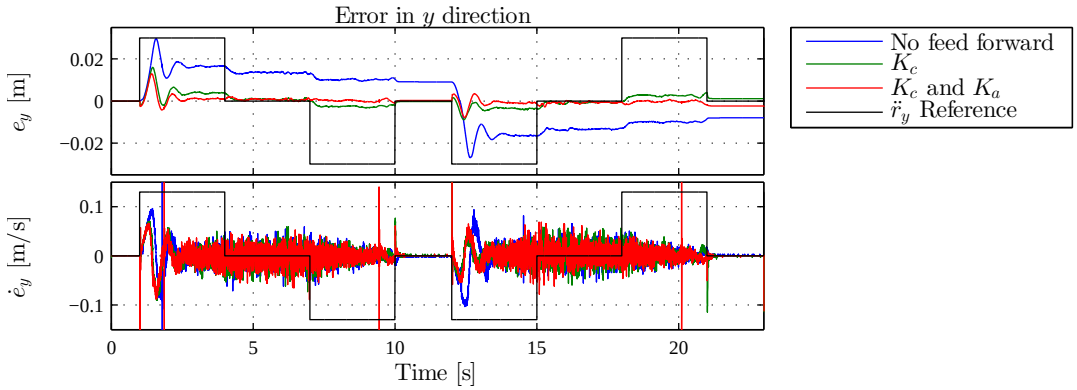


Figure 2.14: Effect of the feed forward in y direction on the position and velocity error. The black line shows a scaled version of the acceleration reference used in y direction.

In Appendix A.3 the tuning results of the other directions are shown. The resulting gains are $\boldsymbol{K}_a = \text{diag}\begin{bmatrix}70 & 70 & 7\end{bmatrix}$ and $\boldsymbol{K}_c = \text{diag}\begin{bmatrix}20 & 80 & 28\end{bmatrix}$. The inertia components in x and y direction are identical as expected because they represent the moving mass. The coulomb friction compensation shows a huge difference between the x and y directions which can be declared by the friction in the roller. The forward direction of motion is the only direction in which the rollers do not rotate. A possible better way to implement the friction compensation would be of the form,

$$\vec{u}_w = \boldsymbol{T}_{\tau,Rw}\boldsymbol{K}_a\ddot{\vec{r}} + \boldsymbol{K}_{v,w}\vec{\omega}_w + \boldsymbol{K}_{c,w}\,\text{sign}(\vec{\omega}_w) + \boldsymbol{K}_{v,r}\vec{\omega}_r + \boldsymbol{K}_{c,r}\,\text{sign}(\vec{\omega}_r), \tag{2.21}$$

where the friction is compensated per wheel based on wheel velocities $\vec{\omega}_w$ and roller velocities $\vec{\omega}_r$. However with the minor improvements showing in the velocity error this is not further investigated in this thesis.

The feed forward compensation presented in this section is a great tool to improve reference tracking on position level. The interface of the base of SERGIO requires to track velocity references and therefore the feed forward is not implemented. In addition it should be investigated what happens if the individual tuned components are combined on the system. If the coulomb friction is indeed dependent on wheel and roller velocity the coulomb term in the feed forward law in (2.20) will probably result in the wrong values when motion in multiple directions is combined.

2.6 Performance

The main goals in the base design was to reduce wheel slip and make it easier to overcome small obstacles in a domestic environment, as stated in Section 1.2. In order to determine if these design goals are met the following experiments are performed,

- *Corridor experiment*: Drive in forward/sideways direction with a constant velocity.
- *Bump experiment*: Drive over a small bump (like a doorstep) in forward direction or under a small angle approach in forward direction.

The corridor experiments are performed on a non compliant concrete floor to emphasize the effect of wheel slip. During these experiments the velocity is measured on three different ways,

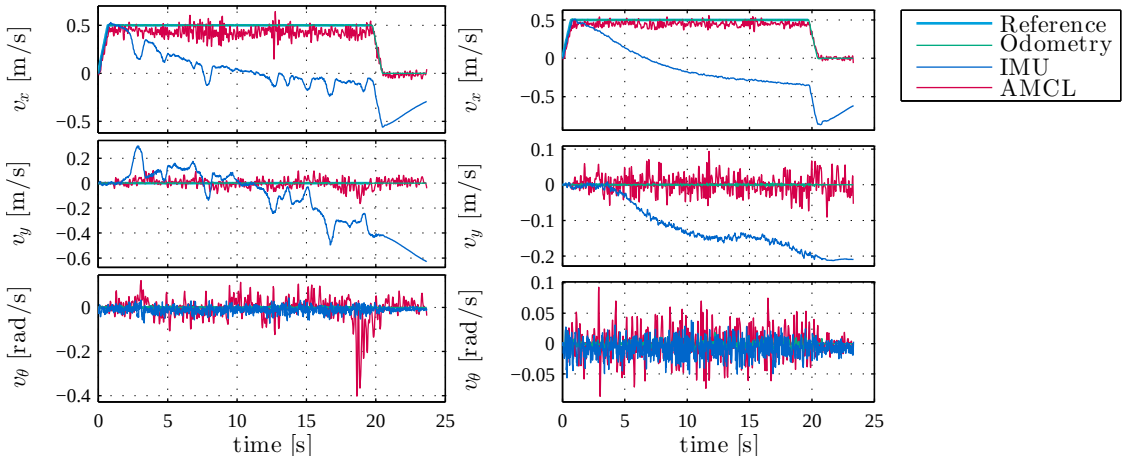
- *Odometry*: The Odometry data contains the robot position with respect to its initial position calculated out of the encoder data of the wheels. The gain data should be differentiated to get the robots velocities.
- *AMCL (Adaptive Monte Carlo Localization)*: An algorithm which estimates the position of the robot based on a known map of the world, data obtained by combining the LRF (Laser Range Finder) and Odometry data. This algorithm is used on AMIGO to estimate its location in a world model. The position data should be differentiated to obtain the robots velocity.
- *IMU (Inertial Measurement Unit)*: This is an external measurement unit mounted on the robot for these experiments. The IMU measures orientation and acceleration of the device. The acceleration data should be integrated to obtain the robots velocities.

All measured velocities will be compared to the reference velocity to obtain the error. It is expected that the error obtained by the Odometry data is small because it is also used by the base to compute the error. All experiments are performed with both AMIGO and SERGIO to show the difference and improvements. In order to make a good comparison both robots use t control structure where the velocities of the individual wheels are controlled.

2.6.1 Data processing

The results from an corridor experiment with a velocity of $0.5\ \frac{m}{s}$ in forward direction for both robots is shown in Figure 2.15. It shows the estimated velocity from the three different methods. The results show that the reference and Odometry data show a similar velocity profile as expected. The AMCL algorithm shows a lot of disturbance in the measured signal which is probably caused by the differentiation of the position signal. The AMCL algorithm will therefore not be used to analyse performance. The IMU signal shows a offset disturbance in the acceleration which lead to a slope in the integrated signal as shown by the velocity in x and y direction. A constant offset would lead to a constant steepness, the velocity in y direction however shows a varying steepness over time. In order to compare the estimated errors from different sensors and measurements some filtering is needed. In Figure 2.16 the IMU velocity error is shown for the discussed experiment. It is chosen to use

a high pass filter (0.2 Hz) to filter out the low frequent disturbance in the IMU acceleration offset. The filtered error shows a fluctuation around zero without the varying slope present. Note that the correction by the filter cannot perfectly filter out the offset error generated by the IMU and therefore results should be interpreted with care. The important part is that it clearly shows the difference between AMIGO and SERGIO.



(a) AMIGO. (b) SERGIO.

Figure 2.15: Reference and measured velocity of the robots in the three directions of movement while moving with a constant velocity of $0.5\ \frac{m}{s}$ in forward direction.

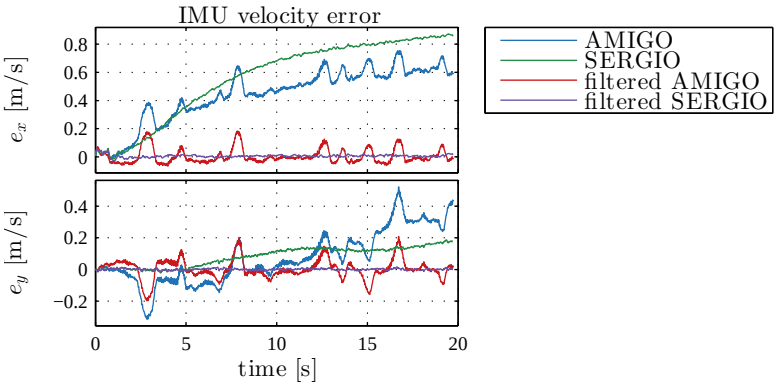


Figure 2.16: Velocity error by IMU both filtered and non filtered for both robots with a reference velocity of $0.5\ \frac{\mathrm{m}}{\mathrm{s}}$ in forward diriction.

2.6.2 Results corridor experiment

The corridor experiment is performed multiple times for both robots with different reference velocities each measurement. The resulting error by Odometry and IMU when moving with a constant velocity of $0.5\ \frac{\mathrm{m}}{\mathrm{s}}$ for both robots is shown in Figure 2.17. The error by Odometry is very small for

both robots which means that the controllers ensure that the individual wheels follow the speed reference quit accurate. The IMU error shows the clear difference between the robots. For SERGIO the IMU error is close to zero which means the reference velocity is followed accurately. For AMIGO the IMU error clearly shows the undesired behaviour caused by wheel slip. Take for example the peaks occurring at 3 s in both x and y direction. Increase in error in forward direction means the robot is slowing down in this direction and the decrease in y direction means the robot is drifting to the side. In terms of forces an example of the slip effect is shown in Figure 2.18. The black arrows show the wheel forces without slip which summed up generate the force acting on the robot represented by the thick arrow in forward direction. If slip occurs at one of the wheels the corresponding force on the robot reduces as presented by the red arrow. This results in a change in direction as shown by the thick red arrow. The error plots clearly shows that this happens a lot on AMIGO and for SERGIO this is cancelled out by the suspension in the base. It is checked if the slip could be detected with use of (2.8). This resulted in a noisy value which is equal to the measurement noise which means it cannot be used to compensate or detect slip.

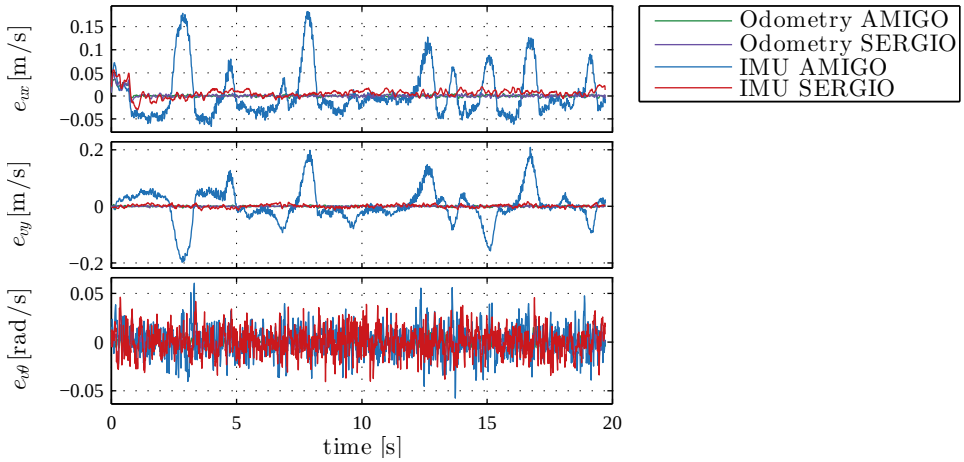


Figure 2.17: Velocity error for both robots with a reference velocity of $0.5\ \frac{\text{m}}{\text{s}}$ in forward direction.

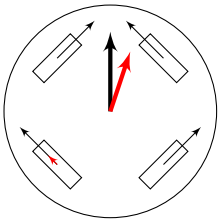


Figure 2.18: Forces applied by the wheels on the robot in forward motion in black and the thick arrow shows the sum of forces.

The corridor experiments is performed for multiple velocities ranging from $0.25\ \frac{\text{m}}{\text{s}}$ up to $1\ \frac{\text{m}}{\text{s}}$ of which the results can be found in Appendix A.4. The results of moving in a sideways direction for SERGIO are also shown in the appendix. The sideways experiment is not performed with AMIGO because it is similar to the forward experiment because of the orientation of the omni wheels. The sideways experiments with SERGIO show a minor offset in the rotation when moving in a sideways direction, which result in a arc like movement throughout the corridor instead of a straight line. A

possible reason could be a miss alignment in the wheels. It is assumed this minor error in rotation can be corrected by the navigation algorithm in the higher level software and is therefore not further investigated.

2.6.3 Results bump experiment

The bump experiment is performed in a straight forward approach and approached under a minor angle of approximately $25°$. In this case the IMU data will not be filtered with the high pass filter because it results in a distorted view of the reality. The results of the straight approach show two minor increases in disturbance in x and y velocity for both AMIGO as for SERGIO when the front wheels and the back wheels hit the bump, as can be seen in Appendix A.4. Both robots cope well with a bump in this experiment. The real difference shows itself when the bump is approached under the minor angle, as shown in Figure 2.19. For SERGIO the difference with no bump is an increase in the error on the velocities when the wheels have to overcome the bump. In case of AMIGO the robot starts drifting when one wheel hits the bump as can be seen at approximately 3 seconds. The wheel that hits the bump first is slipping on the edge of the bump and thereby pushes the robot sideways. Eventually the wheels are able to get the robot over the bump, but it has already drifted a lot sideways. The reason it does not drift in the straight approach is because two wheels approach the bump at the same time and together are able to lift the front of the robot over the bump. Note that again the velocity error by Odometry is close to zero meaning the wheels spin at the correct velocities.

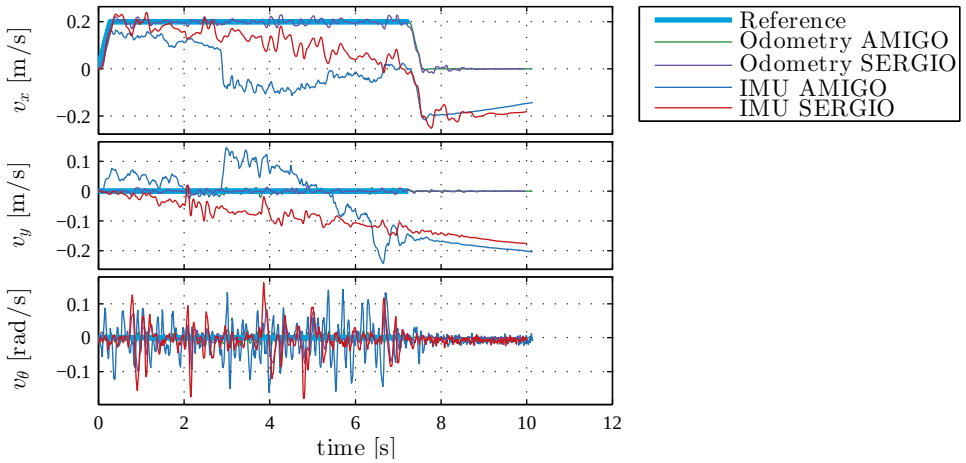


Figure 2.19: Velocity for both robots with a reference velocity of $0.2\ \frac{\text{m}}{\text{s}}$ in forward direction while encountering a small doorstep under an angle of $25°$.

2.6.4 Design validation

From the experimental results discussed in this section it can be concluded that two of the design requirements are met. First the mecanum wheels combined with suspension make it easier for SERGIO to overcome small obstacles. Secondly wheel slip is minimized by the suspension even on non-compliant floors.

2.7 Final implementation

From the time measurements combined with the RGA analysis and the interface of the lower level software it can be concluded that the best control structure for the base of SERGIO is decoupled velocity control. From the time domain measurements it can be concluded that slip in the base of SERGIO is minimal on both compliant and non-compliant floors, meaning there is no need for slip detection or compensation. With the suspended base of SERGIO in combination with a domestic environment it is assumed the wheels stay in contact with the floor at all times in the environment it is designed to operate in. In case one wheel loses contact with the floor the robot is in a state it should not be in and safety implementation should ensure the robot is stopped.

With these assumptions the final controller implemented on the robot using the Orocos real-time tool kit. The real time tool kit is used in the same way as it is used on AMIGO[8], using the 'rtt_control_components' library of the TU/e[17]. In this library all kind of components are defined that can be used to build a complete control scheme in Orocos. Each individual component consists of in and outputs and several settings to configure the component. With the use of a script file the controller is configured on runtime by loading, configuring and starting the correct components.

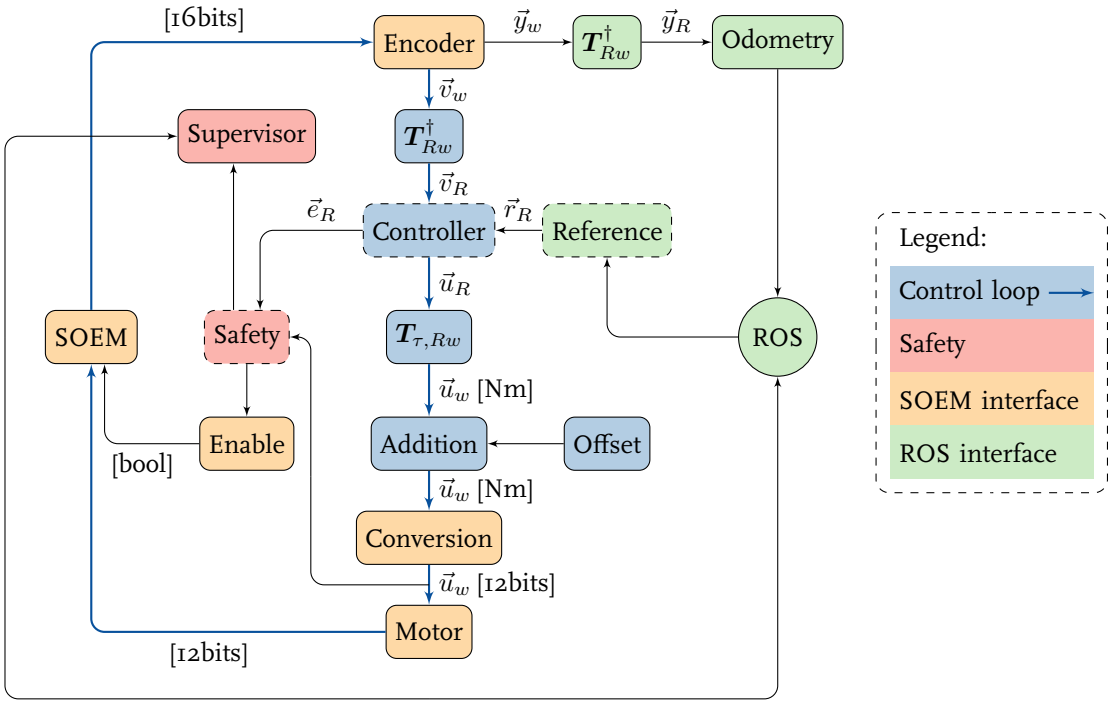


Figure 2.20: Orocos implementation of the low level base controller.

A visualization of all the components used for the base controller is shown in Figure 2.20. In this scheme all the blocks are components from the library except for the ROS circle, which presents the data streams from and to ROS. The blocks presenting the feedback control loop of the system are:

- **SOEM:** Handles all hardware in and outputs through the etherCAT connection.

- **Encoder:** Receives encoder bit values and transfers them to a radial position $\vec{y}_w$ and velocity $\vec{v}_w$.
- $\boldsymbol{T}_{Rw}^{\dagger}$: Matrix transformation from wheel velocities to robot velocities.
- **Controller:** Computes the velocity error $\vec{e}$ and desired control signal $\vec{u}_R$ out of the reference velocity $\vec{r}_R$ and measured velocity $\vec{v}_R$ using several filters.
- $\boldsymbol{T}_{\tau,Rw}$: Matrix transformation of the control signal from robot space to wheel space.
- **Addition:** Adds a constant (received from **Offset** Output value to each wheel to compensate for the fact that a reference signal of zero does not result in an torque of zero generated by the motors. This is caused by a minor offset in the reference voltage on the amplifiers.
- **Conversion:** Transfers the desired wheel torque from Newton-meter to the corresponding 12 bits output value.
- **Motor:** Arranges the control output in the correct messages such that SOEM sends the control signals to the correct hardware outputs.

The green components are used as an interface to and from ROS. The main tasks of these components is translating the message types between ROS and Orocos. In addition the **Reference** component generates a smooth transition from the previous to the newly obtained reference from ROS by limiting the maximum acceleration to $0.7\ \frac{\mathrm{m}}{\mathrm{s}^2}$ in x and y direction and $2\ \frac{\mathrm{rad}}{\mathrm{s}^2}$ in ϕ direction. As a safety feature this component also limits the reference velocities to protect the hardware against its own limitations and failure in the higher level software.

The **Safety** component is added to enable the hardware if the software is ready and safe to start. This component sends an enable/disable signal to the hardware. The safety puts limits on the velocity error, and controller output and disables the hardware if this bound is exceeded. If one of the two limits is exceeded it probably means there is something blocking the motion of the robot and this can damage the robot and/or its environment so it should be prevented.

Finally the **Supervisor** component monitors the hardware and is able to switch the dashed components on and off. This switching is used to enable/disable the control loop. The supervisor disables these components when one of the safety checks is violated or if the hardware receives a disable command from an operator. After a disable action the control loop can only be enabled by the operator to ensure the robot is safe to go back to operational state again.

Chapter 3

Torso

In this chapter the results regarding the torso are presented. An introduction to the torso mechanism is given followed by a literature study on control methods. A model is made of the torso mechanism and experiments are performed to determine the unknown parameters in the model. The identified model is used to estimate the required forces when the arms are attached which result in an advise on the loads of the springs that are going to be used. A first version of a controller is implemented and tested that is able to cope with the non-linearities in the system. Finally the complete control structure including calibration and safety is presented.

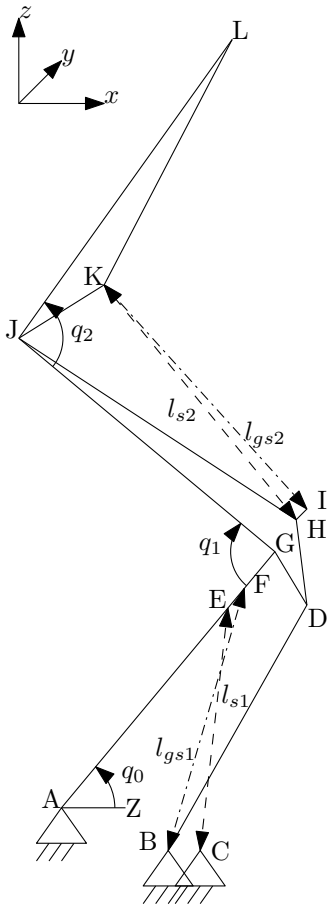
3.1 Design

The main purpose of the torso mechanism is forming the (movable) connection between the base and the arm(s) of the service robot. In SERGIO it is chosen to use a human like mechanism with a ankle, knee and hip joint. The advantage of such a mechanism shows itself in the DOF that result in a large height range and the possibility to bend forward to reach for objects on the floor or on a table. In Figure 3.1a a picture of the torso mechanism is shown with the head also in place. It shows the human like structure, some sensors and electronics like a PC and I/O components. The head is the top part presented by a kinect and microphone.

A schematic drawing of the torso is shown in Figure 3.1b. Note that the schematic drawing is not an exact scaled representation, but just a visualization used to explain the kinematics. The ankle knee and hip joint are denoted by q_0, q_1 and q_2 located at points A, G and J respectively. The ankle joint is actuated by a DC motor driven lead screw (dashed line between C and E) and it rotates the lower leg (line A to G). The knee joint is mechanically coupled to the ankle joint and therefore not actuated separately. The connection is a four bar guiding mechanism realized by the connections A, G, D and B. It rotates the upper leg which is the rigid structure presented by G, J, H and D. The hip joint is again actuated by a DC motor driven lead screw (dashed line between H and K) which rotates the trunk, the rigid structure between J, K and L. Point L is the connection point for both of the arms of SERGIO. To reduce the actuator forces needed, four gas springs will be used to help lift the torso, arms and head. Two are used in the bottom and two in the top represented by the

dashed-dotted lines from B to F and I to K. Note that at this point only one unloaded gas spring is used in the bottom and one in the top because the mass of the arms is not yet in place. Therefore a model of the system with estimated parameters is used to determine the required load of the gas springs. The main challenges in control are found in the non-linear relations between the different components moving the mechanism, like the relation between lead screw length and joint angle. In the next section a study is performed on used methods to control a robot structure like presented here that copes with the non-linearities.

(a) Picture.



(b) Schematic drawing.

Figure 3.1: Torso of SERGIO.

3.2 Related work

In this section several control methods are discussed that are most commonly used on robotic manipulators consisting of multiple joints[20]. The methods discussed in this section are aimed on controlling the positions of joints and end effectors, meaning for example force control methods

are not discussed here. Figure 3.2 will be used as an overview of the different methods that will be addressed in this section. It aims to cover the most basic techniques and methods used, but does not include all available possibilities. The main separation is made between independent joint control (SISO) and multivarialbe control (MIMO) as will be shown in the following two subsections. At the end it is briefly addressed how to apply the presented literature to approach the control design.

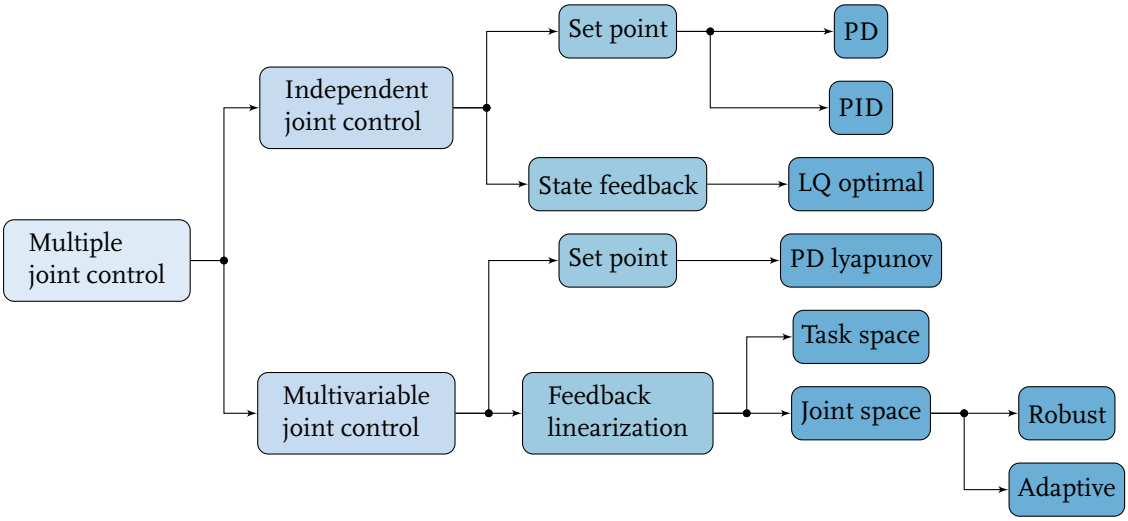


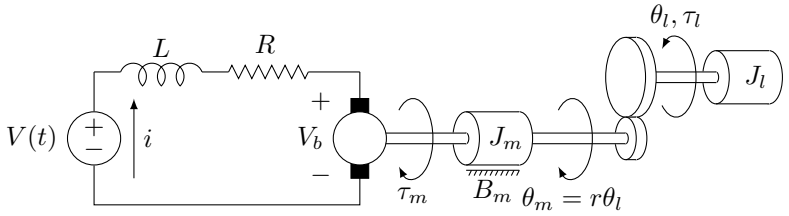
Figure 3.2: Choice on possible commonly used control methods. The dark blue blocks denote final control methods.

3.2.1 Independent joint control

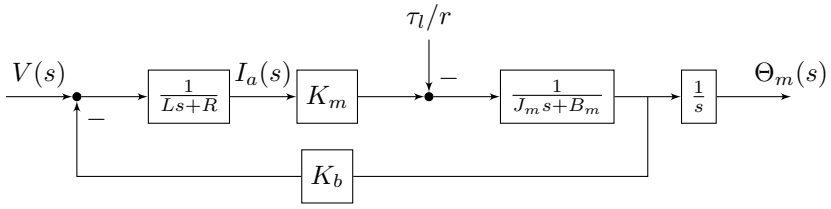
Independent joint control is the most commonly used approach because it is easy to apply and has proven to work well. The main reasons why independent joint control works well is a large gear reduction $r : 1$ in combination with non fast link movements, which make the motor dynamics dominant over the link dynamics. With these assumptions a simplified model of the system can be used to design control as will become clear in this section.

Single joint model

Consider a voltage controlled DC motor driving a joint through a gearbox as shown in Figure 3.3a. The corresponding block scheme is shown in Figure 3.3b gain from Spong et al. [20]. From this block scheme the transfer from supply voltage and load torque to motor angle can be extracted (in



(a) Model.



(b) Block diagram.

Figure 3.3: Illustration of a DC motor driven joint. The symbols used are, supply voltage V, armature inductance L, armature resistance R, armature current I, back emf voltage V_b, back emf constant K_b, motor constant K_m, motor torque τ_m, motor and gearbox inertia combined J_m, friction of motor and gearbox B_m, motor angle θ_m, load angle θ_l, load torque τ_l and gear ratio r.

the Laplace domain),

$$\frac{\Theta_m(s)}{V(s)} = \frac{K_m}{s((Ls+R)(J_m s+B_m)+K_b K_m)}, \tag{3.1}$$

$$\frac{\Theta_m(s)}{\tau_l(s)} = \frac{-(Ls+R)/r}{s((Ls+R)(J_m s+B_m)+K_b K_m)}. \tag{3.2}$$

$$\tag{3.3}$$

It is assumed that the electrical time constant L/R is much smaller than the mechanical time constant J_m/B_m, which is the case for many electromechanical systems. By setting L/R to zero equations above can be rewritten resulting in a simple mass damper system with the load torque seen as a disturbance. The resulting system is shown in Figure 3.4, where $B = B_m + K_b K_m/R$ represents the damping, $u = (K_m/R)V$ is the control input and $d = \tau_l/r$ the disturbance. It shows the joint torque is divided by the gear ratio which supports the assumption to threat the joint torque as a disturbance.

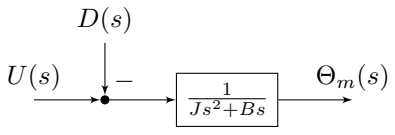


Figure 3.4: Block diagram of the simplified system.

Single joint feed back control

On the simplified model SISO control techniques can be applied. The most used control techniques on such a system are PD (Proportional and Derivative) and PID (Proportional, Integral and Derivative) control on the position or velocity error, also denoted as set point tracking. This leads to the following control law,

$$U(s) = (K_p + \frac{K_I}{s})(\Theta_r(s) - \Theta_m(s)) - K_D s\Theta_m(s), \tag{3.4}$$

where $\Theta_r(s)$ is the reference signal. The controller gains can be chosen according to the following rules,

$$\begin{aligned} K_P &= \omega^2 J, \\ K_D &= 2\zeta\omega J - B, \\ K_I &< \frac{(B + K_D)K_P}{J}, \end{aligned} \tag{3.5}$$

where K_P, K_D and K_I are the proportional, derivative and integral gain respectively, ω is the natural frequency and ζ the damping ratio. The natural frequency and damping ratio can be chosen according to the desired behaviour of the system. It is customary to choose a damping ratio of 1 which makes the system critically damped, resulting in the fastest non-oscillatory response. When integral action is added the inequality in (3.5) should hold to ensure stability of the system according to the Routh-Hurwitz criterion. Besides the above equations there are other limiting factors in control design like saturation and flexibility that should be taken into account. Tuning the parameters is the difficult part, because a model is not always available or it has uncertainties. A commonly used method to tune parameters is to shape a step response on the system.

The previous discussed control technique only uses the position or velocity error based on the difference in sensed values and reference inputs. A more advanced control technique is state feedback. This control technique uses a state space representation of the system to compute the control laws. The internal states of a system as represented in Figure 3.4 are normally chosen to be the position and velocities of both motor and link. The system representation is of the form,

$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= C^T x + Du, \end{aligned} \tag{3.6}$$

where x contains the internal states, u contains the inputs, y contains the outputs and A, B, C and D are the system matrices that determine the behaviour of the system. In essence matrix B maps the effect of the inputs to the states, C maps the states to the outputs, A determines the effect states have on each other and D maps the direct transition from inputs to outputs which is zero in the currently discussed model. As an example the state space representation of the shown system in Figure 3.4 can be found in Appendix B.2. The advantage of using a state space representation is that control can be applied using the states, meaning both motor and link, positions and velocities can be used. By using a smart chosen control law the poles of the system can be placed arbitrarily to gain optimal performance. However there are limiting factors like the available motor torque. A method to place the poles is linear quadratic (LQ) optimal control, which ensures fast convergence while taking the limiting available torque input into account. In order to apply such a control law there are two requirements on the system, namely controllability and observability. Controllability means that a controller can be designed that can reach any chosen state from an arbitrary initial state

within finite time. Observability means that every initial state can be exactly determined out of the measured output and the input in a finite time interval. In simpler words if both conditions hold and a suiting controller is designed the observability ensures the state values are estimated correctly out of input and output signals. The controller can use these estimates to move/control each state to the desired value by reference. The main advantages in this control method is that not only the sensed position (motor or joint), but both can be used in control. Especially when there is flexibility in the system between the motor and joint orientation it is a powerful method. Limitations in the method can be found in system limitations in control output and amplification of noise while computing the state estimates.

The main difference between single joint control methods presented here is the complexity and purpose of use. The advantage of P(I)D control is easy implementation and it has proven to work well. However it can be limited in performance whit joint flexibilities in the system. State feedback control is a more complex method and can cope with joint flexibilities. It also can be used for optimal control design such that performance is maximized within the actuator limits. Both control techniques are limited by motor saturation and cannot cope well with modelling errors.

Feed forward

A great way to improve performance is feed forward. With a known model of the system the inverse of the model can be used as a feed forward. This means the required torque needed to move the system is calculated based on the reference trajectory and added to the controller input signal. In a perfect world without disturbance and a perfect model of the system the feedback could be removed as the feed forward gives the exact input to follow the reference trajectory. Of course feedback is necessary to account for unmodelled dynamics, errors in the parameters and unknown disturbances entering the system. The feed forward in this case gives a good estimate of the required input and the feedback controller only has to filter out disturbances and model errors. In order to use the inverse dynamics of the plant as feed forward the plant should be minimum phase, meaning the inverse of the plant is a stable.

3.2.2 Multivariable joint control

Multivariable joint control is a method that takes into account the coupling dynamics between the joints. These methods are mainly used when the methods discussed before do not meet the desired behaviour. The multivariable joint control also requires a more complex model of the complete system as will be shown here.

Multiple joint model

There are two main methods used to set up the equations of motion, namely with use of the Euler-Lagrange equations or the Newton-Euler equations. Both will result in the same equations of motion. The first method will be briefly explained here as it is easier to apply on a multiple joint model, for a more extensive explanation see Spong et al. [20]. The Lagrangian is the difference between kinetic

and potential energy,

$$\mathcal{L} = \mathcal{K} - \mathcal{P}. \tag{3.7}$$

The kinetic energy exists of the translational and rotational kinetic energy. For a single body the kinetic energy is

$$\mathcal{K} = \frac{1}{2} m \vec{v}^T \vec{v} + \frac{1}{2} \vec{\omega}^T \mathcal{I} \vec{\omega} \tag{3.8}$$

where $\vec{v}$ is the translational velocity of the center of mass, $\vec{\omega}$ is the rotational velocity of the body, m is the mass and $\mathcal{I}$ is the inertia tensor. The potential energy of a link is described by

$$\mathcal{P} = m \vec{g}^T \vec{r}_{CM}, \tag{3.9}$$

where g defines the gravitational constant and $\vec{r}_{CM}$ describes the location of the center of mass of the link. The equations of motion can be computed by taking partial and time derivatives of the Lagrangian,

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\vec{q}}} - \frac{\partial \mathcal{L}}{\partial \vec{q}} = \vec{\tau}^T \tag{3.10}$$

where $\vec{\tau}$ contains the motor input torques, $\vec{q}$ and $\dot{\vec{q}}$ contain the generalized coordinates and velocities respectively and the partial derivative is taken of the Lagrangian to the elements of $\dot{\vec{q}}$ and $\vec{q}$.

In case of robotic manipulators there are two conditions that are commonly satisfied. The first is that the kinetic energy is a quadratic function of the vector $\vec{q}$ of the form,

$$\boldsymbol{K} = \frac{1}{2} \dot{\vec{q}} \boldsymbol{D}(\vec{q}) \dot{\vec{q}} \tag{3.11}$$

where $\boldsymbol{D}$ is a symmetric and positive definite inertia matrix. The second condition states that the potential energy is a function independent of the velocities $\dot{\vec{q}}$, meaning $\boldsymbol{P}(\vec{q})$. The nice thing about these two conditions is that the equations of motion can be written in the form,

$$\boldsymbol{D}(\vec{q})\ddot{\vec{q}} + \boldsymbol{C}(\vec{q}, \dot{\vec{q}})\dot{\vec{q}} + \boldsymbol{g}(\vec{q}) = \vec{\tau}, \tag{3.12}$$

where $\boldsymbol{C}(\vec{q}, \dot{\vec{q}})\dot{\vec{q}}$ contains the centrifugal and Coriolis terms and $\vec{g}(\vec{q})$ contains the gravitational terms. This form has some nice properties useful in multivariable feedback control as will be shown later on in this section.

The motor dynamics as shown in Figure 3.4 can also be added to the model, which for a single motor is,

$$J_m \ddot{\theta}_m + B\dot{\theta}_m = \frac{K_m}{RV} - \frac{1}{r}\tau. \tag{3.13}$$

Using that the motor angle of joint k is $\theta_{m_k} = r_k q_k$ it can be added to the equations of motion resulting in

$$\boldsymbol{M}(\vec{q})\ddot{\vec{q}} + \boldsymbol{C}(\vec{q}, \dot{\vec{q}})\dot{\vec{q}} + \boldsymbol{B}\dot{\vec{q}} + \boldsymbol{g}(\vec{q}) = \vec{u}, \tag{3.14}$$

where $\boldsymbol{M}(\vec{q}) = \boldsymbol{D}(\vec{q}) + \boldsymbol{J}$, with $\boldsymbol{J}$ a diagonal matrix containing the motor inertias on the diagonal scaled by the gear ratio $r_k^2 J_{m_k}$, $\boldsymbol{B}$ is a diagonal matrix containing the friction components $r_k^2 B_{m_k}$ and the input vector $\vec{u}$ contains the motor torques $r_k \frac{K_{m_k}}{R_k} V_k$.

Multiple joint feedback control

A PD controller as used for the single joint control can also be used for multiple joint control. It can be proven that asymptotic tracking is possible when the friction is neglected $\boldsymbol{B} = 0$ and in the absence of gravity $\vec{g}(\vec{q}) = 0$. The PD control law from (3.4) can be written in vector form,

$$\vec{u} = -\boldsymbol{K}_P\vec{e} - \boldsymbol{K}_D\dot{\vec{q}}, \tag{3.15}$$

with $\vec{e}$ is a vector containing the position errors and $\boldsymbol{K}_P$ and $\boldsymbol{K}_D$ diagonal matrices containing the proportional and damping gains per joint. With use of a Lyapunov function and LaSalle's theorem it can be proven that this control function leads to asymptotic stability. However gravitation is often present in the system which results in a steady state error. this can be reduced by adding a the required gravitational torque $\vec{g}(\vec{q})$ to the input. In theory, without the gravitational compensation, the PD controller presented here does not differ anything from the SISO PD controller except for the prove of MIMO stability.

Adding the gravitational torque to the input is a first step to feedback linearisation. The main idea behind feedback linearisation is using an inner and outer control loop. The inner control loop uses the model to linearise the system with the following control law,

$$\vec{u} = \boldsymbol{M}(\vec{q})\vec{a}_q + \boldsymbol{C}(\vec{q}, \dot{\vec{q}})\dot{\vec{q}} + \boldsymbol{g}(q), \tag{3.16}$$

where $\vec{a}_q$ is the new control input for the outer loop. Because $\boldsymbol{M}(\vec{q})$ is invertible it is easy to see that the remaining system is $\ddot{\vec{q}} = \vec{a}_q$, meaning the control input of the outer loop is the acceleration of the system. By choosing the control law as follows,

$$\vec{a}_q = \ddot{\vec{q}}_r - K_P\vec{e} - K_D\dot{\vec{e}}, \tag{3.17}$$

with $\ddot{\vec{q}}_r$ the desired reference acceleration it is actual a PD controller on the linearised system where the reference acceleration $\vec{a}_q$ serves as a feed forward term. One of the difficulties in this control scheme is the computation of the linearising feedback on runtime. Another problem is again modelling errors, but in this case robust and adaptive control methods are developed to cope with these errors.

In the previous method the inner loop decoupled the system to joint space and a control law is used that follows joint reference trajectories. Another addition can be made where the joint space can be transferred to task space with use of the inverse and derivative of the Jacobian. In this method only end effector references have to be computed, which keep in mind the boundedness of the joints. Requirements in this method are nonsingularity of the Jacobian and the inverse of the Jacobian needs to be computed.

The exact theory behind the robust and adaptive control methods is outside the scope of this thesis, but the ideas and (dis)advantages will be briefly addressed here. Both theories aim to ensure performance despite the fact of model errors and uncertainties. The robust control method includes a certain range of uncertainties to the model matrices defining the system. A control law is then designed such that the actual system is stable if it lays within this uncertainty range. Adaptive control uses a model of the system and estimates of the unknown parameters. A control law is designed such that the estimates are adjusted online and the difference between estimates and real plant values converges to a minimal value. This means that both control laws can deal with model errors and uncertainties. The strength of a robust controller is that it performs well while there are model

uncertainties and errors in the system. The disadvantage shows when a repetitive task is performed, tracking errors will repeat itself each iteration. This is where the strength of the adaptive controller shows, meaning when a repetitive tasks is performed the parameters converge to the real values and the tracking error reduces. The downside of Adaptive control are unknown disturbances and mismatches in the dynamic model. If there are dynamics present in the system that are not in the model the adaptive controller cannot account for these mismatches.

The multivariable joint control methods presented here have advantages over the single joint control methods as they contain a more complete model of the system. This makes it possible to guarantee stability on MIMO level. In order to guarantee performance and robustness the adaptive and robust control methods can be used which take into account uncertainties of the plant dynamics.

3.2.3 Contributions

Looking at Figure 3.2 a control structure should be chosen that suits the design and application of the torso. The choice between independent and multivariable joint control is mainly determined by the required operation speeds and the gear ratios used. In case of a service robot the application speeds are not very fast, especially compared to industrial robots. The gear ratio in the presented models is a constant, whereas the gear ratio between joint angle and motor angle in the torso mechanism is a non-linear function. In addition there is the non-linear connection between the ankle and knee joint by the four bar guiding mechanism. It should be invested if similar methods can still be applied. If so the most convenient way to go is independent joint control.

In order to give an advise on the spring stiffness the system should be identified based on a parametric model. With use of experiments the unknown parameters should be estimated. A model is necessary such that the weight of the arms and the effect of the springs can be simulated.

3.3 Model

The torso structure shown in Figure 3.1b can be simplified to the model representation shown in Figure 3.5. The model consists of four point masses representing the lower leg, upper leg, trunk and arms. The lower leg has a length l_1, a mass m_1 and its center of mass is located at a distance l_{cm1} of the ankle joint. Note that the mass and center of mass locations are unknown parameters and it is assumed the center of mass is somewhere on the line between the ankle and knee joint. Similar parameters represent the upper leg with subscript 2 and the trunk with subscript 3. The arms are presented with a point mass m_4 located in the shoulder. The hip joint is actuated with a force F_{ls2} at a distance d_{ls2} of the hip joint. The force is applied at an angle $\theta_{ls2}(q_2)$ dependent on the hip joint angle. In a similar matter the ankle joint is actuated with force F_{ls1}. The knee joint is also actuated by the force F_{ls1} through the coupling to the ankle joint by the four bar guiding mechanism. Note that the spring forces are not present in this drawing, but these can be added in a similar matter as the forces applied by the lead screws.

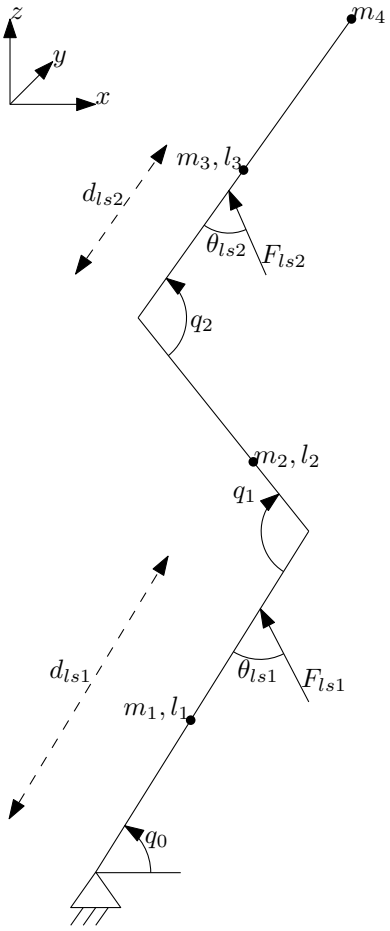


Figure 3.5: Model of the torso mechanism.

3.3.1 Kinematic relations

The two independent coordinates in this system are chosen to be q_0 and q_2, presented by the vector $\vec{q} = \begin{bmatrix} q_0 & q_2 \end{bmatrix}^T$. To set up the equations of motion for this system the relation between all variables and the independent coordinates should be determined. One of these variables is the joint q_1 dependent on q_0. In order to determine the relation between these joints Figure 3.1b is used. The distance between two points for example A and B is defined as $\overline{AB}$ and the angle between three points is defined as $\angle ABC$, which means the angle that is made rotating counter clockwise from A to C around B. The knee angle can be defined as,

$$q_1 = \angle JGD - \angle AGD(q_0), \tag{3.18}$$

where $\angle AGD(q_0)$ can be determined with use of the quadrilateral defined by points A, G, D and B. In this quadrilateral there are four known sides and one known angle $\angle BAG = q_0 + \angle BAZ$. By

dividing the quadrilateral and using the cosine rule,

$$\overline{BG}(q_0) = \sqrt{\overline{AB}^2 + \overline{AG}^2 - 2\overline{AB}\ \overline{AG}\cos(\angle BAG(q_0))}, \tag{3.19}$$

two triangles remain and the angle

$$\angle AGD(q_0) = \angle AGB(q_0) + \angle BGD(q_0), \tag{3.20}$$

can be determined where,

$$\angle AGB(q_0) = \arccos\left(\frac{\overline{AG}^2 + \overline{BG}^2(q_0) - \overline{AB}^2}{2\overline{AG}\ \overline{BG}(q_0)}\right), \tag{3.21}$$

$$\angle BGD(q_0) = \arccos\left(\frac{\overline{DG}^2 + \overline{BG}^2(q_0) - \overline{BD}^2}{2\overline{DG}\ \overline{BG}(q_0)}\right). \tag{3.22}$$

In Appendix B.3 the other variables are described as a function of the generalized coordinates, which includes the length of both lead screws l_{ls}, the angles of the lead screws with respect to the link θ_{ls} and the length of both gas springs l_{gs}.

3.3.2 Equations of motion

The kinematic relations are used to set up the equations of motion with use of the Lagrangian as presented in Section 3.2. Since the links of the robot consists of a lot of separate parts and materials it is hard to define an inertia matrix for the links. It is assumed the links can be estimated by point masses because the rotational speeds the links are subjected to are very low. The kinetic energy for link i is then defined by,

$$\mathcal{K}_i = \frac{1}{2} m_i \vec{v}_i^T \vec{v}_i, \tag{3.23}$$

where $\vec{v}_i$ contains the velocities of the center of mass in x, y and z direction. These velocities are computed by describing the center of mass location in terms of the independent coordinates and taking the time derivative. As example for the lower leg (link 1) this becomes,

$$\vec{r}_{CM1} = \begin{bmatrix} l_{CM1}\cos(q_0) & 0 & l_{CM1}\sin(q_0) \end{bmatrix}^T, \tag{3.24}$$

$$\vec{v}_1 = \begin{bmatrix} -l_{CM1}\sin(q_0)\dot{q}_0 & 0 & l_{CM1}\cos(q_0)\dot{q}_0 \end{bmatrix}^T \tag{3.25}$$

and the calculations for the other links can be found in Appendix B.4. Because the system can only move in 2D space the y direction will be zero for all links. With the determined positions of each center of mass the potential energy by gravity is easily determined as shown in (3.9). Because springs will be used in the torso mechanism these should be included in the potential energy,

$$\mathcal{P}_{gs} = \frac{1}{2} k (l_0 - l(q))^2, \tag{3.26}$$

where k is the spring constant and l_0 the equilibrium length of the spring and $l(q)$ the actual length dependent on the generalized coordinates. For both springs in the legs this results in,

$$\mathcal{P}_{gs1} = N_{gs1} \frac{1}{2} (l_{0,gs1} - l_{gs1}(q))^2, \tag{3.27}$$

where N_{gs1} defines the amount of gas springs used, which is zero at the moment and will be two when all gas springs are placed. The Lagrangian can now be computed and by taking derivatives the equation of motion of the joint dynamics is obtained as shown in (3.12), where $\vec{\tau}$ presents the joint torque.

The drive train dynamics as shown in (3.13) should be adjusted such that it uses the gear ratio as a function of the joint angle. Also the back emf can be neglected because current controllers are used on the torso. This results in the changed representation of the motor dynamics,

$$J_{mg}\ddot{\theta}_m + B_{mg}\dot{\theta}_m = \tau_m - \frac{1}{r_g r_{ls} d_{ls}\sin(\theta_{ls})}\tau, \tag{3.28}$$

where J_{mg} and B_{mg} contain the inertia and friction of motor gearbox and lead screw, r_g is the gear ratio, r_{ls} is the ratio from lead screw rotation to translation and $d_{ls}\sin(\theta_{ls})$ transfers the lead screw force to the effective joint torque. The motor angle and its time derivatives can be described as a function of the joint angle resulting in,

$$\theta_m = r_g r_{ls} l_{ls}(q), \tag{3.29}$$

$$\dot{\theta}_m = r_g r_{ls}\left(\frac{\partial l_{ls}(q)}{\partial q}\right)\dot{q}, \tag{3.30}$$

$$\ddot{\theta}_m = r_g r_{ls}\left(\left(\frac{\partial l_{ls}(q)}{\partial q}\right)\ddot{q} + \left(\frac{\partial l_{ls}^2(q)}{\partial^2 q}\right)\dot{q}^2\right). \tag{3.31}$$

Substituting these equations in (3.28) and combining with (3.12) results in the equations of motion of the complete system. Note that because of the non-linear relations the motor and gearbox inertias result in an inertial and centrifugal term in the equations of motion. Because of the non-linearities the relations and derivatives are difficult to compute and therefore the symbolic toolbox of MATLAB is used from this point on. The equations of motion as presented in this section will be used to identify the model and determine the effect of the non-linearities such that a good choice on controller design can be made.

3.4 Identification

In this chapter the identification of the torso is presented with use of the model set up in the previous section. First simulations are run with the model and estimates of the parameters to determine the dominant factors in the required motor torque. Based on the simulations the identification experiments are set up and performed. First the trunk parameters are estimated followed by the legs. The identified model will be used in the next section to determine the advise on gas spring loads.

3.4.1 Simulation

The required motor torques can be calculated for any desired reference trajectory with use of the equations of motion. The unknown parameters in the model are the masses of the link, their location and the friction term. At this point the friction component will be neglected as it is hard to define a

proper estimate, but it will return in the identification experiments. The main goal of the simulations presented here is to determine the contribution of the different components in the torque. The different components are referred to as the inertia contribution $\boldsymbol{M}(\vec{q})\ddot{\vec{q}}$, the centrifugal and Coriolis contribution $\boldsymbol{C}(\vec{q},\dot{\vec{q}})\vec{q}$ and the gravitational contribution $\boldsymbol{g}(\vec{q})$.

Simulations are performed with the following three input trajectories,

- Move the trunk from one end to the other with the legs positioned at the center of its range.
- Move the legs from one end to the other with the trunk positioned at the center of its range.
- Move at the same time legs and trunk from minimal angle to maximum angle.

All trajectories are performed within four seconds with one second acceleration, two seconds constant velocity and two seconds deceleration. In order to simulate some estimates of parameters are needed. The known measures used for simulation can be found in Appendix B.1. The unknown measures are the masses and their center of mass locations. For the center of mass locations it is estimated they lay on the center of the link, meaning for the lower leg it is located at a distance $l_{cm1} = \frac{1}{2}l_1$ from the ankle joint. The estimated masses are taken from the design estimates[9] that are $m_1 = 3.5$ kg, $m_2 = 5$ kg, $m_3 = 10$ kg, $m_4 = 20$ kg. These simulations are used to identify the current system without the arms, therefore the fourth mass is set to zero.

The results of the third simulation are shown in Figure 3.6 and the results of the other two simulations can be found in Appendix B.5. The reference acceleration is shown in the top plot and is higher for the second joint because it has a larger angle range. The required motor torques show that the dominant factor is the gravitation compensation. Inertia also shows a significant contribution and the centrifugal and Coriolis terms can be neglected. By plotting the inertia term contributions by link and drive train separately, as shown in Appendix B.5, it can be concluded that the main contribution is gain from drive train acceleration including motor, gearbox and lead screw inertia. On the basis of these simulations it can be concluded that the main task in the identification is determining the parameters in the gravitational part of the model followed by the inertia term.

3.4.2 Model parameters

As concluded by simulations the main task is to identify the gravitational term $\boldsymbol{g}$ of the system. In this component there are six unknown parameters, namely the three masses of the links and their center of mass locations. The potential energy of the system by gravitation can be written in the form,

$$\begin{aligned} V_g = {} & g\sin(q_0)(P_1 + l_1 m_4) \\ & + g\sin(q_1(q_0) - q_0)(P_2 + l_2 m_4) \\ & + g\sin(q_2 - q_1(q_0) + q_0)(P_3 + l_3 m_4), \end{aligned} \tag{3.32}$$

with the unknown parameters,

$$P_1 = l_{cm1}m_1 + l_1(m_2 + m_3), \tag{3.33}$$
$$P_2 = l_{cm2}m_2 + l_2 m_3, \tag{3.34}$$
$$P_3 = l_{cm3}m_3, \tag{3.35}$$

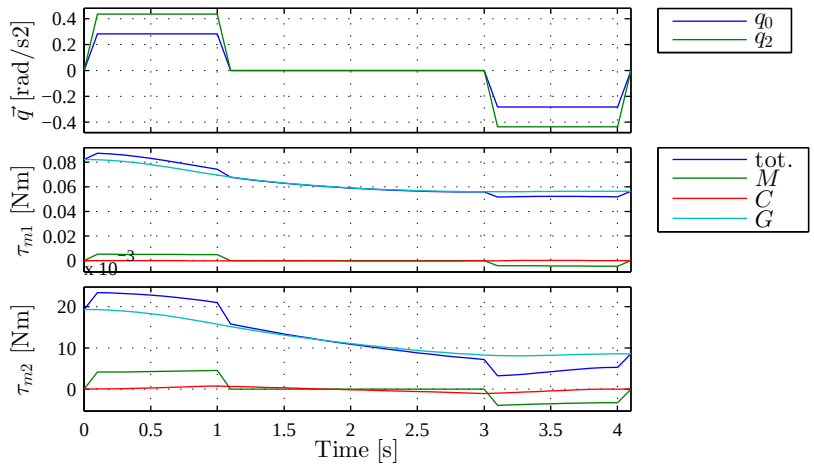


Figure 3.6: Simulation results moving the torso from the lower limits to the upper limits. The upper plot shows the reference accelerations and the lower plots show the torque contributions of the legs τ_1 and the trunk τ_2.

which shows the parameter estimation problem can be reduced to three estimates in stead of six. The gravitation component in the equations of motion is obtained by taking the derivatives to the generalized coordinates,

$$\begin{aligned}\boldsymbol{g}_1(\vec{q}) &= g\cos(q_0)(P_1 + l_1 m_4) \\ &\quad + g\cos(q_1(q_0) - q_0)\left(\frac{\partial q_1(q_0)}{\partial q_0} - 1\right)(P_2 + l_2 m_4) \\ &\quad + g\cos(q_2 - q_1(q_0) + q_0)\left(1 - \frac{\partial q_1(q_0)}{\partial q_0}\right)(P_3 + l_3 m_4), \qquad (3.36)\\ \boldsymbol{g}_2(\vec{q}) &= g\cos(q_2 - q_1(q_0) + q_0)(P_3 + l_3 m_4). \qquad (3.37)\end{aligned}$$

In these equations it shows that parameter P_3 can be determined by measuring the control output of the trunk at different positions of joint q_2. The parameters P_1 and P_2 can be determined by measuring the control input of the legs at different positions of joint q_0. By measuring at static positions the inertia, Coriolis and centrifugal terms are zero. The problem in this method is the large amount of friction in the system and therefore a quasi static measurement is taken where the system moves over its complete range in both directions whit a low constant velocity ($0.004\ \frac{m}{s}$). In the experiments with the trunk the mass of the shoulders is varied such that the effect of a known mass could be taken up in the model resulting in better estimates. For the leg experiments the trunk is placed in different orientations while moving the legs over the full range. The known masses in the shoulders could not be used in the leg experiments because of the motor limits.

Finally the acceleration component is estimated with use of the following equation,

$$\hat{\boldsymbol{M}}_{ls} = \boldsymbol{K}_a \ddot{\vec{l}}_{ls}, \qquad (3.38)$$

where $\boldsymbol{K}_a$ is a diagonal component containing average inertia terms for the system. Because the inertia of the drive train is dominant the parameter is estimated in terms of lead screw length. With

the motor inertia being a diagonal matrix the off-diagonals from the link inertia are neglected at this point and seen as disturbance. The final acceleration term will contain the contribution of the drive train and an more or less average of the diagonal terms in link inertia in terms of lead screw length. The acceleration diagonal elements can be determined by applying an acceleration reference trajectory containing constant velocity, acceleration and deceleration terms.

3.4.3 Trunk identification

The measurement results with the legs positioned at the center of their range are shown in Figure 3.7. In all measurements the feedback controller ensures the error is bounded by 0.0002 rad, meaning the controller output is more or less equal to the required torque. The positive velocity measurement are moving from bended position upwards and even to a small backwards bend which shows by the crossing points of the different mass measurements. The measurement with the additional 20 kg mass placed int he shoulders could not be taken over the full range because the motor limit is at 0.101 Nm.

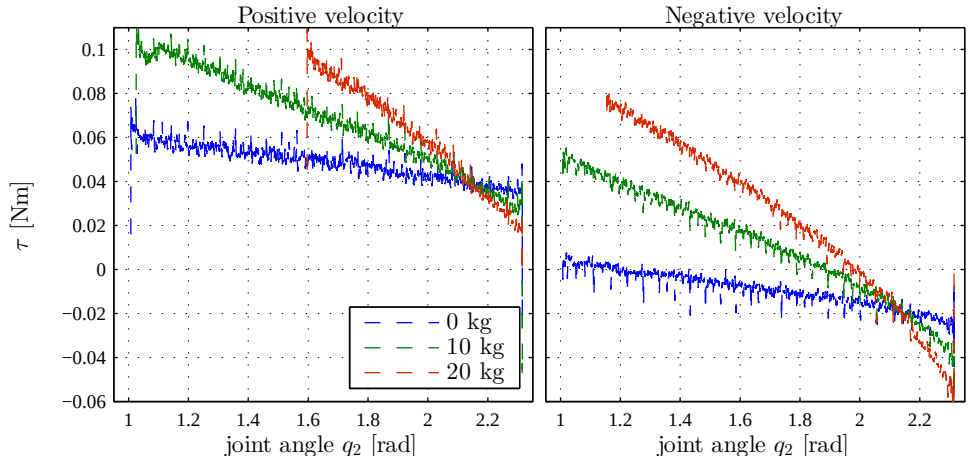


Figure 3.7: Required input torque of the trunk while moving with a constant lead screw velocity of 0.004 $\frac{\text{m}}{\text{s}}$ in positive and negative direction on the left and right figure respectively.

There are several important things that can be extracted from these figures. First of all the equilibrium position of the trunk can be extracted which is the orientation at which the center of mass is located straight above the hip joint. Taking an additional measurement with the legs in another orientation, as shown in Figure B.4 (note that figures denoted with a B can be found in Appendix B.6), resulted in the conclusion that the center of mass is not on the straight line between the hip and shoulder joint but there is a small offset of $2°$. In (3.37) this can be accounted for by adding an offset in the cosine. It also shows that there is a direction dependent friction term present as the average of the positive and negative velocity measurements at this equilibrium result in an offset of $K_{dir,2} \approx 0.0096$ Nm.

Without the presence of viscous friction and by taking into account the direction dependent friction it is assumed that the remaining contributions in the torque signal are caused by the grav-

itational components of both trunk and shoulder mass and coulomb friction. The coulomb friction term is calculated as half the difference between the positive and negative velocity data set, $K_{coul,2} = 0.0291$ Nm. This results in the friction term for the trunk,

$$B_2(\dot{q}) = K_{coul,2}\,\mathrm{sign}(\dot{q}_2) + K_{dir,2}. \tag{3.39}$$

The friction compensation is subtracted from the dataset such that the remaining signal represents the gravitational contribution. With the use of the 'optimization' toolbox in MATLAB the value of the parameter is determined by minimizing the cost function,

$$cost(P_3) = \sum_k \left(\tau_{2,meas}(k) - \boldsymbol{g}_2(q_{2,meas}(k), P_3)\right)^2, \tag{3.40}$$

where k represents a data sample and subscript $meas$ denotes data from measurements. This results in an estimate of $P_3 = 2.47$ kgm. Assuming the mass is located at the center of the link length this results in a mass of 10.5 kg of the trunk which corresponds to the design estimate of 10 kg. In Figure 3.8 the fit of the model with the estimated parameters on the measured data is shown. It shows that it is a quit accurate estimate of the system.

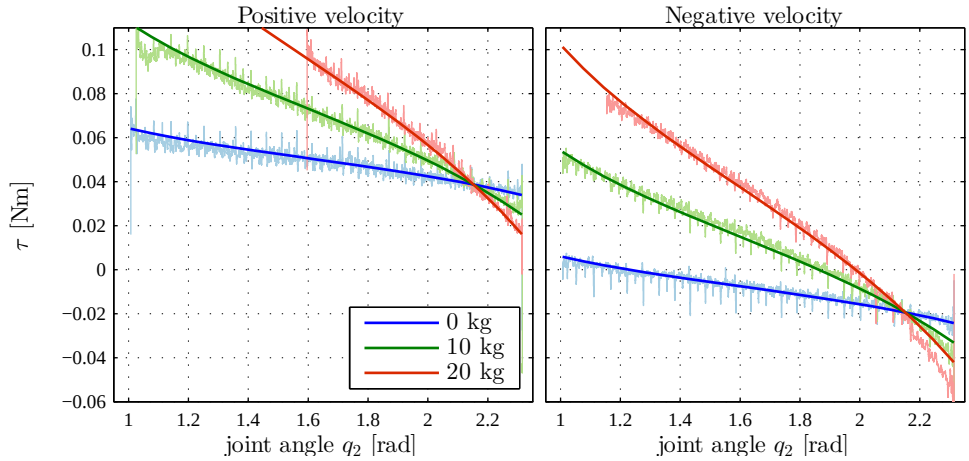


Figure 3.8: Fit of the gravitation and friction compensation by the model on the measured data.

The last parameter that is identified is the inertia component $K_{a,2} = 0.15$. This parameter is tuned by applying a reference trajectory containing accelerations of $0.021\ \frac{\mathrm{m}}{\mathrm{s}^2}$ and constant velocities of $\pm 0.032\ \frac{\mathrm{m}}{\mathrm{s}}$. In Figure 3.9 a scaled version of the acceleration is shown and the position error for a measurement without and with acceleration compensation.

The experimental results presented here resulted in the final identified model of the trunk,

$$\tau_{m,2}(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) = \hat{\boldsymbol{g}}_1(\vec{q}) + \hat{K}_{coul,2}\,\mathrm{sign}(\dot{q}_2) + \hat{K}_{dir,2} + \hat{K}_{a,2}\ddot{l}_{ls,2}, \tag{3.41}$$

where $\hat{\boldsymbol{g}}_1$ means the estimate of the gravitational term working on the hip joint.

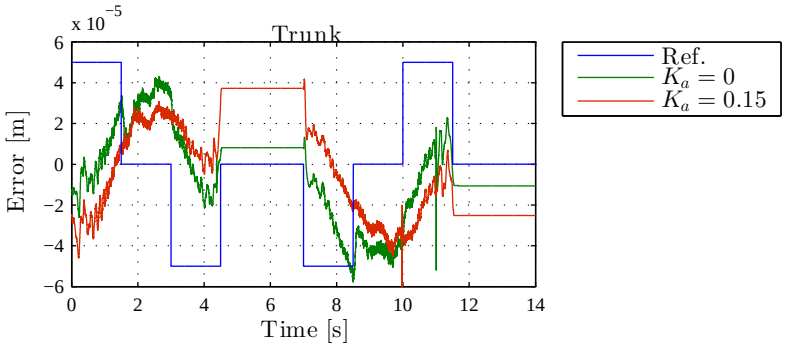


Figure 3.9: Tuning results of the acceleration component for the trunk.

3.4.4 Leg identification

The measurement results of the leg experiments are shown in Figure 3.10. To remove the high frequent noise present the torque signal is filtered with a Butterworth filter as shown in the figure by the non transparent lines. As for the trunk the leg motor torque is also limited to 0.101 Nm and therefore the measurements with the trunk at the two highest angles are not measured over the completer range.

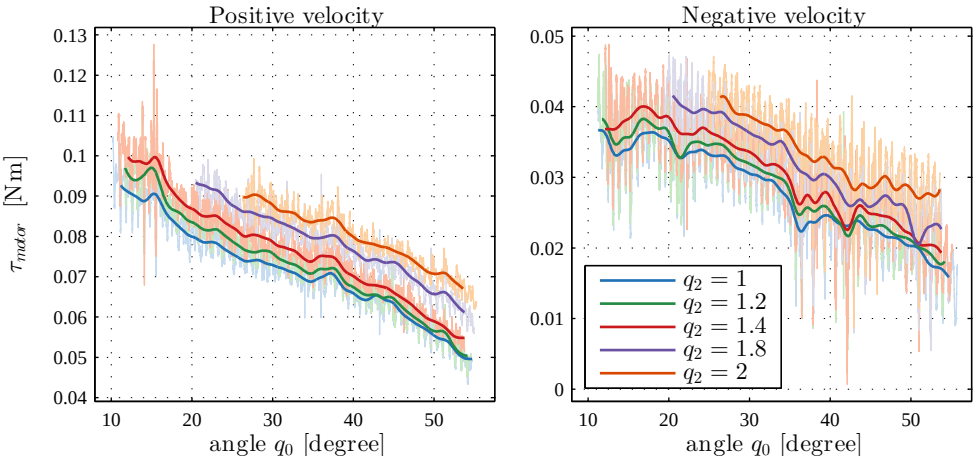
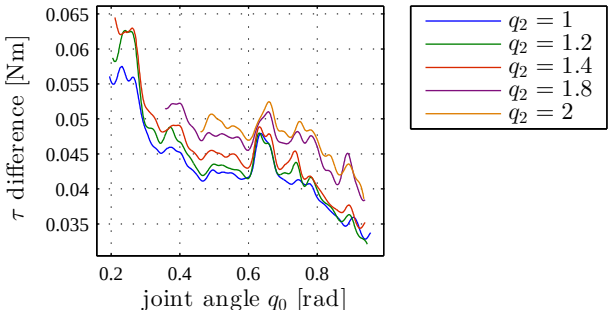


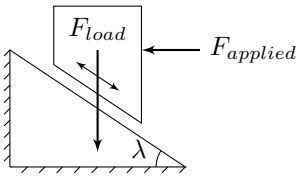
Figure 3.10: Required input torque to move the legs from kneeled to straight position with a constant lead screw velocity of 0.004 $\frac{\text{m}}{\text{s}}$ and the trunk placed at different angles. Both data and filtered data are shown.

Using the same friction compensation as used for the trunk leads to a problem because of the unknown equilibrium of the system. Therefore the direction dependent friction is not used in the legs identification. From the difference between positive and negative datasets it can be concluded that the friction term is proportional to the load as shown in Figure 3.11a. A possible explanation for this can be found in the friction forces in both joints, the four bar guiding mechanism and the drive train (lead screw and gearbox). Friction occurs in all these components and is proportional to the

normal forces between rubbing surfaces.



(a) Torque difference between motion in positive and negative direction with a constant velocity of $0.004\ \frac{\mathrm{m}}{\mathrm{s}}$.



(b) Unwrapped view of a lead screw.

Figure 3.11

An example of such a proportional friction is shown in Figure 3.11b with a load that can move up and down a slope at angle λ and friction occurring between the surfaces. The load force and applied force are both not in the direction of movement and are therefore decoupled to a force in the direction of the slope and a normal force between surfaces. The coulomb friction between two surfaces is dependent on the normal forces between surfaces and therefore the friction is dependent on the load force and the applied force. The resulting forces to raise and lower the weight are,

$$F_{applied,raise} \geq \frac{\mu + \tan(\lambda)}{1 - \mu \tan(\lambda)} F_{load}, \tag{3.42}$$

$$F_{applied,lower} \leq \frac{\mu - \tan(\lambda)}{1 + \mu \tan(\lambda)} F_{load}, \tag{3.43}$$

with μ the friction coefficient. In case of the lead screw the angle λ is known and the friction calculation can be added in this form. However the angles at which forces are applied in the other joints of for example the four bar guiding mechanism are unknown. Therefore a proportional friction component will be added to the model of the form,

$$B_{prop} = K_{prop,2} \boldsymbol{g}_1(\vec{q}) \operatorname{sign}(\dot{q}_0 \boldsymbol{g}_1(\vec{q})), \tag{3.44}$$

with $K_{prop,2}$ the friction gain that should be estimated. At the moment the gas spring forces are zero, but when added they can change the sign of the potential forces resulting in a change of sign in friction. The resulting model that is used to fit to the data is,

$$\vec{\tau}_1 = \boldsymbol{g}_1(\vec{q})(1 + K_{prop,2} \operatorname{sign}(\dot{q}_0 \boldsymbol{g}_1(\vec{q})) + \tau_{coul,1}. \tag{3.45}$$

Using the 'optimization' toolbox of MATLAB the parameters are estimated by minimizing the cost function of the form,

$$cost(\vec{P}) = \sum_k (\tau_{meas,k} - \vec{\tau}_1(q_{0,meas}(k)))^2, \tag{3.46}$$

where $\vec{P}$ contains the unknown parameters. The resulting estimates are,

$$P_1 = 6.5754, \qquad P_2 = 4.3237, \qquad K_{prop,1} = 0.2881, \qquad K_{coul,1} = 0.0075. \tag{3.47}$$

In Figure 3.12 the fit of the model to the filtered data is shown. Using the estimates P_1, P_2 and P_3 to calculate the masses with the center of mass estimated at the center of the link results in $m_1 = 2.7$, $m_2 = -0.08$ and $m_3 = 10.5$. It is of course impossible for a link to have a negative mass. A reason can be the center of mass locations. If for example the center of mass of the trunk is changed from half the link length to $\frac{2}{3}$ times the link length the masses change to 7.9, 5.3 and 8.0 for mass 1, 2 and 3 respectively. Therefore the parameters can still be a reasonable estimate of the real values and the parameters P_1, P_2 and P_3 will still be used.

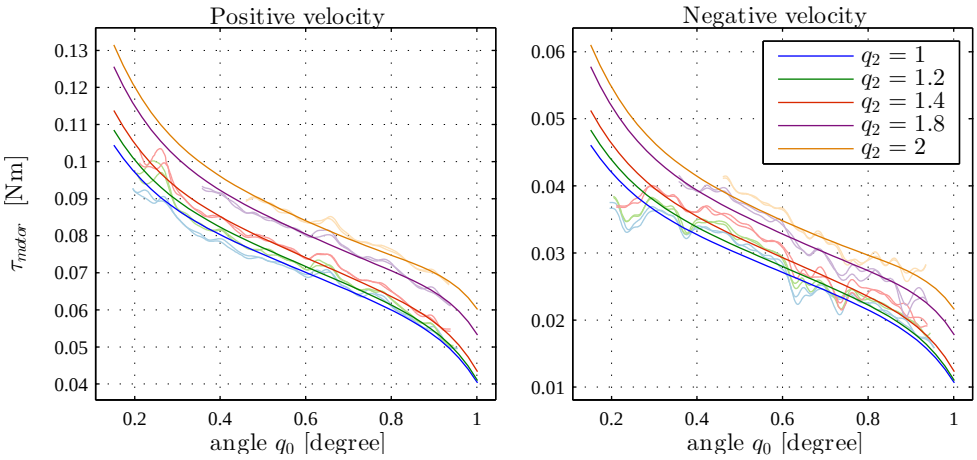


Figure 3.12: fit

As for the trunk the acceleration component of the leg is also tuned with use of an acceleration trajectory as shown in Figure B.6. The resulting acceleration gain parameter is $K_{a,1} = 0.15$.

3.4.5 Identified model

In this section simulations and experiments are used to determine the contribution of the different components in the equations of motion. It shows that the main contributions come from friction and gravitation. The inertia component also includes a small contribution and centrifugal and Coriolis terms can be neglected. The resulting identified model of the torso mechanism without arms and gas springs is,

$$\vec{\tau}_m = \hat{\boldsymbol{M}}(\vec{q})\ddot{\vec{q}} + \hat{\boldsymbol{g}}(\vec{q}) + \hat{\boldsymbol{B}}(\vec{q}, \dot{\vec{q}}) \tag{3.48}$$

3.5 Spring stiffness

In this section the identified model of the previous section will be used to determine the load of the gas springs in both trunk and legs. First the specifications and requirements are presented that are used to determine the load of the springs. Several options of spring load and drive train

combinations that meat certain requirements for the trunk and legs are analysed. At the end the final advice is presented and the resulting reachable space of the shoulders is compared to AMIGO.

3.5.1 Specifications and requirements

In de Roest [9] it is stated the torso mechanism should be able to lift the mass and moment generated by the arms when manipulating objects. The mass of the arms is 10 kg each of which 4.5 kg is the moving part of the arm and the remaining mass is located in the shoulder joints. The robot should be able to handle 1.5 kg with the arms in stretched position. This means that the torso should be able to handle a mass in the shoulders of 20 kg with an additional moment of 0 Nm up to 52 Nm. Note that 52 Nm is two arms lifting a total mass of 3 kg in stretched position which is an extreme case. It is assumed that the mass of the arms is always present in the shoulders, meaning a counterweight should be placed when the torso is used without the arms attached.

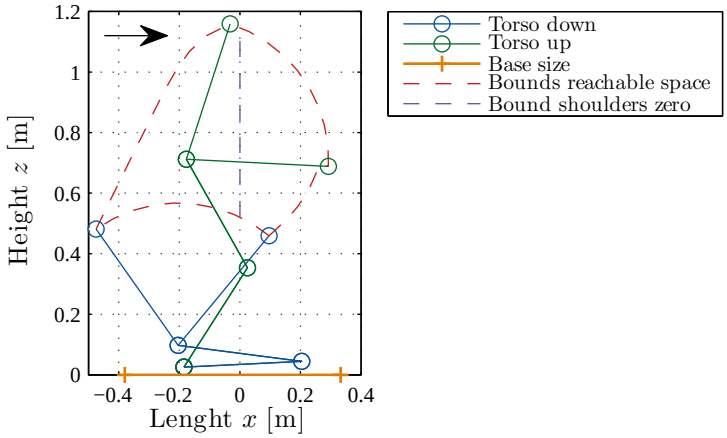


Figure 3.13: Side view of the kinematic orientation limits of the torso. Green and blue show the torso orientation where the legs are in the highest and lowest position respectively and both the maximum and minimum orientation of the trunk is shown. Red shows the bounds of the shoulders reachable space. Purple shows the shoulders center line position, meaning above center base. Orange shows the size of the base and the arrow denotes the forward direction of the robot.

An impression of the kinematic reachable space of the shoulders of the torso is given in Figure 3.13. The figure shows the orientation of the torso in the four limiting positions in blue en green. The the circles show the joint positions from bottom up the ankle joint, knee joint, hip joint and shoulder joint. The red line shows the edge of the shoulders reachable space and the purple line denotes the positions above center base. The drive train of the legs is designed such that the legs could move over the full range of motion while keeping the shoulders located at this center line. In case the trunk bends forward the brake can be applied on the legs such that the required motor limits are reduced. The reduced motor limits are desired to reduce cost. In theory this is a nice feature but in practice however this will be quit inconvenient as explained in the following example

Example: The robot needs to grab an object on a table which requires the robot to bend forward.

Joint goals are computed by the higher level software and the torso moves to this bending positions with use of the brake. If for some reason the object has moved when the robot wants to grab it there are new joint goals computed by the higher level software which result in a slight hight increase of the legs. In order to increase hight the trunk should move back to the centred position followed by changing the leg height and bending forward again.

This is a time consuming and unwanted method to make a minor change in orientation. Therefore the requirements for the legs and trunk are adjusted. In this section there are two different requirement sets analysed and in the end the most suitable solution is chosen:

1. The drive trains in both legs and trunk should be able to move the shoulders without using brakes over the reachable space on the right side of the purple line in Figure 3.13. This means the torso is able to move up and down and bend from center line forward and back to center line. This region is chosen because it is the reachable space that is most likely to be used when manipulating objects.

2. The drive trains should be able to freely move the torso over its complete kinematic range as enclosed by the red dashed line in Figure 3.13. This is the most convenient solution as it result in a dynamic reachable space equal to the kinematic reachable space and no limitations have to be added.

Another requirement for the torso mechanism is to stand up within 4 seconds. This means the legs should be able to move over the full range within four seconds. In order to get smooth motion a constant acceleration and deceleration at the start and stop of motion respectively will be used. With 0.95 rad being the angle range of the legs and using a one second acceleration and deceleration time this results in an acceleration of 0.32 $\frac{\text{rad}}{\text{s}^2}$ and a maximum velocity of 0.32 $\frac{\text{rad}}{\text{s}}$. Taking the same assumption for the trunk such that it should bend over the full range, which is from shoulders zero to maximum bend with the legs in the maximum position, results in an acceleration of 0.41 $\frac{\text{rad}}{\text{s}^2}$ and a maximum velocity of 0.41 $\frac{\text{rad}}{\text{s}}$.

A final remark should be made with respect to the designed drive train. The drive train refers to the combination of motor, gearbox and lead screw. In the design it is stated that both trunk and leg drive train should be able to deliver a pushing force up to ± 1500 N. The currently used combinations however can only deliver ± 1375 N at the trunk and ± 793 N at the legs, which is probably to low.

The goal in this section is to determine the combination of required force by drive train and load of the gas springs for the two defined reachable subspaces. The gas springs and load and mass of the arms are added to the identified model in (3.48) resulting in,

$$\vec{\tau}_m = \hat{\boldsymbol{M}}(\vec{q})\ddot{\vec{q}} + \hat{\boldsymbol{g}}(\vec{q}) + \hat{\boldsymbol{B}}(\vec{q},\dot{\vec{q}}) + \boldsymbol{g}_{springs}(\vec{q},\dot{\vec{q}}) + \boldsymbol{D}_{arms}\ddot{\vec{q}} + \boldsymbol{g}_{arms}(\vec{q}), \tag{3.49}$$

where $\boldsymbol{g}_{arms}$ contains both the mass of the arms and the additional load of 52 Nm. Note that the proportional friction component is multiplied with all three potential force contributions in this equation. With this equation the required motor torque at any combination of position, velocity and acceleration can be calculated.

3.5.2 Trunk analysis

The springs chosen by the design are two gas springs each loaded with a force of 800 N in extended position and supplying an additional 60 N when compressed. The effect these springs have on the drive train force is shown in Figure 3.14. Increasing or decreasing the load will shift the lines down and up respectively. The difference in force for extension and compression is almost equal to two times the 60 N difference because the springs are almost exactly aligned with the drive train.

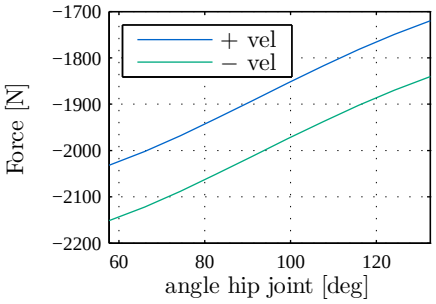


Figure 3.14: Effect of two springs (loaded with 800 N each) in the trunk on the required drive train force dependent on the angle of the trunk. Both in negative (compressed) as positive (extended) motion are shown.

In order to calculate the maximum forces that are required at a certain position of the trunk there are several contributing variables namely the leg orientation q_0, the load of the arms, the acceleration of the trunk $\ddot{q}_2$ and the direction of motion of the trunk $\dot{q}$. It is clear to see that maximum required forces appear when these variables are at the maximum or minimum of their range. The required lead screw force for all combinations of these maxima and minima are shown in Figure 3.15 without using the springs. In the top plots the velocity is positive and maximum positive accelerations are applied and in the bottom plots in negative direction. The left and right plots show the different applied loads by the arms. The blue lines show the maximum and minimum orientation of the legs. This means in order to meat requirement set 2 the area enclosed by these lines should stay within the drive train limits presented by the red dashed lines. In addition the green line is added which denotes the shoulders center position for each orientation of the legs. In order to fit the first requirement set the part enclosed by the blue lines and above the green line should be within the motor limits.

For the first requirement determining the required drive train forces and gas spring loads there are two combinations of variable settings that determine the range. The maximimum required force appears when the trunk has to accelerate in positive direction with a arm load of 52 Nm and the legs in the minimum position, as shown in the top right plot in Figure 3.15 by the light blue line. The opposite of these settings result in the maximum required negative force as shown in the bottom right plot by the dark blue line. Only adjusting the spring loads is not sufficient to cover the complete range and therefore the drive train should also be adjusted. The resulting changed forces are shown in Figure 3.16, with the grey lines representing the non loaded system. The springs used are loaded with 350 N and the drive train should be able to cover a force range of ± 2400 N.

A similar approach can be taken for the second requirement set by taking the center line as lower

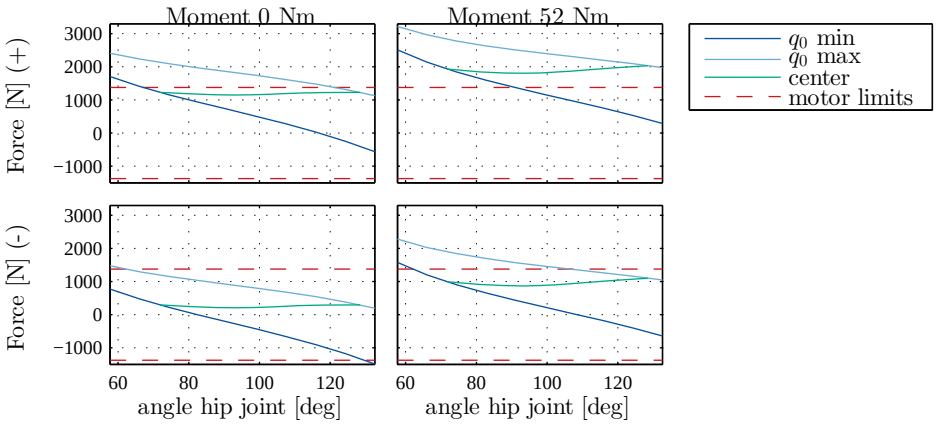


Figure 3.15: Forces required to move the trunk when no gas springs are used. The blue lines show the difference between maximal and minimal position of the legs, the left and right plots show the difference between no moment and maximal moment generated by the arms, the top and bottom plots show the difference in positive and negative velocity and acceleration, the red dashed lines show the motor limits and the green line shows the force to keep the shoulders above the center of the base over the full leg range.

limit. This results in spring loads of 675 N and a drive train range of ±1600 N, as shown in Figure B.7 in Appendix B.7. The reachable space of the trunk with these setting is shown in Figure 3.17, where the reachable space means the trunk is able to move from and to these positions with and without an additional load of 52 Nm.

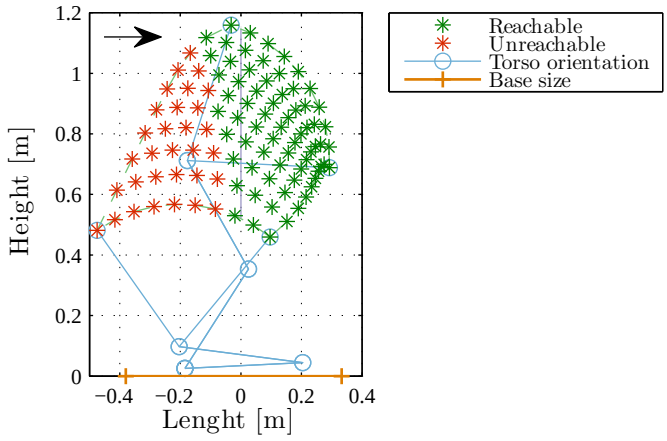


Figure 3.17: Reachable space of the shoulders by the trunk drive train with a spring load of 675 N and a drive train limit of ±1600 N. The green stars are reachable and the red are not.

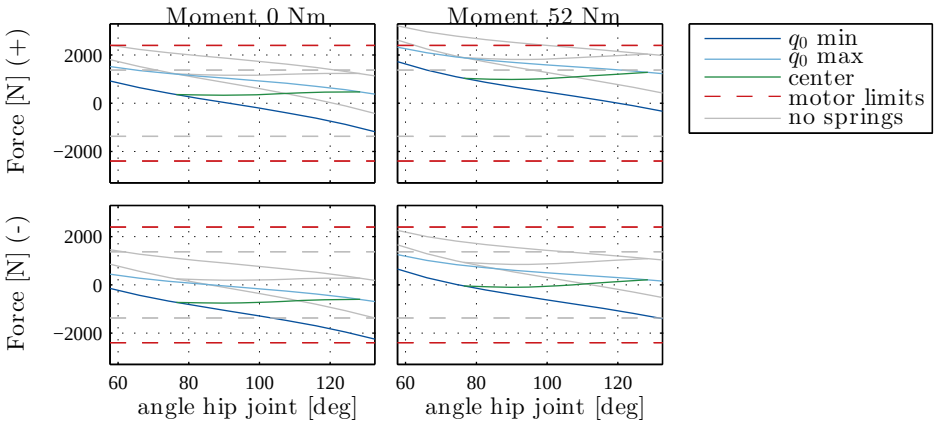


Figure 3.16: Forces required to move the trunk with a spring load of 350 N and the drive train limits increased to 2400 N. The transparent grey lines show the previous settings as in Figure 3.15.

3.5.3 Leg analysis

A similar approach as for the trunk can be applied on the legs. The required forces without gas springs are shown in Figure 3.18. It shows that the maximum forces occur in the top left and bottom right plot. In order to meet the first requirement which covers the full range the spring loads should be 575 N with a drive train force range of ±2500 N, as shown in B.7. For the second requirement set this results in a spring load of 250 N and a force range of ±1800 N. Note that for the legs the part below the green line representing the center line should be within the force limit range.

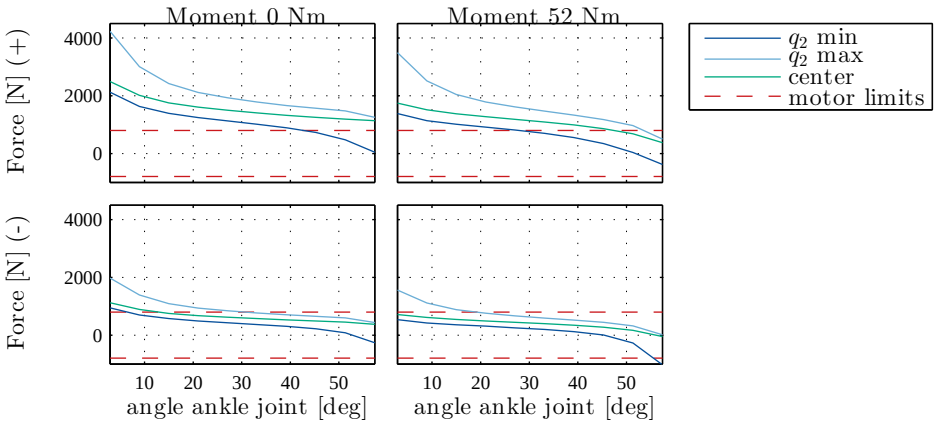


Figure 3.18: Required leg force without springs.

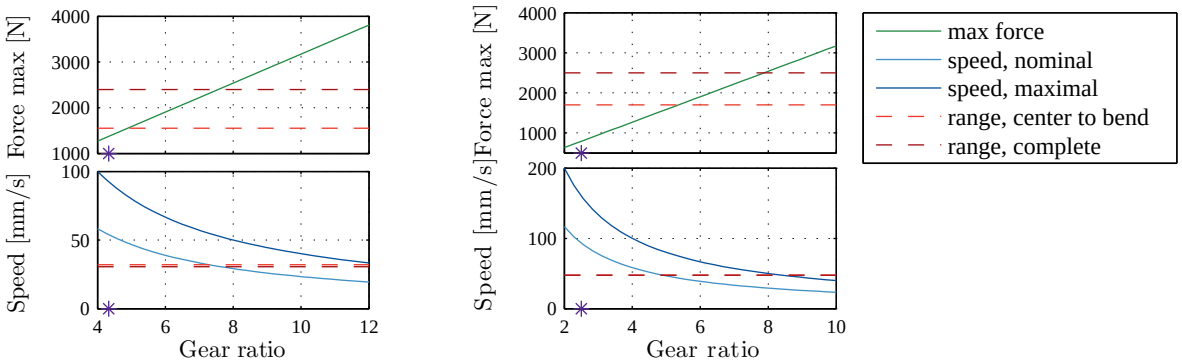
3.5.4 Spring and drive train choice

In this section it is shown that both drive trains cannot deliver the required force needed to move the torso mechanism over the desired range, either by requirement set 1 or 2. The required drive train forces and gas spring loads for both requirement sets are shown in Table 3.1. In order to change the drive train force range there are three parts that can be replaced, namely motor, gearbox and/or lead screw. The current force ranges are ± 1375 N for the trunk and ± 793 N for the legs.

Set	Range	Trunk			Leg		
		Force [N]	Speed [$\frac{\text{mm}}{\text{s}}$]	Load [N]	Force [N]	Speed [$\frac{\text{mm}}{\text{s}}$]	Load [N]
1	Complete	2400	32.0	350	2500	47.6	575
2	Center to bend	1600	30.5	675	1800	47.6	250

Table 3.1: Desired lead screw forces and speeds with the corresponding gas spring loads.

The simplest way to change this range by increasing the gear ratios. Increasing the gear ratios will also increase the required motor speed to move the torso over its full range within four seconds. The used motors have a nominal speed of 7000 rpm and a maximum speed of 12000 rpm. Figure 3.19 shows the effects of changing the gear ratio on the resulting lead screw force and nominal and maximum lead screw speeds. The purple star denotes the current gear ratio which is $4.3:1$ for the trunk and $2.5:1$ for the legs. Both pictures clearly show that there is room for increasing the gear ratios. The red dashed lines in the force plots show the required force to meet the requirement sets. In the speed plot they show which speed results in a four second duration to move over the complete range. For the trunk it shows that the sweet spot is a gear ratio of $8:1$, which will cover the full force range and put the nominal speed almost equal to the desired speed. In case of the legs a gear ratio of $6:1$ will meat requirement set 2. In order to cover the full kinematic range the maximum speed of the legs should be reduced a little the gearbox ratio should be $8:1$.



(a) Trunk. (b) Leg.

Figure 3.19: Force and speed changes by changing gear ratio. The solid lines show the drive train delivered values and the dashed lines show the required values, as can be found in Table 3.1.

The currently used motor gearbox combination in the trunk is from MAXON[5]. Because it is necessary to change the gearbox ratios and it is useful that the torso is not limited in its use it is chosen to meet the first requirement set such that the complete range of the torso is covered. The

first off the shelve available[5] gearbox ratio of $8:1$ or higher is a gearbox with a $12:1$ ratio. This means that the motor is used above its nominal speed. The resulting speed limit of the lead screw becomes $0.033\ \frac{\mathrm{m}}{\mathrm{s}}$. It should be noted that the desired speed of 0.032 is close to the limiting speed and it is possibly a good idea to reduce the speed of the trunk a little to have some safety margin.

The gearbox used in the legs is custom designed such that the motor could be mounted on the side of the lead screw in order to fit in the available space. It is geometrically not possible to increase this gear ratio because of the already tight fit of the motor against the lead screw. It is chosen to place an additional gearbox between the motor and the current gearbox. This demands for slight modifications in the torso box structure such that the full range can still be reached. The lowest gear ratio available of MAXON that fits on the motor is a $3.5:1$ which results in an overall ratio of $8.75:1$. As for the trunk the maximum speed of the lead screws in the legs should be limited below $0.045\ \frac{\mathrm{m}}{\mathrm{s}}$. If the four second requirement is stretched to five seconds it results in speeds of $24\ \frac{\mathrm{m}}{\mathrm{s}}$ and $35.7\ \frac{\mathrm{m}}{\mathrm{s}}$ which have some safety margin on the motor limits. The final choice and limits of the gearbox and spring combinations is shown in Table 3.2.

	Spring load [N]	Gearbox ratio	Maximum Force [N]	Nominal speed $[\frac{\mathrm{m}}{\mathrm{s}}]$	Maximal speed $[\frac{\mathrm{m}}{\mathrm{s}}]$
Trunk	350	$12:1$	2776	0.0267	0.0457
Leg	575	$2.5:1$ and $3.5:1$	3807	0.0194	0.033

Table 3.2: Final choice in gear ratios and spring loads (load per spring) and the resulting limits on lead screw force and speed by the motor limits.

3.5.5 Comparison AMIGO and SERGIO

With the changed gear ratios and adding the gas springs the reachable space of the torso becomes the space enclosed by the red dashed line in Figure 3.20. As a comparison the reachable space of AMIGO is shown presented by the purple line above the center of the base. The shoulders height range covered by SERGIO ranges from 0.46 m up to 1.16 m whereas that of AMIGO ranges from 0.62 m up to 0.95 m. The main advantage of the torso mechanism is the horizontal degree of freedom which allow the shoulders to bend up to 4 cm form the front of the base, which is useful to pick up objects from the floor or from a table.

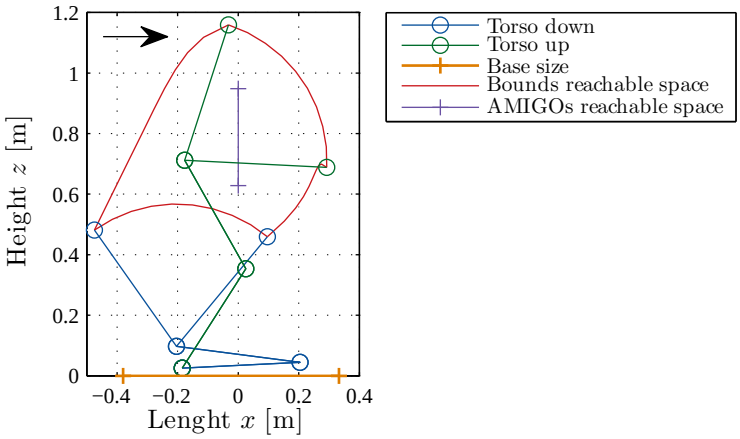


Figure 3.20: Comparison of the reachable space by the shoulders of SERGIO and AMIGO. The purple line shows the reachable space of AMIGO and the red dashed line encloses the reachable space of SERGIO.

3.6 Torso controller design

The low level controller should fit in the interface as shown in Figure 3.21. The higher level software sends three joint goal references $\vec{q}_r$ and receives the actual joint positions $\vec{q}_j$. From the hardware it receives the orientation of the motors $\vec{q}_m$ and it can set a motor torque as output. In addition the length of the gas springs is measured with callipers which should be used at start up to initialize the encoders, as will be discussed in the next section.

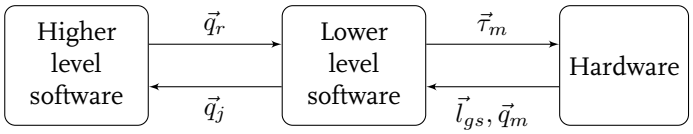


Figure 3.21: Interface of the low level software for the torso.

In this section the discussed control methods presented in Section 3.2 will be used to design a controller for the torso mechanism. A feedback control method is presented and implemented on the robot. For the feed forward compensation the identified model in the previous section will be used. Finally the performance of the controllers is shown with the use of a challenging reference trajectory.

3.6.1 Feedback

As shown in Figure 3.2 there are a lot of methods to choose from for feedback control. As discussed the main reason between independent and dependent joint control is determined by the gear ratio

between motor and joint and the reference speeds the link should be able to follow. The reference speeds of the joints are relative slow, as they should move over approximately a quarter round in four seconds. The gear ratios in the mechanism are a function of the joint angles.

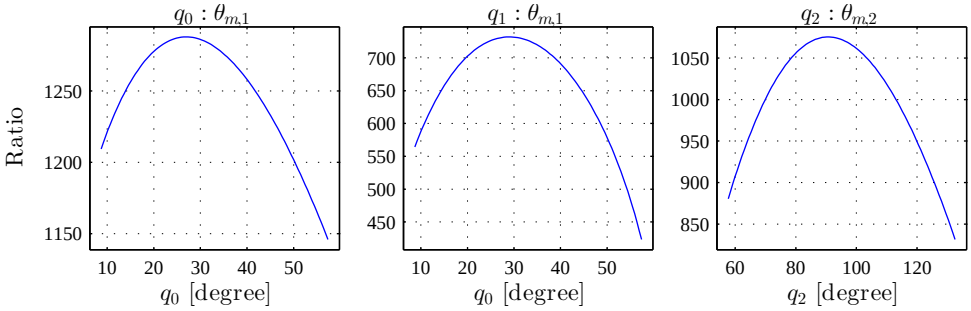


Figure 3.22: Gear ratios as function of the minimal coordinates q_0 and q_2.

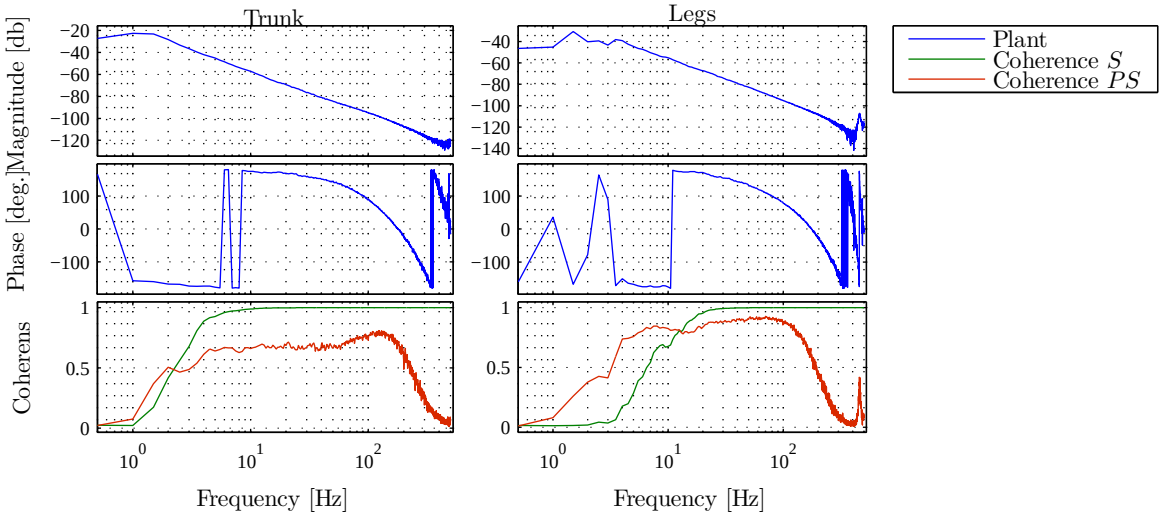
The gear ratios can be calculated by taking the partial derivative,

$$r_{gear}(q_j) = \frac{\partial \theta_m(q_j)}{\partial q_j}. \tag{3.50}$$

In Figure 3.22 the ratios are shown as a function of the independent coordinates. The left and right plot show the gear ratios from ankle joint and hip joint to their corresponding motor angle. The center plot shows the ratio from knee joint to the leg motor including the transfer by four bar guiding mechanism. From the left and center plot it can be concluded that there is a ratio of approximately $1:2$ between the ankle and knee joint. In literature typical gear ratios for independent joint control where between 20 to 200 or more [20]. All three plots clearly show a large gear ratio, meaning individual joint control can be applied on the system and non-linearities can be modelled as disturbances. The dominance of the motor dynamics is also shown in the simulation results as shown in Appendix B.5. Within the set point tracking problem as presented in Figure 3.21 and the dominance of the motor dynamics it is preferred to control the error on lead screw length instead of joint angle. This means the reference angle is converted to the lead screw length that serves as the reference for the controller.

The different choices in independent joint control are PD, PID or LQ optimal control. The main advantage of the last one shows itself in flexible joints. The torso mechanism is made out of sheet steel box structures making it a very rigid construction and flexibility is minimized, eliminating the main preference of LQ optimal control. Therefore a set point tracking controller will be used. The integral action is necessary to compensate for the forces gain from potential energy like the gas springs and gravity from both the torso mechanism as the arms of the service robot.

In order to tune a feedback controller an FRF is measured for both trunk and legs. FRFs are measured for the legs and trunk separately by applying noise in the input and using a sine with a frequency of 0.1 Hz and amplitude of 0.01 m lead screw displacement. The reference is used to minimize the effect of static friction. Figure 3.23 shows the average of multiple measurements at different orientations of the torso mechanism. The individual measurement results can be found in Appendix B.8. From the individual measurements it can be concluded that there is no real difference between different orientations of the trunk and indeed the gear ratio reduces the non linear effects.



(a) Trunk. (b) Leg.

Figure 3.23: Average of the plant response and coherence over multiple measurements for both the trunk and the legs.

Even by applying a load of 10 kg in the shoulders the FRF measurement of the trunk does not show a significant difference as can be found in the appendix. It should be noted that the coherence of the process sensitivity is below 1 which can be a result of non linear effects still present in the system. However the measured response is still a good estimation of the real plant in the frequency range from 5 Hz and 200 Hz. Therefore the average of the FRFs will be used to tune the feedback controller.

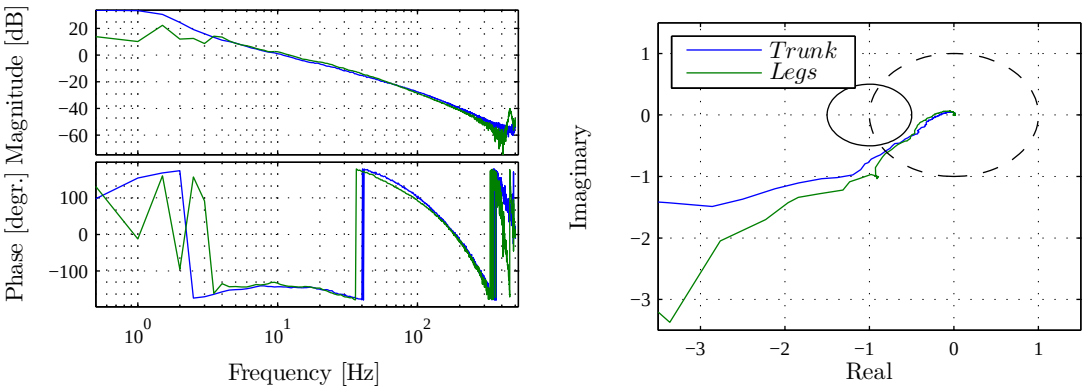
The results of the tuned controller is shown in the open loop response and Nyquist in Figure 3.24. Both controllers contain an integrator with a cut of frequency of 2 Hz, a lead lag filter with the lead frequency at 3.3 Hz and the lag frequency at 30 Hz and a gain of 270 and 250 for the trunk and leg respectively. These controllers result in a bandwidth for both trunk and legs of approximately 11 Hz, phase margins of 35°and modulus margins of 5 dB.

3.6.2 Feed forward

As explained in Section 3.2 feed forward is a good way to improve performance. In case of the single joint control approach a possibility is to only compensate for motor dynamics. However in Section 3.4 a multiple joint model of the complete torso mechanism is identified. In this section it is concluded that the main contributions in the required motor torque are the gravity, friction and inertia compensation. Adding these components to the input results in the following feed forward function,

$$\vec{\tau}_m = \hat{\boldsymbol{M}}(\vec{q}_r)\ddot{\vec{q}}_r + \hat{\boldsymbol{g}}(\vec{q}_r) + \hat{\boldsymbol{B}}(\vec{q}_r, \dot{\vec{q}}_r) \tag{3.51}$$

where subscript r stands for the reference trajectory. When the torso mechanism is adjusted the gas spring terms can be added to the model and if desired gravity compensation for the arms. With the replacement of the gearboxes it might be required to retune the gains in $\hat{\boldsymbol{M}}$ as the speeds of motor



(a) Bode.

(b) Nyquist.

Figure 3.24: Bode and Nyquist of the controlled plant for both trunk and legs.

and gearboxes are also increased.

3.6.3 Performance

The final control scheme used for the torso is shown in Figure 3.25, containing both feedback and feed forward.

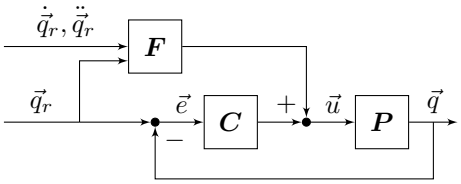


Figure 3.25: Control scheme used in the torso mechanism.

In order to validate the performance of both feedback and feed forward a similar reference trajectory is used as in de Roest [9], in which the torso has to stand up and bend in both top and bottom positions of the legs. The reference trajectory consists of the following steps,

- Start in a kneeled position with the shoulders at centreline.
- At time is 1 s, stand up and move the shoulders to the top center position.
- At time is 7 s, bend forward and back to center position.
- At time is 19 s, move back to the start position.
- At time is 25 s, bend forward and back to center position.

In Figure 3.26 the results of a measurement are shown. During these measurements the integrator is removed from the feedback controller to show that the feed forward compensates for most gravitational effects. The plots show that the error for both joints stay within 0.001 rad which means it is quit accurate. It also shows that there remains a static offset when the reference is constant because the integrator is not used. The torque contributions for both trunk and leg show that the feed forward is the main contribution in the torque input and the feedback is only used to cancel out disturbances and minor model mismatches. This verifies that the identified model is a good representation of the system.

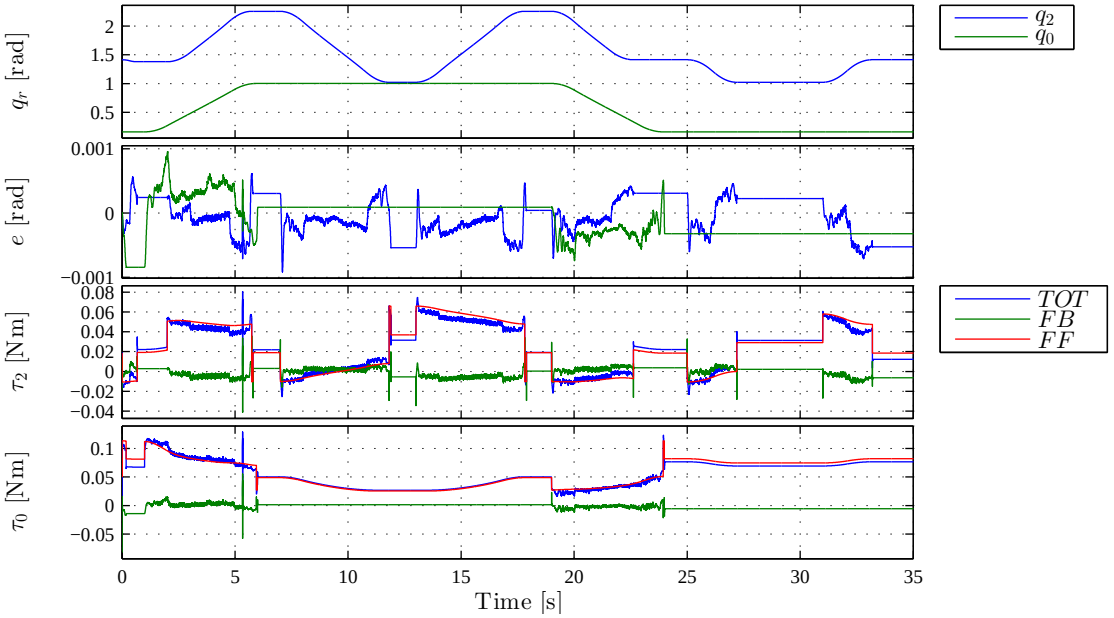


Figure 3.26: Measurement results of the stand up and bend trajectory. In the top two plots the reference and error of both trunk q_2 and leg q_0 are shown. In the bottom two figures the contributions in the control effort in motor torque of both trunk τ_2 and τ_0 are shown, where FB is the feed back, FF the feed forward and TOT the total torque.

In the final controller the integrator is still desired because it reduces the static error, which results in a smaller error on positioning the end effector. It also compensates for the unmodelled dynamics like the mass of the arms and objects that are lifted by the robot. It is possible to include a model of the seven DOF arm in the torso feed forward to compensate for that. However if the performance of controllers with or without feed forward are compared it shows only minor improvements, as can be seen for the trunk in Figure 3.27 and for the legs in Appendix B.9. The blue line shows the result with only feedback which can result in larger errors when motion is started, as occurs at 32 seconds after start up. This is caused by the integrator in the feedback in combination with the high static friction in the system. It allows the control output to vary over a range of approximately 0.05 Nm (25% of the total range) without moving the trunk. At this point it shows worst case scenario, because the current torque is at one end of the range and to follow the next reference point it has to be on the otter side which takes a little time resulting in the peak values in error and control output. A similar effect happens with the added feedforward (red line), but the range at which the integrator can vary is reduced by a half because of the coulomb friction compensation. Note that in both

cases the error peak only reaches up to ± 0.003 rad which is still a negligible value when the torso is moving.

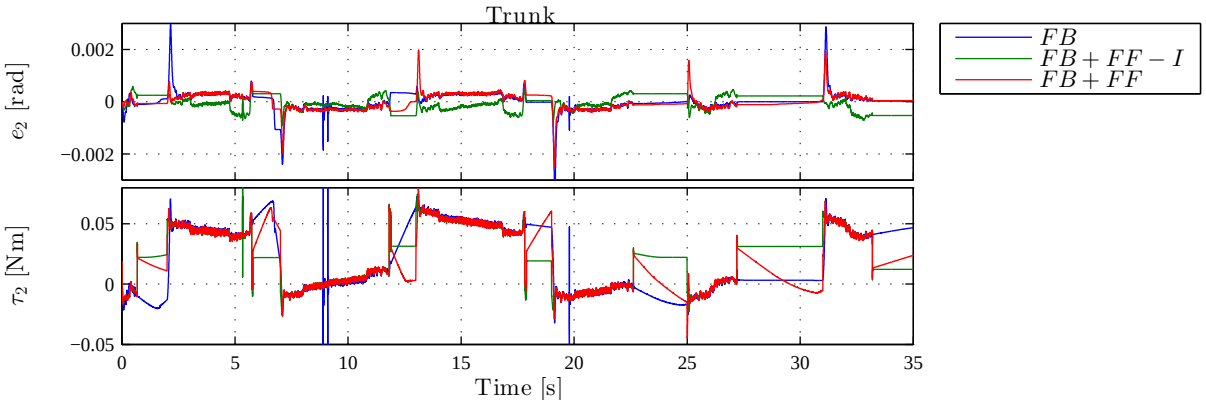


Figure 3.27: Measurement results of the stand up and bend trajectory for the trunk with different controllers, where FB is the proposed feedback controller, FF the feed forward controller and $-I$ means feedback without the integrator. The plots show the joint error and controller output torque.

Another point that should be noticed is the large error peaks at 9 s. At this point there are four consecutive data packages that are missed between the hardware and PC. This means that the measured error increases linearly with the reference velocity while the actual error only changes a little. This is clearly a downside of the soft real-time implementation of a control loop. The exact reason for the missed packages is not known and a temporary solution is to add low pass filter that reduces the large peaks in the error.

Concluding a good tuned feedback controller with integral action is able to result in good set point tracking. Especially in the application of a service robot where the velocities are small which means gravitational forces can be easily accounted for by the integrator. A complete model based feed forward can be added to slightly increase performance during motion. However the results in this section shows that only a coulomb friction compensation is an easy to apply feed forward that results in good similar improvements.

3.7 Final implementation

In this section the implementation of the torso controller in the Orocos real-time tool kit is presented and briefly discussed, as it is quit similar to the previous discussed implementation of the base controller. The main differences can be found in the additional conversion components that transfer joint angles $\vec{q}$ to lead screw lengths, and vice versa and the initialization component.

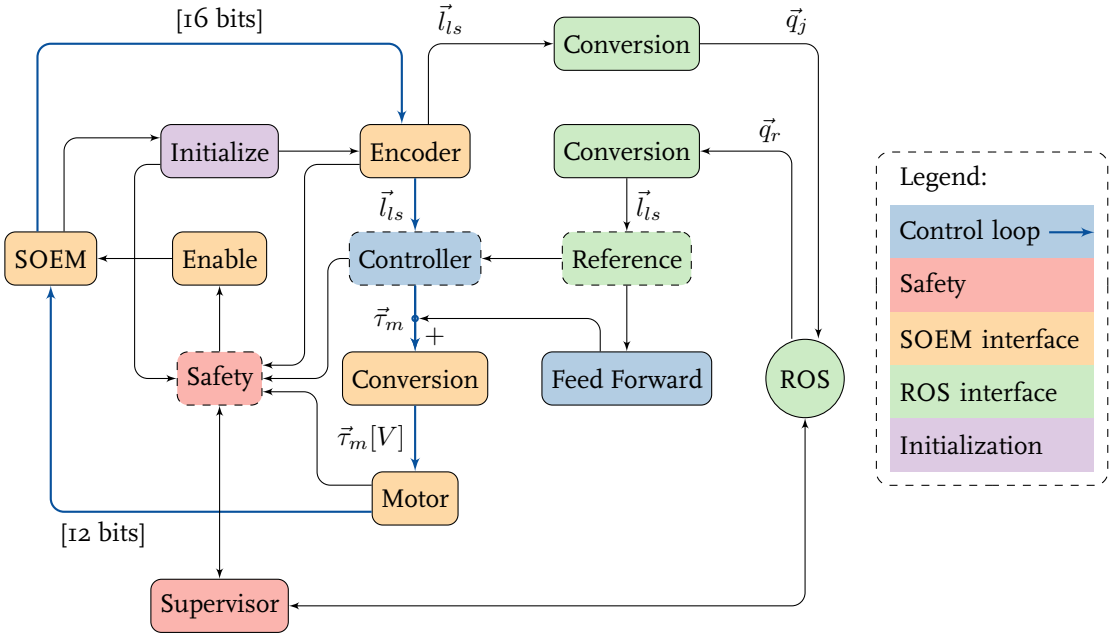


Figure 3.28: Orocos implementation of the low level torso controller.

The initialization component is used to initialize the encoder at start up. On one of the gas springs in both trunk and leg a digital calliper is mounted that measures an absolute position. With use of the non-linear relations described in Appendix B.3 the relation between gas spring length and lead screw length can be determined. The initialization component is also connected to the safety component such that the system can only start when encoders are initialized. In this way a homing procedure where the joints first have to move to an end stop as used in AMIGO is obsolete. This means the torso mechanism is immediately ready at start up.

In the torso some additional safeties are added with respect to the base. Brakes are used to keep the torso in position when the control loop is shut down. The safety component monitors the measured position, error and control output to ensure the system stays within bounds. When one of the monitored signals exceeds the allowed limits the safety component will apply the brakes on the system and disables outputs.

As discussed in Section 3.5 the current drive train is not able to reach all orientations of the torso that are reachable by its kinematics. At this point the range of the torso is not limited. However with the safety implementations the system will go into error when control outputs or errors become to large, meaning it is still safe to use.

Chapter 4

Conclusions & Recommendations

The newly designed and produced service robot SERGIO has a modular design consisting of a base, torso, two arms and a head. In order to get SERGIO operational it requires design and implementation of the low level control, which is successfully done in this thesis for both base and torso. Based on literature several control methods are compared, analysed and tested resulting in the final operational implementation on the robot. In this chapter the concluding remarks on the design choices are given followed by some recommendations.

4.1 Conclusions

Base

With use of a three point measurement a MIMO FRF is constructed of the base of SERGIO. Based on the coherence it is concluded that the diagonal terms form a good representation of the system. Coherence on the off-diagonal terms cannot guarantee correct identification, but still the model was accurate enough to analyse the system and design a proper controller.

Based on an RGA analysis on the model it is determined that decoupled control on each of the robots DOF is preferred over coupled control. It is also noticed that, depending on platform and application, control on the wheel orientations is preferred for safety reasons. When the wheels have good traction, and stay in touch with the floor, decoupled control is the way to go. For SERGIO this iss assumed based on the suspended base and the domestic environment. It is concluded that there is no real advantage in control on position error over control on velocity error. With use of a challenging reference trajectory a time domain comparison lead to the preference for control on the velocity error, because the mean square error was half of the mean square error with velocity control. A decentralized controller on the velocity errors in robot space is successfully implemented and stability is guaranteed with use of MIMO stability theorem found in literature.

The main design goals for the base where to remove slip and make it possible to overcome

small obstacles present in a domestic environment like doorsteps and power cables. Validation experiments for these requirements are successfully performed. The experiments included driving straight through a corridor with a constant velocity (between 0.25 $\frac{\mathrm{m}}{\mathrm{s}}$ and 1 $\frac{\mathrm{m}}{\mathrm{s}}$) and moving over a small doorstep in a straight approach and approach under a small angle. After evaluating the velocity errors, measured with an external sensor, it is concluded that both experiments show slip and drifting sideways is reduced to a minimum. Conducting the same experiments on AMIGO shows that the base performance is greatly improved, since AMIGO does show slip, resulting in drift sideways. This validates the two main design goals.

Torso

A model is made of the torso mechanism consisting of an ankle-, knee- and hip joint. The design is used to determine the non-linear kinematic relations between joints, drive train and springs. With use of literature and the kinematic relations a multivariable joint model is made of the mechanism, including joint and drive train dynamics. Simulations resulted in a simplified model by removing Coriolis and centrifugal terms, as their contribution is minimal. Finally, friction terms are added to the model and experiments are performed resulting in estimates of the unknown model parameters. The identified model contains gravitational-, inertial- and friction terms.

The identified model is used to determine the required gas spring loads and drive train forces. It is concluded that the drive trains of both legs and trunk cannot deliver enough force to cover the kinematic reachable space of the torso mechanism and should therefore be adjusted. The next section gives the recommendations on these adjustments.

The non-linear relations between actuator and joint orientation result in a position dependent gear ratio. It is calculated that the gear ratio from the motors to the joints are high, where the lowest gear ratio is from the leg motor to the knee joint, which varies between $450:1$ and $800:1$. This results in dominance of the motor dynamics over the joint dynamics and therefore it is chosen to use control on the lead screw length, which is linearly related to the motor angle. Because of the large gear ratio the non-linearities can be seen as disturbances and a single joint control approach is used. An FRF is measured and used to design a feedback controller. Using only feedback including integrators results in good set point tracking as the static joint errors stay below 0.0002 rad. In worst case scenario for both ankle- and hip joint errors this results in an error on the shoulder position of 0.0002 m, which is negligible for the applications a domestic service robot is used for.

The inverse dynamics of the identified model are added as feed forward terms to reduce the error during motion. Since the feedback output reduces to a noisy signal around zero, it is validated that the identified model is a good representation of the system.

Comparing measurements with- and without feed forward results in the conclusion that there is only a minor improvement in error reduction. Only peak values at the start of motion are reduced. This result could also be accomplished by tuning a feed forward based on motor speeds and accelerations, compensating for coulomb friction and inertia respectively.

The main design goals for the torso where reducing play and increasing its range. The main play in the robot is induced by the suspended base rather than by play in the rigid torso structure. It is shown that the adjusted design resulted in a large increase in range. The torso mechanism is able

to position the shoulders horizontally up to four centimetres distance from the front of the base. In height it can cover a range from 0.46 m up to 1.1 m. This is a great improvement compared to AMIGO whose height range is from 0.62 m up to 0.95 above the center of the base, without a horizontal degree of freedom.

Implementation

The low level controller for both the base and the torso are successfully implemented on the robot in the provided software architecture. With use of absolute sensors and a designed initialization component the torso is immediately ready to use at start-up, meaning there is no time consuming calibration procedure needed. Safety measures are implemented that stop the system when bounds on error, -reference, -position or -control output are exceeded. The implementation is already put to the test with success at the German Open in Magdenburg during the spring of 2015, where team Tech Unided finished first with SERGIO and AMIGO.

4.2 Recommendations

Base wheels

Both the wheels and motors in the base are used outside their specifications. The wheels in the base are specified to lift a weight up to 15 kg each. The total weight of the base is estimated at 80 kg (base 36.3 kg, torso and head 23.4 kg, two arms 10 kg each) without the covers meaning the load on each wheel will be at least 20 kg. The hardware limits on motor input are stretched to overcome the large friction and enable sideways movement. When the robot becomes fully operational, with both arms in place, it should be evaluated if no damage is done to both wheels and motors.

Torso spring stiffness and gearbox replacement.

To ensure that the drive train can deliver enough force the best and cheapest way is to adjust the gearbox ratios. The optimal gearbox ratio is $8 : 1$ for both legs and trunk. Using of the shelve[5] available gearboxes results in the following recommendation,

- *Trunk:* Replace the currently used gearbox from $4.3 : 1$ to a $12 : 1$ ratio. By combining this with two gas springs, loaded with 350 N each, the complete range is covered. It should be kept in mind that this limits the maximum lead screw speed to 0.045 m/s by the limits in motor speed.
- *Legs:* Add an additional off the shelve gearbox of ratio $3.5 : 1$ between the motor and the currently used gearbox with ratio $2.5 : 1$. By combine this with two gas spring, loaded with 575 N each, the complete leg range is covered. The speed limits of the legs will reduce to 0.033 m/s.

By placing the springs and replacing the gearboxes the dynamics of the system change. It is necessary to adjust the controller to work with the changed hardware.

Currently used software architecture

It is recommended to investigate whether the architecture used to implement the low level control could be improved. There are two main issues causing problems in the control design. First the communication between software and hardware tends to miss multiple consecutive data packages at random moments. This results in unwanted peak values in both error and controller output. At this point low pass filters are used to reduce the peak values. Secondly the loop time of the control loop running on the PC is not always reached. Currently the control loop is implemented in Orocos by placing components in a chain where each component has a specific function. Data transfers between components result in varying time delays which are summed over the loop. It happens that the loop time is not reached because of this effect. If both effects are removed it will improve performance. It should be kept in mind that the configurability of the current structure is a good feature and should be preserved.

Bibliography

[1] Bobbie Robotics, 2013. URL http://www.bobbierobotics.nl/.

[2] MATLAB, 2013. URL mathworks.com.

[3] RoboEarth | A World Wide Web for Robots, 2013. URL http://roboearth.org/.

[4] Robocup 2014, 2014. URL http://www.robocup2014.org/.

[5] Maxon Motors, 2015. URL http://www.maxonmotor.nl/maxon/view/content/index.

[6] H. Asama, M. Sato, L. Bogoni, H. Kaetsu, A. Matsumoto, and I. Endo. Development of an omnidirectional mobile robot with 3 DOF decoupling drive mechanism. *Proc. IEEE Int. Conf. on Robotics and Automation, Nagoya, 1995*, 1995.

[7] H. Bruyninckx. Open robot control software: the OROCOS project. volume 3, 2001.

[8] T.T.G. Clephas. Design and control of a service robot. Master's thesis, Eindhoven University of Technology, Eindhoven, August 2011.

[9] A. de Roest. Mechanical design of a base and torso for a domestic service robot. Master's thesis, Eindhoven University of Technology, Eindhoven, 2013.

[10] Derks. Mechanical design of a 7-dof robot arm for care robots. Master's thesis, Eindhoven University of Technology, Eindhoven, 2013.

[11] Alexander Gloye, Cuneyt Goktekin, Anna Egorova, Oliver Tenchio, and Raul Rojas. Learning to Drive and Simulate Autonomous Mobile Robots. In *RoboCup 2004: Robot Soccer World Cup VIII*, number 3276 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-25046-3, 978-3-540-32256-6.

[12] Kyung-Lyong Han, Oh Kyu Choi, In Lee, Inwook Hwang, J.S. Lee, and Seungmoon Choi. Design and control of omni-directional mobile robot for Mobile Haptic Interface. In *International Conference on Control, Automation and Systems, 2008. ICCAS 2008*, October 2008.

[13] Kyung-Lyong Han, Oh-Kyu Choi, Jinwook Kim, Hyosin Kim, and J.S. Lee. Design and control of mobile robot with Mecanum wheel. In *ICCAS-SICE, 2009*, August 2009.

[14] Ehsan Hashemi, Maani Ghaffari Jadidi, and Navid Ghaffari Jadidi. Model-based pi-fuzzy control of four-wheeled omni-directional mobile robots. *Robotics and Autonomous Systems*, 59(11), November 2011. ISSN 0921-8890.

[15] Lih-Chang Lin and Hao-Yin Shih. Modeling and Adaptive Control of an Omni-Mecanum-Wheeled Robot. *Intelligent Control and Automation*, 4(2), May 2013. ISSN 21530653.

[16] Patrick Muir and Charles Neuman. Kinematic modeling of wheeled mobile robots. Technical Report CMTJ-RI-TR-8G-12, The Robotics Institute, Carnegie Mellon University, Pittsburg, June 1986.

[17] TUe Robotics. rtt control components, 2015. URL `https://github.com/tue-robotics/rtt_control_components.git`.

[18] Raul Rojas and Alexander Gloye Förster. Holonomic Control of a robot with an omni-directional drive. 2006.

[19] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley, 2 edition, 2005. ISBN 13978-0-470-01167-6 (H/B).

[20] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley, 2006.

Appendix A

Base

A.1 Base drive train specifications

In the table below all important transitions in the base drive train are given.

Component	From	To	Ratio		Range	
I/O board output	Board	Volt	0.0024	V/bit	± 5	V
Amplifier	Volt	Current	10	A/V	± 50	A
Motor (nominal)	Current	Torque	0.0292	Nm/A	± 1.2	Nm
Gearbox	Motor	Wheel	$\frac{9}{169}$			
Sensor	Pulses	Rotations	2000	counts/round		
I/O board input	Pulses	16 bits value	1	count/bit	$+65535$	bit

Table A.1

A.2 FRF measurements

In this section some additional results of the FRF measurement on the floor are shown. Figure A.1 shows a magnitude plot of the completes system from wheel torques to wheel orientations. Figure A.2 shows the coherence of the sensitivities for the diagonal terms and off-diagonal terms. It clearly shows that it cannot be guaranteed that the off-diagonal terms are good estimates of the system. Another change in dynamics shows itself when measuring the FRF on a compliant or non compliant floor as shown in Figure A.3. The change in damping results in a change in a shift of the change of mass line from 15 Hz to 10 Hz approximately. In Figure **??** the diagonal elements of the transferred system to robot space are shown by the blue line. An additional measurement is taken where the robot is controlled in its three directions of freedom and noise is injected to one of the directions instead of one wheel. These results are presented by the green line. Remarkable is the change in transfer int he x direction. It shows changing dynamics which is probably caused by the fact that

the rollers on the outside of the wheel are not used in forward direction. From these Figures it can be concluded that there are uncertainties in the model and these should be taken into account when designing a controller. Finally the diagonal elements of the transfers in velocity domain in robot space is shown in Figure A.6.

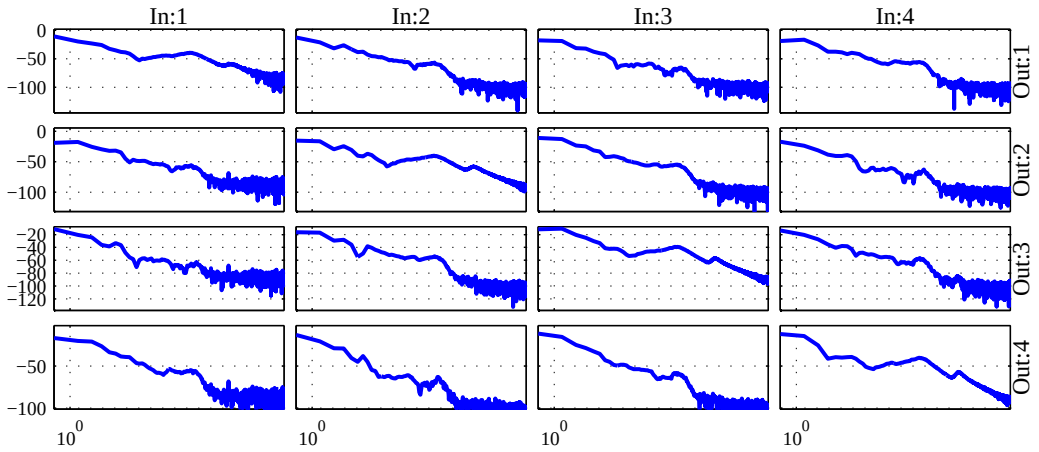


Figure A.1: Magnitude plot of the measured MIMO FRF on the floor.

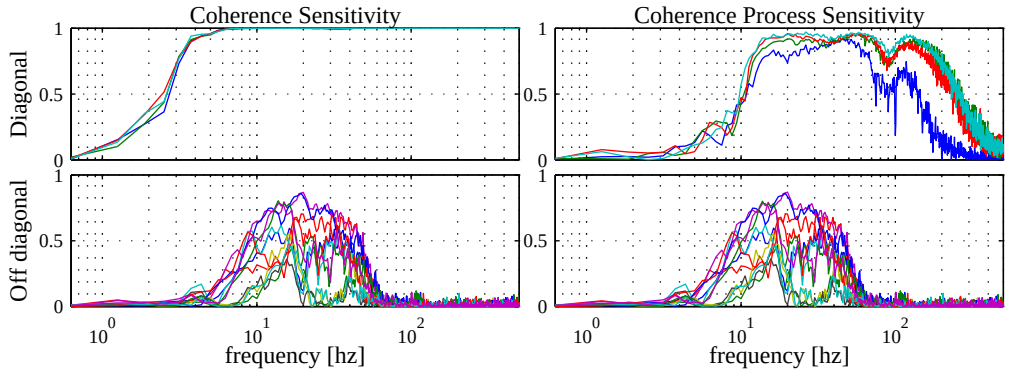


Figure A.2: Sensitivities of the diagonal terms in the top plots and the off-diagonal terms in the bottom plots.

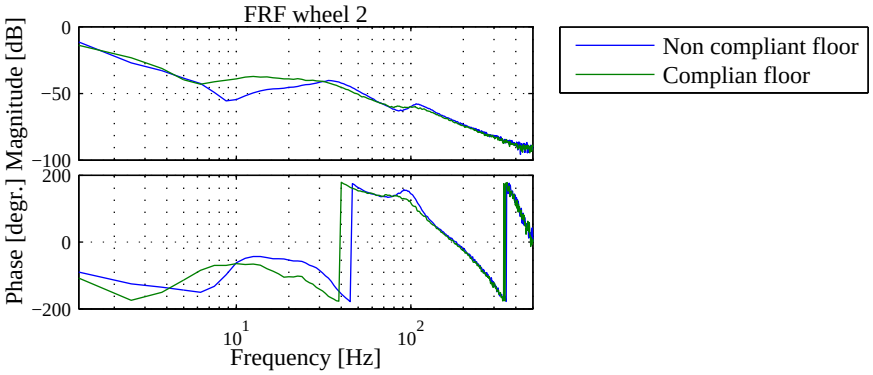


Figure A.3: Comparison FRF measurement on a compliant and non compliant floor.

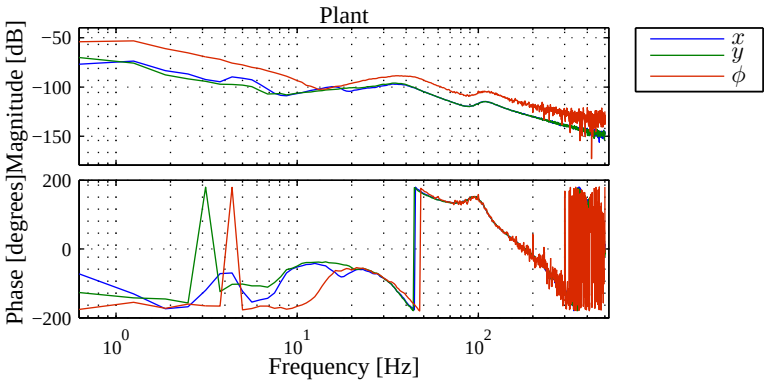


Figure A.4: FRF containing the diagonal terms of the transferred system from wheel to robot space.

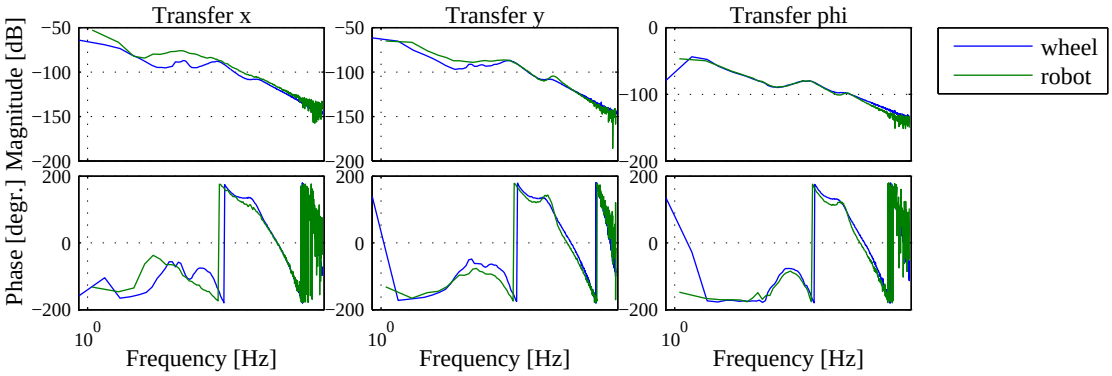


Figure A.5: Diagonal elements of the system represented in the robot space. The blue line is the transferred system out of the measured wheel transfers and the green line is explicitly measured on the three DOF of the robot with a reference velocity in these directions.

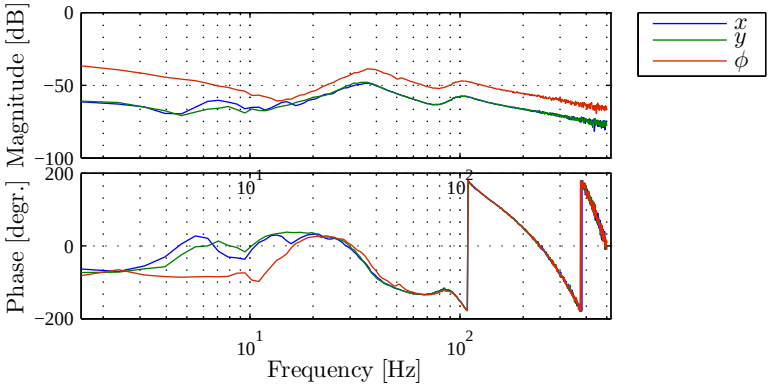


Figure A.6: Diagonal terms of the MIMO FRF from control output to velocities in the robot space. Measured with a velocity controller.

A.3 Feed forward

The two figures in this section show the tuning results in the x and ϕ direction of the base.

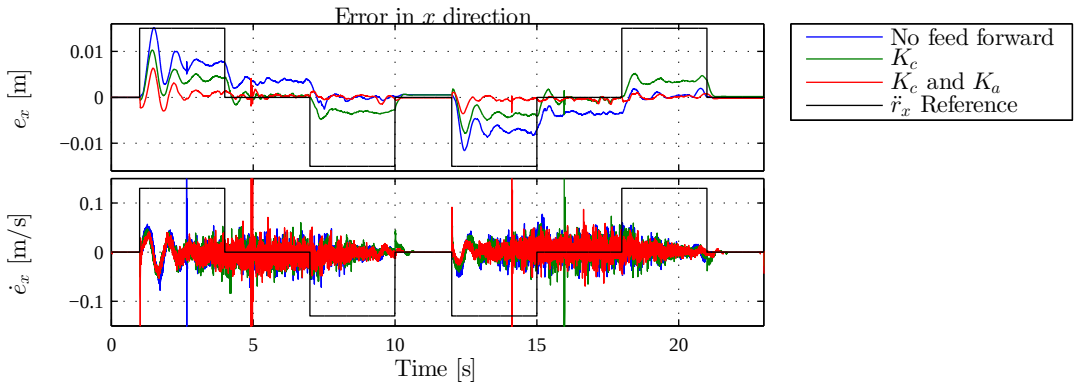


Figure A.7: Effect of the feed forward in x direction on the position and velocity error. The black line shows a scaled version of the acceleration reference.

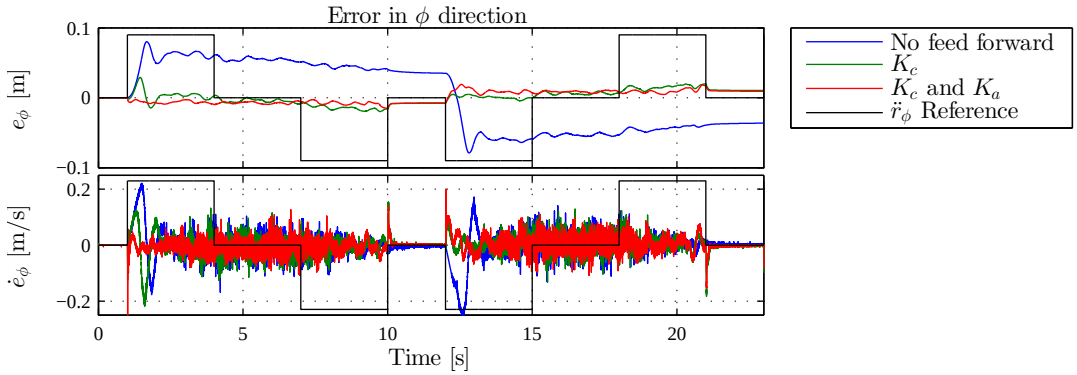


Figure A.8: Effect of the feed forward in ϕ direction on the position and velocity error. The black line shows a scaled version of the acceleration reference used.

A.4 Performance

In this appendix the estimated errors by Odometry and IMU of both SERGIO and AMIGO is shown for the corridor experiments for different constant forward velocities. For SERGIO also the experiments with a constant velocity in sideways direction are shown.

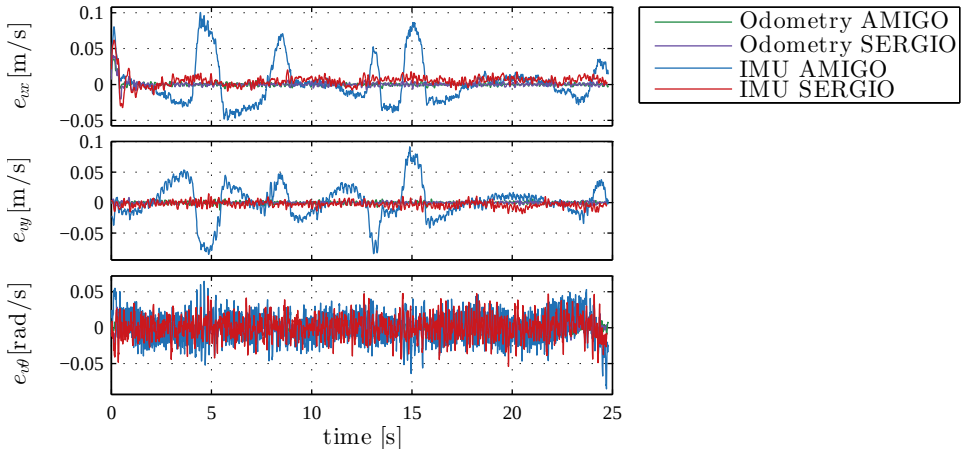


Figure A.9: Velocity error for both robots with a reference velocity of $0.25\ \frac{\text{m}}{\text{s}}$ in forward diriction.

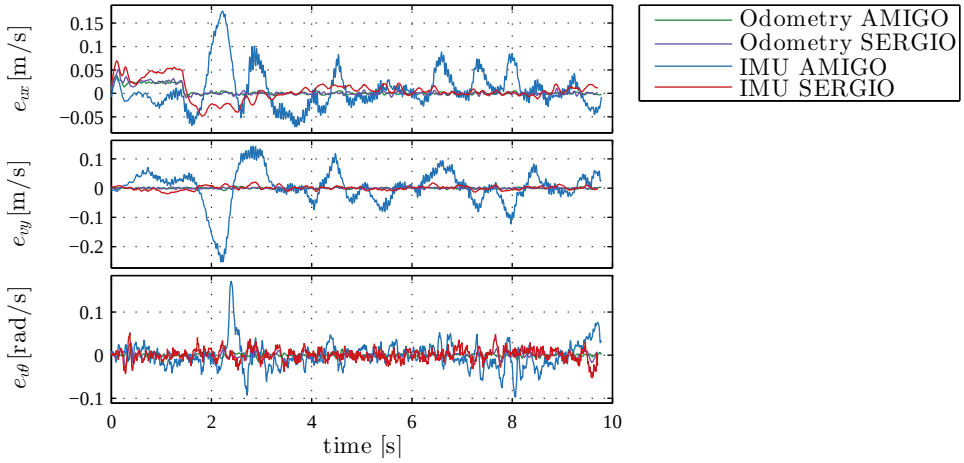


Figure A.10: Velocity error for both robots with a reference velocity of $1.0\ \frac{\text{m}}{\text{s}}$ in forward diriction.

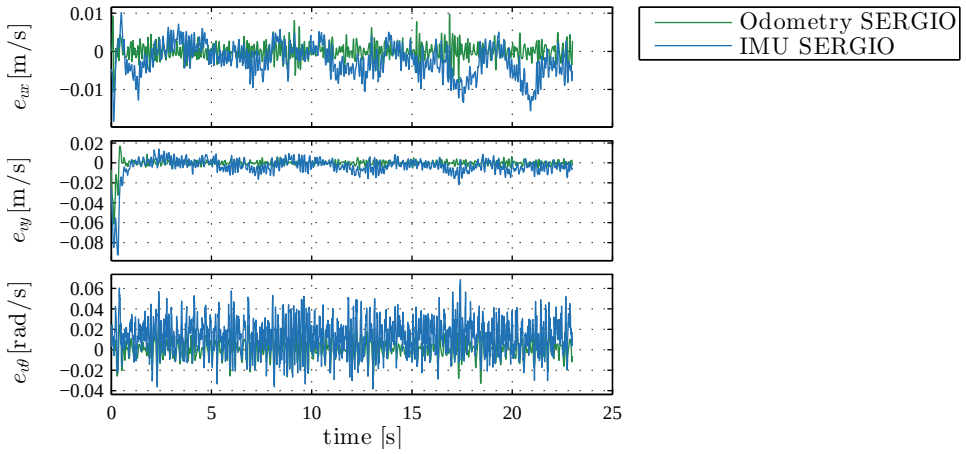


Figure A.11: Velocity error for both robots with a reference velocity of $0.25\ \frac{\mathrm{m}}{\mathrm{s}}$ in sideways diriction.

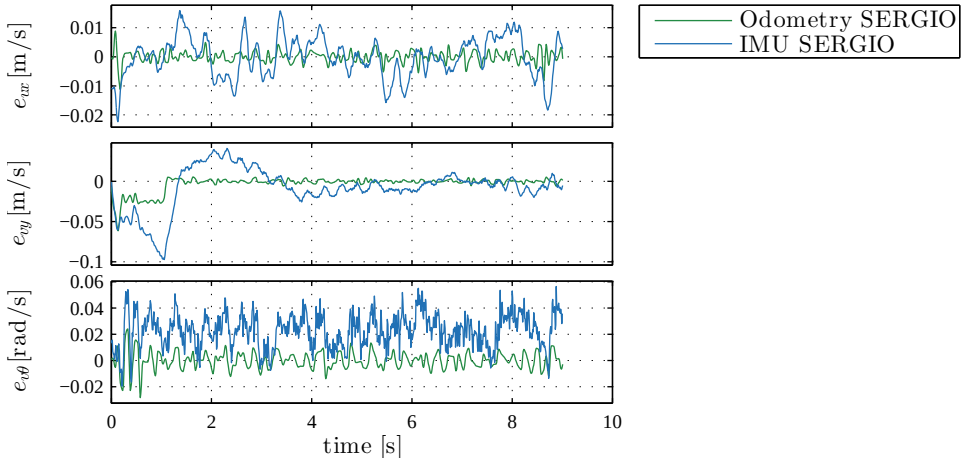


Figure A.12: Velocity error for both robots with a reference velocity of $0.75\ \frac{\mathrm{m}}{\mathrm{s}}$ in sideways diriction.

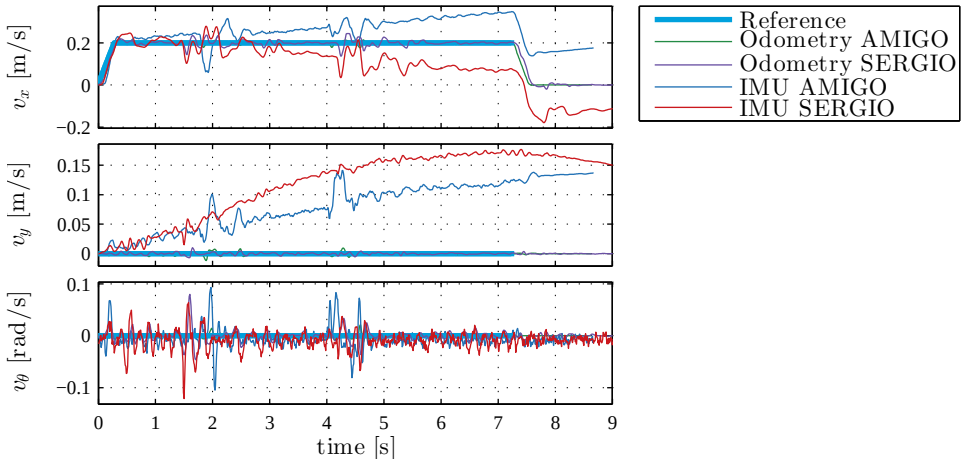


Figure A.13: Velocity for both robots with a reference velocity of $0.2\ \frac{\mathrm{m}}{\mathrm{s}}$ in forward direction while encountering a small doorstep.

Appendix B

Torso

B.1 Specifications

In the table below all known measures of the torso can be found. The motor and I/O specifications of the hardware are the same for as for the base shown in Appendix A.1.

Parameter	Symbol	Value	
Gravitational constant	g	9.81	$\frac{\mathrm{m}}{\mathrm{s}^2}$
Length lower leg	l_1	0.39	m
Length upper leg	l_2	0.4115	m
Length trunk	l_3	0.47	m
Distance ankle joint to lead screw 1 connection	d_{ls1}	0.3509	m
Distance hip joint to lead screw 2 connection	d_{ls2}	0.079	m
Lead of the lead screws	L_{ls}	0.002	m
Gear ratio leg motor (lead screw/motor)	r_{gear1}	$\frac{5}{2}$	-
Gear ratio trunk motor (lead_screw/motor)	r_{gear2}	$\frac{13}{3}$	-

Table B.1

B.2 State space representation of a DC motor driven joint

The state space representation of the system in Figure 3.4 is,

$$\dot{x} = Ax + Bu,$$
$$y = C^T x + Du,$$

with matrices,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{J_l} & -\frac{B_l}{J_l} & \frac{k}{J_l} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k}{J_m} & 0 & -\frac{k}{J_m} & \frac{B_m}{J_m} \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix},$$
$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \qquad D = \begin{bmatrix} 0 \end{bmatrix},$$

and vectors,

$$x = \begin{bmatrix} \theta_l & \dot{\theta}_l & \theta_m & \dot{\theta_m} \end{bmatrix}^T,$$
$$y = \theta_l,$$
$$u = \tau_m.$$

B.3 Kinematic relations.

For the parameters and dimensions used in this appendix see Figures 3.1b and 3.5. The length of the lead screw in the leg and trunk,

$$l_{ls1}(q_0) = \sqrt{\overline{AC}^2 + \overline{AE}^2 - 2\overline{AC}\ \overline{AE}\cos(q_0 + \angle CAG)}, \tag{B.1}$$
$$l_{ls2}(q_2) = \sqrt{\overline{GJ}^2 + \overline{JK}^2 - 2\overline{GJ}\ \overline{JK}\cos(q_2 - \angle GJH - \angle KJL)}. \tag{B.2}$$

The length of the gas springs in the leg and trunk,

$$l_{gs1}(q_0) = \sqrt{\overline{AB}^2 + \overline{AF}^2 - 2\overline{AB}\ \overline{AF}\cos(q_0 + \angle BAG)}, \tag{B.3}$$
$$l_{gs2}(q_2) = \sqrt{\overline{IJ}^2 + \overline{JK}^2 - 2\overline{IJ}\ \overline{JK}\cos(q_2 - \angle GJI - \angle KJL)}. \tag{B.4}$$

The angle at which the force is applied to the lower leg and trunk,

$$\theta_{ls1}(q_0) = \arccos\left(\frac{l_{ls1}^2(q_0) + \overline{AE}^2 - \overline{AC}^2}{2l_{ls1}(q_0)\overline{AE}}\right), \tag{B.5}$$
$$\theta_{ls2}(q_2) = \arccos\left(\frac{l_{ls2}^2(q_2) + \overline{JK}^2 - \overline{HJ}^2}{2l_{ls2}(q_2)\overline{JK}}\right). \tag{B.6}$$

B.4 Locations and velocities of the centres of mass

Locations of the centres of mass, with q_1 a function of q_0:

$$x_{cm1} = l_{cm1}\cos(q_0), \tag{B.7}$$
$$z_{cm1} = l_{cm1}\sin(q_0), \tag{B.8}$$
$$x_{cm2} = l_1\cos(q_0) - l_{cm2}\cos(q_1 - q_0), \tag{B.9}$$
$$z_{cm2} = l_1\sin(q_0) + l_{cm2}\sin(q_1 - q_0), \tag{B.10}$$
$$x_{cm3} = l_1\cos(q_0) - l_2\cos(q_1 - q_0) + l_{cm3}\cos(q_2 - q_1 + q_0), \tag{B.11}$$
$$z_{cm3} = l_1\sin(q_0) + l_2\sin(q_1 - q_0) + l_{cm3}\sin(q_2 - q_1 + q_0), \tag{B.12}$$
$$x_{cm4} = l_1\cos(q_0) - l_2\cos(q_1 - q_0) + l_3\cos(q_2 - q_1 + q_0), \tag{B.13}$$
$$z_{cm4} = l_1\sin(q_0) + l_2\sin(q_1 - q_0) + l_3\sin(q_2 - q_1 + q_0). \tag{B.14}$$

Velocities of the centres of mass with q_1 a function of q_0 and $\dot{q_1}$ a function of q_0 and $\dot{q_0}$:

$$\dot{x}_{cm1} = -l_{cm1}\sin(q_0)\dot{q_0}, \tag{B.15}$$
$$\dot{z}_{cm1} = l_{cm1}\cos(q_0)\dot{q_0}, \tag{B.16}$$
$$\dot{x}_{cm2} = -l_1\sin(q_0)\dot{q_0} + l_{cm2}\sin(q_1 - q_0)(\dot{q_1} - \dot{q_0}), \tag{B.17}$$
$$\dot{z}_{cm2} = l_1\cos(q_0)\dot{q_0} + l_{cm2}\cos(q_1 - q_0)(\dot{q_1} - \dot{q_0}), \tag{B.18}$$
$$\dot{x}_{cm3} = -l_1\sin(q_0)\dot{q_0} + l_2\sin(q_1 - q_0)(\dot{q_1} - \dot{q_0}) - l_{cm3}\sin(q_2 - q_1 + q_0)(\dot{q_2} - \dot{q_1} + \dot{q_0}), \tag{B.19}$$
$$\dot{z}_{cm3} = l_1\cos(q_0)\dot{q_0} + l_2\cos(q_1 - q_0)(\dot{q_1} - \dot{q_0}) + l_{cm3}\cos(q_2 - q_1 + q_0)(\dot{q_2} - \dot{q_1} + \dot{q_0}), \tag{B.20}$$
$$\dot{x}_{cm4} = -l_1\sin(q_0)\dot{q_0} + l_2\sin(q_1 - q_0)(\dot{q_1} - \dot{q_0}) - l_3\sin(q_2 - q_1 + q_0)(\dot{q_2} - \dot{q_1} + \dot{q_0}), \tag{B.21}$$
$$\dot{z}_{cm4} = l_1\cos(q_0)\dot{q_0} + l_2\cos(q_1 - q_0)(\dot{q_1} - \dot{q_0}) + l_3\cos(q_2 - q_1 + q_0)(\dot{q_2} - \dot{q_1} + \dot{q_0}). \tag{B.22}$$

B.5 Simulation results.

The following two figures show the simulation results with the estimated masses. The first figure shows the results of bending the trunk while keeping the legs at center position and the second is standing up while keeping the trunk at center position. Both figures show that the gravitational terms are the dominant factors in the required torque. Figure B.1 shows that the torque changes sign which is cause by crossing the straight up position of the trunk, meaning the trunk is leaning backwards. In the last plot Figure B.3 the total intertia contributions are shown and the inertia by drive train and link separately.

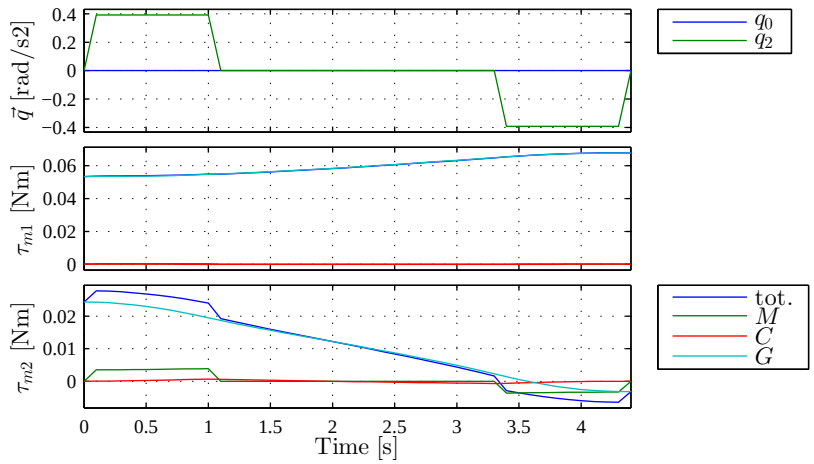


Figure B.1: Simulation results moving the trunk from its lower limit to its upper limit with the legs fixed in the center of its range. The upper plot shows the reference accelerations and the lower plots show the torque contributions of the legs τ_1 and the trunk τ_2.

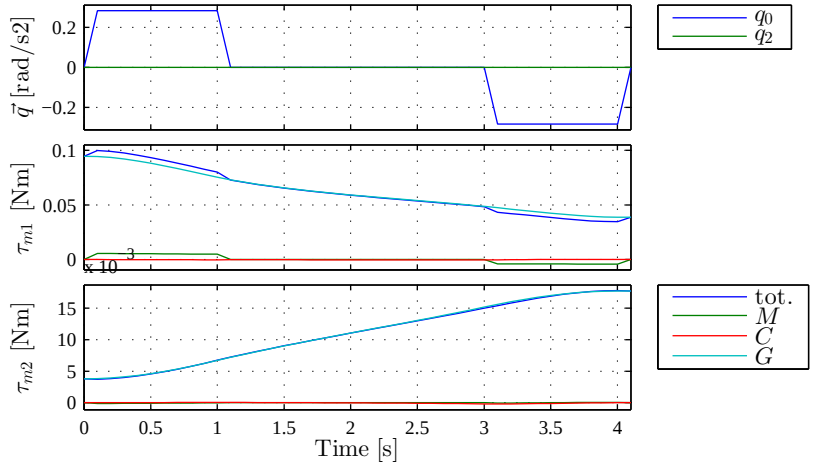


Figure B.2: Simulation results moving the legs from its lower limit to its upper limit with the trunk fixed in the center of its range. The upper plot shows the reference accelerations and the lower plots show the torque contributions of the legs τ_1 and the trunk τ_2.

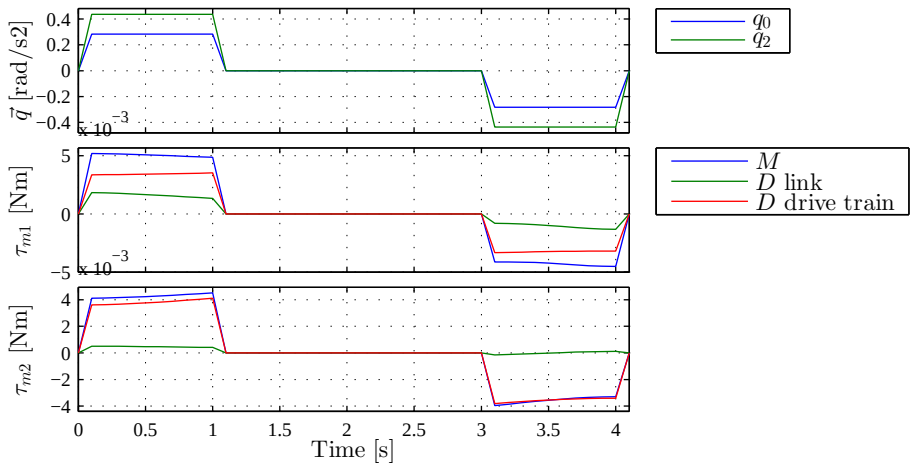


Figure B.3: Simulation results moving the torso from the lower limits to the upper limits. The upper plot shows the reference accelerations and the lower plots show the inertia torque contributions of the legs τ_1 and the trunk τ_2.

B.6 Identification results

In this section some additional measurement results are shown that are used in the identification of the torso mechanism. In Figure B.4 the measurements with different masses are taken in different orientations of the legs to show the offset in the center of mass of the trunk with the trunk link. In Figure B.5 measurement results are shown with different constant reference velocities which show that the change in control output by changing velocities is minimal which means the viscous friction terms can be neglected. In Figure B.6 the tuning results of the acceleration parameter are shown.

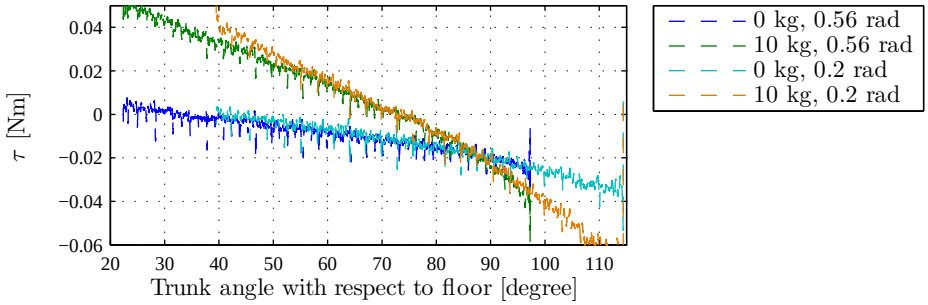


Figure B.4: Required input torque while moving with a constant velocity of $0.004\ \frac{\mathrm{m}}{\mathrm{s}}$ with 0 kg or 10 kg placed in the shoulders and in two different orientations of the legs.

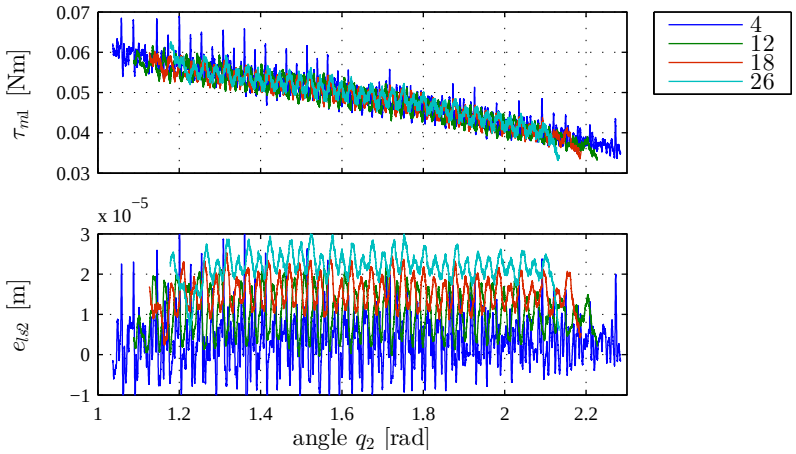


Figure B.5: Required input torque of the trunk while moving with different constant lead screw velocities, legend contains velocities in $\frac{\mathrm{mm}}{\mathrm{s}}$.

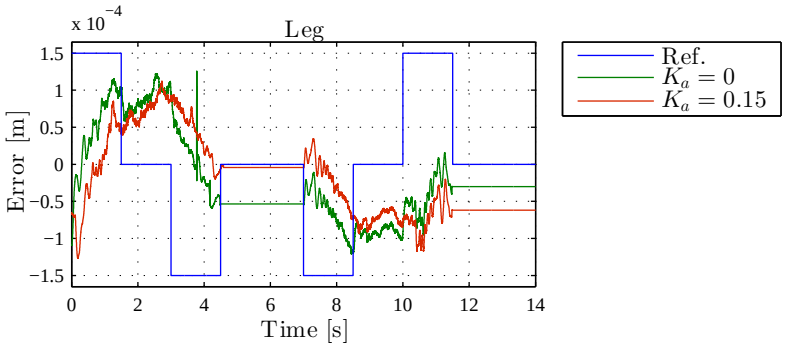


Figure B.6: Tuning results of the acceleration component for the legs.

B.7 Spring load analysis

In this section some additional model results are shown which are used to determine the load of the springs in both trunk and legs.

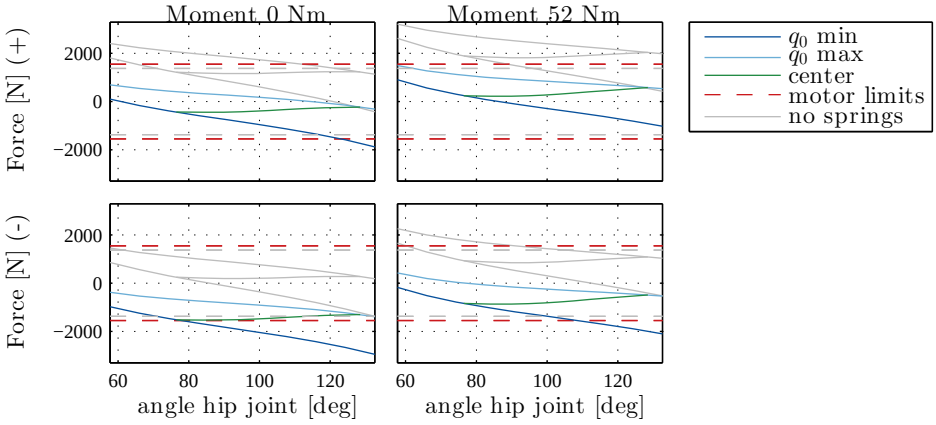


Figure B.7: Forces required to move the trunk with a spring load of 350 N and the drive train limits increased to 2400 N. The transparent grey lines show the previous settings as in Figure 3.15.

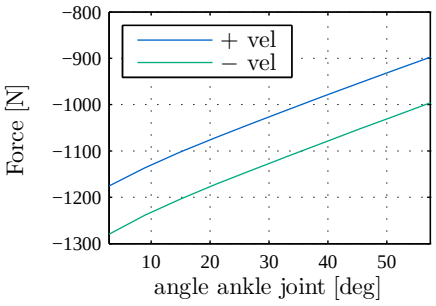


Figure B.8: Effect of two springs (loaded with 500 N each) in the leg on the required drive train force dependent on the angle of the leg. Both in negative (compressed) as positive (extended) motion are shown.

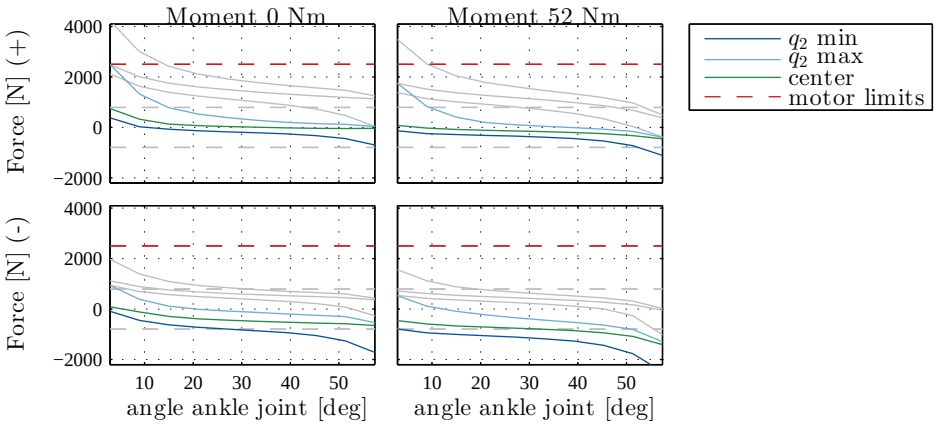


Figure B.9: Forces required to move the legs with spring loads of 575 N and the drive train limits increased to ±2500 N. The transparent grey lines show the settings without springs.

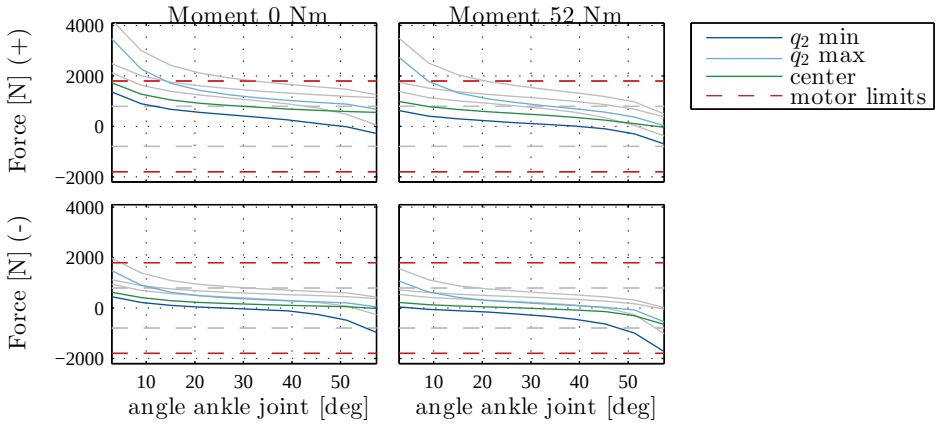


Figure B.10: Forces required to move the Legs with a spring load of 250 N and the drive train limits increased to ±1800 N. The transparent grey lines show the settings without springs.

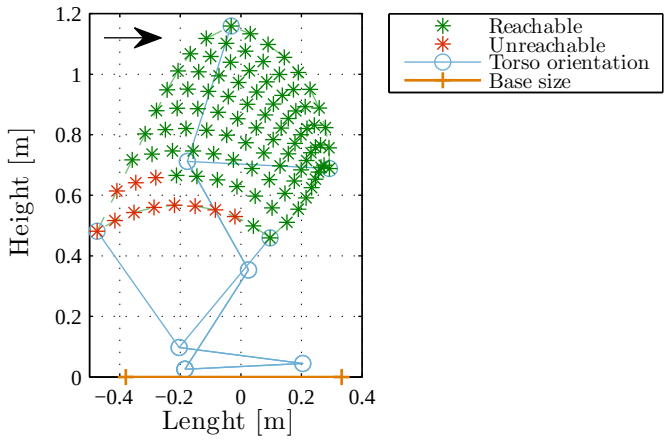


Figure B.11: Reachable space of the shoulders by the leg drive train with a spring load of 250 N and a drive train limit of ±1800 N. The green stars are reachable and the red are not.

B.8 FRF measurements

In this section some additional FRF measurements can be found. In Figure B.12 and B.13 the bode plots and coherence of the leg and trunk respectively can be found measured at different orientations of the torso. The figures clearly shows there is no significant difference between the different orientations. In Figure B.14 the FRF with and without an additional mass placed in the shoulders can be found. As for the different positions the additional mass has no significant effect on changing the response.

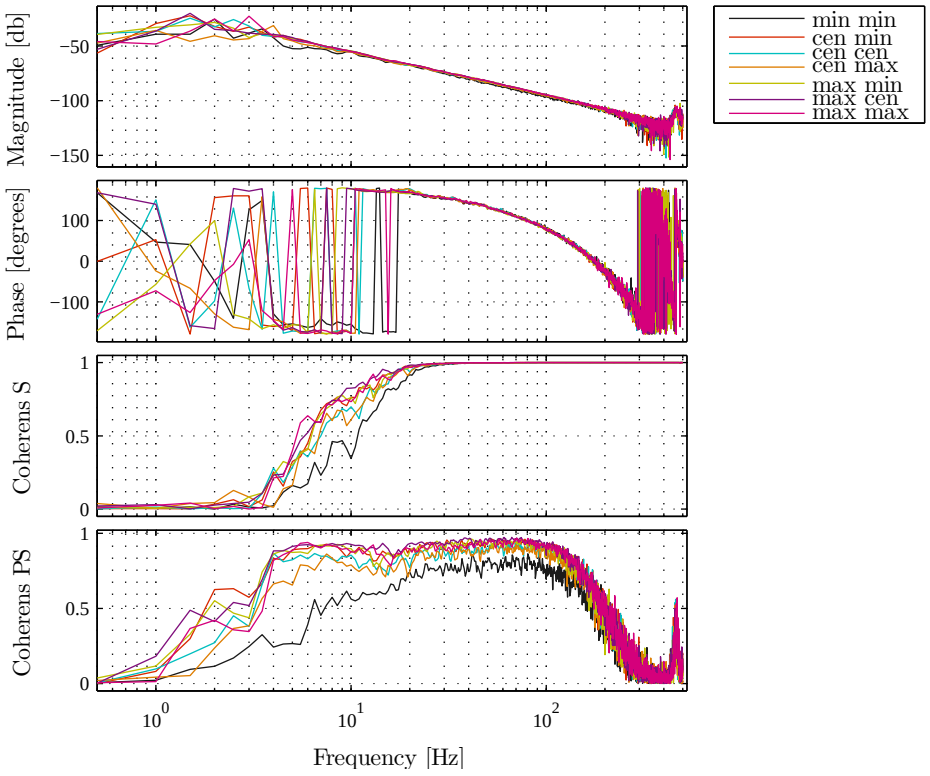


Figure B.12: Results leg FRF measurement at different orientations of the torso. The four plots show the magnitude, phase, coherence of the sensitivity $\mathcal{S}$ and process sensitivity $\mathcal{PS}$ from top down. The different lines show measurements at different orientations of the legs and trunk, where 'max min' means the legs in the maximum position and the trunk at the minimum position ('cen' means center of range). All measurements are taken with a reference sine for the legs with a frequency 10 Hz and amplitude of 0.01 m lead screw length.

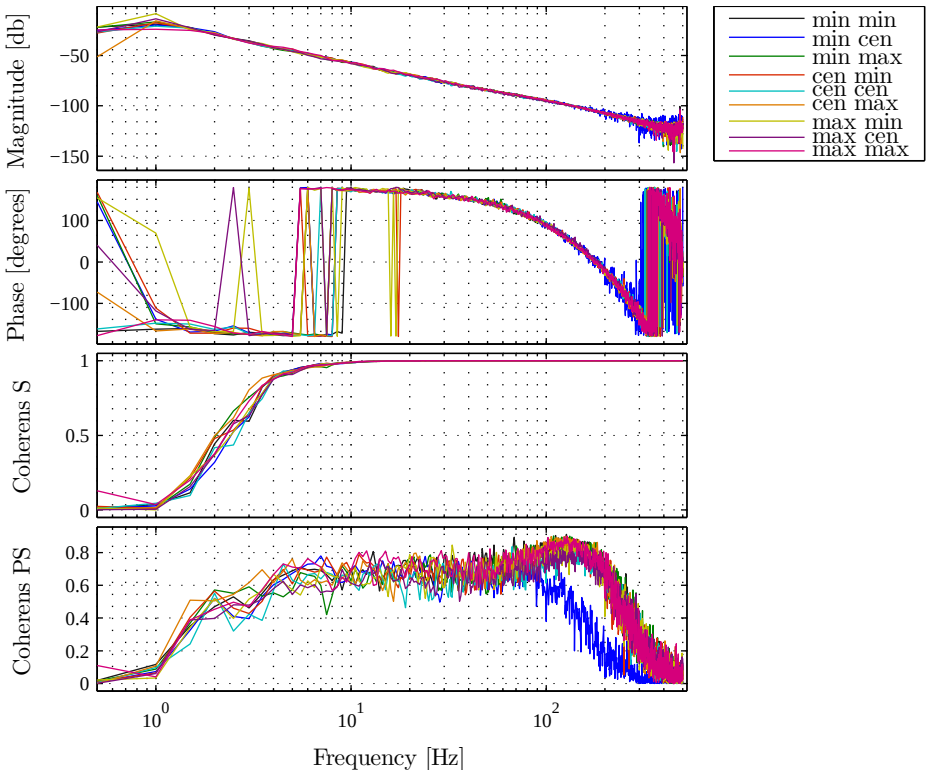


Figure B.13: Results trunk FRF measurement at different orientations of the torso. The four plots show the magnitude, phase, coherence of the sensitivity and process sensitivity from top down. The different lines show measurements at different orientations of the legs and trunk, where 'max min' means the legs in the maximum position and the trunk at the minimum position ('cen' means center of range). All measurements are taken with a reference sine for the trunk with a frequency 10 Hz and amplitude of 0.01 m lead screw length.

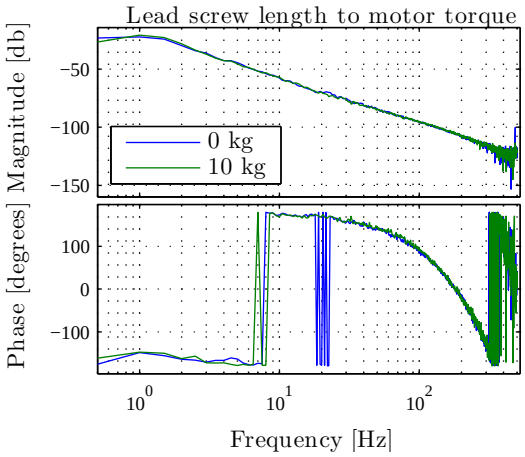


Figure B.14: Trunk transfer from lead screw to hip angle without and with a mass of 10 kg placed in the shoulder. Measured with the legs and trunk positioned at the center of their range. To overcome static friction a sine reference is applied on the hip joint with a frequency of 10 Hz and amplitude of 0.01 m lead screw length.

B.9 Controller comparison

In Figure B.15 the results of different controllers are compared.

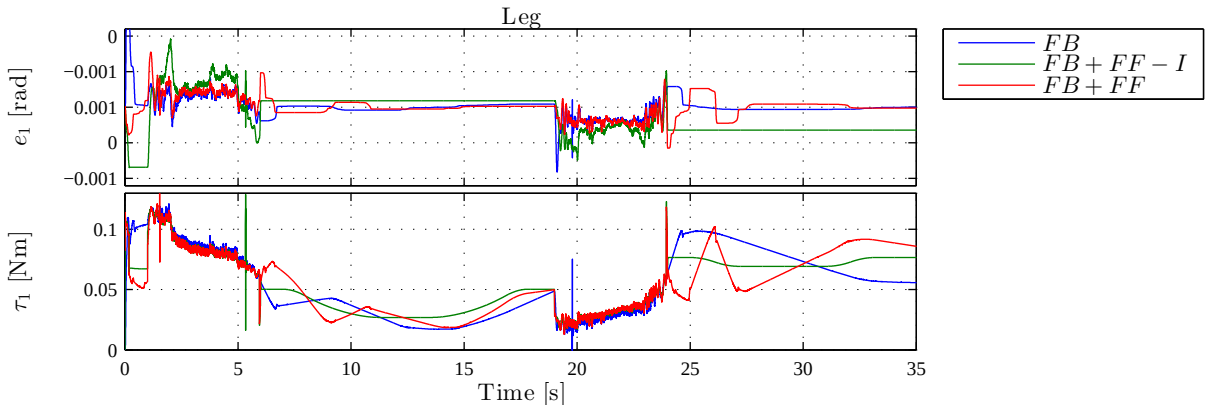


Figure B.15: Measurement results of the stand up and bend trajectory for the legs with different controllers, where FB is the proposed feedback controller, FF the feed forward controller and $-I$ means feedback without the integrator. The plots show the joint error and controller output torque.