Chair of Automatic Control

Technical University of Munich

Anja Götz

Structure preserving simulation for a manipulator with multi-articular elastic couplings

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science

at the TUM School of Engineering and Design

Advisor(s): PD Dr.-Ing. habil. Paul Kotyczka

Submitted by: Anja Götz

Submission date: 26 July 2024 in Garching bei München

Erklärung

Ich versichere hiermit, dass ich die von mir eingereichte Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Garching bei München, den 26. Juli 2024 _____ (Anja Götz)

Freiwillige ergänzende Erklärungen

☐ Mit der zeitlich unbefristeten Aufbewahrung meiner Arbeit in der Lehrstuhlbibliothek erkläre ich mich einverstanden.

☐ Ich stelle die Software, die ich im Rahmen dieser Arbeit entwickelt habe, dem Lehrstuhl für Regelungstechnik unter den Bedingungen der 3-Klausel-BSD-Lizenz zur Verfügung. Ich behalte dabei die sämtlichen Urheber- sowie Nutzungsrechte und werde als Autor namentlich genannt.

Garching bei München, den 26. Juli 2024 _____ (Anja Götz)

Chair of Automatic Control (Prof. Dr.-Ing. habil. Boris Lohmann)
Technical University of Munich
Boltzmannstraße 15
85748 Garching bei München
Germany

Lehrstuhl für Regelungstechnik (Prof. Dr.-Ing. habil. Boris Lohmann)
Technische Universität München
Boltzmannstraße 15
85748 Garching bei München
Deutschland

Abstract

Due to the increasing demand for product individualization in recent years, the focus in automation technology is increasingly shifting to flexibly applicable robots, as these can also perform more complex tasks autonomously, such as loading a machine.
Instead of traditional industrial robots, which – due to the combination of high payload and fast movements – are put in a stationary position, collaborative robots are increasingly being used together with devices providing autonomous movement. This provides a more flexibel use of one robot.
An extension of this idea are legged robots, which can also move autonomously in a less familiar environment and uneven terrain [2, § 1, p. 2017].
Thereby large force peaks can occur. Inspired by the biomechanics of the human body the dynamics of the system can be improved by adding elastic elements connecting the links. This results in complex couplings, the exact effects of which are the subject of current research [22, § 1, p. 2].
This gives rise to the need for efficient simulation methods, a central component of which is the numerical solution of the highly non-linear forward dynamics.
This thesis first describes two approaches from the literature that focus on the conserved quantities of Hamiltonian systems, which are energy conservation and symplecticness [13, § 1.1.1, p. 1]. These can be used to identify non-physical behavior of the numerical solution, which is of great relevance for simulation-based control designs.
Subsequently, the presented time integration methods are implemented for a planar, serial manipulator with additional elastic couplings, whereby the structured derivation of the equations of motion is treated.
Finally, the simulation is studied based on its behavior regarding the conserved properties of the Hamiltonian system, as well as its computational performance to determine an efficient simulation framework for such problem statements.

Kurzfassung

Durch den steigenden Anspruch an Produktindividualisierung der letzten Jahre, verlagert sich der Fokus in der Automatisierungstechnik zunehmen auf flexibel einsetzbare Roboter, da diese auch komplexere Aufgaben – wie bspw. das Bestücken einer Maschine – autonom bewerkstelligen können.
Dafür werden anstelle von klassischen Industrierobotern, welche aufgrund der Kombination von hoher Nutzlast und schneller Bewegungen stationär positioniert werden, zunehmen kollaborative Roboter in Verbindung mit Vorrichtungen zur autonomen Fortbewegung eingesetzt.
Eine Erweiterung dieser Idee sind sogenannte Laufroboter, welche sich auch in einem weniger gut bekannten Umfeld und auf unebenen Terrain autonom fortbewegen können [2, § 1, p. 2017].
Dabei treten u.a. große Kraftspitzen auf, wobei – inspiriert durch die Biomechanik des menschlichen Körpers – die Dynamik des Systems durch elastische Elemente verbessert werden soll. Dabei ergeben sich komplexe Kopplungen, deren genaue Auswirkungen auf die Dynamik Thema aktueller Forschung sind [22, § 1, p. 2].
Daraus leitet sich der Bedarf nach effizienten Simulationsmethoden ab, wobei einen zentralen Bestandteil dabei die numerische Lösung der hochgradig nicht-linearen Vorwärtsdynamik darstellt.
In dieser Arbeit werden zwei Ansätze aus der Literatur beschrieben, deren Fokus auf den Erhaltungsgrößen Hamiltonscher Systeme – die Energie und die symplektischen Struktur – liegt [13, § 1.1.1, p. 1]. Durch diese lässt sich unphysikalisches Verhalten der numerischen Lösung identifizieren. Dies ist von großer Relevanz für simulationsbasierte Regelungsentwürfe.
Anschließend werden die vorgestellten Zeitintegrationsverfahren für einen planaren, seriellen Manipulator mit elastischen Kopplungen implementiert, wobei die strukturierte Herleitung der Bewegungsgleichungen behandelt wird.
Anschließend wird die Simulation auf der Grundlage ihres Verhaltens hinsichtlich der Erhaltungseigenschaften eines Hamiltonschen Systems sowie ihrer Rechenleistung bewertet, um ein effizientes Simulationsframework für solche Problemstellungen zu bestimmen.

Contents

Abstract/Kurzfassung **v**

1 Introduction **1**

1.1 The forward dynamic problem 1

1.2 Time integration of the equations of motion 2

1.2.1 The flow map and numerical time integration 2

1.2.2 Numerical properties 2

2 Serial manipulator **3**

2.1 Geometry of the serial manipulator 3

2.2 Mono- and multi-articular springs 3

2.2.1 Background 3

2.2.2 Attachment of the springs to the serial manipulator 4

2.2.3 Spring energy 5

2.2.4 Implementation 6

3 An energy stable approach using non-minimal coordinates **9**

3.1 Hamiltonian systems 9

3.1.1 Structure of Hamiltonian systems 9

3.1.2 Conserved quantities 10

3.1.3 Fundamental properties of Hamiltonian systems 10

3.1.4 Example for symplectic time integration 11

3.2 Hamiltonian systems in classical mechanics 11

3.2.1 Equations of motion in Hamiltonian form 12

3.2.2 Equations of motion in Lagrangian form 12

3.3 Energy stable time integration 13

3.3.1 Energy conservation of a continuous system 13

3.3.2 Energy conservation of a discrete system 14

3.4 Implementation 15

3.4.1 Geometry and constraints . . . 15
3.4.2 Kinetic energy and mass matrix . . . 16
3.4.3 Potential energy . . . 17
3.4.4 Discrete equations of motion . . . 18

4 Variational integrators on Lie groups 19
4.1 Lie groups and Lie algebra . . . 19
4.2 Screw theory . . . 20
4.2.1 The exponential map . . . 20
4.2.2 Serial kinematics . . . 21
4.2.3 Velocity . . . 22
4.2.4 Manipulator Jacobian . . . 23
4.3 Variational integrators . . . 23
4.3.1 Background . . . 23
4.3.2 Obtaining the continuous Euler-Lagrange equations . . . 24
4.3.3 Discretization of the Euler-Lagrange equations . . . 24
4.4 Implementation . . . 25
4.4.1 Geometry . . . 25
4.4.2 Kinetic energy and mass matrix . . . 26
4.4.3 Potential energy . . . 27
4.4.4 Continuous equations of motion . . . 27
4.4.5 Discrete equations of motion . . . 29

5 Results and discussion 31
5.1 Methodology . . . 31
5.1.1 Simulation parameters . . . 31
5.1.2 Symplecticness . . . 31
5.1.3 Energy conservation . . . 32
5.1.4 Computational performance . . . 32
5.1.5 Influence of the time step . . . 33
5.1.6 Reference simulation . . . 33
5.2 Results . . . 34
5.2.1 Symplecticness . . . 34
5.2.2 Energy conservation under gravitational influence . . . 34
5.2.3 Energy conservation with elastic elements . . . 35
5.2.4 Computational performance . . . 38
5.3 Discussion . . . 39

5.3.1 Use of non-minimal coordinates 39
5.3.2 Energy conservation 40
5.3.3 Computational performance 40
5.4 Summary and outlook 40

List of Figures **43**

List of Tables **45**

References **49**

Chapter 1

Introduction

1.1 The forward dynamic problem

The goal for the first part of the thesis is to obtain the equations of motion (EOM) for a robot modeled by a rigid body system and subsequently carry out a time integration which then yields the actual trajectory.
This problem statement is often referred to as the forward dynamics of a system and can be used for the simulation of such, where the inputs are the control forces and the output is the motion carried out as a result [8, § 6, p. 101]. Such a simulation has a wide field of applications, including testing out new control strategies.
An important tool for constructing the EOM in a structured manner is the Lagrangian framework, which – in contrast to the Newtonian viewpoint – allows using non-inertial coordinates and their respective time derivatives for describing the state of a multi-body-system (MBS). Such systems can also be represented in Hamiltonian form, where instead of the coordinates time derivatives, the generalized momenta are used. The Lagrangian and the Hamiltonian representation are related via the Legendre transformation [18, § 3.1, p. 89,90].
In the the considered case of a chain structure connected by rotational joints, a natural choice for those so-called generalized coordinates $(\mathbf{q}, \dot{\mathbf{q}})$ are the joint angles and their time derivatives. As an arbitrary variation of a configuration $\mathbf{q}$ does not violate the constraints of the system, these are also minimal coordinates and thus no constraint forces appear in the resulting EOM [23, § 3.3, p. 45].
After restructuring the terms, the result is the following second order, non-linear ordinary differential equation (ODE):

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) + g(\mathbf{q}) = \boldsymbol{\tau}, \tag{1.1}$$

containing the symmetric, positive definite mass-matrix $M(\mathbf{q})$, the vector of Coriolis forces $C(\mathbf{q}, \dot{\mathbf{q}})$, the gravity forces $g(\mathbf{q})$ and the non-conservative torques acting on the joints $\boldsymbol{\tau}$, which typically consist of the motor torques and friction terms [23, § 3.3.1, p. 45, 46].
In robotics, this formulation (1.1) is predominantly used, due to its simple structure and absence of unknown constraint forces. However, this choice of coordinates introduces trigonometric terms into the EOM and thus makes it highly non-linear.

1.2 Time integration of the equations of motion

In general the EOM (1.1) cannot be solved analytically due to nonlinearities. Thus for calculating the complete forward dynamics problem one must carry out a numerical time integration. In this Section a brief overview of this concept and the associated numerical properties are given.

1.2.1 The flow map and numerical time integration

The solution of an initial value problem (IVP) of the form:

$$\dot{x} = f(x), \quad x(0) = x_0, \tag{1.2}$$

can be represented by a flow map, which maps an initial condition x_0 to its corresponding state $x(t)$ at time t [1, § 3, p. 79, 80].

$$\Psi_t : x_0 \to x(t) \tag{1.3}$$

For numerically integrating an IVP the ODE is discretized in time and thus the real solution is approximated by a discrete flow map. A common way for obtaining such a numerical scheme is using the fundamental theorem of calculus (1.4) and approximating the exact integral on the right hand side by numerical integration. This results in the group of one-step-methods [1, § 3, p. 80].

$$f(b) - f(a) = \int_a^b f'(x)dx \tag{1.4}$$

1.2.2 Numerical properties

When approximating the solution of an ODE, the error can be divided into two categories. The local error is the difference between the next value at t_{i+1} and the real solution of the IVP with the initial value beeing the result of the last step x_i. It can be approximated by comparing the corresponding Taylor expansions. On the other hand, the global error is the accumulation of local errors over the course of a complete trajectory [1, § 2.1.2, p. 40–42].
As we deal with Hamiltonian systems, one can look at the local and global effects of discretization on the two fundamental properties. Even though the conservation of energy and symplecticity do not guarantee a correct solution, these characteristics are essential for e.g. ensuring passivity when selecting a simulation based control strategy.
It can be shown, that generally conserving both of those properties – the constant Hamiltonian and the symplectic structure – is not possible when numerically integrating the ODE [13, § 1.1.3, p. 3]. Thus for finding an appropriate time integration scheme, the aim is to either preserve one of those properties exactly and keeping the error for the other one reasonably bounded.

Chapter 2

Serial manipulator

2.1 Geometry of the serial manipulator

Both approaches are implemented for the example of a planar kinematic chain with n links, connected with rotational joints, as depicted in figure 2.1.
The links have the mass m_i, the length L_i and the moment of inertia around the z-axis related to its center of mass (COM) J_i^{com}. The gravitational acceleration g is acting in the direction of negative y in inertial coordinates.
As a simplification for the further implementation of the simulation, we assume that the COM of the ith link is located on the connecting axis between the ith and the $(i+1)$th joint, at the distance l_i. However with simple modifications of the implementations this restriction can be avoided, which will be briefly shown in Section 3.4.1.
The COM positions of each link and the positions of the corresponding rotational axes are denoted by

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \qquad \text{and} \qquad \mathbf{X}_i = \begin{bmatrix} X_i \\ Y_i \end{bmatrix}. \tag{2.1}$$

The ith body coordinate systems origin is fixed at the $(i+1)$th rotational axis. Its x-axis coincides with the connecting axis to the $(i+1)$th joint and the z-axis is pointing in the direction of the rotational axis. The relative angle between between the ith link and its predecessor is denoted by Φ_i.
The normalized attitude vectors $\mathbf{q}_i$ are pointing in the direction of the ith x-axis.

2.2 Mono- and multi-articular springs

2.2.1 Background

In traditional industrial robotics the ability to carry out tasks in a short amount of time while maintaining high positional accuracy and stability is of great importance. Achieving this characteristics with an open chain manipulator is a difficult task, as this structure tends to be less stiff than parallel mechanisms. Thus the interface between actuator and load is constructed to be as stiff as possible [21, p. 399].
However in more recent developments there are multiple approaches to use passive elastic elements intentionally to modify the systems characteristics.

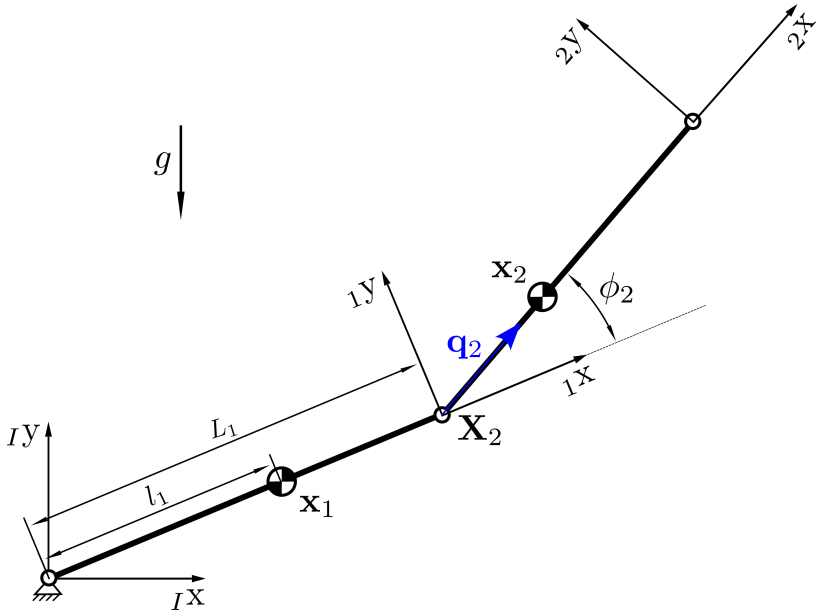


Figure 2.1: Geometry of the serial manipulator

This idea was first implemented in serial elastic actuators (SEA), where passive mechanical springs are built in between actuators and loads. Advantages of the SEA are a better tolerance for force peaks – often arising during unplanned interaction with the environment – and generally more stable force control [21, p. 399].
Inspired by the biomechanics of the human body, where movement is realized by the interaction between tendons, ligaments and muscles, there is another approach for changing the dynamics of a robot. Additionally to the traditional actuated rotational joints, passive or active elastic elements are attached to the system.
There are multiple areas of application, where the additional stored energy is useful. Especially for bipedal robots which carry out jumping motion, this can help to absorb heavy impact and stabilize the gait. Furthermore, the energy consumption of such a robot can be lowered, as an interchange of potential and kinetic energy can take place due to the elastic elements [3, p. 1217].
In this thesis only the latter case – attaching elastic elements to connect the links – is considered. Thereby one generally differentiates between mono- and multi-articular springs. While mono-articular springs only connect two neighbouring links, multi-articular springs span over multiple links [15, p. 1], which leads to a more complex force transmission between the bodies.
It is important to note, that in this thesis the concept of SEA is not considered, although the ideas described in this Section can arise together. We restrict our view to passive and linear mechanical springs spanned between the rigid links of the mechanism, which is described in the following Section 2.2.2.

2.2.2 Attachment of the springs to the serial manipulator

Implementing this idea for the serial manipulator, each link has fastening options for the springs at the end, which is depicted in Figure 2.2. The springs stiffness is denoted as $k_i > 0$ for each element with its current length s_i.
We assume, that the springs are under tension for all configurations, meaning the differ-

ence between slackened and current length Δs_i (2.2) is greater than zero for all times. This later avoids case distinctions, as springs only transmit force under tension.

$$\Delta s_i = s_i - s_{i,0} \tag{2.2}$$

To simplify further calculation, the fastening is located orthogonal to the connection axis of the rotational joints with the distance r_{i+1} at the end of the ith link. However, in general the spring could be attached at an arbitrary point on the body. Here the point for the attachment of the spring is denoted by $\mathbf{R}_{i+1}$.
It is worth noting, that in the considered planar case, the springs can be attached with an offset in z-direction, which avoids collisions between the elastic elements and the mechanism.

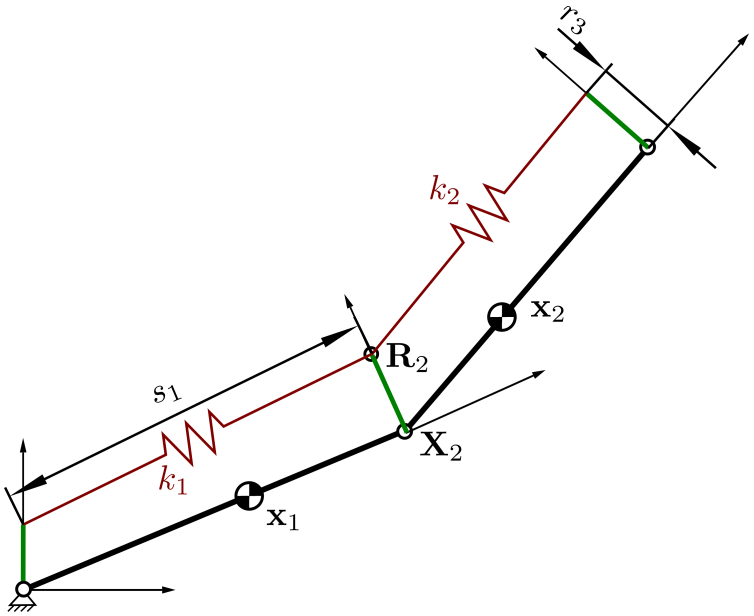


Figure 2.2: Attachment of the springs

In the case of mono-articular springs, each element is attached individually to the fastenings, which is the case depicted in Figure 2.2.
Generally, there exist a multitude of different constructions for multi-articular springs. One possibility for attaching a multi-articular spring to such a mechanism, is to keep the general structure but connect the springs at the fastenings in between the links which they are spanned over, see Figure 2.3.
Thereby the elastic element runs through a cutout in the fastening construction. When calculating the springs length, the exact geometry is neglected and approximated by a bend, as in the mono-articular case.

2.2.3 Spring energy

The springs stored energy S now appears as an additional term in the potential energy V of the system. Assuming linear behavior for the elastic elements, the change of length appears quadratically in S, which leads to a partial derivative of V depending on the configuration. This has to be considered when calculating the EOM and setting up the time integration methods.

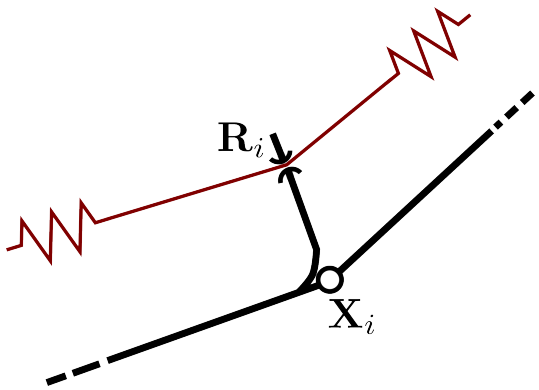


Figure 2.3: Fastening for multi-articular springs

In the mono-articular case there is a spring attached between each link and its predecessor. Thus each element individually leads to an additional potential term S_i:

$$S_{\text{mono-articular}} = \sum_{i=1}^{n} \underbrace{\frac{1}{2} k_i \left(\Delta s_i\right)^2}_{S_i} . \tag{2.3}$$

The springs stored energy in the previously described multi-articular case differs slightly, as instead of an individual contribution of each element, the sum of the length difference for each segment has to be taken into account, which yields:

$$S_{\text{multi-articular}} = \frac{1}{2} k \left(\sum_{i=1}^{n} \Delta s_i \right)^2 . \tag{2.4}$$

This is logical in the sense, that though following the same path as in the mono-articular case example depicted in Figure 2.2, the spring now can not transmit force to a fastening along its normal vector. Note that in the multi-articular case the length of the spring segments is still treated individually, however the stored energy has to be calculated collectively.
As in the later Sections 3.4 and 4.4 the derivatives of S w.r.t. the chosen coordinates are required, the general equations are prepared in the following Section 2.2.4. Therefore the configuration dependent length of the springs is calculated and then used for the gradient of S.

2.2.4 Implementation

The current length of the spring segment s_i can be calculated via the norm of the vector chain, depicted in Figure 2.4.

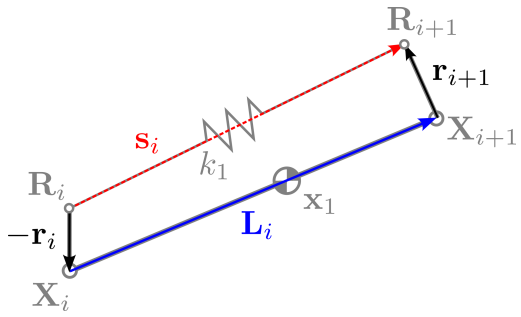


Figure 2.4: Vector chain for calculating the length of the springs

The calculation of the vector connecting two neighboring rotational joints $\mathbf{L}_i$ is straight forward:

$$\mathbf{L}_i = \mathbf{X}_{i+1} - \mathbf{X}_i. \tag{2.5}$$

Due to the orthogonal fastenings, the vector connecting $\mathbf{R}_i$ and $\mathbf{X}_i$ – the point where the fastening is rigidly attached to its respective link with index $i-1$ – can be calculated by forming the cross product of the z-axis and $\mathbf{L}_i$.
Putting together the vector chain and taking its euclidean norm, yields the general expression for s_i:

$$s_i = \left\| -\mathbf{r}_i + \mathbf{L}_i + \mathbf{r}_{i+1} \right\|. \tag{2.6}$$

The exact calculation depends on the choice of coordinates and is thus handled in the Sections 3.4 and 4.4.
As the third entry of the vector $\mathbf{s}_i$ is always zero the last line can be omitted when calculating the norm, which yields the spring length (2.7). Thereby the x and y component of the vector $\mathbf{s}_i$ are denoted by $s_{i,x}$ and $s_{i,y}$.

$$s_i = \sqrt{s_{i,x}^2 + s_{i,y}^2} \tag{2.7}$$

As only the length difference of the spring depends on the configuration the partial derivatives of the springs potential energy arise to:

$$\nabla_{\mathbf{q}} S_{\text{mono-articular}} = \frac{k_i}{s_i} \Delta s_i \left(s_{i,x} \nabla_{\mathbf{q}} s_{i,x} + s_{i,y} \nabla_{\mathbf{q}} s_{i,y} \right), \tag{2.8}$$

$$\nabla_{\mathbf{q}} S_{\text{multi-articular}} = k \left(\sum_{i=1}^{n} \Delta s_i \right) \sum_{i=1}^{n} \frac{1}{s_i} \left(s_{i,x} \nabla_{\mathbf{q}} s_{i,x} + s_{i,y} \nabla_{\mathbf{q}} s_{i,y} \right). \tag{2.9}$$

Chapter 3

An energy stable approach using non-minimal coordinates

As a fundamental property of physical systems, achieving energy conservation over longer simulation times is crucial in simulating multi body systems. In this chapter the idea of modeling the system in terms of the COM positions $\mathbf{x}_i$, which are non-minimal, inertial coordinates is used alongside an approach for energy stable time integration.
As x_i and y_i are linearly dependent, the choice of coordinates introduces holonomic constraints, which can be represented by a set of algebraic equations. Thus constraint forces are added into the EOM.
In the first Section 3.1, the class of Hamiltonian systems and their fundamental properties is introduced, followed by an overview over the necessary modifications of the canonical equations for mechanical systems in Section 3.2.
This provides the basics to then discretize the EOM for a mechanical system in an energy stable manner, see Section 3.3. Thereby the approach presented in the Paper [5] is used. Finally the implementation for the specific case of the serial manipulator presented in the previous Chapter 2 is described in Section 3.4.

3.1 Hamiltonian systems

3.1.1 Structure of Hamiltonian systems

An important class of ODEs are Hamiltonian systems, which can be written in a canonical structure (3.1), with the skew-symmetric matrix J showing the anti-symmetric nature of the ODE [19, § 3.1, p. 38]:

$$\begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{p}} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}}_{J} \begin{bmatrix} \nabla_{\mathbf{q}} H \\ \nabla_{\mathbf{p}} H \end{bmatrix}. \tag{3.1}$$

The phase space variables $\mathbf{q} \in \mathbb{R}^n$ and $\mathbf{p} \in \mathbb{R}^n$ completely describe the current state of the system and are subsequently summarized in the vector

$$\mathbf{u} = \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \end{bmatrix} \in \mathbb{R}^{2n}. \tag{3.2}$$

The function $H : \mathbb{R}^{2n} \to \mathbb{R}$ depending on $\mathbf{u}$ is the so-called Hamiltonian of the system [19, § 3.1, p. 38].
It is important to note, that (q_i, p_i) are conjugated value pairs [16, § 45, p. 145], which will become more clear when using this structure for the description of mechanical systems later.

3.1.2 Conserved quantities

A function $F(\mathbf{u}) : \mathbb{R}^{2n} \to \mathbb{R}$, which is constant along all solutions of (3.1), is called a first integral or a conserved quantity. The existence of such functions implies that the system evolves on lower dimensional manifolds described by these level curves and thus restricts the possible solution space [19, § 3.3, p. 44,45].
As F per definition does not change over time, calculating the time derivative of a first integral has to always yield zero. Hence when exploiting the chain rule, one can observe that the inner product of the gradient of F and $\dot{\mathbf{u}}$ vanishes:

$$\dot{F} = \langle \nabla_{\mathbf{u}} F, \dot{\mathbf{u}} \rangle = 0. \tag{3.3}$$

By inserting the Hamiltonian ODE for $\dot{\mathbf{u}}$ (3.1) into the previous equation (3.3), the necessary condition for a conserved quantity expressed by the function F is obtained [19, § 3.3, p. 45]:

$$(\nabla_{\mathbf{u}} F)^T J \nabla_{\mathbf{u}} H = 0. \tag{3.4}$$

By identifying such first integrals and calculating the deviation from their initial values, one can detect non-physical behavior of numerical solutions [19, § 3.3, p. 48].
This is a useful tool to evaluate numerical methods when a closed solution for the ODE does not exist, which is the case for most real-world problems.

3.1.3 Fundamental properties of Hamiltonian systems

From the concept of first integrals, the two fundamental properties of Hamiltonian systems can be derived – the conservation of the Hamiltonian H and symplecticness [13, § 1.1.1, p. 1].
As J is skew-symmetric, setting $F = H$ in (3.4) easily proves that H is a first integral and thus constant along all possible solutions. It now becomes clear, that the skew-symmetry of the structure matrix is closely connected to the property of a constant Hamiltonian.
The second property – symplecticness of the Hamiltonian flow-map – and its interpretation is not as straight forward.
The formal definition of a map $\Psi : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ depending on $\mathbf{u} \in \mathbb{R}^{2n}$, which is symplectic w.r.t J is given by [13, Eq. 1.2, p. 2]:

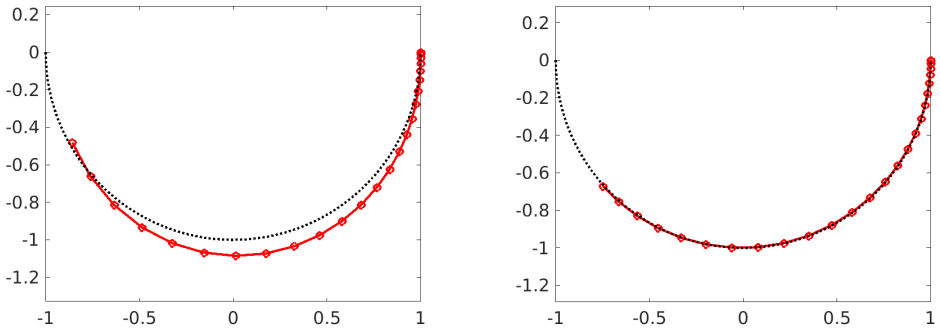
$$(\nabla_{\mathbf{u}} \Psi)^T J \nabla_{\mathbf{u}} \Psi = J. \tag{3.5}$$

This holds true for the Hamiltonian flow-map (3.1) and its structure matrix [19, § 3.5,p. 54].
Symplecticness implies the preservation of volume in the phase space and thus of the underlying geometric structure, which is also referred to as Liouville's theorem [16, § 46, p. 146].
Therefore the conjugate phase space variables q_i and p_i evolve on a co-tangent bundle. In

classical mechanics this can viewed as the direction of motion being part of the tangent space of the configuration manifold. Thus it is a powerful tool to ensure the constraints are not violated, when numerically solving the EOM of a mechanical system.

3.1.4 Example for symplectic time integration

The consequences of preserving symplecticness can be illustrated by looking at a simple pendulum using non-minimal coordinates (x, y). When applying a non-symplectic numerical time-integration method – such as the explicit Euler scheme – the solution then doesn't evolve on the circle the real solution is constricted to, see Fig. 3.1a. Solving the IVP with the symplectic Euler scheme – also known as the implicit midpoint rule – results in a solution which is confined to the correct manifold, see Fig. 3.1b.
This corresponds to the preservation of the underlying geometry, described at the end of the previous section 3.1.3.



(a) explicit Euler (b) symplectic Euler

Figure 3.1: Time integration schemes

3.2 Hamiltonian systems in classical mechanics

The EOM of an unforced, conservative MBS in minimal coordinates can be transformed into a system of the canonical form (3.1), which then means the phase space variables $\mathbf{u}$ consist of the conjugated generalized coordinates $\mathbf{q}$ and the generalized momenta $\mathbf{p}$ [12][§ 4.1, p. 165].
The systems Hamiltonian (3.6) is equivalent to its total stored energy – the sum of the kinetic energy T and the potential energy V. This makes sense, as the property of a constant Hamiltonian in that case corresponds to the property of energy conservation [18][§ 3.4, p. 99].

$$H(\mathbf{q}, \mathbf{p}) = T(\mathbf{q}, \mathbf{p}) + V(\mathbf{q}) \tag{3.6}$$

However the consideration of such simplified systems does not suffice for the purpose of obtaining a suitable simulation for real world problems, as influences like input forces or dissipation can not be modeled in a canonical Hamiltonian system. In addition using non-minimal coordinates introduces constraint forces and changes the structure matrix. Thus, for a certain application of that framework, suitable modifications to equation

(3.1) are necessary. Hence the purpose of this section is to introduce these extensions for our application.

3.2.1 Equations of motion in Hamiltonian form

In our case we consider a MBS with k holonomic constraints represented by algebraic equations in terms of the generalized coordinates $\mathbf{c}(\mathbf{q}) : \mathbb{R}^n \to \mathbb{R}^k$ and input forces $\mathbf{f}(\mathbf{q}) : \mathbb{R}^n \to \mathbb{R}^n$.
The direction of the constraint forces, which are now present due to the choice of using non-minimal coordinates, can be calculated by differentiation of $\mathbf{c}$ w.r.t $\mathbf{q}$ – denoted by $C_q \in \mathbb{R}^{k\times k}$ for better readability –, while the unknown intensities are summarized in the vector $\boldsymbol{\lambda} \in \mathbb{R}^k$ [6][§ 1.3.2, p. 18].
This yields the modified equations [5, § 4.3, eq. 4.1,4.2]:

$$\dot{\mathbf{q}} = \nabla_{\mathbf{p}} H, \tag{3.7}$$

$$\dot{\mathbf{p}} = -\nabla_{\mathbf{q}} H + C_q^T \boldsymbol{\lambda} + \mathbf{f}. \tag{3.8}$$

As $\boldsymbol{\lambda}$ – the so-called Lagrange Multipliers – are unknown, an additional equation is required to solve equation (3.7). Therefore the constraint equations are differentiated w.r.t. time using the chain rule, which yields the missing condition [5, § 4.3, eq. 4.4]:

$$C_q^T \dot{\mathbf{q}} = \mathbf{0}. \tag{3.9}$$

Rewriting the EOM and introducing the time integral of the Lagrange multipliers $\boldsymbol{\Lambda} \in \mathbb{R}^k$ yields the following equation:

$$\underbrace{\begin{bmatrix} & I & -C_q^T \\ -I & & \\ C_q & & \end{bmatrix}}_{\text{modified structure matrix}} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{p}} \\ \dot{\boldsymbol{\Lambda}} \end{bmatrix} = - \begin{bmatrix} H_q \\ H_p \\ H_\Lambda \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{f} \\ \mathbf{0} \end{bmatrix}. \tag{3.10}$$

Note, that the phase space variables $\mathbf{u}$ are now expanded by $\boldsymbol{\Lambda}$, which can again be summarized by:

$$\mathbf{u} = \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \\ \boldsymbol{\Lambda} \end{bmatrix} \in \mathbb{R}^{2n+k}.$$

The modified structure matrix is still skew-symmetric, but not invertible. Hence an explicit flow-map for the ODE can not be set up. However, in the next Section 3.3 we will see, that energy conservation and symplecticness still hold.

3.2.2 Equations of motion in Lagrangian form

Applying the Legendre-transform, the EOM expressed in the Hamiltonian framework, see equation (3.10), can also be written in terms of the generalized coordinates $\mathbf{q}$ and their time derivatives $\dot{\mathbf{q}} = \mathbf{v}$ (3.12), with the Lagrangian

$$L(\mathbf{q}, \dot{\mathbf{q}}) = T(\mathbf{q}, \dot{\mathbf{q}}) - V(\mathbf{q}) \tag{3.11}$$

and the symmetric, positive definite mass-matrix M, which generally depends on the configuration $\mathbf{q}$ [12, § 4.2, p.166, 167]:

$$\begin{bmatrix} & M & -C_q^T \\ -M & & \\ C_q^T & & \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{\Lambda}} \end{bmatrix} = - \begin{bmatrix} L_q \\ L_v \\ L_\Lambda \end{bmatrix}. \tag{3.12}$$

As the proposed energy stable integration method can be applied equally to both formulations, we will use the EOM in Lagrangian form when calculating the entries for the serial manipulator in section 3.4. It provides a more intuitive approach for the derivation of the mass matrix, which only appears as its own inverse in the Hamiltonian form.

3.3 Energy stable time integration

3.3.1 Energy conservation of a continuous system

An energy stable time integration method is proposed by Egger et al. [5]. General evolution problems, which can be expressed in the following gradient form (3.13), are considered [5, § 1, eq. 1.1].

$$E\dot{\mathbf{u}} = -\nabla_{\mathbf{u}} H + \mathbf{f} \tag{3.13}$$

Thereby $E \in \mathbb{R}^{2n \times 2n}$ is a square matrix depending on the vector of state space variables $\mathbf{u} \in \mathbb{R}^{2n}$. H and $\mathbf{f}$ are scalar, respectively vector valued functions, which also depend on the state space.
When comparing the EOM of a mechanical system written in the Hamiltonian framework (3.10) to this general evolution problem described in (3.13), it becomes apparent, that those two equations coincide, with E being the modified structure matrix.
To proove that the energy is conserved in the continous model, one can derive the storage function w.r.t. time using the chain rule and hence exploiting the fact that the evolution problem is a gradient system. The resulting change of the total energy can then can be broken down into two terms: the energy dissipation in the system and the energy change from the external forces [5, § 1,p. 336]:

$$\dot{H} = \langle \nabla_{\mathbf{u}} H, \dot{\mathbf{u}} \rangle = - \underbrace{\langle E\dot{\mathbf{u}}, \dot{\mathbf{u}} \rangle}_{\text{energy dissipation}} + \underbrace{\langle \mathbf{f}, \dot{\mathbf{u}} \rangle}_{\text{external forces}} \tag{3.14}$$

As in our case of a conservative mechanical system E is skew symmetric, the first term will be zero for all possible states $(\mathbf{u}, \dot{\mathbf{u}})$. Thus without the application of external forces H is conserved for all times. In other words, the stored energy can only be changed by the in- and outputs of the system.
When non-conservative forces appear in the mechanical system, E is positive definite, as then energy dissipates, i.e. by friction in the joints of the serial manipulator [5, § 1,p. 336].
The goal when deriving a suitable energy stable integrator is now to keep that property in the discrete case, which is described in the following Section 3.3.2.

3.3.2 Energy conservation of a discrete system

As a MBS is already naturally discretized in space, it is only necessary to discretize the approximate solution of the states $\mathbf{u}$ in time.
There are multiple methods for deriving suitable integrators, one of which is to evaluate the equation over a certain time integral. Thereby both sides are multiplied with an arbitrary test function. The process of the discretization is then carried out by choosing a discrete space for the test- and solution-functions, which produces the method of weighted residuals [9, p. 735, 736].
Choosing different function spaces for the solution and the test functions – also called Petrov-Galerkin approach – the solution $\mathbf{u}_N$ is discretized in time by piecewise polynomial functions of degree $k+1$, while the space for the test functions is those of polynomials of degree k [5, § 3, p. 338].
Choosing $k = 0$ yields a discrete gradient method, which is described in the following based on [5, § 3, p. 339]. It is worth noting, that those integration schemes can also be derived from other frameworks.
As the state space is discretized by linear functionals, $\mathbf{u}$ can be expressed by linear interpolation between the consecutive sample points t_i and t_{i+1}:

$$\mathbf{u}(t) = \mathbf{u}_i + \frac{t - t_i}{\Delta t} \left(\mathbf{u}_{i+1} - \mathbf{u}_i\right), \tag{3.15}$$

$$\dot{\mathbf{u}}(t) = \frac{1}{\Delta t} \left(\mathbf{u}_{i+1} - \mathbf{u}_i\right), \qquad \text{for } t \in [t_i, t_{i+1}]. \tag{3.16}$$

With the test functions being constant, they can be pulled in front of the integral and eleminated on both sides, hence do not play a role in the case of the chosen method with $k = 0$. As in that case the time derivatives of the state space variables $\dot{\mathbf{u}}$ are also constant, they can be pulled out of the respective integrals.
This ultimately leads to the following equation:

$$\int_{t_i}^{t_{i+1}} E \; dt \quad \frac{\mathbf{u}_{i+1} - \mathbf{u}_i}{\Delta t} = -\int_{t_i}^{t_{i+1}} \nabla_{\mathbf{u}} H \; dt + \int_{t_i}^{t_{i+1}} f \; dt. \tag{3.17}$$

With the some simplifications for better readability, denoted by:

$$\bar{\star} = \int_{t_i}^{t_{i+1}} \star dt \qquad \text{and} \qquad \star' = \nabla_{\mathbf{u}} \star, \tag{3.18}$$

the non-linear discrete equation can be written as follows:

$$\bar{E} \; \frac{\mathbf{u}_{i+1} - \mathbf{u}_i}{\Delta t} = -\bar{H}' + \bar{\mathbf{f}}. \tag{3.19}$$

Depending on the underlying evolution problem and the approximation method, these integrals can be evaluated exactly.
If E, H and $\mathbf{f}$ are polynomials of degree p in $\mathbf{u}$, then the exact value can be obtained by using Gauss-quadrature rule with $\frac{p+1}{2}$ points [24, § 3.6, p. 150]. On the other hand, if one of the components is non-polynomial in $\mathbf{u}$, then the numerical integration only produces an approximation of the respective integral, which naturally affects the energy-stability and the exact constraint conservation of the time integration method.
This is especially relevant when including elastic elements like springs, which leads to a non-polynomial term in the gradient of the potential energy, see Section 2.2.
Finally – carrying out the actual time integration – the non-linear equation (3.19) has to be solved for the next state $\mathbf{u}_{n+1}$, which can be done via a line-search method, e.g. the Newton-Raphson algorithm.

3.4 Implementation

3.4.1 Geometry and constraints

The coordinates used for this approach are the COMs positions. However for the constraint relations also include the rotational axes, which consequently need to be calculated. This can be achieved in a recursive manner starting from the base.
Transforming the connection vector between the ith rotation axis and the ith COM and then adding it said rotation axis, yields the following recursive formula:

$$\mathbf{X}_{i+1} = \mathbf{X}_i + A_i \left(\mathbf{x}_i - \mathbf{X}_i\right). \tag{3.20}$$

Choosing a general linear transformation A_i allows for an arbitrary location of the COM, meaning its position is not constricted to the connection between the joints, see Figure 3.2.

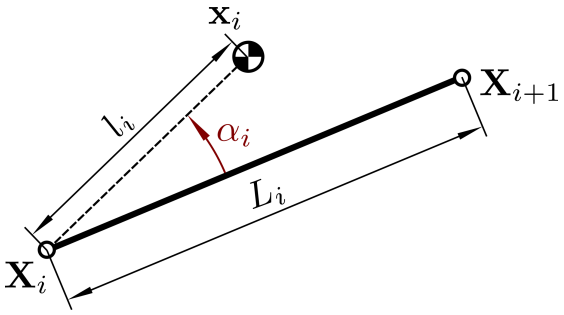


Figure 3.2: Linear Transformation

Although in the following this special case is studied, the calculations for the more general case are described here. A_i thus consists of a rotation and a subsequent scaling, displayed in the equation:

$$A_i = \underbrace{\frac{L_i}{l_{s,i}}}_{\text{scaling}} \underbrace{\begin{bmatrix} \cos\alpha_i & -\sin\alpha_i \\ \sin\alpha_i & \cos\alpha_i \end{bmatrix}}_{\text{rotation}}. \tag{3.21}$$

To ensure that the length l_i of each link stays constant, the following condition needs to be satisfied:

$$(x_i - X_i)^2 + (y_i - Y_i)^2 = l_i^2. \tag{3.22}$$

This yields the n constraint equations $\mathbf{c}$, which can be written in a compact form like so:

$$c_i = (\mathbf{x}_i - \mathbf{X}_i)^T(\mathbf{x}_i - \mathbf{X}_i) - l_i^2, \qquad i = 1, \ldots n. \tag{3.23}$$

For the calculation of the constraint matrix C_q, the recursive formula (3.20) derived w.r.t. $\mathbf{x}$ is used. From the quadratic form of (3.23), one can observe, that the resulting C_q is linear in $\mathbf{x}$, which will be relevant for the time integration later.

3.4.2 Kinetic energy and mass matrix

For the special case of a planar serial manipulator described at the end of chapter 1, the systems kinetic energy T can be written in terms of the inertial coordinates x_i, y_i and φ_i:

$$T(\dot{x}, \dot{y}, \dot{\varphi}) = \sum_{i=1}^{n} \frac{1}{2} m_i {\dot{x}_i}^2 + \frac{1}{2} m_i {\dot{y}_i}^2 + \frac{1}{2} J_i^s {\dot{\varphi}_i}^2. \tag{3.24}$$

As only x_i and y_i are used for the generalized coordinates, we aim to substitute the absolute angular velocities $\dot{\varphi}_i$ in (3.24). Each angle φ_i must satisfy the following trigonometric identities:

$$\Delta x_i := x_i - X_i = l_{s,i} \cos \varphi_i, \tag{3.25}$$
$$\Delta y_i := y_i - Y_i = l_{s,i} \sin \varphi_i. \tag{3.26}$$

After derivation w.r.t. time this results in a kinematic relation of quadratic form between the angular and translational velocities (3.27), which then allows the substitution of $\dot{\varphi}_i^2$ in (3.24).

$${\dot{\varphi}_i}^2 = \frac{{\dot{\mathbf{x}}_i}^T \dot{\mathbf{x}}_i + {\dot{\mathbf{X}}_i}^T \dot{\mathbf{X}}_i - 2{\dot{\mathbf{x}}_i}^T \dot{\mathbf{X}}_i}{l s_{s,i}^2}. \tag{3.27}$$

Along with the translational velocities of the coms $\dot{\mathbf{x}}_i$, the translational velocities of the joints $\dot{\mathbf{X}}_i$ also appear.
$\mathbf{X}_i$ can be obtained by differentiating the recursive formula (3.20) w.r.t. time, with the first axis being fixed at the base.
Rearranging the terms and using the compact notation

$$\tilde{A}_i = I - A_i, \tag{3.28}$$

yields the following equation:

$$\dot{\mathbf{X}}_{i+1} = \tilde{A}_i \dot{\mathbf{X}}_i + A_i \dot{\mathbf{x}}_i. \tag{3.29}$$

With the particular structure of equation (3.29), $\dot{\mathbf{X}}_i$ can be expressed in terms of the generalized velocities $\dot{\mathbf{x}}$ (3.30). For obtaining a more clear structure, the matrices $B_{i,k}$ are introduced.

$$\dot{\mathbf{X}}_i = \sum_{k=1}^{i-1} \tilde{A}_{i-k} \dots \tilde{A}_{k+1} A_k \dot{\mathbf{x}}_k := \sum_{k=1}^{i-1} B_{i,k} \dot{\mathbf{x}}_k. \tag{3.30}$$

Putting all the above components together, the kinetic energy can be expressed in a quadratic form:

$$T(\dot{\mathbf{x}}) = \frac{1}{2} \sum_{i=1}^{n} \begin{bmatrix} \dot{\mathbf{x}}_1^T & \dots & \dot{\mathbf{x}}_{i-1}^T & \dot{\mathbf{x}}_i^T \end{bmatrix} M_i \begin{bmatrix} \dot{\mathbf{x}}_1 \\ \vdots \\ \dot{\mathbf{x}}_{i-1} \\ \dot{\mathbf{x}}_i \end{bmatrix}, \tag{3.31}$$

with the mass matrices for each link M_i:

$$M_i = \begin{bmatrix} B_{i,1}^T B_{i,1} & \dots & B_{i,1}^T B_{i,i-1} & -B_{i,1}^T \\ \vdots & \ddots & \vdots & \vdots \\ B_{i,i-1}^T B_{i,1} & \dots & B_{i,i-1}^T B_{i,i-1} & -B_{i,i-1}^T \\ -B_{i,1} & \dots & -B_{i,i-1} & \begin{bmatrix} m_i + \frac{J_i^s}{l_i^2} & 0 \\ 0 & m_i + \frac{J_i^s}{l_i^2} \end{bmatrix} \end{bmatrix}. \tag{3.32}$$

Those matrices and thus also their allocated matrix M are configuration independent, as the constraints between φ_i and $\mathbf{x}_i$ are applied on a velocity level. This highly simplifies setting up the EOM in Lagrangian form (3.12). While generally M is configuration dependent, with our choice of generalized coordinates and due to the planar structure the mass matrix is now constant. This also leads to a computational advantage, as M only has to be calculated once at the start of a simulation run – just depending on the geometric and inertial parameters of the system.
Assuming the COM lies on the connection axis between joints, the calculation of the mass matrix can be simplified, as the linear transformation now is used purely to scale the vector $(\mathbf{x}_i - \mathbf{X}_i)$ and not rotate it. This leads to decoupled x- and y-coordinates for the COMs and joint positions. Rearranging the vector of the generalized coordinates like so:

$$\mathbf{x} = \begin{bmatrix} x_1 & \dots & x_n & y_1 & \dots & y_n \end{bmatrix}^T, \tag{3.33}$$

gives the mass matrices a block diagonal structure (3.34), which further simplifies the computation.

$$M_i = \begin{bmatrix} M_{i,x} & 0 \\ 0 & M_{i,y} \end{bmatrix} \tag{3.34}$$

3.4.3 Potential energy

The use of the COMs coordinates in an inertial frame allows a straight forward calculation of the gravitational energy G:

$$G(\mathbf{x}) = \sum_{i=1}^{N} m_i g y_i. \tag{3.35}$$

Due to $\mathbf{x}$ appearing linear in G its derivative w.r.t. $\mathbf{x}$ is constant:

$$\nabla_{\mathbf{x}} G = \begin{bmatrix} 0 & \dots & 0 & m_1 g & \dots & m_n g \end{bmatrix}^T. \tag{3.36}$$

Using the vector-chain introduced in Equation (2.6), the springs potential energy S can be calculated.
The entries of the vectors $\mathbf{L}_i$ can simply be obtained by calculating the connection axis between the links COM and the following rotational axis, similar to the method used for the recursive formula (3.20):

$$\mathbf{L}_i = \frac{L_i}{l_i} \begin{bmatrix} x_i - X_i \\ y_i - Y_i \\ 0 \end{bmatrix}. \tag{3.37}$$

The vectors $\mathbf{r}_i$ as described in Section 2.2.4 can be calculated by scaling the cross product between $\mathbf{L}_i$ and the z-axis to the correct length r_i:

$$\mathbf{r}_i = \frac{r_i}{l_i}\left(\begin{bmatrix}0\\0\\1\end{bmatrix} \times \begin{bmatrix}x_i - X_i\\ y_i - Y_i\\ 0\end{bmatrix}\right) = \frac{r_i}{L_i}\begin{bmatrix}-(y_i - Y_i)\\ (x_i - X_i)\\ 0\end{bmatrix}. \tag{3.38}$$

Putting the vector chain together leads to the following expression for the length of the ith spring segment:

$$s_i = \left\| -\frac{r_i}{L_i}\begin{bmatrix}-(y_i - Y_i)\\ (x_i - X_i)\end{bmatrix} + \frac{L_i}{l_i}\begin{bmatrix}x_i - X_i\\ y_i - Y_i\end{bmatrix} + \frac{r_{i+1}}{L_{i+1}}\begin{bmatrix}-(y_{i+1} - Y_{i+1})\\ (x_{i+1} - X_{i+1})\end{bmatrix}\right\|. \tag{3.39}$$

For obtaining the partial derivatives of S in the mono- and multi-articular case the Equations (2.8) and (2.9) can now be used.
Expressing the length s_i in terms of the inertial coordinates yields a vector chain for the ith body containing all the generalized coordinates from the predecessors, as the positions of the joints $\mathbf{X}_i$ are calculated recursively from the base. This has to be taken into account for the partial derivative of the springs potential energy.

3.4.4 Discrete equations of motion

With equations for the kinetic and potential energy in terms of the chosen coordinates and their time derivatives, the gradient of the Lagrangian can be calculated. As $\mathbf{x}_i$ appears linear in $V(\mathbf{x}_i)$, the resulting vector for L_q is constant and thus can also be calculated in advance.

$$L_q = \nabla_{\mathbf{x}} V = \nabla_{\mathbf{x}} G + \nabla_{\mathbf{x}} S \tag{3.40}$$
$$L_v = M\mathbf{v} \tag{3.41}$$
$$L_\Lambda = \mathbf{0} \tag{3.42}$$

Using the Petrov-Galerkin approach with linear interpolation as proposed in [5, § 3, p. 339] for the time discretization of the EOM, leads to the following non-linear, implicit equation for each timestep:

$$\frac{1}{\Delta t}\begin{bmatrix} & M & -\tilde{C}_q^T\\ -M & & \\ \tilde{C}_q & & \end{bmatrix}\begin{bmatrix}\mathbf{q}_{t+1} - \mathbf{q}_t\\ \mathbf{v}_{t+1} - \mathbf{v}_t\\ \mathbf{\Lambda}_{t+1} - \mathbf{\Lambda}_t\end{bmatrix} + \begin{bmatrix}\nabla_{\mathbf{x}}\tilde{V}\\ M\\ \mathbf{0}\end{bmatrix} = \mathbf{0}. \tag{3.43}$$

Thereby the numerical approximation of the integral is denoted by:

$$\tilde{\star} \approx \int_{t_i}^{t_{i+1}} \star \, d\tau \tag{3.44}$$

With the components above, the energy stable time integration scheme described in the previous Section 3.3.2, can now be implemented.
The structure of Equation (3.43) allows for the explicit calculation of the Jacobian needed for the Newton-Raphson linesearch method, which can be used to reduce calculation time.

Chapter 4

Variational integrators on Lie groups

This chapter introduces a structure preserving time integration, which is achieved by using variational integrators based on the Lie groups.
In Section 4.1 a brief introduction about Lie groups and Lie algebra is given. Thereby the focus lies on the aspects, that are important, when using this concept to represent rigid body motions.
Afterwards, in Section 4.2 the key concepts of Screw Theory – offering a natural application for Lie groups – are presented. The Book [20], which offers a detailed insight into the topic, is used.
Finally the theory is tied together by describing the development and implementation of a variational integrator for the EOM, see Sections 4.3 and 4.4.

4.1 Lie groups and Lie algebra

If a matrix group G is a subset of all quadratic, real and invertible matrices $GL(n)$ and in addition satisfies closure under matrix multiplication - meaning that the product of two elements remains in that group - G is a Lie Group [18, § 1.2.6, p. 25]. The exponential map

$$\exp : \mathfrak{g} \to G \tag{4.1}$$

is the connection between such a Lie Group and its corresponding Lie algebra $\mathfrak{g}$ [10, § 14.1, p. 367, 368]. This rather abstract mathematical concept, can be viewed as a form of linearization of G, which then allows a parameterization of G by - in some cases simpler - elements of $\mathfrak{g}$ [10, § 14.2, p. 376, 377].
In the context of describing rigid body motions, there are two important Lie groups:
The first is the group of special orthogonal matrices $SO(3)$ (4.2), which due to their length and angle preserving properties can be used to represent rotations [20, § 2, p. 24].

$$SO(3) : \left\{ R \in \mathbb{R}^{n \times n} \quad | \, RR^T = I, \quad \det(R) = 1 \right\} \tag{4.2}$$

The second is the group of homogenous matrices $SE(3)$ consisting of a rotation matrix $R \in SO(3)$ and an additional vector $\mathbf{p} \in \mathbb{R}^3$ for the translation. Elements of $SE(3)$

are often used for representing an arbitrary rigid body motion without the drawbacks of Euler angles. Formally this group can be represented like so [20, § 3, p. 35]:

$$SE(3) : \left\{ g = \begin{bmatrix} R & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \;\middle|\; R \in SO(3), \quad \mathbf{p} \in \mathbb{R}^3 \right\}. \tag{4.3}$$

The exponential maps with the corresponding Lie algebras $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$ are surjective [10, § 14.2, p. 377], [10, § 14.6, p. 387] and thus locally invertible. Note that surjectivity of the exponential map does not generally hold true for all Lie groups.
It can be shown, that the elements of $\mathfrak{so}(3)$ are skew symmetric matrices with zeros on the diagonal. Hence, there are only three independent entries, which can be summarized in a vector $\mathbf{k} \in \mathbb{R}^3$. Hence a simpler representation of $SO(3)$ is its Lie algebra. The connection between such a matrix and its corresponding vector is given by the hat map [7, § 1, p.1]:

$$\hat{\mathbf{k}} : \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix}. \tag{4.4}$$

Looking at the corresponding rigid body rotation, the vector $\mathbf{k}$ is equivalent to the rotation axis of the motion, and the norm $|\mathbf{k}|$ is the angle φ by which the body is rotated. With the Rodriguez-Formula an element of $SO(3)$ can be directly computed from its Lie algebra [4, § 2.2.1.4, p. 34, 35]:

$$R = I_{3\times 3} + \sin\varphi\hat{\mathbf{k}} + (1 - \cos\varphi)\hat{\mathbf{k}}^2 \tag{4.5}$$

In the following Section 4.2, the concept of the screw theory is introduced, as it offers a comprehensive approach to the Lie group of homogenous matrices $SE(3)$.

4.2 Screw theory

4.2.1 The exponential map

Chasles Theorem states, that every rigid body motion can be described as a rotation about a line in space and a translation parallel to that line. This kind of description is referred to as a screw motion, due to its resemblance to the helical line of a screw. The infinitesimal version of such a motion is called a twist, which is a parameterization of the instantaneous velocity of a rigid body. Although somewhat non-intuitive, this viewpoint supports a global understanding of a systems kinematics. Thus - in contrast to e.g. Euler angles - it doesn't suffer from singularities due to the parameterization. It is worth noting that the concept of using screw coordinates is not unique to twists, but can also be applied to describe force-moment pairs acting on a body.[20, § 2, p. 19, 20]. For obtaining the twist, we look at the velocity of a point $\dot{\mathbf{p}}(t)$ undergoing such a screw motion:

$$\dot{\mathbf{p}}(t) = \boldsymbol{\omega} \times (\mathbf{p}(t) - \mathbf{r}) + h\boldsymbol{\omega}. \tag{4.6}$$

Thereby $\boldsymbol{\omega}$ is the rotational velocity vector pointing in the direction of the twists axis, $\mathbf{r}$ is an arbitrary point on that axis and h is the so called pitch relating the magnitudes

of the translational and rotational motion [20, § 3.2, p. 19, 20].
With the skew symmetric matrix $\hat{\omega}$ denoting the cross-product with $\boldsymbol{\omega}\times$ – which is equivalent to the hat map (4.4) discussed before – and the vector for the translational component of the twist $\mathbf{v} := -(\boldsymbol{\omega} \times \mathbf{r}) + h\boldsymbol{\omega}$, the equation (4.6) can be written in a compact matrix-vector notation:

$$\begin{bmatrix} \dot{\mathbf{p}}(t) \\ 0 \end{bmatrix} = \underbrace{\begin{bmatrix} \hat{\omega} & \mathbf{v} \\ 0 & 0 \end{bmatrix}}_{twist\ \hat{\xi}} \begin{bmatrix} \mathbf{p}(t) \\ 1 \end{bmatrix}. \tag{4.7}$$

With the use of homogenous coordinates $\bar{\mathbf{p}}$ the following first order ODE results:

$$\dot{\bar{\mathbf{p}}} = \hat{\xi}\bar{\mathbf{p}}. \tag{4.8}$$

The matrix $\hat{\xi}$ is called the twist and the six independent entries

$$\xi = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v} \end{bmatrix} \tag{4.9}$$

are called the twist coordinates [20, § 3.2, p.40].
When a constant twist is applied, (4.8) can be solved by [20, § 23.2, p. 40]:

$$\bar{\mathbf{p}}(t) = e^{\hat{\xi}t}\bar{\mathbf{p}}(0). \tag{4.10}$$

As this is a valid solution for any point $\bar{\mathbf{p}}$ on the rigid body, the matrix exponential $\exp(\hat{\xi}t)$ is equal to the homogenous transformation between the configuration of the body at time 0 and time t, thus corresponding to a parametrization of the body-fixed coordinate system. Using unit twists, we can apply them for a given magnitude instead of a time interval. Such a twist can also be applied to a homogenous transformation $g(0) \in SE(3)$.
Looking now back to the definition of the exponential map (4.1), the connection to the previously introduced concept of Lie groups becomes obvious: The twists $\hat{\xi} \in \mathfrak{se}(3)$ are the Lie algebra to the Lie group of homogenous transformations $SE(3)$. The inverse of the corresponding exponential map can be calculated by a closed formula described in [20, Eq. 2.36, p.42], which can be viewed as an extension of the previously introduced Rodriguez-Formula, see Equation (4.5).

4.2.2 Serial kinematics

In our case of a serial manipulator with rotational joints, the magnitude is equivalent to the relative angles $\boldsymbol{\varphi}$ between the links. Sequentially applying constant twists for a kinematic chain up until the jth link – of which we want to calculate the new homogenous transformation $g_j(\boldsymbol{\varphi})$ from the initial transformation $g_j(\boldsymbol{\varphi} = \mathbf{0})$ – yields [20, § 2.2, p. 87]:

$$g_j(\boldsymbol{\varphi}) = e^{\hat{\xi}_1\varphi_1} \dots e^{\hat{\xi}_j\varphi_j} g_j(\mathbf{0}). \tag{4.11}$$

This way the motion between a link and its previous counterpart is not directly represented. Instead the twists are specified w.r.t. the same reference configuration. With

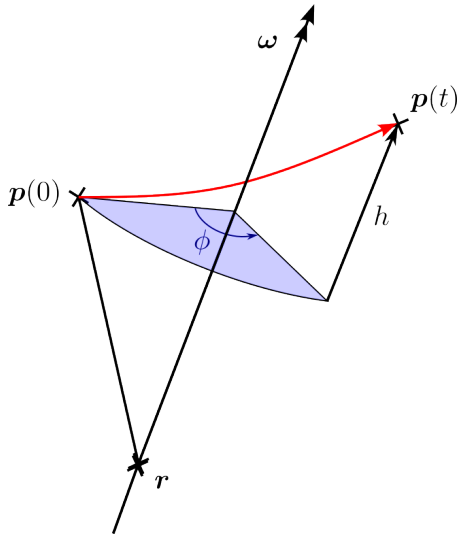


Figure 4.1: Screw motion of a point **p**

$j = n$ and $g_n = g_{tcp}$, we can calculate the position and orientation of the tool center point (TCP) of a serial manipulator from equation (4.11), which is equivalent to solving the forward kinematics of the mechanism.
The relative unit twist for two links connected by a rotational joint ξ_{rot} can be calculated with $||\boldsymbol{\omega}|| = 1$ [20, § 3.3, p. 50]:

$$\xi_{rot} = \begin{bmatrix} -\mathbf{r} \times \boldsymbol{\omega} \\ \boldsymbol{\omega} \end{bmatrix}. \tag{4.12}$$

4.2.3 Velocity

The calculation of the instantaneous velocity of a point on a link is more complicated. As $g(t) \in SE(3)$ is not an element of the euclidian space and consequently $\dot{g}(t) \notin SE(3)$, g can not simply be derived w.r.t. time to obtain the velocity [20, § 4.2, p. 53,54].
Instead the spatial velocity $\hat{V}^s \in \mathfrak{se}(3)$ in inertial coordinates corresponding to a rigid motion $g(t)$ is defined as $\hat{V}^s$ in the following equation [20, Eq. 2.53, p. 54]:

$$\hat{V}^s = \dot{g}g^{-1} = \begin{bmatrix} \hat{\omega}^s & v^s \\ 0 & 0 \end{bmatrix}. \tag{4.13}$$

$\hat{V}^s$ has the form of a twist and thus maps a point on the rigid body to its respective velocity measured in an inertial frame. The associated screw gives the instantaneous axis, pitch and magnitude of the rigid motion described by $g(t)$.
Respectively the spatial velocity $\hat{V}^b \in se(3)$ w.r.t. a body-fixed coordinate system can be calculated as follows [20, Eq. 2.55, p. 55]:

$$\hat{V}^b = g^{-1}\dot{g} = \begin{bmatrix} \hat{\omega}^b & v^b \\ 0 & 0 \end{bmatrix}. \tag{4.14}$$

It is important to note, that $\hat{V}^b$ still represents an absolute velocity, expressed in the instantaneous body coordinates and not a motion relative to the body.

4.2.4 Manipulator Jacobian

We are now interested in how the spatial velocity of a point on our multi body system is related to the rotational velocity of the joints $\dot{\boldsymbol{\varphi}}$, which we can directly control with the input torques.
Calculating the spatial velocity of the TCP from equation (4.13) by using the relation (4.11), yields the following linear mapping [20, Eq. 3.54, p. 116]:

$$V_{tcp}^s = J_{tcp}^s \dot{\boldsymbol{\varphi}} \quad , \tag{4.15}$$

with:

$$J_{tcp}^s = \begin{bmatrix} \xi_1 & \xi_1' & \dots & \xi_j' \end{bmatrix} \quad , \tag{4.16}$$

$$\xi_i' = Ad(e^{\hat{\xi}_1 \varphi_1} \dots e^{\hat{\xi}_{i-1} \varphi_{i-1}}) \xi_i. \tag{4.17}$$

Such a linear relation between the velocity of the TCP and the joint velocities is traditionally obtained by derivation of a local, vector-valued parameterization of the forward kinematics. This Jacobian suffers from singularities due to the parameterization and thus fundamentally differs from the manipulator Jacobian J_{tcp}^s described above. It can be shown, that the ith column of J_{tcp}^s is the ith joint twist, transformed to the current manipulator configuration [20, § 4, p. 115].
In the same way, the body manipulator Jacobian J_{tcp}^b can be defined [20, Eq. 3.55, p. 117]:

$$V_{tcp}^b = J_{tcp}^b \dot{\boldsymbol{\varphi}} \quad , \tag{4.18}$$

with:

$$J_{tcp}^b = \begin{bmatrix} \bar{\xi}_1 & \dots & \bar{\xi}_n \end{bmatrix} \quad , \tag{4.19}$$

$$\bar{\xi}_i = Ad^{-1}(e^{\hat{\xi}_i \varphi_i} \dots e^{\hat{\xi}_n \varphi_n} g_{tcp}(0)) \xi_i. \tag{4.20}$$

Here the columns represent the joints twists w.r.t. the current configuration of the tcp frame.
As twists in different coordinate systems are related via a similarity transformation, the adjoint transformation of a homogeneous transformation matrix $g \in SE(3)$ can be defined as follows [20, Eq. 2.58, p.55]:

$$Ad(g) = \begin{bmatrix} R & \hat{p}R \\ 0 & R \end{bmatrix}. \tag{4.21}$$

4.3 Variational integrators

4.3.1 Background

The general idea for deriving a suitable variational integrator for the a multi body system is to directly discretize Hamilton's principle of stationary action instead of the continous equations of motions. Thereby restricting the motion to the Lie groups described in Section 4.1 when deriving such an integrator, ensures that the variations still lie in the

appropriate configuration space, thus preserving the variational of the system [17, § 2.1, p. 2908, 2909].
In general, dynamical systems evolve on differentiable manifolds embedded in the euclidian space $\mathbb{R}^3$ [18, § 1.2, p. 11,12]. The manifold for a specific system is created by its linearly independent kinematic couplings between the links, which – in the considered case of holonomic constraints – can be represented by a set algebraic equations, depending on the generalized coordinates, which are equivalent to those introduced in Section 3.4.1. Hence this is a partial formalization of the restricted solution space due to the Hamiltonian structure, which we touched on at the beginning of Section 3.1.2.

4.3.2 Obtaining the continuous Euler-Lagrange equations

Considering the special case of a serial manipulator with rotational joints, the geometry of the solution space can be described by a chain of circles, the so called one-spheres S^1, see Equation (4.22). A natural choice of coordinates are the normalized attitude vectors $\mathbf{q}_i$ between the com of a body and its previous joint. As the considered revolute joints only have one degree of freedom, $\mathbf{q}_i$ - if measured in a suitable body-fixed coordinate system associated with the previous link - evolves on a plane and thus can be chosen from $\mathbb{R}^2$ instead of $\mathbb{R}^3$ [18, § 4.1, p. 131,132].

$$S^1 = \{\mathbf{x}_i \in \mathbb{R}^2 : ||\mathbf{x}_i||^2 - 1 = 0\} \tag{4.22}$$

Intuitively, the tangent space of such a sphere is the tangent of the circle at each point. Thus, with the skew-symmetric matrix

$$S = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \tag{4.23}$$

which rotates a vector $\mathbf{q}_i \in \mathbb{R}^2$ orthogonal to its previous configuration, the following globally valid kinematic equations can be formulated [18, § 4.2, p. 132,133]:

$$\dot{\mathbf{x}}_i = \omega_i S \mathbf{x}_i \tag{4.24}$$

By restricting the possible time derivatives, where now only the scalar factor ω_i can be chosen, every motion of the system evolves on the chain of one-spheres $(S^1)^n$.
Hamiltonians principle of stationary action states, that the infinitesimal variation of the action integral along any motion is zero. Calculating those variations by using the exponential map, such that the structural properties remain, ultimately leads to the continous Euler-Lagrange equations on the configuration manifold (4.25). With the kinematic relation (4.24), the scalar rotational velocity for each joint ω_i instead of the time derivative of the generalized coordinates $\dot{\mathbf{q}}_i$ can be used [18, Eq. 4.12, p. 140].

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \omega_i}\right) + \mathbf{q}_i^T S \left(\frac{\partial L}{\partial \mathbf{q}_i}\right) = 0 \tag{4.25}$$

4.3.3 Discretization of the Euler-Lagrange equations

To now obtain the discrete Lagrangian map L_d, which by definition is a variational integrator for the Euler-Lagrange equations, first the Lagrangian is approximated between

two consecutive sample points q_k, q_{k+1}, with the time in between denoted as Δt [17, § 2.1, p. 2909].

$$L_d(q_k, q_{k+1}) \approx \int_0^{\Delta t} L(q_{k,k+1}(t), \dot{q}_{k,k+1}(t))\, dt \tag{4.26}$$

In the discrete case the variation of an action sum instead of action an integral is taken, which yields the discrete Euler-Lagrange equations:

$$D_2 L_d(q_{k-1}, q_k, \Delta t) + D_1 L_d(q_k, q_{k+1}, \Delta t) = 0, \tag{4.27}$$

with D_1 and D_2 denoting the discrete derivatives taken at the start respectively at the end of the corresponding interval [17, § 2.1, p. 2909]. For both approximations appropriate numerical methods can be chosen.
With the choice of the rectangle rule for approximating the Lagrangian and the implicit midpoint rule for its derivative, the discrete EOM for the unforced system result in the following equation, as presented in [14, § 4, p. 233, 234]:

$$\begin{aligned} M_d(q_{k-1}, q_k)\omega'_{k-1} + \frac{\Delta t}{4} C_d(q'_{k-1}, \omega'_{k-1}) - \frac{\Delta t}{2} G(q'_{k-1}) \\ - M_d(q_k, q_{k+1})\omega'_k + \frac{\Delta t}{4} C_d(q'_k, \omega'_k) - \frac{\Delta t}{2} G(q'_k) = 0. \end{aligned} \tag{4.28}$$

Thereby the attitude vectors and rotational velocities in the middle of a time interval $(k, k+1)$ are denoted with with a prime $\star'_k$. The resulting nonlinear equation contains the unknown values of $\star'_k$ and $\star_k$, which are dependant through the kinematic equation (4.24). For the discrete updates compatible with those structural constraints, a matrix rotating the attitude vector $F'_{i,k} \in SO(2)$ used [14, Eq. 10, 11, p. 233]:

$$\mathbf{q}'_{i,k} = F'_{i,k}\mathbf{q}_{i,k} \quad , \tag{4.29}$$

$$\mathbf{q}_{i,k+1} = F'_{i,k}\mathbf{q}'_{i,k} \quad , \tag{4.30}$$

$$\omega'_{i,k} = \frac{1}{\Delta t}\mathbf{q}_{i,k}^T S^T \mathbf{q}_{i,k+1}. \tag{4.31}$$

Hence equation (4.28) together with (4.29) can be solved for the scalar values $f'_{i,k} \in so(2)$, which are connected to $F'_{i,k}$ by the Rodriguez formula. After obtaining the solution for each time step the values for $\mathbf{q}_i$ and $\boldsymbol{\omega}$ can be updated.

4.4 Implementation

4.4.1 Geometry

To ensure the motion of the system is compatible with the constraints of $(S^1)^n$, we use the normalized attitude vectors

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \dots & \mathbf{q}_n \end{bmatrix} \tag{4.32}$$

as our position coordinates.
Thereby each attitude vector $\mathbf{q}_i$ is measured in the coordinate system of its predecessor $(i-1)$. Thus there is a direct relation between each attitude vector $\mathbf{q}_i$ and the relative angle φ_i between link i and link $(i-1)$.

$$\mathbf{q}_i = \begin{bmatrix} \cos(\varphi_i) \\ \sin(\varphi_i) \end{bmatrix} \tag{4.33}$$

4.4.2 Kinetic energy and mass matrix

To derive the equations of motion from Hamiltons Principle of stationary action, we first need to calculate the kinetic energy of the system. As the generalized inertia-matrix

$$M_i = \begin{bmatrix} m_i & & & & & \\ & m_i & & & & \\ & & m_i & & & \\ & & & J_i^{com} & & \end{bmatrix} \in \mathbb{R}^{6x6} \quad , \tag{4.34}$$

is constant when measured in the body coordinates of the respective link, we exploit the following relationship between the absolute translational and rotational velocity in body coordinates and the body manipulator Jacobian

$$V_{com,i}^b = J_{com,i}^b \dot{\varphi}_i \quad , \tag{4.35}$$

to write the kinetic energy T as follows [14, § 3.1, p. 232]:

$$T = \frac{1}{2} \sum_{i=1}^{n} (V_{com,i}^b)^T M_i V_{com,i}^b = \frac{1}{2} \sum_{i=1}^{n} \dot{\varphi}_i (J_{com,i}^b)^T M_i J_{com,i}^b \dot{\varphi}_i. \tag{4.36}$$

Using this quadratic form, the kinetic energy is expressed w.r.t. the angular velocities of the rotational joints, which are denoted as the vector

$$\boldsymbol{\omega} = \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix} \tag{4.37}$$

in the following. With defining the now configuration dependant mass-matrix $M(\mathbf{q})$ as

$$M(\mathbf{q}) = \sum_{i=1}^{n} (J_{com,i}^b)^T M_i J_{com,i}^b, \tag{4.38}$$

we can rewrite the kinetic energy in the more compact form [14, § 3.1, p. 232]:

$$T = \frac{1}{2} \boldsymbol{\omega}^T M(\mathbf{q}) \boldsymbol{\omega}. \tag{4.39}$$

This strongly resembles the classical approach discussed in Section 1.1, as the mass matrix $M(\mathbf{q})$ is exactly the same. We now however express M in terms of the attitude vectors and use screw theory with the resulting body-manipulator Jacobian for its construction:

$$J_{com,i}^b = \begin{bmatrix} \bar{\xi}_{i,1} & \dots & \bar{\xi}_{i,i} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix}, \tag{4.40}$$

$$\bar{\xi}_{i,j} = Ad^{-1}(e^{\hat{\xi}_j \varphi_j} \dots e^{\hat{\xi}_i \varphi_i} g_{com,i}(0)) \xi_j. \tag{4.41}$$

Note that the later links don't have an effect on the Jacobians of the previous links, hence the $(i+1)$th - nth column are zero.

4.4.3 Potential energy

The gravitational potential energy G of the system can be written with the vectors to the COM $\mathbf{r}_{com,i}$ measured in an inertial coordinate system. As the chosen coordinates are written w.r.t the coordinate system of the previous link, applying a transformation from body to inertial coordinates is necessary. This is achieved by applying equation (4.11) to the COM coordinates of the initial configuration.
The gravitational potential energy can thus be calculated as follows [14, § 3.1, p. 233]:

$$G(\mathbf{q}) = \sum_{i=1}^{n} m_i g \mathbf{e}_g^T \mathbf{r}_{com,i}. \tag{4.42}$$

For the calculation of the springs potential energy S, the length of the elastic elements s_i has to be expressed w.r.t. the attitude vectors, analogous to the calculations for the energy stable approach in Section 3.4.3.
This way the springs length result in

$$s_i = \left\| - \begin{bmatrix} 0 \\ r_{i-1} \\ 0 \end{bmatrix} + L_i \begin{bmatrix} q_{i,x} \\ q_{i,y} \\ 0 \end{bmatrix} + r_i \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} q_{i,x} \\ q_{i,y} \\ 0 \end{bmatrix} \right) \right\|, \tag{4.43}$$

which – by omitting the third vector entry – yields:

$$s_i = \left\| \begin{bmatrix} L_i q_{i,x} - r_i q_{i,y} \\ -r_{i-1} + L_i q_{i,y} + r_i q_{i,x} \end{bmatrix} \right\|. \tag{4.44}$$

These results for s_i can be inserted into the Equations (2.3), (2.4), (2.8) and (2.9) to calculate the springs stored energy respectively its gradient.
Furthermore, it is interesting to note, that the attitude vectors are written in the coordinate system of the previous body and normalized, which simplifies the calculation compared to the use of the COMs positions in an inertial coordinate system, as each springs length only depends on the corresponding attitude vector. This allows an individual calculation of the partial derivatives of S_i in the mono-articular case.

4.4.4 Continuous equations of motion

With the chosen coordinates $\mathbf{q}$ and $\boldsymbol{\omega}$ and the total potential energy V, the Lagrangian for the system can be written as [14, Eq. 3, p. 232]:

$$L(\mathbf{q}, \boldsymbol{\omega}) = T(\mathbf{q}, \boldsymbol{\omega}) - V(\mathbf{q}) = \frac{1}{2} \boldsymbol{\omega}^T M(\mathbf{q}) \boldsymbol{\omega} - V(\mathbf{q}). \tag{4.45}$$

While calculating the partial derivative of the Lagrangian w.r.t. $\boldsymbol{\omega}$ is straight forward, as the only term depending on $\boldsymbol{\omega}$ is the vector itself, the derivation w.r.t. $\mathbf{q}$ is more complicated due to the configuration dependent mass matrix $M(\mathbf{q})$.
For this calculation the construction of the mass matrix - the quadratic form with a constant part in the middle (4.38) - can be exploited [11, § 2, p. 2365]:

$$\frac{\partial M(\mathbf{q})}{\partial \star} = \sum_{i=1}^{n} \left(\frac{\partial J_{com,i}^b}{\partial \star} \right)^T M_i J_{com,i}^b + \left(J_{com,i}^b \right)^T M_i \left(\frac{\partial J_{com,i}^b}{\partial \star} \right). \tag{4.46}$$

Thus the only component need is the partial derivative of the body manipulator Jacobian w.r.t. the x and y components of the attitude vectors, denoted by $q_{i,x}$ and $q_{i,y}$.
Hence it is useful to express the matrix exponential of the rotational twists we use to describe the serial kinematic (4.11) in terms of $\mathbf{q}_i$. With using the abbreviation $\tilde{L}_i = \sum_{k=1}^{i-1} L_k$ the twist of the ith joint can be written like so:

$$e^{\hat{\xi}_i\varphi_i} = \begin{bmatrix} \cos(\varphi_i) & -\sin(\varphi_i) & 0 & \tilde{L}_i(1-\cos(\varphi_i)) \\ \sin(\varphi_i) & \cos(\varphi_i) & 0 & -\tilde{L}_i\sin(\varphi_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} q_{i,x} & -q_{i,y} & 0 & \tilde{L}_i(1-q_{i,x}) \\ q_{i,y} & q_{i,x} & 0 & -\tilde{L}_i q_{i,y} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.47)$$

As the body manipulator Jacobian consists of the joints twists written in their respective body frame, the adjoint transformation is needed.

$$Ad(e^{\hat{\xi}_i\varphi_i}) = \begin{bmatrix} q_{i,x} & -q_{i,y} & 0 & 0 & 0 & -\tilde{L}_i q_{i,y} \\ q_{i,y} & q_{i,x} & 0 & 0 & 0 & \tilde{L}_i(q_{i,x}-1) \\ 0 & 0 & 1 & \tilde{L}_i q_{i,y} & \tilde{L}_i(q_{i,x}-1) & 0 \\ 0 & 0 & 0 & q_{i,x} & -q_{i,y} & 0 \\ 0 & 0 & 0 & q_{i,y} & q_{i,x} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.48)$$

Calculating the inverse numerically for each timestep is computationally expensive, thus the inverse is calculated analytically beforehand.

$$Ad^{-1}(e^{\hat{\xi}_i\varphi_i}) = \begin{bmatrix} q_{i,x} & q_{i,y} & 0 & 0 & 0 & \tilde{L}_i q_{i,y} \\ -q_{i,y} & q_{i,x} & 0 & 0 & 0 & \tilde{L}_i(q_{i,x}-1) \\ 0 & 0 & 1 & -\tilde{L}_i q_{i,y} & \tilde{L}_i(q_{i,x}-1) & 0 \\ 0 & 0 & 0 & q_{i,x} & q_{i,y} & 0 \\ 0 & 0 & 0 & -q_{i,y} & q_{i,x} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.49)$$

The differentiation w.r.t $q_{i,x}$ and $q_{i,y}$ of these matrices is now straightforward and results in constant matrices only depending on the geometry of the system. These matrices then can be used for the differentiation of the body manipulator Jacobian w.r.t. the attitude vectors.
As the forward kinematics equation (4.11) appearing in the Jacobian consist of the sequential application of transformations - each only depending on the corresponding

attitude vector - the derivation of its columns respectively the twists is straightforward:

$$\frac{\partial \bar{\xi}_{i,j}}{\partial \mathbf{q}_k} = \begin{cases} Ad^{-1}(e^{\hat{\xi}_j \varphi_j} \dots \frac{\partial e^{\hat{\xi}_k \varphi_k}}{\partial \mathbf{q}_k} \dots e^{\hat{\xi}_i \varphi_i} g_{com,i}(0)) \xi_j, & j < k < i, \\ \mathbf{0}, \quad else. \end{cases} \tag{4.50}$$

This allows the calculation of the continous EOM in the same form as (1.1). Due to the use of Lie groups when constructing the variational integrator, the attitude vectors – which are non-minimal coordinates – can be used for expressing the EOM without explicit constraint forces appearing in the resulting equation.
Although this way of obtaining the EOM seems more complicated than the classical way using minimal coordinates, it holds some significant advantages. There is no need for the explicit differentiation of a local map from joint to cartesian space. The calculation of that local Jacobian either causes a complicated differentiation of trigonometric functions or the need for a recursive approach beginning from the base, a method used for constructing a reference model, see Section 5.1.6. With using the body manipulator Jacobian, the complex trigonometry is hidden in the exponential map.

4.4.5 Discrete equations of motion

The discretization of the Hamiltonian principle of stationary action yields the non-linear equation (4.28). With continous mass-matrix and Coriolis vector obtained in the previous section, this equation can now be solved w.r.t the matrices rotating the attitude vectors for each timestep.
As the attitude vectors are rotated on a plane, those matrices $F'_{i,k}$ are elements of $SO(2)$ and thus can be parametrized by just one independent value.
Using the simplified Rodriguez formula for such a case (4.51) [14, § 4.3, p. 234], allows solving the equation w.r.t. the linearily independent scalars. This ensures that the attitude vectors keep their unit length, which is equivalent to conserving the symplectic property.

$$F'_{i,k} = I_{2\times 2} \cos f'_{i,k} + S \sin f'_{i,k} \tag{4.51}$$

The non-linear solver 'fsolve' implemented in the optimization toolbox of MATLAB [25] is then used.

Chapter 5

Results and discussion

In this chapter the two approaches for the time integration described in the previous Chapters 3 and 4 are evaluated. For comparing the results with a more traditional approach, a reference simulation in minimal coordinates is implemented.
First a short overview of the methodology is given in Section 5.1, where the criteria for the following evaluation of the two methods are given, as well as the simulation parameters. The evaluation is treated in Section 5.2, divided into the described criteria. Afterwards the results are discussed in the last Section 5.3 of this chapter.
In the following, when referring to the method using a variational integrator on Lie groups presented in Chapter 4, the approach is shortened to 'variational integrator'.

5.1 Methodology

5.1.1 Simulation parameters

For comparing the two methods described in the previous sections, a three-link mechanism is chosen. Table 5.1 displays the set of geometric and inertial parameter used – if not specified otherwise –, see Figures 2.1 and 2.2 for referencing the displayed symbols. A standard simulation run to check the validity of both methods, as well as their energy and symplecitc behavior, is defined for a duration of 50 seconds with a time step of $\Delta t = 0.01$s, under the influence of the gravitational acceleration $g = 1\frac{\mathrm{m}}{\mathrm{s}^2}$.
If not specified otherwise, mono-articular springs are used, when studying the effects of additional elastic elements on the time integration schemes.

5.1.2 Symplecticness

As mentioned in Section 3.1.4, evaluating the symplecticness of a time integration method when using non-minimal coordinates can be done by checking if the systems geometric structure is preserved. This requires evaluating if the constraints – coupling our chosen set of coordinates – are fulfilled.
For the energy stable approach, explicit configuration dependent constraint equations are used, see Equation (3.23). Hence those can simply be evaluated for each time step, which results in the following error for each link:

$$e_{\text{constraint},i} = (\mathbf{x}_i - \mathbf{X}_i)^T(\mathbf{x}_i - \mathbf{X}_i) - l_i^2, \qquad i = 1, \dots n. \tag{5.1}$$

	1	2	3
m in kg	1.0	1.0	1.0
J^s in kg $\cdot$ m^2	1.0	1.0	1.0
L in m	1.0	1.0	1.0
l in m	0.5	0.5	0.5
r in m	0.4	0.4	0.4
s_0 in m	0.2	0.2	0.2
k in N/m	1.0	1.0	1.0

Table 5.1: Parameters for the standard simulation run

In the case of the variational integrator, the structure preservation is fulfilled by the attitude vectors $\mathbf{q}_i$ being normal vectors. Thus the deviation of the norm is checked, which results in the following error for each link:

$$e_{\text{norm},i} = \left| \|\mathbf{q}_i\| - 1 \right|. \tag{5.2}$$

The errors introduced in this Section can be interpreted as the deviation of the current link length from its starting length. Explicit time integration schemes using non-minimal coordinates typically suffer from an increasing deviation in link length, which is referred to as the drift-off phenomenon [6, § 5.2, p. 151].

5.1.3 Energy conservation

To evaluate the property of energy conservation for both time integration schemes, the deviation of the systems total energy from its energy at the start of the simulation $t = 0s$ is measured, which yields the following energy-error:

$$e_{\text{energy}} = E - E_{\text{start}}. \tag{5.3}$$

In contrast to the error arising due to constraint violation described in the previous Section 5.2.1, when looking at the energy error the whole system has to be considered as opposed to the single links.

5.1.4 Computational performance

Another important criterion is the computational time required for carrying out the time integration. Thereby the calculation time is compared to the time passing in the simulation, leading to the following equation:

$$t_{\text{realtime}} = \frac{t_{\text{computation}}}{t_{\text{simulation}}}, \tag{5.4}$$

by which the real-time capabilities of the methods can be determined. If $t_{\text{realtime}} < 1$, then in theory the time integration can be carried out in real time, which means that a simulation-based control strategy could be realized. In practice t_{realtime} should be significantly smaller than 1, leaving a margin for other processes.

The simulation is carried out on a commercial PC with the Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz. For the timing the built-in 'timeit'-function in MATLAB [25] is used, averaged over 10 identical simulation runs.
It is important to note, that though these considerations are a good indication for the computational performance, the implemented solvers are not fully optimized and thus the exact results have to be viewed carefully.

5.1.5 Influence of the time step

To study the influence of the time step on the criteria presented in the sections above, otherwise identical simulation runs are carried out with different time steps.
Thereby the sum of the currently considered error $e_\star$ is saved per simulation run and averaged by the number of time steps N, which yields:

$$e_{\text{avg}} = \frac{1}{N} \sum_{t=0}^{N} \|e_\star\| \,. \tag{5.5}$$

This yields a relation between said error and the chosen time step. It is important to note, that Equation (5.5) yields a global error, which naturally differs from the local error arising with each time step.
An approximation for this relation can be obtained from a so-called order-plot. A simple polynomial relation of the form

$$e_{\text{avg}} \sim \Delta t^k \tag{5.6}$$

can be detected by using a double logarithmic scale and identifying the slope of the resulting line, which then correspond to the exponent k. In some instances not the whole range of the plot can be used, as for small time steps rounding errors often dominate and for larger time steps instabilities may arise. In that case, the line is then only fitted into the linear part of the plot.
For obtaining the order-plots the following equidistant range for the time steps is chosen subsequently:

$$\Delta t = \begin{bmatrix} 0.001\text{s} & 0.002\text{s} & \dots & 0.2\text{s} \end{bmatrix} . \tag{5.7}$$

5.1.6 Reference simulation

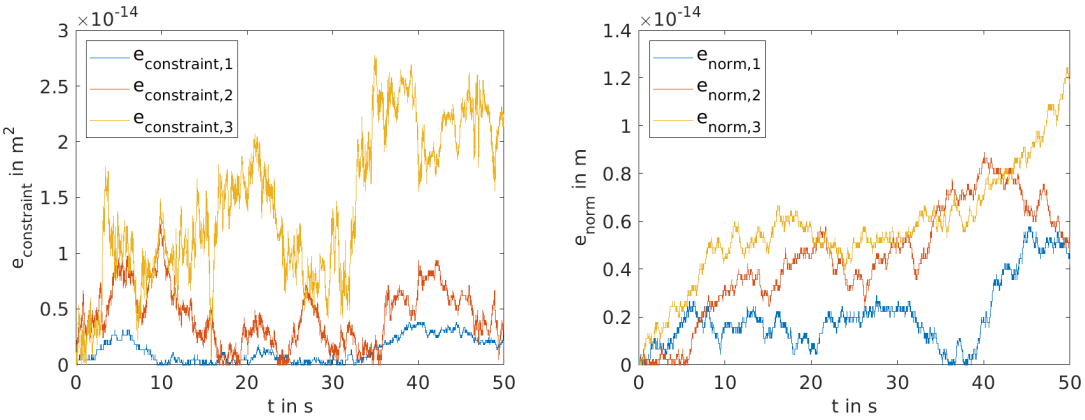
For comparison a reference simulation using a classical method with minimal coordinates – which are naturally the relative rotation angles – is implemented. Thereby the chain structure is used to recursively calculate the positions and orientations of the links, as well as the respective velocities and the local Jacobi matrices, relating the movements of the rotational joints to the absolute velocities of the links. The EOM are then obtained by the application of the projective Newton-Euler equations, where the Jacobi matrices are used to project the general rigid body motion onto the minimal coordinates. This algorithm is described in [8, § 6.1, p. 102–104].
The implicit midpoint rule is then applied for solving the resulting second order ODE. As the state space variables are minimal coordinates and thus compatible with the constraints, naturally the structure of the mechanism is conserved for all time. Thus the criteria of interest regarding the reference simulation are the energy conservation and

the computational performance in comparison to the other approaches.
For the reference model, the mono- and multi-articular springs are not implemented. Furthermore it is worth noting, that due to the use of minimal coordinates, the implementation of linear springs is rather complicated compared to the other methods using non-minimal coordinates respectively attitude vectors.

5.2 Results

5.2.1 Symplecticness

For evaluating if the time integration schemes conserve the geometric structure, the standard simulation run specified in Section 5.1.1 is used.
Both methods show excellent symplectic properties, as the maximum error lies in the region of 10^{-14}, which can be explained by rounding errors due to the floating point precision, see Figure 5.1. This result is logical for the variational integrator, as by



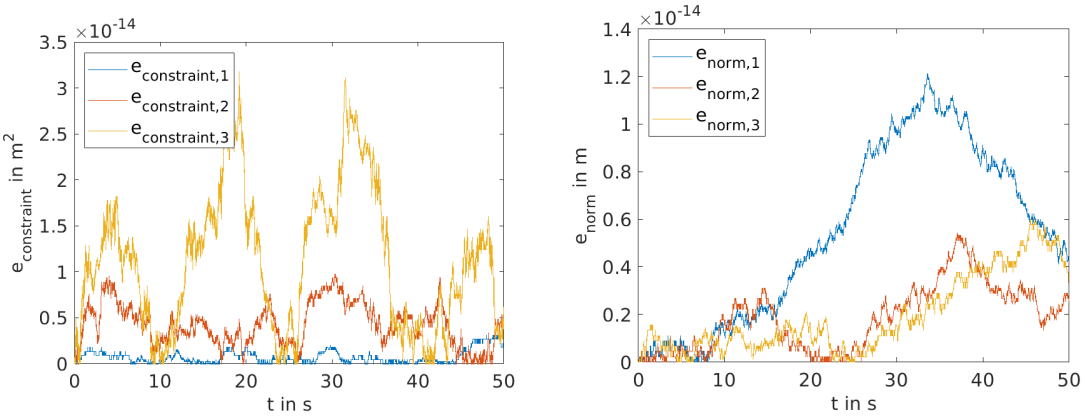
(a) energy stable time integration (b) Variational integrator

Figure 5.1: Constraint error for the standard simulation run under gravitational influence, with $t = 50$s and $\Delta t = 0.01$s

definition its design is symplectic. For the energy stable time integration, the result can be explained due to the linearity of the constraint matrix C_q, which leads to an exact approximation of its value using the implicit midpoint rule. As the elastic elements have no influence on C_q, it also makes sense, that the addition of springs to the mechanism does not change the structure preserving behavior of the integrator, see Figure 5.2.
An increase in step-size also has no influence on the structure preservation.

5.2.2 Energy conservation under gravitational influence

As expected, when using the energy stable time integration method described in Chapter 3 on the serial manipulator without elastic elements, the energy of the system is conserved very well, especially considering the simulation is carried out for a total time of 50s. The maximum deviation lies in the region 10^{-14}, which again can be explained by the floating point precision, see Figure 5.3a.
It is worth noting, that an increase in step size also has no influence on the energy



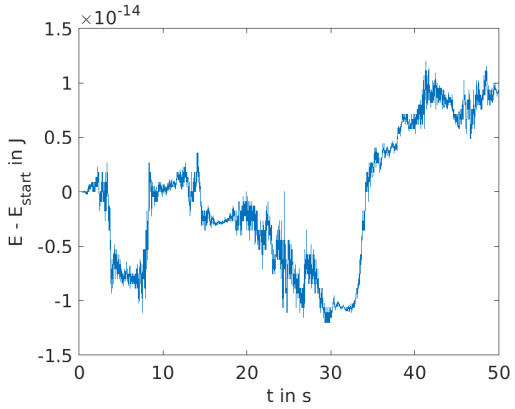
(a) energy stable time integration (b) Variational integrator

Figure 5.2: Constraint error for the standard simulation run with mono-articular springs, with $t = 50$s and $\Delta t = 0.01$s

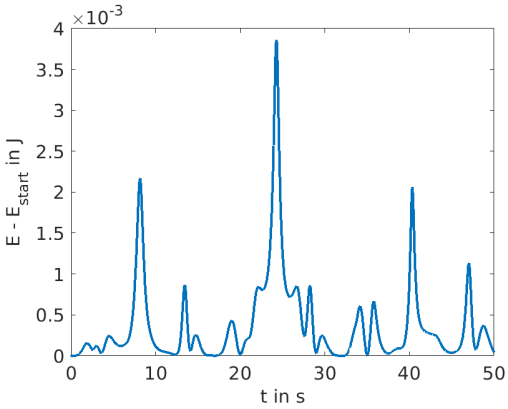
error. This excellent energy behavior can be explained by the exact approximation of the constant gradient of the potential energy – similar to the structure preserving properties explained in the previous Section 5.2.1.
In contrast – when using the variational integrator on the same mechanical system – the maximum energy error is significantly higher in the range of 10^{-3}. The error occurs in spikes throughout the simulation run, but appears to be bounded, see Figure 5.3b.
Carrying out the standard simulation with the classical reference model described in Section 5.1.6, yields a bounded energy error in the range of 10^{-4}, see Figure 5.3c.
When using the variational integrator or the classical method, the error depends on the step-size. The linear section of the logarithmic order-plots for both methods show, that the energy error scales with Δt^2, see Figure 5.4a and 5.4b. Thereby the linear parts used to calculate the slope are marked with a dashed line. As for both instances the implicit midpoint rule – a second order method – is used, this is logical.
For the classical reference model an instability of the global energy error above a time step of $\Delta t = 0.01$s can be observed, marked by the ellipse in Figure 5.4b.
For the variational integrator one can also observe, that the section where the global energy error is dominated by the order of the implemented time integration scheme, goes approximately up to a time step of $\Delta t = 0.01$s.

5.2.3 Energy conservation with elastic elements

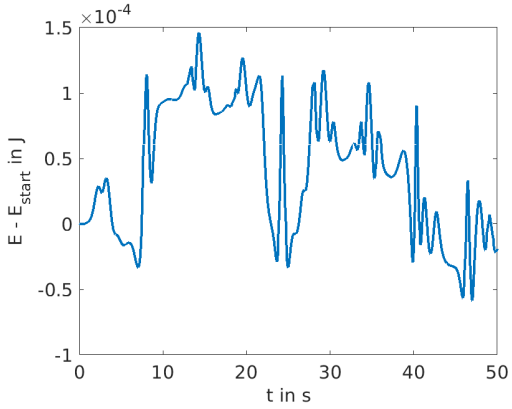
It can be observed, that adding the elastic elements to the mechanism fundamentally changes the energy error for the energy stable implementation, where the maximum error now ranges around 10^{-5}, however is still bounded, see Figure 5.5a. The reason thereby is an inexact approximation of the now non-linear gradient of the potential energy.
As the implementation allows for an easy substitution of this approximation by a higher order method, the same simulation is carried out using the Gauss-Legendre quadrature rule with 3 collocation points, leading to a significantly lower energy error, which now lies in the range of 10^{-13}, see Figure 5.6. As now the energy error for both methods depends on the chosen time step, we are interested in the order-plots. For the variational integrator, the plot – depicted in Figure 5.7b – yields approximately the same results as



(a) Energy stable time integration

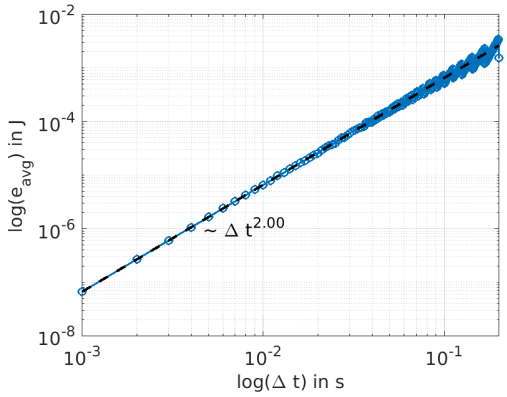


(b) Variational integrator

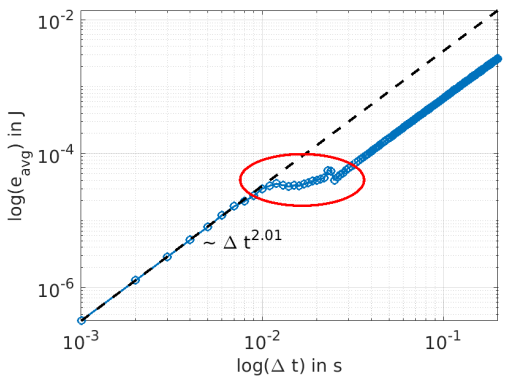


(c) Reference model

Figure 5.3: Energy error for the standard simulation run under gravitational influence, with $t = 50$s and $\Delta t = 0.01$s

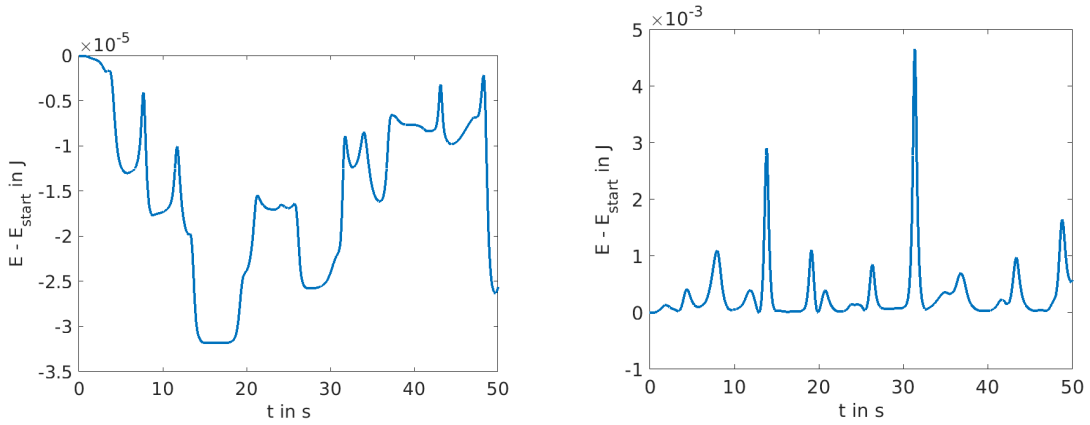


(a) Variational integrator



(b) Reference model

Figure 5.4: Order plots for the average energy error, using the standard simulation run under gravitational influence, with $t = 10s$



(a) Energy stable time integration (b) Variational integrator

Figure 5.5: Energy error for the standard simulation run with mono-articular springs, with $t = 50$s and $\Delta t = 0.01$s

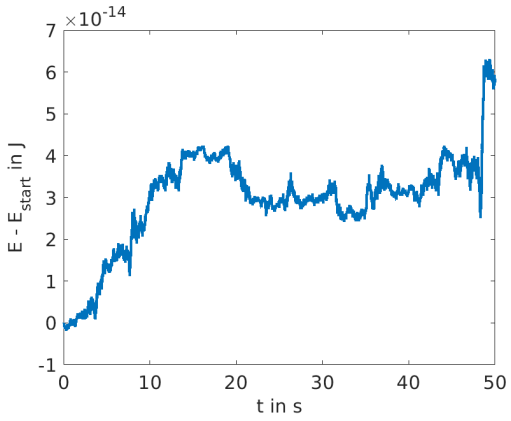
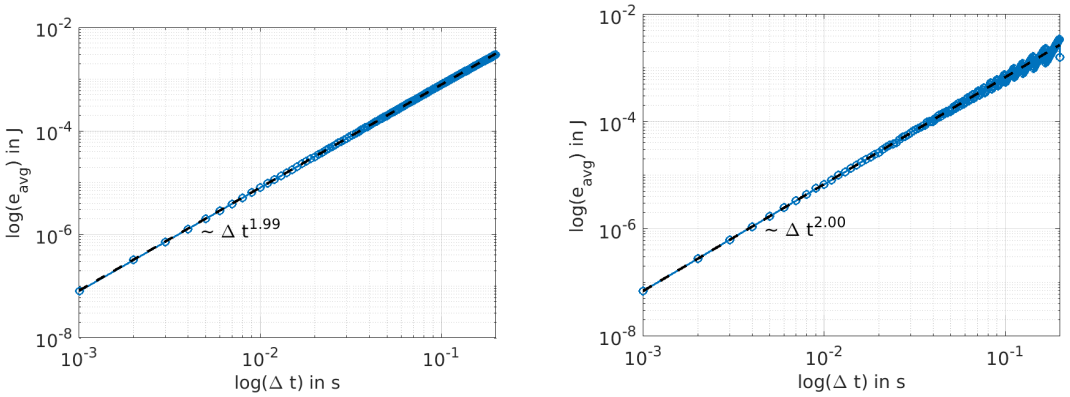


Figure 5.6: Energy error for a standard simulation run with mono-articular springs, using the updated energy stable method with Gauss-Integration (3 points), with $t = 50$s and $\Delta t = 0.01$s

for the simulation run under gravitational influence only, studied in the previous section. This makes sense, as due to the use of local coordinates, the approach yields a non-linear gradient of the potential energy with and without the presence of elastic elements.
The order-plot for the energy stable approach using the implicit midpoint rule, also yields a scaling of Δt^2, see Figure 5.7a. This is again logical, as the approximation via midpoint rule is a second order method.
Updating the time integration scheme by using the mentioned Gauss quadrature with 3 points for the non-linear potential energy of the springs leads to a relation of Δt^6 for the energy error, see 5.8. This relation is expected, as the implemented Gauss quadrature yields a 6th order method. Thereby one can observe that the linear part of the plot is bounded by rounding errors for small time steps, as the energy error nears the range of floating point precision.
Furthermore, it can be observed, that for the energy stable time integration, the method behaves consistently for the whole proposed time step range, defined in Equation (5.7).



(a) Energy stable time integration (b) Variational integrator

Figure 5.7: Order plots for the average energy error, using the standard simulation run with mono-articular springs, with $t = 10$s

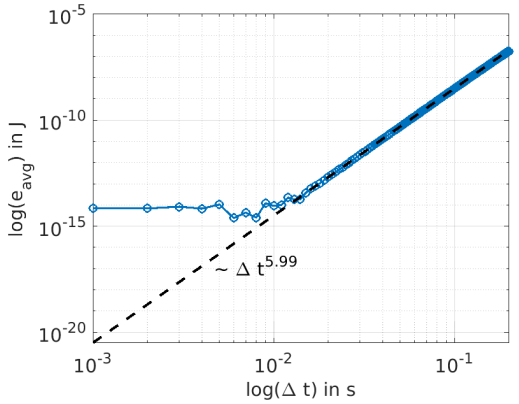


Figure 5.8: Order plot for for the average energy error, using the standard simulation run with elastic elements and the updated energy stable method with Gauss-integration, with $t = 10s$

5.2.4 Computational performance

The absolute duration for carrying out the time integration of the standard simulation runs with respectively without the presence of elastic elements is subsequently summarized in Table 5.2 and 5.3.
As the energy stable approach already has excellent conserving properties when the simulation is run only under gravitational influence, the higher order integration method is not studied in that case. Furthermore as elastic elements are not implemented in the reference model, for this approach only the case without mono-articular springs is considered.

	$t_{\text{computation}}$ in s	t_{realtime} in %
energy stable (midpoint rule)	2.35	4.70
Variational integrator	140.42	280.84
Reference model	47.20	90.40

Table 5.2: Computational performance of the solvers only under gravitational influence, averaged over 10 simulation runs, with $t = 50$s and $\Delta t = 0.01$s

	$t_{\text{computation}}$ in s	t_{realtime} in %
energy stable (midpoint rule)	3.36	6.72
energy stable (Gauss-integration, 3 points)	3.53	7.06
Variational integrator	147.68	295.36

Table 5.3: Computational performance of solvers in the presence of elastic elements, averaged over 10 simulation runs, with $t = 50$s and $\Delta t = 0.01$s

5.3 Discussion

5.3.1 Use of non-minimal coordinates

A significant difference between the presented algorithms and the classical approach is the use of non-minimal coordinates. As thus structure-preservation becomes an issue, the behavior of the time integration schemes concerning this topic was studied in Section 5.2.1. The results yield, that both approaches do not violate the constraints and thus the choice of coordinates does not lead to a disadvantage regarding a possible drift-off phenomenon.
The constant mass-matrix of the energy stable approach, as described in Section 3.4.2, is a direct consequence of the use of non-minimal coordinates, which is beneficial for the energy conserving behavior of the integration scheme, which will be further discussed in the next Section 5.3.2.
Furthermore, the description in positional coordinates instead of using the rotation-angles, allows for a rather straightforward integration of the additional linear elastic elements. Especially the use of an inertial reference frame facilitates the implementation of more complex multi-articular systems, as once the fastening points are expressed in the chosen coordinates, they can be almost arbitrarily interconnected.
The approach using the variational integrator can be easily expanded to 3d, as the description in local coordinates allow the use of a constant moment of inertia – which in the 3d case becomes a tensor instead of a scalar – in body coordinates. This is a significant advantage over the described energy stable approach, as there the inertial reference frame does not allow a straight forward expansion to the 3d case.

5.3.2 Energy conservation

Especially for more complex systems with different form of energy-storage, conserving the property of energy-conservation is crucial.
The approach using a variational integrator conserves the energy well with no significant difference between a simulation with and without elastic elements up to a step size of $\Delta t = 0.01$s. In both cases the maximum error ranges around 10^{-3} for the standard time step and scales with Δt^2. Thus – especially when implementing a higher order method – this approach is feasible for long-time simulation of the proposed mechanical system.
However compared to the classical and the energy stable approach, the energy error is significantly higher. With a maximum error of around 10^{-5}, the described energy stable method performs best in that regard. A modification with a higher order integration method, yields excellent results. It is worth noting, that – due to the specific structure of this approach – one can only us this modification for the non-linear part of the potential energy. This leads to a better computational performance, subsequently discussed in the next Section 5.3.3.
Furthermore the energy stable method is in all considered scenarios consistent concerning the global energy error, as a decrease in step size always decreases said error in the considered time step range – going up to $\Delta t = 0.2$s – which is an important result.

5.3.3 Computational performance

As a last aspect, the computational performance is discussed. When looking at the relation between the duration of the simulation and computation, it becomes clear, that the energy stable approach is the fastest by a rather large margin.
The elastic elements naturally increase the computation time, however the integration scheme is applicable for real-time calculations for both scenarios. Furthermore, the higher order integration methods influence on the performance is relatively small. The most time-consuming scenario yields $t_{\text{realtime}} = 7.06\%$, which makes it applicable for real-time calculations.
In contrast the approach using variational integrator is the slowest, almost requiring triple the simulation time for the computations, which is a strong indication, that this method is not applicable for real-time calculations. The result for the reference model yield $t_{\text{realtime}} = 90.40\%$ in the case of only gravitational influence, which is sorted in between the two other approaches regarding its performance.

5.4 Summary and outlook

After introductory remarks about the forward-dynamics problem statement in Chapter 1, the system considered in this thesis – a planar, serial manipulator with rotational joints and linear, elastic elements between the links – was described in Chapter 2. Thereby a background on elastic elements in robotics was given in Section 2.2.1.
In the next part a comprehensive overview over conserved quantities in Hamiltonian systems was given. Thereby the energy stable method was used to treat the aspect of conserving the Hamiltonian, see Chapter 3. Afterwards the idea of symplectic integrators was introduced on the basis of a variational integrator constructed on Lie Groups, see Chapter 4.
In those two chapters, the respective implementation of the presented serial manipulator

was described in the Sections 3.4 and 4.4. Thereby computations where carried out in MATLAB [25].
In the last Chapter 5.2, first the evaluation criteria where determined, see Section 5.1. This provided the basis for the following discussion of the presented time-integration schemes in Section 5.3.
The described energy stable method was shown to be have excellent properties for studying the effects of mono- and multi-articular springs on such a system, with its structure and energy-preserving behavior as well as real-time capabilities. Due to its excellent energy conservation, this approach is could be interesting for modeling and possibly realizing passivity-based control strategies. Furthermore, the choice of non-minimal coordinates proved advantageous, as it facilitated the implementation of the linear elastic elements.
For the energy stable method the challenge of implementing this approach for the 3d case arises. Thereby also the effects on the simulation properties are of interest.

List of Figures

2.1 Geometry of the serial manipulator 4

2.2 Attachment of the springs 5

2.3 Fastening for multi-articular springs 6

2.4 Vector chain for calculating the length of the springs 7

3.1 Time integration schemes 11

3.2 Linear Transformation 15

4.1 Screw motion of a point $\mathbf{p}$ 22

5.1 Constraint error for the standard simulation run under gravitational influence, with $t = 50$s and $\Delta t = 0.01$s 34

5.2 Constraint error for the standard simulation run with mono-articular springs, with $t = 50$s and $\Delta t = 0.01$s 35

5.3 Energy error for the standard simulation run under gravitational influence, with $t = 50$s and $\Delta t = 0.01$s 36

5.4 Order plots for the average energy error, using the standard simulation run under gravitational influence, with $t = 10s$ 36

5.5 Energy error for the standard simulation run with mono-articular springs, with $t = 50$s and $\Delta t = 0.01$s 37

5.6 Energy error for a standard simulation run with mono-articular springs, using the updated energy stable method with Gauss-Integration (3 points), with $t = 50$s and $\Delta t = 0.01$s 37

5.7 Order plots for the average energy error, using the standard simulation run with mono-articular springs, with $t = 10$s 38

5.8 Order plot for for the average energy error, using the standard simulation run with elastic elements and the updated energy stable method with Gauss-integration, with $t = 10s$ 38

List of Tables

5.1 Parameters for the standard simulation run 32

5.2 Computational performance of the solvers only under gravitational influence, averaged over 10 simulation runs, with $t = 50$s and $\Delta t = 0.01$s . . . 39

5.3 Computational performance of solvers in the presence of elastic elements, averaged over 10 simulation runs, with $t = 50$s and $\Delta t = 0.01$s 39

References

[1] David Betounes. *Differential Equations: Theory and Applications*. Springer New York, 2010. ISBN 9781441911636. doi: 10.1007/978-1-4419-1163-6. URL http://dx.doi.org/10.1007/978-1-4419-1163-6.

[2] Priyaranjan Biswal and Prases K. Mohanty. Development of quadruped walking robots: A review. *Ain Shams Engineering Journal*, 12(2):2017–2031, June 2021. ISSN 2090-4479. doi: 10.1016/j.asej.2020.11.005. URL http://dx.doi.org/10.1016/j.asej.2020.11.005.

[3] R. Blickhan. The spring-mass model for running and hopping. *Journal of Biomechanics*, 22(11–12):1217–1227, January 1989. ISSN 0021-9290. doi: 10.1016/0021-9290(89)90224-8. URL http://dx.doi.org/10.1016/0021-9290(89)90224-8.

[4] Peter Corke. *Robotics, Vision and Control*. Springer Berlin Heidelberg, 2011. ISBN 9783642201448. doi: 10.1007/978-3-642-20144-8. URL http://dx.doi.org/10.1007/978-3-642-20144-8.

[5] Herbert Egger, Oliver Habrich, and Vsevolod Shashkov. On the energy stable approximation of hamiltonian and gradient systems. *Computational Methods in Applied Mathematics*, 21(2):335–349, dec 2020. doi: 10.1515/cmam-2020-0025.

[6] Edda Eich-Soellner and Claus Führer. *Numerical Methods in Multibody Dynamics*. Vieweg+Teubner Verlag, 1998. doi: 10.1007/978-3-663-09828-7.

[7] Kenth Engø. On the bch-formula in so(3). *BIT Numerical Mathematics*, 41(3):629–632, Jun 2001. ISSN 1572-9125. doi: 10.1023/A:1021979515229. URL https://doi.org/10.1023/A:1021979515229.

[8] Roy Featherstone. *Rigid Body Dynamics Algorithms*. Springer US, 2008. ISBN 9780387743141. doi: 10.1007/978-1-4899-7560-7. URL http://dx.doi.org/10.1007/978-1-4899-7560-7.

[9] Bruce A Finlayson and Laurence Edward Scriven. The method of weighted residuals—a review. *Appl. Mech. Rev*, 19(9):735–748, 1966.

[10] Jean Gallier. *Basics of Classical Lie Groups: The Exponential Map, Lie Groups, and Lie Algebras*, pages 367–414. Springer New York, New York, NY, 2001. ISBN 978-1-4613-0137-0. doi: 10.1007/978-1-4613-0137-0_14. URL https://doi.org/10.1007/978-1-4613-0137-0_14.

[11] Gianluca Garofalo, Christian Ott, and Alin Albu-Schaffer. On the closed form computation of the dynamic matrices and their differentiations. In *2013 IEEE/RSJ*

International Conference on Intelligent Robots and Systems. IEEE, November 2013. doi: 10.1109/iros.2013.6696688. URL http://dx.doi.org/10.1109/IROS.2013.6696688.

[12] Claude Gignoux and Bernard Silvestre-Brac. *Solved Problems in Lagrangian and Hamiltonian Mechanics*. Springer Netherlands, 2009. ISBN 9789048123933. doi: 10.1007/978-90-481-2393-3. URL http://dx.doi.org/10.1007/978-90-481-2393-3.

[13] Ernst Hairer. Long-time energy conservation of numerical integrators. *Lecture Notes Ser. FoCM Santander 2005*, 331, 06 2006. doi: 10.1017/CBO9780511721571.005.

[14] Maximilian Herrmann, Dhananjay Tiwari, Paul Kotyczka, and Ravi Banavar. Coordinate-invariant modeling and control of a three-dof robot manipulator. *IFAC-PapersOnLine*, 54:230–236, 01 2021. doi: 10.1016/j.ifacol.2021.11.083.

[15] Yuta Ishikawa, Hiroyuki Nabae, Gen Endo, and Koichi Suzumori. Stability analysis of multi-serial-link mechanism driven by antagonistic multiarticular artificial muscles. *ROBOMECH Journal*, 9(1), May 2022. ISSN 2197-4225. doi: 10.1186/s40648-022-00226-8. URL http://dx.doi.org/10.1186/s40648-022-00226-8.

[16] L.D. LANDAU and E.M. LIFSHITZ. *Mechanics (Third Edition*. Butterworth-Heinemann, Oxford, third edition edition, 1976. ISBN 978-0-7506-2896-9. doi: https://doi.org/10.1016/C2009-0-25569-3. URL https://www.sciencedirect.com/book/9780750628969.

[17] Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. Lie group variational integrators for the full body problem, 2005.

[18] Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. *Global Formulations of Lagrangian and Hamiltonian Dynamics on Manifolds*. Springer International Publishing, 2018. doi: 10.1007/978-3-319-56953-6.

[19] Benedict Leimkuhler and Sebastian Reich. *Simulating Hamiltonian Dynamics*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.

[20] Richard M. Murray, S. Shankar Sastry, and Li Zexiang. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., USA, 1st edition, 1994. ISBN 0849379814.

[21] G.A. Pratt and M.M. Williamson. Series elastic actuators. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, IROS-95. IEEE Comput. Soc. Press. doi: 10.1109/iros.1995.525827. URL http://dx.doi.org/10.1109/IROS.1995.525827.

[22] Felix Ruppert and Alexander Badri-Spröwitz. Series elastic behavior of biarticular muscle-tendon structure in a robotic leg. *Frontiers in Neurorobotics*, 13, August 2019. ISSN 1662-5218. doi: 10.3389/fnbot.2019.00064. URL http://dx.doi.org/10.3389/fnbot.2019.00064.

[23] Bruno Siciliano and Oussama Khatib, editors. *Springer Handbook of Robotics*. Springer International Publishing, 2016. doi: 10.1007/978-3-319-32552-1.

[24] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer New York, 2002. ISBN 9780387217383. doi: 10.1007/978-0-387-21738-3. URL http://dx.doi.org/10.1007/978-0-387-21738-3.

[25] The MathWorks Inc. Matlab version: 9.13.0 (r2022b), 2022. URL https://www.mathworks.com.