Human balance and posture control during quiet stance

Citation for published version (APA):

Document status and date:
Published: 01/01/2014

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 16. Oct. 2020
Human balance and posture control
during quiet stance

F.N. Hoogeboom

D&C 2014.35

Traineeship report

Coach(es):    Dr.ir. A. Shukla
              Dr.ir. R.H.B. Fey

Supervisor:   Prof.dr. H. Nijmeijer

Eindhoven University of Technology
Department of Mechanical Engineering
Dynamics & Control

Eindhoven, July, 2014
Contents

Abstract v
Samenvatting vii

1 Introduction i
  1.1 Motivation ........................................... 1
  1.2 Literature review .................................. 2
  1.3 Problem definition ................................ 2
  1.4 Outline ............................................. 3

2 Modeling of human balance dynamics 5
  2.1 Rigid surface model ............................... 5
  2.2 Compliant surface model ........................... 6
  2.3 Central nervous system ............................. 7
  2.4 Time domain simulations ........................... 10

3 Delay systems 11
  3.1 Delay differential equations ....................... 11
  3.2 Periodic solutions of DDEs ......................... 12
  3.3 Shooting for DDEs ................................ 12
  3.4 Collocation for DDEs ............................... 13

4 Collocation for piecewise-smooth delay systems 19
  4.1 PWS delay systems .................................. 19
  4.2 Test models ......................................... 21
  4.3 Rigid surface model ................................ 22
  4.4 Compliant surface model ......................... 24

5 Continuation and bifurcation analysis 27
  5.1 Continuation of periodic orbits ................... 27
  5.2 Numerical problems ................................ 27
  5.3 Numerical results .................................. 30

6 Conclusions and recommendations 31
  6.1 Conclusions ......................................... 31
  6.2 Recommendations ................................. 32

References 33

iii
Contents

A Time domain simulations
   A.1 Tilt angle .................................................. 36
   A.2 Ankle angle .................................................. 44
   A.3 Toe height .................................................. 48

B Compliant surface model ........................................... 53

C Collocation for PWS delay systems
   C.1 Collocation for delay systems ................................. 57
   C.2 Collocation for PWS delay systems ........................... 58
Abstract

Human balance and postural control is a crucial factor in allowing us to perform normal tasks during life. With Parkinson's disease, the ability to maintain uprightness reduces. Therefore, there has been an interest in the engineering community to use modeling for understanding the mechanism for loss of balance. These models are validated with results from clinical balance tests. However, these tests for following the longitudinal course of Parkinson's are performed on different types of underground with associated stiffness, whereas the mathematical models used so far considered the underground to be infinitely stiff. Therefore, the focus of this study lies on observing and understanding the differences in dynamical behavior between a rigid surface model and a model that includes finite stiffness of the underground.

The existing rigid surface model is an inverted pendulum model, which rotates around the ankle joint. The compliant surface model is an extension of this model by adding springs and dampers under the ankle.

The torque to correct posture is exerted by the ankle muscles and changes instantaneously when state-depended thresholds are passed. This introduces a discontinuity in the equations of motion. Also, it takes some time before the central nervous system (CNS) has processed the signals from various body sensors and can react if needed. Due to both facts the equations of motion are described by a discontinuous delay differential equation (DDE).

Simulations are performed with both models to compare their dynamical behavior. The results show that different types of periodic solutions exist. This means that the number of boundary crossings within a periodic solution can change when a parameter value is varied.

For the dynamic analysis, it is important to find periodic solutions. Since the model is described by a piecewise-smooth (PWS) delay system, no standard numerical tools are available. Therefore, a short background study on numerical tools for delay systems is performed. Based on this study, the collocation method is chosen to be altered to make it work for PWS delay systems.

The results computed with this method show for the given set of parameters almost no differences in the periodic solutions obtained with the two models. So for the given the parameters, it is not necessary to use the compliant surface model since the rigid surface model gives the same results.

To follow branches of periodic solutions, an arclength continuation method is adopted. Convergence problems with this method occur when a part of the branch is followed. Some problems could be solved in this study, but others require further research to make the continuation method usable for PWS delay systems.
Samenvatting

Menselijk evenwicht en beheersing van de lichaamshouding is een cruciale factor in ons leven bij het uitvoeren van allerlei normale taken. Maar met Parkinson verminderd de mogelijkheid om stabiel rechtop te blijven staan. Daardoor is er de laatste jaren aandacht besteed vanuit de academische wereld om wiskundige modellen te ontwikkelen die kunnen verklaren waardoor de verminderde balans ontstaat. Deze modellen worden gevalideerd met meetdata van klinische testen. Alleen de testen welke uitgevoerd worden om het verloop van Parkinson te volgen worden uitgevoerd op verschillende ondergronden. Waar de tot nu toe ontwikkelde modellen altijd een vaste ondergrond beschouwde. Daardoor ligt de nadruk van deze studie op de verschillen tussen een model met vaste ondergrond en een model waar de ondergrond een mate van flexibiliteit heeft.

Het bestaande vaste ondergrond model is een omgekeerd slinger model welke scharniert om het enkelgewricht. Het model met verende ondergrond is een uitbreiding van dit model door veren en dampers te modeleren onder de enkel.

Het koppel dat uitgeoefend wordt door de enkel spieren om het lichaam te corrigeren verandert instantaan wanneer vooropgestelde drempels in de toestand ruimte worden overschreden. Dit zorgt ervoor dat er een discontinuïteit ontstaat in de bewegingsvergelijkingen. Ook het verwerken van de signalen van verschillende lichaamsensoren door het centrale zenuwstelsel kost tijd en daardoor zijn de bewegingsvergelijkingen in de vorm van een discontinue differentiaalvergelijking met vertraging.

Simulaties zijn uitgevoerd met beide modellen om het dynamisch gedrag te vergelijken. De resultaten hiervan laten zien er meerdere mogelijkheden bestaan van periodieke oplossingen. Hiermee wordt bedoeld dat het aantal grensoverschrijdingen tijdens een periodieke oplossingen kan veranderen wanneer een parameter wordt varieerd.

Voor een dynamische analyse is het belangrijk om periodieke oplossingen te vinden. Omdat het model is beschreven door een stuksgewijs glad vertragingssysteem zijn er geen pasklare numerieke hulpmiddelen voorhanden. Daarom is een korte theoretische studie uitgevoerd op numerieke hulpmiddelen voor het vinden van periodieke oplossingen voor vertragingssystemen. Daarna is gekozen om de collocatie methode aan te passen zodat deze ook gebruikt kan worden voor stuksgewijs gladde vertragingssystemen. De resultaten berekend met deze methode laten zien dat er bijna geen verschil zit in periodieke oplossingen tussen beide systemen.

Om takken van periodieke oplossingen te volgen is een booglengte continuering methode toegepast. Probleem met deze methode is dat convergentie opdoemen wanneer een gedeelte van een tak is gevolgd. Sommige van deze problemen konden worden opgelost in deze studie waar andere meer onderzoek nodig hebben voordat een succesvolle bifurcatie analyse gedaan kan worden.
1

Introduction

1.1 Motivation

Human postural control and balance is the process in which an upright stance is maintained under a variety of internal and external influences. It includes the CNS, the musculoskeletal system and sensorimotor processes. The CNS controller integrates real time sensory data coming from vision, the vestibular system, and the somatosensory system and corrects the posture if needed by activation of the ankle and/or hip muscles. A schematic overview of this situation is depicted in Figure 1.1.

There exists, however, a continuous trade-off between stability and required control effort. As a result, there always exists a small back and forth sway in the sagittal plane. This is the vertical plane which passes from front to back separating the body into right and left halves. Human balance can be compromised by various reasons including disease. In this case of Parkinson’s disease, this is most commonly specified by the Unified Parkinson’s Disease Rating Scale (UPDRS) [15]. This is a rating scale used to follow the progressing of Parkinson’s disease. Following this scale over time gives an impression of the progression of the disease, and to make life of the patients more tolerable, treatment in the way of exercises might help. Therefore, there has been an effort in the engineering community to use modeling for understanding the fundamental mechanism for the loss of balance and to investigate which treatment might be beneficial.

However, all the models so far consider an inverted pendulum on a rigid surface, whereas the clinical tests for the UPDRS are performed on different types of underground such as foam. Also, the rigid surface models do not seem to fit experimental data. For this reason, this research is conducted
to develop a model that can produce realistic characteristics of postural balance and to investigate differences in dynamics between this model and the models that are used so far.

The remaining of this chapter start with a literature review, in which a brief overview is given on the ideas and knowledge that have been established on the topics that are considered in this study. Subsequently, the objectives for this research are described. This chapter ends with an outline of the report.

1.2 Literature review

Human balance and posture control has been investigated in literature. A thorough explanation of human balance and posture control during standing and walking is presented in [20]. It explains the differences between center of mass (COM) and center of pressure (COP). Furthermore, it studies different stabilizing strategies such as ‘ankle-strategy’, ‘hip strategy’, or even a combined strategy.

Stability and bifurcation of postural control with and without wobble board are investigated in [5]. In this study, a model with nonlinear muscle stiffness and geometric nonlinearities is used.

Next to human modeling, the CNS control system is also a topic of research. In [12], two different control strategies, continuous or intermittent, are investigated. The results of this research show that the CNS controller observe continuously and acts intermittently. This result is used in [1] where for human standing event-driven intermittent control is proposed, in which control is altered when state-dependent thresholds are passed. The analysis method is, however, not preferred since the model is linearized and the time delay is approximated by a Taylor expansion. The same model and approach is followed in [13] apart from that internal disturbances are modelled. A different approach is investigated in [11]: this study introduces model predictive control (MPC) as an optimal control model of posture and movement.

Besides literature on human modelling and control we also need to investigate the dynamical behavior of the system. For this periodic solution solvers are sought. A vast amount of literature is available on the popular shooting method. This method is, however, restricted to smooth dynamical systems. In [10], an approach is discussed, which is suitable for non-smooth systems. Apart from the shooting method, the more robust collocation method is also widely available, see for example [19]. In [17], this method is extended such that it can be used for non-smooth ordinary differential equations (ODEs). The shooting method for delay systems is also explained by different authors, see for example [6, 8, 9]. In [7], collocation for delay systems is explained. Again this method is restricted to smooth dynamical systems. An approach to overcome this problem is established in [3] by smoothing the non-smooth dynamics. The same author has also proposed a method for computing the periodic solutions for PWS delay systems [2].

1.3 Problem definition

The basic objective of this work is to develop two models of a human upright posture during quiet stance and perform a parametric stability analysis on both models to investigate differences in dynamical behavior. This objective can be divided in the following sub-objectives

1. Model a human being in quiet stance on a rigid or on a compliant surface.
2. Incorporate the CNS as an intermittent controller.
3. Implement a strategy to calculate periodic solutions for a discontinuous dynamic system with time-delay because standard ordinary solvers cannot be applied for this type of equations.
4. Compute branches of periodic solutions such that the differences between both models can be examined and the influence of parameters on the stability can be analyzed.

1.4 Outline

The outline of this report is as follows. The next chapter starts with the derivation of the equation of motion, which captures the system dynamics for both models that are used in study. Furthermore, the CNS controller is discussed together with model parameters. The chapter ends with simulation results. In Chapter 3, a short introduction on delay systems is presented together with a definition of periodic solutions. Also, two numerical tools are presented used for obtaining these periodic solutions: the popular shooting method and the more robust collocation method. Chapter 4 presents an altered version of the collocation method, which makes it suitable for PWS delay systems. It includes a description of the method together with test results to validate the Matlab program before it is used to analyze the models that are used in this study. In Chapter 5, an arclength continuation method is adopted that can compute branches of periodic solutions for PWS delay systems. Chapter 6 ends this study with the conclusions. Additionally, recommendations are provided for further research.
Modeling of human balance dynamics

In this study two different models are used: a human body model standing on a rigid and on a compliant surface. In order to study the dynamic behavior of both models, we need to derive the equations of motion. First, the simplest model is discussed, namely the rigid surface model. The equations of motion for this model are rather straightforward. Subsequently, the model is extended with an flexible foundation, which results in two extra degrees of freedom. Lagrange’s equations are used to derive the equations of motion. Next, the CNS controller is discussed, which is the same for both models. Finally, we present simulation results to explore the differences in system dynamics between both models.

2.1 Rigid surface model

The rigid surface human body model used in this study is obtained from [1, 13] and is schematically depicted in Figure 2.1. This model is a simplified representation of a human body standing quietly on a rigid foundation.

![Human body model](image)

Figure 2.1: Human body standing quietly on a rigid foundation modelled as a single degree of freedom inverted pendulum model, which rotates around the ankle joint.

When standing quiet, most of the counteracting torque to stabilize the body for small perturbations is exerted by the ankle muscles; this is also known as ‘ankle strategy’. Therefore, the human body is modelled as a single degree of freedom inverted pendulum model which rotates around the
Modeling of human balance dynamics

The equation of motion for this model are

\[ I_b \ddot{\theta} = m_b g h \sin(\theta) + u_a, \]  

(2.1)

where \( I_b \) represents the moment of inertia of the human body around the ankle joint, \( \theta \) the tilt angle, \( m_b \) the upper body mass, \( g \) the gravity acceleration, \( h \) the distance between ankle joint and center of mass of the upper body, and \( u_a \) the total ankle torque exerted by the controller.

One remark is that this model is only valid for small internal and external perturbations. Because when large perturbations are involved, the resulting ground force reaction vector \( F_v \) might become near the edge of foot support. In that case, the human body uses a different stabilizing strategy; for instance, a combined strategy where both hip and ankle muscles are activated.

In order to check whether this model is valid, we need to calculate the location where the ground force reaction vector acts on the foot, denoted by COP. For this calculation we need the ground force reaction vector \( F_v \) and the torque at the toe \( u_t \). With those two we can calculate the distance of the COP from the toe as follows

\[ l_{cop} = \frac{u_t}{F_v}. \]  

(2.2)

When the calculated distance becomes negative or exceeds the length of the foot, the used model is not valid. To determine the forces of constraint that act on the foot, a method is used that introduces a extra degree of freedom for each constraint. The resulting Lagrange equations of motion can then be solved for the generalized forces of constraint.

### 2.2 Compliant surface model

The compliant surface model is the same as the previous model except that the underground has a finite stiffness where the previous model underground was considered infinitely stiff. This introduces two extra degrees of freedom, namely displacement of the whole body in vertical direction and rotation of the ankle around the toe. Displacement in horizontal direction of the toe is for simplicity ignored since the term will be very small and it has no contribution to the research objective. A schematic representation of the compliant surface model is depicted in Figure 2.2. The generalized coordinates of this model are given by \( q = [\theta, \phi, y]^T \).

In contrast to the rigid surface model, the equations of motion of this model are not straightforward and are therefore derived using Lagrange’s equations

\[ \frac{d}{dt} \left( L_{\dot{q}} \right) - L_\dot{q} = (Q^{nc})^T, \]  

(2.3)

where \( L \) is the Lagrangian, which is defined as the difference between the kinetic energy and potential energy \( L = T - V \), and \( Q^{nc} \) are the non-conservative generalized forces. The column of generalized coordinates are denoted by \( q \), and \( (\cdot)_{\dot{q}} \) denotes \( \frac{\partial(\cdot)}{\partial q} \).

The kinetic energy of the system given by the translational and rotational velocity of both point masses equals

\[ T = \frac{1}{2} m_b \left( \dot{y} + l_1 \cos(\alpha + \phi) \dot{\phi} - h \sin(\theta) \dot{\theta} \right)^2 + l_1 \sin(\alpha + \phi) \dot{\phi} - h \cos(\theta) \dot{\theta} \right)^2 \]  

\[ + \frac{1}{2} m_a \left( \dot{y}^2 + d_0 \dot{\phi}^2 + 2 d_0 \cos(\alpha + \phi) \dot{y} \dot{\phi} \right). \]
Figure 2.2: Human body standing quietly on a compliant surface modeled as an inverted pendulum model, which rotates around the ankle joint. The ankle itself can displace in vertical direction and rotate around the toe. Horizontal displacement of the toe is ignored.

The potential energy contains the energy stored in both springs and the gravitational potential energy of both point masses and is given by

\[
V = m_a g (y + d_0 \sin(\alpha + \phi)) + m_b g (y + l_1 \sin(\alpha + \phi) + h \cos(\theta)) + \frac{1}{2} k (y - y_0)^2 \\
+ \frac{1}{2} k (y - y_0 + l_0 \sin(\phi))^2.
\]

The column of non-conservative generalized forces is defined as

\[
Q^{nc} = \sum_{i=1}^{n_F} \left( \frac{\partial \vec{r}_i}{\partial \dot{q}} \right)^T \bar{F}^{nc}_i + \sum_{j=1}^{n_M} \left( \frac{\partial \theta_j}{\partial \dot{q}} \right)^T \bar{w}(\theta_j) \cdot \bar{M}^{nc}_j,
\]

where \(n_F\) represents the number of non-conservative forces, and \(n_m\) represents the number of non-conservative generalized moments. In this case there is only one non-conservative moment, namely the torque \(u_a\) applied by the controller. Consequently, the column of non-conservative generalized forces is given by \(u_a\) for the rigid surface model and \([u_a, 0, 0]^T\) for the compliant surface model.

Model parameters belonging to both models are listed in Table 2.1 and are obtained from [1, 13] except for the ankle and foam parameters. The foam stiffness coefficient \(k\) is obtained from [14] (in fact it gives three stiffness values, soft, medium and firm), and the foam damping coefficient \(d\) is constructed by assuming a dimensionless damping of 1.5. This means an over-damped system since on foam the ankle is expected to return to an equilibrium without oscillations. The ankle parameters are obtained from various sources which cannot be used as reference.

2.3 Central nervous system

Both formulated models are an inherently unstable system due to the inverted pendulum. Therefore, a controller is present that provides the ankle joint with a torque to stabilize the human body standing
Table 2.1: Model parameter values used in this study.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m_a )</td>
<td>60</td>
<td>[kg]</td>
<td>Ankle mass</td>
</tr>
<tr>
<td>( m_b )</td>
<td>60</td>
<td>[kg]</td>
<td>Upper body mass</td>
</tr>
<tr>
<td>( I_b )</td>
<td>60</td>
<td>[kgm²]</td>
<td>Inertia of the body around the ankle</td>
</tr>
<tr>
<td>( h )</td>
<td>1</td>
<td>[m]</td>
<td>Distance between center of mass and ankle joint</td>
</tr>
<tr>
<td>( g )</td>
<td>9.81</td>
<td>[m/s²]</td>
<td>Acceleration of gravity</td>
</tr>
<tr>
<td>( k_{soft} )</td>
<td>6909</td>
<td>[N/m]</td>
<td>Soft foam stiffness coefficient</td>
</tr>
<tr>
<td>( d_{soft} )</td>
<td>1932</td>
<td>[Ns/m]</td>
<td>Soft foam damping coefficient</td>
</tr>
<tr>
<td>( k_{medium} )</td>
<td>33943</td>
<td>[N/m]</td>
<td>Medium foam stiffness coefficient</td>
</tr>
<tr>
<td>( d_{medium} )</td>
<td>4281</td>
<td>[Ns/m]</td>
<td>Medium foam damping coefficient</td>
</tr>
<tr>
<td>( k_{firm} )</td>
<td>79911</td>
<td>[N/m]</td>
<td>Firm foam stiffness coefficient</td>
</tr>
<tr>
<td>( d_{firm} )</td>
<td>6569</td>
<td>[Ns/m]</td>
<td>Firm foam damping coefficient</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>0.53</td>
<td>[rad]</td>
<td>Geometric ankle angle</td>
</tr>
<tr>
<td>( d_\theta )</td>
<td>0.2</td>
<td>[m]</td>
<td>Distance of center of mass of the ankle to toe</td>
</tr>
<tr>
<td>( l_\theta )</td>
<td>0.275</td>
<td>[m]</td>
<td>Length of foot</td>
</tr>
<tr>
<td>( l_\parallel )</td>
<td>0.23</td>
<td>[m]</td>
<td>Distance from toe to ankle</td>
</tr>
<tr>
<td>( y_0 )</td>
<td>0.132</td>
<td>[m]</td>
<td>Undeformed spring length</td>
</tr>
</tbody>
</table>

quitely. This controller consists of two parts, namely a passive and an active part \( u_a = u_{a_{pas}} + u_{a_{act}} \). The passive part arises from the intrinsic mechanical impedance of the ankle joint and is modelled as

\[
u_{a_{pas}} = -K\theta - B\dot{\theta},
\]

where \( K \) is the intrinsic stiffness coefficient, and \( B \) the intrinsic viscosity coefficient. The active part represent the ankle torque activated by the CNS controller. This controller uses three major sensory systems for balance and posture control [20]. The first system is vision, which is primarily involved in planning our locomotion. The second system, our 'gyro' or vestibular system senses linear and angular acceleration. And the last system is the somatosensory system, which is a multitude of sensors that sense the position and velocity of all body segments, their contact with external objects, and the orientation of gravity. To model this complex behavior into a mathematical model, event-driven intermittent control has been proposed in which control is altered when state-dependent thresholds are passed [1, 12] and is modelled as

\[
u_{a_{act}} = \begin{cases} 
-P\theta_{\tau} - D\dot{\theta}_{\tau}, & \text{if } \theta_{\tau}(\dot{\theta}_{\tau} - a\theta_{\tau}) > 0 \text{ and } \theta_{\tau}^2 + c\dot{\theta}_{\tau}^2 > r^2, \\
0, & \text{otherwise},
\end{cases}
\]

where \( P \) is the intermittent stiffness coefficient, \( D \) the intermittent viscosity coefficient, \( \alpha \) the slope of the boundary line for the switching controller, \( c \) a constant to correct dimensions for addition, \( r \) the radius of the deadzone around the origin, \( \theta_{\tau} \) denotes \( \theta(t-\tau) \), and \( \dot{\theta}_{\tau} \) denotes \( \dot{\theta}(t-\tau) \). The controller parameter values are listed in Table 2.2.

The non-linearity in (2.5) represents the limited sensitivity of the sensors detecting the body tilt and corresponding falling velocity [1], and the choice can be explained with the 'phase portrait' of the rigid surface model without active control \( u_{a_{pas}} \). Note that the phase space of a DDE is infinitely dimensional (this will be explained in the next chapter) and a state is therefore described by a state segment. The \( \theta-\dot{\theta} \) plane can therefore not be a 'phase portrait' of the system. Nevertheless, with keeping this carefully in mind, we refer to it as the 'phase portrait'. This 'portrait' is depicted in
2.3 Central nervous system

Table 2.2: Controller parameter values.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K$</td>
<td>471</td>
<td>[Nm/rad]</td>
<td>Intrinsic stiffness coefficient</td>
</tr>
<tr>
<td>$B$</td>
<td>4</td>
<td>[Nms/rad]</td>
<td>Intrinsic viscosity coefficient</td>
</tr>
<tr>
<td>$P$</td>
<td>147.15</td>
<td>[Nm/rad]</td>
<td>Intermittent stiffness coefficient</td>
</tr>
<tr>
<td>$D$</td>
<td>10</td>
<td>[Nms/rad]</td>
<td>Intermittent viscosity coefficient</td>
</tr>
<tr>
<td>$\tau$</td>
<td>0.2</td>
<td>[s]</td>
<td>Delay in feedback loop</td>
</tr>
<tr>
<td>$r$</td>
<td>0.004</td>
<td>[rad − rad/s]</td>
<td>Radius of the deadzone in the phase plane</td>
</tr>
<tr>
<td>$a$</td>
<td>-0.1</td>
<td>[1/s]</td>
<td>Slope of boundary line</td>
</tr>
<tr>
<td>$c$</td>
<td>1</td>
<td>[s²]</td>
<td>Constant to correct dimension</td>
</tr>
</tbody>
</table>

Figure 2.3 and shows the local direction of the vector field. The solid lines with arrows represent the eigenvectors of the system, stable if the arrows are pointing towards the origin, unstable if not.

Figure 2.3: ‘Phase portrait’ of the rigid surface model with controller $u_a = u_a_{pas}$. The eigenvectors of this system are represented by the solid lines with arrows and indicates the stable and unstable manifold.

Note that in the ‘phase portrait’, the trajectories are parallel to the stable eigenvector far from the origin but that they become tangent to the stable eigenvector as they approach the origin. Therefore, once a state segment lies along the stable manifold, it is not necessary to apply a control action since the state segment automatically converges towards the origin, which is the control objective. After some time, when the state segment approaches the origin, it will start to drift away from the stable manifold towards the unstable manifold and a control action is required to change the local direction of the vector field and direct the state segment back towards the stable manifold. Therefore the phase space is separated in two different regions. One where only passive control is used and one where passive and active control is used. The region where only passive control is used, which lies along the stable eigenvector, is shown in gray and the white region indicates where also active control is
used. The deadzone around the origin is present to match the model with the normal sway behavior of a human body while standing quietly.

This altering of control input, which is called ‘event-driven intermittent’ control, makes the system a PWS delay system since the system is composed of different smooth vector fields.

2.4 Time domain simulations

In order to gain some insight in the system dynamics before we continue with obtaining periodic orbits and to explore difference in system dynamics between both systems, simulations are made. Since both systems use event-driven intermittent control, it requires event detection and the need to restart the simulation with updated model equations each time an event has occurred.

Each simulation is performed with the same initial conditions, namely a small perturbation from the upright position $\theta = 0.01 \text{ [rad]}$ and for the compliant surface model an initial height equal to the foam thickness. The initial conditions of the other states are zero. This gives us the opportunity to compare the effects of different model parameters such as stiffness of the foundation.

The results of all the simulations for the four different stiffness values of the foundation (rigid, soft, medium, and firm) are shown in Appendix A. From these results, we can deduce that independent of the foundation stiffness value the solutions appear to have a similar shape for the given set of parameters. Of course one can notice differences in dynamic behavior and especially in the first part of the trajectory since the compliant surface model also captures the dynamic behavior of the foot, but after the foot has settled to an equilibrium the sway behavior $\theta$ appears to be similar. However, the difference in initial dynamic behavior can have a significant influence since the location of the COP might become near the edge of foot support and the person might tip over or a different strategy have to be used. The formulated models are then not valid anymore and must considered to be unstable. This check is easy for the rigid surface model as explained in Section 2.1 but the check for the compliant surface model requires further research.
Delay systems

In the previous chapter, the equations of motion have been derived that describe the system dynamics. Furthermore, figures have been presented that show the simulation results. In these figures, one can notice that sometimes the solutions converge to a limit cycle. In this chapter, numerical tools are introduced that can determine these limit cycles and stability as function of parameters. First, a brief explanation of delay differential equations (DDEs) is given. Subsequently, periodic solutions of DDEs are discussed. Finally, an elaboration on the numerical tools used in this study to calculate periodic solutions and their stability is given.

3.1 Delay differential equations

DDEs are differential equations, which not only depend on the value of the state at the present time but also on values of the state at other moments in time. Those equations are quite common in science and engineering because due to finite response times delays can be put into almost any system. Consider for example a control system, the controller acts than on the error signal, which is the difference between the measurement and reference signal. The controller needs, however, some time to sample, encode, and process the signal and reacts therefore not instantaneously. So a delay is present in the feedback loop. In most systems, the time delay is small and the effect on the solution is hardly noticeable and can therefore be neglected, but in some systems it is vital to take time delay into account.

DDEs exist in a lot of different types, such as time/state dependent delays, dependency on the future or on the state derivative, and many more different types. In this report we restrict ourselves to autonomous delay differential equations

\[ \dot{x} = f(x(t), x(t - \tau_1), \ldots, x(t - \tau_k), \gamma), \]

where \( x \in \mathbb{R}^n, f : \mathbb{R}^{n \times (k+1)} \times \mathbb{R} \to \mathbb{R}^n \) is a smooth nonlinear function, delays \( \tau_i \geq 0 \), and parameters \( \gamma \in \mathbb{R} \).

For an ODE, a unique response can be computed starting with an initial condition \( x(t_0) \) and in the case for a non-autonomous system with also initial time \( t_0 \). For a DDE, one needs to provide an initial state segment, \( x(t_0 + s) \) for \( s \in [-\tau, 0] \), where \( \tau = \max_i \tau_i \). Because to compute the rate of change at \( t_0 \), one needs to compute \( x(t_0) \) together with \( x(t_0 - \tau) \), and for the rate of change at \( t_0 + \epsilon \), one needs to compute \( x(t_0 + \epsilon) \) together with \( x(t_0 + \epsilon - \tau) \), where \( \epsilon \in [0, \tau] \). Consequently, the state space of DDEs is infinite dimensional.

Since the initial state segment is arbitrary, the left and right derivative normally not match at the initial time \( t_0 \) because the left derivative is arbitrary and the right derivative is determined by the DDE. Consequently, for most problems there exist a discontinuity at the initial time \( t_0 \). An example of this is depicted in Figure 3.1. As can be seen in this figure, a breakpoint or first order derivative discontinuity is present at the initial time \( t_0 = 0 \), and this discontinuity is propagated in time as a second order derivative discontinuity at \( \tau \), even when \( f \) is infinitely smooth. More generally, a
Figure 3.1: A typical response of a DDE with a first order discontinuity at initial time $t_0 = 0$ which is propagated in time as a $(n+1)$-th order discontinuity at $n\tau$.

discontinuity in the $(n+1)$-th order derivative occurs at $n\tau$. This automatically means that solutions are smoothed as time increases.

### 3.2 Periodic solutions of DDEs

A periodic solution $x^*(t)$ of (3.1), is a solution of (3.1), which is periodic with time $T > 0$

$$x^*(t + T) = x^*(t) \quad \forall t.$$  \hspace{1cm} (3.2)

As explained in the previous section, it is not sufficient to consider only one time point because of the dependency on the past. Instead, a whole function segment, $x^*(t + s)$ for $s \in [-\tau, 0]$, needs to be repeated to consider it a periodic solution. The problem can be formulated as a two point-boundary value problem (TP-BVP)

$$\dot{x} = f(x(t), x(t - \tau_1), \ldots, x(t - \tau_k)),$$  \hspace{1cm} (3.3a)

$$x_0 = x_T,$$  \hspace{1cm} (3.3b)

$$p(x_0, T) = 0,$$  \hspace{1cm} (3.3c)

where $x_0$ is the function segment $x(0 + s)$ for $s \in [-\tau, 0]$, $x_T$ is the function segment $x(T + s)$ for $s \in [-\tau, 0]$, and $p(x_T, T)$ is an extra condition to remove translational invariance in time because every function segment on the periodic solution is a solution.

Stable periodic solutions of a DDE can be found by numerical time integration since the solution automatically converges towards the stable orbit. Both stable and unstable solutions, however, can be computed by solving the above TP-BVP by either a shooting or collocation method.

### 3.3 Shooting for DDEs

Different techniques exist for finding periodic solutions of nonlinear systems. The shooting method is a popular one and is discussed in this section.

A periodic solution is uniquely determined by an initial function segment $x_0$ and the period $T$. In order to find those two, the TP-BVP (3.3) is slightly altered using

$$r(x_0, T) := x_0 - x_T = 0,$$  \hspace{1cm} (3.4a)

$$p(x_0, T) = 0.$$  \hspace{1cm} (3.4b)
where \( \begin{bmatrix} \{0 \} & \{0 \} \end{bmatrix} \), with \( n \), is solved using Newton’s iterations, which is a local linearization around the solution, by applying

\[
x^{(k+1)}_0 = x^{(k)}_0 + \Delta x^{(k)}_0, \quad T^{(k+1)} = T^{(k)} + \Delta T^{(k)},
\]

obtained from solving

\[
\begin{align*}
\frac{\partial r(x_0, T)}{\partial x_0} \Delta x_0 + \frac{\partial r(x_0, T)}{\partial T} \Delta T &= -r(x_0, T), \\
\frac{\partial p(x_0, T)}{\partial x_0} \Delta x_0 + \frac{\partial p(x_0, T)}{\partial T} \Delta T &= -p(x_0, T),
\end{align*}
\]

or after evaluating the partial derivatives and defining \( M_T(T, x_0) \) as the nonlinear solution operator which integrates the initial condition \( x_0 \) over time \( T \) as

\[
\begin{bmatrix} \frac{\partial M_T(T, x_0)}{\partial r(x_0, T)} - I & \frac{\partial M_T(T, x_0)}{\partial p(x_0, T)} \frac{\partial r(x_0, T)}{\partial T} \frac{\partial p(x_0, T)}{\partial T} \end{bmatrix} \left\{ \begin{array}{c} \Delta x_0^{(k)} \\ \Delta T^{(k)} \end{array} \right\} = - \left\{ \begin{array}{c} r \\ p \end{array} \right\} \bigg|_{(x_0^{(k)}, T^{(k)})}.
\]

Note, however, that the system only converges when the initial guess \((x_0^{(0)}, T^{(0)})\) is close enough to the actual solution. Otherwise, the algorithm might diverge and a new initial point needs to be provided, for example, with functions that are both convex and concave.

At convergence, the nonlinear solution operator \( M_T(T, x_0) \) equals the Monodromy matrix and the eigenvalues of this matrix, the so called Floquet multipliers, determine the stability of the periodic solution. If all the eigenvalues lie within the unit circle in the complex plane, besides the trivial eigenvalue at 1 with eigenvector equal to the initial vector field, then locally asymptotic orbital stability can be guaranteed.

### 3.4 Collocation for DDEs

A more robust alternative to shooting methods are collocation methods [18]. In collocation a periodic solution is computed by using piecewise-polynomials that satisfy the differential equation at a set of collocation points on \([0, T]\). The following approach is a summary obtained from [18].

Before computing the periodic solution, a scaled time variable is defined \( s := t/T \) and substituted into (3.3) together with \( y(s) := x(t) \) to obtain the time scaled system

\[
\begin{align*}
dy/ds &= T f(y(s), y(s - \tau_1/T), \ldots, y(s - \tau_\gamma/T), \gamma), & \text{for } s \in [0, 1], \\
y_1(\theta) &= y_0(\theta) = 0, & \text{for } \theta \in [-\tau/T, 0], \\
p(y, T) &= 0.
\end{align*}
\]

Now let \( \Pi_m^n = \{0 = s_0 = s_1 < \ldots < s_m = 1\} \) be a mesh on \([0, 1]\) with \( m + 1 \) mesh points. This mesh is periodically extended to the left to obtain a mesh \( \Pi_{m+1}^n = \{s_{-\ell}, \ldots, 0, \ldots, s_m = 1\} \) on \([s_{-\ell}, 1] = [-\tau/T, 1] \) with \( l + m \) intervals, and determine \( l \) such that \( s_{-\ell} \leq -\tau/T < s_{-\ell+1} \).

Denote by \( \mathbb{P}_d^m \) the set of polynomials of degree not exceeding \( d \). A function \( y \in \mathcal{C}([s_{-\ell}, 1], \mathbb{R}^n) \), where \( \mathcal{C}([s_{-\ell}, 1], \mathbb{R}^n) \) represents the space of continuous functions mapping the delay interval into \( \mathbb{R}^n \), is then approximated by an element \( u \) from the following space of piecewise polynomials

\[
\mathbb{P}_d^m(\Pi_{m+1}^n) := \left\{ u \in \mathcal{C}([s_{-\ell}, T], \mathbb{R}^n) : u|_{[s_i, s_{i+1}]} \in \mathbb{P}_d^m|_{[s_i, s_{i+1}]} \right\},
\]

for \( i = -\ell, \ldots, m - 1 \).
The elements can be represented in a convenient way. For this let \( \Delta s_i := s_{i+1} - s_i \) and \( s_{i+\frac{1}{2}} := s_i + \frac{1}{2}\Delta s_i \) for \( i = -l, \ldots, L - 1 \) and \( j = 0, \ldots, d \). It is possible to choose scalar basis functions \( \phi_{i+\frac{1}{2}} \in P^1(\Pi^n_d) \) that satisfy \( \phi_{i+\frac{1}{2}}(s_{i+\frac{1}{2}}) = \delta_{i,j}, \) for \( i, j = -l, \ldots, m - 1 \) and \( j = 0, \ldots, d \), with \( \delta_{i,j} \) the Kronecker delta. See for example Figure 3.2, in which Lagrange basis polynomials of degree \( d = 5 \) are depicted. Clearly, at the mesh points only one polynomial equals one and the others equal zero.

\[
\begin{array}{c}
\psi_0 & \psi_1 & \psi_2 & \psi_3 & \psi_4 & \psi_5 \\
\end{array}
\]

Figure 3.2: Lagrange basis functions \( \psi_j \) of degree \( d = 5 \).

Hence, a piecewise polynomial \( u \in P^n_d(\Pi^m_{-\ell}) \) can be written as a linear combination of \( \phi_{i+\frac{1}{2}} \) with ‘coordinates’ \( u_{i+\frac{1}{2}} := u(s_{i+\frac{1}{2}}) \in \mathbb{R}^n \),

\[
u(s) = \left( \phi_{-\ell}(s)u_{-\ell} + \sum_{i=-\ell}^{d-1} \sum_{j=1}^{d} \phi_{i+\frac{1}{2}}(s)u_{i+\frac{1}{2}} \right) + \sum_{i=0}^{m-1} \sum_{j=1}^{d} \phi_{i+\frac{1}{2}}(s)u_{i+\frac{1}{2}}.
\]

The approximation \( u(s) \) is therefore completely defined by the ‘coordinates’ \( u_{i+\frac{1}{2}} \). Because polynomials on adjacent intervals share the value at the common mesh points, this representation is automatically continuous. It does, however, not have to be continuously differentiable at the mesh points.

Let us now define \( u_s \) as the vector of length \( N := n(\ell d + 1) \) that groups the \( u_{i+\frac{1}{2}} \) from the first bracketed part of the summation above and \( u_t \) as the vector of length \( N_c := nmd \) that groups the \( u_{i+\frac{1}{2}} \) from the second part of this summation. Clearly, \( \dim P^n_d(\Pi^m_{-\ell}) = N + N_c \). Furthermore, the vector \( u_f \) is defined as the vector that contains the last \( N \) components of the vector \( [u_s^T, u_t^T]^T \). Hence,

\[
u_s := \begin{bmatrix} u_{-\ell} \\ \vdots \\ u_{i+\frac{1}{2}} \\ \vdots \\ u_0 \end{bmatrix}, \quad u_f := \begin{bmatrix} u_{m-\ell} \\ \vdots \\ u_{i+\frac{1}{2}} \\ \vdots \\ u_m \end{bmatrix} \in \mathbb{R}^N \quad \text{and} \quad u_t = \begin{bmatrix} u_{\frac{1}{2}} \\ \vdots \\ u_{i+\frac{1}{2}} \\ \vdots \\ u_m \end{bmatrix} \in \mathbb{R}^{N_c}. \quad (3.8)
\]

Next a set of collocation points is defined as \( C(\Pi^m) := \bigcup_{i=0}^{m-1} \bigcup_{\ell=1}^{d} \{ \ell \} \cup_{i=0}^{m-1} \{ c_{1,\nu} := s_i + c_{\nu}\Delta s_i \} \) on \( [0, 1] \) based on the given set of collocation parameters \( c_1 < c_2 < \cdots < c_d \) on \( [0, 1] \) with \( c_1 \neq 0 \) and/or
$c_d \neq 1$. In this study, gauss points are used, that is the zeros of the Legendre polynomial of degree $d$ on $[0, 1]$. The Legendre polynomial and collocation parameters are depicted in Figure 3.3. One can notice that the collocation points do not coincide with the mesh points. This improves the accuracy of the computed orbit [6].

![Figure 3.3: Collocation points defined as the zeros of the Legendre polynomial.](image)

Then the idea of a collocation method for approximating a solution to the DDE (3.1) is to find a function $u \in \mathbb{P}_d^m(\Pi^m)$ that satisfies the DDE (3.7a) in a finite set of points $C(\Pi^m)$, an example of this is shown in Figure 3.4, and also fulfils the relevant initial or boundary value conditions (3.7b) and (3.7c).

![Figure 3.4: Collocation polynomial $u(s)$ restricted to satisfy the DDE at the collocation points $c_j$ for $j = 1, \ldots, d$ and the value at $s_{i+1}$.](image)

The group of $md$ vector-valued collocation requirements is defined as

$$r_1 := T f(\bar{u}_{i, \nu}) - u'(c_{i, \nu}), \quad (3.9)$$

where $u'(c_{i, \nu})$ denotes the derivative of $u(c_{i, \nu})$, and with $\bar{u}_{i, \nu} := (u(c_{i, \nu}), \ldots, u(c_{i, \nu} - \tau_{i, \nu}/T))$, for $i = 0, \ldots, m - 1$ and $\nu = 1, \ldots, d$. The discretization of the periodicity condition (3.7b) requires the equality of the start and final segment, $u_s$ and $u_f$ respectively. In total the discretization of (3.7) reads

$$r_1(u_s, u_f, T) = 0_{N \times 1}, \quad (3.10a)$$
$$r_2(u_s, u_f) := u_f - u_s = 0_{N \times 1}, \quad (3.10b)$$
$$p(u_s, u_f) = 0. \quad (3.10c)$$
The choice of basis functions $\phi_{i+k/\Delta}$, allows to write the restriction $u|_{s_i,s_{i+1}}$, for $i = -\ell, \ldots, m-1$, in the form

$$u|_{s_i,s_{i+1}}(s) = \sum_{j=0}^{d} \phi_{i+\frac{j}{\Delta}}(s) u_i + \frac{j}{\Delta},$$

where

$$\psi_j(\xi) := \prod_{k=0, k \neq j}^{d} \frac{\xi d - k}{j - k}, \quad \text{for } j = 0, \ldots, d,$$

are the Lagrange polynomials of degree $d$ on $[0, 1]$, which are shown in Figure 3.2. Hence, $\psi_j(j/d) = \delta_j,j$ for $j, i = 0, \ldots, d$. Substitution of (3.11) in (3.9) gives

$$T f(\hat{u}_i) - \sum_{j=0}^{d} \psi_j(c_i) u_i \equiv 0_{n \times 1},$$

for $i = 0, \ldots, m-1$ and $\nu = 1, \ldots, d$.

The nonlinear collocation system (3.10) consist of $N + N_e + 1$ equations and the same number unknowns, namely $[u_i^T, u_{i+1}^T]^T$ and $T$. Since we are unable to solve this system directly, we use Newton’s method. This method makes a local linearization of the nonlinear function $f(x_0)$, which actually is its tangent, and computes the $x$-intercept of this tangent line. This gives the new point $x_1$, and the procedure is repeated till the norm of the residual $f(x_1)$ is inside a predefined boundary. To use this method for our system of equations, we linearize the system with respect to $u$ and $T$ and write it in matrix representation. For the first $N_e$ rows we obtain

$$\sum_{j=0}^{d} \left( T \sum_{k=0}^{N_e} \psi_j\left( \frac{(c_{i,\nu} - \tau_{s_k}/T) - s_{i,\nu,k}}{\Delta s_{i,\nu,k}} \right) \frac{\partial f}{\partial y^k}() \Delta u_{i,\nu,k} + \frac{j}{\Delta} \right) - \psi_j(c_i) u_i = 0_{n \times 1},$$

for $i = 0, \ldots, m-1$, $\nu = 1, \ldots, d$. Using (3.8) and

$$A := \frac{\partial r_1}{\partial u_s} \in \mathbb{R}^{N_e \times N_e}, \quad B := \frac{\partial r_1}{\partial u_s} \in \mathbb{R}^{N_e \times N_e},$$

the terms in (3.13) can be reordered to obtain

$$A \Delta u_s + B \Delta u_t + r_{1,T} \Delta T = -r_1$$

The next $N$ rows comes from the linearization of the periodicity condition

$$\Delta u_f = \Delta u_s = -r_2,$$

and the last row comes from the linearization of the phase condition

$$p_u \Delta u_s + p_u \Delta u_t + p_T \Delta T = -p.$$
where $p_{us}, p_{ut},$ and $p_T$ are partial derivatives. The matrix representation, which is depicted in Figure 3.5, assumes the form $J\Delta x = -r$, where $J$ represents the jacobian, $x$ the vector of independent variables, and $r$ the function residuals. Each iteration the update is computed with

$$\Delta x = -J^{-1}r,$$

(3.18)
till the norm of the residuals are within a certain predefined boundary.

Figure 3.5: Matrix representation of the linearization of the system of nonlinear collocation equations. It is used to compute the update $\Delta x$ from $J\Delta x = -r$. 

$$
\begin{bmatrix}
A & B & r_{1,T} \\
-I_N & I_N & \Delta u_s \\
p_{us} & p_{ut} & p_T \\
\end{bmatrix}
\begin{bmatrix}
\Delta u_s \\
\Delta u_t \\
\Delta T \\
\end{bmatrix}
= -
\begin{bmatrix}
r_1 \\
r_2 \\
 p \\
\end{bmatrix}.$$
4

Collocation for piecewise-smooth delay systems

In the previous chapter, the collocation method for delay systems has been described. This procedure, however, is only valid for systems with smooth dynamics. In this chapter, the collocation method is extended such that it can handle PWS systems. In other words, systems that consist of different smooth vector fields coupled via switching boundaries. The method that is used for this is obtained from the paper of David Barton [2]. We have tried to adopt the shooting method and make that work PWS system but we were not successful in this, since it is difficult to determine saltation matrices for infinite dimensional state segments.

This chapter starts with the definition of PWS systems as used in this study together with an explanation of the collocation method for PWS delay systems. In Section 4.2, results from [2] are recalculated to validate the Matlab implementation. Next in Sections 4.3 and 4.4, the validated program is used on the models that were presented in Chapter 2.

4.1 PWS delay systems

A PWS-DDE is defined as a collection of smooth vector fields and its simplest form is given as

\[
\dot{x}(t) = \begin{cases} 
  f_1(x(t), x(t - \tau)), & \text{if } h(x(t), x(t - \tau)) \leq 0, \\
  f_2(x(t), x(t - \tau)), & \text{if } h(x(t), x(t - \tau)) > 0,
\end{cases}
\]

where \( x(t) \in \mathbb{R}^n \), and \( f_1, f_2, h \) are sufficiently smooth functions. This system is composed of two smooth vector fields and a switch between these field is made when the state segment crosses the switching manifold \( h(x(t), x(t - \tau)) = 0 \).

This definition is, however, not valid for systems that are composed of more than two vector fields and possible non trivial boundaries between them. (This means when there is no strict order of switching between them.) Therefore, we use the definitions given by [17] and the slightly altered version for delay systems [2]. A PWS-DDE is then defined as a collection of smooth vector fields

\[
\dot{x}(t) = f_m(x^t),
\]

indexed by a mode \( m \) in some finite set of modes \( M \). Where \( x^t \in C([-\tau, 0], \mathbb{R}^n) \) is the solution segment \( x(t + s) \) for \( s \in [-\tau, 0], \) with \( \tau = \max_i \tau_i \). Here \( C([-\tau, 0], \mathbb{R}^n) \) is the space of continuous functions mapping the delay interval into \( \mathbb{R}^n \). Furthermore, denote by \( E \) the finite set of events, and associate to each event \( e \) a pair \( \pi_e = (m_{in}, m_{out}) \), a smooth event function \( h_e : C([-\tau, 0], \mathbb{R}^n) \to \mathbb{R} \), and a smooth jump function \( g_e : C([-\tau, 0], \mathbb{R}^n) \to C([-\tau, 0], \mathbb{R}^n) \). The switching manifold defined by the event function \( h_e = 0 \) marks the transition point between the (potentially) different vector fields \( (f_{m_{in}}, f_{m_{out}}) \), and the jump function \( g_e \) determines the instantaneous change of state when an impact with the manifold occurs.

A solution of a PWS-DDE on a finite interval \([t_0, t_N]\) is a sequence \( \xi = \{x_j : [t_{j-1}, t_j] \to \mathbb{R}^n\}_{j=1}^N \) of \( N \) smooth curves, an associated sequence of modes \( \{m_j\}_{j=1}^N \), and an associated sequence of events \( \{e_j\}_{j=1}^{N-1} \), such that the corresponding tangent vector \( x^t_j \) equals \( f_{m_j}(x^t_j) \), and such that, for
\( j \in [1, N-1] \), the pair of modes \((m_j, m_{j+1})\) and the pair of states segments \((x_{t_j}^i, x_{t_{j+1}}^i)\) are connected by the event \(e_j\), that is
\[
h_{e_j}(x_{t_j}^i) = 0, \tag{4.3}
\]
and
\[
g_{e_j}(x_{t_j}^i) = x_{t_{j+1}}^i. \tag{4.4}
\]

The concatenation \(\Sigma = \{m_1, e_1, m_2, e_2, \ldots, m_{N-1}, e_{N-1}, m_N\}\) is defined as the solution’s signature. This definition makes it possible to formulate a particular periodic orbit as a solution of a multi point-boundary value problem (MP-BVP) and takes the form
\[
\dot{x}_i(t) = f_{m_i}(x_{t_i}^i) \quad \text{for} \quad 0 \leq t \leq T_i, \tag{4.5a}
\]
where \(i = 0, 1, \ldots, N-1\), with \(N\) scalar boundary conditions given by
\[
h_{e_i}(x_{T_i}^i) = 0, \tag{4.5b}
\]
and \(N\) function boundary conditions
\[
x_{(i+1) \text{ mod } N}^0 = g_{e_i}(x_{T_i}^i). \tag{4.5c}
\]

Since that time spent in each interval \(T_i\) is normally not known a priori, it is normal to rescale time just as we did in the previous chapter by defining \(s := t/T\) and \(y(s) := x(t)\) such that the time spent in each interval is unity, and the true time appears as a parameter in the MP-BVP
\[
\dot{y}_i(s) = T_i f_{m_i}(y_s^i) \quad \text{for} \quad 0 \leq s < 1, \tag{4.6a}
\]
\[
h_{e_i}(y_s^i) = 0, \tag{4.6b}
\]
\[
y_{(i+1) \text{ mod } N}^0 = g_{e_i}(y_s^i), \tag{4.6c}
\]
where \(y_s^i\) is the function segment \(y_i(s + \theta)\) for \(-\tau/T_i \leq \theta \leq 0\).

The approach that is taken to solve this system is the same as in the previous chapter. First, the solution is discretized with orthogonal polynomials \(u(s)\). Then, the obtained system of nonlinear algebraic collocation equations are solved with Newton’s method. A flowchart describing this approach is presented in Appendix C.

The collocation method for PWS delay systems is implemented in Matlab by adjusting an existing Matlab program DDE-BIFTOOL [7], which originally is created for smooth delay systems. Each interval has its own mesh defined as \(\Pi_i := [t_0 = -\tau, \ldots, t_{L_\tau} = 0, \ldots, t_{L_\tau + L} = 1]\), where the interior mesh points are arbitrary.

Since a discontinuity in the derivative of the solution occurs every time the state segment impact the switching boundary and these discontinuities are propagated in time as explained in the previous chapter. It may be beneficial to use adaptive (non equispaced) meshing that include the points \(n\tau\) for \(n = 1, \ldots, d\) in the mesh and also extra points where the derivatives are steep. This reduces the required number of mesh points. Another important fact is that each interval has the same number and spacing of mesh points in the time delay section at the begin and end of the interval because these sections have to be equal for the periodicity condition.

It can happen that after Newton updates on the period times \(T_i\) collocation points become outside the mesh interval \([−\tau, 1]\). A remesh of the interval is then required.

For obtaining an initial guess, a simulation of an ‘initial value’ problem is performed because the system will automatically converge towards a stable periodic solution. The simulation result is discretised and split into parts before it can be used in the collocation algorithm.
4.2 Test models

Since the described collocation method is based on the paper of David Barton [2], some of his models are reconstructed to evaluate whether the method is implemented correctly.

First order model

The first model that is compared is a simple delayed oscillator that has a discontinuous vector field at \( x(t - \tau) = 0 \)

\[
\dot{x}(t) = \alpha x(t) - x^3(t - \tau) + \begin{cases} 
\beta, & \text{if } x(t - \tau) \geq 0, \\
\gamma, & \text{if } x(t - \tau) < 0,
\end{cases}
\]

(4.7)

where \( \alpha = 0.4, \beta = -0.15, \gamma = 0.68, \) and \( \tau = 1 \). The numbers are assumed to be dimensionless since there are no units given in [2]. The event function is rather simple and is defined as \( x(t - \tau) = 0 \). The jump function equals identity because when the state segment impacts the switching boundary only the solution derivative is changed; the solution segment itself remains the same. The results after convergence of the collocation algorithm are shown in Figures 4.1 and 4.2 together with the figures obtained from the paper of David Barton. As can be seen, the time response appears to be equal where the phase plane projection slightly differs. The reason for this is unknown.

![Figure 4.1: Time series comparison between first order model result of David Barton [2] (left) and our result (right).](image)

Second order model

The second test model is a second-order DDE that caricatures a pupil light reflex and is described by the following system

\[
\ddot{x}(t) + cx(t) + x(t) = - \begin{cases} 
\Theta_1, & \text{if } x(t - \tau) \geq 0, \\
\Theta_2, & \text{if } x(t - \tau) < 0,
\end{cases}
\]

(4.8)

where \( c = 0.2, \Theta_1 = -\Theta_2 = 1, \) and \( \tau = 6.5 \). The numbers are assumed to be dimensionless since there are no units given in [2]. The event function and jump function for this model are the same as the previous model. In this case both the time series as well the phase-plane representation appears to be similar, as can be seen in Figures 4.3 and 4.4. One note for the time series plot is that the response is mirrored in the vertical axis since a different start segment is chosen. This does not
4.3 Rigid surface model

The previous section has shown that the implementation of the method appears to be correct. In this section, the method is used to compute periodic solutions of the rigid surface model as described in Chapter 2

\[ I_b \ddot{\theta} + B \dot{\theta} + K \theta - m_b g h \sin(\theta) = \begin{cases} -P \theta_{\tau} - D \dot{\theta}_{\tau} & \text{if } \theta_{\tau}(\dot{\theta}_{\tau} - a \theta_{\tau}) > 0 \text{ and } \theta_{\tau}^2 + \epsilon \dot{\theta}_{\tau}^2 > r^2, \\ 0, & \text{otherwise}, \end{cases} \]

(4.9)

where \( \theta_{\tau} \) denotes \( \theta(t - \tau) \) and \( \dot{\theta}_{\tau} \) denotes \( \dot{\theta}(t - \tau) \). The parameter values used for the method are as described in Chapter 2.

The alternating controller provides the model with an alternating torque. This step in the torque upon impact with the switching boundary results via the laws of Newton in a step in the acceleration profile, but the velocity and position changes not instantaneously. Hence, the jump function \( g \) equals...
identity. The event function $h$ requires more attention since it consist of two functions coupled with an and statement. Therefore, we determine whether the end segment lies on the circle or on the line and use the appropriate condition for that case.

To check whether the computed solution is an actual limit cycle we start a numerical integration on the computed branch. Both the computed and numerically integrated solution are depicted in Figure 4.5. As can be seen, the solution obtained by numerical integration fits the computed solution perfect as differences between them are not noticeable. This conclusion is strengthened by the two norm of the error between the solutions, which also is depicted in the figure.

![Phase-plane representation comparison between the second order model result of David Barton [2] (left) and our result (right).](image)

Figure 4.4: Phase-plane representation comparison between the second order model result of David Barton [2] (left) and our result (right).

![Comparison of tilt angle $\theta$ results between the collocation algorithm and numerical integration started on the branch. The left plot shows time series solutions obtained by both numerical integration (solid) and the collocation solution (dashed). The right plot shows the phase plane representation solutions obtained by numerical integration (black) and the collocation solution (gray). Also the two norm of the error between the solutions is depicted.](image)

Figure 4.5: Comparison of tilt angle $\theta$ results between the collocation algorithm and numerical integration started on the branch. The left plot shows time series solutions obtained by both numerical integration (solid) and the collocation solution (dashed). The right plot shows the phase plane representation solutions obtained by numerical integration (black) and the collocation solution (gray). Also the two norm of the error between the solutions is depicted.
4.4 Compliant surface model

In this section, the collocation method for PWS delay systems is used to compute periodic solutions of the compliant surface model. This is done for different kinds of underground stiffness values. Apart from underground stiffness, the same model parameters and initial conditions are used as with the rigid surface model, which makes a comparison possible. Since the same controller is used as the rigid surface model, the jump and event function remains the same.

The result after convergence of the collocation algorithm for the tilt angle $\theta$ with the softest underground stiffness value presented in Chapter 2 is depicted in Figure 4.6. As can be seen, the result appears to be similar to the rigid surface model despite some small perturbations after the switch has occurred. A result that matches with the previous findings in Chapter 2, namely that dynamics of the compliant surface model are similar to the rigid surface except for some initial behavior. However, this is only valid for the given set of parameters. With smaller values of underground stiffness, more differences are noticeable.

![Figure 4.6: Comparison of tilt angle $\theta$ results between the collocation algorithm and numerical integration started on the branch. The left plot shows time series solutions obtained by numerical integration (solid) and the collocation solution (dashed). The right plot shows the phase plane representation solutions obtained by numerical integration (black) and the collocation solution (gray). Also the two norm of the error between the solutions is depicted.](image)

Figure 4.7 presents the time series and phase plane representation of the ankle angle $\phi$ during the periodic solution. From the left figure we can deduce that the ankle has made a rotation and remains (almost) steady in that point since both the solution and first derivative of the solution are respectively on an (almost) constant value and an (almost) zero value. In the right figure we notice a small difference between the solution obtained by numerical integration and the collocation solution. Remarkable is that it occurs in the first sections after the solution impacts the switching boundary. Therefore, it might be caused by an insufficient number of mesh points in the time delay sections of the collocation solution.

In Figure 4.8, the position of the toe $y$ presented during the limit cycle for the soft underground stiffness. From the left figure we can deduce that the vertical displacement of the ankle during the periodic solution remain (almost) steady since both the solution and first derivative of the solution are on an (almost) constant and an (almost) zero value. This conclusion is strengthened with the right figure, which shows the ‘phase portrait’ of the ankle displacement. As can be seen, the displacement and velocity values are negligible. Consequently, it means that the compliant surface model appears
to show similar dynamics during the periodic solution as the rigid surface model.

The figures for the other underground stiffness are for readability presented in Appendix B since they are similar to the presented ones and do not provide extra information.
Continuation and bifurcation analysis

In the previous chapter, a method was explained that can determine periodic solutions in PWS delay systems. This method, refines an already good estimate of the solution. This can be used in the continuation of branches of periodic solutions, which yields a branch of solutions in the solutions-parameter space.

This chapter starts with an expansion of the previously explained method to make it suitable for continuation procedures. Subsequently, numerical problems are explained, which we encountered in performing the continuation procedure. Finally, a bifurcation analysis is performed.

5.1 Continuation of periodic orbits

The periodic solution solver described in the previous chapter can be used in a continuation code. Initially we need a periodic solution. Then a predictor is used to construct a first approximation for the next periodic solution on the branch, i.e. for a slightly different value of the system parameter, and subsequently, the collocation method is used for the corrector iterations that move the estimate onto the solution branch. However, the corrector computes the solution at the same parameter value as the predicted point. This approach might therefore fail in the neighborhood of a cyclic fold bifurcation point. In order to solve this, we allow the system parameter $\gamma$ to vary and we add a parameterizing equation to the MP-BVP (4.6) to obtain a fully determined system of equations. For the parameterizing equation, we use the pseudo-arclength equation [18]

$$
\beta(y, T, \gamma) := \omega_y \int_0^1 (\delta pr y(s))^T \Delta y(s) ds + \omega_T (\delta pr T) \Delta T + \omega_\gamma (\delta pr \gamma)^T \Delta \gamma = 0,
$$

where $(\delta pr y, \delta pr T, \delta pr \gamma)$ is the predictor direction and the scalars $\omega_y, \omega_T, \omega_\gamma$ are positive weights.

For the continuation method, the principles of operation are used as implemented in DDE-BIFTOOL [7]. This involves a secant predictor and for steplength control a constant factor greater than 1 is multiplied with the previous used steplength if the previous point was computed successfully. If the corrected point is rejected because its accuracy does not meet the requirements then a new point is predicted with linear interpolation between the last two successfully computed points. When the corrections of this interpolated point succeeds, it is inserted in the branch array on the correct position, that is between the last successfully computed points. If the correction of this point fails again, the last successfully computed point is rejected to avoid a possible branch switch and the interpolation procedure is repeated.

5.2 Numerical problems

After implementation of the algorithm it was noticed that the solution did not converge to the branch after some successfully computed branch points. A reason for this could be the event function $h$.  


This function is namely not continuously differentiable since it is a function composed of two inequalities coupled via an and statement. In order to change this function in a continuously differentiable function, let us define the first condition as
\[ c_1(\theta, \dot{\theta}) := \theta \left( \dot{\theta} - a \theta \right), \] (5.2)
and the second condition as
\[ c_2(\theta, \dot{\theta}) := \theta^2 + c \dot{\theta}^2 - r^2. \] (5.3)
The old event function is then given as
\[ h(\theta, \dot{\theta}) = c_1 > 0 \text{ and } c_2 > 0, \] (5.4)
and we have seen that this function consist of two straight lines with a circle around the origin. So we want a function that switches between the two conditions in the neighborhood of the circle as a step. In order to do this we make use of a hyperbolic tangent
\[ h(\theta, \dot{\theta}) = \frac{\tanh (c_2/\varepsilon) + 1}{2} c_1 + \left(1 - \frac{\tanh (c_2/\varepsilon) + 1}{2}\right) c_2 > 0, \] (5.5)
where \( \varepsilon \) determines the degree of smoothing and must be chosen sufficiently small to ensure that a step between one and zero is approximated on the circle boundary. In this study we use \( \varepsilon = 1.0 \cdot 10^{-10} \) which makes that the hyperbolic tangent is one everywhere in the state space except inside the circle where it is zero.

Of course differences in controller active regions between the old and new event function can be expected, and therefore they are both shown in Figure 5.1. As can be seen, there is almost no difference and therefore we do not expect a large change in system dynamics between the old and new event function.

The new function can be differentiated with respect to \( \theta \) as
\[ \frac{dh(\theta, \dot{\theta})}{d\theta} = \frac{\partial c_1}{\partial \theta} \tanh (c_2/\varepsilon) + 1 - \frac{\partial c_2}{\partial \theta} \tanh (c_2/\varepsilon) - 1 - \frac{\partial c_2}{\partial \theta} \frac{\tanh (c_2/\varepsilon)^2 - 1}{2\varepsilon} \]
\[ + \frac{\partial c_2}{\partial \theta} \frac{c_1 \tanh (c_2/\varepsilon)^2 - 1}{2\varepsilon}, \] (5.6)
and with respect to \( \dot{\theta} \) as
\[ \frac{dh(\theta, \dot{\theta})}{d\dot{\theta}} = \frac{\partial c_1}{\partial \dot{\theta}} \tanh (c_2/\varepsilon) + 1 - \frac{\partial c_2}{\partial \dot{\theta}} \tanh (c_2/\varepsilon) - 1 - \frac{\partial c_2}{\partial \dot{\theta}} \frac{\tanh (c_2/\varepsilon)^2 - 1}{2\varepsilon} \]
\[ + \frac{\partial c_2}{\partial \dot{\theta}} \frac{c_1 \tanh (c_2/\varepsilon)^2 - 1}{2\varepsilon}, \] (5.7)
where the partial derivatives are the same as with the old condition.

A problem that we encountered were the derivative files with respect to state and parameters variables because the symbolic function file tool from Matlab does not allow to include switch statements. This problem is solved by pre-multiplying the derivative vector with Lagrange basis polynomials. Since the Lagrange basis polynomials have the properties of a Kronecker delta, it allows us to compute the derivative of the system with respect to one of the 19 parameters in the same file.
Another problem is the predictor. A secant predictor is used, which is a linear extrapolation of the last two branch points. It uses therefore the difference between the last two successfully computed branch points. However, as we have seen in Chapter 2, different types of periodic solutions are possible. This predictor itself is therefore not suitable to compute the whole branch since the mode of the orbit changes as a bifurcation point is passed. The paper of David Barton [2] does not describe this problem since there only multiple period orbits occur. A solution to this problem might be to compute the stability along the branch and restart the algorithm at points where a discontinuous bifurcation occurs. This can be noticed by a sudden jump in eigenvalues of the Monodromy matrix [16]. This is, however, not yet implemented in the algorithm.

When using the method we noticed that frequently problems occur with the delay sections. This means that collocation points move outside the mesh after updates of the Newton method. Now it is solved by remeshing the interval but this is not so reliable since extrapolation is used. It might therefore be worthwhile to investigate whether a reduced linearized system without the delay sections can be made especially since all the jump functions equal identity. Now the delay sections are updated twice, on the end of the $i$th interval and on the start on the $(i+1)$th interval. Another remark on the delay section is that they have a fixed number of intervals and this is the same for the time section. The ratio of intervals between the delay section and time section highly influences the convergence rate of the algorithm. This problem will automatically disappear when an equidistant mesh is used without delay sections. Furthermore, it might be worthwhile to investigate whether a condensation procedure for the linearized system can be implemented as used in several collocation schemes [7,19].

Figure 5.1: Difference in controller active regions (white area) in the phase plane between the old (non continuously differentiable) function (left) and the new (continuously differentiable) function (right).
5.3 Numerical results

The aforementioned problems make a complete bifurcation analysis impossible. Therefore, we present only one branch where the intermittent stiffness coefficient $P$ is varied. This branch is depicted in Figure 5.2. A selection of other solutions that are possible with corresponding stiffness value are shown in Figure 5.3.

![Figure 5.2: Branch of periodic solutions.](image)

![Figure 5.3: A selection of types of periodic solutions that can occur while continuing a branch.](image)
6

Conclusions and recommendations

6.1 Conclusions

In this report, the dynamics of a human being standing quietly on a rigid and on a compliant surface are examined. The rigid surface model is an already existing inverted pendulum model, which rotates around the ankle joint. The compliant surface model is an extension of this model, which is realized by adding springs and dampers under the ankle. Consequently, two extra degrees of freedom are added, i.e. a rotation around the toe and vertical displacement of the ankle. The equations of motion for the latter model are derived using Lagrange’s equations of motion.

The ankle controller is constructed by adopting an event-driven intermittent control strategy, in which control is altered when state-dependent thresholds are passed. Due to the switching nature of this controller, both systems are discontinuous and are described by PWS DDEs.

In order to gain insight in the dynamics, it is necessary to simulate the system. Normal ODE solvers could not be applied for these kind of systems. Therefore, a delay solver with event-detection was used. The simulation results have shown that both models converge to almost the same periodic solution after initial dynamic behavior. Furthermore, in both models it was shown that the number of boundary crossings during a periodic solution can change when a parameter is varied. However, one remark has to be made since the COP is not calculated and when the COP becomes near the edge of foot support, a different stabilizing approach must be used because otherwise the model will tip over. Other strategies are ‘hip strategy’ where the hip muscles are used, ‘combined strategy’ where both ankle and hip muscles are used, or making a step. In all of these cases, the used models are not valid.

To compute periodic solutions of both systems, no on hand numerical tools were available. Therefore, a mathematical background study was performed on delay systems specifically concentrated on periodic solutions and numerical tools to obtain these periodic solutions. It was concluded, the collocation method is the most robust method to compute periodic solutions of delay systems. This method was chosen to be modified such that it can be used for PWS delay systems. A drawback of this approach was that the method is quite complex and only one concisely written reference [2] was available. This explains the significant number of problems which where encountered implementing the algorithm in Matlab.

To validate the implementation, test models were made. Based on these test the implementation appears to be correct as the results are (almost) the same as the results presented in [2]. Subsequently, the method was used to compute periodic solutions of both models. These were compared with numerical integration solutions and identical results were obtained.

In order to perform a bifurcation analysis, pseudo-arclength continuation has been applied. A problem that occurs with this method is that a part of the branch successfully can be computed until the corrector steps do not converge. Two reasons can be given for this. One reason is that the used predictor, a secant, is not suitable to compute the whole branch, since different types of periodic solutions along the branch exist and therefore the type of the orbit might change as a parameter is varied. This change in mode cannot be predicted by the secant predictor because it uses
the difference between the last two successfully computed points. A solution may be to restart the continuation procedure each time an orbit mode change occurs. But to know where this occurs, we need stability information because the eigenvalues of the Monodromy matrix jump at a discontinuous bifurcation. The second reason is the remesh of the intervals when collocation points move outside the mesh after updates of the Newton method. The latter uses a extrapolation procedure and might therefore be unreliable. To solve this, a reduced linearized system without delay sections may be preferred. Whether this approach is possible for this system is not known.

Overall we can conclude that the collocation method for calculation of periodic solutions of PWS delay systems has proven to be useful but it has some flaws, which needs to be solved, before it can be used to perform a continuation/bifurcation analysis. This might be the reason why no one has adopted this method since 2009 when it became available.

6.2 Recommendations

A Recommendation regarding the model is

- Collect data from a clinical test with known foam properties and perform with the data a model validation.

Recommendations regarding the collocation and the continuation method for PWS delay systems are

- Improve the algorithms by using a simpler model with known dynamics before applying them to the models used in this report. This still requires a significant amount of time since it involves some difficult problems and is maybe more suitable for a graduation thesis.

- Investigate and implement a condensation procedure, which eliminate the delay sections from the linearized system used in the collocation method.

When this approach is not followed, the following recommendations may be useful

- Convert the DDEs to ODEs via the method of steps [4] and use an already proven method for PWS systems such as [17]. However, to obtain an accurate approximation a high-dimensional set of ODEs is needed, and this leads to expensive numerical procedures.

- Convert the PWS delay system to normal delay system by smoothing the impact function with for instance a hyperbolic tangent function. Again the bifurcation can be performed with an already proven tool namely DDE-BIFTOOL [7]. This tool contains a lot of features regarding delay systems and can therefore be of great support in analysing the systems behavior. However, as mentioned by [2], some points might be missed by smoothing the system and the resulting DDEs may become very stiff, which again may lead to numerical problems.
References


Time domain simulations

This chapter shows the time domain simulation results used in Chapter 2.
A.1 Tilt angle

Figures A.1 - A.8 shows the tilt angle $\theta$ for the four different types of underground stiffness.

![Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.1$.](image)

Figure A.1: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.1$. 
Figure A.2: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.2$. 
Figure A.3: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -1.4$, and $\tau = 0.1$. 
Figure A.4: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.2$. 
Figure A.5: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -0.4$, and $\tau = 0.1$. 
Figure A.6: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -0.4$, and $\tau = 0.2$. 

(a) rigid  
(b) soft  
(c) medium  
(d) firm
Figure A.7: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -1.4$, and $\tau = 0.1$. 
A.1 Tilt angle

Figure A.8: Simulation results for the tilt angle $\theta$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -1.4$, and $\tau = 0.2$. 
A.2 Ankle angle

Figures A.9 - A.16 shows the toe angle $\phi$ for the three different types of underground stiffness.

Figure A.9: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.1$. 
Figure A.10: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.2$.

Figure A.11: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -1.4$, and $\tau = 0.1$.

Figure A.12: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -1.4$, and $\tau = 0.2$. 
Figure A.13: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 235, 44$, $a = -0.4$, and $\tau = 0.1$.

Figure A.14: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 235, 44$, $a = -0.4$, and $\tau = 0.2$.

Figure A.15: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 235, 44$, $a = -1.4$, and $\tau = 0.1$. 

Figure A.16: Simulation results for the ankle angle $\phi$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -1.4$, and $\tau = 0.2$. 
A.3 Toe height

Figures A.17 - A.24 show the toe height $y$ for the three different types of underground stiffness.

![Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.1$.](image)

Figure A.17: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.1$. 
Figure A.18: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -0.4$, and $\tau = 0.2$.

Figure A.19: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -1.4$, and $\tau = 0.1$. 
Figure A.20: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 147.15$, $a = -1.4$, and $\tau = 0.2$.

Figure A.21: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 235.44$, $a = -0.4$, and $\tau = 0.1$. 
Figure A.22: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 235, 44, a = -0.4$, and $\tau = 0.2$.

Figure A.23: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 235, 44, a = -1.4$, and $\tau = 0.1$. 
Figure A.24: Simulation results for the toe height $y$ on different kinds of underground stiffness with parameters $P = 235, 44$, $a = -1.4$, and $\tau = 0.2$. 
Compliant surface model

Medium foam stiffness

The following results are from the compliant surface model with the medium stiffness underground and are depicted in Figures B.1, B.2 and B.3.

Figure B.1: Time series (left) and the phase-plane representation (right) of the converged solution $\theta$. The first derivative of the solution is shown in gray.

Figure B.2: Time series (left) and the phase-plane representation (right) of the converged solution $\phi$. The first derivative of the solution is shown in gray.
Firm foam stiffness

Figures B.4, B.5 and B.6 shows the results for the firm underground stiffness used in Chapter 4.
Figure B.5: Time series (left) and the phase-plane representation (right) of the converged solution $\phi$. The first derivative of the solution is shown in gray.

Figure B.6: Time series (left) and the phase-plane representation (right) of the converged solution $y$. The first derivative of the solution is shown in gray.
Collocation for PWS delay systems

C.1 Collocation for delay systems

In order to explain collocation, consider the simplest DDE
\[
\dot{x} = f(x(t), x(t - \tau)).
\] (C.1)

Then introduce a mesh with \( L + 1 \) mesh points \( \{0 = t_0 < t_1 < \cdots < t_k = 1\} \). This mesh is periodically extended with \( \ell \) points to obtain a mesh on \([-\tau/T, 1]\). Representation points in each interval \([t_i, t_{i+1}]\) are defined as
\[
t_{i + \frac{j}{d}} = t_i + \frac{j}{d}(t_{i+1} - t_i), \text{ for } j = 0, \ldots, d.
\]

Now we can approximate (C.1) on the mesh \([-\tau/T, 1]\) by a set of polynomials
\[
u(t) = \sum_{j=0}^{d} u(t_{i + \frac{j}{d}}) \psi_{i,j}(t), \ t \in [t_i, t_{i+1}],
\] (C.2)
where \( \psi_{i,j} \) are Lagrange polynomials.

Figure C.1: Lagrange polynomials \( \psi_{i,j} \) of degree \( d = 5 \).

Since Lagrange polynomials are orthogonal at the representation points (this means only one basis polynomial equals one while the others are zero) the approximating polynomial \( u(t) \) is completely determined by the ‘coordinates’
\[
u_{i + \frac{j}{d}} := u(t_{i + \frac{j}{d}}), \ i = -\ell, \ldots, L - 1, \ j = 0, \ldots, d - 1,
\]
and \( u_L := u(t_L) \).

The unknown 'coordinates' are determined by substituting (C.2) in (C.1) and to ensure that the approximation \( u(t) \) satisfy the DDE at the collocation points

\[
c_{i,j} = t_i + c_j(t_{i+1} - t_i), \quad i = 0, \ldots, L - 1, \quad j = 1, \ldots, d.
\]

Normally, the scaled and shifted roots of the Legendre polynomial are used for this. But other points are also possible, for example the mesh points. However, this would increase the error at the mesh points between the collocation solution and the DDE.

The obtained system of nonlinear algebraic equation is solved with Newton iterations since we are unable to solve this directly. This delivers, however, only a small solution. In order to find periodic solution we have to add two conditions. For this, let us define the starting vector as

\[
u_s := [u_{-\ell}, \ldots, u_{i+\frac{d}{2}}, \ldots, u_0]^T,
\]

and final vector as

\[
u_f := [u_{L-\ell}, \ldots, u_{i+\frac{d}{2}}, \ldots, u_L]^T.
\]

With those two vectors, we can introduce the first condition. This condition ensures periodicity and requires therefore equality of both vectors \( u_s = u_f \). The second condition is a phase condition \( p(u) = 0 \). This condition is needed to remove translational invariance in time because every segment on the periodic solution is a solution.

The previous system of collocation equations is extended with those two equations and the solution newly obtained system is a periodic solution of (C.1).

### C.2 Collocation for PWS delay systems

Now the collocation method has been shown, it can be extended such that it can handle PWS delay systems. For this a PWS delay system is given in its simplest form as

\[
\dot{x}(t) = \begin{cases} \ f_1(x(t), x(t-\tau)), & \text{if } h(x(t), x(t-\tau)) \leq 0, \\ f_2(x(t), x(t-\tau)), & \text{if } h(x(t), x(t-\tau)) > 0. \end{cases}
\]

Difference with the previous section is that we now not have one solution but two. Namely, a solution for each vector field, where we have assumed that only one event occurs \( h(x(t), x(t-\tau)) = 0 \) during the periodic solution. (Otherwise we would have \( n \) solutions where \( n \) can be any factor of two).

With two solutions, there are also two start and two final vectors. In order to find periodic solutions, we have connect these vectors. This is done by the so called jump functions \( g \), which describes the instantaneous change of state upon impact with the switching manifold and are defined as

\[
g(u_{f_1}) = u_{s_2}, \\
g(u_{f_2}) = u_{s_1},
\]

or when the jump functions equal identity as in this report

\[
u_{f_1} = u_{s_2}, \\
u_{f_2} = u_{s_1}.
\]

Furthermore, we have to restrict that the final vectors of both solutions are on the switching manifold. For this, the event function \( h \) is used

\[
h(u_{f_1}) = 0, \\
h(u_{f_2}) = 0.
\]
With everything set, a large system of nonlinear algebraic equations remains. This system must be solved to obtain the collocation solution $u$ and period times $T_i$. For this, Newton iterations are used. The jacobian used in this method is shown in Figure C.2.

Figure C.2: Structure of linearized system used in the collocation method for PWS delay system for a periodic solution with one event and two different vector fields.

One can notice the two double staircase bands. One for each collocation interval. Furthermore, below the staircase bands, four sloped lines are noticeable. These are for periodicity conditions. And on the last two lines, which cannot be seen, the derivatives of the event functions are presented.