Analysis and Control of a Simple Nonlinear Limb Model

by

Dávid Csercsik

Submitted to the department of Control Engineering and Information Technology in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

at the

Budapest University of Technology and Economics

Supervisor:
Prof. Katalin Hangos
Systems and Control Laboratory
Computer and Automation
Research Institute Budapest

Department Consultant:
Dr. István Harmati
Department of Control Engineering and Information Technology
Faculty of Control Engineering and Informatics
Budapest University of Technology and Economics

BME
2005
Contents

1 Introduction 5
  1.1 Literature review on human locomotion analysis and control . 5
  1.2 Notation list . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
  1.3 Structure of the diploma work . . . . . . . . . . . . . . . . 7

2 Construction of a simple limb model in state-space 8
  2.1 Limb and muscle modelling . . . . . . . . . . . . . . . . . . 8
  2.2 System description . . . . . . . . . . . . . . . . . . . . . . . 8
    2.2.1 Modelling assumptions . . . . . . . . . . . . . . . . . . 9
  2.3 Muscle properties . . . . . . . . . . . . . . . . . . . . . . . . 10
    2.3.1 Force-length characteristics \( F_L(l_C) \) . . . . . . . . 10
    2.3.2 Force-contraction velocity characteristics \( F_v(v_C) \) . 11
    2.3.3 Passive force \( F_P \) . . . . . . . . . . . . . . . . . . . 12
    2.3.4 Muscle activation \( q \) . . . . . . . . . . . . . . . . . . 13
    2.3.5 Muscle force \( F_C \) . . . . . . . . . . . . . . . . . . . 13
  2.4 Tendon properties . . . . . . . . . . . . . . . . . . . . . . . . 14
    2.4.1 The tendon’s dynamical equations . . . . . . . . . . . . 14
  2.5 Dynamic equations at the limb level . . . . . . . . . . . . . . 14
    2.5.1 Forces . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
    2.5.2 Joint torques . . . . . . . . . . . . . . . . . . . . . . . 15
    2.5.3 Movement . . . . . . . . . . . . . . . . . . . . . . . . . 15
  2.6 State-space model equations and its variables . . . . . . . . . 17
    2.6.1 Variables in the state-space model . . . . . . . . . . . . 17
    2.6.2 State-space equations . . . . . . . . . . . . . . . . . . . 17
    2.6.3 The domain of the model and the model parameters . . . 19
  2.7 Model verification . . . . . . . . . . . . . . . . . . . . . . . . 21
  2.8 A biological extension: A simple Gamma-loop model . . . . . 24
    2.8.1 The simplified model of the Gamma-loop . . . . . . . . 24
    2.8.2 Simulation results . . . . . . . . . . . . . . . . . . . . . 26

3 Model analysis 28
  3.1 Linear analysis . . . . . . . . . . . . . . . . . . . . . . . . . 30
    3.1.1 Linearization around a steady-state point . . . . . . . . 30
    3.1.2 Analysis of the linearized model at the first steady-state point . . . . . . . . . . 31
    3.1.3 Analysis of the linearized model at the second steady-state point . . . . . . . . . . 33
  3.2 Relative degree . . . . . . . . . . . . . . . . . . . . . . . . . 34
    3.2.1 The structure graph . . . . . . . . . . . . . . . . . . . . 35
3.3 Controllability ................................................... 36
  3.3.1 Algorithm for constructing the controllability distribu-
         tion ................................................... 36
  3.3.2 Structural controllability and observability .......... 37
  3.3.3 Structural controllability analysis of the model ...... 37
3.4 Observability .................................................. 37
  3.4.1 Algorithm for constructing the observability
         co-distribution ....................................... 37
  3.4.2 Structural observability analysis of the model ..... 38
3.5 Stability ...................................................... 39
  3.5.1 Lyapunov-stability .................................... 39
  3.5.2 Strong asymptotic stability .......................... 39
  3.5.3 Lyapunov’s first theorem ............................... 39
  3.5.4 Estimating the stability region in the neighbor-
         hood of the steady-state point with quadratic Lyapunov-
         function candidate ................................. 40
3.6 Zero dynamics .................................................. 42
  3.6.1 Zero dynamics state-space equations ................ 44
  3.6.2 Computing the zeroing input .......................... 45
  3.6.3 Zero-dynamics analysis results ....................... 45
4 Controller design ............................................. 46
  4.1 Performance specification ................................. 46
    4.1.1 Control aims ...................................... 46
    4.1.2 Control domain ................................... 46
    4.1.3 Disturbances ...................................... 46
  4.2 Linear SISO LQ regulator ................................. 48
    4.2.1 Simulation results of the SISO LQ controller .... 50
  4.3 Linear MISO LQ regulator ................................. 52
    4.3.1 Simulation results of the MISO LQ controller with
           disturbance ....................................... 52
    4.3.2 Trajectory following results of the MISO LQ controller
           with disturbance ................................... 54
    4.3.3 Regulation results of the MISO LQ controller with dis-
           turbance ........................................... 55
  4.4 Input-Output linearization ............................... 56
    4.4.1 Simulation results of the IOL-PP controller without
           disturbance ....................................... 58
    4.4.2 Trajectory following results of the IOL-PP controller
           with disturbance ................................... 59
4.4.3 Regulation results of the IOL-PP controller with disturbance .......................... 60
4.5 Fuzzy Inference System .............................................. 61
  4.5.1 Basic concepts in fuzzy theory ................................ 61
  4.5.2 The applied fuzzy controller ................................. 66
  4.5.3 Simulation results of the fuzzy controller without disturbance .......................... 69
  4.5.4 Trajectory following results of the Fuzzy controller with disturbance ...................... 70
  4.5.5 Regulation results of the Fuzzy controller with disturbance .................................. 71
4.6 Comparison of the different control methods ................................................. 72
  4.6.1 Comparison of the results without disturbance ............................................... 72
  4.6.2 Comparison of the results with disturbance ................................................ 73
  4.6.3 Comparison results: a discussion ......................................................... 74
5 Conclusions and future work ............................................. 75
6 Acknowledgements ....................................................... 76
7 Appendix ................................................................. 77
  7.1 The gamma-loop ..................................................... 77
    7.1.1 Muscle spindles ............................................... 77
    7.1.2 The servomechanical Gamma-loop ........................................ 78
Abstract

Motivation and Aim:
Even the simplest limb model exhibits strongly nonlinear dynamic behavior that calls for applying the results of nonlinear systems and control theory. The analysis and control of limb models are important in the fields of designing and controlling artificial limbs, muscle prosthesis and in neuro-physiological investigations. The aim of this study is to investigate the possibility to applying input-output linearization [11] for nonlinear control of a simple limb model.

Material and Methods:
A nonlinear input-affine state-space model has been developed for a simple one-joint system with a flexor and an extensor muscle (see figure 1) which is suitable for nonlinear systems analysis and control. The model takes the nonlinear properties of the force-length relation and the force-contraction velocity relation into account.

Exerted forces depend linearly on the activation state of muscles, and a viscoelastic tendon is considered following the principles in [28], [24] and [22]. This model has been extended with a simple model of the gamma-loop mechanism, but only the non-extended model is used for the control studies. The inputs of the model are the normalized activation signal of muscles, the output is the joint angle, and the number of state variables is 8.

As preliminary model analysis we performed stability, controllability and observability analysis of the linearized model around steady-state points. Moreover, the relative degree of the model and the stability of its zero-dynamics [3] were also determined. Both regulating and servo controllers were designed for the simple limb model and were compared to the standard reference case being an LQ-controller designed for the locally linearized system. A pole-placement control was designed for the input-output linearized system, and also a fuzzy controller was designed.

Result:
The model was verified against engineering intuition and proved to be suitable for controller design purposes. The model analysis showed that the nonlinear limb model was controllable and was in the edge of stability because of the Hamiltonian properties of the model. The relative degree of the model is 3 for both of the inputs with a stable zero dynamics. Therefore input-output linearization was applicable and a 3rd order linear system was obtained in the new coordinates. A simple pole-placement controller was designed for this input-output linearized model. Both the pole-placement, the fuzzy, and the reference LQ-controller were suitable for control purposes but the fuzzy and the LQ-controller were more sensitive to the disturbances.
1 Introduction

"Human locomotion is a complex movement of a body. For a successful movement, interactions among muscular-skeletal system and central nervous system (CNS) are needed. The CNS controls the movement by sending activation signals to the correct skeletal muscle at correct moment. The effect of this signal is that the muscle initiates a movement by exerting force to a body segment. Even in the case of a simple movement the contribution of a large number of muscles of different size and shape is necessary. A common movement such as locomotion require more muscles. With a feedback system that uses different kind of receptors, the CNS controls the movement. In healthy humans all these complex coordinated actions lead unconsciously to a smooth movement." [8]

The study of human locomotion has gained more attention recently with the development of analytic and computational tools with which to examine it. A much researched subject within the field today is the effort to model human motor control systems using control theoretic methods. Analytic, computational, and experimental studies of locomotion can produce models that provide further insight into the design and functioning of human motor control systems, as well as provide directions for research into diagnostics and therapies for muscle- and nerve-related disorders affecting these systems.

The aim of this work is to investigate the difficulties related to the control of a such strongly nonlinear system, as a human limb, and to design a controller for a simple nonlinear limb model. For this aim we need a model suitable for the mathematical tools of linear and nonlinear control theory and system analysis - a nonlinear state space model.

1.1 Literature review on human locomotion analysis and control

Several control models exist that take different important aspects of human locomotion control into account, in various cases of control tasks. The most relevant results of recent investigations on locomotion analysis and control are listed and summarized in the following list:

- A multi level control model including timing and learning was developed by Levine and Loeb [18].

- A study proposed by Levine and Zajac [20] has shown, that in the case of the pedaling problem, the control to achieve maximal acceleration for a simple skeletal system is bang-bang.
• A further research of acceptable controllers (incl. input-output linearization) for the cycling problem was proposed by Abbot in the degree thesis [1].

• The study of Dingwell [4] focuses on the problem of determining if techniques for analyzing continuous nonlinear and chaotic systems series data could be applied to kinematic data collected during continuous overground walking.

• The article of Kaplan [12] provides an optimal control method for muscle activity in the case of the pedalling problem using non-derivative quasi-Newton methods and numerical gradient methods.

• The study of Verschueren [25] addressed the involvement of proprioceptive input of the muscle spindles in the spatiotemporal control of human locomotion.

1.2 Notation list

The most important notations that are used during model development, and analysis are summarized in the table below.

<table>
<thead>
<tr>
<th>Notation</th>
<th>means</th>
<th>Index</th>
<th>refers to</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F$</td>
<td>force</td>
<td>$CE$</td>
<td>contracting element - muscle</td>
</tr>
<tr>
<td>$g$</td>
<td>gravity</td>
<td>$act$</td>
<td>activation</td>
</tr>
<tr>
<td>$l$</td>
<td>length</td>
<td>$CE + T$</td>
<td>muscle-tendon system</td>
</tr>
<tr>
<td>$q$</td>
<td>activity state</td>
<td>$PE$</td>
<td>passive force of muscle</td>
</tr>
<tr>
<td>$v$</td>
<td>velocity</td>
<td>$lig$</td>
<td>ligament</td>
</tr>
<tr>
<td>$u$</td>
<td>activation signal</td>
<td>$T$</td>
<td>tendon</td>
</tr>
<tr>
<td>$x$</td>
<td>state-space variable</td>
<td>$s$</td>
<td>muscle spindle</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>joint angle</td>
<td>1</td>
<td>flexor</td>
</tr>
<tr>
<td>$\omega$</td>
<td>joint angle velocity</td>
<td>2</td>
<td>extensor</td>
</tr>
<tr>
<td>$\Theta$</td>
<td>moment of inertia</td>
<td>$opt$</td>
<td>optimal</td>
</tr>
<tr>
<td>$\tau$</td>
<td>time</td>
<td>$c$</td>
<td>controllability</td>
</tr>
<tr>
<td>$\xi$</td>
<td>position angle</td>
<td>$o$</td>
<td>observability</td>
</tr>
<tr>
<td>$L$</td>
<td>Lie derivative</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$V$</td>
<td>Lyapunov function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$J$</td>
<td>Jacobian matrix</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Hungarian scientific background of locomotion analysis:

Modelling of limb movement patterns based on neuronal activity has been developed in Hungary for some years. Differential neuro-muscular-skeletal structures have been studied by Laczko et al. [15, 14, 13] using mathematical models and computer simulation.

A neuro-mechanical model was developed by Fazekas in [6, 7]. It takes into account the one and two-joint muscles. It handles the force-length and force-frequency relationship, passive force, geometric and inertial properties of the limb, the maximal isometric forces and the gravitational effect.

1.3 Structure of the diploma work

The description of the construction method and the derived equations of the simple limb model can be found in chapter 2. Chapter 3 describes the analysis results of the simple limb model. In chapter 4 the description, and results of the applied control methods can be found. Conclusions of the work and possible future works are summarized in chapter 5.
2 Construction of a simple limb model in state-space

2.1 Limb and muscle modelling

Our aim is to create a model of a simple one-joint system with a flexor and an extensor muscle (see in figure 1) which is suitable for nonlinear systems analysis and control.

The models of dynamics of a one-joint system with two muscles contain the equations of multi-rigid-limb system and the equations of the dynamics of muscle contraction. The crucial component of this system is the model that generates the exerting muscle forces.

Muscle modelling:

Different muscle models exist that focus different important aspects of muscle functioning. In some of the models individual muscle characteristics are used, and some of the models deal with generalized muscle characteristics that are valid for all muscle. But most of the existing models deal only with a partial functioning of the muscle such as a force depending on the muscle length or muscle geometry or firing rates of nerves etc.

There are some models that deal with the movement pattern generation according to the force exertion and neuronal input. For example Cheng [3] created this kind of model.

A simple nonlinear limb model:

In this chapter the model equations are derived from first engineering (medical) principles based on the detailed model of Fazekas [8], Zajac [28] and Van Soest [21]. Therefore the equations are transformed into a state-space model form.

2.2 System description

A simple two dimensional limb is considered that consists of two muscles and one joint. The schematic picture of the limb is shown in figure 1. The input of the model consists of the activation signal or stimulating signal of each muscles as a function of time. Activation signals are normalized, i.e. their values are between zero and one.

In general case, the output of the model is the joint angle.
2.2.1 Modelling assumptions

- Dimension: The model is two dimensional.
- Structure: We model 2 muscles, one joint.
- Segments: The bones are totally rigid.
- Gravity: Gravity appears in the direction -y only.
- Geometry: We do not model the geometry of bones and the muscle.
- Muscle Properties: The model takes into account only the following properties of the muscle: Force-length dependency, Force-contraction velocity dependency, Passive Force. The characteristics are the same by the flexor, and the extensor muscle.
- Moment arms: Moment arms are constants.
- Aponeurosis: There is not aponeurosis.
- Pennate effect: We do not deal with the pennate effect.
- Tendon: We suppose viscoelastic tendon.
• Fatigue: In general case, we do not model fatigue, potentialization and short time histories in the muscles.
• Activation dynamics: We use first-order activation dynamics model.
• Viscosity: We do not model viscosity in the muscle.

2.3 Muscle properties

To construct a state-space limb model, at first we need to examine the properties of its components (muscles, tendon, etc). We start with the mathematical equations describing the behaviour of the muscles.

2.3.1 Force-length characteristics \((FL(l_{CE}))\)

The equation describes the quotient of the actual force, and the maximal force of the muscle, at given length \((l_{CE})\). We use a parabolic function:

\[
FL(l_{CE}) = \frac{c}{\omega^2} \left( \frac{l_{CE}}{l_{opt}^{CE}} \right)^2 - 2c \left( \frac{l_{CE}}{l_{opt}^{CE}} \right) + c + 1
\] (1)

where \(l_{CE} \text{ [m]}\) is the actual length of the muscle, \(l_{opt}^{CE} \text{ [m]}\) is the optimal length of the muscle, at which it can produce it's maximal force. The coefficient \(c\) is defined by: \(c = -\frac{1}{\omega^2}\) where \(\omega\) is a constant, describing the range of length, where the muscle is able to work. As Soest [22], we define \(\omega = 0.56\).

The equation (1) is only valid in the range of \(1 - \omega \leq \left( \frac{l_{CE}}{l_{opt}^{CE}} \right) \leq 1 + \omega\)
outside this range \(FL\) is 0. We can see the characteristics in figure 2.

![Figure 2: FL(l_{CE})](image)
2.3.2 Force-contraction velocity characteristics ($F_v(v_{CE})$)

The force produced by a muscle depends on its actual contraction velocity. We use a similar function to Hill’s equation [10], extended by Van Soest [21] that is:

\[
F_v(v_{CE}) = \begin{cases} 
\frac{(1+A_{ref})B_{ref}}{v_{CE}^{ref}} - A_{ref} & \text{if } v_{CE} > 0 \\
\frac{c_1}{v_{CE}^{ref}} - c_2 & \text{otherwise}
\end{cases}
\]

(2)

where

\[
c_1 = -\frac{B_{ref}(1-F_{asy})^2}{2+2A_{ref}}, \quad c_2 = -F_{asy}, \quad c_3 = \frac{B_{ref}(1-F_{asy})}{2+2A_{ref}}
\]

Furthermore, $v_{CE}$ is the actual contraction velocity of the muscle (positive, if the muscle is contracting, and negative if the muscle is extracting), $A_{ref}$ and $B_{ref}$ are constants, $F_{asy}$ is a constant, which shows at eccentric contraction the quotient of the actual available force, and the maximal isometric force.

This function is not continuously differentiable, so to avoid the problems, we use an approximating smooth function that fits to the requirements of nonlinear analysis (the parameters of the function were found in experimental ways):

\[
F_v(v_{CE}) = -\frac{3}{2} \arctan(9/5 v_{CE} - \frac{9}{25}) \pi^{-1} + \frac{167}{200}
\]

(3)

The original and the approximated functions are shown in figure 3.

Figure 3: $F_v(v_{CE})$ and its approximation
2.3.3 Passive force \((F_{PE})\)

For describing the passive force \(F_{PE}\) as a function of \(l_{CE}\) we use modified sigmoid functions for both the flexor and extensor muscles:

\[
F_{PE} = (1 - \text{sigm}[7(-l_{CE} + 0.3333))^{10})F_{PE}^{max}
\]

(4)

where the function identifier ‘sigm’ stands for the function:

\[
y = \text{sigm}[x] = \frac{1}{1 + e^{-100x}}
\]

with \(F_{PE}^{max}\) being the maximal passive force. We denote the flexor muscle’s passive force by \(F_{PE1}\) and the extensor muscle’s passive force by \(F_{PE2}\). 

![Figure 4: The function \(F_{PE}\)](image)
2.3.4 Muscle activation (q)

The differential equation of the muscle shows the connection between \( q(t) \), the activation state of the muscle and the activation signal \( u(t) \). With \( u(t) \in [0, 1] \) the equation taken from Zajac [28] is:

\[
\frac{dq}{dt} = - \left( \frac{1}{\tau_{act}} (\beta + [1 - \beta]u(t)) \right) q + \frac{1}{\tau_{act}} u(t) 
\]

(5)

where \( \tau_{act} [s] \) is the activation time, showing how quick the muscle reacts on the external activation signal coming from the nervous system. \( \beta \) is a constant, describing the correlation between the decrease of the activation state and the external activation signal. If \( \beta = 1 \) then the external activation signal does not affect the decrease of the activation state, if \( \beta = 0 \) then it strongly affects it. \( q_1(t) \) denotes the activation state of the flexor muscle, and \( q_2(t) \) denotes the activation state of the extensor muscle.

2.3.5 Muscle force (\( F_{CE} \))

We can compute the force of the muscle with the characteristics above in the following form:

\[
F_{CE} = F_{max}F_L(l_{CE})F_v(v_{CE})q + F_{PE}
\]

(6)

where \( F_{max} \) is the maximal force of the muscle. We use \( F_{CE1} \) for the flexor muscle’s force and \( F_{CE2} \) for the extensor muscle’s force.

\[
F_{CE1} = F_{max}F_L(l_{CE1})F_v(v_{CE1})q_1 + F_{PE1}
\]
\[
F_{CE2} = F_{max}F_L(l_{CE2})F_v(v_{CE2})q_2 + F_{PE2}
\]

(7)

where \( q_1 \) is the flexor muscle’s activation state, \( q_2 \) is the extensor muscle’s activation state, \( l_{CE1} [m] \) is the length of the flexor muscle, \( l_{CE2} [m] \) is the length of the extensor muscle, \( v_{CE1} [m/s] \) is the contraction velocity of the flexor muscle and \( v_{CE2} [m/s] \) is the contraction velocity of the extensor muscle.
2.4 Tendon properties

2.4.1 The tendon’s dynamical equations

We need tendons to transport the force from the muscles to the bones. We describe the tendon’s behaviour with the following equations:

\[
\frac{\partial l_T}{\partial t} = v_T \\
\frac{\partial v_T}{\partial t} = -\frac{k_T (l_T - l_T^{\text{slack}}) + s_T v_T - F_T}{z_T}
\] (8)

where the first equation is valid only, if \( l_T > l_T^{\text{slack}} \). \( l_T \) [m] is the actual length of the tendon, \( l_T^{\text{slack}} \) [m] is the tendon’s length in the case, when forces do not appear. \( v_T \) [m/s] is the extracting velocity of the tendon (it is positive, if the tendon extracts), \( k_T \) [N/m] is the elasticity constant of the tendon, \( s_T \) [Ns/m] and \( z_T \) \([\text{Ns}^2/\text{m}]\) are constants referring to the dynamics of the tendon, \( F_T \) [N] is the force acting on the tendon.

2.5 Dynamic equations at the limb level

2.5.1 Forces

Besides the force of the muscle, we use a sigmoid function for modelling the forces of ligaments and bones in the end-positions of the joint. These functions are sigmoid in both of the angle of the joint and in the angle velocity to take unelastic properties (the properties making the joint available to dissipate motion energy) of the joint into account. The equation for describing the forces of the ligaments is as follows:

\[
F_{\text{lig}} = F_{\text{lig}}^{\text{max}} \text{sign}(0.3(\alpha - \rho))\text{sign}(0.5\omega)
\] (9)

where \( \alpha \) [rad] is the external joint angle, \( F_{\text{lig}}^{\text{max}} \) [N] is the maximum force of the ligaments and bones, \( \rho \) [rad] is a constant, describing at which angle the force appears and \( \omega \) [rad/s] is the joint angle velocity. We use \( F_{\text{lig}1} \) for the force, which appears at 0 angle, when the limb can not be more extended (at the maximum length of the flexor muscle), so it wants to flex the limb, and \( F_{\text{lig}2} \) for the extensor ligamental force.

The flexing forces can be computed as:

\[
F_{\text{flexor}} = F_{CE1} + F_{\text{lig}1}
\] (10)

The extending forces can be computed as:

\[
F_{\text{extensor}} = F_{CE2} + F_{\text{lig}2}
\] (11)
We suppose, for the sake of simplicity, that the forces of the ligaments and bones appear at the same point as the forces of the muscle.

Furthermore, apart from the most extended and most flexed states, we can neglect the forces of ligaments (and in some cases the passive force too), to further simplify the model for symbolic analysis.

In this model the force of the tendon is of the same size as the force of the muscle and ligaments, it just appears in the opposite direction. For example: $F_T = -F_{\text{flexor}}$ where $F_T [N]$ is the tendon’s force.

### 2.5.2 Joint torques

The movement is determined not directly by the forces conveyed by the tendon, but by their torques. For example the flexor tendon’s torque can be written as:

$$M_1 = -F_T d = F_{\text{flexor}} d$$

where $d [m]$ is the moment arm, the distance between the axis of the joint, and the point, where the forces appear.

The resultant joint torque can be computed as:

$$M = M_1 - M_2$$

### 2.5.3 Movement

Now we know how the forces are generated, and transported to the bones. Let us then examine how the limb moves. We use the form of Newton’s II law applied to rotation movement as follows:

$$\frac{d\alpha}{dt} = \omega$$

$$\frac{d\omega}{dt} = \frac{1}{\Theta + ml_{\text{COM}}^2} \left( M + ml_{\text{COM}} \cos(\alpha + \xi)g_y \right)$$

where $\alpha [\text{rad}]$ is the joint angle, $\xi [\text{rad}]$ is the angle between the global coordinate-system’s $x$ axis, and the not-moving upper segment of the limb (in our model $\xi$ is always equal to $-\pi/2$), $\omega [\text{rad/s}]$ is the angle velocity, $\Theta [kgm^2]$ is the moment of inertia defined to the mass-centre point of the bone, $m [\text{kg}]$ is the mass of the moving limb part, $l_{\text{COM}} [\text{m}]$ is the distance between the moving limb part’s center of mass point and the joint axis, $M [\text{Nm}]$ is the resulting joint torque, and $g = [g_x, g_y] [m/s^2]$ is the vector of gravitational acceleration.
What’s left, is to define the connection between the joint angle, and the muscle length. For the limb the actuator system does not matter, so at the level of the segments we talk about muscle-tendon complexes. So, what we can directly compute from the joint angle, is the length of these muscle-tendon complexes. We will denote these muscle-tendon complexes with $CE + T$ in the subscript. We can compute the length of the muscle-tendon complexes approximately with the following equations:

In the case of flexor muscle:

$$l_{CE+T} = \sqrt{d^2 + (d^{prox})^2 - 2dd^{prox}\sin\alpha(t)} + \sqrt{d^2 + (d^{dist})^2 - 2dd^{dist}\sin\alpha(t)}$$ \hspace{1cm} (15)

In the case of extensor muscle:

$$l_{CE+T} = \sqrt{d^2 + (d^{prox})^2 + 2dd^{prox}\sin\alpha(t)} + \sqrt{d^2 + (d^{dist})^2 + 2dd^{dist}\sin\alpha(t)}$$ \hspace{1cm} (16)

where $l_{CE+T}$ [m] is the length of the muscle-tendon complex, $d$ [m] is the moment arm, $d^{prox}$ [m] is the distance between the joint, and the origin of the muscle, $d^{dist}$ [m] is the distance between the joint, and the insertion of the muscle, $\alpha$ [rad] is the joint angle. The extracting velocity of the muscle-tendon complex can be computed as:

$$v_{CE+T} = \frac{dl_{CE+T}}{dt}$$ \hspace{1cm} (17)

From these values we can compute the length, and contraction velocity of the muscles, if we know the length and extraction velocity of the tendons. Both for flexor and extensor muscles:

$$l_{CE}(t) = l_{CE+T}(t) - l_T(t)$$

$$v_{CE}(t) = v_{CE+T}(t) + v_T(t)$$ \hspace{1cm} (18)
2.6 State-space model equations and its variables

2.6.1 Variables in the state-space model

We have the following state variables:

- Joint angle: $\alpha$ [rad]
- Joint angle velocity $\omega$ [rad/s]
- Muscle activation state (for flexor and extensor muscle): $q_1, q_2$
- Tendon length (of the flexor and the extensor muscle): $l_{T1}, l_{T2}$ [m]
- Tendon extracting velocity (for flexor and extensor muscle): $v_{T1}, v_{T2}$ [m/s]

So we get: $x = [q_1 \ q_2 \ \alpha \ \omega \ l_{T1} \ l_{T2} \ v_{T1} \ v_{T2}]^T$

We have the following input variables:

- Flexor muscle’s activation signal: $u_1(t)$
- Extensor muscle’s activation signal: $u_2(t)$

The output of the model is the joint angle: $y = \alpha = x_3$

2.6.2 State-space equations

\[
\frac{dq_1}{dt} = - \left( \frac{1}{\tau_{act}}(\beta + [1 - \beta]u_1(t)) \right) q_1 + \frac{1}{\tau_{act}}u_1(t)
\]

\[
\frac{dq_2}{dt} = - \left( \frac{1}{\tau_{act}}(\beta + [1 - \beta]u_2(t)) \right) q_2 + \frac{1}{\tau_{act}}u_2(t)
\]

\[
\frac{d\alpha}{dt} = \omega
\]

\[
\frac{d\omega}{dt} = \frac{1}{\Theta + ml_{COM}^2} (M(q_1, q_2, \alpha, \omega, l_{T1}, l_{T2}, v_{T1}, v_{T2}) + ml_{COM}\cos(\alpha + \xi)g_y)
\]

\[
\frac{dl_{T1}}{dt} = v_{T1}
\]

\[
\frac{dl_{T2}}{dt} = v_{T2}
\]

\[
\frac{dv_{T1}}{dt} = - \frac{k_t(l_{T1} - l_{slack}) + s_Tl_{T1} - F_{flexor}(q_1, \alpha, \omega, l_{T1}, v_{T1})}{z_T}
\]

\[
\frac{dv_{T2}}{dt} = - \frac{k_t(l_{T2} - l_{slack}) + s_Tv_{T2} - F_{extensor}(q_2, \alpha, \omega, l_{T2}, v_{T2})}{z_T}
\]  

(19)
With the notation of $x_1, x_2, ..., x_8$ for state space variables the equations are as follows:

\[
\frac{dx_1}{dt} = -\left( \frac{1}{\tau_{act}}(\beta + [1 - \beta]u_1(t)) \right) x_1 + \frac{1}{\tau_{act}} u_1(t)
\]

\[
\frac{dx_2}{dt} = -\left( \frac{1}{\tau_{act}}(\beta + [1 - \beta]u_2(t)) \right) x_2 + \frac{1}{\tau_{act}} u_2(t)
\]

\[
\frac{dx_3}{dt} = x_4
\]

\[
\frac{dx_4}{dt} = \frac{1}{\Theta + ml^2_{COM}} (M(x_1, x_2, x_3, x_4, x_5, x_7, x_8) + ml_{COM} \cos(x_3 + \xi) g_y)
\]

\[
\frac{dx_5}{dt} = x_7
\]

\[
\frac{dx_6}{dt} = x_8
\]

\[
\frac{dx_7}{dt} = -k_1(x_5 - l_{flex}^k) + s_T x_7 - F_{flexor}(x_1, x_3, x_4, x_5, x_7)
\]

\[
\frac{dx_8}{dt} = -k_1(x_6 - l_{flex}^k) + s_T x_8 - F_{flexor}(x_2, x_3, x_4, x_6, x_8)
\]

\[y = x_3\]
2.6.3 The domain of the model and the model parameters

State and input variables:

<table>
<thead>
<tr>
<th>symbol</th>
<th>description</th>
<th>dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1$</td>
<td>the flexor muscle’s activation state</td>
<td>-</td>
</tr>
<tr>
<td>$q_2$</td>
<td>the extensor muscle’s activation state</td>
<td>-</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>joint angle</td>
<td>[rad]</td>
</tr>
<tr>
<td>$\omega$</td>
<td>joint angle velocity</td>
<td>[rad/s]</td>
</tr>
<tr>
<td>$l_{T1}$</td>
<td>the length of the flexor muscle’s tendon</td>
<td>[m]</td>
</tr>
<tr>
<td>$l_{T2}$</td>
<td>the length of the extensor muscle’s tendon</td>
<td>[m]</td>
</tr>
<tr>
<td>$v_{T1}$</td>
<td>the extraction velocity of the flexor muscle’s tendon</td>
<td>[m/s]</td>
</tr>
<tr>
<td>$v_{T2}$</td>
<td>the extraction velocity of the extensor muscle’s tendon</td>
<td>[m/s]</td>
</tr>
<tr>
<td>$u_1$</td>
<td>the flexor muscle’s activation signal</td>
<td>-</td>
</tr>
<tr>
<td>$u_2$</td>
<td>the extensor muscle’s activation signal</td>
<td>-</td>
</tr>
</tbody>
</table>

Variables, which can be computed from the state variables by using equation (18), and are used in the functions $F_{\text{flexor}}$ and $F_{\text{extensor}}$:

<table>
<thead>
<tr>
<th>symbol</th>
<th>description</th>
<th>dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_{CE1}$</td>
<td>the length of the flexor muscle</td>
<td>[m]</td>
</tr>
<tr>
<td>$l_{CE2}$</td>
<td>the length of the extensor muscle</td>
<td>[m]</td>
</tr>
<tr>
<td>$v_{CE1}$</td>
<td>the contraction velocity of the flexor muscle</td>
<td>[m/s]</td>
</tr>
<tr>
<td>$v_{CE2}$</td>
<td>the contraction velocity of the extensor muscle</td>
<td>[m/s]</td>
</tr>
</tbody>
</table>

19
The ranges of the variable’s values in the simulation environment:

<table>
<thead>
<tr>
<th>variable</th>
<th>minimum value</th>
<th>maximum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1$</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$q_2$</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>-0.2</td>
<td>$2\pi$</td>
</tr>
<tr>
<td>$\omega$</td>
<td>-30</td>
<td>30</td>
</tr>
<tr>
<td>$l_{T1}$</td>
<td>0.09</td>
<td>0.11</td>
</tr>
<tr>
<td>$l_{T2}$</td>
<td>0.09</td>
<td>0.11</td>
</tr>
<tr>
<td>$v_{T1}$</td>
<td>-10</td>
<td>10</td>
</tr>
<tr>
<td>$v_{T2}$</td>
<td>-10</td>
<td>10</td>
</tr>
<tr>
<td>$l_{CE1}$</td>
<td>0.19</td>
<td>0.41</td>
</tr>
<tr>
<td>$l_{CE2}$</td>
<td>0.19</td>
<td>0.41</td>
</tr>
<tr>
<td>$v_{CE1}$</td>
<td>-2</td>
<td>2</td>
</tr>
<tr>
<td>$v_{CE2}$</td>
<td>-2</td>
<td>2</td>
</tr>
<tr>
<td>$u_1$</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The constants that appear in the model are as follows:

<table>
<thead>
<tr>
<th>constant</th>
<th>dimension</th>
<th>value</th>
<th>constant</th>
<th>dimension</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega$</td>
<td></td>
<td>0.56</td>
<td>$\tau_{act}$</td>
<td>[s]</td>
<td>0.012</td>
</tr>
<tr>
<td>$l_{opt}$</td>
<td>[m]</td>
<td>0.3</td>
<td>$k_T$</td>
<td>[N/m]</td>
<td>12500</td>
</tr>
<tr>
<td>$l_{slack}$</td>
<td>[m]</td>
<td>0.1</td>
<td>$s_T$</td>
<td>[Ns/m]</td>
<td>1250</td>
</tr>
<tr>
<td>$A_{ref}$</td>
<td></td>
<td>0.41</td>
<td>$z_T$</td>
<td>$[N\cdot s^2/m]$</td>
<td>1</td>
</tr>
<tr>
<td>$B_{ref}$</td>
<td></td>
<td>5.2</td>
<td>$d$</td>
<td>[m]</td>
<td>0.05</td>
</tr>
<tr>
<td>$c_1$</td>
<td></td>
<td>-0.461</td>
<td>$d_{prox}$</td>
<td>[m]</td>
<td>0.2</td>
</tr>
<tr>
<td>$c_2$</td>
<td></td>
<td>-1.5</td>
<td>$d_{dist}$</td>
<td>[m]</td>
<td>0.2</td>
</tr>
<tr>
<td>$c_3$</td>
<td></td>
<td>-0.992</td>
<td>$g_x$</td>
<td>$[m/s^2]$</td>
<td>0</td>
</tr>
<tr>
<td>$F_{asy}$</td>
<td>[N]</td>
<td>1.5</td>
<td>$g_y$</td>
<td>$[m/s^2]$</td>
<td>-10</td>
</tr>
<tr>
<td>$F_{PE\max}$</td>
<td>[N]</td>
<td>200</td>
<td>$\xi$</td>
<td>[rad]</td>
<td>$-\pi/2$</td>
</tr>
<tr>
<td>$F_{\max}$</td>
<td>[N]</td>
<td>200</td>
<td>$l_{COM}$</td>
<td>[m]</td>
<td>0.15</td>
</tr>
<tr>
<td>$F_{\max}^{lig}$</td>
<td>[N]</td>
<td>1000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The constants $\omega$, $A_{ref}$, $B_{ref}$, $c_1$, $c_2$, $c_3$ are determinated form [10], $\tau_{act}$ is determined from [27], $l_{CE}$, $l_{slack}$, $F_{\max}$ and $k_T$ from [26], $F_{asy}$ from [22], $F_{\max}^{lig}$ from [5], $d$ from [19], $l_{COM}$ from [29]. $s_T$ and $z_T$ were determinated from modelling aims. We have to note, that the main aim of the model constructing was the simulation of the limb’s general behavior. For the control analysis of the simple model, the accurate values of the parameters not have significant importance.
2.7 Model verification

The model verification has been performed by simulation where the model response has been tested against engineering intuition. The following test case was used: The activation signal of the flexor muscle was equal to 1 from $t=0$ to $t=0.5$, 0 from $t=0.5$ to $t=1$, 0.5 from $t=1$ to $t=1.5$ and 0.6 from $t=1.5$ to $t=2$. The extensor muscle’s activation signal was equal to 0 from $t=0$ to $t=0.5$, 1 from $t=0.5$ to $t=1$, 0.5 from $t=1$ to $t=1.5$ and 0.1 from $t=1.5$ to $t=2$, as it can be seen in the following figure.

![Figure 5: The muscle activation signals](image)

The muscle activation states are shown in figure 6, as functions of time ($q_1$ is the flexor muscle’s activation, $q_2$ is the extensor’s):

![Figure 6: The muscle activation states](image)
Figure 7 shows $\alpha$ and $\omega$ as functions of time:

Figure 7: $\alpha$ and $\omega$

The length of the tendons - $l_T_1$ (the flexor muscle’s tendon) and $l_T_2$ (the extensor muscle’s tendon) - are depicted in figure 8:

Figure 8: The length of the tendons

The length of the muscles are seen in figure 9 with $l_{CE1}$ being the flexor muscle’s length and $l_{CE2}$ being the extensor muscle’s length:
In figures 7 and 8 we can clearly see the strong forces of the ligaments appearing at the maximal flexed/extended states.

Taking into account the aim of the model construction, the simplifications used in the construction process and the simulation results, the model behaves as one expects, thus the model is verified and is acceptable for controller design purposes.
2.8 A biological extension: A simple Gamma-loop model

The $\gamma$-loop is a well known solution in human and other species for some problems emerging in muscle actuation. The short description of the mechanism, and its components can be found in the appendix. (see more in [23])

Figure 10: The servomechanical $\gamma$-loop as depicted in [23]

2.8.1 The simplified model of the Gamma-loop

Simplifying assumptions

- The receptors of the muscle spindles can sense only the length difference between the muscle fibers inside the muscle spindles $l_s$, and the surrounding muscle fibers, of which length are commensurable to the working muscle length $l_{CE}$.

- The activation signal of the spinal neurons is linear in the signal of the muscle spindles’ detector.
• In this case, there’s no other input to the system, only the muscle spindle’s activation signals.

In this case, we can extend our model with two more state-space variables: With the lengths of the muscle fibers of the muscle spindles ($l_{s1}$ in the case of the flexor muscle, and $l_{s2}$ in the case of the extensor muscle).

If we suppose, that the normal length of the muscle spindles (and the surrounding muscle fibers) is always one tenth of the working muscle, we can describe the behavior of the new variables with the following equations:

$$\begin{align*}
\frac{dl_{s1}}{dt} &= c_P(l_{CE1}/10 - l_{s1}) - c_Gu_{s1}(t) \\
\frac{dl_{s2}}{dt} &= c_P(l_{CE1}/10 - l_{s2}) - c_Gu_{s2}(t)
\end{align*}$$

(21)

where $c_P$ [1/s] is a constant showing how quick the length of the muscle spindles (and the muscle fibers inside the muscle spindles) follows the length of the working muscle ($l_{CE1}$, $l_{CE2}$), $c_G$ is a constant showing how sensitive the muscle spindle is to the external activation signal of the descending tracts, and $u_{s}(t)$ is the activation signal of the descending tracts acting on the the muscle fibers inside the muscle spindles. If we consider the normalized values of $l_{CE1}/10 - l_{s1}$ and $l_{CE2}/10 - l_{s2}$ as the signals of the muscle spindles’s receptors we can compute the activation states of the working muscles (denoted with $q_1$ and $q_2$ in the equation (20)):

$$\begin{align*}
q_1 &= c_G(l_{CE1}/10 - l_{s1}) \\
q_2 &= c_G(l_{CE2}/10 - l_{s2})
\end{align*}$$

(22)
2.8.2 Simulation results

We act on the flexor muscle’s spindle with the following activation signal:

![Activation signal of the flexor muscle’s spindle](image)

**Figure 11:** the activation signal of the flexor muscle’s spindle

We can see the muscle lengths, and the length of the muscle spindles (multiplied with 10 for the better comparability) in the next figure:

![Length of muscles and muscle spindles](image)

**Figure 12:** the length of the muscles and the muscle spindles
In the next two figures, we can see the working muscles’s activity states $(q_1$ and $q_2)$ the angle, and the angle velocity of the joint:

Figure 13: the activation state of the working muscles

![Figure 13: the activation state of the working muscles](image)

Figure 14: joint angle and velocity

We can see, that in the beginning as the joint angle decreases the flexor muscle begins to extract. The flexor muscle’s spindle follows the extraction with a little delay, so it generates a little activity in the flexor muscle - and so a little force, slowing down the decreasing of the joint angle a bit. As the activation signal’s impulse reaches the flexor muscle’s spindle, it contracts, generating a big activation state growing in the flexor muscle, that increases the joint angle.

We can see, that in general case, the $\gamma$-loop mechanism prevents the joint from sudden movements, providing more smooth movement of the limb.

Furthermore, the mechanism increases the relative degree (see later) of the system.
3 Model analysis

The aim of the model analysis is to determine the dynamic properties, that influence the controller design (stability controllability, observability, zero dynamics etc.).

Let us write the state-space model in the following input affine form, that can be obtained from equation (20):

\[ \dot{x} = f(x) + \sum_{i=1}^{m} g_i(x)u_i(t) \]

\[ y = h(x) \]  

(23)

where:

\[
\begin{pmatrix}
-\frac{1}{\tau_{\text{act}}} \beta x_1 \\
-\frac{1}{\tau_{\text{act}}} \beta x_2 \\
x_4 \\
\frac{1}{l + m_{\text{com}}^2} (M(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) + m_{\text{com}} \cos(x_3) g_y) \\
x_7 \\
x_8 \\
-k_T (x_5 - l_{\text{slack}}) + s_T x_7 - F_{\text{tor}}(x_1, x_3, x_4, x_5, x_7) \\
-\frac{k_T (x_5 - l_{\text{slack}}) + s_T x_7 - F_{\text{tor}}(x_1, x_3, x_4, x_5, x_7)}{z_T} \\
-k_T (x_6 - l_{\text{slack}}) + s_T x_8 - F_{\text{tor}}(x_2, x_3, x_4, x_6, x_8) \\
-\frac{k_T (x_6 - l_{\text{slack}}) + s_T x_8 - F_{\text{tor}}(x_2, x_3, x_4, x_6, x_8)}{z_T}
\end{pmatrix}
\]
We can observe, that the state function $f$ is nonlinear, but both the input functions $g_1$ and $g_2$ and the output function $h$ are linear.
3.1 Linear analysis

The well-known standard methods of linear analysis can only be applied for linear time-invariant (LTI) systems with the following state-space model:

\[ \dot{x} = Ax + Bu \\
y = Cx \]  \hspace{1cm} (24)

3.1.1 Linearization around a steady-state point

We call \( x_0(u_0) \) a steady-state point of the system, if

\[ \dot{x} = f(x)|_{x_0} + \sum_{i=1}^{m} g_i(x)u_i(t)|_{u_0} = 0 \]

From the input-affine state space model in equation (23) we can obtain a LTI model by linearizing it locally around a steady-state point \( x_0(u_0) \), as in [2].

Let us take the centered variables:

\[ \tilde{x} = x - x_0 \quad \tilde{u} = u - u_0 \]

To obtain the LTI model form:

\[ \dot{\tilde{x}} = \tilde{A}\tilde{x} + \tilde{B}\tilde{u} \]
\[ \tilde{y} = \tilde{C}\tilde{x} \]  \hspace{1cm} (25)

we can compute the matrixes \( \tilde{A} \), \( \tilde{B} \) and \( \tilde{C} \) in the neighborhood of the steady-state point \( (x_0, u_0) \) in the following way:

\[
\begin{align*}
\tilde{A} &= J(f, x)|_{x(0)} + J(g, x)|_{x(0)}u(0) \\
\tilde{B} &= g(x_0) \\
\tilde{C} &= J(h, x)|_{x(0)}
\end{align*}
\]  \hspace{1cm} (26)

where \( J(v, x) \) is the Jacobian-matrix of the multivariate function \( v \):

\[ J_{j,i}^{(v, x)} = \frac{\partial v_j}{\partial x_i} \]
Two steady-state points were selected for linear analysis purposes:

- The most extended steady-state of the limb (very low effect of gravity)
  - both of the muscles are totally inactive.
- π/2 joint angle (no effects of ligaments and passive force)

### 3.1.2 Analysis of the linearized model at the first steady-state point

Equation (27) contains the state matrix $\tilde{A}$ in the first steady state point:

\[
\tilde{A} = 10^{4} * \begin{pmatrix}
-0.0042 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -0.0042 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.0001 & 0 & 0 & 0 & 0 \\
0.0112 & -0.0102 & -0.0047 & -0.0004 & -0.000 & 0.000 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.0001 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0001 \\
0.052 & 0 & -0.0004 & -0.0002 & -1.25 & 0 & -1.25 & 0 \\
0 & 0.0048 & 0.000 & 0.000 & 0 & -1.25 & 0 & -1.25
\end{pmatrix}
\]

The rank of $A$ is 8.

We obtain the following results at the steady-space point in the neighborhood of the most extended state of the limb:

Steady-state conditions (in case of $u_1 = u_2 = 0$):

\[
\begin{pmatrix}
x_{1,0} = 0 \\
x_{2,0} = 0 \\
x_{3,0} \simeq 0.004516 \\
x_{4,0} = 0 \\
x_{5,0} \simeq 0.1 \\
x_{6,0} = 0.1 \\
x_{7,0} = 0 \\
x_{8,0} = 0
\end{pmatrix}
\]

The $\tilde{A}$ matrix’s eigenvalues are the following:
If the eigenvalues of the state-matrix $\tilde{A}$ have negative real part, then the linear system is stable, and this is the case in the first steady-state point.

Next we analyze the controllability and observability testmatrices, constructed from the $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$ matrices of the linearized system - see [17].

In the first steady-state point:

$$\tilde{B} = \begin{pmatrix} 65.7255 & 0 \\ 0 & 83.333 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} , \quad \tilde{C} = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]$$

Controllability testmatrix:

$$M_c = [B \ AB \ldots A^{n-1}B]$$

(30)

If the rank of the controllability testmatrix is equal to $n$ (the number of states) the system is state-controllable, which means also output controllability in our case.

Observability testmatrix:

$$M_0 = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{pmatrix}$$

(31)
If the rank of the observability test matrix is equal to \( n \) (the number of states) the system is observable.

The results of the analysis in the first steady-state point is as follows:

- The linearized system is stable.
- The linearized system is controllable.
- The linearized system is observable.

### 3.1.3 Analysis of the linearized model at the second steady-state point

We can find another steady state-point at \( \pi/2 \) joint angle with the following steady-state conditions (of course, in this case the flexor muscle is active - we take a case, when only the flexor muscle is active - \( u_1 = 0.0554 \) - and the extensor muscle is totally inactive - \( u_2 = 0 \)):

\[
\begin{bmatrix}
  x_1(0) = 0.1050 \\
  x_2(0) = 0 \\
  x_3(0) = \pi/2 \\
  x_4(0) = 0 \\
  x_5(0) = 0.101439 \\
  x_6(0) = 0.1 \\
  x_7(0) = 0 \\
  x_8(0) = 0
\end{bmatrix}
\]  

(32)

In this point:

\[
\tilde{A} = 10^4 \begin{bmatrix}
  -0.0052 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & -0.0042 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 0 & 0.0001 & 0 & 0 & 0 & 0 \\
  0.0095 & -0.0083 & -0.0009 & 0.000 & -0.0206 & 0.000 & -0.0037 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0.0001 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0001 \\
  0.0045 & 0 & -0.0004 & 0.000 & -1.2597 & 0 & -0.1267 & 0 \\
  0 & 0.0039 & 0.000 & 0.000 & 0 & -1.2500 & 0 & -0.1250
\end{bmatrix}
\]  

(33)

The rank of the matrix \( \tilde{A} \) in this case is 8.
The eigenvalues are as follows:

\[
\begin{pmatrix}
-1.2570 \\
0.0000 + 0.0030i \\
0.0000 - 0.0030i \\
-0.0100 \\
-1.2399 \\
-0.0101 \\
-0.0520 \\
-0.0420 \\
\end{pmatrix}
\]

(34)

\(\tilde{B}, \tilde{C}\) are the same as above, and \(M_C, M_O\) are computed the same way. The results of the analysis in the second steady-state point is as follows:

- The linearized system is at the edge of stability (it has a pole-pair with 0 real-part)
- The linearized system is controllable
- The linearized system is observable

### 3.2 Relative degree

The SISO (single input, single output) nonlinear system

\[
\begin{align*}
\dot{x} &= f(x) + g(x)u \\
y &= h(x)
\end{align*}
\]

(35)

has relative degree \(r\) at a point \(x^0\) if

1. \(L^k_g L^k_f h(x) = 0\) for all \(x\) in a neighborhood of \(x^0\) and all \(k > r - 1\)

2. \(L^r_g L^{-1}_f h(x^0) \neq 0\)

where \(L_g h(x) = \frac{dh(x)}{dx} g(x)\) is the Lie-derivative of \(h(x)\) along \(g\). Instead of performing the Lie-derivatives we can easily determine the relative degree of a system, using graph-theoretic methods [2]. We have to determine the length of the shortest directed path from the input to the output vertex in the structure graph (see below), and this length -1 is the relative degree of the system.
3.2.1 The structure graph

The structure graph of a system is constructed in the following way:

- The vertices of the graph are the state-variables, the input variables, and the output variables.

- A directed path leads from the vertex $V_1$ to the vertex $V_2$ if and only if the variable $V_2$ depends on $V_1$ (If $V_2$ is a state-variable, this means that $V_1$ can be found in the state-equation describing the time derivative of $V_2$. An output variable depends on a state variable if and only if the state variable can be found in its output equation).

Figure 15 shows the structure graph of our simple limb model.
In our case the length of the shortest directed path from $u_1$ (or form $u_2$) to $y$ is 4, therefore the relative degree of the system is 3.

![Figure 15: The structure graph](image_url)
3.3 Controllability

In general case we study the controllability of the original nonlinear state-space model (see in equation (20)) with the tool of the controllability distribution ($\Delta_c$) (see [11]).

3.3.1 Algorithm for constructing the controllability distribution

1. Starting point

$$\Delta_0 = \text{span}\{g_1, g_2\} \hspace{1cm} (36)$$

2. Development of the controllability distribution

$$\Delta_k = \Delta_{k-1} \bigcup \sum_{i=0}^{m} [g_i, \Delta_{k-1}] \hspace{1cm} (37)$$

where:

$$[g_i, \Delta_{k-1}] = \text{span}\{[g_i, \phi_1], ..., [g_i, \phi_l]\}$$

where $\phi_1...\phi_l$ are the vectors spanning $\Delta_{k-1}$, and $[f, g]$ denotes the Lie-product of $f$ and $g$:

$$[f, g](x) = \frac{\partial g(x)}{\partial x}f(x) - \frac{\partial f(x)}{\partial x}g(x)$$

$\frac{\partial g(x)}{\partial x}$ is the Jacobian matrix of the function $g$ with respect to its independent variable $x$.

3. Stopping condition

If $\Delta_k = \Delta_{k-1}$, then $\Delta_c = \Delta_k$

If the dimension of the controllability distribution is equal to $n$ (the number of states), then the system is controllable.

In our case the symbolic expressions needed for computing of the controllability distribution are so complex, that it is impossible to determine the Lie-brackets with the available methods, so we use another method for controllability analysis.
3.3.2 Structural controllability and observability

Structural controllability and observability [2] apply for a set of systems with the same structure, i.e., with the same structure graph.

A set of linear or linearized systems with structure matrices \( ([A], [B], [C]) \) is structurally controllable or observable if

- The matrix \([A]\) is of full structural rank (maximal possible rank within the class specified by the structure matrix)

- The system structure graph \(S\) is input connectable or output connectable. For controllability there should be at least one directed path from any of the input variables to each of the state variables. For observability there should be at least one directed path from each of the state variables to one of the state variables.

Structural controllability implies that the points where the system is not controllable are forming a set with 0 measure.

3.3.3 Structural controllability analysis of the model

Because the rank of the matrix \( \tilde{A} \) of the linearized system is 8 in both investigated steady-state points, its structural rank is 8. We can apply the structural controllability method for the entire set of the locally linearized systems.

We can see in figure 15, that we can find directed paths from the inputs to the states, so the system is structurally controllable. This is in a good agreement with the results of the linear controllability analysis (see section 3.1).

3.4 Observability

3.4.1 Algorithm for constructing the observability co-distribution

Similarly to the controllability-analysis, in the general case we study the observability of the model with the observability co-distribution \( \Omega_o \).

1. Starting point

\[
\Omega_0 = \text{span}\{g_1, g_2\}
\]  

(38)

2. Development of the observability co-distribution
\[ \Omega_k = \Omega_{k-1} \bigcup \sum_{i=0}^{m} L_{g_i} \Omega_{k-1} \]  

(39)

where: \( L_{g_i} \Omega_{k-1} \) is the Lie-derivative of \( \Omega_{k-1} \) along \( g_i \), and the expansion to the spanned sub-space is the same as above.

3. **Stopping condition**
   If \( \Omega_k = \Omega_{k-1} \), then \( \Omega_o = \Omega_k \)

   If the dimension of the controllability distribution is equal to \( n \) (the number of states), the system is controllable.

### 3.4.2 Structural observability analysis of the model

Similarly to the controllability, we study structural observability. In figure 15 we can find directed paths from the states to the output, so the system is structurally observable. This is in a good agreement with the results of the linear observability analysis (see section 3.1)
3.5 Stability

We can easily determine the stability of a linear system, by analyzing its eigenvalues as in section 2.1. In the nonlinear case, we use the Lyapunov-theory.

3.5.1 Lyapunov-stability

definition:
If \(\xi(t)\) is a solution of the differential equation \(\dot{x} = f(x, t)\), we say that the solution \(\xi(t)\) is a Lyapunov stable solution, if for \(\forall \ t_0 \in [a, \infty)\) and for \(\forall \ \varepsilon > 0 \ \exists \ \delta(\varepsilon, t_0) > 0\), that if \(x(t)\) is a solution and \(\|x(t_0) - \xi(t_0)\| < \delta(\varepsilon, t_0)\), then \(\forall \ t \in [t_0, \infty) \ \|x(t) - \xi(t)\| < \varepsilon\).

3.5.2 Strong asymptotic stability

definition:
If \(\xi(t)\) is a solution of the differential equation \(\dot{x} = f(x, t)\), we say that the solution \(\xi(t) = 0\) is a strong asymptotic-stable solution, if for \(\forall \ t_0 \in [a, \infty)\) and for \(\forall \ \varepsilon > 0 \ \exists \ \delta(\varepsilon, t_0) > 0\), that if \(x(t)\) is a solution and \(\|x(t_0) - \xi(t_0)\| < \delta(\varepsilon, t_0)\), then \(\forall \ t \in [t_0, \infty) \ \|x(t) - \xi(t)\| \to 0\).

3.5.3 Lyapunov’s first theorem

Let us suppose, that \(x_0\) is a steady-state point and \(V(x)\) is a positive definite scalar-valued function (Lyapunov-function). If
\[
\dot{V}(t, x) = \frac{dV(t, x(t))}{dt} \leq 0
\]
then the steady-state point is Lyapunov-stable.

If
\[
\dot{V}(t, x) = \frac{dV(t, x(t))}{dt} < 0
\]
then the steady-state point \(x_0\) is asymptotical stable in strong sense (see more in [17]).
3.5.4 Estimating the stability region in the neighborhood of the steady-state point with quadratic Lyapunov-function candidate

Let us take the second steady-space point, and the following quadratic Lyapunov function candidate:

\[ V[x(t)] = (x - x^*)^T Q (x - x^*) \]  
(40)

where \( x^* \) is equal to the vector (32), and \( Q \) is a diagonal unit weighting matrix. This results in the Lyapunov-function candidate:

\[
V[x(t)] = (x_1 - 0.105)^2 + x_2^2 + (x_3 - 1/2 \pi)^2 + x_4^2 \\
+ (x_5 - 0.101)^2 + (x_6 - 0.1)^2 + x_7^2 + x_8^2
\]  
(41)

If we compute \( \frac{dV}{dt} \), we can give a conservative estimation of the stability region. The points were \( \frac{dV}{dt} \) is negative, belong to the strong asymptotically stable region of the steady-state point.

\[
\frac{dV}{dt} = \frac{\partial V}{\partial x} \frac{dx}{dt} = \frac{\partial V}{\partial x} \bar{f}(x)
\]  
(42)

where \( \bar{f}(x) = f(x) + g(x)C(x) \) with \( C(x) \) being the static feedback law. In the case of the first steady-state point \( C(x) = u_1 = 0.0554 \).

At the steady-space point (32) the value of \( \frac{dV}{dt} \) is of course zero, because the system is on the edge of stability. 
Along \( \alpha = x_3, \ l_{T1} = x_5 \) and \( l_{T2} = x_6 \) the function \( \frac{dV}{dt} \) does not change, because of the Hamiltonian properties of the system (the description of Hamiltonian systems can be found in [9]).

We can study the behavior of \( \frac{dV}{dt} \) in the neighborhood of the state-point described in (32), along the directions parallel to the axes \( x_1, x_2, x_4, x_7, x_8 \). So, we can get cuts of the function (42). The following figures depict \( \frac{dV}{dt} \) as a function of state-space variable pairs.
Figure 16: The change of $\frac{\partial V}{\partial t}$ along $q_1$ and $\omega$

Figure 17: The change of $\frac{\partial V}{\partial t}$ along $q_1$ and $q_2$

Figure 18: The change of $\frac{\partial V}{\partial t}$ along $q_2$ and $\omega$
3.6 Zero dynamics

We call the systems behavior zero dynamics in the case, when its is output forced to be identically zero (all time derivatives of the output are zero). We need to examine the zero dynamics’s stability for controller design purposes.

Let us study the zero-dynamics for the input-output pair: $u_1$ (the flexor muscle’s activation signal), and $h(x) = x_3 - \pi/2$. In this case $y \equiv 0$ output means $90^\circ$ joint angle.

$$\alpha = x_3 \equiv \pi/2 \Rightarrow \dot{x}_3 = 0 \Rightarrow \omega = x_4 \equiv 0 \Rightarrow \dot{x}_4 = 0$$

From equation (20):

$$\dot{x}_4 = \frac{1}{\Theta + ml_{COM}^2}(M + ml_{COM}cos(x_3 + \xi)g_y) = 0 \quad (43)$$

with $\alpha = \pi/2$ and $\xi = -\pi/2$ we get $cos(\alpha + \xi) = 1$.

Next we define a constant denoted by $H_1$:

$ml_{COM}g_y \doteq H_1 < 0$ where $g_y = -10 m/s^2$.

Equation (43) can be equal to 0, if only if the joint torque $M$ is equal to $H_1$, thus equation (43) implies

$M = -H_1$
We know from equation (12) that:

\[ M = M_f - M_e = F_{\text{flexor}}d - F_{\text{ext}}d = (F_{CE1} + F_{\text{big1}})d - (F_{CE2} + F_{\text{big2}})d \]

In the position \( \alpha = \pi/2 \) we can neglect the forces of ligaments (because they only appear at the position near to the maximal flexed/extended states), and we suppose, that the extensor muscle is totally inactive, so we can also neglect \( F_{CE2} \). This simplifies the equation above to

\[ M = F_{CE1}d = (F_{\text{max}} FL(l_{CE})F_{v}(v_{CE})x_1 + F_{PE})d \]

Passive force of the muscle does also not appear in this position, so we neglect it, and obtain the simplified equation for the joint torque:

\[ M = F_{CE1}d = (F_{\text{max}} FL(l_{CE})F_{v}(v_{CE})x_1)d \] (44)

If \( \alpha \equiv \pi/2, \; d^{\text{prox}} = d^{\text{dist}} = 0.2m \) and \( d = 0.01 \) we can compute the flexor muscle’s length from (18) as:

\[ l_{CE} = 0.3368 - x_5 \]

So, we can determinate \( FL(l_{CE}) \), which will only depend on \( x_5 = l_{T1} \) by using equation (1).

Furthermore, if

\[ 1 - \omega \leq \left( \frac{l_{CE}}{l_{opt}} \right) \leq 1 + \omega \]

then we can compute the flexor muscle’s contracting velocity from the time derivative of the muscle length:

\[ v_{CE} = \frac{dl_{CE}}{dt} \]

If we take \( \alpha = \pi/2 \), \( v_{CE} \) will depend only on \( x_7 = v_{T1} \)

Next we can compute \( F_{v}(v_{CE}) \) by using the equation (3).
From equation (44) and $M = -H_1$ we get:

$$\frac{-H_1}{dF_{\text{max}}FL(l_{CE}(x_5))F_v(v_{CE}(x_7))} = x_1 = \frac{-H_2}{FL(l_{CE}(x_5))F_v(v_{CE}(x_7))}$$

(45)

where $H_2$ is a newly defined constant.

In this case zero-dynamics means that the torques originating from the tendon’s dynamics have to be balanced by the change of muscle activation state. The states of tendon change the length (and so the contracting velocity) of muscles - in this way they change the muscle’s force. This means the change of the joint torques. The muscle activation also changes the muscle’s force, balancing the equilibrium of the joint torques.

### 3.6.1 Zero dynamics state-space equations

With the simplifications of the zero dynamics above, and from equation (20) we can write the zero-dynamics state-space model in the following form:

\[
\begin{align*}
\dot{x}_1 &= \frac{dq_1}{dt} = -\left(\frac{1}{\tau_{act}}(\beta + [1 - \beta]u_1(t))\right) \frac{-H_2}{FL(l_{CE}(x_5))F_v(v_{CE}(x_7))} + \frac{1}{\tau_{act}} u_1(t) \\
\dot{x}_2 &= \frac{dq_2}{dt} = 0 \\
\dot{x}_3 &= \frac{da}{dt} = x_4 = 0 \\
\dot{x}_4 &= \frac{d\alpha}{dt} = \frac{1}{\Theta + m_C^{COM}} \left( (F_{\text{max}}FL(l_{CE}(x_5))F_v(v_{CE}(x_7))) \frac{-H_2}{FL(l_{CE}(x_5))F_v(v_{CE}(x_7))} d + H_1 \right) = 0 \\
\dot{x}_5 &= \frac{dl_1}{dt} = x_7 \\
\dot{x}_6 &= \frac{dl_2}{dt} = x_8 \\
\dot{x}_7 &= \frac{dv_{\ell_1}}{dt} = \frac{-k_l(x_5 - l_1^{\text{slack}} + s_7x_7 - F_{\text{max}}FL(l_{CE}(x_5))F_v(v_{CE}(x_7)))}{z_T} + \frac{-H_2}{z_T} \\
&= \frac{-k_l(x_5 - l_1^{\text{slack}} + s_7x_7 + F_{\text{max}}H_2}{z_T} \\
\dot{x}_8 &= \frac{dv_{\ell_2}}{dt} = \frac{-k_l(x_6 - l_1^{\text{slack}} + s_8x_8}{z_T} \\
\end{align*}
\]

(46)
3.6.2 Computing the zeroing input

If we know the form of \( q_1 = x_1 \) as the function of time in equation (45), we can compute its time-derivative:

\[
\frac{dx_1}{dt} = \frac{d}{dt} F_L(l_{CE})(x_5) F_v(v_{CE})(x_7)
\]

\[
= H_2 \frac{1}{(F_L(l_{CE})(x_5) F_v(v_{CE})(x_7))^2} \cdot \left( F_v(v_{CE})(x_7) \frac{d}{dt} F_L(l_{CE})(x_5) + F_L(l_{CE})(x_5) \frac{d}{dt} F_v(v_{CE})(x_7) \right)
\]

By the derivation rule of composite functions we can compute \( \frac{d}{dt} F_L(l_{CE}(x_5)) \) which will depend on \( x_5 \) and \( x_7 \).

Similarly, we can also compute \( \frac{d}{dt} F_v(v_{CE}) \).

If we substitute the results to (47) we can compute \( \frac{dx_1}{dt} \).

If we take the first equation of (46), and we know the form of \( \frac{\partial x_1}{\partial t} \), we can compute the zeroing input as follows:

\[
\frac{dx_1}{dt} = \frac{1}{\tau_{act}} \beta x_1 + \left( -\frac{1}{\tau_{act}} (1 - \beta) x_1 + \frac{1}{\tau_{act}} \right) u(t)
\]

and therefore

\[
u(t) = \frac{\frac{dx_1}{dt} + \frac{1}{\tau_{act}} \beta x_1}{\frac{1}{\tau_{act}} (1 - \beta) x_1 + \frac{1}{\tau_{act}}}
\]

where \( \tau_{act} \) and \( \beta \) are the same as in equation (5).

3.6.3 Zero-dynamics analysis results

If we apply the input described in equation (48) to the open-loop system, we can notice that the output is indeed identically zero.

If we analyze the zero-dynamics in some points of the state-space (this was done mostly in steady-state points) we get the result that the linearized zero-dynamics is stable in every investigated case. So we can suppose, that the zero-dynamics of the system is stable. This result is important for input-output type controller design.
4 Controller design

The aim of this chapter is to propose different controllers to the simple limb model and to investigate and compare their performance. For this purpose we return back to the original model (\(\gamma\)-loop not included) described in equation (20). In addition, we suppose that all of the state-space variables are available for the control system for a state feedback.

4.1 Performance specification

Any controller design starts with the explicit definition of the criteria for the control system’s performance.

4.1.1 Control aims

Based on the most simple tasks a human upper limb should perform, we determine three basic tasks for the control system:

- Stability: We expect the closed loop system to be stable in the region we apply the selected control method.
- Regulation and Trajectory following: We expect the applied control solving this two basic tasks in limb movement control.
- Disturbance-rejection: We expect the controller being not seriously sensitive for disturbances.

4.1.2 Control domain

Our aim is to control the output of the system in the neighborhood of \(\alpha = \pi/2\) angle. The values of other the state-space variables can change in the domain described in section 2.6.3. The value of the actuation signal can change between 0 and 1 as described in 2.3.4 and 2.6.3.

4.1.3 Disturbances

We complete the model with three disturbances:

- An external mass on the edge of the moving limb-part: This can be a weight lifted by the limb.
• A simplified fatigue model - the drift of the $\tau_{act}$ parameter: As with time the ATP concentration decreases in the working muscle, the activation time increases.

• Spontaneous muscle activity: This phenomenon can be considered for example, as a simplified effect of the Parkinson’s disease.

The state-space equations, modified by the disturbances are in the following form:

$$
\dot{x}_1 = -\left(\frac{1}{\tau_{act}(t)}(\beta + [1 - \beta]u_1(t))\right)x_1 + \frac{1}{\tau_{act}(t)}u_1(t) + d_1(t)
$$

$$
\dot{x}_2 = -\left(\frac{1}{\tau_{act}(t)}(\beta + [1 - \beta]u_2(t))\right)x_2 + \frac{1}{\tau_{act}(t)}u_2(t) + d_2(t)
$$

$$
x_3 = x_4
$$

$$
\dot{x}_4 = \frac{1}{\Theta + ml_{COM}}(M(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) + ml_{COM}\cos(x_3 + \xi)g_y + m_D l_{fo} \cos(x_3 + \xi)g_y)
$$

$$
\dot{x}_5 = x_7
$$

$$
\dot{x}_6 = x_8
$$

$$
\dot{x}_7 = \frac{k_z(x_5 - l_{slack}) + sT x_{7} - F_{extor}(x_1, x_3, x_4, x_5, x_7)}{sT}
$$

$$
\dot{x}_8 = \frac{k_z(x_6 - l_{slack}) + sT x_{8} - F_{extor}(x_2, x_3, x_4, x_6, x_8)}{sT}
$$

where $m_D$ is the external mass, $l_{fo}$ is the length of the moving limb-part (the forearm), $d_1(t)$ and $d_2(t)$ are the spontaneous muscle activities as functions of time.
In order to design a linear SISO (single input, single output) LQ regulator [17] we use the linearized system model obtained around the steady-state point at $\pi/2$ rad and the centered variables $\tilde{x} = x - x_0$ $\tilde{u} = u - u_0$, as in equation (26). We use the flexor muscle’s activation signal as the only input to the system. The other input, $u_2$ is set to be identically zero.

\[
\begin{bmatrix}
  x_1(0) = 0.4031 \\
  x_2(0) = 0 \\
  x_3(0) = \pi/2 \\
  x_4(0) = 0 \\
  x_5(0) = 0.101439 \\
  x_6(0) = 0.1 \\
  x_7(0) = 0 \\
  x_8(0) = 0
\end{bmatrix}
\]

(50)

\[u_0 = 0.2528\]

We can apply an LQ servo controller (see below) to minimize the cost function:

\[
\int (\tilde{x}'Q\tilde{x} + \tilde{u}'R\tilde{u}) \ dt
\]

(51)
If A, B and C are the matrices of the linearized system as in section 4.1 (denoted there by \( \tilde{A}, \tilde{B}, \tilde{C} \)) we design an LQ controller that is in the form of a state feedback \( u(t) = -Kx(t) \) for the linearized system.

The extended state space model used for designing the LQ-servo controller is as follows:

\[
\begin{bmatrix}
\dot{x}(t) \\
\dot{\tilde{z}}(t)
\end{bmatrix} =
\begin{bmatrix}
A & 0 \\
-C & 0
\end{bmatrix}
\begin{bmatrix}
x(t) \\
z(t)
\end{bmatrix} +
\begin{bmatrix}
B \\
0
\end{bmatrix}
\]

\[
y(t) =
\begin{bmatrix}
C & 0
\end{bmatrix}
\begin{bmatrix}
x(t) \\
z(t)
\end{bmatrix}
\]

(52)

where \( \dot{\tilde{z}}(t) = r(t) - y(t) \) and \( r(t) \) is the reference signal of the system.

We use the following matrices for the penalty function:

\[
Q =
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.01 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 00000
\end{pmatrix}
\]

(53)

\[
R = 10
\]

The penalty matrices’s values were found in experimental ways, taking the importance of reference following into account.

Solving the Ricatti-equation with MATLAB (command \texttt{lqr}) we obtain the following value for K:

\[
K = [1.168 \quad -1.219 \quad 7.060 \quad 0.287 \quad 1.722 \quad -0.000 \quad -0.006 \quad -0.000 \quad -100.000]
\]
4.2.1 Simulation results of the SISO LQ controller

In this case (used as reference case) no disturbances were used. In the following figure we can see the reference signal and the output of the system. On the figures the real (decentered) values of the variables are shown.

![Figure 21: The output and the reference signal](image)

In this case the reference signal was \( r(t) = \frac{\pi}{2} + \frac{\pi}{16} \sin(3t) \)

The muscle activation states \( (q_1, q_2) \) and the muscle activation signals (or actuation signals - \( u_1, u_2 \)) are depicted in the following figures:

![Figure 22: Muscle activation states and muscle activation signals](image)

If we use the same reference signal, and the following matrices for the penalty function,
\[ Q = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1000 \end{pmatrix} \] (54)

we get the results depicted on the following figures:

![Graph](image)

Figure 23: The output, the reference signal and the muscle activations

We can see, that the closed loop system follows the reference signal with more delay. The appearance of the delay in the trajectory following can be explained with the system’s relative degree (3). In other words between the input and the output 3 integrators can be found.
4.3 Linear MISO LQ regulator

We are able to use both of the muscle activity signals as input, and extend the LQ-servo controller to the MISO (multiple input, single output) system, for more efficiency and robustness with the same linearization method as in section 4.1.

We use the same Q as in equation (54) for the penalty function, and:

\[ R = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix} \]

Solving the Ricatti-equation with MATLAB, we get the following value for K:

\[ K = \begin{bmatrix} 0.554 & -0.498 & 3.924 & 0.137 & 0.702 & -0.011 & -0.003 & 0 & -62.934 \\ -0.632 & 0.730 & -4.885 & -0.174 & -0.942 & -0.008 & 0.003 & 0 & 77.713 \end{bmatrix} \]

4.3.1 Simulation results of the MISO LQ controller without disturbance

In this case the reference signal was the same as above -

\[ r(t) = \frac{\pi}{2} + \frac{\pi}{16} \sin(3t) \]

In the following figure we can see the reference signal and the output of the system:
Figure 24: The output and the reference signal by MISO LQ control

The muscle activation states, and activation signals are shown in the figures below, as functions of time:

Figure 25: Muscle activation states, and activation signals

We can see that if the control uses both of the muscles, the result is more accurate trajectory following.
4.3.2 Trajectory following results of the MISO LQ controller with disturbance

If we use the same LQ servo-regulator and the same reference signal as in the previous subsection (4.3.1), and apply the following disturbances:

- 1 kg mass on the end of the limb (twice as far from the joint as the mass centre of the moving limb part)
- Fatigue - Drift of $\tau_{\text{act}}$: $\tau_{\text{act}}(t) = (1 + 0.5t)\tau_{\text{act}}$

then we get the results shown in the following figures:

![Figure 26: The output and the reference signal](image1)

![Figure 27: Muscle activation states, and activation signals](image2)
4.3.3 Regulation results of the MISO LQ controller with disturbance

If the reference signal is constant, we apply $d_1(t)$ and $d_2(t)$ for generating spontaneous muscle activation for disturbance. This method can be considered, for example, as a simple model of the effect of the Parkinson’s disease.

\[
d_1(t) = |1.5 \times \sin(14 \times t) \times \sin(16 \times t)|
\]
\[
d_2(t) = |1.5 \times \sin(15 \times t) \times \sin(17 \times t)|
\]

The results are depicted in the following figures:

Figure 28: The output and the reference signal

Figure 29: Muscle activation states, and activation signals
4.4 Input-Output linearization

In this section we use only the flexor muscle’s activation signal as input to the system (the extensor muscle’s activation signal is identically zero), to get a SISO (single input single output) structure.

As described in [9] for a nonlinear n-dimensional SISO system with relative degree \( r < n \) we need to apply the feedback

\[
u = \frac{1}{L_\phi L_{f \phi}^{-1} h(x)}(-L_f^r h(x) + v(t)) \tag{56}
\]

and a suitable nonlinear co-ordinate transformation to obtain:

- a linear subsystem of order \( r \) which is influenced by the chosen input \( u \) - including the external input \( v(t) \) and
- a nonlinear subsystem described by the zero dynamics

This means, that the state-space model of the input-output linearized closed loop system is the following in the new co-ordinates:

\[
\begin{align*}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\quad \vdots \\
\dot{z}_{r-1} &= z_r \\
\dot{z}_r &= v \quad y = z_1
\end{align*}
\tag{57}
\]

where \( L_f h(x) \) denotes the Lie-derivative of \( h(x) \) along \( f \).

In our case \( r = 3 \), so a three dimensional subsystem will be linear in the new coordinates \( z_1, z_2, z_3 \). \( z_1, z_2, z_3 \) can be determined by using the co-ordinate-transformation \( z_i = L_{fi}^{-1} h(x) \).

The remaining part is described by the zero dynamics, which has to be stable to apply the control. In section 3.6 we have investigated the stability of the zero dynamics and have found that is stable.

After input-output linearization (IOL) we can apply any kind of the linear controllers, for example a pole-placement control (PP) for \( v(t) \) as the new input, with the correction of the reference signal \( r(t) \) as described in [17]. The signal flow diagram of the pole-placement servo controller applied to the input-output linearized system is depicted in figure 30. If we choose the poles to \( P_{1,2,3} = -50 \), we get the the result \( K = [125000\ 7500\ 150] \), \( Nx = [1; 0; 0] \), \( Nu = 0 \).
If we use poles with lower magnitude, the reference following will be less accurate (more delay appears), and if we use greater ones, the actuation signal will be larger. This contradicts to the control requirements described in section 2.3.4 (the value of the actuation signal can be only in the range of \([0, 1]\)).

Figure 30: Input-output linearization and pole placement control with reference signal
4.4.1 Simulation results of the IOL-PP controller without disturbance

If we apply the control solutions described above, we get the following results:

![Image](image1.png)

Figure 31: $\alpha$ and the reference signal by input-output linearization control with pole-placement

![Image](image2.png)

Figure 32: Muscle activation states, and activation signals
4.4.2 Trajectory following results of the IOL-PP controller with disturbance

If we use the same disturbances as in section 4.3.2 and the same reference signal with the poles $P_{1,2,3} = -50$ we get the following results:

![Graph showing trajectory following results with reference signal and disturbance](image)

Figure 33: $\alpha$ and the reference signal by input-output linearization control with pole-placement

![Graph showing muscle activation states and activation signals](image)

Figure 34: Muscle activation states, and activation signals
4.4.3 Regulation results of the IOL-PP controller with disturbance

We use the same functions for $d_1(t)$ and $d_2(t)$ as described in subsection 4.3.3, and set the poles to $P_{1,2,3} = -50$. The results can be seen on the following two figures:

Figure 35: $\alpha$ and the reference signal by input-output linearization control and pole-placement

Figure 36: Muscle activation states, and activation signals

The activation of the extensor muscle originates only from the used disturbance.
4.5 Fuzzy Inference System

Soft-computing methods are also quite prevalent for controlling nonlinear systems [16], because of the application of these methods do not need the explicit knowledge of mathematical model of the system. We study the acceptability of a simple Fuzzy-controller for the model, with rules and membership functions designed in intuitive and experimental ways.

4.5.1 Basic concepts in fuzzy theory

At first, we define the basic concepts used in the theory of fuzzy systems, and fuzzy control.

Fuzzy set:

The fuzzy set $A$ defines on a set $X$ the degree of belonging to $A$ for the elements $x \in X$. $A$ can be defined with the pair $(X, \mu_A)$, where $\mu_A : X \rightarrow [0, 1]$ is the membership function of $A$.

The main difference between the fuzzy and the 'conventional' set theory is the continual membership function. In 'conventional' set theory an element is either member of the set, or it is not. In fuzzy set theory for example a value of a temperature-variable can be the member of the warm set with 0.5 degree.

Operations on fuzzy sets:

In our case we use the following operations:

- Union: $\mu_{A \cup B}(x) = max(\mu_A(x), \mu_B(x))$
- Intersection: $\mu_{A \cap B}(x) = min(\mu_A(x), \mu_B(x))$
- Complement: $\mu_{\overline{A}} = 1 - \mu_A(x)$

Direct product of fuzzy sets:

Let $A_i$ be a fuzzy set on the set $X_i$ with the membership function $\mu_{A_i}(x_i)$, $A_i = (X_i, \mu_{A_i})$, $i = 1, 2, ..., n$. The membership function of the fuzzy direct product $A_1 \times A_2 \times ... \times A_n$ is defined by

$$\mu_{A_1 \times A_2 \times ... \times A_n}(x_1, x_2, ..., x_n) = \mu_{A_1}(x_1) \land \mu_{A_2}(x_2) \land ... \land \mu_{A_n}(x_n)$$

where $\land$ is a suitable T-norm, for example $min$ (see more in [16]).
Fuzzy relation:

\[ \mu_R : X_1 \times X_2 \times \ldots \times X_n \rightarrow [0,1] \]

R fuzzy relation \(\leftrightarrow (X_1 \times X_2 \times \ldots \times X_n, \mu_R)\)

where \(\mu_R(x_1, x_2, \ldots, x_n)\) defines the possibility of the elements \(x_1, x_2, \ldots, x_n\) being in relation \(R\).

Cylindrical extension:

If the fuzzy relation \(R\) is defined on \(X_{i_1} \times \ldots \times X_{i_r}\) where \((i_1, \ldots, i_r) \subset 1, \ldots, n\), then the cylindrical extension of \(R\) to the set \(X_1 \times \ldots \times X_n\) is

\[ \mu_{ce(R)}(x_1, \ldots, x_n) = \mu_R(x_{i_1}, \ldots, x_{i_r}) \]

Projection:

If \(R\) is a fuzzy relation defined on \(X_1 \times \ldots \times X_n\), the projection of the relation \(R\) to the space \(X_{i_1} \times \ldots \times X_{i_r}\) is a relation \(P = Proj_{X_{i_1} \times \ldots \times X_{i_r}}\), which relation’s membership function is with the notation \(\{j_1, \ldots, j_{n-r}\} = \{1, \ldots, n\}/\{i_1, \ldots, i_r\}\)

\[ \mu_P(x_{i_1}, \ldots, x_{i_r}) = \sup_{x_{j_1}, \ldots, x_{j_{n-r}}} \mu_R(x_{j_1}, \ldots, x_{j_{n-r}}) \]

Fuzzy composition:

Let it be

\[ X_1 \times \ldots \times X_{m-1} \times X_{m-1} \times \ldots \times X_r \] the set of the fuzzy relation \(R\)

\[ X_m \times \ldots \times X_r \times X_{r+1} \times \ldots \times X_n \] the set of the fuzzy relation \(S\)

\[ X_1 \times \ldots \times X_{m-1} \times X_{r+1} \times \ldots \times X_n \] the set of the fuzzy relation \(R \circ S\)

We get the fuzzy composition with cylindrical extending (ce) both of the relations to the \(X_1 \times \ldots \times X_n\) product-space, take the intersection of the relations, and project the result to the \(X_1 \times \ldots \times X_{m-1} \times X_{r+1} \times \ldots \times X_n\) product-space.

\[ R \circ S = Proj_{X_1 \times \ldots \times X_{m-1} \times X_{r+1} \times \ldots \times X_n} [ce(R) \cap ce(S)] \]

\[ \mu_{R \circ S}(x_1, \ldots, x_{m-1}, x_r, \ldots, x_n) = \sup_{x_m, \ldots, x_r} (\mu_R(x_1, \ldots, x_r) \wedge \mu_S(x_m, \ldots, x_n)) \]

Fuzzy composition is important in fuzzy controller design, because (as we will later see) this method is the connection between the rules of a controller, and the input data.
Fuzzy rule base:
We use Mamdani-implication: \[ a \rightarrow b \approx a \cdot b \]

Implication: \( R \): If \( x \) is \( A \), then \( y \) is \( B \). \( R_i = A \rightarrow B \) is a relation over \( X \times Y \).

We have to note that this method is not compatible with the Bool-algebra, but it can be easily computed, and it is used in the most case in the practice.

The projection of the relation \( R = A \rightarrow B \) to the \( Y \) fuzzy set gives the composition rule of the fuzzy implication:

\[
\text{Proj}_Y(R) = \text{Proj}_Y(A \rightarrow B) = A \circ B
\]

where \( \circ \) is the fuzzy composition.

The general form of the fuzzy implication can be described with the \( R_1, R_2, \ldots, R_n \) relations \( (R = \cup R_i) \) of the rule-base.

\[
R_i: \text{if } x_1 \text{ is } X_1^i \text{ and } \ldots \text{ and } x_n \text{ is } X_n^i \text{ then } y \text{ is } Y^i
\]

In the case of fuzzy control, the \( R = \cup R_i \) rule base (the summation of the if...then... implications) from the linguistical statements of the professional knowledge.

Matching data to a relation:
Let \( x_j^* \) the measured value of the variable \( x_j \). Let \( x_j^* \) match the \( A_j \) fuzzy set. In the case of \( R_i \), let the fuzzy set \( A_j \) be defined on the set \( X_j^i \).

In this case

The cylindrical extended values of the input data:

\[
D = \text{ce}(A_1) \cup \ldots \cup \text{ce}(A_n), \quad \mu_D(x_1, \ldots, x_n, y) = \mu_{A_1}(x_1) \land \ldots \land \mu_{A_n}(x_n)
\]

The fuzzy form of the \( R_i \) Mamdani implication (as a relation):

\[
R_i = \text{ce}(X_1^i) \cup \ldots \cup \text{ce}(X_n^i) \cup \text{ce}(Y^i),
\]

\[
\mu_{R_i}(x_1, \ldots, x_n, y) = \mu_{X_1^i}(x_1) \land \ldots \land \mu_{X_n^i}(x_n) \land \mu_{Y^i}(y)
\]
In the general case of a fuzzy controller $x$ could be, for example, the difference between the reference signal, and the output of the system, and $y$ could be the actuation signal.

The composition rule of the fuzzy implication (using that $ce(A_j) \cap ce(B_j) = ce(A_j \cap B_j)$ in the case of identical sets):

\[
D \circ R_i = Proj_Y(D \cap R_i)
\]

\[
\mu_{D \circ R_i}(y) = \sup_{x_1, \ldots, x_N} \mu_D(x_1, \ldots, x_N, y) \land \mu_{R_i}(x_1, \ldots, x_N, y) =
\]

\[
\sup_{x_1, \ldots, x_N} [\mu_{A_1}(x_1) \land \ldots \land \mu_{A_N}(x_N)] \land [\mu_{X_1^i}(x_1) \land \ldots \land \mu_{X_N^i}(x_N) \land \mu_{Y^i}(y)] =
\]

\[
\sup_{x_1, \ldots, x_N} [\mu_{A_1}(x_1) \land \mu_{X_1^i}(x_1)] \land \ldots \land [\mu_{A_N}(x_N) \land \mu_{X_N^i}(x_N)] \land \mu_{Y^i}(y) =
\]

we can define the

\[
\tau_{ij} = \sup_{x_j} \mu_{A_j}(x_j) \land \mu_{X_j^i}(x_j)
\]

\[
\tau_i = \tau_{i1} \land \ldots \land \tau_{iN}
\]

firing rates, so

\[
\mu_{D \circ R_i}(y) = \tau_y \land \mu_{Y^i}(y)
\]

Figure 37: The illustration of the rule If $x_1$ is $x_1^i$ and $x_2$ is $x_2^i$ then $y$ is $y^i$
The Mamdani min-max implication algorithm:

1. Evaluate $\tau_{ij} j = 1, ..., N$ firing rates, and $\tau_i$ for all $R_i$ relation
   \[
   \tau_{ij} = \mu_{X_j^i}(x_j^*) \quad \tau_i = \min_j(\tau_{ij})
   \]

2. Evaluate $\mu_{D\circ R_i}$ for all $y$
   \[
   \mu_{D\circ R_i}(y) = \begin{cases} 
   \mu_{Y_i^i}(y) & \text{if } \mu_{Y_i^i}(y) < \tau_i \\
   \tau_i & \text{otherwise}
   \end{cases}
   \]

3. Evaluate the output of the fuzzy system for all $y$
   \[
   \mu_{D\circ R}(y) = \max_i(\mu_{D\circ R_i}(y))
   \]
4.5.2 The applied fuzzy controller

We used the Mamdani min-max implication algorithm described above. We used the following variables as input for the fuzzy inference system (weighted with constants):

- The difference between the reference signal and the output:
  \[ r(t) - x_3(t) = r(t) - \alpha(t) \]
- The joint angle velocity: \( x_4 = \omega(t) \)
- The flexor muscle’s activation state: \( x_1(t) = q_1(t) \)
- The extensor muscle’s activation state: \( x_2(t) = q_2(t) \)

Where \( x_1, x_2, x_3 \) and \( x_4 \) denotes the state-space variables of the model (not the input of the fuzzy inference system).

The output of the fuzzy inference system were the muscle activity signals (weighted with constants).

We used the following membership functions as input (\( r(t) - \alpha \) was weighted with 10):

![Figure 38: The membership functions of \( r(t) - \alpha \)](image)

66
We used the following membership functions as output. Outputs were weighted with 0.6:
We used the following rules:

- If $r - \alpha$ is $ps$, then $u_1$ is 5 and $u_2$ is 1
- If $r - \alpha$ is $ns$, then $u_1$ is 1 and $u_2$ is 3
- If $r - \alpha$ is $pb$, then $u_1$ is 6 and $u_2$ is 1
- If $r - \alpha$ is $nb$, then $u_1$ is 1 and $u_2$ is 4
- If $r - \alpha$ is not $nb$ and $\omega$ is $pb$, then $u_1$ is 1 and $u_2$ is 6
- If $r - \alpha$ is not $pb$ and $\omega$ is $nb$, then $u_1$ is 7 and $u_2$ is 1
- If $r - \alpha$ is $z$ and $\omega$ is $ps$, then $u_1$ is 3 and $u_2$ is 5
- If $r - \alpha$ is $z$ and $\omega$ is $ns$, then $u_1$ is 3 and $u_2$ is 1
\begin{itemize}
\item If $r - \alpha$ is not $pb$ and $q_1$ is big, then $u_1$ is 2 and $u_2$ is 4
\item If $r - \alpha$ is not $nb$ and $q_2$ is big, then $u_1$ is 5 and $u_2$ is 2
\end{itemize}

4.5.3 Simulation results of the fuzzy controller without disturbance

If we use the same reference signal as in 4.3.1, and apply the fuzzy controller described above, we get the following results:

![Graph 1](image1)

Figure 44: The output $\alpha$ and the reference signal by fuzzy control

![Graph 2](image2)

Figure 45: Muscle activation states, and activation signals
4.5.4 Trajectory following results of the Fuzzy controller with disturbance

In the next two figures, we can see the effect of disturbance to the fuzzy control system. The applied disturbance was the same as in subsections 4.3.2 and 4.4.2 with the difference, that the external mass was only 0.333 kg.

![Figure 46: The output α and the reference signal](image1)

We can see, that the fuzzy control system is more sensitive to disturbances.

![Figure 47: Muscle activation states, and activation signals](image2)
4.5.5 Regulation results of the Fuzzy controller with disturbance

The applied disturbance, the spontaneous muscle activity, was the same as in subsections 4.3.3 and 4.4.3.

Figure 48: The output $\alpha$ and the reference signal

Figure 49: Muscle activation states, and activation signals
4.6 Comparison of the different control methods

We compared the results of the MISO LQ regulator designed for the locally linearized model, the IOL-PP control, and the fuzzy control. The main comparison viewpoints were accuracy, output signal smoothness and disturbance sensitivity.

4.6.1 Comparison of the results without disturbance

If we compare the trajectory following of the different control methods in ideal environment, we get the following results:

![Figure 50: Trajectory following with lin. MISO LQ and IOL-PP control](image1)

![Figure 51: Trajectory following with fuzzy control](image2)
4.6.2 Comparison of the results with disturbance

Let us compare the results of the most difficult task (the trajectory following with disturbance) for the three investigated controllers, depicted in the figures below:

Figure 52: Trajectory following with lin. MISO LQ and IOL-PP control

Figure 53: Trajectory following with fuzzy control
4.6.3 Comparison results: a discussion

In the figures 50 and 51 we can see, that in ideal circumstances (i.e. without disturbances) all of the the LQ regulator, the IOP-PP and the fuzzy controller provides acceptable output, that is within the tolerance limit. The LQ regulator and the IOP-PP provides more smooth output than the fuzzy controller, but we have to note, that the fuzzy controller requires only 4 state-space variables for computing the feedback (the other methods require all 8 state-space variables for the feedback), and does not require the mathematical equations of the model. On the other hand, the required computing capacity is the highest the case of the fuzzy controller.

In the figures 52 and 53 we can observe, that the input-output linearized system is more stable in the case of the appearing disturbances, and provides a more smooth and more accurate output signal, than the other solutions. We have to note, that the fuzzy controller was tested mainly in ideal circumstances, and designed mainly for trajectory following. Probably the tuning of the membership functions and rules would lead to more robustness in an environment accused with disturbances in both cases of the control tasks.

Moreover, we can note, that in some cases of the fuzzy and the MISO-LQ control of the locally linearized system, the control uses both of the muscles in the same time. This method (using both muscles at the same time) is not regarded to be the optimal, if we consider, for example, the ATP consumption in the muscles. Because of the extensor muscle’s activity the flexor muscle has to work on a higher activity level. Still this method of the linear control provides more robustness (compared to the SISO-LQ control methods).
5 Conclusions and future work

We have seen that the methods of nonlinear systems and control theory can be successfully applied to the investigated simple limb model. The problem that we had to solve in the case of applying nonlinear control (input-output linearization) was the complexity of the functions needed for the linearizing feedback. In fact, both of the control tasks (trajectory following and regulation) can be better solved with nonlinear control methods, as it can be clearly seen in the results of the simulation. On the other hand, if the expectations are not too high or the computing capacity is low, a linear control also can be successfully applied. If the model equations or some important parameters are not available, a fuzzy control can also be applied.

The possible tasks for the research in the future could be for example:

- Extend the state-space model for more muscles and/or joints.
- Extend the model to 3D.
- Extend the control to the system including the gamma-loop.
- Utilize the Hamiltonian properties of the system for a nonlinear PD control.
- Utilize the properties of the linearized subsystem for an adaptive fuzzy control.
- Explore the similarities and differences of the control of the model, and the physiological control of a real human limb, search the control loops in the central nervous system.
- Analyze the muscle activity states generated by the model in the case of various movement patterns, and compare them with EMG signals recorded on real patients doing the same movements, for the development of the model to a more realistic form.
- Extend the control with a simple model of adaptation motor learning, for example as an adaptation to changes in the model parameters or in the disturbance parameters.
6 Acknowledgements

I would like to say thanks for helping my work to the following people, and institutes: My supervisor Prof. Katalin Hangos, my department consultant Dr. István Harmati, Csaba Fazekas, the Systems and Control Laboratory at the Computer and Automation Research Institute Budapest, Dr. Péter Ván.
7 Appendix

7.1 The gamma-loop

![Diagram of the servomechanical γ-loop](image)

Figure 54: The servomechanical γ-loop

7.1.1 Muscle spindles

Muscle spindles are found within the fleshy portions of muscles, embedded in so-called extrafusal muscle fibers. They are composed of 3-10 intrafusal muscle fibers, of which there are two types, nuclear bag fibers and nuclear chain fibers and the axons of sensory neurons. Axons of motoneurons also terminate in muscle spindles; they make synapses at either or both of the ends of the intrafusal muscle fibers and regulate spindle sensitivity. Muscle spindles are encapsulated by connective tissue, and are aligned parallel to extrafusal muscle fibers, unlike Golgi tendon organs, which are oriented in series.
The muscle spindle has both sensory and motor components. Primary and secondary sensory fibers spiral around and synapse on the central portions of intrafusal fibers, providing the sensory component of the structure via stretch-sensitive excitatory ion-channels of the axons. The motor component is provided by a gamma motoneuron that innervates the spindle and causes a slight contraction of the end portions of the intrafusal muscle fibers when activated.

7.1.2 The servomechanical Gamma-loop

The length of the muscle spindles on the one hand serve as a sensory input for the central nervous system, providing information about the length of the muscle, and so the state of the joints too. On the other hand they serve as a component for a servomechanical $\gamma$-loop. Inside the muscle spindles are muscle fibers, thinner and shorter than the working muscle fibers, providing the most part of the force, doing the most of the muscle work. These muscle fibers inside the muscle spindles are innervated by the $\gamma$-nerves originating from the $\gamma$-neurons in the spinal marrow. In general case the $\gamma$-neurons regulate the length of the muscle spindles to match the actual length, and stretch state of the working muscle.

The servomechanical Gamma-loop effect:

If the $\gamma$-neurons are stimulated by the impulses from the descending tracts in the spinal marrow, the muscle fibers inside the muscle spindle contract, providing difference between the stretch state of the working muscle and the muscle spindles, which is instantly detected by the receptors of the muscle spindle. Consequently the motoneurons in the spinal marrow stimulate the working muscle to contract. Because of the impulses from the descending tracts through the $\gamma$-neurons first reach the muscle spindle, then return to the spinal marrow, and just then reach the working muscle, we call this loop-like path $\gamma$-loop. The consequence is, that the central nervous system does not need nerves descending to all of the muscle fibers, furthermore the impulses of the central nervous system can affect the activation of the muscle in a way, that the final effect depends on the actual state (length) of the muscle. [23]
References


