

# Thesis Title



Jonas Stolle

Bachelor Thesis  
May 2021

*Supervisors:*  
Moritz Geilinger  
Prof. Dr. Stelian Coros

**ETH**

Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich





# Abstract

This thesis addresses the development of a novel sample thesis. We analyze the requirements of a general template, as it can be used with the  $\text{\LaTeX}$  text processing system. (And so on...) The abstract should not exceed half a page in size!



# Contents

<b>List of Figures</b>	<b>v</b>
<b>List of Tables</b>	<b>vii</b>
<b>1. Introduction</b>	<b>1</b>
<b>2. Related Work</b>	<b>3</b>
2.1. Appearance Modeling . . . . .	3
2.1.1. Taxonomy . . . . .	3
2.1.2. Appearance Models . . . . .	3
2.2. Human Skin Rendering . . . . .	3
<b>3. Your Central Work</b>	<b>5</b>
3.1. Definitions and Problem Formulation . . . . .	5
3.1.1. Dynamical Systems . . . . .	5
3.1.2. Phase Space and Attractors . . . . .	7
3.1.3. Parameter space and Disturbance space . . . . .	7
3.1.4. robustness Measure . . . . .	8
3.2. Code Implementation . . . . .	9
3.2.1. Framework Overview . . . . .	9
3.2.2. Solver . . . . .	9
3.2.3. Detecting Attractors . . . . .	9
3.2.4. "Evaluating" Convergence . . . . .	9
3.2.5. minRad algorithm . . . . .	9
3.2.6. boundarys PS DS . . . . .	10
3.2.7. Multithreading . . . . .	10

*Contents*

3.2.8. case specific adaptations . . . . .	10
3.2.9. Optimization and Complexity reduction . . . . .	10
3.3. Results . . . . .	10
<b>4. Conclusion and Outlook</b>	<b>11</b>
<b>A. Information For The Few (Appendix)</b>	<b>13</b>
A.1. Foo Bar Baz . . . . .	13
A.2. Barontes . . . . .	13
A.3. A Long Table with Booktabs . . . . .	13

## List of Figures

## *List of Figures*



# List of Tables

A.1. Wordlist . . . . . 13

## *List of Tables*

# 1

## Introduction

## *1. Introduction*

# 2

## **Related Work**

Sample references are [?] and [?].

### **2.1. Appearance Modeling**

#### **2.1.1. Taxonomy**

#### **2.1.2. Appearance Models**

### **2.2. Human Skin Rendering**

2. *Related Work*

# 3

## Your Central Work

### 3.1. Definitions and Problem Formulation

In this section describes the relevant physical definitions used and referred to in the later code implementation.

Examples will be given in terms of a legged quadruped robot as most of the testing of the framework was done with that.

#### 3.1.1. Dynamical Systems

Dynamical systems are distinguished by an evolution of their state  $\mathbf{x}(t)$  through time. This evolution is fully described by a set of ordinary differential equations of the form  $\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t), t)$  which simplifies to  $\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t))$  under the assumption that the dynamical system is autonomous, i.e. not explicitly dependent on time.

When trying to solve for the explicit solution  $\mathbf{x}(t)$  an initial value at time 0  $\mathbf{x}_0 = \mathbf{x}(t_0)$  is needed. For simplicity we set  $t_0 = 0$ . With this the initial value problem IVP can be formulated:

$$\text{find } \mathbf{x}(t) \tag{3.1}$$

$$\text{s.t. } \dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t)) \tag{3.2}$$

$$\text{and } \mathbf{x}(0) = \mathbf{x}_0 \tag{3.3}$$

Its elements are a concatenation of coordinates describing the instantaneous constellation of a system and their temporal derivatives at a time  $t$ . As explicit expressions of  $\mathbf{x}(t)$  can rarely be found analytically, it is

### 3. Your Central Work

Don't explain why things are the way they are, just state them.

Its elements are coordinates in a chosen set of coordinates. For convenience we choose the so called generalized coordinates  $\mathbf{q}(t) = \begin{pmatrix} q_1(t) & q_2(t) & \dots & q_n(t) \end{pmatrix}^T$ , which describe the system configuration with a minimal amount of coordinates, the number of which coincided with the degrees of freedom of the system. The individual  $q$  might represent angles, positions along a specific direction or even along a constrained curve.

to acquire the equations of motion one needs to solve a  $2n$  dimensional system of differential equations of the general form

$$\begin{aligned}\dot{\mathbf{q}}(t) &= \mathbf{f}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) \\ \ddot{\mathbf{q}}(t) &= \mathbf{g}(\mathbf{q}(t), \dot{\mathbf{q}}(t))\end{aligned}$$

with  $\mathbf{f}$  and  $\mathbf{g}$  being some general nonlinear functions under the assumption that the system is autonomous.  $\mathbf{q}(t)$  describes the evolution through time.  $\mathbf{q}(t)$  can only be analytically found in a few cases and quickly breaks down when contact forces and other nonlinearities come into play.

$\mathbf{q}$  and  $\dot{\mathbf{q}}$  are sufficient to describe the system as they are precisely the initial conditions for solving for the trajectory in future time.

This is something for the appendix It is often much simpler to formulate the dynamics in terms of differential equations, which in the case of mechanical systems result in ordinary differential equations with the first and second time derivative of  $\mathbf{q}$ .  $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})$   $\ddot{\mathbf{q}} = \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}})$

order reduction:

$$\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}] \Rightarrow \dot{\mathbf{x}} = [\dot{\mathbf{q}}, \ddot{\mathbf{q}}] = \mathbf{F}(\mathbf{x}) = [\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{x})]$$

$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$

The trajectory of a system is the evolution of the state  $\mathbf{x}$  through time starting from an initial state at an initial time  $\mathbf{x}_0 = \mathbf{x}(t_0)$ . Explicit for trajectories  $\mathbf{x}(t) = \mathbf{F}(t)$  are generally hard to find which is why system dynamics are usually described in the form of an initial value problem using systems of differential equations  $\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t), t)$ .  $\dot{\mathbf{q}}(t) = \begin{pmatrix} \dot{q}_1(t) & \dot{q}_2(t) & \dots & \dot{q}_n(t) \end{pmatrix}^T$  is denoted as the vector of generalized velocities.

For mechanical systems the

For ease of notation we define the state  $\mathbf{x}(t)$  of a system by the concatenation of the generalized coordinates and velocities.

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{q}(t) & \dot{\mathbf{q}}(t) \end{pmatrix}^T$$

the solution of  $\mathbf{x}$  may depend on an arbitrary order of time derivatives, however it can always be reduced to a first order system of ode's.

we look at dynamical systems of the form of ivps

Here we assume that the differential equations do not explicitly depend on time and are therefore autonomous  $\mathbf{F}(\mathbf{x}(t), t) = \mathbf{F}(\mathbf{x}(t))$ .



define state and trajectory first.

Explic The dynamic of a system can often be described by a system on differential equations, The dynamical systems we dealt were autonomous and nonlinear in nature. Autonomous as we did not state, derivatives, trajectory

state ivp generalized coordinates and their derivatives trajectories

#### 3.1.2. Phase Space and Attractors

The phase space is a particular instance of state spaces, restricted to the set of generalized coordinates  $q_i$  and their corresponding generalized velocities  $\dot{q}_i$ , sometimes in form of generalized momenta (via scaling by mass or inertia). It encompasses all possible initial conditions of a mechanical system and solutions of the ivp result in trajectories through the phase space, starting at

Starting with a system and an initial condition that is part of an attractor

Attractors are sets of states in the phase space In a general nonlinear dynamical system, the evolution of the state can be divided into two main events using the concept of attractors. Attractors are sets of states toward which the system trajectories will converge or diverge from. If the set only consists of one state it is called a fixed point, else a periodic orbit where the trajectory will visit all states of the set periodically if given enough time. We propose methods for dealing with both cases. The set of initial conditions in the phase space for which the resulting trajectories converge towards the attractor is called the basin of attraction. If an initial condition does not lie in this set, it's trajectory diverges from the attractor. Formally, a trajectory converges to an attractor  $A$  if  $\lim_{t \rightarrow \infty} \mathbf{x}(t) \in A$

When referring to a system being robust to a disturbance, it means that it's trajectory in the phase space will return to the specified attractor under the influence of that particular disturbance.

#### 3.1.3. Parameter space and Disturbance space

separate from "dynamics", focus on

dimension disturbance space equals  $d$

In reference they did it with the phase space

We actively separate phase space and disturbance space for the concept to be more versatile

The disturbance space is defined as the set containing all combinations of disturbances, against which the robustness of the system is evaluated. The choice of disturbances in "REFERENCE" are the initial conditions in the phase space (reference to the section in related work or copy that here) as they can be interpreted as the direct result of external forces. In general however, the disturbances may be chosen arbitrarily and the resulting robustness measure will quantify robustness against these. Examples of this were initial tilting of a quadruped over pitch and roll axes and oscillations with the disturbance space spanning the amplitudes and frequencies.

### 3. Your Central Work

Tuple combinations are favoured because of their ease of visualization and mild computational power requirements.

"REFERENCE" chose the DS to coincide with the phase space, as elements of it can be directly interpreted as the result of external forces on the system. This choice is not always feasible nor necessary, as will be shown in the following sections.

It may coincide with the phase space as implemented in "REFERENCE", in which case disturbances can be interpreted as the effects of external forces acting on the system. This moves the system onto a different trajectory, however it will be shown in the following sections, that this is not always feasible nor necessary. Especially in systems with a large number of degrees of freedom  $n$ , the resulting disturbance space is  $2n$  dimensional.

case DS = phase space:  $d$

dimension parameter space =  $p$  The two main variable spaces of relevance move are the parameter space (PS), where each dimension represents the value of a chosen system parameter. The goal of finding a robustness measure can be thought of as  $RM : \mathbb{R}^p \rightarrow \mathbb{R}$ , i.e. a mapping from the  $p$ -dimensional parameter space to a scalar value. (really it also depends on the disturbance space) Finding the best system parameters with respect to the resulting robustness can be formulated as an optimization problem ...

state space  $v$  phase space

#### 3.1.4. robustness Measure

The basic idea of robustness

First explain the basics of the REF implementation Then expand to general disturbance spaces with a pointer to the meaning of DS = phase space

The goal aim of the robustness measure is to quantify the robustness of a system with a particular parameter constellation with respect to a set of disturbances. As in "REFERENCE", the basic idea is that the more disturbances the system can endure, i.e. it converges to the attractor, the more robust it is. Discretizing the DS and simply counting the total number of disturbances the system is robust to is impractical because of the possibly fractal nature of the boundary of the basin of attraction (ref) and the fact that the number of evaluations is  $O((1/h)^d)$ , meaning that higher resolution and a larger DS will drastically increase computational time. *Torem*

In other words, any combination of disturbances which lies on the surface of the hypersphere with the minimal radius in the DS will result in a trajectory that converges to the attractor in the phase space.

Taking sampling initial conditions from the phase space as disturbances gives a nice physical interpretation and in "REF" this was denoted as general robustness (or was it?), however as a counterexample, periodic or constant disturbances are clearly not covered by this. This is why we worked with general disturbance spaces that can be chosen to work for the particular application at hand. Note that the trajectory will still lie in the phase space and convergence evaluation

## 3.2. Code Implementation

### 3.2.1. Framework Overview

### 3.2.2. Solver

Starting with a system of ODE's and as corresponding set of initial states, the first step towards computing robustness measure is finding the resulting trajectories. This is achieved by numerically integrating the differential equations over small timesteps until either convergence is detected or the  $t_{max}$  is reached. With the chosen solver it is imperative to keep this timestep constant when comparing different robustness measures. The time step has a direct influence on the solvers accuracy and by that onto the system dynamics. It is advised to find a value that is a sufficient compromise between accuracy and computational time and keeping this fixed from there on.

DDE provides  $q$ ,  $\dot{q}$  and  $\ddot{q}$  at every timestep when computing the trajectories.

As initial conditions for the IVP

### 3.2.3. Detecting Attractors

In order to evaluate whether a trajectory converges

General idea with similarity of states to attractor state

Fixed points

fixed points are defined states where  $\dot{x}(t) = 0 \quad \forall t > t_0$ , i.e.  $q(t) = q_{fp}, \dot{q}(t) = \ddot{q}(t) \forall t > t_0$

Periodic Orbits Poincare => issues

Issue with wrong attractor

Simplification

### 3.2.4. "Evaluating" Convergence

Clearly it is impractical to let  $t$  go to infinity, which is why some  $t_{n, max}$  need to be defined at which the simulation

### 3.2.5. minRad algorithm

maybe not go into too much detail on how it works (look at ref)

tuning

visualizations

### 3. *Your Central Work*

#### **3.2.6. boundarys PS DS**

elephant in the room changing even the unit changes the scaling and therefore the robustness value.

explain and give examples on how boundaries were chosen.

was noted in REF that it can be an ellipse as well, but they didn't know how to choose it's parameters. We do this by analysis and finding reasonable boundaries in the DS.

order of complexity depending on PS and DS

#### **3.2.7. Multithreading**

computational time can be reduced by pseudocode

#### **3.2.8. case specific adaptations**

applyParameters

simAndEval

#### **3.2.9. Optimization and Complexity reduction**

cmaes complexity reductions should probably be noted where they are implemented (i.e. not in this section)

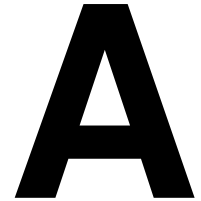
### **3.3. Results**

optimization of robustness examples

# 4

## **Conclusion and Outlook**

#### 4. *Conclusion and Outlook*



## Information For The Few (Appendix)

### A.1. Foo Bar Baz

### A.2. Barontes

### A.3. A Long Table with Booktabs

*Table A.1.: A sample list of words.*

ID	Word	Word Length	WD	ETL	PTL	WDplus
1	Eis	3	4	0.42	1.83	0.19
2	Mai	3	5	0.49	1.92	0.19
3	Art	3	5	0.27	1.67	0.14
4	Uhr	3	5	0.57	1.87	0.36
5	Rat	3	5	0.36	1.71	0.14
6	weit	4	6	0.21	1.65	0.25
7	eins	4	6	0.38	1.79	0.26
8	Wort	4	6	0.30	1.62	0.20
9	Wolf	4	6	0.18	1.54	0.19
10	Wald	4	6	0.31	1.63	0.19
11	Amt	3	6	0.30	1.67	0.14

continued on next page

## A. Information For The Few (Appendix)

**Table A.1.: (Continued)**

ID	Word	Word Length	WD	ETL	PTL	WDplus
12	Wahl	4	7	0.36	1.77	0.42
13	Volk	4	7	0.45	1.81	0.20
14	Ziel	4	7	0.48	1.78	0.42
15	vier	4	7	0.38	1.81	0.42
16	Kreis	5	7	0.26	1.62	0.33
17	Preis	5	7	0.28	1.51	0.33
18	Re-de	4	7	0.22	1.56	0.33
19	Saal	4	7	0.75	2.10	0.43
20	voll	4	7	0.48	1.82	0.24
21	weiss	5	7	0.21	1.59	0.36
22	-ger	5	7	1.16	2.69	0.59
23	bald	4	7	0.18	1.56	0.19
24	hier	4	7	0.40	1.70	0.43
25	neun	4	7	0.17	1.52	0.26
26	sehr	4	7	0.36	1.85	0.43
27	Jahr	4	7	0.50	1.82	0.43
28	Gold	4	7	0.04	1.35	0.20
29	Ter	5	8	0.15	1.39	0.59
30	Tei-le	5	8	0.30	1.71	0.46
31	Na-tur	5	8	0.18	1.59	0.41
32	Feu-er	5	8	0.30	1.71	0.45
33	Rol-le	5	8	0.15	1.46	0.45
34	Rock	4	8	0.29	1.68	0.25
35	Spass	5	8	0.28	1.64	0.32
36	Gte	5	8	0.49	1.75	0.66
37	En-de	4	8	0.36	1.72	0.33
38	Kunst	5	8	0.26	1.59	0.35
39	Li-nie	5	8	0.45	1.88	0.63
40	Bme	5	8	0.48	1.92	0.45
41	Bh-ne	5	9	0.94	2.48	0.62
42	Bahn	4	9	0.21	1.62	0.42
43	Br-ger	6	9	0.38	1.70	0.65
44	Druck	5	9	0.60	2.03	0.31
45	zehn	4	9	0.41	1.84	0.42
46	Va-ter	5	9	0.36	1.78	0.40
47	Angst	5	9	0.29	1.56	0.35
48	lei-der	6	9	0.13	1.47	0.52
49	hfig	6	9	0.82	2.31	0.52
50	le-ben	5	9	0.38	1.85	0.40
51	aus-ser	6	9	1.20	2.26	0.57
52	be-vor	5	9	1.28	2.75	0.39
continued on next page						



**Table A.1.:** (Continued)

ID	Word	Word Length	WD	ETL	PTL	WDplus
53	Kai-ser	6	9	0.92	2.37	0.53
54	Markt	5	9	0.23	1.58	0.28
55	Os-ten	5	9	0.21	1.54	0.48
56	Krieg	5	9	0.33	1.67	0.50
57	Mann	4	9	0.31	1.47	0.25
58	Hal-le	5	9	0.24	1.65	0.45
59	heu-te	5	9	0.44	1.87	0.46
60	in-nen	5	10	0.36	1.80	0.45
61	Na-men	5	10	0.28	1.72	0.41
62	jetzt	5	10	0.70	2.07	0.32
63	kei-ner	6	10	0.28	1.62	0.53
64	Schu-le	6	10	1.02	2.12	0.48
65	Ar-beit	6	10	0.34	1.70	0.52
66	An-teil	6	10	0.27	1.63	0.53
67	di-rekt	6	10	0.67	2.04	0.47
68	vor-her	6	10	0.78	2.25	0.47
69	wol-len	6	10	0.44	1.85	0.51
70	Kampf	5	10	0.70	1.96	0.27
71	dern	6	10	1.18	2.62	0.65
72	lau-fen	6	10	0.21	1.64	0.52
73	Eu-ro-pa	6	10	0.23	1.53	0.66
74	statt	5	10	1.61	2.86	0.39
75	Wes-ten	6	10	0.29	1.60	0.54

*A. Information For The Few (Appendix)*