Abstract

In this thesis, two different control strategies are applied to the forward dynamic simulation of multibody systems in order to track a given reference motion. For this purpose, two different computational models are presented: a four-bar linkage model with one degree of freedom; and a two-dimensional human body model that consists of 12 segments with 14 degrees of freedom. The forward dynamic analysis of the two models is implemented using the matrix-R formulation and carried out by means of a variable-step integration solver. Furthermore, an analysis and comparison of different numerical integration methods are carried out. The joint forces and torques, which are applied to the multibody systems in order to drive their motion, are provided through an inverse dynamic analysis. In order to stabilize the simulation and to enable the tracking of a reference motion, two control methods are introduced: a proportional derivative control and a computed torque control using feedback linearization. The design of both control approaches is developed and applied to the forward dynamic simulation of both models. The system performance is evaluated by comparing the results with the reference motion. The reference human motion of a healthy subject was captured previously in a biomechanics laboratory. Moreover, the robustness of the computed torque control approach is analysed. In addition, environmental and social impacts of this thesis are outlined and an economical consideration is included.
Application of different Control Strategies to the FD Simulation of Human Gait
Contents

List of Figures V
List of Tables VII
Symbols and Abbreviations IX

1 Introduction 1
   1.1 Motivation, Scope and Objectives ................................. 1
   1.2 Thesis Contents ................................................. 2

2 State of the Art 3

3 Multibody Dynamics 7
   3.1 Biomechanics .................................................... 7
   3.2 Model Applications ............................................. 8
      3.2.1 Four-bar Linkage Model .................................. 9
      3.2.2 2D Biomechanical Human Body Model ...................... 10
   3.3 Multibody Formulation. Kinematic Analysis ..................... 12
      3.3.1 Constraint Equations ..................................... 15
   3.4 Multibody Formulation. Dynamic Analysis ...................... 16
      3.4.1 Mass Matrix of Planar Bodies ............................... 17
      3.4.2 Generalized Force Vector ................................ 20
      3.4.3 Solution of the Equations of Motion ...................... 21
      3.4.4 Integration of the Equations of Motion ................... 25

4 Inverse Dynamic Analysis 29
   4.1 IDA applied to Four-Bar Linkage Model ......................... 30
   4.2 IDA applied to 2D Human Body Model ............................ 32
      4.2.1 Experimental Setup ....................................... 35
      4.2.2 Motion Reconstruction .................................... 36
      4.2.3 Body Segment Parameters of the Human Body Model ...... 38
      4.2.4 Driving Torques of the Human Body Model ................ 40

5 Forward Dynamic Analysis 41
   5.1 FDA applied to Four-Bar Linkage Model ......................... 43
   5.2 FDA applied to 2D Human Body Model ............................ 48
IV Application of different Control Strategies to the FD Simulation of Human Gait

6 Control Strategies 51
6.1 PD Control ........................................... 53
   6.1.1 PD Control applied to the FDA of the Four-Bar Linkage Model ... 55
   6.1.2 PD Control applied to the FDA of the 2D-Human Body Model ... 57
6.2 Computed Torque Control with Feedback Linearization .......... 59
   6.2.1 Feedback Linearization ........................................ 59
   6.2.2 Computed Torque Control applied to Multibody Systems ..... 62
   6.2.3 Computed Torque Control Design for Multibody Systems ..... 63
   6.2.4 CTC Control applied to the FDA of the Four-Bar Linkage Model . 67
   6.2.5 CTC Control applied to the FDA of the 2D-Human Body Model . 68

7 System Performance and Control Robustness 73
   7.1 Comparison of PD and CTC Control for Four-Bar Linkage Model .... 73
   7.2 Robustness of CTC Control ........................................ 76

8 Environmental and Social Impacts and Economic Consideration 81
   8.1 Environmental Impact ........................................... 81
   8.2 Social Impact ..................................................... 81
   8.3 Economic Consideration .......................................... 81

9 Conclusions 83
   9.1 Future Work ..................................................... 84

Bibliography XIII
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Motion prediction: From the beginning to recent research.</td>
<td>4</td>
</tr>
<tr>
<td>3.1</td>
<td>Human body planes and directions [46].</td>
<td>7</td>
</tr>
<tr>
<td>3.2</td>
<td>Human gait cycle classified in phases and periods [39].</td>
<td>8</td>
</tr>
<tr>
<td>3.3</td>
<td>Four-bar linkage model.</td>
<td>9</td>
</tr>
<tr>
<td>3.4</td>
<td>2D biomechanical model of the human body.</td>
<td>10</td>
</tr>
<tr>
<td>3.5</td>
<td>Defined configuration of the planar model.</td>
<td>11</td>
</tr>
<tr>
<td>3.6</td>
<td>Angular variable between two segments linked by a revolute joint.</td>
<td>15</td>
</tr>
<tr>
<td>3.7</td>
<td>Representation of a generic rigid body using natural coordinates.</td>
<td>18</td>
</tr>
<tr>
<td>3.8</td>
<td>Assembly of the element mass matrices into the global mass matrix [32].</td>
<td>19</td>
</tr>
<tr>
<td>3.9</td>
<td>Generic concentrated force applied on a planar rigid body.</td>
<td>20</td>
</tr>
<tr>
<td>3.10</td>
<td>Length of first segment of four-bar linkage model using the method of Lagrange multipliers and the stabilized Lagrange method.</td>
<td>22</td>
</tr>
<tr>
<td>3.11</td>
<td>Scheme of the iterative procedure to integrate the equations of motion with the trapezoidal rule [32].</td>
<td>26</td>
</tr>
<tr>
<td>4.1</td>
<td>Block diagram of an inverse dynamic analysis.</td>
<td>29</td>
</tr>
<tr>
<td>4.2</td>
<td>Defined angle motion $\alpha(t)$ of the four-bar linkage model and its derivatives $\dot{\alpha}(t)$ and $\ddot{\alpha}(t)$ with $\omega = \frac{\pi}{2} \text{ rad s}^{-1}$.</td>
<td>31</td>
</tr>
<tr>
<td>4.3</td>
<td>Comparison of the driving torques using different dynamic solving methods with the driving torque achieved by means of the Lagrangian multipliers method.</td>
<td>32</td>
</tr>
<tr>
<td>4.4</td>
<td>Gait analysis laboratory configuration [31].</td>
<td>35</td>
</tr>
<tr>
<td>4.5</td>
<td>Phases of one gait cycle captured in the laboratory [31].</td>
<td>36</td>
</tr>
<tr>
<td>4.6</td>
<td>3D view of the human skeleton with the set of 37 markers used [32].</td>
<td>37</td>
</tr>
<tr>
<td>4.7</td>
<td>Reference motion of the lower limb.</td>
<td>39</td>
</tr>
<tr>
<td>4.8</td>
<td>Anthropometric measurements of the lower extremities [32].</td>
<td>39</td>
</tr>
<tr>
<td>4.9</td>
<td>Lower limb joint forces and torques.</td>
<td>40</td>
</tr>
<tr>
<td>5.1</td>
<td>Ideal closed loop behaviour of IDA and FDA. Using the IDA's output, the driving forces and torques $Q_{\text{IDA}}$ as an input for the FDA, the FDA's output, the motion $z$ should be coincident with the reference input of the IDA.</td>
<td>42</td>
</tr>
<tr>
<td>5.2</td>
<td>Block diagram of forward dynamic analysis.</td>
<td>42</td>
</tr>
<tr>
<td>5.3</td>
<td>FDA results using the variable-step solver ode23 compared with the reference angle motion $\alpha(t)$ (driving torque $Q_{\text{IDA}}$ linear interpolated, simulation time steps $\Delta t$: 10ms and 1ms).</td>
<td>44</td>
</tr>
</tbody>
</table>
5.4 FDA results using the fixed-step solver ode4 compared with the reference angle motion $\alpha(t)$ (time steps $\Delta t$: 10ms and 1ms). .......................... 45
5.5 FDA results using the fixed-step solver trapezoidal rule compared with the reference angle motion $\alpha(t)$ (time steps $\Delta t$: 10ms and 1ms). . . . 45
5.6 FDA results using the variable-step solver ode23 ($Q_{IDA}$ interpolated with a B-spline curve) and the fixed-step solvers trapezoidal rule and ode4 compared with the reference angle motion $\alpha(t)$ (simulation time step: $\Delta t = 1$ms). ................................................................. 46
5.7 FDA results using the variable-step solver ode23 ($Q_{IDA}$ interpolated with B-spline curve) compared with reference angular velocity $\dot{\alpha}(t)$ and the reference angular acceleration $\ddot{\alpha}(t)$ (simulation time step: $\Delta t = 1$ms). . . . 47
5.8 FDA results for human body model of ankle, knee and hip flexion angles compared to the reference angles. .......................................................... 49

6.1 Block diagram of a closed-loop system with output feedback. ......... 51
6.2 Block diagram of the closed-loop nonlinear MIMO system using PD control. .......................... 54
6.3 ........................................................................................................................................ 56
6.4 FDA results of four-bar linkage model using PD control. ................. 58
6.5 PD control torque of four-bar linkage model. ................................. 58
6.6 Closed loop system of a nonlinear plant using feedback linearization. . 62
6.7 Closed loop system of a multi-body system using feedback linearization. . 63
6.8 Linear control approach of the linearized nonlinear MIMO system using feedback linearization. .......................... 66
6.9 Tracking error of $\alpha$ for forwards dynamic simulation of four-bar linkage model with a wrong initial state vector using different CTC control approaches. ................................................................. 67
6.10 FDA results of four-bar linkage model using CTC control. ............... 69
6.11 CTC control torque of four-bar linkage model. ....................... 69
6.12 Tracking errors of the independent coordinates of the human body model using CTC control. .......................................................... 70
6.13 CTC control forces and torques applied to the human body model. .... 71

7.1 Motion of forward dynamic simulation of four-bar linkage model introducing a mass fault to the system at 1s (without control). .......... 74
7.2 Tracking error of $\alpha$ for the forward dynamic simulation of four-bar linkage model with mass error introduced to the system at 1s. .......... 75
7.3 Motion of forward dynamic simulation of four-bar linkage model introducing a torque fault to the system at 2s (without control). .......... 75
7.4 Tracking error of $\alpha$ and its derivatives for the forward dynamic simulation of four-bar linkage model with torque error introduced to the system at 2s. 76
7.5 Independent coordinates in the forward dynamic simulation of the human body model with a mass fault of 2.5% introduced to the system from the beginning compared to the reference motion. ............... 79
List of Tables

3.1 Parameters of the four-bar linkage model. ......................................... 9
3.2 Anthropometric data for the 12 segments of the 2D model (adapted from [31]). .......................................................... 11
4.1 Placement of the set of markers used [32]. ....................................... 38
5.1 Comparison of NRMSE and simulation time dependent on integration method and time step used for the forward dynamic analysis of the four-bar linkage model. ................................................... 47
6.1 Comparison of NRMSE, settling time and simulation time dependent on adjusted control parameters $K_p$ and $T_d$ of PD controller (using wrong initial values). .................................................. 57
6.2 Comparison of NRMSE, settling time and simulation time dependent on adjusted control parameter $k_p$ of CTC controller (using wrong initial values). .................................................. 68
7.1 Comparison of both control strategies PD and CTC in the forward dynamic simulation of the four-bar linkage model. ......................... 74
7.2 Robustness analysis of CTC control approach in the forward dynamic simulation of the four-bar linkage model. Fault cases are introduced from the beginning (t=0s). ............................................. 78
8.1 Project costs. .................................................................................... 82
8.2 Calculation of the final project cost. .................................................. 82
Application of different Control Strategies to the FD Simulation of Human Gait
Symbols and Abbreviations

Greek Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha )</td>
<td>angle of first segment of the four-bar linkage model</td>
</tr>
<tr>
<td>( \dot{\alpha} )</td>
<td>angular velocity</td>
</tr>
<tr>
<td>( \ddot{\alpha} )</td>
<td>angular acceleration</td>
</tr>
<tr>
<td>( \alpha_i )</td>
<td>relative orientation between two segments in human body model</td>
</tr>
<tr>
<td>( \alpha_k )</td>
<td>penalty factor</td>
</tr>
<tr>
<td>( \epsilon )</td>
<td>specified error tolerance</td>
</tr>
<tr>
<td>( \varphi )</td>
<td>angular coordinate between two linked segments</td>
</tr>
<tr>
<td>( \lambda )</td>
<td>vector of Lagrange multipliers</td>
</tr>
<tr>
<td>( \mu_k )</td>
<td>damping ratio in penalty formulation</td>
</tr>
<tr>
<td>( \omega )</td>
<td>constant of stabilized Lagrange method or angular frequency defining the angle motion of the four-bar linkage model</td>
</tr>
<tr>
<td>( \zeta )</td>
<td>constant of stabilized Lagrange method</td>
</tr>
<tr>
<td>( \Phi )</td>
<td>vector of kinematic constraints</td>
</tr>
<tr>
<td>( \Phi_q )</td>
<td>Jacobian matrix of the kinematic constraints</td>
</tr>
<tr>
<td>( \dot{\Phi}_q )</td>
<td>time derivative of the Jacobian matrix ( \Phi_q )</td>
</tr>
<tr>
<td>( \Phi_t )</td>
<td>vector containing the partial derivatives of the constraints with respect to time</td>
</tr>
<tr>
<td>( \dot{\Phi}_t )</td>
<td>time derivative of ( \Phi_t )</td>
</tr>
<tr>
<td>( \Omega_k )</td>
<td>natural frequency in penalty formulation</td>
</tr>
</tbody>
</table>

Roman Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( {a, b} )</td>
<td>lengths of the unit vectors ( {u, v} )</td>
</tr>
<tr>
<td>( e )</td>
<td>tracking error</td>
</tr>
</tbody>
</table>
Application of different Control Strategies to the FD Simulation of Human Gait

\[ \dot{e} \] time derivative of tracking error \( e \)

\[ e(t) \] error signal

\[ e_z \] tracking error of the independent coordinates \( z \)

\[ \dot{e}_z \] time derivative of the tracking error \( e_z \)

\[ f(x) \] nonlinear vector field of state space vector \( x \)

\[ f(z) \] system dynamics function

\[ f_z \] Jacobian matrix of the system dynamics function

\[ g \] gravitational acceleration

\[ g(x) \] nonlinear matrix of state space vector \( x \)

\[ h \] incremental time step

\[ h(x) \] nonlinear vector field of state space vector \( x \)

\( i, j, k \) generic definition of rigid body points

\[ k_p \] proportional gain of joint stiffness matrix \( K_p \)

\[ l_i \] length of a four-bar linkage segment

\( m \) number of kinematic constraints in Chapter 3 or mass of an element in Sections 3.4.1 and 3.4.2

\[ m_i \] mass of a segment

\[ m_r \] number of scleronomic constraints

\[ m_s \] number of rheonomic constraints

\( n \) number of generalized coordinates

\[ n_d \] number of dependent coordinates

\[ n_i \] number of independent coordinates

\[ q \] vector of generalized coordinates

\[ \dot{q} \] vector of generalized velocities

\[ \ddot{q} \] vector of generalized accelerations

\[ \bar{q} \] planar generalized coordinates

\[ q^d \] vector of dependent coordinates

\[ q^i \] vector of independent coordinates

\[ q_0 \] vector of approximate initial generalized coordinates at a time instance

\[ \dot{q}_e \] vector of coordinates associated with the element \( e \)

\[ r(t) \] reference signal

\[ r_k \] relative degree of the \( k^{th} \) output

\[ r_G \] position of the COM of a segment in the local coordinate system

\[ r_P \] position of point \( P \) expressed in global coordinate system

\[ \dot{r}_P \] velocity point \( P \)

\[ \bar{r}_P \] position of point \( P \) expressed in the segment local coordinate system
### Symbols and Abbreviations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s$</td>
<td>Laplace operator</td>
</tr>
<tr>
<td>$t$</td>
<td>time variable</td>
</tr>
<tr>
<td>$t_d$</td>
<td>derivative time of derivative time matrix $T_d$</td>
</tr>
<tr>
<td>$t_{1%}$</td>
<td>settling time</td>
</tr>
<tr>
<td>$u$</td>
<td>input vector for the plant (combined with actuator)</td>
</tr>
<tr>
<td>$v$</td>
<td>synthetic input vector of CTC control</td>
</tr>
<tr>
<td>$v(t)$</td>
<td>output signal of the actuator</td>
</tr>
<tr>
<td>${u, v}$</td>
<td>unit vectors used to define the local basis of segment in Section 3.4.2</td>
</tr>
<tr>
<td>$x$</td>
<td>sensor measurement signal or state space vector</td>
</tr>
<tr>
<td>$y$</td>
<td>output vector of the plant</td>
</tr>
<tr>
<td>$z$</td>
<td>vector of independent coordinates</td>
</tr>
<tr>
<td>$\dot{z}$</td>
<td>vector of independent velocities</td>
</tr>
<tr>
<td>$\ddot{z}$</td>
<td>vector of independent accelerations</td>
</tr>
<tr>
<td>$A$</td>
<td>rotation matrix in Section 3.4.1 or system matrix of state space representation in Section 6.2.3</td>
</tr>
<tr>
<td>$A_K$</td>
<td>system matrix of controlled system</td>
</tr>
<tr>
<td>$B$</td>
<td>input matrix of state space representation</td>
</tr>
<tr>
<td>$B(x)$</td>
<td>input-output-decoupling matrix</td>
</tr>
<tr>
<td>$CO$</td>
<td>controllability matrix</td>
</tr>
<tr>
<td>$F$</td>
<td>generic force vector</td>
</tr>
<tr>
<td>$G$</td>
<td>centre of mass of a segment</td>
</tr>
<tr>
<td>$I$</td>
<td>identity matrix</td>
</tr>
<tr>
<td>$I_g$</td>
<td>identity matrix ($g \times g$)</td>
</tr>
<tr>
<td>$I_G$</td>
<td>moment of inertia with respect to the COM</td>
</tr>
<tr>
<td>$K$</td>
<td>total number of time steps</td>
</tr>
<tr>
<td>$K_d$</td>
<td>damping matrix</td>
</tr>
<tr>
<td>$K_p$</td>
<td>joint stiffness matrix</td>
</tr>
<tr>
<td>$L$</td>
<td>length of a segment</td>
</tr>
<tr>
<td>$L_f, h, L_g, h$</td>
<td>Lie derivatives</td>
</tr>
<tr>
<td>$L_i$</td>
<td>length of a human body segment</td>
</tr>
<tr>
<td>$M$</td>
<td>global system mass matrix</td>
</tr>
<tr>
<td>$M_e$</td>
<td>mass matrix of the element $e$</td>
</tr>
<tr>
<td>$P$</td>
<td>generic point of an element</td>
</tr>
<tr>
<td>$Q$</td>
<td>generalized force vector</td>
</tr>
<tr>
<td>$Q_e$</td>
<td>generalized force vector of the element $e$</td>
</tr>
<tr>
<td>$Q_{in}$</td>
<td>input vector of forces and torques</td>
</tr>
</tbody>
</table>
Application of different Control Strategies to the FD Simulation of Human Gait

\( Q_C \quad \text{control forces and torques} \\
Q_{\text{IDA}} \quad \text{driving forces and torques obtained through IDA} \\
R \quad \text{transformation matrix} \\
T \quad \text{kinetic energy or total simulation time} \\
T_d \quad \text{derivative time matrix} \\
V \quad \text{error optimization criteria} \\
W \quad \text{diagonal weighting matrix} \\
\{X,Z\} \quad \text{global coordinate system} \\
\{\bar{X},\bar{Z}\} \quad \text{local coordinate system}

### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSP</td>
<td>Body Segment Parameter</td>
</tr>
<tr>
<td>CFP</td>
<td>Corrected Force Plate</td>
</tr>
<tr>
<td>COM</td>
<td>Center of Mass</td>
</tr>
<tr>
<td>CREB</td>
<td>Biomedical Engineering Research Centre</td>
</tr>
<tr>
<td>CTC</td>
<td>Computed Torque Control</td>
</tr>
<tr>
<td>DAE</td>
<td>Differential Algebraic Equation</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree of Freedom</td>
</tr>
<tr>
<td>EKF</td>
<td>Extended Kalman Filter</td>
</tr>
<tr>
<td>ETSEIB</td>
<td>School of Industrial Engineering of Barcelona</td>
</tr>
<tr>
<td>FDA</td>
<td>Forward Dynamic Analysis</td>
</tr>
<tr>
<td>FP</td>
<td>Force Plate</td>
</tr>
<tr>
<td>IDA</td>
<td>Inverse Dynamic Analysis</td>
</tr>
<tr>
<td>MIMO</td>
<td>multiple input multiple output</td>
</tr>
<tr>
<td>NRMSE</td>
<td>Normalized Root-Mean-Square Error</td>
</tr>
<tr>
<td>PD</td>
<td>Proportional Derivative</td>
</tr>
<tr>
<td>SISO</td>
<td>Single Input Single Output</td>
</tr>
<tr>
<td>SSA</td>
<td>Singular Spectrum Analysis</td>
</tr>
<tr>
<td>ODE</td>
<td>Ordinary differential equation</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root-Mean-Square Error</td>
</tr>
<tr>
<td>UPC</td>
<td>Technical University of Catalonia</td>
</tr>
<tr>
<td>SCI</td>
<td>spinal-cord-injured</td>
</tr>
<tr>
<td>SMC</td>
<td>Sliding mode control</td>
</tr>
</tbody>
</table>
1 Introduction

Modern medicine technologies require mechatronic system approaches to improve the quality of life of patients. A close collaboration between medical staff and engineers in multidisciplinary teams is necessary to ensure effective health care. Therefore, biomechanics is a very attractive area of research in mechanical engineering and of high importance to the health care field.

This work is part of a national R+D project entitled “Design of an innovative gait-assistive active orthosis for incomplete spinal cord injured subjects based on motion analysis and prediction methods and complex musculoskeletal models”, which is developed in the context of a research line on Biomechanics in the Department of Mechanical Engineering at the School of Industrial Engineering of Barcelona (ETSEIB) and the Biomedical Engineering Research Centre (CREB) of the Technical University of Catalonia (UPC).

Incomplete spinal-cord-injured (SCI) subjects suffer from constrained motor or sensory function below the injury level. A restricted mobility and less efficient locomotion increase the risk of falling and affect the security in their daily life. The gait of SCI individuals can be greatly improved by means of the use of active orthoses and exoskeletons. Those are powered robotic devices that assist the subject’s motion, augmenting his or her musculoskeletal function. These devices are useful to recover gait function, and to increase the patient’s self-esteem and quality of life.

1.1 Motivation, Scope and Objectives

The main objective of the general project is the development of a computer application which enables to virtually test different types and designs of active orthoses for gait assistance on the computational model of a disabled subject in order to predict the motion. This computer application needs to combine the model of a real patient (whose data, motion and myographic signals will have been acquired) with the active orthosis model. The simulation of the resulting motion of the patient wearing the orthosis can be extrapolated to real applications and enables human motion prediction. For this purpose, a human multibody model has been developed, tested and validated.

Due to the unstable character of human gait in the Forward Dynamic Analysis (FDA) approach, control modules are of particular importance to ensure stability and robust-
ness in human gait. Therefore, the objective of this thesis is to apply different control strategies to the forward dynamic simulation of human gait using a simple 2D model. The control strategy emulates the central nervous system function, which is in charge of controlling the muscle actuation in the real case.

### 1.2 Thesis Contents

A brief overview about the state of the art regarding forward dynamics techniques in biomechanics and optimization-based methods in order to predict human motion are given in Chapter 2.

In Chapter 3 multibody dynamics concepts are introduced. Two models (four-bar linkage model and two-dimensional human body model) are presented and the kinematic and dynamic analyses are described. Furthermore, solving and integration methods of the equations of motion are explained.

Chapter 4 contains the methodology of the joint forces and torques generation of the four-bar linkage model and the two-dimensional human body model by means of inverse dynamic analysis. Moreover, it describes the motion reconstruction of a real subject during healthy gait and the experimental setup of the biomechanics laboratory.

These calculated joint forces and torques are used as inputs for the forward dynamic analysis in order to reproduce the original motion, which is discussed in Chapter 5. The forward dynamic approach is applied to both models and the integration methods are compared.

Due to the unstable character of human gait and numerical errors during the integration of the equations of motion, the forward dynamic simulation is unstable without any control approach. Chapter 6 presents the application of two suitable control strategies (PD and CTC control) applied on the multibody models. The design of each control strategy is developed and the results of the forward dynamic simulation using the applied control methods are compared to the original captured motion.

The performance of the two control strategies is evaluated and compared in Chapter 7 for the four-bar linkage model. Moreover, the robustness of the CTC control approach is investigated.

Chapter 8 analyses the environmental and social impact of this project and includes an economical consideration.

Finally, the conclusions of this thesis are made and some extensions and future lines are proposed in Chapter 9.
2 State of the Art

In the field of biomechanics, forward dynamics techniques are used in order to predict body movements from known muscle forces or resultant joint torques, respectively, based on the principles of neural and optimal control. This approach can be suitable for the investigation of aspects of muscle function and energetic cost, for the simulation of gait disorders or for the prediction of human motion due to the combined actuation of the musculoskeletal system and assistive devices, such as exoskeletons or orthoses [32].

During the last years, a growing interest in motion prediction has appeared since it contributes to the anticipation of surgery results, to an enhanced design of prosthetic/orthotic devices, or to the study of human motion dynamics performing various tasks. In order to anticipate the adaptation of patient’s gait to mechanical interventions such as prosthetic devices or surgery, the biomechanics community attempts to predict the human motion of real subjects under virtual conditions.

In order to manage the actuation of the human body model which drives the motion, the forward dynamic analysis needs a robust control strategy. The resulting motion can be computed by integrating the equations of motion with respect to time. The most challenging aspect of the forward dynamic analysis is the characterization and implementation of the control method in order to drive the model. The motion of human beings is controlled by the cooperation of the nervous and the musculoskeletal systems. Physically, the central nervous system sends electrical signals which stimulate the muscle. Due to this stimulation, the muscle produces the force which actuates the skeleton. Meanwhile, the brain senses and processes information enabling the central nervous system to adapt for the next muscle stimulation. This human motion control ability is remarkable and the implementation of this unknown procedure is one of the biggest challenges in the forward dynamic analysis of biomechanical systems. An appropriate control strategy to generate a forward dynamic simulation which is fully consistent with human motion has not been clarified yet.

One of the first studies of human locomotion in forward dynamic simulations due to dynamic optimization has been carried out by Chow and Jacobson in 1971. They described the motion of the lower extremity during level walking using a five segment model with five degrees of freedom (see Figure 2.1a). A performance criterion was introduced which minimizes the muscular effort during the motion [10].
Application of different Control Strategies to the FD Simulation of Human Gait

(a) Biped gait, 1971 [10]. (b) Hybrid dynamic motion prediction method, 2013 [33].

Figure 2.1: Motion prediction: From the beginning to recent research. a) Two basic configuration of biped gait by Chow and Jacobson, 1971 [10]. b) Hybrid dynamic motion prediction method by Pasciuto, 2013 [33].

Until today, several researchers are investigating the prediction of human motion by means of forward dynamic simulations using different control methods in order to generate human-like motion. Thelen et al., for example, use a computed muscle control algorithm to vary muscle excitations in order to track experimental joint kinematics within a forward dynamic simulation [41, 42]. In [28], a neuromuscular locomotion controller is developed that realizes human-like responses to unexpected disturbances during locomotion.

The approaches most frequently used in literature in order to predict human motion are optimization-based methods [50, 47, 1, 37, 11, 21, 45, 43]. In the dynamic optimization methodologies, the muscle force histories can be defined in terms of parameters that are treated as design variables of the optimization algorithm. The main shortcoming of optimization-based methods in the forward dynamic simulation is the computational cost which increases considerably due to numerical integration of the equations of human motion [4]. Two different dynamic optimization-based approaches have been developed: data-based methods, which assume that the most realistic motion is the one that resembles an observed reference motion in similar conditions; and knowledge-based methods, which assume that the most realistic motion is the one which follows a specific motion control law [34].
The data-based method is commonly called the tracking problem since the control's task is to guide the system's motion to follow the observed reference motion [28]. This method is widely used and the main advantage lies in the intrinsic realism of the reference motion to be resembled [33]. Moreover, the strategies and styles that people adopt to carry out a task can be identified by changing the reference motion. The main drawback of this method is the restriction of being able to reasonably predict only tasks which are present in the database [33].

The knowledge-based methods solve the optimization problem by introducing objective functions whose minimization represents the control law in order to drive the motion. Most objective functions are energy related such as mechanical energy [37], metabolic energy [21] or squared torques at the joints [11], whose minimization is supposed to represent the most realistic motion. The knowledge-based method is applicable for walking and running modes [11, 45, 43]. In [49] the basic walk-to-stand- and slow to fast-transitions for human motion prediction are studied. Furthermore, Van den Bogert et al. (2012) introduced an adaptive neuromuscular control based on energy cost optimization, which has the capability to predict the effect of mechanical equipment properties on human performance. The knowledge-based method relies more to the principal aim of motion prediction since it does not require the reference motion as an input. It can uncover the principals of neuromuscular coordination and has potential applications in predicting patient responses to surgical intervention and in the design of prosthetic/orthotic devices [1, 48]. However, the identification of appropriate objective functions has proven to be difficult [34] and it is hard to obtain completely human-like motion. The objective functions are limited since they only partially represent the human neural control system. Often, a composite objective function combining several objective functions is needed to obtain realistic results [49].

In recent years, the potential of combining both data-based and knowledge-based methods has been investigated by several researchers. Hays et al. (2011) present a new framework through dynamics formulation for under-actuated systems where actuated state and unactuated input trajectories are prescribed and uncertainty statistics of the multibody dynamical system are quantified in the nonlinear programming optimization process [18]. Fluit et al. (2012) demonstrate in [16] a computationally efficient, three-dimensional, torque actuated and forward-dynamics based model of gait. A global controller is used for every gait phase which is fed with cyclic gait descriptors such as step width, step length and velocity of the COM and an optimization algorithm which quadratically minimizes the derivative of the joint moments every 0,01s [16].

Since 2012, hybrid predictive methods have been developed in order to control the musculoskeletal system more efficiently and to improve the human motion prediction, especially towards clinical applications by Xiang et al. [48] and Pasciuto et al. [33, 34].
Xiang et al. (2012) incorporate motion capture data into the optimization-based formulation to predict natural and subject specific human motions [48]. The method includes three procedures which contain each a sub-optimization problem. The efficiency of the method is demonstrated by simulating a box-lifting motion.

The hybrid dynamic motion prediction by Pasciuto et al. (2013) introduces instead knowledge in the data-based framework. Knowledge in the prediction is introduced to the optimization problem in the form of a dynamic motion control law, which is followed while resembling the actually performed reference motion from the database. Furthermore, the dynamic equilibrium of the human model, considering its interactions with the environment, is ensured (see Figure 2.1b). The proposed method has been applied to clutch pedal depression motions and the results are presented for three different predictions [33]. In 2014, Pasciuto et al. introduced a hybrid dynamic motion prediction whose objective function is composed of a weighted combination of data-based and knowledge-based contributions [34]. The prediction of the hybrid method is compared to data-based and knowledge-based methods which are applied to clutch pedal depression motions. The solutions favor the hybrid prediction method which seems to combine the advantages of both data- and knowledge-based approaches, enhancing the realism of motion prediction [34].

Further control approaches which can be interesting for the biomechanics research community are described in [22] and [44]. Kistemaker et al. (2012) describe a simplified position and movement control using a combined muscle spindle and Golgi tendon organ feedback applied to one- and two-degree of freedom arm models [22]. Next to an enhanced system’s response which responds fast, reaches the steady state and achieves small static position errors due to the combined feedback, the study might furthermore provide insight about strong physiological couplings between muscle spindle and Golgi tendon organ which enable a better understanding of the human nervous system.

Van den Bogert at el. (2012) present an optimal control of an energy-storing prosthetic knee in [44]. The objective function for optimal control is based on tracking of joint angles, tracking of joint moments, and the energy cost of operating the valves of the hydraulic actuator. The solutions of the optimal control are obtained, based on data collected from three subjects during walking, running, and a sit-stand-sit cycle.

In summary, motion prediction in forward dynamics is part of recent research but predictive simulation has not yet found widespread application because of its high computational cost [4]. In addition, there is no generally accepted optimality criterion for human gait and an efficient prediction simulating the control of the human central nervous system is not discovered so far.
3 Multibody Dynamics

The analysis of human gait using multibody dynamics techniques requires the use of simplified models of the human body. Therefore, a two-dimensional biomechanical human body model with 14 Degrees of Freedom (DOF) is presented in this chapter. Furthermore, a second multibody model - a four-bar linkage model - is introduced in this chapter for simplicity, since it has only one DOF. Positions, velocities and accelerations of the coordinates can be achieved due to the kinematic analysis. For this purpose, a set of constraint equations is presented. In the dynamic analysis, the equations of motion are derived and, moreover, different solving and integration methods of the equations of motion are described.

3.1 Biomechanics

In order to describe the observed human locomotion in the following chapters, the human body planes and directions are characterized in Figure 3.1.

![Figure 3.1: Human body planes and directions [46].](image-url)
The human gait cycle is represented in Figure 3.2. It usually begins with the right foot strike and finishes with the next right foot strike. Human gait can be divided in stance phase and swing phase. While the stance phase contains periods of single and double support, the swing phase is divided into three different swing periods (initial, mid and terminal swing).

Figure 3.2: Human gait cycle classified in phases and periods [39].

### 3.2 Model Applications

The following section presents two model applications used in this thesis. A simple four-bar linkage model is introduced to apply multibody dynamics techniques and test possible suitable control strategies to a one DOF system. This model is very useful in order to understand the methodology of inverse and forward dynamic analysis, reducing the complexity of the problem and decreasing computational time. However, in order to simulate human gait, a two-dimensional biomechanical human body model is used. Both models are formed by several rigid bodies connected by frictionless joints with idealized torque actuators. In this section, the topology of both models is described and the dynamic parameters are listed.
3.2.1 Four-bar Linkage Model

In order to analyse the approaches of multibody dynamics, a simple four-bar linkage model is developed. The model, which is shown in Figure 3.3, is formed by four bars, where one bar is fully constrained and not movable. That means, beginning and end points A and B are fixed at (0, 0) and (10, 0). Therefore three bars remain with different lengths and masses (see Table 3.1). The Center of Mass (COM) of each bar \( \mathbf{r}_G = (\mathbf{x}_G, \mathbf{z}_G)^T \), which is expressed using a local coordinate system \( \{X, Z\}_S \) (see Figure 3.4b), is located at the middle point (see Table 3.1). The moment of inertia of each bar \( \bar{I}_G \) is determined with respect to its COM (see Table 3.1). The motion of the bars can be described with five dependent generalized coordinates \( \mathbf{q} = \{x_1, z_1, x_2, z_2, \alpha\}^T \). The angle \( \alpha \) is imposed to be in the range of \( \alpha \in [-45^\circ, 45^\circ] \).

![Figure 3.3: Four-bar linkage model.](image)

**Table 3.1: Parameters of the four-bar linkage model.**

<table>
<thead>
<tr>
<th>No.</th>
<th>Length</th>
<th>COM Location</th>
<th>Mass</th>
<th>Moment of Inertia</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( l_i )</td>
<td>( \mathbf{x}_G = \frac{l_i}{2} )</td>
<td>( \mathbf{z}_G = 0 )</td>
<td>( m_i )</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>4</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>2,5</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>
3.2.2 2D Biomechanical Human Body Model

The two-dimensional biomechanical human body model with 14 DOF, used in this thesis, has been developed previously in [32]. The model is a multibody system which consists of 12 rigid bodies (Figure 3.4a): trunk, head, two arms, two forearms, two shanks, two thighs and two feet. The rigid bodies are linked with revolute joints and the model is constrained to move in the sagittal plane (Figure 3.1). In this case the sagittal plane is defined as the \((X,Z)\)-plane, where the horizontal global axis \(X\) points the direction of motion and the axis \(Z\) is perpendicular to the floor and points upwards.

![Human body segments](image1)

**Figure 3.4:** 2D biomechanical model of the human body [31].

Table 3.2 shows the anthropometric data of the model. The data acquisition is related to a healthy subject and will be explained in detail in Section 4.2.3. In order to express the position of the COM of each segment \(\mathbf{r}_G = \{\mathbf{x}_G, \mathbf{z}_G\}^T\), a local coordinate system with the origin at the proximal joint and local axes \(\{\mathbf{X}, \mathbf{Z}\}_S\) is used (Figure 3.4b). The moment of inertia of each segment \(\mathbf{I}_G\) is stored with respect to its COM [32].

The human body model can be defined with 13 points, representing the eight revolute joints combined with the end points of the five extreme segments (Figure 3.5a). Each point is expressed using X- and Z- Cartesian coordinates (hence, 26 variables). Further-
Table 3.2: Anthropometric data for the 12 segments of the 2D model (adapted from [31]).

<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>Length $L_i$ [m]</th>
<th>COM Location $\bar{x}_G$ [m]</th>
<th>Mass $m_i$ [kg]</th>
<th>Principal Moment of Inertia $I_G$ [kg·m²]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Trunk</td>
<td>0,498</td>
<td>0,230</td>
<td>37,801</td>
<td>44,448</td>
</tr>
<tr>
<td>2</td>
<td>Head</td>
<td>0,343</td>
<td>0,139</td>
<td>5,119</td>
<td>2,163</td>
</tr>
<tr>
<td>3</td>
<td>Right arm</td>
<td>0,311</td>
<td>0,146</td>
<td>1,922</td>
<td>1,190</td>
</tr>
<tr>
<td>4</td>
<td>Right forearm</td>
<td>0,376</td>
<td>0,171</td>
<td>2,299</td>
<td>0,974</td>
</tr>
<tr>
<td>5</td>
<td>Left arm</td>
<td>0,311</td>
<td>0,146</td>
<td>1,922</td>
<td>1,190</td>
</tr>
<tr>
<td>6</td>
<td>Left forearm</td>
<td>0,376</td>
<td>0,171</td>
<td>2,299</td>
<td>0,974</td>
</tr>
<tr>
<td>7</td>
<td>Right thigh</td>
<td>0,417</td>
<td>0,154</td>
<td>9,284</td>
<td>12,215</td>
</tr>
<tr>
<td>8</td>
<td>Right shank</td>
<td>0,422</td>
<td>0,170</td>
<td>4,008</td>
<td>5,153</td>
</tr>
<tr>
<td>9</td>
<td>Right hindfoot</td>
<td>0,143</td>
<td>0,037 -0,023</td>
<td>1,027</td>
<td>0,441</td>
</tr>
<tr>
<td>10</td>
<td>Left thigh</td>
<td>0,417</td>
<td>0,154</td>
<td>9,284</td>
<td>12,215</td>
</tr>
<tr>
<td>11</td>
<td>Left shank</td>
<td>0,422</td>
<td>0,170</td>
<td>4,008</td>
<td>5,153</td>
</tr>
<tr>
<td>12</td>
<td>Left hindfoot</td>
<td>0,143</td>
<td>0,037 -0,023</td>
<td>1,027</td>
<td>0,441</td>
</tr>
</tbody>
</table>

![Points and joints](image1.png)

![Relative (green and red) and absolute (blue) angles. In green: right leg & arm; in red: left leg & arm.](image2.png)

Figure 3.5: Defined configuration of the planar model (adapted from [31]).
more, 12 angular coordinates ($\alpha_i$) are described: 11 relative angles and one absolute angle ($\alpha_0$) which orientate the trunk with respect to the vertical direction in the sagittal plane (Figure 3.5b). Therefore, the generalized coordinates vector $q$ contains 38 variables.

3.3 Multibody Formulation. Kinematic Analysis

In order to understand human motion, the kinematics of general motion of a multibody system is studied. A kinematic analysis gives information about the motion of a multibody system regardless of the forces acting on the system. The motion depends on geometry and configuration of the rigid bodies. A set of restrictions is used in order to constrain these rigid bodies with respect to each other. The kinematic relation between the coordinates of the mechanical system can be modeled by constraint equations.

This section presents the tools used to determine position, velocity and acceleration of the system by use of constraint equations. The formulations are taken from [32].

If the generalized coordinates, describing the multibody system configuration, are dependent, they can be related to each other using kinematic constraint equations. These equations contain rigid body constraints to assure the characteristics of each segment and rheonomic constraints (driver constraints) to guide the system motion. The constraint equations can be written as:

$$\Phi(q, t) = \begin{bmatrix} \Phi_1(q) \\ \vdots \\ \Phi_{ms}(q) \\ \Phi_{ms+1}(q, t) \\ \vdots \\ \Phi_{ms+mr}(q, t) \end{bmatrix} = 0 \quad (3.1)$$

The generalized coordinates vector $q$ contains $n$ coordinates, which are dependent to each other. In general, these coordinates can be divided in $n_d$ dependent coordinates and $n_i$ independent ones ($n = n_d + n_i$), where the independent coordinates represent the DOF of the system.

$$q = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{bmatrix} q^d \\ q^i \end{bmatrix} \quad (3.2)$$

The superscripts ‘d’ and ‘i’ of Eq. (3.2) indicate dependent and independent coordinates. The independent coordinates chosen for the used models will be explained in Chapter 4.
Eq.(3.1) represents the vector of the \((m = ms + mr)\) kinematic constraints \(\Phi\). The \(ms\) scleronomic constraints do not depend on time, whereas the \(mr\) restrictions are rheonomic constraints with an explicit dependency on time. The formulation of the constraint equations is explained more in detail in Section 3.3.1.

It has to be mentioned that in this thesis, the number \(ms\) of scleronomic constraints equals the number \(n_d\) of dependent coordinates. Furthermore, in the inverse dynamic analysis, which will be explained in Chapter 4, the number \(mr\) of rheonomic constraints equals the number \(n_i\) of independent coordinates. Therefore, in the special case of the inverse dynamic analysis, the total number of constraints is consistent with the number \(n\) of the generalized coordinates \((m = n)\). In the case of the forward dynamic analysis, which will be explained in Chapter 5, no rheonomic constraints are used \((mr = 0)\). Hence the total number of constraints is conform to the number \(n_d\) of the dependent coordinates \((m = ms = n_d)\).

In order to obtain the generalized coordinates \(q\), Eq.(3.1) has to be solved. Because these equations are usually non-linear, the numerical interpolation method by Newton-Raphson is used. It is an iterative method which linearizes Eq.(3.1) by using the first two terms of its expansion in a Taylor series and an approximate initial position \(q_0\) [13].

It achieves quadratic convergence near the solution and can be written as:

\[
\Phi(q, t) \approx \Phi(q_0, t) + \Phi_q(q_0)(q - q_0) = 0
\]

where \(\Phi_q(q_0)\) is the Jacobian matrix of the constraints evaluated at the approximate solution \(q_0\). The Jacobian matrix contains the partial derivatives of the constraints with respect to the generalized coordinates. It can be written as:

\[
\Phi_q(q) = \begin{bmatrix}
\frac{\partial \Phi_1}{\partial q_1} & \frac{\partial \Phi_1}{\partial q_2} & \cdots & \frac{\partial \Phi_1}{\partial q_n} \\
\frac{\partial \Phi_2}{\partial q_1} & \frac{\partial \Phi_2}{\partial q_2} & \cdots & \frac{\partial \Phi_2}{\partial q_n} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial \Phi_m}{\partial q_1} & \frac{\partial \Phi_m}{\partial q_2} & \cdots & \frac{\partial \Phi_m}{\partial q_n}
\end{bmatrix}
\]

(3.4)

Due to the fact that \(q_0\) is an approximate solution, Eq.(3.3) does not meet the constraints. Hence, the equation can be written as:

\[
\Phi_q(q_j)(q_{j+1} - q_j) = -\Phi(q_j) \quad ; \quad j = 0, 1, 2, \ldots
\]

(3.5)

In order to solve for \(q_{j+1}\), the Jacobian matrix \(\Phi_q(q_j)\) has to be inverted. In the case of the inverse dynamic analysis, where \(m = n\), this matrix is a square matrix. Since the constraint equations are independent (see Section 4.1 and 4.2), the determinant of \(\Phi_q\) is different from zero and therefore, the Jacobian matrix is invertible. It yields:

\[
q_{j+1} = \Phi_q(q_j)^{-1}(-\Phi(q_j) + \Phi_q(q_j)q_j)
\]

(3.6)
But in the case of the forward dynamic analysis, where \( m = n_d \), the Jacobian matrix has to be divided into two matrices referring to the dependent and independent coordinates [9]:

\[
\begin{bmatrix}
\Phi^d_q | \Phi^i_q
\end{bmatrix}
\begin{bmatrix}
q^d \\
q^i
\end{bmatrix} = \Phi^d_q q^d + \Phi^i_q q^i
\] (3.7)

Here, \( \Phi^d_q \) is a square matrix. Assuming, that the independent coordinates are known, the dependent ones can be obtained iteratively by:

\[
q^d_{j+1} = \Phi^d_q(q_j)^{-1}\left(-\Phi(q_j) + \Phi_q(q_j)q^d_j - \Phi^i_q(q_j)q^i_j\right) \quad ; \quad j = 0, 1, 2, \ldots
\] (3.8)

The interpolated value of \( q \) is obtained in an iterative process with Eq. (3.6) or Eq. (3.8) till the algorithm converges to a solution which satisfies all the constraints of Eq. (3.1) with a specified tolerance \(|\Phi| < 10^{-9}\).

In order to obtain the generalized velocity vector \( \dot{q} \), the velocity constraints - which are the time derivative of the kinematic constraints in Eq. (3.1) - need to be solved:

\[
\dot{\Phi}(q, \dot{q}, t) \equiv \frac{d\Phi(q, t)}{dt} = \Phi_q \dot{q} + \Phi_t = 0
\] (3.9)

where the vector \( \dot{\Phi}_t \) contains the partial derivatives of the constraints with respect to time. With the known configuration \( q \), the Jacobian matrix \( \Phi_q \) can be calculated as well as \( \dot{\Phi}_t \). The generalized velocities are obtained, as explained before for the configuration \( q \), differently (depending if it is an inverse dynamic analysis (IDA) or a forward dynamic analysis (FDA)) using the following equations [9]:

IDA: \( \dot{q} = \left(\Phi_q\right)^{-1}\left(-\dot{\Phi}_t\right) \) (3.10)

FDA: \( \dot{q}^d = \left(\Phi^d_q\right)^{-1}\left(-\Phi^i_q \dot{q}^i\right) \)

The generalized acceleration vector \( \ddot{q} \) can be obtained similarly by using the derivative of Eq. (3.9) with respect to time:

\[
\ddot{\Phi}(q, \dot{q}, \ddot{q}, t) \equiv \frac{d\dot{\Phi}(q, \dot{q}, t)}{dt} = \Phi_q \ddot{q} + \dot{\Phi}_q \dot{q} + \dot{\Phi}_t = 0
\] (3.11)

where \( \dot{\Phi}_q \) is the time derivative of the Jacobian matrix and \( \dot{\Phi}_t \) is the time derivative of \( \Phi_t \). With known position and velocity vectors, the acceleration vector can be calculated for IDA and FDA with [9]:

IDA: \( \ddot{q} = \left(\Phi_q\right)^{-1}\left(-\dot{\Phi}_q \dot{q} - \dot{\Phi}_t\right) \) (3.12)

FDA: \( \ddot{q}^d = \left(\Phi^d_q\right)^{-1}\left(-\Phi^i_q \dot{q}^i - \Phi^i_t \dot{q}^i\right) \)
Hence, positions, velocities and accelerations are obtained for both IDA and FDA. It has to be considered, that the FDA requires the independent coordinates $\mathbf{q}^i$ and the independent velocities $\dot{\mathbf{q}}^i$. The independent accelerations $\ddot{\mathbf{q}}^i$ are obtained using the independent coordinates $\mathbf{q}^i$ and the independent velocities $\dot{\mathbf{q}}^i$.

### 3.3.1 Constraint Equations

As explained before, the configuration of the multibody system is defined by using dependent coordinates. The Cartesian coordinates of points and joints are complemented with trunk and joint angles to drive the motion easily. In order to relate the coordinates to each other, a set of kinematic constraints has to be imposed.

Two types of kinematic constraints are introduced: rigid body constraints assuring rigid body characteristics of each segment and driver constraints prescribing the motion of the system. It has to be mentioned, that the introduced restrictions define a planar model. In the case of a three-dimensional model the constraint formulations can be found in [32].

#### Rigid Body Constraints

Each segment of the model is defined by two extreme points. Two rigid bodies which are linked by a revolute joint, share one point and are related to each other with an angle $\varphi$ between the two segments (see Figure 3.6).

![Figure 3.6: Angular variable between two segments linked by a revolute joint.](image)

For each segment a constant distance $L$ between the two extreme points has to be ensured [13]. Hence, in the general case, illustrated in Figure 3.6, two constraints can be imposed:

$$
(x_j - x_i)^2 + (z_j - z_i)^2 - L_{ij}^2 = 0
$$

$$
(x_k - x_j)^2 + (z_k - z_j)^2 - L_{jk}^2 = 0
$$

(3.13)
Furthermore, the angular coordinate $\varphi$ between two linked segments can be related to the segments' end points by:

\[
(x_i - x_j)(z_k - z_j) + (z_i - z_j)(x_k - x_j) - L_{ij}L_{jk} \sin \varphi = 0
\] (3.14)

Note that this equation is not valid when $\varphi$ achieves $\pm 90^\circ$ and a formulation using the cosine has to be used instead [13]. For this thesis, it is ensured that the angles of the multibody systems never achieve a value of $\pm 90^\circ$.

The rigid body constraints are a type of scleronomic constraints and so they do not depend on time. Eq.(3.13) is used as many rigid bodies are existing and Eq.(3.14) appears as many angles are considered in the model. Hence, the number of rigid body constraints is conform to the number of rigid bodies plus angles existing in the generalized coordinates vector.

### Driver Constraints

In order to impose the motion of the multibody system explicitly, the driver constraint for a joint angle (Figure 3.6) can be described as:

\[
\varphi(t) - \varphi = 0
\] (3.15)

where $\varphi$ represents here the relative angle between two segments and $\varphi(t)$ is an analytical expression of a time function which imposes the motion.

Moreover, driver constraints describe the position of a joint with respect to the (x,z)-coordinates. The driver constraints of the joint $j$ in Figure 3.6 can be expressed as:

\[
\begin{align*}
x_j(t) - x_j &= 0 \\
z_j(t) - z_j &= 0
\end{align*}
\] (3.16)

In this formulations, the driver constraints are rheonomic constraints since they do depend on time explicitly. The number of driver constraints equals the number of DOF of the system.

### 3.4 Multibody Formulation. Dynamic Analysis

The dynamic analysis determines the motion of a multibody system that results from the application of external forces or from kinematically driven degrees of freedom [13].
In this thesis, a multibody dynamics methodology is used in order to obtain the dynamic equations of motion. As explained in [12], the Lagrange equations using dependent coordinates $q$ can be written as:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}} \right) - \frac{\partial T}{\partial q} + \Phi_q^T \lambda = Q \quad (3.17)$$

where $T$ is the kinetic energy of the multibody system, $\lambda$ the vector of the Lagrange multipliers and $Q$ the generalized force vector. $\Phi_q^T \lambda$ are the generalized forces associated with the imposed constraints.

The kinetic energy can be expressed as:

$$T = \frac{1}{2} q^T M \dot{q} \quad (3.18)$$

where $M$ is the mass matrix of the system. If the mass matrix is constant and does not depend upon the system configuration, Eq.(3.17) can be written as:

$$M \ddot{q} + \Phi_q^T \lambda = Q \quad (3.19)$$

In general the system needs as many equations as unknown variables are existing. Besides $n$ generalized accelerations $m$ Lagrange multipliers has to be determined. Therefore, the system has $(n + m)$ unknown values and needs next to the $n$ equations in Eq.(3.19) $m$ further equations. Using the $m$ constraint equations from Section 3.3.1, the system can be expressed as:

$$\begin{cases} M \ddot{q} + \Phi_q^T \lambda = Q \\ \Phi(q, t) = 0 \end{cases} \quad (3.20)$$

This system is a Differential Algebraic Equation (DAE) system with $n$ second order Ordinary Differential Equations (ODE) and $m$ algebraic constraints. A detailed derivation of the equations of motion can be found in [13].

The following sections explain the determination of the mass matrix $M$ and the generalized force vector $Q$.

### 3.4.1 Mass Matrix of Planar Bodies

In each of the elements of a multibody system, which undergoes a given motion, inertia forces arise. In order to obtain the mass matrix of the multibody system, the formation of these inertia forces needs to be analysed. The form of the mass matrix is provided in natural coordinates.
Figure 3.7: Representation of a generic rigid body using natural coordinates.

Figure 3.7 shows a planar element, whose motion is defined by the basic points $i$ and $j$ in global coordinates $\{X,Z\}$ as well as in local coordinates $\{\bar{X},\bar{Z}\}$ with its origin in point $i$ and axis $\bar{X}$ going through point $j$. The centre of mass $G$ is considered generally as a point $(\bar{x}_G,\bar{z}_G)$.

The location of a generic point $P$ of the element can be defined as:

$$r_P = r_i + A \hat{r}_P \quad (3.21)$$

where $A$ is the rotation matrix. With the distance $L_{ij}$ between points $i$ and $j$ the rotation matrix can be described with the coordinates of both points [12]:

$$A = \frac{1}{L_{ij}} \begin{pmatrix} x_j - x_i & z_i - z_j \\ z_j - z_i & x_j - x_i \end{pmatrix} \quad (3.22)$$

The derivation of Eq.(3.21) with respect to time expresses the velocity of point $P$:

$$\dot{r}_P = \dot{r}_i + \dot{A} \hat{r}_P \quad (3.23)$$

with

$$\dot{A} = \frac{1}{L_{ij}} \begin{pmatrix} \dot{x}_j - \dot{x}_i & \dot{z}_i - \dot{z}_j \\ \dot{z}_j - \dot{z}_i & \dot{x}_j - \dot{x}_i \end{pmatrix} \quad (3.24)$$

The kinetic energy $T$ of the element can be expressed as:

$$T = \frac{1}{2} \int \dot{r}_P^T \dot{r}_P \ dm \quad (3.25)$$
Combining Eq. (3.23) and (3.25), the kinetic energy can be written as:

\[
T = \frac{1}{2} \left[ \int \dot{\mathbf{r}}_i^T \mathbf{r}_i dm + \int \dot{\mathbf{r}}_i^T \mathbf{A} \mathbf{\ddot{r}}_p dm + \int \dot{\mathbf{r}}_p^T \mathbf{A} \dot{\mathbf{r}}_i dm + \int \dot{\mathbf{r}}_p^T \mathbf{A} \dot{\mathbf{r}}_p dm \right] \tag{3.26}
\]

With the mass \( m_i \) of the element and the moment of inertia \( I_i \) with respect to point \( i \), Eq. (3.26) can be simplified to:

\[
T = \frac{1}{2} \left[ m_i \dot{\mathbf{r}}_i^T \dot{\mathbf{r}}_i + 2m_i \dot{\mathbf{r}}_i^T \mathbf{A} \mathbf{\ddot{r}}_G + \frac{I_i}{L_{ij}^2} \left( \dot{x}_j - \dot{x}_i \right)^2 + \left( \dot{z}_j - \dot{z}_i \right)^2 \right] \tag{3.27}
\]

Rewriting the formulation, the kinetic energy can be expressed with the mass matrix of the element \( \mathbf{M}_e \) and the velocity vector of the points \( i \) and \( j \) (\( \mathbf{q}_e = \{ \dot{x}_i, \dot{z}_i, \dot{x}_j, \dot{z}_j \}^T \)):

\[
T = \frac{1}{2} \mathbf{q}_e^T \begin{pmatrix}
\mathbf{M}_e & 0 \\
0 & \mathbf{M}_e
\end{pmatrix} \mathbf{q}_e \tag{3.28}
\]

Hence, the mass matrix of each planar rigid body defined with 2 points is a 4 \( \times \) 4 matrix. In order to obtain the global system mass matrix, all element mass matrices have to be assembled. If for example the mass of a two-dimensional element of the system is obtained using the coordinates \( q_3, q_4, q_5, q_6 \), the elements of the mass matrix \( \mathbf{M}_2 \) will be integrated to the global mass matrix \( \mathbf{M} \) as shown in Figure 3.8. Overlapping elements of mass matrices can appear in the global mass matrix. They illustrate, that at these coordinates more than one rigid body is connected.

**Figure 3.8:** Assembly of the element mass matrices into the global mass matrix [32].
3.4.2 Generalized Force Vector

In the following, a formulation of a force applied to a rigid body will be explained. Continuing with the explained methodology before, Figure 3.9 shows a force $\mathbf{F}$ applied on the point $P$ of the rigid body, which is defined using two points $i$ and $j$.

![Figure 3.9: Generic concentrated force applied on a planar rigid body.](image)

Two unit vectors can be defined as a function of the point coordinates [12]:

$$\mathbf{u} = \frac{1}{L_{ij}} \left\{ \begin{array}{c} x_j - x_i \\ z_j - z_i \end{array} \right\} \quad ; \quad \mathbf{v} = \frac{1}{L_{ij}} \left\{ \begin{array}{c} z_i - z_j \\ x_j - x_i \end{array} \right\}$$

(3.29)

where $L_{ij}$ is the distance between the two points $i$ and $j$.

Hence, the position vector of point $P$ can be written as:

$$\mathbf{r}_P = \mathbf{r}_i + a\mathbf{u} + b\mathbf{v}$$

(3.30)

where $a$ and $b$ are the lengths of the unit vectors. Inserting the unit vector, it can be described as:

$$\mathbf{r}_P = \left\{ \begin{array}{c} x_i \\ z_i \end{array} \right\} + \frac{a}{L_{ij}} \left\{ \begin{array}{c} x_j - x_i \\ z_j - z_i \end{array} \right\} + \frac{b}{L_{ij}} \left\{ \begin{array}{c} z_i - z_j \\ x_j - x_i \end{array} \right\}$$

(3.31)

Reorganizing the formulation, it can be expressed with the coordinates of the two points $i$ and $j$:

$$\mathbf{r}_P = \frac{1}{L_{ij}} \left[ \begin{array}{ccc} L_{ij} - a & b & -b \\ -b & L_{ij} - a & b \\ c & d \end{array} \right] \left\{ \begin{array}{c} x_i \\ z_i \\ x_j \\ z_j \end{array} \right\} = \mathbf{C}_p \mathbf{q}_e$$

(3.32)
In order to express the applied force $F$ related to the coordinates of the two points $i$ and $j$, the principle of virtual work is used [13].

$$Q_e = C_p^T F$$  \hspace{1cm} (3.33)

The generalized force vector $Q_e$ for each element of a two-dimensional model contains - depending on the formulation - external or external and internal forces. In this thesis, two external forces are applied to the element (depending on the formulation). On the one hand the external forces are represented by a torso that is applied to the element. This external forces only appear in the forward dynamic analysis and will be discussed in Chapter 5. On the other hand, external forces are applied by means of concentrated loads. In order to determine these external forces, following settings can be made:

$$F = \begin{cases} \mathbf{0} \\ -mg \end{cases}, \quad a = \bar{x}_G \quad \text{and} \quad b = \bar{z}_G$$  \hspace{1cm} (3.34)

The generalized force vector $Q_e$ for one segment due these external forces can be obtained with:

$$Q_e = \frac{1}{L_{ij}} \begin{bmatrix} L_{ij} - \bar{x}_G & -\bar{z}_G \\ \bar{z}_G & L_{ij} - \bar{x}_G \\ -\bar{z}_G & \bar{z}_G \end{bmatrix} \begin{bmatrix} 0 \\ -mg \end{bmatrix}$$  \hspace{1cm} (3.35)

The global generalized force vector $Q$ is an assembly of the generalized forces $Q_e$ at every segment like explained previously for the global mass matrix in Figure 3.8.

### 3.4.3 Solution of the Equations of Motion

Considering the global mass matrix $M$ of the system and the generalized forces $Q$ are known, the equations of motion in Eq.(3.20) can be solved. There are several methods to obtain the generalized accelerations $\ddot{q}$ and the Lagrange multipliers $\dot{\lambda}$. The generalized positions can be determined using integration methods, which are explained in Section 3.4.4. However, four different formulations to solve the equations of motion with both dependent and independent coordinates will be discussed and evaluated in this section.

**Lagrange Multipliers Method**

The generalized accelerations $\ddot{q}$ can be calculated with Eq.(3.11) and therefore, Eq.(3.19) gives a solution for the Lagrange multipliers $\dot{\lambda} \in \mathbb{R}^m$. In matrix formulation it can be written as:

$$\begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} Q \\ -\Phi_q \ddot{q} - \dot{\Phi}_f \end{bmatrix}$$  \hspace{1cm} (3.36)
The main advantage of the dynamic formulation in dependent coordinates using Lagrange multipliers is the conceptual simplicity of the method and that the forces associated with the constraints can be calculated with a minimum additional effort ($\lambda$ can be obtained directly).

But numerical integration causes mistakes and the constraint conditions are progressively violated leading to unacceptable results in long time simulations [13]. The unstable behaviour of the numerical integration is caused by using the kinematic acceleration Eq.(3.11) which is obtained by differentiating the constraint Eq.(3.1) twice with respect to time. In the numerical integration process, this Eq.(3.11) is being integrated again twice with respect to time. Even though the initial condition guarantees $\Phi = 0$, the round-off errors which appear during the numerical integration do not satisfy the constraint equations. The effects of these errors increase even more with time. This can be shown for example for the rigid body constraint equations of constant distance between two points $i$ and $j$ explained in Section 3.3.1:

$$\mathbf{r}_i - \mathbf{r}_j)^T \mathbf{r}_i - \mathbf{r}_j) - L_{ij}^2 = 0$$

(3.37)

Differentiating this equation twice with respect to time, it yields:

$$\mathbf{r}_i - \mathbf{r}_j)^T \ddot{\mathbf{r}}_i - \ddot{\mathbf{r}}_j + \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j)^T \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j = 0$$

(3.38)

Integrating Eq.(3.38), the information about the constant distance $L_{ij}$ got lost. Hence, during the simulation this distance ceases to be constant [13] as shown in Figure 3.10, where the length of the first bar of the four-bar linkage model decreases over long simulation time using the method of the Lagrange multipliers (red curve). The same happens to other constraints which contain constant terms.

![Figure 3.10: Length of first segment of four-bar linkage model using the method of Lagrange multipliers and the stabilized Lagrange method.](image)

Final Degree Project - Olga Pätkau
Stabilized Lagrange Method

In order to stabilize the constraint conditions for long time simulations the stabilized Lagrange method from Baumgarte [5] is used. Here, the differential constraint equations are replaced by the following system:

\[ \ddot{\Phi} + 2\xi \omega \dot{\Phi} + \omega^2 \Phi = 0 \]  \hspace{1cm} (3.39)

where \( \xi \) and \( \omega \) are appropriately chosen constants. According to [12] the constants are chosen to be \( \xi = 1 \) and \( \omega = 10^{\frac{1}{s}} \) in order to avoid overshoot. With \( \dot{\Phi} = \Phi_q \ddot{q} + \dot{\Phi}_q \dot{q} + \dot{\Phi}_t \), the new set of equations of motion can be written again in matrix formulation:

\[
\begin{bmatrix}
    M & \Phi_q^T \\
    \Phi_q & 0
\end{bmatrix}
\begin{bmatrix}
    \ddot{q} \\
    \lambda
\end{bmatrix} =
\begin{bmatrix}
    Q \\
    -\dot{\Phi}_q \ddot{q} - \dot{\Phi}_t - 2\xi \omega \dot{\Phi} - \omega^2 \Phi
\end{bmatrix}
\]  \hspace{1cm} (3.40)

The stabilized Lagrange method by Baumgarte is general, simple and numerically efficient. Figure 3.10 shows for example that the length of the first segment of the four-bar linkage model remains constant using this method (black curve). However, not all possible instabilities, such as near kinematic singular configurations, are solved [17].

Penalty Formulation

In the penalty formulations proposed in [6], the Lagrange multipliers are eliminated from the equations of motion. This method incorporates the constraint equations as a dynamical system penalized by a penalty factor \( \alpha_k \).

\[ \lambda = \alpha_k (\dot{\Phi} + 2\Omega_k \mu_k \dot{\Phi} + \Omega_k^2 \Phi) \]  \hspace{1cm} (3.41)

with \( \alpha_k = 10^8 \left( \left[ \frac{kg}{m^2} \right] \text{ or } \left[ \frac{kgm^2}{rad} \right] \right) \), the damping ratio \( \mu_k = 1 \) and the natural frequency \( \Omega_k = 10^{\frac{1}{s}} \). The new system contains \( n \) ordinary differential equations with \( n \) unknown accelerations \( \ddot{q} \).

\[ M \ddot{q} + \Phi_q^T \lambda = Q \]  \hspace{1cm} (3.42)

Using Eq.(3.41) and \( \dot{\Phi} = \Phi_q \ddot{q} + \dot{\Phi}_q \dot{q} + \dot{\Phi}_t \), the system can be written as:

\[ (M + \Phi_q^T \alpha_k \Phi_q)\ddot{q} = Q - \Phi_q^T \alpha_k (\dot{\Phi}_q \ddot{q} + \dot{\Phi}_t + 2\Omega_k \mu_k \dot{\Phi} + \Omega_k^2 \Phi) \]  \hspace{1cm} (3.43)

Solving the system for accelerations, the Lagrange multipliers \( \lambda \) can be calculated afterwards with following equation:

\[ \lambda = \alpha_k (\Phi_q \ddot{q} + \dot{\Phi}_q \dot{q} + \dot{\Phi}_t + 2\Omega_k \mu_k \dot{\Phi} + \Omega_k^2 \Phi) \]  \hspace{1cm} (3.44)

The penalty method ensures stability of the constraint conditions and is also more suitable at singular positions of the multibody system. The difficulty of this method is to choose the penalty factor. Large values ensure convergence to the constraints but may also lead to numerical conditioning problems and develop round-off errors [13].
Matrix-R Formulation

The matrix-R formulation is based on the use of a projection matrix to transform the dependent generalized coordinates into independent coordinates $z$ of the system [12]. The transformation at the velocity level can be expressed as:

$$\dot{q} = R\dot{z}$$  \hspace{1cm} (3.45)

The time derivative of the velocities in dependent coordinates is given by:

$$\ddot{q} = R\ddot{z} + \dot{R}\dot{z}$$  \hspace{1cm} (3.46)

Integrating it into the equations of motion (3.20) yields:

$$MR\ddot{z} + M\dot{R}\dot{z} = Q$$  \hspace{1cm} (3.47)

The generalized force vector $Q$ contains in this formulation external forces (gravitational forces and lumbar joint forces) as well as internal forces which are the internal joint torques.

The system in Eq.(3.47) is not square and has more equations than unknowns. It will be extended by pre-multiplying all terms with $R^T$:

$$R^T M\ddot{z} = R^T Q - R^T M\dot{R}\dot{z}$$  \hspace{1cm} (3.48)

Matrix $R$ is expressed as:

$$R = \begin{bmatrix} - (\Phi_d q)^{-1} \Phi_i q \\ I_g \end{bmatrix}$$  \hspace{1cm} (3.49)

where $I_g$ is the $(g = n - m)$ identity matrix and the superscripts ‘d’ and ‘i’ indicate dependent and independent coordinates. Hence, the equation of motion in independent coordinates can be formulated compact as:

$$\bar{M} \ddot{z} = \bar{Q}$$  \hspace{1cm} (3.50)

A great advantage of the matrix-R formulation is an important reduction in the number of equations to be integrated. But most important is that the algebraic constraint equations (3.1) do not appear any more in the equations of motion using this formulation. Therefore, the instability problem in the integration of the constraint equations, which was explained before, disappear as well. Without the constraint equations the DAE system becomes an ODE system. In order to integrate an ODE system, many well-known methods exist. However, this has a price in terms of computational effort. Pre- and post-processing work is required to obtain the independent constraint equations.
and the absolute motion of each joint and element. Moreover, the equations of motion contain matrices that, although small, are full and sometimes expensive to evaluate [13].

Despite the higher computational effort, the advantages in the integration process predominate. Therefore, in this thesis, the matrix R formulation is chosen to be the appropriate solving method. In Section 4.1 the four methods are compared for the inverse dynamic analysis for the four bar-link model.

### 3.4.4 Integration of the Equations of Motion

In order to integrate the equations of motion transformed in a second order ODE system, Matlab provides several numerical algorithms called the ode-solvers. Two different ode solvers from Matlab as well as the algorithm of the well-known trapezoidal rule will be introduced in this chapter and compared in Chapter 5. Particularly, only explicit one-step solvers are investigated.

**ODE Solver Matlab**

The first ode solver to be chosen is a variable-step solver. Depending on the dynamics of the model, the step size can vary from step to step. In particular, a variable-step solver increases or reduces the step size to meet the error tolerances that are specified. The step size is dynamically adjusted as necessary. Hence, the number of steps can be reduced and the simulation time might be shortened significantly [27]. In this thesis, the solver ode23 is chosen. It solves nonstiff differential equations with the method of Runge-Kutta [13]. More precisely it is based on Runge-Kutta(2,3) pair of Bogacki and Shampine which is explained in detail in [7]. It is a one-step solver and efficient at crude tolerances and as well in the presence of mild stiffness [27].

The difficulty of the variable-step solver in this project is that the reference data (such as reference motion and actuator torques and forces), which come from the laboratory and are obtained in the inverse dynamic analysis (explained in Chapter 4), are only available in fixed time steps. But if the solver adjusts the step size dynamically during the simulation, the reference values need to be evaluated at these specific chosen times. In order to obtain the reference data in the time steps chosen by the variable-step solver, linear interpolation methods and B-splines formulations have been used. The methods and results will be discussed in Chapter 5.

Another possibility to handle the availability of the reference data in fixed time steps is using a fixed-step solver. Therefore, the second ode solver from Matlab to be chosen
is the ode4 solver. It is a non-adaptive solver using as well the Runge-Kutta method of order 4 [26].

**Trapezoidal Rule**

In addition to the ode solvers which are explained before, the trapezoidal rule is implemented in order to compare the results. The algorithm, which is taken from [32] and represented in Figure 3.11, is explained in the following:

The position and velocity vectors at time step $n + 1$ are given by:

\[
\begin{align*}
z_{n+1} &= z_n + \frac{h}{2}(\dot{z}_n + \dot{z}_{n+1}) \\
\dot{z}_{n+1} &= \dot{z}_n + \frac{h}{2}(\ddot{z}_n + \ddot{z}_{n+1})
\end{align*}
\]  

where $h$ is the used time step. If the position at time step $n + 1$ is given as an input variable, Eq.(3.51) can be rewritten as:
\[
\dot{z}_{n+1} = \frac{2}{h}z_{n+1} - \left(\frac{2}{h}z_n + \dot{z}_n\right) \\
\ddot{z}_{n+1} = \frac{4}{h^2}z_{n+1} - \left(\frac{4}{h^2}z_n + \frac{4}{h}\dot{z}_n + \ddot{z}_n\right)
\] (3.52)

Combining Eq.(3.52) with the equations of motion in Eq.(3.48), the residual \( f(z) \) at time step \( n + 1 \) can be defined as:

\[
f(z) = (M \ddot{z} - \bar{Q})_{n+1} = M \left(\frac{4}{h^2}z_{n+1} - \frac{4}{h^2}z_n - \frac{4}{h}\dot{z}_n - \ddot{z}_n\right) - \bar{Q} = 0
\] (3.53)

In order to resolve this equation, an iterative Newton-Raphson process is used to adjust the position at the time step \( n + 1 \):

\[
\left[\frac{\partial f(z)}{\partial z}\right]^j \Delta z_{n+1}^{i+1} = -f(z)^j
\] (3.54)

\[
z_{n+1}^{j+1} = z_{n+1}^j + \Delta z_{n+1}^{j+1}
\] (3.55)

In [14] it is explained that the tangent matrix can be approximated with :

\[
f_z \equiv \left[\frac{\partial f(z)}{\partial z}\right] \approx R^T MR + \frac{h^2}{4} R^T \left(-\frac{\partial \bar{Q}}{\partial z}\right) R + \frac{h}{2} R^T \left(-\frac{\partial \bar{Q}}{\partial \dot{z}}\right) R + MR
\] (3.56)

To start the iterative procedure shown in Figure 3.11, an initial guess of the position at time step \( n + 1 \) has to be calculated:

\[
z_{n+1}^0 = z_n + h\dot{z}_n + \frac{h^2}{2}\ddot{z}_n
\] (3.57)

Then, using Eq.(3.52) the corresponding velocities and accelerations can be obtained. With the residual \( f(z) \) and the tangent matrix \( f_z \), the Newton-Raphson iteration process can be solved in order to correct the position at time step \( n + 1 \) with Eq.(3.55). The new velocities and accelerations at time step \( n + 1 \) can now be calculated again by means of Eq.(3.52) until the condition \( \|\Delta z\| < 10^{-9} \) is achieved.
## 4 Inverse Dynamic Analysis

The Inverse Dynamic Analysis (IDA) involves the kinematic and dynamic analysis of human motion by means of multibody system dynamics techniques. The purpose of the IDA is to determine the actuating or driving forces and torques that produce a specific motion, as well as the reactions that appear at each one of the multibody system’s joints [13].

The fact that an IDA yields the driving forces and torques is extremely important since these torques are necessary to control a system in order to follow a desired trajectory. To achieve these driving torques, the desired motion has to be described kinematically and body segment parameters (BSP) are required.

In this thesis, inverse dynamics techniques are applied to both introduced models (four-bar linkage model and two-dimensional human body model). While the reference motion and dynamic parameters of the four-bar linkage model are chosen, the human body model contains data from a real healthy subject. Therefore, the acquisition of the reference human motion and the estimation method calculating the BSP of the human body model will be described in this chapter.

The general structure of an IDA is shown in Figure 4.1. It consists of a kinematic and a dynamic analysis as explained in Chapter 3.

![Figure 4.1: Block diagram of an inverse dynamic analysis.](image)

The generalized coordinates $\mathbf{q}$, velocities $\dot{\mathbf{q}}$ and accelerations $\ddot{\mathbf{q}}$ can be obtained for each instance of time in the kinematic analysis using the reference motion as input data. The desired motion of the multibody system $\mathbf{z}_{\text{ref}}$ is given by driver constraints.
(Section 3.3.1), which describe the motion of the DOF or independent coordinates \( z = q^i \), respectively. As explained in Section 3.3, an approximate initial position \( q_0 \) at time \( t = 0 \) is needed for the iterative solution of the generalized coordinates. Because the independent coordinates are already given as an input (\( z_{\text{ref}} \)), only the initial approximate positions of the dependent coordinates \( q^d_0 \) are required. Moreover, a set of BSP has to be provided (see Figure 4.1).

Once the generalized positions \( q \) and their time derivatives \( \dot{q} \) and \( \ddot{q} \) are calculated at each time instance, the inverse dynamics problem can be solved in order to obtain the driving forces and torques \( Q_{\text{IDA}} \) (the BSP are needed in the IDA as well, see Figure 4.1). The dynamic analysis is solved by means of the velocity transformation formulation matrix \( R \) in Eq.(3.48), where the required actuation is provided in the form of generalized forces \( Q \) associated to the independent coordinates \( z \) [13]. Eq.(3.48) can be rewritten as:

\[
R^T M \ddot{z} = R^T (Q_0 - MR \dot{z}) + Q_{\text{IDA}}
\]

being \( R^T Q = R^T Q_0 + Q_{\text{IDA}} \), where \( Q_0 \) are the known generalized forces (in this thesis, the gravitational forces) and \( Q_{\text{IDA}} \) are the generalized forces associated to the independent coordinates and therefore the unknowns of the IDA:

\[
Q_{\text{IDA}} = R^T M \ddot{z} - R^T (Q_0 - MR \dot{z})
\]

The application of the IDA to both models used in this thesis (four-bar linkage model and two-dimensional human body model) will be described in the following sections.

### 4.1 IDA applied to Four-Bar Linkage Model

The four-bar linkage model, which is introduced in Section 3.2.1, is a system with one DOF. Its dynamic parameters are chosen and can be seen in Table 3.1. The generalized coordinates vector contains five coordinates \( q = \{x_1, y_1, x_2, y_2, \alpha\}^T \), where \( \alpha \) is the independent coordinate and therefore, equivalent to the DOF of the system (\( z = \alpha \)).

The constraint equations vector \( \Phi \) contains four rigid body constraints as well as one driver constraint:

\[
\Phi = \begin{cases} 
(x_1 - x_A)^2 + (z_1 - z_A)^2 - L_1^2 \\
(x_2 - x_1)^2 + (z_2 - z_1)^2 - L_2^2 \\
(x_B - x_2)^2 + (z_B - z_2)^2 - L_3^2 \\
L_1 \sin \alpha - (z_1 - z_A) \\
\alpha(t) - \alpha
\end{cases} = 0
\]
The motion \( \alpha(t) \) represents the desired motion \( z_{ref} \) and is defined by a sine-function which can be seen in Figure 4.2. Hence, \( \alpha(t) \) is limited to be in the range of \( \alpha \in [-45°, 45°] \). The angular velocity \( \dot{\alpha}(t) \) and the angular acceleration \( \ddot{\alpha}(t) \) are the time derivatives of the angle motion \( \alpha(t) \).

\[
\alpha(t) = \frac{\pi}{4} \sin \left( 2 \omega t - \frac{\pi}{2} \right)
\]

\[
\dot{\alpha}(t) = \frac{\pi \omega}{2} \cos \left( 2 \omega t - \frac{\pi}{2} \right)
\]

\[
\ddot{\alpha}(t) = -\pi \omega^2 \sin \left( 2 \omega t - \frac{\pi}{2} \right)
\]

\[\text{Time } t \hspace{1cm} \alpha(t) \hspace{1cm} \dot{\alpha}(t) \hspace{1cm} \ddot{\alpha}(t)\]

0 0.5 1 1.5 2 0 50 0 100 0

0 0.5 1 1.5 2 -100 0 0

0 0.5 1 1.5 2 -200 0 500

Figure 4.2: Defined angle motion \( \alpha(t) \) of the four-bar linkage model and its derivatives \( \dot{\alpha}(t) \) and \( \ddot{\alpha}(t) \) with \( \omega = \frac{\pi \text{ rad}}{2 \text{ s}} \).

In conclusion, the actuating or driving torque related to the angle \( \alpha \) can be calculated since the motion of \( \alpha(t) \) is given. Section 3.4.3 describes four different methods in order to solve the equations of motion (Lagrange multipliers method, stabilized Lagrange method, penalty formulation and matrix-R formulation). In this section, the inverse dynamics solutions of these four solving methods will be compared for the four-bar linkage model.

The Lagrangian multiplier \( \lambda_\alpha \) related to the independent coordinate \( \alpha \) is obtained three different times by means of the Lagrangian method in Eq.(3.36), the stabilized Lagrangian method in Eq.(3.40) and the penalty formulation in Eq.(3.44). The corresponding generalized force \( Q_{IDA} \), which is in this case a scalar, is obtained by means of the matrix-R formulation in Eq.(4.2). Because the Lagrangian multipliers method is the original formulation of the dynamics problem, its solution will be treated as the reference solution of the IDA. The solutions of the other three solving methods, which are introduced in order to stabilize the equations of motion in the case of numerical integration (see Section 3.4.3), will be compared to the first one.

Figure 4.3 shows the driving torque of \( \alpha \) of each stabilized solving method compared to the torque achieved by means of the Lagrangian multipliers method. It can be seen that in the inverse dynamic analysis of the four-bar linkage model all three methods achieve results for the driving torque, which are very similar to the original driving torque by means of the Lagrangian multipliers method. Therefore, all three methods seem to be adequate in order to solve inverse dynamics problems. However, the matrix-
R formulation shows the best results since the maximum absolute torque error of \(6.82 \times 10^{-13}\) is much lower than the others.

![Comparison of the driving torques using different dynamic solving methods with the driving torque achieved by means of the Lagrangian multipliers method.](image)

**Figure 4.3:** Comparison of the driving torques using different dynamic solving methods with the driving torque achieved by means of the Lagrangian multipliers method.

Considering that, in Section 3.4.3 the matrix R formulation is chosen to be the most appropriate method in order to avoid stabilization problems during numerical integration, which appear in the FDA (see Chapter 5). Therefore, in the following sections, the equations of motion are solved using the matrix-R formulation.

The IDA in matrix-R formulation is implemented in Matlab with the time steps of 10ms and 1ms. Hence, the driving torque \(Q_{\text{IDA}}\) is provided in both time steps and a time period of 10s.

### 4.2 IDA applied to 2D Human Body Model

The results of an IDA applied to a human body model have different applications. They are suitable in order to recognize normal and pathological gait patterns, to determine muscle forces together with an optimization approach, or in order to study the control of the motion carried out by the central nervous system [32].

The IDA provides the joint driving forces and torques, which are generated by the musculoskeletal system during human locomotion, using acquired kinematic data and estimated body segment parameters [32].
The IDA of the two-dimensional human body model used in this thesis (Section 3.2.2) was implemented prior this thesis by means of matrix-R formulation with a time step of 10ms in [32]. In this section, the procedure of the IDA applied to the human body model will be briefly explained and the results will be shown since they are used for the following chapters.

The human body model is a system with 14 DOF and is shown in the Figures 3.4 and 3.5. It consists of 38 generalized coordinates \( q = \{ x_1, z_1, x_2, z_2, \ldots, x_{13}, z_{13}, \alpha_0, \alpha_1, \ldots, \alpha_{11} \}^T \), where 14 coordinates are independent. The independent coordinates vector contains the position of the hip joint (P_4), as well as 12 angles (one absolute and 11 relative) that define the human body pose \( z = \{ x_4, z_4, \alpha_0, \ldots, \alpha_{11} \}^T \).

The corresponding generalized forces and torques \( Q_{\text{IDA}} \) of the fully actuated model are calculated via IDA and correspond to these independent coordinates. Hence, the contact wrench of the human body model acts on the hip joint. This approach is realistically not meaningful since in reality contact wrenches are acting on the subject’s feet.

In order to correspond the generalized forces and torques to the actual contact forces and the net joint motor torques, the contact wrench must be expressed at the contacting feet. During single support phases, the reaction forces can be translated to the contacting foot, and the actual joint torques can be recalculated properly, so that their corresponding generalized forces equate to those obtained with a contact wrench acting on the hip joint. However, during double support phases, it cannot be exactly known how this wrench is shared between the two foot-ground contacts. In [38] and [24] several methods for solving the inverse dynamics problem of the human gait during the double-support phase are presented and compared. For the two dimensional human body model of this thesis, the Corrected Force Plate (CFP) sharing method was used in [23] and [32] in order to solve the double support indeterminacy and to estimate the amount of the total contact wrench assigned to each foot.

The sharing problem by itself is known as a challenging difficulty in biomechanics’ research. Hence, in order to avoid this sharing problem and to reduce the complexity of human gait analysis, the contact between the human body and the environment is characterized, in this thesis, at the hip joint. Though being aware that this case does not represent the realistic approach, the thesis focuses on the analysis of the methodology since the main objective is the motion control in the forward dynamic analysis, which uses the results of the IDA as an input (see Chapter 5).

The vector of kinematic constraints \( \Phi \) of the human body model is expressed on the following page. It contains 24 rigid body constraints and 14 driver constraints.
Application of different Control Strategies to the FD Simulation of Human Gait

\[
\Phi = \begin{pmatrix}
(x_1 - x_2)^2 + (z_1 - z_2)^2 - L_2^2 \\
(x_2 - x_3)^2 + (z_2 - z_3)^2 - L_3^2 \\
(x_3 - x_4)^2 + (z_3 - z_4)^2 - L_4^2 \\
(x_5 - x_6)^2 + (z_5 - z_6)^2 - L_6^2 \\
(x_6 - x_7)^2 + (z_6 - z_7)^2 - L_7^2 \\
(x_7 - x_4)^2 + (z_7 - z_4)^2 - L_4^2 \\
(x_8 - x_9)^2 + (z_8 - z_9)^2 - L_2^2 \\
(x_8 - x_{10})^2 + (z_8 - z_{10})^2 - L_3^2 \\
(x_{10} - x_{11})^2 + (z_{10} - z_{11})^2 - L_4^2 \\
(x_{12} - x_8)^2 + (z_{12} - z_8)^2 - L_2^2 \\
(x_{13} - x_{12})^2 + (z_{13} - z_{12})^2 - L_6^2 \\
(x_8 - x_4) + L_1 \sin \alpha_0 \\
(x_3 - x_4)(z_8 - z_4) - (z_3 - z_4)(x_8 - x_4) - L_1 L_7 \sin \alpha_1 \\
(x_2 - x_3)(z_4 - z_3) - (z_2 - z_3)(x_4 - x_3) - L_7 L_8 \sin \alpha_2 \\
(x_3 - x_2)(z_1 - z_2) - (z_3 - z_2)(x_1 - x_2) - L_8 L_9 \sin \alpha_3 \\
(x_7 - x_4)(z_8 - z_4) - (z_7 - z_4)(x_8 - x_4) - L_1 L_{10} \sin \alpha_4 \\
(x_6 - x_7)(z_4 - z_7) - (z_6 - z_7)(x_4 - x_7) - L_{10} L_{11} \sin \alpha_5 \\
(x_7 - x_6)(z_5 - z_6) - (z_7 - z_6)(x_5 - x_6) - L_{11} L_{12} \sin \alpha_6 \\
(x_9 - x_8)(z_4 - z_8) - (z_9 - z_8)(x_4 - x_8) - L_1 L_2 \sin \alpha_7 \\
(x_4 - x_8)(z_{10} - z_8) - (z_4 - z_8)(x_{10} - x_8) - L_1 L_3 \sin \alpha_8 \\
(x_{11} - x_{10})(z_8 - z_{10}) - (z_{11} - z_{10})(x_8 - x_{10}) - L_3 L_4 \sin \alpha_9 \\
(x_4 - x_8)(z_{12} - z_8) - (z_4 - z_8)(x_{12} - x_8) - L_1 L_5 \sin \alpha_{10} \\
(x_{13} - x_{12})(z_8 - z_{12}) - (z_{13} - z_{12})(x_8 - x_{12}) - L_5 L_6 \sin \alpha_{11}
\end{pmatrix} = 0
\]
The driver constraints guide the system to follow a certain reference motion \( z_{\text{ref}} = \{x_{4,\text{ref}}, z_{4,\text{ref}}, \alpha_{0,\text{ref}}, \ldots, \alpha_{11,\text{ref}}\}^T = z(t) \) which is, in this thesis, a healthy human gait pattern. In order to obtain the driving forces and torques, the kinematic information of this reference motion as well as the BSP, which are presented in Table 3.2, are needed. The following sections will explain how these data have been acquired experimentally.

### 4.2.1 Experimental Setup

The experiments took place prior to this thesis in a laboratory which includes an opto-electronic system in order to capture the motion and two force plates to measure the foot-ground contact wrenches (Figure 4.4). The data of the motion are collected using 12 100 Hz Optitrack FLEX:V100R2 cameras that acquire the 3D trajectories of 37 passive markers which are attached to the human body. Two AMTI AccuGait Force Plates (FP) are located on a walkway, where the subject walks, in order to measure the foot-ground contact wrenches [32]. However, as explained before, these measurements are not used in this thesis.

![Gait analysis laboratory configuration](image)

*Figure 4.4: Gait analysis laboratory configuration [31].*
The human motion which is recorded contains more than one gait cycle. It starts at the heel strike of the right foot (0 % of gait cycle), includes also the next heel strike of the same foot (100 %) and finishes at the toe off of the left foot belonging to the next cycle (116 %) [31]. However, in this thesis, the captured data of just one whole gait cycle (100% ≈ 1.56s) is used (Figure 4.5).

![Figure 4.5: Phases of one gait cycle captured in the laboratory [31].](image)

### 4.2.2 Motion Reconstruction

The subject selected to perform the experiments is a healthy male adult at the age of 27 with a weight of 80 kg and a height of 1.75 m. In order to obtain the kinematic information of the motion, a set of 37 markers are attached to the subject and their trajectories are recorded. The marker positions are expressed in the global coordinate system using the axes \{X, Y, Z\} (Figure 4.4). The 37-marker protocol used is illustrated in Figure 4.6 and in Table 4.1 the anatomical points used to place the markers are listed.

The planar coordinates \( \tilde{q} \) are obtained from the selected X and Z components of the 3D joints which are obtained from the 3D markers captures [32]. Skin motion, muscle deformation and 2D simplification leads to violations of the kinematic constraints. In order to solve these violations, a new set of coordinates \( q \) imposing the kinematic consistency at position level, can be calculated by means of a minimization problem:

\[
\min_{q} V = \frac{1}{2} (q - \tilde{q})^T W (q - \tilde{q}) \quad \text{s.t.} \quad \Phi(q) = 0
\]  

(4.3)

where \( W \) is a diagonal weighting matrix that allows assigning different weights to the coordinates depending on their expected errors [32]. The minimization is only subjected to scleronomic constraints \( \Phi(q) = 0 \), and \( q \) includes only the end points of the
segments. Therefore, the kinematic data consistency can be guaranteed by using a so-called augmented Lagrangian minimization process as described in [2].

![3D view of the human skeleton with the set of 37 markers used](image)

**Figure 4.6:** 3D view of the human skeleton with the set of 37 markers used [32].

The set of the independent reference coordinates $\mathbf{z}_{\text{ref}}$ (the two Cartesian coordinates of the hip joint (P4), and the absolute and relative angles of the human body (see Figure 3.5)) is calculated from the kinematically consistent data set obtained above.

Usually, the obtained signals feature low-amplitude high-frequency noise, which is amplifying in case of numerical differentiations in order to calculate the corresponding velocities and accelerations. In [32] the displacement signals are filtered using an algorithm based on Singular Spectrum Analysis (SSA) [3]. The SSA decomposes the original signal into independent additive components with decreasing weight. Hence, the signal latent trend can be extracted from the inherent random noise introduced by the motion capture process.

The position histories are approximated using B-spline curves and the velocity and acceleration values are obtained by analytical spline differentiation techniques. It is described in detail in [32].
Table 4.1: Placement of the set of markers used [32].

<table>
<thead>
<tr>
<th>No.</th>
<th>Placement</th>
<th>No.</th>
<th>Placement</th>
</tr>
</thead>
<tbody>
<tr>
<td>M₁</td>
<td>Right metatarsal head V</td>
<td>M₈</td>
<td>Left metatarsal head V</td>
</tr>
<tr>
<td>M₂</td>
<td>Right calcaneus</td>
<td>M₉</td>
<td>Left calcaneus</td>
</tr>
<tr>
<td>M₃</td>
<td>Right lateral malleolus</td>
<td>M₁₀</td>
<td>Left lateral malleolus</td>
</tr>
<tr>
<td>M₄</td>
<td>Right tibial tuberosity</td>
<td>M₁₁</td>
<td>Left tibial tuberosity</td>
</tr>
<tr>
<td>M₅</td>
<td>Right lateral femoral epicondyle</td>
<td>M₁₂</td>
<td>Left lateral femoral epicondyle</td>
</tr>
<tr>
<td>M₆</td>
<td>Right femoral greater trochanter</td>
<td>M₁₃</td>
<td>Left femoral greater trochanter</td>
</tr>
<tr>
<td>M₇</td>
<td>Right ASIS</td>
<td>M₁₄</td>
<td>Left ASIS</td>
</tr>
<tr>
<td>M₁₅</td>
<td>Sacrum</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M₁₆</td>
<td>Right acromion in the shoulder girdle</td>
<td>M₂₃</td>
<td>Left acromion in the shoulder girdle</td>
</tr>
<tr>
<td>M₁₇</td>
<td>Right deltoid tuberosity</td>
<td>M₂₄</td>
<td>Left deltoid tuberosity</td>
</tr>
<tr>
<td>M₁₈</td>
<td>Right lateral humeral epicondyle</td>
<td>M₂₅</td>
<td>Left lateral humeral epicondyle</td>
</tr>
<tr>
<td>M₁₉</td>
<td>Middle of right forearm</td>
<td>M₂₆</td>
<td>Middle of left forearm</td>
</tr>
<tr>
<td>M₂₀</td>
<td>Right radial styloid in the wrist</td>
<td>M₂₇</td>
<td>Left radial styloid in the wrist</td>
</tr>
<tr>
<td>M₂₁</td>
<td>Right metacarpal head V</td>
<td>M₂₈</td>
<td>Left metacarpal head V</td>
</tr>
<tr>
<td>M₂₂</td>
<td>Right metacarpal head II</td>
<td>M₂₉</td>
<td>Left metacarpal head II</td>
</tr>
<tr>
<td>M₃₀</td>
<td>1ˢᵗ vertebra of the thoracic spine</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M₃₁</td>
<td>Right side of the head</td>
<td>M₃₃</td>
<td>Left side of the head</td>
</tr>
<tr>
<td>M₃₂</td>
<td>Top of the head</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M₃₄</td>
<td>Right metatarsal head I</td>
<td>M₃₆</td>
<td>Left metatarsal head I</td>
</tr>
<tr>
<td>M₃₅</td>
<td>Right distal phalange of the third toe</td>
<td>M₃₇</td>
<td>Left distal phalange of the third toe</td>
</tr>
</tbody>
</table>

Hence, the required kinematic data set \((\mathbf{z}_{\text{ref}}, \dot{\mathbf{z}}_{\text{ref}}, \ddot{\mathbf{z}}_{\text{ref}})\) to perform the IDA is completely known and its consistency with the rigid body assumption of the multibody system is guaranteed at position, velocity and acceleration levels. Figure 4.7 shows the reference motion of the independent coordinates of both legs as well as the hip joint positions and the absolute trunk angle.

4.2.3 Body Segment Parameters of the Human Body Model

As it can be seen in Figure 4.1, body segment parameters (BSP) have to be obtained in order to complete the biomechanical model definition.

As explained in [32], recommended anthropometric parameters from literature are used for the upper body and a reduced set of anthropometric measurements have been taken on the subject in order to extract the inertial properties of the lower limb segments.
Figure 4.7: Reference motion of the lower limb.

(Figure 4.8). The anthropometric data are presented in Table 3.2. The moment of inertia of the segments is referred to their COM.

Figure 4.8: Anthropometric measurements of the lower extremities [32].
The choice of the parameters affects the results of the inverse and forward dynamic analysis. In [32] a sensitivity analysis on the influence of BSP errors on the inverse dynamics results is presented.

### 4.2.4 Driving Torques of the Human Body Model

Finally, all necessary data and parameters are obtained and the IDA can be performed as explained in this chapter. Figure 4.9 shows the plots of the lower limb forces and torques. As mentioned before, these forces and torques are not the ones acting on the real human being since the contact between the human body and the environment is characterized, in this thesis, at the hip joint. However, they represent the input values for the FDA, which will be explained in the following chapter.

![Figure 4.9: Lower limb joint forces and torques.](image)

The IDA has been discretized in this thesis to a time step of 1ms. Therefore, the actuating forces and torques are available with the prior time step of 10 ms and a new time step of 1ms for a time period of 1,56s.
5 Forward Dynamic Analysis

In the Forward Dynamic Analysis (FDA), the motion of a multibody system is obtained for a specific time period as a consequence of the applied forces/torques and given initial conditions. The advantage of the FDA is to enable the simulation and prediction of the actual behaviour of the system since the motion is always the result of the forces and torques that produce it [13].

In forward dynamics the nonlinear ODE system from Eq.(3.50) is numerically integrated, starting from an initial condition. The mathematical integration process is computationally intensive. Hence, it is very important to choose an efficient numerical integration method for dealing with and solving this problem [13].

Consequently, in this chapter, the three introduced integration methods from Section 3.4.4 (ode23, ode4 and trapezoidal rule) will be analysed and compared in order to achieve the most efficient forward dynamic simulation.

The required input forces and torques of the FDA to predict the system’s motion are provided from the IDA in the previous chapter. Because the IDA was carried out in order to determine the actuating forces and torques \( Q_{\text{IDA}} \) that produce a certain reference motion \( z_{\text{ref}} \), it is expected that the FDA of the multibody system, using these actuating forces and torques, will ideally reach the same motion as in the reference (Figure 5.1) [32].

But, as pointed out in the literature, because of numerical errors in the integration process of the FDA and the unstable character of the human motion, the obtained motion in the FDA \( z_{\text{FDA}} \) does not equal the reference motion \( z_{\text{ref}} \), which is the input of the IDA [32].

Figure 5.2 shows the general structure of an FDA representing Eq.(4.1). Besides using the IDAs output, the driving forces and torques \( Q_{\text{IDA}} \), the process requires exact initial values of the independent coordinates \( z_{\text{ref}}(t = 0) \) and the independent velocities \( \dot{z}_{\text{ref}}(t = 0) \) as well as approximated initial values of the dependent coordinates \( q_{d}^{0}(t = 0) \). Like the IDA, the FDA uses the BSP from Table 3.2.

Since the FDA is not using the motion as an input, the constraint equations do not contain any driver constraints. Therefore, as explained in Section 3.3 the total number of constraints equals the number of the dependent coordinates. As described in Section 3.3, in order to obtain the generalized coordinates \( q \), the generalized velocities
Figure 5.1: Ideal closed loop behaviour of IDA and FDA. Using the IDA’s output, the driving forces and torques $\mathbf{Q}_{\text{IDA}}$ as an input for the FDA, the FDA’s output, the motion $\mathbf{z}$ should be coincident with the reference input of the IDA.

Figure 5.2: Block diagram of forward dynamic analysis.

$\dot{\mathbf{q}}$ and the generalized accelerations $\ddot{\mathbf{q}}$, the independent coordinates $\mathbf{z}$ as well as their derivatives $\dot{\mathbf{z}}$ and $\ddot{\mathbf{z}}$ have to be known (see Eqs.(3.8), (3.10) and (3.12)).

In this chapter, forward dynamics techniques are applied to both models used in this thesis (four-bar linkage model and two-dimensional human body model). The integration methods will be compared for the four-bar linkage model and the FDA results of both models will be discussed by comparing them to the reference motion.
In order to compare the different performances, the Root-Mean-Square Error (RMSE) is introduced in Eq. (5.1). It measures the difference between a reference independent coordinate \( z_{\text{ref},i} \) and the corresponding predicted independent coordinate by means of FDA \( z_{\text{FDA},i} \). The mean of the squared error is calculated over the total simulation time \( T \) with a total number of time steps \( K \).

\[
\text{RMSE}(z_i) = \sqrt{\frac{1}{K} \sum_{t=0}^{T} (z_{\text{ref},i} - z_{\text{FDA},i})^2} \tag{5.1}
\]

where the total number of time steps can be calculated with:

\[
K = \frac{T}{\Delta t} + 1 \tag{5.2}
\]

Finally, the Normalized Root-Mean-Square Error (NRMSE) is obtained, which refers each RMSE value to the range of the independent coordinate to enable a comparison in %.

\[
\text{NRMSE}(z_i) = \frac{\text{RMSE}(z_i)}{z_{\text{max},i} - z_{\text{min},i}} \cdot 100 \tag{5.3}
\]

### 5.1 FDA applied to Four-Bar Linkage Model

As explained before, the numerical integration process is computationally intensive and the integration method has to be chosen carefully in order to enable an efficient simulation. The different integration methods, introduced in Section 3.4.4 (variable-step solver ode23, fixed-step solver ode4 and fixed-step trapezoidal solver), which are used in the forward dynamic simulation of the four-bar linkage model, will be analysed and the results will be compared with respect to their NRMSE and simulation time.

The FDA of the four-bar linkage model is simulated with a total time \( T \) of 10s and an initial time step \( \Delta t \) of 10ms as well as with a reduced time step \( \Delta t \) of 1ms. The first integration method, which will be investigated, is the variable-step solver ode23. Because this solver works with variable time steps, it is necessary to develop a technique in order to add the driving torque of the angle \( \alpha \) provided from the IDA. This torque is evaluated in the IDA at both discrete time steps 10ms and 1ms but not as a function of time. Since the ode23 solver is choosing the time steps as necessary, it is important to find appropriate values of the torque at these specific time steps. In order to add the torque, two approaches have been developed. In the first approach the actuating torque is added by means of linear interpolation of the torque between two nearby time steps. The second approach approximates the torque history using a B-spline curve. A De Boor
algorithm is used with a tolerance of $10^{-9}$, which guarantees that the RMSE between the IDA’s output and the new torque is less than 0.001%.

Figure 5.3 shows the motion of the independent coordinate $\alpha$ for the forward dynamic simulation of the four-bar linkage model with the simulation time steps $\Delta t$ of 10ms (solid blue) and 1ms (dashed blue) using the variable-step solver ode23 within a specified tolerance of $10^{-9}$ and the linear interpolation method in order to add the driving torque of the IDA. The independent coordinate $\alpha$ of the FDA is compared with the reference angle motion (solid black), which is defined in Figure 4.2. It can be seen that with a time step of 10ms the FDA is able to reproduce the reference motion of the independent coordinate $\alpha(t)$ properly at the beginning of the simulation but after circa 5s the simulation starts to drift away and the error of the angle increases. The reasons for the divergence, as explained before, are the numerical errors during the integration process and the unstable character of the human motion.

Reducing the time step to 1ms improves the performance of the FDA significantly since the values of the added torques are more precisely. The angle $\alpha$ follows the reference motion $\alpha(t)$ circa 9s, but then it starts to differ as well. Moreover, the simulation time of the FDA, using the reduced time step, increases excessively. Table 5.1 shows the simulation results of this chapter with respect to the NRMSE and the simulation time.

Instead of adding the driving torque with the linear interpolation approach, the B-spline curve approach has an additional benefit regarding the NRMSE but the simulation time is even higher (see Table 5.1).
In addition, the fixed-time step integration methods are analysed. The ode4 solver and the trapezoidal solver do not require a special technique to add the driving torque. It can be added at each time instance since it has been already evaluated at the time step which the fixed-step integration solver is using.

The forward dynamic simulation results of both integration methods ode4 (in red) and trapezoidal rule (in green) can be seen in Figures 5.4 and 5.5 respectively for both time steps $\Delta t$ 10ms (solid) and 1ms (dashed).

**Figure 5.4:** FDA results using the fixed-step solver ode4 compared with the reference angle motion $\alpha(t)$ (time steps $\Delta t$: 10ms and 1ms).

**Figure 5.5:** FDA results using the fixed-step solver trapezoidal rule compared with the reference angle motion $\alpha(t)$ (time steps $\Delta t$: 10ms and 1ms).
First of all, it can be noticed that the forward dynamic simulation using the fixed-step solver ode4 or the trapezoidal rule performs very accurately at the beginning of the simulation but after a few seconds it drifts away, as seen before with the ode23 solver (Figure 5.3).

Reducing the time step $\Delta t$ to 1ms yields better results regarding the NRMSE but still worse than using the variable-step solver ode23. However, the simulation times for the fixed-step solvers are lower than the simulation time of the ode23 solver (see Table 5.1).

Figure 5.6 shows the comparison between the three solvers using the time step $\Delta t$ of 1ms. It is obvious that the variable-step solver ode23 shows the best performance though it starts to differ at the end of the simulation time as well.

The corresponding forward dynamic simulation results of the angular velocity $\dot{\alpha}$ and the angular acceleration $\ddot{\alpha}$ using the ode23 solver with a time step $\Delta t$ of 1ms and adding the driving torque with a B-spline curve are shown in Figure 5.7. Table 5.1, as mentioned before, shows a summary of the compared integration methods with respect to the NRMSE of the independent coordinate $\alpha$ and their derivatives and the simulation time of the FDA.

In summary, the variable-step solver features a better performance than the fixed-step solvers for a simulation time step of 1ms. Even though the integration process is more accurate, the simulation time increases using the ode23 solver. Within the both fixed-
Figure 5.7: FDA results using the variable-step solver ode23 (Q_{IDA} interpolated with B-spline curve) compared with reference angular velocity $\dot{\alpha}(t)$ and the reference angular acceleration $\ddot{\alpha}(t)$ (simulation time step: $\Delta t = 1\text{ms}$).

Table 5.1: Comparison of NRMSE and simulation time dependent on integration method and time step used for the forward dynamic analysis of the four-bar linkage model.

<table>
<thead>
<tr>
<th>Integration method</th>
<th>Added Torque</th>
<th>Time step</th>
<th>NRMSE($\alpha$)</th>
<th>NRMSE($\dot{\alpha}$)</th>
<th>NRMSE($\ddot{\alpha}$)</th>
<th>Sim. time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>[ms]</td>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[s]</td>
</tr>
<tr>
<td><strong>ode23</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>lin. interp.</td>
<td>10</td>
<td>10</td>
<td>41,56</td>
<td>69,84</td>
<td>14,61</td>
<td>41,29</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1,68</td>
<td>2,56</td>
<td>1,22</td>
<td>184,58</td>
</tr>
<tr>
<td>B-splines</td>
<td>10</td>
<td>10</td>
<td>21,95</td>
<td>13,33</td>
<td>8,9132</td>
<td>65,66</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1,04</td>
<td>1,51</td>
<td>0,74</td>
<td>229,73</td>
</tr>
<tr>
<td><strong>ode4</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>discretized</td>
<td>10</td>
<td>10</td>
<td>41,57</td>
<td>69,86</td>
<td>14,63</td>
<td>17,88</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>20,77</td>
<td>16,55</td>
<td>11,24</td>
<td>171,04</td>
</tr>
<tr>
<td><strong>trapezoidal</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>discretized</td>
<td>10</td>
<td>10</td>
<td>633,64</td>
<td>461,30</td>
<td>352,81</td>
<td>18,38</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>29,91</td>
<td>54,91</td>
<td>8,39</td>
<td>169,95</td>
</tr>
</tbody>
</table>
step solvers, the ode4 solver shows a better performance with respect to the error expansion in time than the trapezoidal solver.

The selection of the appropriate integration method is therefore dependent on accuracy and computational effort. The forward dynamic simulation performs the lowest errors, regardless of the computational effort, using the ode23 solver with the B-spline curve in order to add the driving torque and a simulation time step of 1ms. In order to save computational time, and thus permitting higher NRMSE of the FDA, other integration methods can be taken into account.

However, none of the integration methods is able to perform a stable reproduction of the reference motion in a forward dynamic simulation during all the simulation time $T$. In Chapter 6, the unstable behaviour of the FDA will be improved by using control strategies in order to stabilize the performance.

### 5.2 FDA applied to 2D Human Body Model

The two-dimensional human body model with 14 DOF is in comparison to the four-bar linkage model more complex. Consequently, the forward dynamic simulation of this model is even more unstable and not capable to simulate the full gait cycle with a total time $T$ of 1,56s.

As explained in Section 3.4.4 and pointed out before with the four-bar linkage model, the FDA performance is highly unstable due to numerical integration errors and the unstable character of human gait. Even reducing the time step to 1ms and choosing the best integration method, which is the variable-step solver ode23, the simulation still is completely unstable. Figure 5.8 shows the joint flexion-extension angles (in blue) and their reference values (in black) for the ankle, knee and hip joints of both legs.

It can be seen that after 0,08s ($\approx 5,1\%$ of the gait cycle) the simulation results already start to differ and the obtained motion does not correspond to normal human gait.

Hence, it can be said that the ideal loop from Figure 5.1 cannot be ensured and the FDA cannot achieve the desired motion without any control effort. Possible control strategies in order to stabilize the FDA will be discussed in the following chapter.
Figure 5.8: FDA results for human body model of ankle, knee and hip flexion angles compared to the reference angles.
6 Control Strategies

The FDA results of the multibody systems, used in this thesis, were illustrated in Chapter 5. It has been explained that the forward dynamic simulation cannot be stable during the total simulation time $T$. Even though the most appropriate integration method has been chosen, the motion of the four-bar linkage model differs from the reference motion after few seconds. And in the case of the human body model, a forward dynamic simulation without any control can just reproduce the reference motion for 5, 1% of the human gait cycle. Moreover, the forward dynamic simulation does not include any uncertainties of the multibody system, but errors can occur in the input data as well as in the model parameters [32]. In Chapter 7 it will be shown that considering those errors affects the results of the FDA significantly. Without a control strategy the forward dynamic simulation cannot react to disturbances and the performance degrades even more.

Therefore, in order to achieve the same motion in the FDA as given in the reference data, i.e. to guarantee the correctness of Figure 5.1, a control system needs to be implemented in the forward dynamic simulation to enable accuracy and a stable system performance.

Figure 6.1 shows a general block diagram of a closed loop system. The closed loop system consists of a plant (system to be controlled), sensors to evaluate the system state, actuators to translate the control output into meaningful data to the plant and a controller to guide the system in order to follow a certain reference input [29].

![Block diagram of a closed-loop system with output feedback.](image)

**Figure 6.1:** Block diagram of a closed-loop system with output feedback.

The control variable is the system’s output $y(t)$. It is measured by several sensors which provide the processed signal $x(t)$. The error $e(t)$, which is the difference between the reference value $r(t)$ and the sensor measurement $x(t)$, is fed to the controller in order
to reduce the error and bring the output of the system to the desired reference value. The output of the controller \( u(t) \) is processed by the actuator, which provides the input value \( v(t) \) for the plant. Commonly actuator and plant are combined to one process [29].

Besides in engineering, feedback control systems can be found in various non-engineering fields as well. The human body, for example, is a highly advanced feedback control system [29]. Body temperature and blood pressure are kept constant and human motion can be performed without stability problems by means of physiological feedback. Therefore feedback is a vital function which makes the human body relatively insensitive to external disturbances and functioning properly in a changing environment [29].

From an engineering point of view, many control systems are existing. Chapter 2 discusses available control strategies and their usual applications in biomechanics. This thesis uses the application of a Proportional Derivative (PD) controller to improve the performance of the forward dynamic simulation. Moreover, the Computed Torque Control (CTC) with feedback linearization is introduced and analysed in this chapter.

In the following, the second order ODE system from Eq.(4.1) will be expressed in nonlinear state space representation. The multibody system which needs to be stabilized, is a nonlinear ‘square’ Multiple Input Multiple Output (MIMO) system (i.e. same number of inputs as outputs). In general, nonlinear square MIMO systems of second order can be expressed in nonlinear state space representation as:

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

(6.1)

where \( u \in \mathbb{R}^m \) is the system input, \( y \in \mathbb{R}^m \) the system output and \( x \in \mathbb{R}^{2n} \) the state vector containing \( n \) coordinates and \( n \) velocities [40]. The vector fields \( f(x) \) and \( h(x) \) and the matrix \( g(x) \) are nonlinear functions of the state space vector \( x \) which is described as:

\[
x = \begin{bmatrix} x_1, \ldots, x_n, \dot{x}_1, \ldots, \dot{x}_n \end{bmatrix}^T
\]

(6.2)

For a multibody system, which is explained in Chapter 3, the state vector contains the independent coordinates \( z \in \mathbb{R}^{n_i} \) as well as the independent velocities \( \dot{z} \in \mathbb{R}^{n_i} \) (\( n_i \) is the number of independent coordinates, see Section 3.3).

\[
x = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1, \ldots, z_{n_i}, \dot{z}_1, \ldots, \dot{z}_{n_i} \end{bmatrix}^T
\]

(6.3)

The input vector \( u \) contains the forces and torques that are needed to generate the reference motion. Therefore, it is a sum of \( m \) forces and torques generated through the
6 Control Strategies

IDA ($Q_{\text{IDA}}$), and $m$ forces and torques provided by the controller ($Q_C$). The input vector can be expressed as:

$$ u = Q_{\text{in}} = Q_{\text{IDA}} + Q_C $$

(6.4)

The output vector $y$ is defined as the vector of independent coordinates $z$. It can be seen that, in this case, the number $m$ of inputs and outputs equals the number $n_i$ of the state space positions. Using Eq.(4.1), the nonlinear state space representation for a second order multibody system in matrix-R formulation can be described as:

$$
\begin{align*}
\dot{x} &= \left\{ \ddot{z}, \dot{z} \right\} = \left\{ \hat{z}_1, \ldots, \hat{z}_{n_i} \right\}^T (R^T M R)^{-1} \left[ R^T (Q_0 - M \dot{R} \dot{z}) \right] + \left[ \begin{array}{c} 0 \\ (R^T M R)^{-1} \end{array} \right] Q_{\text{in}} \\
y &= z = \left\{ z_1, \ldots, z_{n_i} \right\}^T
\end{align*}
$$

(6.5)

In the following sections, two control strategies (PD control and CTC control) will be introduced and applied to both models (four-bar linkage model and two-dimensional human body model).

### 6.1 PD Control

The first control approach is a Proportional Derivative (PD) control. Additionally to the actuating forces and torques from the IDA, the PD control provides the necessary forces and torques to reproduce the desired reference motion. The input vector with the driving forces and torques can be achieved by:

$$ Q_{\text{in}} = Q_{\text{IDA}} + Q_{\text{PD}} $$

(6.6)

In order to stabilize the forward dynamic simulation, the joints are spanned with linear or linear rotational springs and dampers (depending on whether the independent coordinates are position or angular coordinates, respectively). With the joint stiffness matrix $K_p$ and the damping matrix $K_d$, the vector of joint forces and torques applied by the PD controller can be calculated according to:

$$ Q_{\text{PD}} = K_p e_z + K_d \dot{e}_z $$

(6.7)

where $e_z$ is the tracking error of the independent coordinates and $\dot{e}_z$ is the derivation of the tracking error with respect to time, respectively:

$$ e_z = \left\{ z_{\text{ref}} - z \right\}^T \quad \text{and} \quad \dot{e}_z = \left\{ \dot{z}_{\text{ref}} - \dot{z} \right\}^T $$

(6.8)

Using the state vector and the reference independent coordinates as well as the reference velocities, which are described in Chapter 4, the control joint forces and torques
\( \mathbf{Q}_{PD} \) can be calculated and added to the forces and torques \( \mathbf{Q}_{IDA} \) which are provided by the IDA. Figure 6.2 shows the general block diagram of the nonlinear MIMO system using a PD control.

\[
x = \begin{bmatrix} \dot{z} \\ z \end{bmatrix}, \quad \mathbf{y} = \mathbf{h}(\mathbf{x}) + \mathbf{Q}_{in} + \mathbf{Q}_{PD} + \mathbf{Q}_{IDA}
\]

**Figure 6.2:** Block diagram of the closed-loop nonlinear MIMO system using PD control.

The joint stiffness matrix \( \mathbf{K}_p \) is called the proportional gain matrix. It is amplifying the tracking error with an adjustable gain matrix and determines the reaction to the current error. Large gain values lead to a higher response of the controller to potential disturbances. However, if this gain parameters are chosen too high, the system might oscillate and become unstable.

The damping matrix \( \mathbf{K}_d \) is called the derivative gain matrix. The derivative part of the PD control anticipates future behaviour of the tracking error since the response of the derivative component is proportional to the rate of change of the error signal [29]. The adjustable derivative time in matrix \( \mathbf{T}_d \) \( (\mathbf{K}_d = \mathbf{K}_p \mathbf{T}_d) \) is the time interval by which the rate action advances the effect of the proportional control action [29]. Hence, the derivative gain matrix has an anticipating correcting effect. It can reduce the magnitude of overshoot, eliminates oscillations and enables the system to respond faster to external or internal disturbances.

Mostly, in order to improve the performance, the PD controller is extended by using an integral term which integrates the error over time to overcome the steady-state error. However, the integral action may cause overshoot, oscillation or even instability problems and is not used in this thesis [29].

The gain matrices, which are diagonal, can be expressed as:

\[
\mathbf{K}_p = \begin{bmatrix} k_{p,1} & 0 & \cdots & 0 \\ 0 & k_{p,2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & k_{p,n} \end{bmatrix}, \quad \mathbf{K}_d = \mathbf{K}_p \begin{bmatrix} t_{d,1} & 0 & \cdots & 0 \\ 0 & t_{d,2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & t_{d,n} \end{bmatrix}
\]

In order to avoid oscillations and minimize overshoot, the gain parameters \( k_{p,i} \) and \( t_{d,i} \) have to be chosen carefully.
The adjustment of the control parameters is carried out using a wrong initial state vector. As explained in Chapter 5 and shown in Figure 5.2, the FDA requires exact initial values of the independent coordinates $z_{\text{ref}}$ and independent velocities $\dot{z}_{\text{ref}}$. Providing wrong initial values, the PD control should be able to react to the fault case and guide the motion of the system until it follows again the reference motion. The magnitude of the error and the time to correct the error can be improved by adjusting the control parameters manually. With this technique, appropriate control parameters have been chosen for the PD control of the four-bar linkage model and the gained information will be transferred to the human body model.

The following sections analyse and discuss the application of the PD control to both models (four-bar linkage model and two-dimensional human body model).

### 6.1.1 PD Control applied to the FDA of the Four-Bar Linkage Model

The four-bar linkage model is a system with one DOF (see Section 3.2.1). Hence, the system's output $z$ is a scalar containing the independent coordinate $\alpha$ and the system's input $Q_{\text{in}}$ is a scalar torque which is driving the system's motion. A system with just one DOF is called Single Input Single Output (SISO) system. Consequently, the control torque of the PD controller $Q_{\text{PD}}$, and therefore the joint stiffness matrix $K_p$ and the damping matrix $K_d$ are scalars as well.

In order to apply a PD control to the forward dynamic simulation of the four-bar linkage model an integration method has to be chosen. As shown in Chapter 5, the integration method can influence the FDA results significantly. Three different numerical integration solvers were introduced in Section 3.4.4 and their performances were compared in Chapter 5. In order to be able to compare the discussed control strategies in the next Chapter, it is important to chose the most appropriate integration method and to use it consistently for each model.

In Chapter 5 the advantage of the variable-step solver ode23 is pointed out. In comparison to the fixed-step solvers, the variable-step solver guarantees a consistent level of accuracy during all time. Sharp changes in the solution can be coped by the variable-step solver by reducing the time step, where fixed-step solvers may overrun the problematic zone [36]. Hence the variable-step solver is more reliable and will be used for the four-bar linkage model. Though, it has to be mentioned that in the case of real-time applications, it is recommended to use a fixed-step solver [26].

Thus, the variable-step solver ode23 with a simulation time step of 1ms and a specified tolerance of $10^{-9}$ is chosen to be the appropriate integration method for the forward dynamic simulation of the four-bar linkage model. In order to decrease the simulation time, the torque for the angle $\alpha$ provided by the IDA will be added by linear inter-
polation. Even though the B-spline approach shows better results (see Table 5.1), the performance is improving insufficiently while the simulation time increases significantly using the PD control.

As explained before, the adjustment of the scalar control parameters $K_p$ and $T_d$ is carried out using a wrong initial state vector $x(t = 0)$, which contains the initial independent coordinate $\alpha_{ref}(t = 0)$ and the initial independent velocity $\dot{\alpha}_{ref}(t = 0)$. Instead of the exact initial state vector of $x_{exact}(t = 0) = \left\{ \frac{-\pi}{4}, 0 \right\}^T$ (see Figure 4.2), wrong initial values are used $\left\{ x_{false}(t = 0) = \{0, 1\}^T \right\}$. Figure 6.3a shows in red the system performance of the FDA without applying a control system, compared to the reference motion in black. The system's motion is not able to reproduce the desired reference motion $\alpha(t)$ from the beginning. The PD control instead, is able to react to the wrong initial values and guide the motion back to the desired trajectory (see Figure 6.3b).

![Tracking error of the system without using a control.](image1.png)

![Tracking error of the system using PD control.](image2.png)

**Figure 6.3:** Tracking error of $\alpha$ for forward dynamic simulation of four-bar linkage model using a wrong initial state vector. a) Without control. b) Using PD control.

The control parameters influence the magnitude of error as well as the time it takes to correct the initial error. In order to determine the control parameters, two characteristics are defined: the NRMSE from Eq.(5.3) and the settling time $t_{1\%}$. The settling time is defined as the time period until the absolute tracking error is smaller than 1% of the range of the motion $\alpha(t)$. Hence, the settling time is expressed as the time when:

$$\left| \alpha(t) - \alpha \right| \leq 0,01 \cdot (\alpha_{max}(t) - \alpha_{min}(t))$$  \hspace{1cm} \text{(6.10)}

These two characteristics can be improved by adjusting the control parameters $K_p$ and $T_d$ manually. Several sets of control parameters are chosen in order to compare their performance for correcting the initial error.
In Table 6.1 two selected sets of control parameter adjustments are described and analysed. The performances of these two PD controllers for the forward dynamic simulation of the four-bar linkage model with wrong initial values are shown in Figure 6.3b.

**Table 6.1:** Comparison of NRMSE, settling time and simulation time dependent on adjusted control parameters $K_p$ and $T_d$ of PD controller (using wrong initial values).

<table>
<thead>
<tr>
<th>Control</th>
<th>Control parameters</th>
<th>NRMSE($\alpha$)</th>
<th>Settling time</th>
<th>Simulation time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$K_p$ [Nm]</td>
<td>$T_d$ [s]</td>
<td>[%]</td>
<td>$t_{1%}$ [s]</td>
</tr>
<tr>
<td>PD1</td>
<td>10.000</td>
<td>0.01</td>
<td>6.4609</td>
<td>1.45</td>
</tr>
<tr>
<td>PD2</td>
<td>100.000</td>
<td>0.01</td>
<td>2.3728</td>
<td>0.28</td>
</tr>
</tbody>
</table>

It can be seen that the PD2 controller, which has a higher parameter $K_p$, is able to correct the initial error with a settling time $t_{1\%}$ of 0.28s and an NRMSE of 2.3728%, therefore faster and more accurately than the PD1 controller. Choosing values higher than the ones in Table 6.1 leads to instabilities and the system performance degrades. Hence, the scalar control parameters of the PD control for the forward dynamic simulation of the four-bar linkage model are chosen as $K_p = 100.000 \text{Nm/rad}$ and $T_d = 0.01$ s.

Figure 6.4a shows the trajectory error of the FDA using the PD2 controller. Note, that the initial state vector now is given correctly. Finally, the four-bar linkage model follows the prescribed motion during all the analysis and the controlled angle $\alpha$ is concurrent with the reference motion. The PD controller achieves a high accuracy of the motion with a tracking error in the order of $10^{-7}$ degrees for the four-bar linkage model. The control torque $Q_{pd}$, illustrated in Figure 6.5, is in the order of $10^{-3}$Nm. It can be seen that the control torque is needed consistently during all the simulation. Furthermore, the angular velocity and angular acceleration follow the desired values as well very accurately with errors in the order of $10^{-5}\text{deg/s}$ and $10^{-3}\text{deg/s}^2$, as it can be seen in Figure 6.4b.

### 6.1.2 PD Control applied to the FDA of the 2D-Human Body Model

The gained information of the PD control approach has been transferred to the two-dimensional human body model. The same integration method (the variable-step solver ode23 with a simulation time step of 1ms and a specified tolerance of $10^{-9}$) is used for the forward dynamic simulation and the control parameter adjustments are adapted to the 14 DOF system ($k_{p,i} = 100.000$ [N/m] or [Nm/rad] and $t_{d,i} = 0.01$ [s] for
Application of different Control Strategies to the FD Simulation of Human Gait

\[ \alpha(t) \]

\[ \alpha_{PD} \]

\[ (\dot{\alpha}(t) - \dot{\alpha}) \] [deg/s]

\[ (\ddot{\alpha}(t) - \ddot{\alpha}) \] [deg/s²]

\[ \dot{Q}_{PD} \] [Nm]

\[ Q_{PD} \] [Nm]

\[ i = 1, \cdots, 14 \]. But the simulation is not able to achieve successful results. Even though the PD control performs highly accurate for the four-bar linkage model, it performs very poorly in the case of the human body model. The simulation stops after \( 5.62 \cdot 10^{-5} \) s because the integration tolerances cannot be met without reducing the time step below the smallest allowed value of \( 1.08 \cdot 10^{-19} \) s.

Obviously the PD control does not generally provide optimal motion trajectory control. The fundamental difficulty of using a PD control is that the feedback signals are multiplied with constant parameters. The control system has no direct knowledge of the process but the overall performance is reactive and compromises to changes. Furthermore, this control strategy has difficulties in the presence of non-linearities (changing process behaviour) and has a lag in responding to large disturbances. Moreover, the control parameter gain matrices are diagonal and do not take possible coupling among
DOF into account. All these difficulties do not influence the four-bar linkage model with one DOF, but all the more they affect the human body model with 14 DOF.

The same results have been achieved in [40], where a simple PD controller is used to reach a desired position of a two DOF system but in order to follow a desired trajectory the controller cannot perform effectively.

6.2 Computed Torque Control with Feedback Linearization

The Computed Torque Control (CTC) with feedback linearization is an approach to nonlinear control design and has been applied successfully in various practical control problems, such as the control of helicopters, high performance aircraft, industrial robots and biomedical devices [40], [8].

The main idea of this approach is to algebraically transform a nonlinear system into a linear one, so that linear control techniques can be applied. This technique is called feedback linearization. The transformation of the system into an equivalent system of simpler form is also used in the development of robust or adaptive nonlinear control [40], [35].

Due to this transformation, the new system can be controlled with $m$ linear independent controllers. With the controller output, the necessary control torque $Q_C$ for the nonlinear system can be calculated. Therefore the control technique is commonly referred to the computed torque control in robotics [40].

6.2.1 Feedback Linearization

The approach of the feedback linearization took a rapid development in the early 80’s. It is a powerful tool for path generation and may be the only "general" approach for nonlinear systems with predictive control [25]. Feedback linearization differs entirely from conventional linearization (i.e. Jocobian linearization, Taylor approximation etc.) in that it linearizes the system by exact state transformations [19], [40]. The important advantage for this thesis, is that it can be extended to MIMO systems with a decoupling condition, which will be explained in the following.
The nonlinear square MIMO system from Eq. (6.1) is linear referring to the input vector \( \mathbf{u} \). Considering the single entries of the input and output vectors, the nonlinear system can be rewritten as:

\[
\begin{align*}
\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \sum_{i=1}^{m} \mathbf{g}_i(\mathbf{x}) \mathbf{u}_i \\
\mathbf{y} &= \{h_1(\mathbf{x}), \ldots, h_m(\mathbf{x})\}^T
\end{align*}
\] (6.11)

where \( \mathbf{g}_i(\mathbf{x}) \in \mathbb{R}^{2n} \) is a vector and \( h_i(\mathbf{x}) \) and \( \mathbf{u}_i \) are scalars.

The time derivative of the \( k^{th} \) output can be expressed as:

\[
\dot{y}_k = \frac{\partial h_k(\mathbf{x})}{\partial t} = \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial t} = \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) + \sum_{i=1}^{m} \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}_i(\mathbf{x}) \mathbf{u}_i
\] (6.12)

where \( L_f h_k(\mathbf{x}) \) and \( L_{g_i} h_k(\mathbf{x}) \) are called the Lie derivatives. They are the directional derivatives of \( h_k \) along the direction of the vectors \( \mathbf{f} \) and \( \mathbf{g}_i \) [40].

\[
L_f h_k(\mathbf{x}) = \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) \quad ; \quad L_{g_i} h_k(\mathbf{x}) = \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}_i(\mathbf{x})
\] (6.13)

Usually, if the order of the system is higher than one, \( L_{g_i} h_k(\mathbf{x}) \) is equal to zero. If every \( L_{g_i} h_k(\mathbf{x}) \) is equal to zero (i.e. for all \( i = 1, \ldots, m \)), the second time derivative of the \( k^{th} \) output has to be calculated:

\[
\ddot{y}_k = L_f^2 h_k(\mathbf{x}) + \sum_{i=1}^{m} L_{g_i} L_f h_k(\mathbf{x}) \mathbf{u}_i
\] (6.14)

where

\[
L_f^2 h_k(\mathbf{x}) = \frac{\partial L_f h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) \quad \text{and} \quad L_{g_i} L_f h_k(\mathbf{x}) = \frac{\partial L_f h_k(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}_i(\mathbf{x})
\] (6.15)

The relative degree \( r_k \) of each output is given by:

\[
L_{g_i} L_f^{j-1} h_k(\mathbf{x}) = 0, \quad \forall j < r_k
\] (6.16)

The \( k^{th} \) output has to be differentiated with respect to time as long as:

\[
L_{g_i} L_f^{r_k-1} h_k(\mathbf{x}) \neq 0
\] (6.17)
This means every output $y_k$ of the $m$ outputs needs to be differentiated with respect to time as long as at least one input value $u_i$ appears in the equation.

$$\dot{y}_k = L_f h_k(x) + \sum_{i=1}^{m} L_{gi} h_k(x) u_i$$

$$\ddot{y}_k = L_f^2 h_k(x) + \sum_{i=1}^{m} L_{gi} L_f h_k(x) u_i$$

$$\vdots$$

$$y_k^{(r_k)} = L_f^{(r_k)} h_k(x) + \sum_{i=1}^{m} L_{gi} L_f^{(r_k-1)} h_k(x) u_i$$

Hence with the input vector $u \in \mathbb{R}^m$, a new system of equations can be expressed as:

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} L_f^{(r_1)} h_1 \\ \vdots \\ L_f^{(r_m)} h_m \end{bmatrix} + \begin{bmatrix} L_{g1} L_f^{(r_1-1)} h_1 & \cdots & L_{gm} L_f^{(r_1-1)} h_1 \\ \vdots & \ddots & \vdots \\ L_{g1} L_f^{(r_m-1)} h_m & \cdots & L_{gm} L_f^{(r_m-1)} h_m \end{bmatrix} u$$

Note that $h_k$ is a function of the state vector $x$. The $(m \times m)$ matrix $B(x)$ is called the invertibility or input-output-decoupling matrix. Assuming the decoupling condition, that $B(x)$ is non-singular, i.e. fully ranked ($\text{rank}(B(x)) = m$), the input vector can be defined as:

$$u = B^{-1}(x)(v - a(x))$$

where $v \in \mathbb{R}^m$ is the synthetic input vector. Hence a decoupled set of equations can be obtained:

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix}$$

This set of equations is linear and can be controlled using linear control techniques. Figure 6.6 shows the block diagram of a closed loop system of a nonlinear plant using feedback linearization. It has to be mentioned that this approach has a number of difficulties and limitations. First of all, the full state has to be measured. Moreover, internal dynamic problems can occur if the relative degree $r_k$ of any output is smaller than the order of the system. Then the nonlinear system is only partly linearized and the stability of the internal dynamics has to be analysed [40]. Furthermore, this approach needs confidence in the model since no robustness is guaranteed in the presence of parameter uncertainty or unmodeled dynamics [40] (see Chapter 7). The control parameters may take very large values [25].
6.2.2 Computed Torque Control applied to Multibody Systems

Applying the feedback linearization approach to the second order multibody system in Eq.(6.5) and using Eq.(6.4), the derivatives of the $m$ outputs can be expressed as:

$$
\begin{bmatrix}
\ddot{z}_1 \\
\vdots \\
\ddot{z}_m
\end{bmatrix} = \left( R^T MR \right)^{-1} \left[ R^T (Q_0 - MR \dot{z}) + Q_{IDA} \right] + \left( R^T MR \right)^{-1} Q_{CTC}
$$

(6.22)

As explained at the beginning of this chapter, the number $m$ of inputs and outputs equals the number $n_i$ of the state space positions.

Using the decoupling condition explained before, the control vector $Q_{CTC}$, which contains forces and/or torques, is the input vector for the nonlinear system and can be computed by:

$$
Q_{CTC} = R^T MR (v - a(x))
$$

(6.23)

Hence, using Eq.(6.22) and Eq.(6.23) with the synthetic input vector $v \in \mathbb{R}^m$, the nonlinear system in Eq.(6.5) can be transformed into a new decoupled system, which is fully linear.

$$
\begin{bmatrix}
\ddot{z}_1 \\
\vdots \\
\ddot{z}_m
\end{bmatrix} = \begin{bmatrix}
v_1 \\
\vdots \\
v_m
\end{bmatrix}
$$

(6.24)

Figure 6.7 shows the block diagram of the feedback linearization approach applied to a multibody system in a closed loop.

The system of Eq.(6.24) has $m$ relative degrees ($r_1, \ldots, r_m$). The scalar $r = r_1 + \ldots + r_m$ is called the total relative degree. If the total relative degree $r$ equals the size of the
state vector \((2n)\), no internal dynamics occur \([40]\). In other words, each relative degree of freedom has to be equal to the order of the system. Since the system is of second order and all relative degrees \(r_k = 2\), as it can be seen in Eq.(6.22), the multibody system is fully linearized and thus internal dynamic problems are avoided.

### 6.2.3 Computed Torque Control Design for Multibody Systems

As it can be seen in Figure 6.7, the linearized system can be controlled by using a linear control approach. In the following section, the control design of the computed torque control approach will be explained.

As it is shown in Eq.(6.24), the new linear system contains \(m\) decoupled equations of second order. Hence, each independent coordinate can be treated separately. Therefore, in order to design the system's controller, it is sufficient to analyze just one equation of the system and to adopt the control adjustments of this equation to the other ones.

The linearized system for one output coordinate is given by:

\[
\ddot{z} = v
\]  

(6.25)

The system in Eq.(6.25) can be expressed in state space representation as:

\[
\begin{bmatrix}
\dot{z} \\
\ddot{z}
\end{bmatrix} = \begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix} \begin{bmatrix}
z \\
\dot{z}
\end{bmatrix} + \begin{bmatrix}
0 \\
1
\end{bmatrix} v
\]  

(6.26)

The eigenvalues of the matrix \(A\) - which represent the roots or poles of the system - can be calculated with the characteristic polynomial \([30]\):

\[
\left| (sI - A) \right| = s^2 = 0
\]  

(6.27)
where \( I \) is the identity matrix and \( s \) the Laplace operator representing the roots or poles. Eq.(6.27) shows that both roots are zero \( (s_1 = s_2 = 0) \). Hence, this system is unstable without an applied control approach [30].

An unstable system can be stabilized by a control technique. But before applying a suitable control strategy, the controllability of the system has to be proved. The system, expressed in Eq.(6.26), is completely state controllable if and only if the vectors \( B \) and \( AB \) are linearly independent, or the \((2 \times 2)\) controllability matrix \( CO \) is of rank 2 [29], where

\[
CO = [B|AB] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}
\]

(6.28)

Since the rank of the controllability matrix \( CO \) in Eq.(6.28) is 2, it is proofed that the unstable system in Eq.(6.26) is completely state controllable.

The synthetic input \( v \) is then chosen as [40]:

\[
v = \ddot{z}_{ref} + k_d \dot{e} + k_p e
\]

(6.29)

where \( k_d \) and \( k_p \) are the control parameters and the tracking error \( e \) and its derivative \( \dot{e} \) are expressed as:

\[
e = z_{ref} - z \quad \text{and} \quad \dot{e} = \dot{z}_{ref} - \dot{z}
\]

(6.30)

With \( \ddot{e} = \ddot{z}_{ref} - \ddot{z} \), the error dynamics can be expressed in state space representation [30]:

\[
\begin{bmatrix} \dot{e} \\ \ddot{e} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix}
\]

(6.31)

The eigenvalues of the matrix \( A_K \) - which represent the roots of the controlled system - can be calculated using again the characteristic polynomial:

\[
|sI - A_K| = s(s - k_d) + k_p = 0
\]

(6.32)

The roots can be expressed as functions of the control parameters \( k_d \) and \( k_p \):

\[
s_{1,2} = -\frac{k_d}{2} \pm \sqrt{\left(\frac{k_d}{2}\right)^2 - k_p}
\]

(6.33)

In order to achieve a stable closed loop system, two necessary and sufficient conditions for the design of the controller for each output \( z \) can be derived:
1. Stability: To obtain a stable system performance the real part of the roots $s_1$ and $s_2$ has to be negative.

\[-\frac{k_d}{2} + \sqrt{\left(\frac{k_d}{2}\right)^2 - k_p} < 0\]
\[\left(\frac{k_d}{2}\right)^2 - k_p < \frac{k_d^2}{4}\]
\[k_p > 0\]  

(6.34)

2. No overshoot: To avoid overshoot, the roots $s_1$ and $s_2$ should not contain an imaginary part.

\[\sqrt{\left(\frac{k_d}{2}\right)^2 - k_p} \geq 0\]
\[\left(\frac{k_d}{2}\right)^2 \geq k_p\]
\[k_d \geq \frac{2}{\sqrt{k_p}}\]  

(6.35)

Usually, as shown in [15] and [40], a critical damp response is used, i.e. $k_d$ is set to:

\[k_d = 2\sqrt{k_p}\]  

(6.36)

In order to choose the adjustable control parameter $k_p$ and calculate the second parameter $k_d$, an arbitrary pole placement technique is used. If all state variables are measurable and available for feedback and if the system is completely state controllable, poles (or roots) of the closed loop can be placed to any desired location through appropriate state feedback gain parameters $k_p$ and $k_d$ [29]. That means, by placing the poles $s_1$ and $s_2$, the gain parameters $k_p$ and $k_d$ result from the chosen poles (see Eq.(6.33)). The desired closed loop pole locations determine the speed and the damping of the response. The selection of the poles is a compromise between rapidity of the response of the error vector and the sensitivity to disturbances and measurement noises [29]. A fast response requires a large control signal [29].

Using the critical damping in Eq.(6.36), both poles are set to the same place (see Eq.(6.33)). In other words, in order to meet the two conditions from Eq.(6.34) and Eq.(6.36), both poles must have the same real negative value, but no imaginary part since the imaginary part of the roots induce overshoot of the response.

In order to choose the location of both poles, several different sets of desired poles have to be designed and the corresponding control gain parameters determined [29]. After examining the different response curves, the most appropriate gain parameter $k_p$ has to be chosen. It is commonly used (see [29] and [20]) to evaluate the best overall system performance by means of the settling time. The criterion for the settling time depends
on the particular situation [29]. The settling time, in this thesis, is defined in Section 6.1.1.

As explained in the beginning of this section, the gained information for the control design of one independent coordinate can be adopted to the other \((m-1)\) equations of the linear decoupled system in Eq.(6.24). Consequently, \(m\) independent controllers using Eq.(6.29) have to be applied to the transformed system. In matrix formulation it can be expressed as:

\[
\begin{bmatrix}
\ddot{z}_1 \\
\vdots \\
\ddot{z}_m
\end{bmatrix} =
\begin{bmatrix}
\ddot{z}_{\text{ref},1} \\
\vdots \\
\ddot{z}_{\text{ref},m}
\end{bmatrix} +
\begin{bmatrix}
k_{d,1} & \cdots & 0 \\
0 & \cdots & k_{d,m}
\end{bmatrix}
\dot{e} +
\begin{bmatrix}
k_{p,1} & 0 \\
0 & \cdots & k_{p,m}
\end{bmatrix}
e,
\] (6.37)

where the tracking error vector \(e\) and its time derivative \(\dot{e}\) are defined as:

\[
e = \begin{bmatrix}
z_{\text{ref},1} - \dot{z}_1 \\
\vdots \\
z_{\text{ref},m} - \dot{z}_m
\end{bmatrix}
\quad \text{and} \quad
\dot{e} = \begin{bmatrix}
\dot{z}_{\text{ref},1} - \ddot{z}_1 \\
\vdots \\
\dot{z}_{\text{ref},m} - \ddot{z}_m
\end{bmatrix}
\] (6.38)

The general structure of the nonlinear MIMO system using feedback linearization in order to compute the control torque can be seen in Figure 6.8. The linear control approach requires the availability of the full state space vector \(x = \{z, \dot{z}\}^T\) as well as the reference state space vector \(x_{\text{ref}} = \{z_{\text{ref}}, \dot{z}_{\text{ref}}\}^T\) and the reference acceleration vector \(\ddot{z}_{\text{ref}}\).

![Figure 6.8: Linear control approach of the linearized nonlinear MIMO system using feedback linearization.](image)

In the following section, three different pole placement sets will be investigated for the four-bar linkage model and the CTC control will be designed by means of the most appropriate settling time.

Consequently, as explained before, the chosen control design for the one DOF four-bar linkage model can be adapted to the 14 controllers of the human body model.
6.2.4 CTC Control applied to the FDA of the Four-Bar Linkage Model

The forward dynamic analysis is carried out using the same integration method as in Section 6.1.1 (variable-step solver ode23 with a simulation time step of 1ms, forces and torques from IDA are linearly interpolated) to provide consistent results.

In order to choose the poles, a wrong initial vector $x_{\text{false}}(t = 0) = \{0, 1\}^T$ is used, as in Section 6.1.1. Three different sets of the two poles $s_1, s_2$ at the locations $(-1), (-10), (-100)$ are investigated. The response is examined and the corresponding settling time is determined. Table 6.2 shows the three different sets of the control parameters and Figure 6.9 shows the tracking error of the FDA of the four-bar linkage model using these three different CTC controls.

![Figure 6.9: Tracking error of $\alpha$ for forwards dynamic simulation of four-bar linkage model with a wrong initial state vector using different CTC control approaches.](image)

First of all, it can be seen that all three controllers are able to correct the wrong initial state vector. Furthermore, due to the use of the critical damp response, the motion is correcting the error without any overshoot. The magnitude and the settling time are dependent on the chosen poles and therefore, dependent on the control design. In Table 6.2 the three control designs are compared regarding to their NRMSE, settling times and simulation times. It can be pointed out that the CTC3 control, which has both poles placed at $(-100)$, shows all around the best performance. The NRMSE is $1.8116\%$ and the response just needs $0.06s$ until it follows again the reference data with a difference of $1\%$. 
Table 6.2: Comparison of NRMSE, settling time and simulation time dependent on adjusted control parameter $k_p$ of CTC controller (using wrong initial values).

<table>
<thead>
<tr>
<th>Control POles</th>
<th>Control parameters</th>
<th>NRMSE($\alpha$)</th>
<th>Settling time</th>
<th>Simulation time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$k_p \left[ \frac{1}{s^2} \right]$</td>
<td>$k_d \left[ \frac{1}{s} \right]$</td>
<td>[%]</td>
<td>$t_{1%}$ [s]</td>
</tr>
<tr>
<td>CTC1</td>
<td>-1</td>
<td>1</td>
<td>2</td>
<td>27,0602</td>
</tr>
<tr>
<td>CTC2</td>
<td>-10</td>
<td>100</td>
<td>20</td>
<td>5,8872</td>
</tr>
<tr>
<td>CTC3</td>
<td>-100</td>
<td>10.000</td>
<td>200</td>
<td>1,8116</td>
</tr>
</tbody>
</table>

As explained in Section 6.2.3, the selection of poles is a compromise between rapidity of the response of the error vector and the sensitivity to disturbances and measurements noises. In this thesis, the poles of the CTC control of the four-bar linkage model are placed both at $(-100)$. Therefore, the CTC3 control of Table 6.2 is used with the control parameters of $k_p = 10.000 \frac{1}{s^2}$ and $k_d = 200 \frac{1}{s}$.

Figure 6.10a shows the error trajectory of the FDA using the CTC3 controller. Note, that the initial state vector now is given correctly. The four-bar linkage model follows the prescribed motion during all the analysis and the controlled angle $\alpha$ is concurrent with the reference motion. The CTC controller achieves a high accuracy of the motion with a tracking error in the order of $10^{-8}$ degrees for the four-bar linkage model. The control torque $Q_{CTC}$, illustrated in Figure 6.11, is in the order of $10^{-4}$Nm. It can be noticed that it is needed consistently during all the simulation and it is one order of magnitude lower than the PD control torque $Q_{PD}$ (see Figure 6.5). The angular velocity and angular acceleration follow the desired values with errors in the order of $10^{-6} \frac{deg}{s}$ and $10^{-3} \frac{deg}{s^2}$ respectively, as it can be seen in Figure 6.10b. In Chapter 7 both control strategies, PD and CTC control will be compared for the four-bar linkage model.

6.2.5 CTC Control applied to the FDA of the 2D-Human Body Model

As explained in Section 6.2.3, the control design of the four-bar linkage model can be adapted to the two-dimensional human body model. It is shown that the nonlinear MIMO system with 14 DOF can be transformed into a linear system with 14 second order decoupled equations. Hence, the control design of the independent coordinate $\alpha$ of the four-bar linkage model can be adapted to all the 14 independent coordinates of the human body model. Applying the same strategy, the diagonal matrices $K_p$ and $K_d$
in Eq.(6.37) contain the values $k_{p,1} = \ldots = k_{p,14} = 10.000 \frac{1}{s^2}$ and $k_{d,1} = \ldots = k_{d,14} = 200 \frac{1}{s}$.

The forward dynamic simulation of the human body model is carried out using as well the variable-step solver ode23 with a simulation time step of 1ms, a relative tolerance of $10^{-3}$ and an absolute tolerance of $10^{-6}$. The forces and torques provided by the IDA are added by linear interpolation, as explained before.

Using the adapted CTC control strategy the forward dynamic simulation is able to reproduce the reference human motion with a very high accuracy during the whole human gait cycle. Figure 6.12 shows the tracking error of the hip joint position, the absolute trunk angle and the relative angles of the lower limb joints of both legs, which are in the order of $10^{-5}$ until $10^{-3}$ degrees and $10^{-6}$ m, respectively.

Figure 6.10: FDA results of four-bar linkage model using CTC control.

Figure 6.11: CTC control torque of four-bar linkage model.
The corresponding control forces and torques of the CTC control with the order of $10^{-2}$ until $10^{-1}$Nm and $10^0$N, respectively, are shown in Figure 6.13.

In order to analyse the system performance of the forward dynamic analysis using the CTC control, the stability and robustness of the control strategy regarding model uncertainties and input data disturbances are tested the following chapter.
Figure 6.13: CTC control forces and torques applied to the human body model.
7 System Performance and Control Robustness

In Chapter 6 two control strategies have been introduced (PD control and CTC control). Both control strategies have been applied to the four-bar linkage model and their results have been plotted and analysed. In this chapter, these two control strategies are compared with respect to the obtained NRMSE of the controlled variable and its derivatives, control torque and simulation time. Moreover, different fault cases are introduced and the system’s performance is investigated for both control approaches.

The forward dynamic analysis of the two-dimensional human body model requires, as explained in Chapter 5, next to the forces and torques generated through the IDA, exact initial values (independent coordinates and velocities) as well as the body segment parameters. As explained in Section 6.2.1, the CTC control approach needs confidence in the model since the robustness cannot be guaranteed in the presence of parameter uncertainties or unmodeled dynamics. Therefore, in this chapter, the robustness of the CTC control is analysed with respect to possible uncertainties of the required data using the same variables as before.

7.1 Comparison of PD and CTC Control for Four-Bar Linkage Model

Two controllers have been designed for the four-bar linkage model in Chapter 6 in order to reproduce the reference angle motion \( \alpha(t) \). The forward dynamic simulation performances of both controllers are compared in Table 7.1 with respect to the NRMSE of the independent coordinate \( \alpha \), the angular velocity \( \dot{\alpha} \) and the angular acceleration \( \ddot{\alpha} \), the control torque range and the simulation time. The range of the control torque is defined as the difference between maximum and minimum torque applied to the model:

\[
\Delta Q_C = Q_{C,\text{max}} - Q_{C,\text{min}}
\]  

(7.1)

It can be seen that both controllers perform very accurately. Especially the CTC control approach features very low NRMSE of the independent coordinate \( \alpha \) and its derivatives. The applied CTC control torque in order to guide the system’s motion is lower than the PD control torque and the CTC controller is faster than the PD controller with respect to the simulation time (see Table 7.1).
Table 7.1: Comparison of both control strategies PD and CTC in the forward dynamic simulation of the four-bar linkage model.

<table>
<thead>
<tr>
<th>Control</th>
<th>NRMSE(α) [%]</th>
<th>NRMSE(α̇) [%]</th>
<th>NRMSE(α̈) [%]</th>
<th>Control torque range ∆Qc [Nm]</th>
<th>Simulation time [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD</td>
<td>2.95 × 10⁻⁷</td>
<td>4.17 × 10⁻⁶</td>
<td>1.36 × 10⁻⁴</td>
<td>2.37 × 10⁻³</td>
<td>238.24</td>
</tr>
<tr>
<td>CTC</td>
<td>8.74 × 10⁻⁹</td>
<td>1.15 × 10⁻⁶</td>
<td>7.11 × 10⁻⁵</td>
<td>7.27 × 10⁻⁴</td>
<td>212.79</td>
</tr>
</tbody>
</table>

In order to compare the stabilization performances of both controllers, two different fault cases are introduced to the model. In the first fault case, the masses \( m_1 \) and \( m_3 \) of the four-bar linkage model are increased by 20% and the mass \( m_2 \) of the second segment is decreased by 20%. This fault case is introduced to the model at the time of 1s. If these mass changes occur in the forward dynamic simulation without any applied control approach, the system’s motion is not able to follow the reference data, as it can be seen in Figure 7.1. Figure 7.2 shows the corresponding tracking error of the forward dynamic simulation applying no control, the PD control or the CTC control.

Figure 7.1: Motion of forward dynamic simulation of four-bar linkage model introducing a mass fault to the system at 1s (without control).

It can be seen that the PD and CTC control strategies are able to stabilize the forward dynamic simulation though they are not aware of the mass changes in the plant of the system. Comparing the order of the tracking errors, it can be pointed out that the CTC controller features, with a tracking error of the order \( 10^{-3} \) degrees, a more accurate performance than the PD controller.

The second fault case analyses the influence of the torque which is provided by the IDA. In this case, the torque is reduced to 50% after 2s of the simulation. The change occurs again in the nonlinear plant of the system and neither the PD controller nor the CTC controller are aware of this change. First of all, it can be seen in Figure 7.3 that the
Figure 7.2: Tracking error of $\alpha$ for the forward dynamic simulation of four-bar linkage model with mass error introduced to the system at 1s.

forward dynamic simulation shows an unstable behaviour without applying a control approach. Figure 7.4 shows the tracking error and its derivatives during the FDA for the considered control methods.

Introducing the torque fault case, both controllers are capable of stabilising the system’s motion very fast and accurately. The CTC control approach features again a better performance than the PD controller since its tracking error is of one magnitude order lower. Furthermore, it can be seen that the CTC control performs without overshoot and the settling time in order to correct the introduced error is much lower using the CTC control than the PD control (see Figure 7.4).

It can be said that the CTC control approach shows overall the best performance in order to control the forward dynamic simulation of multibody systems used in this thesis. Even though the PD performs still very well in the case of one DOF, its performance fails in the case of MIMO systems, as shown in Section 6.1.2.
7.2 Robustness of CTC Control

Section 4.2.3 describes the acquisition of the body segment parameters for the two-dimensional human body model. These values are estimated and can contain uncertainties. However, the CTC control approach requires confidence in the model since parameter uncertainties can degrade the performance of the controller [40]. Because this control approach seems not to feature robust characteristics, special attention is turned to the analysis of the robustness of the CTC control approach with respect to uncertainties of model parameters and disturbances of the provided forces and torques from the IDA.

In [32] it is pointed out that the IDA of the human body model is especially affected by changes of the mass and length parameters of the human body model. Hence, the CTC control approach is tested by changing these values of the nonlinear system. Note that the CTC control approach and therefore, the implementation of the feedback linearization is not aware of any uncertainties applied to the model parameters and does not consider any disturbances acting on the plant.

The robustness analysis is primarily made for the four-bar linkage model. The FDA is carried out using the variable-step solver ode23 and the torque provided by the IDA is added by linear interpolation at a simulation time step of 10ms. Uncertainties of system input data increase the simulation time considerably (approximately 25 hours) since the ode23 solver chooses very small variable time steps to keep errors under a certain tolerance. The robustness analysis focus mainly on changes in NRMSE values and the control torque range at system degradations. Therefore, in order to reduce the simulation time, the forward dynamic simulation is carried out with a relative tolerance of
\(10^{-3}\) and an absolute tolerance of \(10^{-6}\). Table 7.2 shows the simulation results of the system without introduced fault with the used tolerances in comparison to the tolerances of \(10^{-9}\) used before. Reducing the tolerance reduces the simulation time by 90\% but yields higher NRMSE values of the independent coordinate and its derivatives since the integration solver is less accurate. In conclusion, the control torque range \(\Delta Q_{\text{CTC}}\) increases because the tracking error is tolerated to be higher. It has to be considered, that the NRMSE values are still very low and the control torque range of 35Nm is still just 6\% of the torque range which is provided by the IDA (see Figure 4.3). Therefore, the new tolerances of \(10^{-3}\) and \(10^{-6}\) are used for the different fault cases which are introduced to the system.

The masses and lengths of the nonlinear system with one DOF are randomly changed in the magnitude of 10\%, 25\% and 50\%. A further fault case investigates the influence of the applied torque from the IDA to the robustness of the CTC control. The IDA torque, which is added by linear interpolation at each variable time step of the ode23 solver, is disturbed as well in the magnitude of 10\%, 25\% and 50\%. Table 7.2 shows the NRMSE of the independent coordinate \(\alpha\) and its derivatives, the control torque range \(\Delta Q_{\text{CTC}}\), which is defined in Eq.(7.1) and the simulation time for all examined fault cases with different applied error magnitudes in \%.

It can be seen that the CTC control approach still performs well at all three fault cases with applied errors until 50\%. Logically the control torque range increases with higher errors since it has to correct the disturbances. Increasing mass uncertainties yields especially higher position errors, whereas length uncertainties cause as well higher velocity errors. For this fault case very high control torques are required. The IDA torque fault case implicates, in addition, high angular acceleration errors. In this case, the disturbance occurs randomly in every time step. Hence, the simulation time increases excessively.

In the case of the human body model, the analysis of the robustness features more difficulties. The forward dynamic simulation cannot be carried that simply as before with the four-bar linkage model. Changes in the masses of the human body segments of 1\% already show immense differences from the captured human motion and changing the masses more than 2,5\%, the simulation drifts away. Figure 7.5 shows the independent coordinates of the lower limb segments in the forward dynamic simulation of the human body model with a mass fault of 2,5\% introduced to the system from the beginning compared to the reference motion. It can be seen that even little changes in all segment masses lead to wrong angles at some specific time instances of the simulation. The system is even more sensitive concerning uncertainties of the model’s lengths. Here, the only possible changes are 0,01\%. With changes higher than this value, the simulation cannot be carried out. Disturbing the IDA forces and torques, leads to similar results. The simulation can only perform with disturbances lower than 0,01\%. This uncertainty can be avoided by not using the forces and torques from the IDA as an input for the FDA.
Table 7.2: Robustness analysis of CTC control approach in the forward dynamic simulation of the four-bar linkage model. Fault cases are introduced from the beginning (t=0s).

<table>
<thead>
<tr>
<th>System state</th>
<th>Error [%]</th>
<th>NRMSE((\alpha)) [%]</th>
<th>NRMSE((\dot{\alpha})) [%]</th>
<th>NRMSE((\ddot{\alpha})) [%]</th>
<th>Control torque range (\Delta Q_{CTC}) [Nm]</th>
<th>Sim. time [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>No fault (\epsilon_{abs}=10^{-9}) (\epsilon_{rel}=10^{-9})</td>
<td>8.82 (\cdot) 10(^{-9})</td>
<td>1.15 (\cdot) 10(^{-6})</td>
<td>7.10 (\cdot) 10(^{-5})</td>
<td>7.26 (\cdot) 10(^{-4})</td>
<td>248.60</td>
<td></td>
</tr>
<tr>
<td>No fault (\epsilon_{abs}=10^{-6}) (\epsilon_{rel}=10^{-3})</td>
<td>0.0004</td>
<td>0.0433</td>
<td>24166</td>
<td>35.55</td>
<td>26.29</td>
<td></td>
</tr>
<tr>
<td>Mass fault</td>
<td>10</td>
<td>0.0020</td>
<td>0.0406</td>
<td>24396</td>
<td>37.79</td>
<td>24.65</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>0.0049</td>
<td>0.0389</td>
<td>25921</td>
<td>73.99</td>
<td>28.52</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>0.0096</td>
<td>0.0364</td>
<td>28640</td>
<td>146.99</td>
<td>27.64</td>
</tr>
<tr>
<td>Length fault</td>
<td>10</td>
<td>0.0102</td>
<td>0.0568</td>
<td>21618</td>
<td>156.36</td>
<td>28.49</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>0.0287</td>
<td>0.0999</td>
<td>20466</td>
<td>380.47</td>
<td>27.47</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>0.0737</td>
<td>0.2556</td>
<td>22270</td>
<td>943.85</td>
<td>26.11</td>
</tr>
<tr>
<td>Torque fault</td>
<td>10</td>
<td>0.0008</td>
<td>0.0432</td>
<td>31833</td>
<td>77.01</td>
<td>34.26</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>0.0013</td>
<td>0.0496</td>
<td>61843</td>
<td>139.09</td>
<td>66.61</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>0.0018</td>
<td>0.0643</td>
<td>117417</td>
<td>211.98</td>
<td>108.0</td>
</tr>
</tbody>
</table>

since the CTC control approach can also perform without input forces and torques. The applied control forces and control torques would be consequently higher. These results proved the less robust performance of the CTC control in the case of MIMO multibody system approaches.

Additionally, a wrong initial state vector is investigated. Setting all independent coordinates and their velocities to zero, the CTC control works very well and is able to reproduce the captured reference motion with an error of 1% after a settling time of 0.09s.

Finally, the control performance has been tested with respect to gained information of the degraded system. Degrading the system by any fault case explained before and, simultaneously, forwarding this information to the CTC controller, the performance is as perfect as seen before in Section 6.2.5.

In conclusion it can be summarized, that though the CTC control works more accurate in comparison to the PD control, it is highly dependent on the knowledge of exact system
input data. It is shown that the CTC control performance is very robust for one DOF systems but less robust for MIMO systems. It can be assumed that changes in a MIMO system plant, without forwarding these changes to the controller, lead to significant impairments of the feedback linearization. The linear system in Eq.(6.24) is not valid any more and thus, the control approach in Eq.(6.29) becomes ineffective. Hence, the CTC control approach is not sufficiently robust against model parameter uncertainties and input data disturbances and needs a high confidence in the model if the DOF is higher than one.

A robustness extension for the CTC approach is described in [40] and an adaptive control using feedback linearization is introduced in order to improve the control performance.

**Figure 7.5:** Independent coordinates in the forward dynamic simulation of the human body model with a mass fault of 2.5% introduced to the system from the beginning compared to the reference motion.
8 Environmental and Social Impacts and Economic Consideration

8.1 Environmental Impact

This thesis contains a marginal environmental impact since the project was carried out consistently on the basis of simulation. Besides the use of electricity, pen and paper, a computer has been used to conduct the simulations and to write the thesis. The project team is aware of treating the waste of electrical and electronic equipment after their end of life cycle individually in agreement with the Directive 2012/19/EU of the European Parliament and of the Council of 04 July 2012 on waste electrical and electronic equipment (WEEE).

8.2 Social Impact

Due to the fact that this project is based completely on simulative investigation, no significant social impact can be mentioned. It has to be considered that the project itself is referred to current research of the biomechanics community regarding to human motion prediction which has an immense social and medical impact to improve the quality of life of incomplete spinal-cord-injured subjects.

8.3 Economic Consideration

The economic cost of this project consists of fixed and variable costs. Since all fixed costs (investments) are not only beneficial for this project but are instead available for following projects during their life expectancy, the costs for this project are recalculated and hence, given per hour. Therefore, all cost factors are treated variable.

The project costs can be divided in preparation cost to enable the project and execution costs which appear during the project work (see Table 8.1). Table 8.2 contains the calculation of the final project cost. Fixed cost factors are recalculated to variable cost expenses given in euros per hour. Fixed costs and their life expectancy are estimated. The Matlab license (500€) is available for one year (52 weeks, 7 days a week, 24 hours
a day). The costs for the project, which lasts six months, are therefore the half of the license cost. The computer (759€) and the office equipment (projected cost of 300€) have life expectancies of 4 and 30 years considering an utilization of 52 weeks a year, 5 days a week and 10 hours per day. The consumable material is just considered for this project and lasts therefore the same amount of hours as the student is working on the project. The working hours of the supervisors and the student are estimated as well as the administrative hours. Computer and light uses 40W with estimated electricity costs of 0,2 \( \frac{€}{kWh} \). The computer is running during all the project working time (1.200 hours) and the light is used for one third of the project time (400 hours). It has to be mentioned that the final project cost of 18.436€ does not consider the cost of the occupied working place by the student. This cost is unknown.

<table>
<thead>
<tr>
<th>Table 8.1: Project costs.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Preparation of project condition</strong></td>
</tr>
<tr>
<td>• Purchase of hardware (computer)</td>
</tr>
<tr>
<td>• Purchase of Software (Matlab)</td>
</tr>
<tr>
<td>• Office equipment (chair, desk)</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 8.2: Calculation of the final project cost.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cost factor</strong></td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>Computer</td>
</tr>
<tr>
<td>Matlab</td>
</tr>
<tr>
<td>Office equipment</td>
</tr>
<tr>
<td>Supervisors</td>
</tr>
<tr>
<td>Student</td>
</tr>
<tr>
<td>Administrative cost</td>
</tr>
<tr>
<td>Energy</td>
</tr>
<tr>
<td>Consumable material</td>
</tr>
</tbody>
</table>

18.436,28
9 Conclusions

In this thesis, two different computational models are presented: a four-bar linkage model with one degree of freedom; and a two-dimensional human body model that consists of 12 segments with 14 degrees of freedom.

A forward dynamic simulation of both models is carried out by means of multibody dynamics techniques in order to follow a given reference motion. While the reference motion of the four-bar linkage model is defined in this thesis, the reference human motion of a healthy subject was captured prior to this thesis in a biomechanics laboratory.

Several methods in order to solve the equations of motion are described and compared. The matrix-R formulation reduces the number of equations to be integrated importantly and overcomes the stability problem, which occurs using the other solving methods, without loosing considerable accuracy in the results. Therefore, the equations of motion are solved in this thesis, using the matrix-R formulation.

Different numerical integration methods are introduced in order to carry out the forward dynamic simulation of multibody systems. Comparing their performances, the variable-step solver ode23 from Matlab improves the numerical integration performance with respect to the fixed-step solvers by 19%. Hence, the forward dynamic simulations, in this thesis, are carried out using the variable-step integration solver.

The joint forces and torques, which are applied to the two multibody systems in order to drive their motion, are provided through a previous inverse dynamic analysis. Furthermore, the forward dynamic analysis requires body segment parameters of the model and initial values of the independent coordinates.

Due to numerical errors during the integration and the unstable character of human gait, the results of the forward dynamic simulation of both models differ from the reference motion if no control method is applied. Therefore, two different control approaches are introduced and designed: a Proportional Derivative (PD) control and a Computed Torque Control (CTC) using feedback linearization.

Applying these two control methods to the forward dynamic simulation of the four-bar linkage model, very accurate dynamic results can be achieved, which are following the reference motion as desired. Comparing both control approaches in several fault cases shows that the CTC control performs without overshoot, faster and more accurately with a factor of 10 than the PD control.
Even though the PD control approach performs still very well in the case of the four-bar linkage model, it has its limitations in the case of the human body model, where the performance of the forward dynamic simulation fails. The CTC control instead is able to guide the human motion during the whole simulation following the reference data with a very high accuracy and fast response.

In order to test the CTC control, the robustness of this control method is analysed introducing several fault cases to the models. While the CTC control performance is very robust for the four-bar linkage model with one DOF, it performs less robust for the human body model with 14 DOF.

In summary, it can be pointed out that the CTC control with feedback linearization is a powerful predictive control approach for nonlinear multibody systems by means of path generation with a high potential for human motion control.

9.1 Future Work

In order to improve the thesis outcome in the future, some extensions and future work are proposed in the following:

- More precisely tuning of the control parameters.
- Improving the numerical integration process (stiffness analysis, implicit solvers, multistep solvers).
- Improving the robustness of the CTC control approach.
- CTC control applied as adaptive control.
- Adding a contact model in order to emulate the actual foot-ground interaction.
- Using a three-dimensional human body model for more realistic simulations.
- Using an Extended Kalman Filter (EKF) if the system is not fully state observable.
- Using more data acquisitions of more subjects performing different motion tasks.
Bibliography


[38] Ren, L., Jones, R. K., and Howard, D. Whole body inverse dynamics over a complete gait cycle based only on measured kinematics. *Journal of Biomechanics* 41, 12 (2008), 2750–2759.


[48] Xiang, Y., Arora, J. S., and Abdel-Malek, K. Hybrid predictive dynamics: a new
