University of Rijeka

Faculty of Civil Engineering

Jan Tomec

Stable Long-Term Numerical Integration of Motion of Beams and Their Interaction

Doctoral Thesis

Rijeka

2024

University of Rijeka

Faculty of Civil Engineering

Jan Tomec

Stable Long-Term Numerical Integration of Motion of Beams and Their Interaction

Doctoral Thesis

Supervisor: Gordan Jelenić

Co-supervisor: Sara Grbčić Erdelj

Rijeka

2024

Supervisor: prof. Gordan Jelenić

Co-supervisor: doc. Sara Grbčić Erdelj

Thesis defended in Rijeka, _____ in front of the dissertation committee:

1. doc. Edita Papa Dukić, University of Rijeka, Faculty of Civil Engineering
2. prof. Olivier Brüls, University of Liege, Faculty of Applied Sciences
3. prof. Johannes Gerstmayr, University of Innsbruck, Institute of Mechatronics

Acknowledgments

Očetu v spomin.

Anji za brezmejno ljubezen.

Gordanu za izvrsno vođenje i potporu.

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 860124.

Abstract

This thesis explores the advancement and refinement of numerical models for beams within the finite element method, a critical tool widely adopted across industries for simulations. Specifically, my research has been focused on the dynamics of beams and their conservation properties when considered using geometrically non-linear models. The second research question concerns contacts between beams. Both of these are significant as model reduction techniques, which allow faster and more complex simulations of our physical environment.

The core objectives of this work are to investigate the conservation of mechanical properties of beams for long-term stable formulations and to develop a robust method for understanding and simulating beam-to-beam contact mechanics across diverse scenarios. Our methodology blends theoretical analysis with numerical simulations, focusing on geometrically-exact beam theory and beam-to-beam contacts, supported by the development of open-source research code.

As a result, I have developed a novel dynamic beam finite element, which is capable of conserving linear and angular momentum and almost conserve energy. This element features an advanced interpolation of position and rotation on SE(3) group, which allows the element to be objective and locking-free. The conservation properties have been achieved using the Lie midpoint rule for time integration and an independent velocity field.

Additionally, I have devised several contact formulations for line-to-line contacts, notably applying the existing mortar method to beams, and introducing a novel, unbiased contact formulation. Both of these methods have been combined with the Lagrange-multiplier and the penalty method and compared using benchmark tests.

The thesis contributes to the field by offering novel approaches in beam dynamics and contacts, characterised by robustness and stability. I have discussed their advantages and drawbacks in detail and point out potential areas for further investigation.

Keywords: Finite element method, Geometrically-exact beam theory, Energy–momentum conservation/decay, Mortar method, Lie groups

Sažetak

Ova disertacija istražuje napredak i poboljšanje numeričkog modela grede u okviru metode konačnih elemenata, ključnog alata koji je široko prihvaćen u industriji za simulacije. Konkretno, naše istraživanje je bilo usmjereno na dinamiku greda i sačuvanja mehaničkih svojstva kada se ti razmatraju korištenjem geometrijski nelinearnih modela. Drugo istraživačko pitanje odnosi se na kontakte između greda. Oba su značajna kao tehnike redukcije modela, koje omogućavaju brže i kompleksnije simulacije.

Osnovni ciljevi ovog rada su istražiti sačuvanje mehaničkih svojstva greda kod dugoročno stabilne formulacije i razviti robustnu metodu za razumijevanje i simuliranje mehanike kontakta grede s gredom u različitim scenarijima. Naša metodologija kombinira teorijsku analizu s numeričkim simulacijama, fokusirajući se na teoriju geometrijski točnih greda i kontakte grede s gredom, uz podršku razvoja otvorenog istraživačkog koda.

Kao rezultat, razvili smo novi dinamički konačni element za grede, koji je sposoban sačuvati linearnu i kutnu količinu kretanja i gotovo sačuvati energiju. Ovaj element sadrži naprednu interpolaciju pozicije i rotacije na SE(3) grupi, što elementu omogućava da bude objektivan i bez *shear-locking*-a. Svojstva sačuvanja postignuta su korištenjem Liejevog srednjeg pravila za integraciju vremena i neovisnog polja brzine.

Osim toga, osmislili smo nekoliko formulacija kontakta za kontakte tipa linija s linijom, naime primjenjujući postojeću mortar metodu na grede, te uvođenjem nove, nepristrane formulacije kontakta. Obe ove metode kombinirane su s Lagrangeovim množiteljima i *penalty* metodom te uspoređene brojnim testovima.

Disertacija doprinosi području nudeći nove pristupe u dinamici greda i kontakta, karakterizirane robustnošću i stabilnošću. Detaljno smo raspravljali o njihovim prednostima i nedostacima te istaknuli potencijalna područja za daljnje istraživanje.

Ključne riječi: Metoda konačnih elemenata, Geometrijski točna teorija greda, Očuvanje/propadanje energije-količine kretanja, Mortar metoda, Liejeve grupe

Contents

Chapter 1

Introduction

When the project started, I could hardly foresee where the research would lead us. Only when I began delving into the field of beam mechanics did I realise the elegance of its theory and the exciting new developments using mathematical groups. And, as it often happens, my journey has not been straightforward but consisted of several fundamental twists in my thinking. Only at the end does one realise that even dead-ends significantly contribute to the final result and are thus a necessary part of any scientific exploration. Nevertheless, the research has been fruitful, manifesting itself in three high-quality papers, all published in different tier one journals; discussed also in this manuscript.

My research has been financed through a European training network program with an acronym THREAD. This project, from its inception, has been oriented towards real-life industrial problems and positioned as a bridge between cutting-edge academic research and state-of-the-art industrial technology. This pragmatism has significantly influenced my work, not only by predetermining the general topic but also through hands-on secondment at our industrial partner, SPACE-SI, providing me with insights into potential application details and requirements.

Model reduction is indeed very interesting to our industrial partners, as it is necessary for any more complex analysis in mechanics. Our protagonist, beam theory, provides a prime example. It condenses the generalised three-dimensional system into a one-dimensional problem and can be applied to model a wide range of slender structures, including ropes, cables, wire harnesses, textile yarns, and hoses. Complications arise when one decides to consider geometrical or material non-linearities or environmental constraints. Different approaches to dealing with these complications pave the way for a variety of formulations at all levels, from nuances in the fundamental theory to details in the numerical implementation.

My research has been focused on the numerical implementation of a particular beam the-

ory in the finite element method, known as geometrically-exact beam theory. Although many successful implementations already exist, I have been interested in determining whether it is possible to produce a finite element in combination with an appropriate time integration scheme, which would allow linear and angular momentum and mechanical energy conservation, all using only group operations to ensure that the solution remains on the manifold. Understanding the fundamentals of group theory is crucial for developing a stable and quickly converging method by introducing the minimal amount of numerical bias into the algorithm, which creeps in through the spatial or the temporal discretisation. Furthermore, in dynamics, energy-momentum preserving algorithms are known to produce long-term stable results which I wanted to apply also to the non-linear geometrically-exact beam theory.

I aimed to pair these beam dynamics with an appropriately advanced contact formulation. This had led me to explore various formulations for general contact mechanics, and I came across the mortar method. Recognising that an appropriate mortar formulation for beam-to-beam contacts was still missing, I decided to fill that void. The original idea was to integrate the dynamic and contact formulations in a few comprehensive examples; however, I fell just shy of this goal due to external factors. Nevertheless, the developed contact and dynamics models, although not combined, represent state-of-the-art research. The findings have not only unveiled novel insights into the intricacies of beam mechanics in the finite element framework but also opened the door for further exploration in this dynamic and evolving field.

1.1 Objectives

The overarching goal of this PhD research has been to improve the existing techniques for numerical modelling of beam-like structures for practical applications. To achieve this, a set of specific objectives has been delineated, each contributing to the comprehensive understanding of beam dynamics in a three-dimensional environment. In this section, I outline the key objectives that drive my inquiry:

- investigation of the conservation of mechanical properties on Lie groups for beams,
- application of such properties to a time integrator and a corresponding dynamic beam element formulation,
- assessment of the practical characteristics of such formulation in terms of stability and robustness,
- combination of the formulation with objective beam interpolation,

- understanding of the mechanics of unilateral or bilateral constraints on beams in the form of contact,
- development of a suitable robust algorithm for handling contacts whose geometry cannot be predicted in advance,
- integration of such algorithm within the finite element framework working for the most common types of existing beam elements.

By addressing these objectives, this thesis aims to facilitate the adoption of complex numerical simulations in industrial applications, advancing the current state of knowledge in the field and creating opportunities for future research and applications.

1.2 Overview

This thesis delves into two aspects of beam mechanics: dynamics and interactions. Although they can be treated separately, I will try to talk about them interleaved to follow the initial (unachieved) goal of the project, to combine them into dynamic interactions. This will perhaps allow for easier understanding of some of the motivations behind the specific requirements in formulations and design decisions.

In Chapter 2 I overview the existing literature on the topic to give a reader the sense of the current state-of-the-art. Theoretical and numerical developments are elaborated in Chapter 3. Chapter 4 presents the resulting formulations' performance against a suite of benchmark test cases. The results of the research are critically overviewed and contextualised in Chapter 5. Finally, a conclusion is drawn in Chapter 6

Chapter 2

Background: State-of-the-Art

The goal is to implement theoretical developments within the finite element method for use in numerical analysis. In the following, I highlight some significant recent developments that have led to the current state-of-the-art. This discussion aims to provide a better understanding of how the research presented in this thesis relates to the existing context of scientific research in beam mechanics.

2.1 Finite element approaches in geometrically-exact beam theory

Geometrically-exact beam theory is a mathematical and computational framework used to model the deformation and behaviour of slender beams in a way that accounts for geometric non-linearity. It provides a more accurate representation of the geometry and deformation of beams compared to linear beam theories, which assume small deformations.

- **Large deformations:** The theory allows for the consideration of large deformations, where the geometric non-linearity becomes significant.
- **Rotations and displacements:** It accounts for both translational and rotational degrees of freedom, acknowledging that as the beam deforms, cross-sections may rotate as well as translate.
- **Shear deformation:** It includes the effects of shear deformation, recognizing that as the beam deforms, cross-sections may undergo shearing.
- **Small strains:** The assumption of small strains is often used. The theory itself allows the extension to finite strains. However, these are usually combined with non-linear material

models.

This theory is particularly useful in situations where the deformations of the beam are substantial, and the assumptions of linear beam theories, such as Euler-Bernoulli beam theory or Timoshenko beam theory, are not valid. It has been first derived by Simo in [Simo, 1985] building upon the work of his predecessors, for example the two-dimensional case developed by Reissner in [Reissner, 1972].

A beam is geometrically treated as a curve in a three-dimensional space, where each point on the line has a two-dimensional (planar) shape, a cross-section, attached to it. This enables a one-dimensional parametric description of the entire volume of the beam. The beam's motion is separated into the translation of the centreline and the rotation of the cross-section, which fully describe all possible deformation states - tension, bending, shearing, and torsion. However, here also lies the main issue when trying to implement this model numerically. The rotation field belongs to a special orthogonal group, SO(3), which is part of the Lie group framework. Being a Lie group means that only group operations are allowed on its members, and thus, commutativity does not generally apply.

The first finite element implementation of the geometrically-exact beam theory has been developed by Simo and Vu-Quoc in [Simo and Vu-Quoc, 1986] for a quasi-static case and later for a dynamic case in [Simo and Vu-Quoc, 1988]. In it, the rotation field is not interpolated directly; the authors acknowledge that the total rotation is only required at the integration points and so prefer to interpolate the rotation increment which is in the (linear) tangent space and thus simple to interpolate, for example, using Lagrange polynomials. Rotations in the integration points are then updated using these interpolated values, which are mapped from the rotation vector to the rotation matrix using the exponential map. A detailed discussion on the mapping relations for different frames of rotation vector has been conducted a few years prior in [Argyris, 1982].

In [Borri and Bottasso, 1994a] and [Borri and Bottasso, 1994b] Borri and Bottasso propose a new helicoidal interpolation for beam finite elements based on constant curvature. They highlight the invariance properties of such interpolation that make it particularly attractive for non-linear analysis.

Ibrahimbegović and Mamouri in [Ibrahimbegovic and Mamouri, 2002] conduct a thorough investigation into different options for interpolation of rotation, in particular the interpolation of iterative or incremental rotational updates and total rotations.

So far, the interpolation of beam has always featured a two field approach where position and rotation are treated separately. In [Bottasso and Borri, 1998] Bottasso and Borri show that the

entire beam mechanics can be transported to the group of rigid body motions SE(3) to provide a unified approach in considering translation and rotation. They utilise the left-invariant field defining the fixed-pole frame to define the beam equations and derive the weak form. They decide to avoid interpolating the beam itself and rather interpolate the incremental updates, which are in the tangent space. This allows an arbitrary-order finite element.

An important aspect of the interpolation is its objectivity and strain-invariance which is connected to numerical shear-locking of a finite element. These properties have been elusive for a non-linear finite elements incorporating Lie-group space, such as 3D rotation. In [Jelenić and Crisfield, 1999] Jelenić and Crisfield propose a method on how to interpolate rotations objectively using logarithmic map on SO(3) which led to an objective finite element.

In [Sonneville et al., 2014] Sonneville, Cardona and Brüls show that the beam mechanics can be transported to SE(3) to provide a unified approach in considering translation and rotation, similar to the work of Bottasso and Borri in [Bottasso and Borri, 1998]. Unlike the latter, Sonneville et al. decided to use the right-invariant field corresponding to the material frame (instead of fixed-pole) and to use 4 by 4 matrix representation of the SE(3) group instead of the 6 by 6 matrix representation of the adjoint of the SE(3) group. Finally, they have been able to derive an objective interpolation similar to that in [Jelenić and Crisfield, 1999] but on SE(3). They present only a two-node finite element, which features constant curvature and provides analytical expressions for all concerned fields.

Concerning the time integration, soon after [Simo and Vu-Quoc, 1988], Cardona and Géradin in [Cardona and Géradin, 1989] point out the shortcomings of the formulation for practical applications. They show that in order to time integrate constrained systems, the algorithmic damping at infinite frequency has significant importance. The formulation for time integration of rotations developed in [Simo and Vu-Quoc, 1988] is based on iterative updates and is restricted to the algorithm for which they have shown convergence in the paper. As an alternative, Cardona and Géradin suggest to use incremental rotations as the unknowns, meaning that the rotation increment from one time step to another is updated in the linear tangent space. They show how to adapt any standard algorithm to the SO(3) group using this technique. As a result, they are able to utilise the Hilber-Hughes-Taylor [Hilber et al., 1977] scheme with integrated damping of higher frequencies.

The integration of the dynamic equations presents a real challenge as most available techniques this far are either numerically unstable, or present undesirable high frequency oscillations of a purely numerical origin, as pointed out in [Bauchau et al., 1995]. This has led to a series of attempts to develop an energy/momentum conservative scheme, e.g. [Simo et al.,

1992, Bauchau et al., 1995, Simo et al., 1995, Bottasso and Borri, 1998, Ibrahimbegovic and Mamouri, 2002, Romero and Armero, 2002] and more recently [Zupan and Zupan, 2018], primarily using the interpolation of incremental motion. All these formulations aim to reduce the numerical instabilities using algorithmic conservation of mechanical properties or controlled decay. They all use a variant of the midpoint rule, adopted for Lie groups in some way or another.

2.2 Finite element approaches in beam-to-beam contact mechanics

Handling constraints is a huge field in mechanics and extends also to beam-to-beam contacts. Historically, these have been handled using a node-to-segment approach which is a simple method in which we check the penetration of any node in another mechanical body and apply repelling forces, if necessary, via Lagrange multipliers or using the penalty method [Wriggers and Zavarise, 2004]. This approach is not the most appropriate for beams as it requires a lot of nodes while the beams are more efficient if they are long and thus have fewer nodes. Thus we often encounter in beam-to-beam contact mechanics a Gauss-point-to-segment method [Chamekh et al., 2009, Meier et al., 2016], where, as the name suggest, we test integration points instead of nodes. This improves the contact detection ability as the number of Gauss points can be simply adjusted through integration order.

A novel approach is suggested by Durville in [Durville, 2005], where he builds an intermediate geometry to have a symmetrical treatment of the contact. In this way he does not require a master beam to define a gap function as it can be measured normally from the intermediate curve and thus preserve symmetry in the contact. The intermediate curve has to be constructed for each pair of elements that are close enough to present a possibility of a contact. This approach is one of the few capable of simulating a knot tightening as presented in [Durville, 2012]. Nevertheless, it does not seem to be widely adopted, probably because of its complexity in requiring a detection of contact regions, which can be a tricky programming job for a general contact formulation. This method also presents one of the first attempts at tackling the line-to-line contact, where a contact between two beams is continuous.

Within the framework of point-to-point contact model, a comparison between the penalty method and the Lagrange-multiplier method has been drafted in [Litewka, 2005]. In the penalty method, we consider a contact as a spring while in the Lagrange-multiplier method we compute the actual forces for the given contact conditions. The Lagrange-multiplier method solves the contact exactly and does not permit any penetration, while the penalty method always leaves

some residual penetration. In the author's opinion, the Lagrange-multiplier method is to be preferred, its major drawback being the complexity of the code.

Line-to-line contact has been again considered in [Chamekh et al., 2009], where the authors develop the Gauss-point-to-segment method. They evaluate the interface constraints in Gauss points and enforce them using the penalty method. The developed method is applied to a self-contact and is not generalised to multi-body scenarios.

Zavarise and De Lorenzis in [Zavarise and De Lorenzis, 2009] and [Zavarise and De Lorenzis, 2011] talk about enhancing a node-to-segment contact method. This method has been introduced by [Hughes et al., 1976] for purely geometric non-penetration conditions. In this method the contact conditions are enforced by preventing the nodes on one surface (slave) from penetrating the contact segments on the counterpart contact surface (master). As such it might be computationally efficient, but not the most appropriate for beam-to-beam contacts involving large deformations.

Advancements in the contact with friction are introduced in [Neto et al., 2014], where the authors present a new tangential gap function in order to consider the rolling of a beam and frictional moment. Each contact point is no longer modelled as the contact point on the centreline with the offset of the cross-section radius (if circular cross-section), but as a contact point on a rigid sphere with the centre on the centreline but also with orientation in compliance with the beam deformation. The idea can generally be adopted in all approaches.

In [Neto et al., 2015], the authors introduce a new test case for self-contact, that is experimentally supported. The article focuses on presenting a comprehensive literature overview and developing a new verification test. The authors precisely describe the pinball-searching technique for the contact search algorithm. It is based on spheres, centred at each element, which enables for the coarse search which then leads to the fine minimum gap search. The authors use a point-to-point contact model with slip and stick effect to model friction. They use accumulated tangential gap for differentiation between stick and slip regimes. The authors consider only quasi-static cases.

[Litewka, 2015a] continues the previous work from [Litewka, 2015b] by enhancing his frictionless three-point contact model for a case with friction. The three-point contact model is a version of the point-to-point model, where we add two extra contact points based on the geometry of a beam. Their purpose is to distribute the force to a larger area. The author describes how to apply friction to this method. He uses the Coulomb friction, where he distinguishes between the stick and the slip regimes. The frictional coefficient is, however, not velocity dependent, but relies on the elasto-plastic analogy. For this we need to introduce the tangential gap function. It

is set to 0 when the contact between two elements is first established and then it is updated with each time step as long as the elements are in contact. The frictional force is a linear function of the tangential gap and as long as it is smaller than the critical value ($F_T \leq \mu F_N$), the elements are in the stick regime. Once the gap becomes large enough and the condition is no longer fulfilled, the friction steps into the slipping regime. The author considers only quasi-static cases.

In [Meier et al., 2016], the authors compare line-to-line methods. They use Gauss-point-to-segment method - in this method, we enforce interface constraints in the quadrature points. They compare two different approaches for the penalty method. In the first, they introduce the penalty force at the definition of the constraints, thus obtaining the penalty method, while with the second approach they apply Lagrange parameter method for the constraints and insert nodal penalty forces instead of the unknown nodal Lagrange multipliers, thus getting the augmented Lagrange method. They do this to keep the system of equations smaller. Another argument in favour of the penalty method, presented by the authors, is the following scenario. If we compress two parallel beams, they should bounce back, but since the cross-sections are rigid, the result with the Lagrange parameters will not produce any movement as it does not allow any penetration. On the other hand, penalty method will allow some penetration, which will produce accelerations upon release. A flaw in this argument might be that the reaction in the penalty method is completely artificial and has no physical meaning. If there is the assumption of rigid cross-sections, the solution with no movement is more appropriate. If we wish to add movement due to cross-section compression, an appropriate solution would be to include Hertz contact pressure and deformation [Kawa and Litewka, 1999], which would then produce physically justified movement. In this article the authors also show, that a line-to-line formulation always possesses a unique solution and is thus applicable for any geometrical contact configuration.

Based on the description from [Meier et al., 2018], there are two main approaches in modelling non-friction contacts between beams (or between a beam and a rigid body). A point-to-point approach considers a discrete contact force at the closest point of the beams. This approach is simpler and only works for larger angles. A line-to-line approach assumes distributed contact forces along the beams, which works for all cases, even smaller angles. In [Meier et al., 2017] the authors developed all-angle beam contact (ABC) formulation, which is based on the combination of the before-mentioned ones. In that article they used Kirchhoff-Love elements. In [Meier et al., 2018] they now apply this theory on a Simo-Reissner beam. The displacement field is interpolated using the Hermite polynomials (using the tangent at the end points, similar to the simple Euler-Bernoulli beam), while the rotation field is interpolated using Lagrangian polynomials. Such approach was first introduced by [Litewka, 2007]. It is important, because in contact modelling,

one requires a C^1-continuous centreline to enable smooth kinematics (unique tangent at any point on the beam). A point-to-point beam contact model enforces the contact constraint by prohibiting penetration of the two beams at the closest point. This point is the solution of the bilateral closest point projection. The contact force that is transferred between the two beams (obtained by a penalty function) is in this formulation a discrete force acting at the closest points of the beam centrelines in normal direction (after computing the residual, this force is distributed among the nodal forces of the element). A line-to-line model is based on a line constraint enforced along the entire beam length. A distinction has to be made between the master and the slave beam. The closest master point to a given slave point is determined as the solution of the unilateral minimal distance problem. Thus in contrast to the point contact model, the normal vector is still perpendicular to the master beam, but not to the slave beam.

Chapter 3

Theory

This chapter covers the essential principles of Lie groups, geometrically exact beam theory, dynamics, and contacts, along with my personal insights that contributed to the development of new finite elements.

3.1 A primer on Lie groups

Understanding the Lie groups is fundamental for development of efficient beam elements, specifically to address element's temporal and spatial discretisation which introduces the minimal amount of numerical bias. This enables a new way of numerical treatment of beams which inherently underscores their mathematical properties, such as interdependency between cross-section orientation and bending. In the following I try to shed a light on the absolute essentials of Lie groups.

3.1.1 Lie groups

Mathematically, the geometry of a beam can be represented using Lie groups. A Lie group is a differentiable manifold with an additional composition rule between the group members. In the context of a beam, the orientation of its cross-section can be described as a member of the $SO(3)$ group (a group of 3D rotations), while its position can be described as a member of the 3D linear space, $\mathbb{R}^3$. However, the position and the orientation can also be treated as a single member of the special Euclidean group $SE(3)$.

When working with finite elements it is especially useful to adopt the matrix representation of a Lie group. For a linear space, this would be the vector itself. The $SO(3)$ group is a group of all orthogonal matrices with determinant equal to 1 so the matrix representation are such

orthogonal matrices. An $SE(3)$ group member can be viewed as a pair of an $\mathbb{R}^3$ member and an $SO(3)$ member:

$$\mathbf{H} = (\boldsymbol{r},\ \boldsymbol{\Lambda}), \tag{3.1}$$

where $\mathbf{H} \in SE(3)$, $\boldsymbol{r} \in \mathbb{R}^3$and $\boldsymbol{\Lambda} \in SO(3)$. In the matrix representation looks like a 4×4 matrix:

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{r} \\ \mathbf{0}^T & 1 \end{bmatrix}. \tag{3.2}$$

Sometimes, in literature, we also encounter a 6×6 matrix representation of the $SE(3)$ group. This is discussed in Section 3.1.3.

Each group also has its own group operation. Group operations on members produce new members of the same group and are thus critical to the development of advanced numerical algorithms. For a linear space, the group operation is the addition. For groups $SO(3)$ and $SE(3)$ this would be matrix multiplication.

3.1.2 Lie algebras

Each Lie group also has its corresponding Lie algebra, which is its tangent space at the identity. Tangent space is linear and thus isomorphic to the Euclidean (linear) space. We denote the mapping from a Euclidean space to a Lie algebra using a hat operator, which for the cases under consideration maps a 3D vector to Lie algebra $\mathfrak{so}(3)$ and a 6D vector to Lie algebra $\mathfrak{se}(3)$:

$$\hat{\cdot} : \mathbb{R}^3 \to \mathfrak{so}(3) \wedge \mathbb{R}^6 \to \mathfrak{se}(3). \tag{3.3}$$

The matrix representation of the elements of the two Lie algebras can be written as

$$\hat{\boldsymbol{\omega}} = \widehat{\begin{Bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{Bmatrix}} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}, \quad \hat{\boldsymbol{x}} = \widehat{\begin{Bmatrix} \boldsymbol{u} \\ \boldsymbol{\omega} \end{Bmatrix}} = \begin{bmatrix} \hat{\boldsymbol{\omega}} & \boldsymbol{u} \\ \mathbf{0}^T & 0 \end{bmatrix}, \tag{3.4}$$

where $\boldsymbol{u}, \boldsymbol{\omega}$ are arbitrary vectors in $\mathbb{R}^3$ and $\boldsymbol{x}$ in $\mathbb{R}^6$. In a linear space, Lie algebra corresponds to its group.

3.1.3 Adjoint representation

An adjoint representation of a Lie group G is an alternative representation of a group element which transforms an element of the group's Lie algebra $\mathfrak{g}$ regarded as an element of a linear space. If $h \in G$ and $x \in \mathbb{R}^n$, where n is an appropriate dimension so that $\hat{x} \in \mathfrak{g}$, then

$$\mathrm{Ad}_h : \mathfrak{g} \to \mathfrak{g}; \quad \mathrm{Ad}_h(\hat{x}) = h\hat{x}h^{-1} \tag{3.5}$$

and using a linear tilde operator $\tilde{\cdot}$

$$\mathrm{Ad}_h(\hat{x}) = \widehat{\tilde{h}x}. \tag{3.6}$$

Specifically, the adjoint representation of the $SO(3)$ group maps to itself, i.e. $\tilde{h} = h$ (or, $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}$ with group-specific notation from Section 3.1.1), while for the $SE(3)$ group $\tilde{h}$ can be easily expressed using matrix representation (using notation $\mathbf{H}$ from Section 3.1.1):

$$\widetilde{\begin{bmatrix} \mathbf{\Lambda} & \boldsymbol{r} \\ \mathbf{0}^T & 1 \end{bmatrix}} = \begin{bmatrix} \mathbf{\Lambda} & \hat{\boldsymbol{r}}\mathbf{\Lambda} \\ \mathbf{0} & \mathbf{\Lambda} \end{bmatrix}, \tag{3.7}$$

which produces a 6×6 matrix. The adjoint representation of a Lie group is very useful specifically because of its ability to convert members of lie algebra as if they were part of the linear space (3.6).

Adjoint representation of a Lie algebra is defined using a Lie bracket. For two elements x, y of the Lie algebra $\mathfrak{g}$ with a Lie bracket defined as $[x, y] = xy - yx$, the adjoint representation of $\mathfrak{g}$ is

$$\mathrm{ad}_x : \mathfrak{g} \to \mathfrak{g}; \quad \mathrm{ad}_x(y) = [x, y]. \tag{3.8}$$

Since Lie algebra is isomorphic to linear space, its adjoint representation can also be expressed as a linear operator $\tilde{\cdot}$. The following relationship exists

$$\mathrm{ad}_x(y) = \widehat{\tilde{x}y}. \tag{3.9}$$

Specifically, the adjoint representation of the $\mathfrak{so}(3)$ Lie algebra maps to itself, i.e. $\tilde{x} = x$, while for the $\mathfrak{se}(3)$ Lie algebra, it is defined using the matrix representation as

$$\widetilde{\begin{bmatrix} \hat{\boldsymbol{\omega}} & \boldsymbol{u} \\ \mathbf{0}^T & 0 \end{bmatrix}} = \begin{bmatrix} \hat{\boldsymbol{\omega}} & \hat{\boldsymbol{u}} \\ \mathbf{0} & \hat{\boldsymbol{\omega}} \end{bmatrix}. \tag{3.10}$$

3.1.4 Exponential map

A member of a Lie algebra can be mapped to its Lie group via exponentiation

$$\exp : \mathfrak{so}(3) \to SO(3) \wedge \mathfrak{se}(3) \to SE(3). \tag{3.11}$$

Exponential map can be expressed through a series as

$$\exp(\mathbf{X}) = \sum_{i=0}^{\infty} \frac{\mathbf{X}^i}{i!}, \tag{3.12}$$

for any square matrix $\mathbf{X}$. For $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$, its differential is expressible as

$$\mathrm{d}\exp(\hat{\boldsymbol{x}}) = \exp(\hat{\boldsymbol{x}})\widehat{\mathbf{T}(\tilde{\boldsymbol{x}})\mathrm{d}\boldsymbol{x}}, \tag{3.13}$$

where $\mathbf{T}$ is a tangential map with the following series expansion

$$\mathbf{T}(\tilde{\boldsymbol{x}}) = \sum_{i=0}^{\infty} (-1)^i \frac{\tilde{\boldsymbol{x}}^i}{(i+1)!}. \tag{3.14}$$

Certain groups have also a closed form solution of the series, among which are also the groups in question. For $SO(3)$, the exponential map is analytically expressible as

$$\exp(\hat{\boldsymbol{\omega}}) = \mathbf{I} + \frac{\sin(\omega)}{\omega}\hat{\boldsymbol{\omega}} + \frac{1-\cos(\omega)}{\omega^2}\hat{\boldsymbol{\omega}}^2, \qquad \omega = \sqrt{\boldsymbol{\omega}^T\boldsymbol{\omega}}, \tag{3.15}$$

while the tangential map in $SO(3)$ can be expressed as

$$\mathbf{T}(\tilde{\boldsymbol{\omega}}) = \mathbf{I} - \frac{1-\cos(\omega)}{\omega^2}\hat{\boldsymbol{\omega}} + \frac{1-\sin(\omega)/\omega}{\omega^2}\hat{\boldsymbol{\omega}}^2. \tag{3.16}$$

The maps in $SE(3)$ read

$$\exp(\hat{\boldsymbol{x}}) = \begin{bmatrix} \exp(\hat{\boldsymbol{\omega}}) & \mathbf{T}(\tilde{\boldsymbol{\omega}})^T\boldsymbol{u} \\ \mathbf{0}^T & 1 \end{bmatrix}, \tag{3.17}$$

$$\mathbf{T}(\tilde{\boldsymbol{x}}) = \begin{bmatrix} \mathbf{T}(\tilde{\boldsymbol{\omega}}) & \mathbf{T}_{U\Omega}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{\omega}}) \\ \mathbf{0} & \mathbf{T}(\tilde{\boldsymbol{\omega}}) \end{bmatrix}, \tag{3.18}$$

where

$$\begin{aligned} \mathbf{T}_{U\Omega}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{\omega}}) = &- \frac{1-\cos(\omega)}{\omega^2}\hat{\boldsymbol{u}} + \frac{1-\sin(\omega)/\omega}{\omega^2}(\hat{\boldsymbol{u}}\hat{\boldsymbol{\omega}} + \hat{\boldsymbol{\omega}}\hat{\boldsymbol{u}}) \\ &+ \frac{\boldsymbol{u}^T\boldsymbol{\omega}}{\omega^2}\left[\left(2\frac{1-\cos(\omega)}{\omega^2} - \frac{\sin(\omega)}{\omega}\right)\hat{\boldsymbol{\omega}} + \left(\frac{1-\cos(\omega)}{\omega^2} - 3\frac{1-\sin(\omega)/\omega}{\omega^2}\right)\hat{\boldsymbol{\omega}}^2\right]. \end{aligned} \tag{3.19}$$

A logarithmic map is the inverse function of an exponential map. Analytical expressions exist also for these and can be found in literature, e.g. [Spurrier, 1978, Sonneville et al., 2014], as can also the inverses of the tangent maps.

3.1.5 Non-commutativity of derivatives

It is worth noting that derivatives are, in general, non-commutative on a Lie algebra. Let us assume that we have a general derivative, for example wrt. time or arc-length coordinate denoted as $\mathrm{d}\mathbf{H} = \mathbf{H}\widehat{\mathrm{d}\boldsymbol{h}}$, with $\mathrm{d}\boldsymbol{h} \in \mathbb{R}^6$ and a different derivative, for example a variation $\delta\mathbf{H} = \mathbf{H}\widehat{\delta\boldsymbol{h}}$, with $\delta\boldsymbol{h} \in \mathbb{R}^6$, then $\delta(\mathrm{d}\boldsymbol{h}) \neq \mathrm{d}(\delta\boldsymbol{h})$ even though $\delta(\mathrm{d}\mathbf{H}) = \mathrm{d}(\delta\mathbf{H})$. To derive a relation between them we first note

$$\delta(\mathrm{d}\mathbf{H}) = \mathrm{d}(\delta\mathbf{H}) \quad \Rightarrow \quad \mathbf{H}\widehat{\delta\boldsymbol{h}}\widehat{\mathrm{d}\boldsymbol{h}} + \mathbf{H}\delta(\widehat{\mathrm{d}\boldsymbol{h}}) = \mathbf{H}\widehat{\mathrm{d}\boldsymbol{h}}\widehat{\delta\boldsymbol{h}} + \mathbf{H}\mathrm{d}(\widehat{\delta\boldsymbol{h}}) \tag{3.20}$$

After reducing the equation by multiplying from the left with $\mathbf{H}^{-1}$, the expression can be simplified to

$$\delta(\mathrm{d}\boldsymbol{h}) = \widetilde{\mathrm{d}\boldsymbol{h}}\delta\boldsymbol{h} + \mathrm{d}(\delta\boldsymbol{h}). \tag{3.21}$$

3.2 Geometrically exact beam theory

A beam is characterised by the position of its centreline to which an oriented rigid cross-section is attached. As such, a beam can be placed on the special Euclidean group $SE(3)$ by parametrising its configuration as

$$\mathbf{H} : \mathbb{R} \times [0, L] \subset \mathbb{R} \to SE(3); (t, s) \mapsto \mathbf{H}(t, s). \tag{3.22}$$

Traditionally, the position and the orientation have often been considered separately

$$\boldsymbol{r} \in \mathbb{R}^3, \quad \boldsymbol{\Lambda} \in SO(3). \tag{3.23}$$

In $SE(3)$, its configuration can be represented using a configuration matrix, expressed as

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{r} \\ \mathbf{0}^T & 1 \end{bmatrix}. \tag{3.24}$$

The main difference stemming from the two approaches is the group operation, i. e. the composition rule between two group members. Whereas $\mathbb{R}^3 \times SO(3)$ composes the two parts individually, e.g.

$$\boldsymbol{r}_3 = \boldsymbol{r}_1 + \boldsymbol{r}_2 \quad \wedge \quad \boldsymbol{\Lambda}_3 = \boldsymbol{\Lambda}_1 \boldsymbol{\Lambda}_2, \tag{3.25}$$

the $SE(3)$ approach uses the matrix product on $SE(3)$ members, e. g.

$$\mathbf{H}_3 = \mathbf{H}_1 \mathbf{H}_2 = \begin{bmatrix} \boldsymbol{\Lambda}_1 & \boldsymbol{r}_1 \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_2 & \boldsymbol{r}_2 \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Lambda}_1 \boldsymbol{\Lambda}_2 & \boldsymbol{r}_1 + \boldsymbol{\Lambda}_1 \boldsymbol{r}_2 \\ \mathbf{0}^T & 1 \end{bmatrix}, \tag{3.26}$$

which differs from the result in (3.25) in the linear (position) part.

Differentials of $\mathbf{H}$ can be defined using left- and right-invariant vector fields, which are associated with the fixed-pole and material frame of reference, respectively. The following uses dot and prime notation for temporal and spatial derivatives, respectively. For example, velocity in the material frame can be defined as

$$\dot{\mathbf{H}} = \mathbf{H} \hat{\boldsymbol{v}}. \tag{3.27}$$

Velocity $\boldsymbol{v} \in \mathbb{R}^6$ represents a combined vector of linear and angular velocity. It should be noted, that the operations in the adjoint representation are analogue, i.e. $\dot{\tilde{\mathbf{H}}} = \tilde{\mathbf{H}} \tilde{\boldsymbol{v}}$. Similarly goes for spatial derivative

$$\mathbf{H}' = \mathbf{H} \hat{\boldsymbol{e}}, \tag{3.28}$$

where $\boldsymbol{e} \in \mathbb{R}^6$ represents the deformation gradient, and also $\tilde{\mathbf{H}}' = \tilde{\mathbf{H}} \tilde{\boldsymbol{e}}$. The strain $\boldsymbol{\varepsilon} \in \mathbb{R}^6$ follows as

$$\boldsymbol{\varepsilon} = \boldsymbol{e} - \boldsymbol{e}_0 \tag{3.29}$$

with $\boldsymbol{e}_0$ being the deformation gradient of the initial state. The strain is a six-dimensional vector expressed in the material frame where the linear component represent the tensile and the shear stresses in the direction of the cross-section axes, while the angular component represents the torsion and the bending curvatures about the cross-section axes.

3.2.1 Potential and kinetic energy

While the utilised beam model is geometrically nonlinear and allows finite deformations, material nonlinearities are not supported by the formulation. The following developments thus assume that the material is linear elastic and is not applicable to nonlinear material models.

For a linear-elastic material, the kinetic and the potential energy of a beam are defined as [Sonneville et al., 2014]

$$EK = \frac{1}{2}\int_L \boldsymbol{v}^T \mathbf{M}\boldsymbol{v}\mathrm{d}s, \tag{3.30}$$

$$EP = \frac{1}{2}\int_L \boldsymbol{\varepsilon}^T \mathbf{K}\boldsymbol{\varepsilon}\mathrm{d}s. \tag{3.31}$$

Matrices $\mathbf{M}$ and $\mathbf{K}$ hold material data about stiffness and inertial properties of the cross-section. Usually they are defined around the main axes of the cross-section as $\mathbf{M} = \mathrm{diag}(\rho A, \rho A, \rho A, \rho(I_1 + I_2), \rho I_1, \rho I_2)$ and $\mathbf{K} = \mathrm{diag}(EA, GA\kappa_1, GA\kappa_2, GI_t, EI_1, EI_2)$. Material parameters E and G are Young's and shear modulus, while the rest are geometric properties of the cross-section:

- A ... area,
- A_i ... area multiplied by the shear coefficient for the i-th principal axis,
- J_t ... torsional moment of inertia,
- I_i ... second moment of inertia around the i-th principal axis.

Variations of the kinetic and potential energy can now be computed. To ease the derivation we first resolve the variation of the strain using (3.29) and (3.21):

$$\delta\boldsymbol{\varepsilon} = \delta(\boldsymbol{e} - \boldsymbol{e}_0) = \delta\boldsymbol{e} = \tilde{\boldsymbol{e}}\delta\boldsymbol{h} + (\delta\boldsymbol{h})' \tag{3.32}$$

Equations (3.30) and (3.31) are varied with the help of derivatives (3.27) and (3.28) while acknowledging the relationship between mixed derivatives (3.21)

$$\begin{aligned} \delta EK &= \int_L \delta\boldsymbol{v}^T \mathbf{M}\boldsymbol{v}\mathrm{d}s \\ &= \int_L (\tilde{\boldsymbol{v}}\delta\boldsymbol{h} + \delta\dot{\boldsymbol{h}})^T \mathbf{M}\boldsymbol{v}\mathrm{d}s, \end{aligned} \tag{3.33}$$

$$\begin{aligned}\delta EP &= \int_L \delta\boldsymbol{\varepsilon}^T \mathbf{K}\boldsymbol{\varepsilon}\mathrm{d}s \\ &= \int_L (\tilde{\boldsymbol{e}}\delta\boldsymbol{h} + \delta\boldsymbol{h}')^T \mathbf{K}\boldsymbol{\varepsilon}\mathrm{d}s \\ &= \left[\delta\boldsymbol{h}^T \mathbf{K}\boldsymbol{\varepsilon}\right]_0^L + \int_L \delta\boldsymbol{h}^T \tilde{\boldsymbol{e}}^T \mathbf{K}\boldsymbol{\varepsilon} - \delta\boldsymbol{h}^T \mathbf{K}\boldsymbol{\varepsilon}'\mathrm{d}s.\end{aligned} \tag{3.34}$$

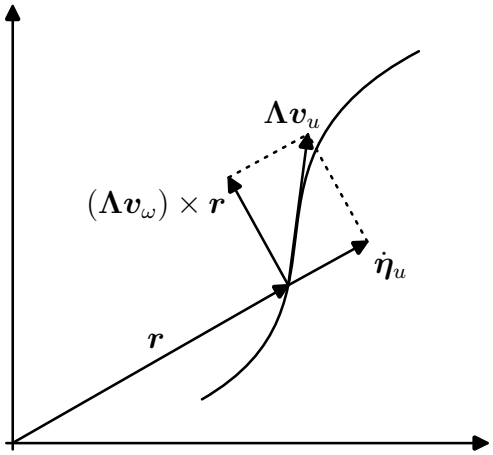


Figure 3.1: Spatial linear velocity $\boldsymbol{\Lambda v}_u$, fixed-pole linear velocity $\dot{\boldsymbol{\eta}}_u$ and the velocity due to change of orientation of the material particle $(\boldsymbol{\Lambda v}_\omega) \times \boldsymbol{r}$.

Virtual work of external forces in material frame can be expressed as

$$\delta W = \delta\boldsymbol{h}(0)^T \overline{\boldsymbol{P}}_0 + \delta\boldsymbol{h}(L)^T \overline{\boldsymbol{P}}_L + \int_L \delta\boldsymbol{h}^T \overline{\boldsymbol{q}}\,\mathrm{d}s, \tag{3.35}$$

where $\overline{\boldsymbol{q}} \in \mathbb{R}^6$ is the distributed external load and $\overline{\boldsymbol{P}}_0, \overline{\boldsymbol{P}}_L \in \mathbb{R}^6$ are the concentrated external loads at the boundaries, all expressed in the material frame.

The equation of motion can be derived using Hamilton's principle, which asserts that the action integral within a specific time interval $[t_a, t_b] \subset \mathbb{R}$, remains unchanged (i.e., is stationary) for the actual path taken by the system between any two points in time

$$\int_{t_a}^{t_b} (\delta EK - \delta EP + \delta W)\,\mathrm{d}t = 0. \tag{3.36}$$

Utilising the result from (3.33) and *per-partes* integration allows us to rewrite the inertial term in (3.36) as

$$\int_{t_a}^{t_b} \delta EK\,\mathrm{d}t = \left[\int_L \delta\boldsymbol{h}^T \mathbf{M}\boldsymbol{v}\,\mathrm{d}s\right]_{t_a}^{t_b} - \int_{t_a}^{t_b}\int_L \delta\boldsymbol{h}^T \mathbf{M}\dot{\boldsymbol{v}} - \delta\boldsymbol{h}^T \tilde{\boldsymbol{v}}^T \mathbf{M}\boldsymbol{v}\,\mathrm{d}s\,\mathrm{d}t. \tag{3.37}$$

Since the variations are fixed at t_a and t_b, the first term on the right hand side vanishes [Sonneville et al., 2014]. All of the remaining terms are within the time integral which is equated to zero. As a result, the remaining integrand itself can be directly equated to zero, yielding

$$-\delta\boldsymbol{h}(0)^T\overline{\boldsymbol{P}}_0 - \delta\boldsymbol{h}(L)^T\overline{\boldsymbol{P}}_L + \left[\delta\boldsymbol{h}^T\mathbf{K}\boldsymbol{\varepsilon}\right]_0^L + \int_L \delta\boldsymbol{h}^T\left(\mathbf{M}\dot{\boldsymbol{v}} - \tilde{\boldsymbol{v}}^T\mathbf{M}\boldsymbol{v} - \mathbf{K}\boldsymbol{\varepsilon}' + \tilde{\boldsymbol{e}}^T\mathbf{K}\boldsymbol{\varepsilon} - \overline{\boldsymbol{q}}\right)\mathrm{d}s = 0. \tag{3.38}$$

Variations can be equivalently expressed also in the fixed-pole frame as

$$\delta\mathbf{H} = \widehat{\delta\boldsymbol{\eta}}\mathbf{H}, \tag{3.39}$$

where $\delta\boldsymbol{\eta} \in \mathbb{R}^6$. They are related to material frame variations via (see Figure 3.1 for graphical illustration)

$$\delta\boldsymbol{\eta} = \tilde{\mathbf{H}}\delta\boldsymbol{h}. \tag{3.40}$$

Remark 1. *The relationship between the material and the fixed-pole variations in* (3.40) *may be understood by illustrating its expression over an infinitesimal time segment in which case those variations turn into respective velocities*

$$\dot{\boldsymbol{\eta}} = \tilde{\mathbf{H}}\dot{\boldsymbol{h}}$$

where $\dot{\boldsymbol{h}}$ *is the vector of material velocities* $\boldsymbol{v}^T = \left\{\boldsymbol{v}_u^T \quad \boldsymbol{v}_\omega^T\right\}$*, with* $\boldsymbol{v}_u$ *and* $\boldsymbol{v}_\omega$ *being the respective linear and angular material velocities. Likewise, the fixed-pole velocity vector* $\dot{\boldsymbol{\eta}}^T = \left\{\dot{\boldsymbol{\eta}}_u^T \quad \dot{\boldsymbol{\eta}}_\omega^T\right\}$ *consist of the linear fixed-pole velocity* $\dot{\boldsymbol{\eta}}_u$ *and the angular fixed-pole velocity* $\dot{\boldsymbol{\eta}}_\omega$*. The above relationship thus provides*

$$\dot{\boldsymbol{\eta}}_u = \boldsymbol{\Lambda}\boldsymbol{v}_u + \hat{\boldsymbol{r}}\boldsymbol{\Lambda}\boldsymbol{v}_\omega = \boldsymbol{\Lambda}\boldsymbol{v}_u - \widehat{\boldsymbol{\Lambda}\boldsymbol{v}_\omega}\boldsymbol{r}$$
$$\dot{\boldsymbol{\eta}}_\omega = \boldsymbol{\Lambda}\boldsymbol{v}_\omega$$

indicating that the fixed-pole angular velocity is in fact equal to the spatial angular velocity $\boldsymbol{\Lambda}\boldsymbol{v}_u$ *obtained by push-forward. The fixed-pole linear velocity differs from the spatial linear velocity owing to the presence of the term* $\dot{\boldsymbol{\eta}}_\omega \times \boldsymbol{r}$*, which represents the part of the linear velocity caused by the change of orientation of the material particle. The fixed-pole linear velocity thus represents the part of the spatial velocity vector from which the velocity due to the change in the particle orientation has been extracted. A simple 2D illustration is provided in Figure 3.1. Likewise, the fixed-pole variation* $\delta\boldsymbol{\eta}$ *consists of a linear and an angular part, of which the latter is the same as the spatial angular variation, while the former is obtained from the spatial linear variation by extracting the variation due to the change of orientation of the material particle.*

If we define the fixed-pole momentum $\boldsymbol{p} \in \mathbb{R}^6$ as

$$\boldsymbol{p} = \tilde{\mathbf{H}}^{-T}\mathbf{M}\boldsymbol{v}, \tag{3.41}$$

the fixed-pole internal forces $\boldsymbol{f} \in \mathbb{R}^6$ as

$$\boldsymbol{f} = \tilde{\mathbf{H}}^{-T}\mathbf{K}\boldsymbol{\varepsilon} \tag{3.42}$$

and the fixed-pole external distributed $\boldsymbol{q} \in \mathbb{R}^6$ and concentrated $\boldsymbol{P} \in \mathbb{R}^6$ forces as

$$\boldsymbol{q} = \tilde{\mathbf{H}}^{-T}\overline{\boldsymbol{q}}, \quad \boldsymbol{P} = \tilde{\mathbf{H}}^{-T}\overline{\boldsymbol{P}} \tag{3.43}$$

to obtain the following auxiliary results

- from (3.40) and (3.43) $\delta\boldsymbol{h}^T\overline{\boldsymbol{P}} = \delta\boldsymbol{\eta}^T\overline{\boldsymbol{P}}$ and $\delta\boldsymbol{h}^T\overline{\boldsymbol{q}} = \delta\boldsymbol{\eta}^T\boldsymbol{q}$,
- from (3.40) and (3.42) $\delta\boldsymbol{h}^T\mathbf{K}\boldsymbol{\varepsilon} = \delta\boldsymbol{\eta}^T\boldsymbol{f}$,
- from (3.27), (3.40) and (3.41)

$$\delta\boldsymbol{\eta}^T\dot{\boldsymbol{p}} = \delta\boldsymbol{h}^T\tilde{\mathbf{H}}^T\left(\tilde{\mathbf{H}}^{-T}\mathbf{M}\dot{\boldsymbol{v}} - \tilde{\mathbf{H}}^{-T}\tilde{\boldsymbol{v}}^T\tilde{\mathbf{H}}^T\tilde{\mathbf{H}}^{-T}\mathbf{M}\boldsymbol{v}\right) = \delta\boldsymbol{h}^T\left(\mathbf{M}\dot{\boldsymbol{v}} - \tilde{\boldsymbol{v}}^T\mathbf{M}\boldsymbol{v}\right),$$

- from (3.28), (3.40) and (3.42)

$$\delta\boldsymbol{\eta}^T\boldsymbol{f}' = \delta\boldsymbol{h}^T\tilde{\mathbf{H}}^T\left(\tilde{\mathbf{H}}^{-T}\mathbf{K}\boldsymbol{\varepsilon}' - \tilde{\mathbf{H}}^{-T}\tilde{\boldsymbol{e}}^T\tilde{\mathbf{H}}^T\tilde{\mathbf{H}}^{-T}\mathbf{K}\boldsymbol{\varepsilon}\right) = \delta\boldsymbol{h}^T\left(\mathbf{K}\boldsymbol{\varepsilon}' - \tilde{\boldsymbol{e}}^T\mathbf{K}\boldsymbol{\varepsilon}\right),$$

it can be easily shown that the equilibrium equation in the material frame (3.38) reduces to

$$-\delta\boldsymbol{\eta}(0)^T\boldsymbol{P}_0 - \delta\boldsymbol{\eta}(L)^T\boldsymbol{P}_L + \left[\delta\boldsymbol{\eta}^T\boldsymbol{f}\right]_0^L + \int_L \delta\boldsymbol{\eta}^T\left(\dot{\boldsymbol{p}} - \boldsymbol{f}' - \boldsymbol{q}\right)\mathrm{d}s = 0. \tag{3.44}$$

Integration *per partes* leads to an alternative expression for the weak formulation in the fixed-pole frame

$$-\delta\boldsymbol{\eta}(0)^T\boldsymbol{P}_0 - \delta\boldsymbol{\eta}(L)^T\boldsymbol{P}_L + \int_L \delta\boldsymbol{\eta}^T\left(\dot{\boldsymbol{p}} - \boldsymbol{q}\right) + \delta\boldsymbol{\eta}'^T\boldsymbol{f}\mathrm{d}s = 0. \tag{3.45}$$

3.2.2 Spatial discretisation

The domain of a beam is now subdivided into N_{ele} finite elements with N_{nod} total nodes and $N_{\text{nod,glob}}$ global nodes in the mesh and the weak formulation (3.45) becomes

$$-\sum_{i=1}^{N_{\text{nod,glob}}} \underline{\delta\boldsymbol{\eta}}_i^T\underline{\boldsymbol{P}}_i + \sum_{k=1}^{N_{\text{ele}}}\int_{L_k} \delta\boldsymbol{\eta}_k^T\left(\dot{\boldsymbol{p}}_k - \boldsymbol{q}_k\right) + \delta\boldsymbol{\eta}_k'^T\boldsymbol{f}_k\mathrm{d}s_k = 0, \tag{3.46}$$

where $L_k \in \mathbb{R}$ represents the undeformed length of k-th finite element. The two summations control the assembly of finite elements, see e.g. [Zienkiewicz et al., 2005].

The underlined symbols represent nodal values. On the element-level we introduce interpolation of $\delta\boldsymbol{\eta}$ (index k will be omitted on the element-level). We have selected the ordinary Lagrange interpolation to develop the momentum and near-energy conserving element, [Tomec and Jelenić, 2024b], but other interpolation are also valid and lead to different finite elements, e.g. [Bottasso and Borri, 1998, Sonneville et al., 2014].

$$\delta\boldsymbol{\eta} = \boldsymbol{\Psi}\underline{\delta\boldsymbol{\eta}}, \tag{3.47}$$

where $\underline{\delta\boldsymbol{\eta}}$ is a column vector consisting of nodal values and shape functions matrix $\boldsymbol{\Psi}$ is defined as

$$\boldsymbol{\Psi} = \begin{bmatrix} \psi_1\mathbf{I} & \psi_2\mathbf{I} & \dots & \psi_i\mathbf{I} & \dots & \psi_N\mathbf{I} \end{bmatrix}, \tag{3.48}$$

where N is the number of nodes on the element, ψ_i is the i-th interpolation function and $\mathbf{I}$ is an identity matrix of size 6. We set the following constraint on the choice of the interpolation functions (the reason will become clear in the next section)

$$\sum_{j=1}^{N} \Psi_j(s) = 1 \quad \Rightarrow \quad \sum_{j=1}^{N} \Psi_j'(s) = 0. \tag{3.49}$$

Equation (3.46) can now be written as

$$-\underline{\delta\boldsymbol{\eta}}^T \ \underline{\boldsymbol{P}} + \sum_{k=1}^{N_{\text{ele}}} \underline{\delta\boldsymbol{\eta}}_k \int_{L_k} \boldsymbol{\Psi}_k^T \left(\dot{\boldsymbol{p}}_k - \boldsymbol{q}_k\right) + \boldsymbol{\Psi}_k'^T \boldsymbol{f}_k \mathrm{d}s_k = 0, \tag{3.50}$$

where $\underline{\delta\boldsymbol{\eta}}_k$ refers to nodal variations which contribute to element k.

Remark 2. *The discretisation introduced so far is incomplete as we have so far only interpolated the test functions. The interpolation of the fields is introduced in Section 3.4, as it is independent of the interpolation of the test functions.*

3.3 Momentum and near-energy conserving time integration

The following discusses the time stepping algorithm used to advance time $t \in \mathbb{R}$ from time t_n to $t_{n+1} = t_n + \Delta t$. All quantities indexed with n or $n+1$ refer to these two time instants. The configuration is updated using a Lie group version of the update equation so that the solution remains on the manifold. The update can be expressed using the time-step increments in the fixed-pole frame as

$$\mathbf{H}_{n+1} = \exp(\hat{\boldsymbol{\vartheta}})\mathbf{H}_n, \tag{3.51}$$

where increments $\boldsymbol{\vartheta} \in \mathbb{R}^6$. They can be mapped between the fixed-pole $\boldsymbol{\vartheta}$ and the material frame version $\boldsymbol{t} \in \mathbb{R}^6$ using the following relation

$$\boldsymbol{\vartheta} = \tilde{\mathbf{H}}_{n+1}\boldsymbol{t}. \tag{3.52}$$

Using the definition (3.51), the above expression can be written as $\boldsymbol{\vartheta} = \exp(\hat{\boldsymbol{\vartheta}})\mathbf{H}_n\boldsymbol{t}$. After multiplying both sides with $\exp(-\hat{\boldsymbol{\vartheta}})$ from the left and using identity (C.3), an alternative form of the map (3.52) is obtained

$$\boldsymbol{\vartheta} = \mathbf{H}_n\boldsymbol{t}. \tag{3.53}$$

The midpoint rule, which is an implicit second-order time integration scheme, suggests to evaluate the equilibrium equation at the middle of the step, instead of at the end at t_{n+1}. This instance can be interpreted as $t_m = (t_n + t_{n+1})/2$, however, it should be emphasised that the midpoint quantities need not be determined through time evolution; rather, they are chosen in a manner that enables algorithmic conservation of mechanical properties. For this reason, we first present the undetermined quantities in the balance equation set at the midpoint and only later, as we inspect the properties of the conservation of momentum and energy, reverse engineer the proper definitions. The Lie midpoint rule requires the update equation to be conducted using the Lie group operations as presented in (3.51) to respect the specific solution manifold. Following the Lie midpoint rule, the equilibrium equation at the middle of the step becomes

$$-\underline{\delta\boldsymbol{\eta}}^T\underline{\boldsymbol{P}}_m + \sum_{k=1}^{N_{\text{ele}}} \underline{\delta\boldsymbol{\eta}}_k^T \int_{L_k} \boldsymbol{\Psi}_k^T\left(\dot{\boldsymbol{p}}_{k,m} - \boldsymbol{q}_{k,m}\right) + \boldsymbol{\Psi}_k'^T\boldsymbol{f}_{k,m}\mathrm{d}s_k = 0 \tag{3.54}$$

where index m denotes the middle of the step.

3.3.1 Conservation of momentum

Conservation of the total generalised momentum is established when $\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \mathbf{0}$. To check if this is true we start with

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \sum_{k=1}^{N_{\text{ele}}} \int_{L_k} (\boldsymbol{p}_{k,n+1} - \boldsymbol{p}_{k,n})\mathrm{d}s_k. \tag{3.55}$$

Because of the first property in (3.49) of the interpolation functions, we can write

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \sum_{k=1}^{N_{\text{ele}}} \int_{L_k} \sum_{j=1}^{N} \Psi_{kj}(\boldsymbol{p}_{k,n+1} - \boldsymbol{p}_{k,n})\mathrm{d}s_k. \tag{3.56}$$

The upper form suggests to use as the midpoint approximation of the generalised momentum the first-order Taylor series expansion, which is

$$\dot{\boldsymbol{p}}_m \approx \frac{\boldsymbol{p}_{n+1} - \boldsymbol{p}_n}{\Delta t}, \tag{3.57}$$

which renders (3.56) to be

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \Delta t \sum_{k=1}^{N_{\text{ele}}} \int_{L_k} \sum_{j=1}^{N} \Psi_{kj} \dot{\boldsymbol{p}}_{k,m} \mathrm{d}s_k. \tag{3.58}$$

Proving the conservation of the momentum now requires us to show that this quantity is indeed changed only by external forces when solving the balance equation (3.54). The first term in (3.54) vanishes for the assumed Neumann boundary conditions. This leads to the following element residual

$$\boldsymbol{R} = \int_L \boldsymbol{\Psi}^T \left(\dot{\boldsymbol{p}}_m - \boldsymbol{q}_m \right) + \boldsymbol{\Psi}'^T \boldsymbol{f}_m \mathrm{d}s \tag{3.59}$$

assembled by the nodal contributions $\boldsymbol{R}_j$, which can be inserted into (3.58) to give

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \Delta t \sum_{k=1}^{N_{\text{ele}}} \sum_{j=1}^{N} \left(\boldsymbol{R}_j + \int_{L_k} \Psi_{kj} \boldsymbol{q}_{k,m} - \Psi'_{kj} \boldsymbol{f}_{k,m} \mathrm{d}s_k \right). \tag{3.60}$$

Because of the second property of the interpolation function (3.49) and in the absence of external loading we are thus left with

$$\boldsymbol{J}_{n+1} - \boldsymbol{J}_n = \Delta t \sum_{k=1}^{N_{\text{ele}}} \sum_{j=1}^{N} \boldsymbol{R}_j \tag{3.61}$$

which obviously vanishes in the equilibrium thus proving the total momentum conservation in the absence of external loading.

We have thus showed that in order to algorithmically conserve generalised momentum, assuming the Neumann boundary conditions and the absence of distributed loading, in combination with the selected discretised midpoint equilibrium equation (3.54), the midpoint approximation of the generalised momentum should be set as the first-order Taylor series expansion of the momentum time derivative (3.57).

3.3.2 Conservation of energy

The difference in the mechanical energy after a time step can be computed as

$$E_{n+1} - E_n = EK_{n+1} - EK_n + EP_{n+1} - EP_n \tag{3.62}$$

$$= \frac{1}{2} \int_L \left(\boldsymbol{v}_{n+1} + \boldsymbol{v}_n \right)^T \mathbf{M} \left(\boldsymbol{v}_{n+1} - \boldsymbol{v}_n \right) \mathrm{d}s + \frac{1}{2} \int_L \left(\boldsymbol{\varepsilon}_{n+1} + \boldsymbol{\varepsilon}_n \right)^T \mathbf{K} \left(\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_n \right) \mathrm{d}s. \tag{3.63}$$

Let us relate increments and velocities via a midpoint-rule approximation in the material frame

$$\Delta t \frac{\boldsymbol{v}_{n+1} + \boldsymbol{v}_n}{2} = \boldsymbol{t}, \tag{3.64}$$

Using the fixed-pole-material frame relation (3.52) and (3.53), the update can be expressed also using the increment in the fixed-pole frame

$$\Delta t \frac{\boldsymbol{v}_{n+1} + \boldsymbol{v}_n}{2} = \tilde{\mathbf{H}}_{n+1}^{-1} \boldsymbol{\vartheta} = \tilde{\mathbf{H}}_n^{-1} \boldsymbol{\vartheta}. \tag{3.65}$$

The velocities in the above expression conveniently remain in the material frame.

Remark 3. *The use of numerical integration relaxes the relation between the time step and the velocity from* (3.65) *to hold only at integration points.*

Let us consider now the increment of strain obtained by differentiation of the update equation (3.51)

$$\mathbf{H}_{n+1} \hat{\boldsymbol{e}}_{n+1} = \exp(\hat{\boldsymbol{\vartheta}}) \mathbf{H}_n \hat{\boldsymbol{e}}_n + \exp(\hat{\boldsymbol{\vartheta}}) \widehat{\mathbf{T}(\tilde{\boldsymbol{\vartheta}}) \boldsymbol{\vartheta}'} \mathbf{H}_n \tag{3.66}$$

$$\hat{\boldsymbol{e}}_{n+1} = \hat{\boldsymbol{e}}_n + \mathbf{H}_n^{-1} \widehat{\mathbf{T}(\tilde{\boldsymbol{\vartheta}}) \boldsymbol{\vartheta}'} \mathbf{H}_n. \tag{3.67}$$

After transportation from the Lie algebra $\mathfrak{se}(3)$ to linear space $\mathbb{R}^6$ we obtain

$$\tilde{\mathbf{H}}_n^{-1} \, \mathbf{T}(\tilde{\boldsymbol{\vartheta}}) \boldsymbol{\vartheta}' = \boldsymbol{e}_{n+1} - \boldsymbol{e}_n = \boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_n. \tag{3.68}$$

Expressions (3.65) and (3.68) inserted back into (3.63) return

$$E_{n+1} - E_n = \int_L \boldsymbol{\vartheta}^T \frac{\tilde{\mathbf{H}}_{n+1}^{-T} \mathbf{M} \boldsymbol{v}_{n+1} - \tilde{\mathbf{H}}_n^{-T} \mathbf{M} \boldsymbol{v}_n}{\Delta t} + \boldsymbol{\vartheta}'^T \, \mathbf{T}^T(\tilde{\boldsymbol{\vartheta}}) \tilde{\mathbf{H}}_n^{-T} \mathbf{K} \frac{\boldsymbol{\varepsilon}_{n+1} + \boldsymbol{\varepsilon}_n}{2} \mathrm{d}s, \tag{3.69}$$

equivalently written using identity (C.11) and expressions (3.41), (3.42) and (3.51) also as

$$E_{n+1} - E_n = \int_L \boldsymbol{\vartheta}^T \frac{\boldsymbol{p}_{n+1} - \boldsymbol{p}_n}{\Delta t} + \boldsymbol{\vartheta}'^T \frac{\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T \boldsymbol{f}_{n+1} + \mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T \boldsymbol{f}_n}{2} \mathrm{d}s \tag{3.70}$$

and with

$$\boldsymbol{f}_m = \frac{\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T \boldsymbol{f}_{n+1} + \mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T \boldsymbol{f}_n}{2} \tag{3.71}$$

also as

$$E_{n+1} - E_n = \int_L \boldsymbol{\vartheta}^T \dot{\boldsymbol{p}}_m + \boldsymbol{\vartheta}'^T \boldsymbol{f}_m \mathrm{d}s, \tag{3.72}$$

which alludes to the form of the weak formulation (3.45) and (3.54) and implies energy conservation for $\dot{\boldsymbol{p}}_m$ and $\boldsymbol{f}_m$ as defined in (3.57) and (3.71). For a time-stepping algorithm to have the energy conservative property, the time-step increment $\boldsymbol{\vartheta}$ must be discretised using the same shape functions as the variation $\delta\boldsymbol{\eta}$.

3.3.3 Energy decaying algorithm

Energy decaying algorithms are desired to dampen higher frequencies in order to improve the stability of the algorithm [Ibrahimbegovic and Mamouri, 2002, Arnold and Brüls, 2007]. As demonstrated by [Bottasso and Borri, 1998, Ibrahimbegovic and Mamouri, 2002], energy decay in the midpoint rule is accomplished by including a damping term in the midpoint force (3.71) to obtain

$$\boldsymbol{f}_m^d = \overbrace{\frac{1}{2}\left(\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_{n+1} + \mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_n\right)}^{\boldsymbol{f}_m} + \frac{1-\tau}{2}\left(\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_{n+1} - \mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_n\right) \tag{3.73}$$

$$= \left(1-\frac{\tau}{2}\right)\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_{n+1} + \frac{\tau}{2}\,\mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_n \tag{3.74}$$

for $\tau \in (0,1]$. Such definition dampens out the deformations while having no effect on the rigid-body motion. Setting $\tau = 1$ recovers the energy-preserving midpoint-force (3.71).

Under the assumptions stated in Section 3.3.2, the total energy of the beam is conserved. By rewriting the expression from (3.72) using the decaying midpoint force (3.73) we get

$$\int_L \boldsymbol{\vartheta}^T\dot{\boldsymbol{p}}_m + \boldsymbol{\vartheta}'^T\boldsymbol{f}_m^d \mathrm{d}s = 0. \tag{3.75}$$

Expanding this expression reveals

$$\begin{aligned}
&\int_L \boldsymbol{\vartheta}^T\dot{\boldsymbol{p}}_m + \boldsymbol{\vartheta}'^T\boldsymbol{f}_m + \frac{1-\tau}{2}\boldsymbol{\vartheta}'^T\left(\mathbf{T}(-\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_{n+1} - \mathbf{T}(\tilde{\boldsymbol{\vartheta}})^T\boldsymbol{f}_n\right)\mathrm{d}s \\
&= \frac{1}{2}\int_L (\boldsymbol{v}_{n+1}+\boldsymbol{v}_n)^T\mathbf{M}\,(\boldsymbol{v}_{n+1}-\boldsymbol{v}_n)\,\mathrm{d}s + \frac{1}{2}\int_L (\boldsymbol{\varepsilon}_{n+1}+\boldsymbol{\varepsilon}_n)^T\,\mathbf{K}\,(\boldsymbol{\varepsilon}_{n+1}-\boldsymbol{\varepsilon}_n)\,\mathrm{d}s \\
&\qquad + \frac{1-\tau}{2}\int_L (\boldsymbol{\varepsilon}_{n+1}-\boldsymbol{\varepsilon}_n)^T\,\mathbf{K}\,(\boldsymbol{\varepsilon}_{n+1}-\boldsymbol{\varepsilon}_n)\,\mathrm{d}s = 0.
\end{aligned} \tag{3.76}$$

Defining the decaying term $\mathcal{D}$ as

$$\mathcal{D} = \frac{1-\tau}{2}\int_L (\boldsymbol{\varepsilon}_{n+1}-\boldsymbol{\varepsilon}_n)^T\,\mathbf{K}\,(\boldsymbol{\varepsilon}_{n+1}-\boldsymbol{\varepsilon}_n)\,\mathrm{d}s \tag{3.77}$$

allows us to reorganize the terms into

$$E_{n+1} - E_n + \mathcal{D} = 0. \tag{3.78}$$

3.3.4 External forces

Since the increment of work done by external forces can be calculated as

$$\Delta W = \int_L \boldsymbol{\vartheta}^T\boldsymbol{q}_m \mathrm{d}s, \tag{3.79}$$

the midpoint load can be defined as the average load

$$\boldsymbol{q}_m = \frac{1}{2}\left(\boldsymbol{q}_{n+1} + \boldsymbol{q}_n\right). \tag{3.80}$$

3.4 Finite element formulation

A finite element has its configuration state interpolated using the implicit $SE(3)$ interpolation, as defined in [Sonneville et al., 2017]. Such interpolation provides objective, strain-invariant and locking-free elements as theoretically explained in [Sonneville et al., 2017]. An analytical expression can be used to define the first-order interpolation, which involves two nodes, see [Sonneville et al., 2014]. However, for higher-order interpolations, an implicit definition is required, which involves solving an additional nonlinear 6D equation.

Let us define the implicit $SE(3)$ configuration interpolation for N nodes per element as

$$\sum_{i=1}^{N} \psi_i \boldsymbol{\phi}_i = \mathbf{0}, \quad \exp(\hat{\boldsymbol{\phi}}_i) = \underline{\mathbf{H}}_i^{-1}\mathbf{H}, \tag{3.81}$$

where $\boldsymbol{\phi}_i \in \mathbb{R}^6$ and shape functions ψ_i are Lagrange polynomials. The auxiliary quantities $\boldsymbol{\phi}_i(s)$ can be physically interpreted as generalised displacement-rotation vectors between the configuration of the i-th node and the configuration at the arc-length coordinate s. This equation needs to be solved using a Newton-Raphson algorithm as presented in A.

The finite element is designed to have $\boldsymbol{\vartheta}$ as the nodal unknowns as they are conveniently in the tangent space.

Remark 4. *The interpolation of the configuration has been introduced independently of the test functions' interpolation, which have been defined in* (3.49) *to facilitate momentum conservation. The selected configuration interpolation is in violation of the conservation of energy requirement that the time-step increment $\boldsymbol{\vartheta}$ must be expressible with the same shape functions as the variation $\delta\boldsymbol{\eta}$ after discretisation. However, using the described algorithm, mesh refinement leads to a smaller discrepancy between the two, which in turn leads to better energy conservation. See Appendix B, where this is theoretically showed. This is also illustrated with numerical examples.*

3.4.1 Explicit first-order interpolation

For two nodes, the $SE(3)$ interpolation (3.81) reduces to the helicoidal interpolation and can be thus expressed explicitly. Between nodes A and B, the configuration is interpolated as

$$\mathbf{H}(s) = \mathbf{H}_A \exp\left(\frac{s}{L}\hat{\boldsymbol{d}}\right), \quad \exp(\hat{\boldsymbol{d}}) = \mathbf{H}_A^{-1}\mathbf{H}_B, \tag{3.82}$$

where $\boldsymbol{d} \in \mathbb{R}^6$. The spatial derivative of the curve $\mathbf{H}(s)$ is constant

$$\mathbf{H}'(s) = \frac{1}{L}\mathbf{H}\hat{\boldsymbol{d}}, \tag{3.83}$$

yielding

$$\boldsymbol{e} = \frac{\boldsymbol{d}}{L}. \tag{3.84}$$

Instantaneous velocity $\boldsymbol{v}$ is not interpolated, see the explanation under the next section. The algorithm is dedscribed in Box 3.1.

3.4.2 Implicit interpolation

Spatial differentiation of (3.81) yields

$$\sum_{i=1}^{N} \psi_i' \boldsymbol{\phi}_i + \psi_i \boldsymbol{\phi}_i' = \mathbf{0}, \quad \exp(\hat{\boldsymbol{\phi}}_i)\widehat{\mathbf{T}(\tilde{\boldsymbol{\phi}}_i)\boldsymbol{\phi}_i'} = \exp(\hat{\boldsymbol{\phi}}_i)\hat{\boldsymbol{e}}. \tag{3.85}$$

From the second equation in (3.85) we can express $\boldsymbol{\phi}_i'$

$$\boldsymbol{\phi}_i' = \mathbf{T}^{-1}(\tilde{\boldsymbol{\phi}}_i)\boldsymbol{e}. \tag{3.86}$$

Inserting it into the first equation in (3.85) yields

$$\boldsymbol{e} = -\boldsymbol{\Psi}^* \sum_{i=1}^{N} \psi_i' \boldsymbol{\phi}_i, \tag{3.87}$$

where

$$\boldsymbol{\Psi}^* = \left[\sum_{i=1}^{N} \psi_i \, \mathbf{T}^{-1}(\tilde{\boldsymbol{\phi}}_i) \right]^{-1} \tag{3.88}$$

Note that this expression can be explicitly evaluated if the implicitly defined $\mathbf{H}(s)$ has already been computed.

Time differentiation of (3.81) yields

$$\sum_{i=1}^{N} \psi_i \dot{\boldsymbol{\phi}}_i = \mathbf{0}, \quad \exp(\hat{\boldsymbol{\phi}}_i)\widehat{\mathbf{T}(\tilde{\boldsymbol{\phi}}_i)\dot{\boldsymbol{\phi}}_i} = \exp(\hat{\boldsymbol{\phi}}_i)\hat{\boldsymbol{v}} - \underline{\hat{\boldsymbol{v}}}_i \exp(\hat{\boldsymbol{\phi}}_i) \tag{3.89}$$

from where we get

$$\dot{\boldsymbol{\phi}}_i = \mathbf{T}^{-1}(\tilde{\boldsymbol{\phi}}_i)\boldsymbol{v} - \mathbf{T}^{-1}(-\tilde{\boldsymbol{\phi}}_i)\underline{\boldsymbol{v}}_i \tag{3.90}$$

and finally

$$\boldsymbol{v} = \boldsymbol{\Psi}^* \sum_{i=1}^{N} \psi_i \, \mathbf{T}^{-1}(-\tilde{\boldsymbol{\phi}}_i)\underline{\boldsymbol{v}}_i. \tag{3.91}$$

Interpolation of velocity can be now expressed using an interpolation matrix as

$$\boldsymbol{v} = \boldsymbol{\Psi}^* \mathbf{Q} \underline{\boldsymbol{v}}, \tag{3.92}$$

where $\mathbf{Q}$ is a $6 \times 6N$ matrix, defined as

$$\mathbf{Q} = \left[\psi_1 \, \mathbf{T}^{-1}(-\tilde{\boldsymbol{\phi}}_1) \quad \ldots \quad \psi_N \, \mathbf{T}^{-1}(-\tilde{\boldsymbol{\phi}}_N) \right]. \tag{3.93}$$

Again, as with the spatial derivative (3.87), the velocity interpolation matrix is explicitly defined. Furthermore, matrix $\boldsymbol{\Psi}^*$ is the same in both cases and can be reused.

The above interpolation of velocity is, however, in conflict with equation (3.65) which relates the time step $\boldsymbol{\eta}$ computed from (3.51) and velocity. Explicit expression of velocity in terms of the nodal increments is thus not possible. However, since we only require the relation from (3.65) to hold at integration points, we can avoid an explicit definition of the velocity field and resort to storing the values at the integration points and update them according to (3.65). See the algorithm in Box 3.1.

3.4.3 Time integrator

The presented formulation can be solved using the Newton-Raphson iterative procedure as explained in the algorithm in Box 3.1. The system of equations to be solved at each iteration i based on residual (3.59) reads

$$\mathbf{K}_{\text{glob}}(\boldsymbol{\vartheta}^i_{\text{glob}})\Delta\boldsymbol{\vartheta}^i_{\text{glob}} = -\boldsymbol{R}_{\text{glob}}(\boldsymbol{\vartheta}^i_{\text{glob}}), \tag{3.94}$$

where Δ is the linearisation operator, $\mathbf{K}_{\text{glob}}$ is the gradient of the assembled residual $\boldsymbol{R}_{\text{glob}}$ wrt. the global unknowns, i.e. time-step increments $\boldsymbol{\vartheta}_{\text{glob}}$. The time-step increments are updated between iterations straightforwardly as

$$\boldsymbol{\vartheta}^{i+1}_{\text{glob}} = \boldsymbol{\vartheta}^i_{\text{glob}} + \Delta\boldsymbol{\vartheta}^i_{\text{glob}}. \tag{3.95}$$

To evaluate the time-step increment at the integration points, one can simply invert (3.51) and obtain

$$\hat{\boldsymbol{\vartheta}}(s) = \log(\mathbf{H}_{n+1}(s)\mathbf{H}_n^{-1}(s)), \tag{3.96}$$

where values $\mathbf{H}_n$ and $\mathbf{H}_{n+1}$ are obtained via interpolation.

Box 3.1: Algorithm for updating the finite element and solving the balance equation

1. Initiate new time step by storing the converged values in nodes $\mathbf{H}_{n+1} \leftarrow \mathbf{H}_n$ and in integration points $\boldsymbol{v}_{n+1} \leftarrow \boldsymbol{v}_n$, $\boldsymbol{q}_{n+1} \leftarrow \boldsymbol{q}_n$ and $\boldsymbol{\vartheta} \leftarrow \Delta t \boldsymbol{v}_n$.
2. Update $\mathbf{H}_{n+1}$ in nodes using (3.51) and $\boldsymbol{v}_{n+1}$ in integration points using (3.65).
3. Compute interpolated values of $\mathbf{H}_{n+1}$ and $\mathbf{H}_n$ in the integration points using (3.81) as explained in A or use (3.82) for first-order elements. Evaluate strains $\boldsymbol{\varepsilon}_{n+1}$ and $\boldsymbol{\varepsilon}_n$ in the integration points using (3.81) and (3.87) or (3.29), (3.82) and (3.84) for first-order elements.
4. Compute the momentum $\boldsymbol{p}_{n+1}$ and $\boldsymbol{p}_n$ using (3.41) and internal forces $\boldsymbol{f}_{n+1}$ and $\boldsymbol{f}_n$ using (3.42), all in the integration points.
5. Compute the residual vector from (3.57), (3.59), (3.74) and (3.80).
6. Check the norm of the residual vector against the tolerance.
7. Solve (3.94) and update $\boldsymbol{\vartheta}$ using (3.95).
8. Continue with 2.

3.5 Contact between beams

Beam-to-beam contact takes place in two different ways based on the nature of the contact force. One is a line-to-line contact, where the force is distributed along the length of the actual contact zone. It describes well an interaction between parallel beams pressed against or twisted around each other. The second one is a point-to-point contact with the contact zone collapsing to a point. It happens when the beams touch at an angle. In this thesis we discuss the line-to-line contact which can be viewed as the more general case capable to adequately describe as its specific case also the point-to-point contact.

3.5.1 Contact constraints

Contact can be mathematically represented through the well-known Karush-Kuhn-Tucker (KKT) constraints [Popp et al., 2009]. Using the Lagrange-multiplier field λ to represent the negative

distributed contact force results in the following constraints:

$$g \geq 0 \tag{3.97a}$$

$$-\lambda \geq 0 \tag{3.97b}$$

$$\lambda g = 0 \tag{3.97c}$$

where g denotes the gap function. The gap is a geometrical measure of the distance between two beams (denoted as beam 1 and beam 2), which we want in a line-to-line method to be a continuous function, which serves to prevent penetration between the beams. It is defined as the distance between the two beam centrelines $\boldsymbol{r}_1(s_1)$ and $\boldsymbol{r}_2(s_2)$ reduced by the thicknesses of both beams

$$g = ||\boldsymbol{p}|| - \rho_1 - \rho_2, \tag{3.98}$$

where $\boldsymbol{p} = \boldsymbol{r}_1(s_1) - \boldsymbol{r}_2(s_2)$ and ρ_i is the cross-section radius of beam i. The geometry of the cross-section assumed in the contact formulation does not need to correspond exactly to the cross-section used in the beam theory. To relate the arc-length parameters s_i parametrising different centrelines, a projection map is used.

3.5.2 Contact potential and virtual work

Contribution of the frictionless contact to the total energy of the system can be defined by the contact potential as shown in [Wriggers and Simo, 1985]

$$\Pi_N = \int_{\Gamma_c} \lambda(s) g(s) \mathrm{d}s, \tag{3.99}$$

where Γ_c is the contact region and s is an arc-length parameter along the integration domain.

The virtual work arising from the contact potential is computed by varying this expression, obtaining

$$\delta\Pi_N = \int_{\Gamma_c} \delta\lambda g + \delta g \lambda \, \mathrm{d}s \tag{3.100}$$

where the explicit dependence of the fields λ and g on s has been omitted for clarity and the variation of the gap δg is yet to be determined. The gap function (3.98) is varied as

$$\begin{aligned}
\delta g(s) &= \delta \sqrt{\boldsymbol{p}^T \boldsymbol{p}} \\
&= \delta \boldsymbol{p}^T \frac{\boldsymbol{p}}{||\boldsymbol{p}||} \\
&= \left(\delta \left[\boldsymbol{r}_1(s)\right] - \delta \left[\boldsymbol{r}_2(s)\right]\right)^T \boldsymbol{n} \\
&= \left(\delta \boldsymbol{r}_1 + \frac{\mathrm{d}\boldsymbol{r}_1}{\mathrm{d}s_1}\delta s_1 - \delta \boldsymbol{r}_2 - \frac{\mathrm{d}\boldsymbol{r}_2}{\mathrm{d}s_2}\delta s_2\right)^T \boldsymbol{n},
\end{aligned} \tag{3.101}$$

where $\boldsymbol{n} = \boldsymbol{p}/||\boldsymbol{p}||$ is the contact normal and $\delta\boldsymbol{r}_1$, $\delta\boldsymbol{r}_2$ are the variations of the centreline positions.

We can now express the contact virtual work as

$$\delta\Pi_N = \int_{\Gamma_c} \delta\lambda g + (\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2)^T \boldsymbol{n}\lambda + \left(\boldsymbol{r}_1'\delta s_1 - \boldsymbol{r}_2'\delta s_2\right)^T \boldsymbol{n}\lambda \,\mathrm{d}s. \tag{3.102}$$

3.5.3 Mortar method

The mortar method is a discretisation method for contact between two, generally non-conforming, meshes. A mortar and a non-mortar side of the contact are assigned to allow simple parametrisation of the contact. A contact-force field is constructed using the existing domain discretisation on the non-mortar side.

Parametrisation

In the mortar method, the integration variable s simply corresponds to s_1 ($s = s_1$). The contact region Γ_c then reduces to the domain of beam 1, specifically, $s \in [0, L_1]$. This leads to the following development in the virtual work expression (3.102):

$$\delta\Pi_N = \int_0^{L_1} \delta\lambda g + (\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2)^T \boldsymbol{n}\lambda - \delta s_2 \boldsymbol{r}_2'^T \boldsymbol{n}\lambda \,\mathrm{d}s_1, \tag{3.103}$$

because the parameter s_1 is fixed its variation δs_1 disappears.

Projection

In the mortar method, the orthogonal projection is used via

$$\boldsymbol{p}^T \boldsymbol{r}_2' = 0, \tag{3.104}$$

which implied $\boldsymbol{r}_2'^T \boldsymbol{n} = 0$ and thus the virtual work in (3.103) becomes

$$\delta\Pi_N = \int_0^{L_1} \delta\lambda g + (\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2)^T \boldsymbol{n}\lambda \,\mathrm{d}s_1. \tag{3.105}$$

Note that δs_2 has disappeared from the expression, but it will come back with the linearisation. It is thus practical to derive it here and to explain its meaning, which follows from the interdependence between coordinates s_1 and s_2 based on the selected correlation equation - here the orthogonal projection (3.104). Because a contact relates two otherwise independent curves, its complete variation requires differentiation of both curves. The selected projection fixes the

arc-length parameter on one side making it effectively independent of the other curve. The dependent side, however, does not experience the same simplification. Its complete variation thus also consists of differentiation with respect to the other curve.

Variation of (3.104)

$$\left(\delta \boldsymbol{r}_2' + \boldsymbol{r}_2'' \delta s_2\right)^T \boldsymbol{p} + \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2 - \boldsymbol{r}_2' \delta s_2\right)^T \boldsymbol{r}_2' = 0$$

thus leads to

$$\delta s_2 = \frac{\left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{r}_2' + \delta \boldsymbol{r}_2'^T \boldsymbol{p}}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''}. \tag{3.106}$$

Linearisation

To apply the Newton-Raphson algorithm, we now linearise the contact virtual work (3.105):

$$\Delta(\delta \Pi_N) = \int_0^{L_1} \delta\lambda \Delta g \,\mathrm{d}s_1 + \int_0^{L_1} \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{n} \Delta\lambda \,\mathrm{d}s_1 + \int_0^{L_1} \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \Delta\boldsymbol{n} \lambda \,\mathrm{d}s_1 - \int_0^{L_1} \delta \boldsymbol{r}_2'^T \boldsymbol{n} \lambda \Delta s_2 \,\mathrm{d}s_1, \tag{3.107}$$

where all of the terms are already known except for $\Delta\boldsymbol{n}$. It can be simply computed as

$$\begin{aligned}
\Delta \boldsymbol{n} &= \Delta \left(\frac{\boldsymbol{p}}{||\boldsymbol{p}||} \right) \\
&= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||} \left(\Delta \left[\boldsymbol{r}_1(s_1) \right] - \Delta \left[\boldsymbol{r}_2(s_1) \right] \right) \\
&= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||} \left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 - \boldsymbol{r}_2' \Delta s_2 \right),
\end{aligned}$$

where $\hat{\cdot}$ maps to a skew symmetric matrix so that $\hat{\boldsymbol{a}}\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$ for $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$. Because $\boldsymbol{n}^T \boldsymbol{r}_2' = 0$ and $||\boldsymbol{n}|| = 1$, identity $\hat{\boldsymbol{n}}^2 \boldsymbol{r}_2' = -\boldsymbol{r}_2'$ applies allowing the following development

$$\begin{aligned}
\Delta \boldsymbol{n} &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||} \left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 \right) - \frac{1}{||\boldsymbol{p}||} \boldsymbol{r}_2' \Delta s_2 \\
&= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||} \left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 \right) - \frac{1}{||\boldsymbol{p}||} \boldsymbol{r}_2' \frac{\boldsymbol{r}_2'^T \left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 \right) + \boldsymbol{p}^T \Delta \boldsymbol{r}_2'}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''} \\
&= -\frac{1}{||\boldsymbol{p}||} \left[\hat{\boldsymbol{n}}^2 + \frac{\boldsymbol{r}_2' \boldsymbol{r}_2'^T}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''} \right] \left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 \right) - \frac{\boldsymbol{r}_2' \boldsymbol{n}^T}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''} \Delta \boldsymbol{r}_2'.
\end{aligned} \tag{3.108}$$

By making use of the following substitutions

$$\mathbf{A} = \frac{1}{||\boldsymbol{p}||} \left[\hat{\boldsymbol{n}}^2 + \frac{\boldsymbol{r}_2' \boldsymbol{r}_2'^T}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''} \right], \qquad \mathbf{B} = \frac{\boldsymbol{r}_2' \boldsymbol{n}^T}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''}, \qquad \mathbf{C} = \frac{\boldsymbol{n} \boldsymbol{p}^T}{\boldsymbol{r}_2'^T \boldsymbol{r}_2' - \boldsymbol{p}^T \boldsymbol{r}_2''} \tag{3.109}$$

the linearised virtual work (3.107) can be written as

$$\begin{aligned}\Delta(\delta\Pi_N) = &\int_0^{L_1} \delta\lambda \boldsymbol{n}^T \left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) \,\mathrm{d}s_1 + \int_0^{L_1} \left(\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2\right)^T \boldsymbol{n}\Delta\lambda \,\mathrm{d}s_1 \\ &- \int_0^{L_1} \left(\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2\right)^T \lambda \left[\mathbf{A}\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) + \mathbf{B}\Delta\boldsymbol{r}_2'\right] \,\mathrm{d}s_1 \\ &- \int_0^{L_1} \delta\boldsymbol{r}_2'^T \lambda \left[\mathbf{B}^T\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) + \mathbf{C}\Delta\boldsymbol{r}_2'\right] \,\mathrm{d}s_1.\end{aligned} \tag{3.110}$$

Discretisation

A beam is discretised by splitting the entire domain into subdomains corresponding to the space of the arc-length parameter of an individual finite element. Since the virtual work (3.50) is defined through integration along the entire domain, splitting it directly corresponds to summation across all elements. Each element thus behaves as a continuous beam, but with its primary variables defined through interpolation of nodal values. Since this procedure is standard and known as the assembly of the finite elements, we will simplify the notation by skipping the reference to a particular element and re-use the notation from the continuum.

Although many different interpolations are possible, here we limit our attention only to a separate interpolation of the displacement and the rotation field. For a reference on how to transform the formulation to use the $SE(3)$ interpolation in combination with the material frame based residuals we refer the reader to [Bosten et al., 2022]. Polynomial-based interpolation of the position field can be found in many elements [Simo and Vu-Quoc, 1986, Ibrahimbegovic, 1997, Crisfield and Jelenić, 1999]. Let us define it and its variation for some element on beam i as

$$\boldsymbol{r}_i = \mathbf{N}_i^T \boldsymbol{X}_i, \qquad \delta\boldsymbol{r}_i = \mathbf{N}_i^T \delta\boldsymbol{X}_i, \tag{3.111}$$

where $\mathbf{N}_i$ is a matrix of shape functions

$$\mathbf{N}_i^T = \begin{bmatrix} N_{i1} & 0 & 0 & N_{i2} & 0 & 0 & \dots & N_{iM_{\mathrm{nodes},i}} & 0 & 0 \\ 0 & N_{i1} & 0 & 0 & N_{i2} & 0 & \dots & 0 & N_{iM_{\mathrm{nodes},i}} & 0 \\ 0 & 0 & N_{i1} & 0 & 0 & N_{i2} & \dots & 0 & 0 & N_{iM_{\mathrm{nodes},i}} \end{bmatrix}, \tag{3.112}$$

where $M_{\mathrm{nodes},i}$ is the number of nodes of an element on beam i and N_{ij} is a polynomial from Lagrange basis for node j of this element. The size of the matrix depends on the interpolation order. The vector $\boldsymbol{X}_i$ is the vector of nodal displacements of an element on beam i. Together with its variation $\delta\boldsymbol{X}_i$ they are both elements of $\mathbb{R}^{3M_{\mathrm{nodes},i}}$.

The Lagrange-multiplier field discretisation follows the discretisation of beam 1. This in principle means that each beam element on beam 1 has a corresponding contact element with its own interpolation of the Lagrange-multiplier field

$$\lambda = \boldsymbol{\Phi}^T \boldsymbol{\Lambda}, \qquad \delta\lambda = \boldsymbol{\Phi}^T \delta\boldsymbol{\Lambda}, \tag{3.113}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{M_{\text{nodes},\Lambda}}$ is the vector of nodal Lagrange multipliers and

$$\boldsymbol{\Phi}^T = \begin{Bmatrix} \Phi_1 & \Phi_2 & \dots & \Phi_{M_{\text{nodes},\Lambda}} \end{Bmatrix}, \tag{3.114}$$

where $M_{\text{nodes},\Lambda}$ is the number of nodes of the contact element and Φ_j is a polynomial from Lagrange basis for node j of this element. The contact force

- domain remains non-overlapping even after discretisation as its parametrisation follows the non-mortar side beam so that each subdomain, defined by an element, is assigned a unique portion of the whole domain,

- is not necessarily continuous.

The discrete contact virtual work of an individual contact element is obtained by inserting the interpolated fields into (3.105)

$$\delta\Pi_N^{(el.)} = \delta\boldsymbol{\Lambda}^T \int_0^{L_1} \boldsymbol{\Phi} g \,\mathrm{d}s_1 + \delta\boldsymbol{X}_1^T \int_0^{L_1} \mathbf{N}_1 \boldsymbol{n} \lambda \,\mathrm{d}s_1 - \delta\boldsymbol{X}_2^T \int_0^{L_1} \mathbf{N}_2 \boldsymbol{n} \lambda \,\mathrm{d}s_1, \tag{3.115}$$

where L_1 and s_1 are the length and the coordinate of the corresponding beam 1 element. The whole domain is evaluated as the sum of all comprising elements

$$\delta\Pi_N = \sum_{(el.)} \delta\Pi_N^{(el.)}. \tag{3.116}$$

From (3.115) we can extract the residual vectors for individual contact element

$$\boldsymbol{R}_\lambda^{(el.)} = \int_0^{L_1} \boldsymbol{\Phi} g \,\mathrm{d}s_1, \qquad \boldsymbol{R}_1^{(el.)} = \int_0^{L_1} \mathbf{N}_1 \boldsymbol{n} \lambda \,\mathrm{d}s_1, \qquad \boldsymbol{R}_2^{(el.)} = -\int_0^{L_1} \mathbf{N}_2 \boldsymbol{n} \lambda \,\mathrm{d}s_1. \tag{3.117}$$

The linearised contact virtual work for an individual element (3.110) similarly becomes

$$\begin{aligned}
\Delta(\delta\Pi_N^{(el.)}) = {} & \delta\boldsymbol{\Lambda}^T \int_0^{L_1} \boldsymbol{\Phi}\boldsymbol{n}^T\mathbf{N}_1^T \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_1 - \delta\boldsymbol{\Lambda}^T \int_0^{L_1} \boldsymbol{\Phi}\boldsymbol{n}^T\mathbf{N}_2^T \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_2 \\
& + \delta\boldsymbol{X}_1^T \int_0^{L_1} \mathbf{N}_1\boldsymbol{n}\boldsymbol{\Phi}^T \,\mathrm{d}s_1\, \Delta\boldsymbol{\Lambda} - \delta\boldsymbol{X}_2^T \int_0^{L_1} \mathbf{N}_2\boldsymbol{n}\boldsymbol{\Phi}^T \,\mathrm{d}s_1\, \Delta\boldsymbol{\Lambda} \\
& - \delta\boldsymbol{X}_1^T \int_0^{L_1} \lambda\mathbf{N}_1\mathbf{A}\mathbf{N}_1^T \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_1 \\
& + \delta\boldsymbol{X}_1^T \int_0^{L_1} \lambda \left[\mathbf{N}_1\mathbf{A}\mathbf{N}_2^T - \mathbf{N}_1\mathbf{B}\mathbf{N}_2'^T\right] \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_2 \\
& + \delta\boldsymbol{X}_2^T \int_0^{L_1} \lambda \left[\mathbf{N}_2\mathbf{A}\mathbf{N}_1^T - \mathbf{N}_2\mathbf{B}^T\mathbf{N}_1'^T\right] \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_1 \\
& - \delta\boldsymbol{X}_2^T \int_0^{L_1} \lambda \left[\mathbf{N}_2\mathbf{A}\mathbf{N}_2^T - \mathbf{N}_2\mathbf{B}\mathbf{N}_2'^T - \mathbf{N}_2'\mathbf{B}^T\mathbf{N}_2^T + \mathbf{N}_2'\mathbf{C}\mathbf{N}_2'^T\right] \,\mathrm{d}s_1\, \Delta\boldsymbol{X}_2
\end{aligned} \tag{3.118}$$

from where we can extract the tangent stiffness matrices

$$\begin{aligned}
\mathbf{K}_{\lambda 1}^{(el.)} = \left(\mathbf{K}_{1\lambda}^{(el.)}\right)^T = \int_0^{L_1} \boldsymbol{\Phi}\boldsymbol{n}^T\mathbf{N}_1^T \,\mathrm{d}s_1, \qquad \mathbf{K}_{\lambda 2}^{(el.)} = \left(\mathbf{K}_{2\lambda}^{(el.)}\right)^T = -\int_0^{L_1} \boldsymbol{\Phi}\boldsymbol{n}^T\mathbf{N}_2^T \,\mathrm{d}s_1, \\
\mathbf{K}_{11}^{(el.)} = -\int_0^{L_1} \lambda\mathbf{N}_1\mathbf{A}\mathbf{N}_1^T \,\mathrm{d}s_1, \\
\mathbf{K}_{12}^{(el.)} = \int_0^{L_1} \lambda\mathbf{N}_1 \left[\mathbf{A}\mathbf{N}_2^T - \mathbf{B}\mathbf{N}_2'^T\right] \,\mathrm{d}s_1, \\
\mathbf{K}_{21}^{(el.)} = \int_0^{L_1} \lambda\mathbf{N}_2 \left[\mathbf{A}\mathbf{N}_1^T - \mathbf{B}^T\mathbf{N}_1'^T\right] \,\mathrm{d}s_1, \\
\mathbf{K}_{22}^{(el.)} = -\int_0^{L_1} \lambda \left[\mathbf{N}_2\mathbf{A}\mathbf{N}_2^T - \mathbf{N}_2\mathbf{B}\mathbf{N}_2'^T - \mathbf{N}_2'\mathbf{B}^T\mathbf{N}_2^T + \mathbf{N}_2'\mathbf{C}\mathbf{N}_2'^T\right] \,\mathrm{d}s_1.
\end{aligned} \tag{3.119}$$

A contact involves three different sets of the nodal degrees of freedom: two sets for each beam element and one set for the Lagrange multipliers. This is reflected in the subscripts of the residual vectors and stiffness matrices. For example, $\boldsymbol{R}_{\lambda}^{(el.)}$ refers to the vector of residual forces at the Lagrange multiplier nodes, while $\mathbf{K}_{21}^{(el.)}$ refers to the tangent stiffness matrix that is obtained when we linearise the residual vector of the second beam element with respect to the degrees of freedom of the first element.

3.5.4 Penalty method

Within the mathematical model, Lagrange multipliers provide the optimal way to enforce contact constraints. However, in some cases it is beneficiary to use a less strict enforcement of contact condition using the penalty method.

The mortar method may be adapted for use with a penalty approach in which case the contact force is defined as

$$\lambda = \varepsilon g \tag{3.120}$$

$$\varepsilon = \begin{cases} \varepsilon_0, & \text{if } g < 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.121}$$

where ε_0 is the penalty parameter. The contact potential (3.99) is multiplied with $1/2$ to simplify later expressions for the residual forces and thus becomes

$$\Pi_N = \frac{1}{2} \int_{\Gamma_c} \lambda g \mathrm{d}s. \tag{3.122}$$

Using the discretisation from the previous section, the contact-element residuals (3.117) eventually turn into

$$\boldsymbol{R}_1^{(el.)} = \int_0^{L_1} \mathbf{N}_1 \boldsymbol{n} \lambda \, \mathrm{d}s_1, \qquad \boldsymbol{R}_2^{(el.)} = -\int_0^{L_1} \mathbf{N}_2 \boldsymbol{n} \lambda \, \mathrm{d}s_1, \tag{3.123}$$

while the contact-element tangent stiffness matrices (3.119) turn into

$$\begin{aligned}
\mathbf{K}_{11}^{(el.)} &= -\int_0^{L_1} \lambda \mathbf{N}_1 \mathbf{A} \mathbf{N}_1^T - \varepsilon \mathbf{N}_1 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_1^T \, \mathrm{d}s_1, \\
\mathbf{K}_{12}^{(el.)} &= \int_0^{L_1} \lambda \mathbf{N}_1 \left[\mathbf{A} \mathbf{N}_2^T - \mathbf{B} \mathbf{N}_2'^T \right] - \varepsilon \mathbf{N}_1 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_2^T \, \mathrm{d}s_1, \\
\mathbf{K}_{21}^{(el.)} &= \int_0^{L_1} \lambda \mathbf{N}_2 \left[\mathbf{A} \mathbf{N}_1^T - \mathbf{B}^T \mathbf{N}_1'^T \right] - \varepsilon \mathbf{N}_2 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_1^T \, \mathrm{d}s_1, \\
\mathbf{K}_{22}^{(el.)} &= -\int_0^{L_1} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_2 \mathbf{B} \mathbf{N}_2'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_2^T + \mathbf{N}_2' \mathbf{C} \mathbf{N}_2'^T \right] - \varepsilon \mathbf{N}_2 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_2^T \, \mathrm{d}s_1.
\end{aligned} \tag{3.124}$$

Remark 5. *Elimination of the Lagrange-multiplier field transforms the mortar method using a weak enforcement of the contact conditions to a collocation method with a strong enforcement of the contact conditions at the integration points. The algorithm becomes similar to the Gauss-point-to-segment method developed by Chamek et al. [Chamekh et al., 2009] and Meier et al. [Meier et al., 2016].*

3.5.5 Unbiased line-to-line method

Now we present our new formulation developed to provide an unbiased distributed contact element, which does not discriminate between the two sides of the contact. This is achieved by coupling the beam elements prone to contact, where each pair of elements defines a particular contact element.

Parametrisation and projection

The design goal of the unbiased line-to-line contact formulation is to obtain reliable and unique results regardless of the choice of curves 1 and 2, which must be reflected also in the discretised form within the finite element method. For this purpose we propose the following definition of the parametric variable s (illustrated in Figure 3.2)

$$s = \frac{s_1 + s_2}{2} \tag{3.125}$$

with $s_i \in [0, L_i]$ for $i = 1, 2$ (where L_i is the undeformed arc-length of the centreline i), in combination with an appropriate projection function

$$\boldsymbol{p}^T \boldsymbol{t} = 0, \quad \boldsymbol{t} = \boldsymbol{r}_1' + \boldsymbol{r}_2'. \tag{3.126}$$

Parameters s_1 and s_2 are then computed by solving (3.125) and (3.126) using the Newton-Raphson method as explained in Appendix F.

Equations (3.125) and (3.126) are not only symmetric, thus providing the required indifference of the solution to the curve selection, but also provide a similar simplified virtual contact work as in the mortar method (3.105) as will be shown later in the section. The integration domain must contain the entire contact zone, i. e. the entire subdomain where $g(s) < 0$ must be included. Therefore, the integral boundaries must be accurately obtained. They are computed through projections, summarised in the following algorithm

Box 3.1: Determining integration boundaries

1. Compute projections at boundaries $s_2(s_1 = 0)$, $s_1(s_2 = 0)$, $s_2(s_1 = L_1)$ and $s_1(s_2 = L_2)$*. Store each projection as a pair of beam arc-length parameters $c_k = (s_{1k}, s_{2k})$.

   * *Although a projection is straightforward to compute using the algorithm explained in Appendix F, the correct solution is a global minimum which is not always the one that is obtained, depending on the initial value for the Newton-Raphson method.*

2. Either none or two of these projections can fall within the domain $s_{ik} \in [0, L_i]$ for $i = 1, 2$. Numerical errors might cause a different number of projections to be within the domain, too. If there is only one projection within the domain, another one, that is the closest to it, is added (if $s_{ik} \notin [0, L_i]$, then choose projection c_k so that $\min_k(|L_i/2 - s_{ik}|)$). Otherwise proceed.

3. If there are no projections within the domain $s_{ik} \in [0, L_i]$ for $i = 1, 2$, the possibility of contact between the two beams is disregarded. Otherwise, the integration domain is computed considering all valid projections as follows. The contact domain parameter s is first computed as $s_k = (s_{1k} + s_{2k})/2$. To handle different beam parametrisations, $\bar{s}_k = (s_{1k} - s_{2k})/2$ is also considered. The distance tensor is next constructed by computing $d_{kl} = |s_k - s_l|$ and $\bar{d}_{kl} = |\bar{s}_k - \bar{s}_l|$ for k and l going through all projections. The longest integration domain is then selected by finding the maximum value of the two distance tensors. Furthermore, if the maximum value is in $\bar{d}_{kl}$, parametrisation of one beam must be inverted.

4. Finally, the integration boundaries are chosen to correspond to the selected maximum value of the distance tensors.

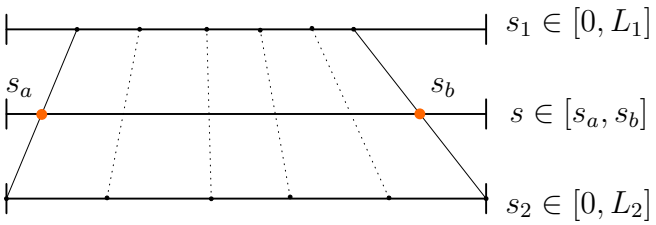


Figure 3.2: Illustration of the integration parameter s

With the integration boundaries determined, we can proceed with the derivation of the virtual

work. Since s is fixed, variation of (3.125) reduces to

$$\delta s_1 = -\delta s_2 \tag{3.127}$$

and the variation of the gap function (3.101) becomes

$$\begin{aligned}\delta g(s) &= \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2 + \frac{\mathrm{d}\boldsymbol{r}_1}{\mathrm{d}s_1}\delta s_1 + \frac{\mathrm{d}\boldsymbol{r}_2}{\mathrm{d}s_2}\delta s_1\right)^T \boldsymbol{n} \\ &= \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{n}. \end{aligned} \tag{3.128}$$

The virtual contact work follows from (3.102), however, since the integration limits s_a and s_b are variable, the Leibniz integral rule applies leading to

$$\delta\Pi_N = g(s_b)\lambda(s_b)\delta s_b - g(s_a)\lambda(s_a)\delta s_a + \int_{s_a}^{s_b} \delta\lambda g + \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{n}\lambda + \delta s_1 \left(\boldsymbol{r}_1' + \boldsymbol{r}_2'\right)^T \boldsymbol{n}\lambda \,\mathrm{d}s \tag{3.129}$$

and because of the zero-work constraint (3.97c) and the projection (3.126) it reduces to

$$\delta\Pi_N = \int_{s_a}^{s_b} \delta\lambda g + \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{n}\lambda \,\mathrm{d}s. \tag{3.130}$$

The variation of projection function (3.126) reads

$$\begin{aligned}\left(\delta\left[\boldsymbol{r}_1(s)\right] - \delta\left[\boldsymbol{r}_2(s)\right]\right)^T \boldsymbol{t} + \left(\delta\left[\boldsymbol{r}_1'(s)\right] + \delta\left[\boldsymbol{r}_2'(s)\right]\right)^T \boldsymbol{p} = 0 \\ \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2 + \boldsymbol{r}_1'\delta s_1 - \boldsymbol{r}_2'\delta s_2\right)^T \boldsymbol{t} + \left(\delta \boldsymbol{r}_1' + \delta \boldsymbol{r}_2' + \boldsymbol{r}_1''\delta s_1 + \boldsymbol{r}_2''\delta s_2\right)^T \boldsymbol{p} = 0,\end{aligned}$$

which after substituting (3.127) yields

$$\delta s_2 = \frac{\left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{t} + \left(\delta \boldsymbol{r}_1' + \delta \boldsymbol{r}_2'\right)^T \boldsymbol{p}}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T \boldsymbol{p}}. \tag{3.131}$$

Linearisation

The linearisation is performed as in the mortar method with the variation of projection function now being different. Linearised virtual work (3.130) is

$$\Delta(\delta\Pi_N) = \int_{s_a}^{s_b} \delta\lambda\Delta g \,\mathrm{d}s + \int_{s_a}^{s_b} \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \boldsymbol{n}\Delta\lambda \,\mathrm{d}s + \int_{s_a}^{s_b} \left(\delta \boldsymbol{r}_1 - \delta \boldsymbol{r}_2\right)^T \Delta\boldsymbol{n}\lambda \,\mathrm{d}s - \int_{s_a}^{s_b} \left(\delta \boldsymbol{r}_1' + \delta \boldsymbol{r}_2'\right)^T \boldsymbol{n}\lambda\Delta s_2 \,\mathrm{d}s. \tag{3.132}$$

Extra terms due to the Leibniz integration rule for variable integration boundaries are neglected because the algorithm to determine the boundaries is non-differentiable (non-smooth switches between projection $s_1(s_2 = 0)$ and $s_2(s_1 = 0)$ or $s_1(s_2 = L_2)$ and $s_2(s_1 = L_1)$, see Box 3.1). The normal vector is linearised in a similar way to (3.108), eventually leading to

$$\begin{aligned}\Delta\boldsymbol{n} &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||}\left(\Delta\left[\boldsymbol{r}_1(s)\right] - \Delta\left[\boldsymbol{r}_2(s)\right]\right) \\ &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||}\left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 + \boldsymbol{r}_1'\Delta s_1 - \boldsymbol{r}_2'\Delta s_2\right) \\ &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||}\left(\Delta \boldsymbol{r}_1 - \Delta \boldsymbol{r}_2 - \left(\boldsymbol{r}_1' + \boldsymbol{r}_2'\right)\Delta s_2\right).\end{aligned}$$

and using the $\hat{\boldsymbol{n}}^2\left(\boldsymbol{r}_1' + \boldsymbol{r}_2'\right) = -\boldsymbol{t}$ as in (3.108) we get

$$\begin{aligned}\Delta\boldsymbol{n} &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||}\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) - \frac{1}{||\boldsymbol{p}||}\boldsymbol{t}\Delta s_2 \\ &= -\frac{\hat{\boldsymbol{n}}^2}{||\boldsymbol{p}||}\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) - \frac{1}{||\boldsymbol{p}||}\boldsymbol{t}\frac{\boldsymbol{t}^T\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) + \boldsymbol{p}^T\left(\Delta\boldsymbol{r}_1' + \Delta\boldsymbol{r}_2'\right)}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}} \\ &= -\frac{1}{||\boldsymbol{p}||}\left[\hat{\boldsymbol{n}}^2 + \frac{\boldsymbol{t}\boldsymbol{t}^T}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}}\right]\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) - \frac{\boldsymbol{t}\boldsymbol{n}^T}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}}\left(\Delta\boldsymbol{r}_1' + \Delta\boldsymbol{r}_2'\right)\end{aligned} \tag{3.133}$$

from where we can similarly to (3.109) recognise the following three matrices

$$\mathbf{A} = \frac{1}{||\boldsymbol{p}||}\left[\hat{\boldsymbol{n}}^2 + \frac{\boldsymbol{t}\boldsymbol{t}^T}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}}\right], \qquad \mathbf{B} = \frac{\boldsymbol{t}\boldsymbol{n}^T}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}}, \qquad \mathbf{C} = \frac{\boldsymbol{n}\boldsymbol{p}^T}{\boldsymbol{t}^T\boldsymbol{t} + \left(\boldsymbol{r}_1'' - \boldsymbol{r}_2''\right)^T\boldsymbol{p}}. \tag{3.134}$$

We can insert the results directly into (3.132) to obtain

$$\begin{aligned}\Delta(\delta\Pi_N) = &\int_{s_a}^{s_b}\delta\lambda\boldsymbol{n}^T\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right)\,\mathrm{d}s + \int_{s_a}^{s_b}\left(\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2\right)^T\boldsymbol{n}\Delta\lambda\,\mathrm{d}s \\ &- \int_{s_a}^{s_b}\left(\delta\boldsymbol{r}_1 - \delta\boldsymbol{r}_2\right)^T\lambda\left[\mathbf{A}\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) + \mathbf{B}\left(\Delta\boldsymbol{r}_1' + \Delta\boldsymbol{r}_2'\right)\right]\,\mathrm{d}s \\ &- \int_{s_a}^{s_b}\left(\delta\boldsymbol{r}_1' + \delta\boldsymbol{r}_2'\right)^T\lambda\left[\mathbf{B}^T\left(\Delta\boldsymbol{r}_1 - \Delta\boldsymbol{r}_2\right) + \mathbf{C}\left(\Delta\boldsymbol{r}_1' + \Delta\boldsymbol{r}_2'\right)\right]\,\mathrm{d}s.\end{aligned} \tag{3.135}$$

Discretisation

Beam discretisation is already defined in (3.111) and (3.112). Accordingly, the contact integral (3.130) must be divided into subdomains to allow numerical integration. The integral is defined using the integration parameter s which depends on the arc-length parameters of the beams. This forces us to use segment-based integration which, on the positive side, avoids integrating over discontinuities but suffers from increased computational cost [Farah et al., 2015]. Each segment defines a contact element. The segments are constructed by coupling the beam elements that are prone to contact and evaluating their individual contact zone using the algorithm in Box 3.1.

Unfortunantely, the discretised centrelines are only C^0 continuous so the contact segments are not necessarily continuously defined along the contact, see Figures 3.4 and 3.5. This means that the discretised Lagrange-multiplier field must be defined individually per element and cannot be connected with neighbouring elements, resulting in a non-conforming Lagrange-multiplier field. Compared to the standard mortar method, this leads to an increased number of Lagrange multipliers' degrees of freedom and possible over-constraining, which can be avoided using the constant Lagrange multipliers' field interpolation. Since beams are often interpolated with higher

order polynomials, the whole affair could also be remedied using $\mathcal{C}^{(1)}$ continuous interpolation of displacement, e. g. [Litewka, 2007], as it would result in a continuous contact zone in the discretised setting. This would result in a framework where contact elements have a smooth transition between them and would furthermore allow $\mathcal{C}^{(0)}$ continuous discretisation of the Lagrange multiplier field. However, this is left for future research.

As a result, each unbiased contact element couples a pair of beam elements with independent Lagrange multiplier nodes. This is in contrast to the mortar method where a contact element couples a single non-mortar-side beam element to potentially multiple mortar-side beam elements.

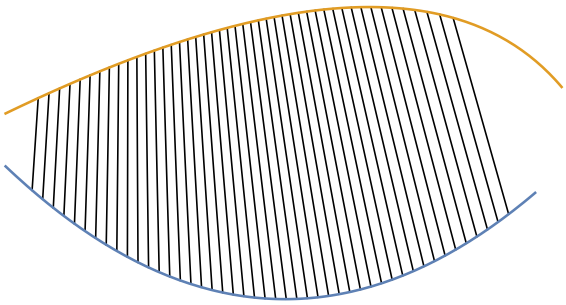


Figure 3.3: Continuous contact formulation

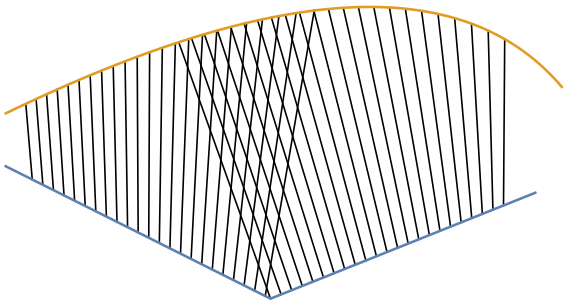


Figure 3.4: Discretised contact

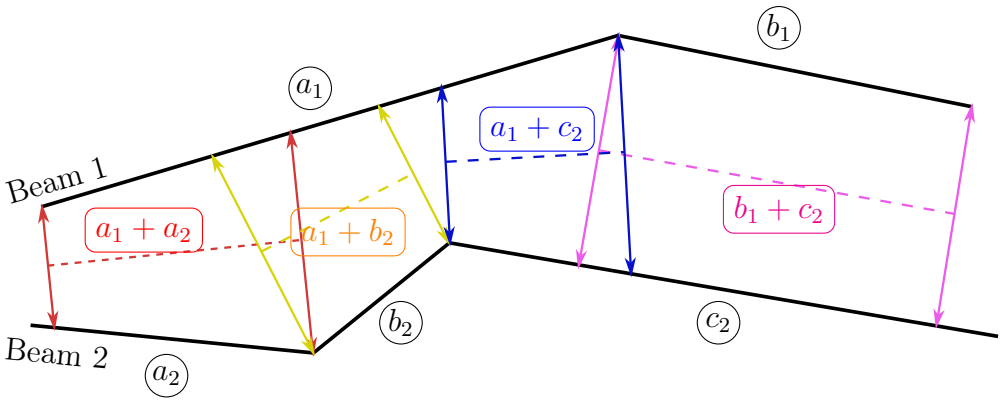


Figure 3.5: Example of contact discretisation with unbiased line-to-line elements

The contact element is, despite different discretisation, almost the same as in the mortar method. The Lagrange-multiplier field is interpolated as in (3.113), the nodes however now being specific to each element and not shared by the neighbouring elements. The resulting residual vectors and tangent matrices are also similar to (3.117) and (3.119). The discrete virtual work is obtained by inserting (3.111) and (3.113) into (3.130) and reads

$$\delta\Pi_N^{(el.)} = \delta\boldsymbol{\Lambda}^T \int_{s_a}^{s_b} \boldsymbol{\Phi} g \,\mathrm{d}s + \delta\boldsymbol{X}_1^T \int_{s_a}^{s_b} \mathbf{N}_1 \boldsymbol{n} \lambda \,\mathrm{d}s - \delta\boldsymbol{X}_2^T \int_{s_a}^{s_b} \mathbf{N}_2 \boldsymbol{n} \lambda \,\mathrm{d}s, \tag{3.136}$$

which differs from (3.115) only in the integration variable and domain, and from where we can recognise the residual vectors

$$\boldsymbol{R}_{\lambda}^{(el.)} = \int_{s_a}^{s_b} \boldsymbol{\Phi} g \,\mathrm{d}s, \qquad \boldsymbol{R}_{1}^{(el.)} = \int_{s_a}^{s_b} \mathbf{N}_1 \boldsymbol{n} \lambda \,\mathrm{d}s, \qquad \boldsymbol{R}_{2}^{(el.)} = -\int_{s_a}^{s_b} \mathbf{N}_2 \boldsymbol{n} \lambda \,\mathrm{d}s. \tag{3.137}$$

The linearised virtual work (3.135) similarly becomes

$$\begin{aligned}
\Delta(\delta\Pi_N^{(el.)}) = {} & \delta\boldsymbol{\Lambda}^T \int_{s_a}^{s_b} \boldsymbol{\Phi}\boldsymbol{n}^T \mathbf{N}_1^T \,\mathrm{d}s \Delta\boldsymbol{X}_1 - \delta\boldsymbol{\Lambda}^T \int_{s_a}^{s_b} \boldsymbol{\Phi}\boldsymbol{n}^T \mathbf{N}_2^T \,\mathrm{d}s \Delta\boldsymbol{X}_2 \\
& + \delta\boldsymbol{X}_1^T \int_{s_a}^{s_b} \mathbf{N}_1 \boldsymbol{n} \boldsymbol{\Phi}^T \,\mathrm{d}s \Delta\boldsymbol{\Lambda} - \delta\boldsymbol{X}_2^T \int_{s_a}^{s_b} \mathbf{N}_2 \boldsymbol{n} \boldsymbol{\Phi}^T \,\mathrm{d}s \Delta\boldsymbol{\Lambda} \\
& - \delta\boldsymbol{X}_1^T \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_1 \mathbf{B} \mathbf{N}_1'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_1^T + \mathbf{N}_1' \mathbf{C} \mathbf{N}_1'^T\right] \,\mathrm{d}s \Delta\boldsymbol{X}_1 \\
& + \delta\boldsymbol{X}_1^T \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_1 \mathbf{B} \mathbf{N}_2'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_2^T - \mathbf{N}_1' \mathbf{C} \mathbf{N}_2'^T\right] \,\mathrm{d}s \Delta\boldsymbol{X}_2 \\
& + \delta\boldsymbol{X}_2^T \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_2 \mathbf{B} \mathbf{N}_1'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_1^T - \mathbf{N}_2' \mathbf{C} \mathbf{N}_1'^T\right] \,\mathrm{d}s \Delta\boldsymbol{X}_1 \\
& - \delta\boldsymbol{X}_2^T \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_2 \mathbf{B} \mathbf{N}_2'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_2^T + \mathbf{N}_2' \mathbf{C} \mathbf{N}_2'^T\right] \,\mathrm{d}s \Delta\boldsymbol{X}_2,
\end{aligned} \tag{3.138}$$

from where the contact-element tangent stiffness matrices follow as

$$\begin{aligned}
\mathbf{K}_{\lambda 1}^{(el.)} = \left(\mathbf{K}_{1\lambda}^{(el.)}\right)^T = \int_{s_a}^{s_b} \boldsymbol{\Phi}\boldsymbol{n}^T \mathbf{N}_1^T \,\mathrm{d}s, \qquad \mathbf{K}_{\lambda 2}^{(el.)} = \left(\mathbf{K}_{2\lambda}^{(el.)}\right)^T = -\int_{s_a}^{s_b} \boldsymbol{\Phi}\boldsymbol{n}^T \mathbf{N}_2^T \,\mathrm{d}s, \\
\mathbf{K}_{11}^{(el.)} = -\int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_1 \mathbf{B} \mathbf{N}_1'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_1^T + \mathbf{N}_1' \mathbf{C} \mathbf{N}_1'^T\right] \,\mathrm{d}s, \\
\mathbf{K}_{12}^{(el.)} = \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_1 \mathbf{B} \mathbf{N}_2'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_2^T - \mathbf{N}_1' \mathbf{C} \mathbf{N}_2'^T\right] \,\mathrm{d}s, \\
\mathbf{K}_{21}^{(el.)} = \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_2 \mathbf{B} \mathbf{N}_1'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_1^T - \mathbf{N}_2' \mathbf{C} \mathbf{N}_1'^T\right] \,\mathrm{d}s, \\
\mathbf{K}_{22}^{(el.)} = -\int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_2 \mathbf{B} \mathbf{N}_2'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_2^T + \mathbf{N}_2' \mathbf{C} \mathbf{N}_2'^T\right] \,\mathrm{d}s.
\end{aligned} \tag{3.139}$$

Penalty method

The penalty method can be introduced very similarly to the mortar method, see (3.120), (3.121) and (3.122). The unbiased contact-element residuals (3.137) eventually become

$$\boldsymbol{R}_1^{(el.)} = \int_{s_a}^{s_b} \mathbf{N}_1 \boldsymbol{n} \lambda \, \mathrm{d}s, \qquad \boldsymbol{R}_2^{(el.)} = -\int_{s_a}^{s_b} \mathbf{N}_2 \boldsymbol{n} \lambda \, \mathrm{d}s. \tag{3.140}$$

and its tangent stiffness matrices (3.139) become

$$\begin{aligned}
\mathbf{K}_{11}^{(el.)} &= -\int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_1 \mathbf{B} \mathbf{N}_1'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_1^T + \mathbf{N}_1' \mathbf{C} \mathbf{N}_1'^T \right] - \varepsilon \mathbf{N}_1 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_1^T \, \mathrm{d}s, \\
\mathbf{K}_{12}^{(el.)} &= \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_1 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_1 \mathbf{B} \mathbf{N}_2'^T + \mathbf{N}_1' \mathbf{B}^T \mathbf{N}_2^T - \mathbf{N}_1' \mathbf{C} \mathbf{N}_2'^T \right] - \varepsilon \mathbf{N}_1 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_2^T \, \mathrm{d}s, \\
\mathbf{K}_{21}^{(el.)} &= \int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_1^T + \mathbf{N}_2 \mathbf{B} \mathbf{N}_1'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_1^T - \mathbf{N}_2' \mathbf{C} \mathbf{N}_1'^T \right] - \varepsilon \mathbf{N}_2 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_1^T \, \mathrm{d}s, \\
\mathbf{K}_{22}^{(el.)} &= -\int_{s_a}^{s_b} \lambda \left[\mathbf{N}_2 \mathbf{A} \mathbf{N}_2^T - \mathbf{N}_2 \mathbf{B} \mathbf{N}_2'^T - \mathbf{N}_2' \mathbf{B}^T \mathbf{N}_2^T + \mathbf{N}_2' \mathbf{C} \mathbf{N}_2'^T \right] - \varepsilon \mathbf{N}_2 \boldsymbol{n}\boldsymbol{n}^T \mathbf{N}_2^T \, \mathrm{d}s.
\end{aligned} \tag{3.141}$$

3.5.6 Active set strategy

Active set strategy is used to determine which nodes participate in the contact. For the activation of nodes, residual vectors $\boldsymbol{R}_\lambda$ from (3.117) (mortar) and (3.137) (unbiased) are used as they conveniently correspond to the weighted gap. The activation criteria then reads

$$\sum_{\substack{(el.) \text{ if} \\ (n) \in (el.)}} R_\lambda^{(n,el.)} < 0, \tag{3.142}$$

where (n) denotes a particular node. Once the node is activated, the residual starts to contribute to the global solution and goes to zero as the iterative procedure converges. For deactivation, the value of the Lagrange multiplier at a node is checked. If it is positive (representing an attractive force), the node is deactivated.

Chapter 4

Numerical examples

I have tested the developed finite elements extensively, which is also reported in the produced papers [Tomec and Jelenić, 2022, Tomec and Jelenić, 2024a, Tomec and Jelenić, 2024b]. To test the dynamic formulation, I have chosen an existing benchmark test which features large rotations and long-term free-flight motion. The contact formulations have been subjected to several tests using a numerical framework, specifically programmed for this problem. It enabled me to closely compare specific aspects of contact methods.

4.1 Momentum and near-energy conservative beam element

4.1.1 Flying spaghetti

This examples is taken from [Simo and Vu-Quoc, 1988]. An impulse initiates the motion of a free-flying beam, as presented in Figure 4.1. The impulse is generated by a generalised force $\left\{F_x, 0, 0, 0, M_y, M_z\right\}$, where $F_x = M_z/10$, $M_y = M_z/2$ and M_z is given as a function of time in Figure 4.2. The following are the material parameters: $\rho A = 1$, $\rho I = 10$, $EA = GA = 10^4$, $GJ_t = EI = 500$. All integrals are computed using the Gaussian quadrature rule with the same number of integration points as the number of nodes per element ('*full integration*'). The motion of the beam at various times, revealing its large deformations and finite rotations, is displayed in Figure 4.3.

The mechanical energy and generalised momentum are expected to grow during the loading part due to the work of external forces. In the second part (load-free flight), the energy and momentum are expected to remain constant as no external forces work on the system. The

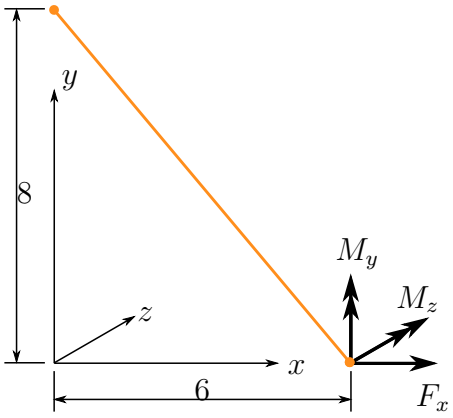


Figure 4.1: Flying spaghetti: setup

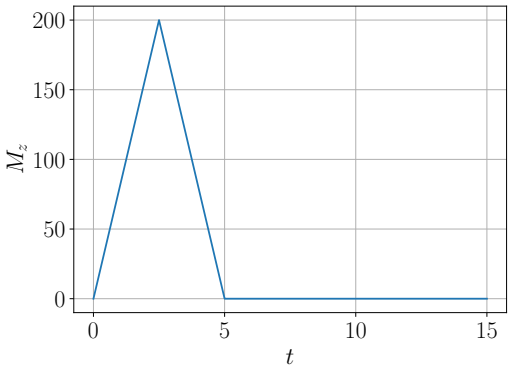


Figure 4.2: Flying spaghetti: Time dependency of load

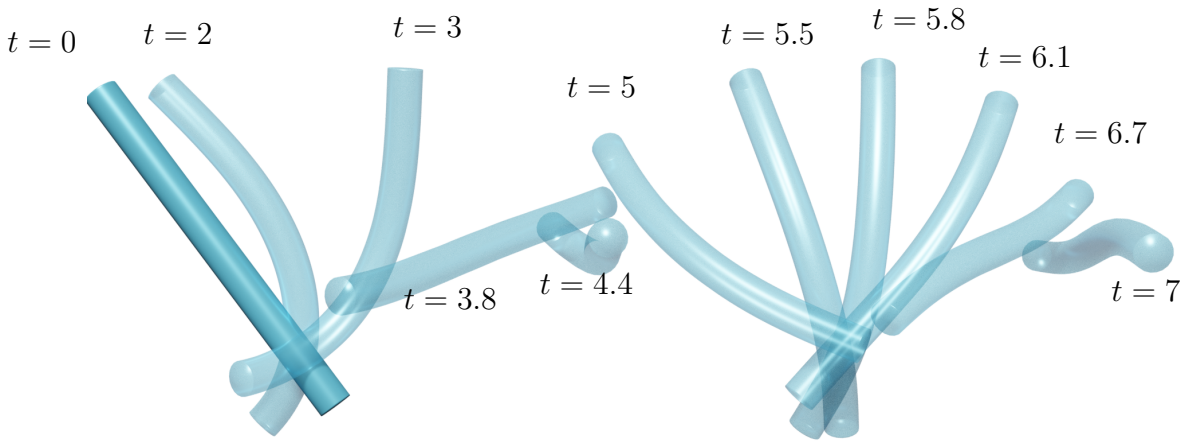


Figure 4.3: Flying spaghetti: Trace of deformed configurations at given times for 10 first-order elements and $\Delta t = 0.1$

generalised momentum is plotted in Figure 4.4 where we can observe conservation of momentum during the flight. The combined kinetic and potential energy is plotted in Figure 4.7 for different spatial discretisations with time step 0.1. As expected from the theory, we can observe deviations from the constant value for coarser meshes, however, with mesh refinement the energy quickly converges towards a constant value.

This is shown also in Figure 4.8, which plots standard deviation of energy during the flight for different mesh sizes. From the standard deviation, the magnitude of the absolute error can be estimated as one σ, which is where, for large-enough sets, lay 68 % of all samples. With the average value of the mechanical energy in the unloaded portion of the test, of 723, the relative error can be estimated as $\varepsilon_{\text{rel.}} = 10^{\log_2(N^{-0.85}/2)}$, where N is the number of nodes in the mesh. For an example, for the case using 4 second-order elements, the relative error is 4×10^{-5}.

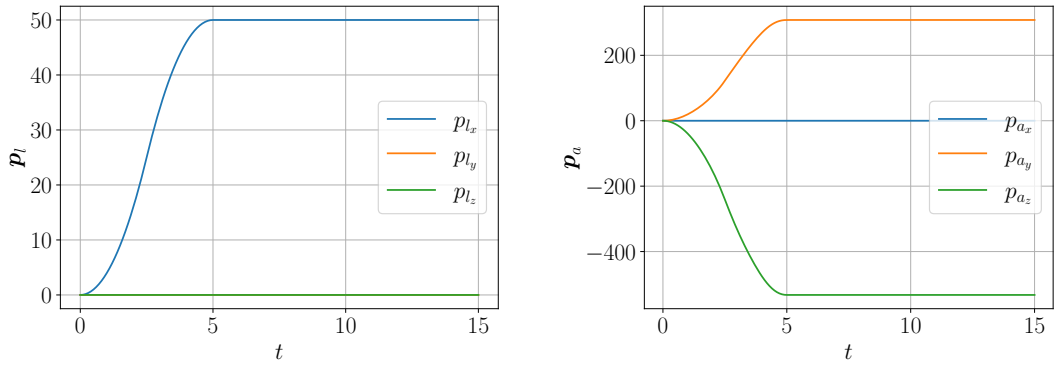
To test convergence with respect to the time-step size, two meshes, ten first-order elements

and ten second-order elements, have been subjected to $t \in [0, 6]$ analysis using different time steps. The convergence is plotted in Figure 4.5. Since no analytical solution to the problem exists, the solution computed with the smallest time step 2^{-8} at a given discretisation is taken as the reference solution. The error is defined as the norm of the difference in displacement for the loaded end node compared to the reference solution at $t_t = 6$

$$\epsilon = \text{norm}(\boldsymbol{r}_L(t_t) - \boldsymbol{r}_{L,\text{ref}}(t_t)). \tag{4.1}$$

The simulation has been run with the integration parameters $\tau = 1$ and $\tau = 0.7$. The order of convergence has been estimated from the sequence for $\tau = 1$ cases as 2.02 which is expected for the midpoint rule. When using the energy decay mechanism the convergence order is reduced as expected for this sort of energy decaying mechanism - we have measured it at 1.57.

Figure 4.6 shows a comparison of mechanical energy over time for different values of τ, which introduces damping. The results are obtained using 4 second-order elements and $\Delta t = 0.1$.



(a) Linear momentum

(b) Angular momentum

Figure 4.4: Flying spaghetti: Plot of linear and angular momentum for two first-order elements with time step 0.1. Components of linear momentum p_{l_y} and p_{l_z} and angular momentum p_{a_x} are zero.

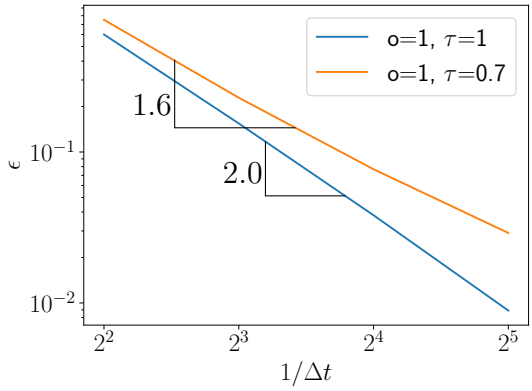


Figure 4.5: Convergence with respect to the time-step size using 10 first-order elements

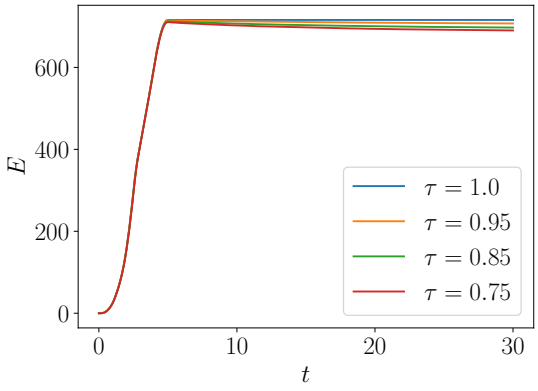


Figure 4.6: Comparison of energy decay for different values of parameter τ

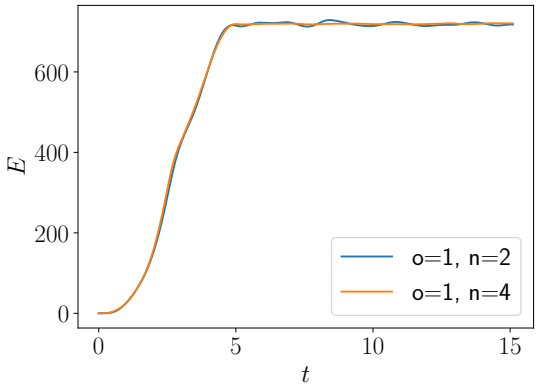


Figure 4.7: Flying spaghetti: Plot of kinetic and potential energy for two different meshes with two and four first-order elements

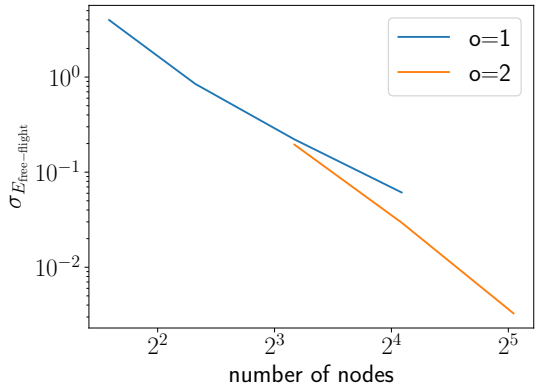


Figure 4.8: Flying spaghetti: Convergence of the load-free flight phase mechanical energy towards a constant value for different discretisations with ten first and ten second-order elements

4.2 Beam-to-beam contact

This section puts the presented contact formulations to test to assess their properties and compare them. In all examples geometrically-exact beam elements developed by Simo and Vu-Quoc [Simo and Vu-Quoc, 1986] are used. Abbreviations for different contact elements are summarised in Table 4.1. Both formulations are designed for arbitrary order of interpolation of Lagrange multiplier field.

Table 4.1: Abbreviations for contact elements

Abbreviation	Method
M2	mortar method with two nodes per element,
M1	mortar method with a single node per element,
MP	mortar method with penalty method,
U1	unbiased line-to-line method a single node per element,
UP	unbiased line-to-line method with penalty method.

4.2.1 Patch test

A patch test is conducted to test line-to-line method's ability to compute the exact gap value and to handle activation of nodes during sliding. A beam consisting of two elements is positioned above a rigid surface simulated by a three first-order-element beam with fixed degrees of freedom. The problem geometry is given in Figure 4.9. Both beams have a circular cross-section with radius $\rho = 0.005$ and material properties $E = 210 \times 10^9$ and $\nu = 0.3$. An external uniformly distributed loading $q = 1$ is applied to the top beam along the entire length of the beam. In the first load step, an equilibrium in contact is sought, while the next 100 steps consist of pushing the top beam horizontally along the bottom beam by a step $\Delta x = 0.01$. The tolerance is set to 10^{-12} for the energy norm.

The exact solution to this problem is trivial, i.e. the gap must remain equal to zero throughout the analysis. Numerical solution of the problem using the mortar method is presented in Figure 4.10. The gap error is computed as the norm of all computed values of gap in a converged solution. Using 4 integration points per element, the values are close to zero for all Lagrange-multiplier based methods as summarised in table 4.2. The non-penetration condition is thus fulfilled within the numerical machine precision. The mortar method is stable during sliding due to the weak fulfilment of the non-penetration condition. All Newton-Raphson loops require a single iteration to converge.

For the unbiased method with a single Lagrange-multiplier node per element, the results

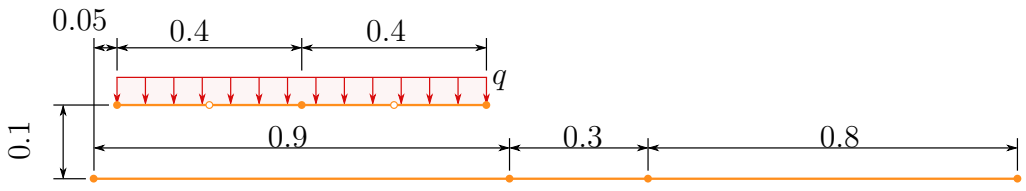


Figure 4.9: Patch test: setup

Table 4.2: Patch test: Norm of gap evaluated at integration points after sliding

Method	M2	M1	U1
gap	3.88×10^{-18}	3.47×10^{-18}	2.78×10^{-16}

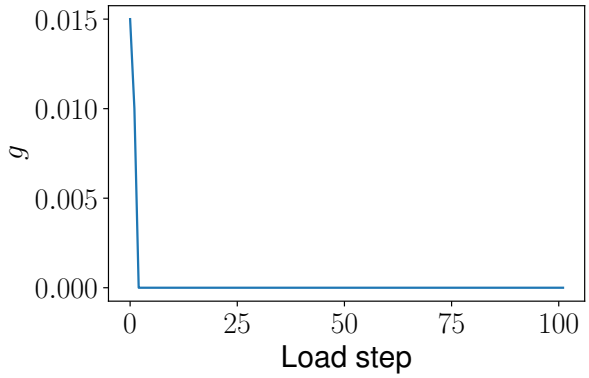


Figure 4.10: Patch test: average gap during load steps using the mortar method

are also close to zero. With this we can confirm that the unbiased method passes the patch test for zero-order interpolation. For higher-order interpolations of the Lagrange-multiplier field, the Newton-Raphson method experiences instabilities during sliding and consequently diverges. It is a consequence of discontinuous Lagrange multiplier interpolation which results in poorly conditioned tangent stiffness matrix. It also hints to shortcomings in the algorithm for determination of the integration boundaries, where the main complexity lies in handling the variety of possible special cases due to numerical imprecision. The unbiased line-to-line method therefore best works in combination with a constant Lagrange-multiplier interpolation.

Comparing mortar method to the penalty method (see also [Meier et al., 2016], where the penalty method is also used), the results presents a considerable improvement. With the penalty method some penetration, proportional to the chosen parameters, is expected. Secondly, the numerical results in the original study are dependent on the number of integration points. Even with increasing the number of the integration points, the solution does not converge to the expected solution of -0.002 (a solution expected for the chosen penalty parameters). Furthermore, the gap varies with load steps due to sliding. Both problems have been averted with the mortar method. We can see in Figure 4.10 that the gap is stable and constant with load steps and sliding appearing to have no effect on the result. The exact solution is reached already with 2 Gauss points.

In [Bosten et al., 2022] a simplified version of this test has been studied. Two beams of equal length but different discretization are pressed together. They arrive to the same conclusion that

the gap equals to zero within the machine precision and that the system converges in a single Newton-Raphson iteration. The effects of the discontinuities due to sliding have not been tested in [Bosten et al., 2022].

4.2.2 Contact objectivity test

The following test has been designed to test objectivity of a line-to-line contact formulation with respect to beam numbering. It has been specifically designed to emphasize the bias in the selection of the contact sides. Two horizontal cantilever beams are placed in parallel one above the other at the distance $2\rho = 0.006$ between. They are fixed on the same side. A vertical point force is applied to the top beam at its free end to bend it towards the bottom beam to create a contact which displaces also the other beam. Both beams are made of the same material with parameters $EA = 13351$, $GA_1 = 4982$, $GA_2 = 4982$, $GI_t = 0.0623$, $EI_1 = 0.08345$, $EI_2 = 0.08345$ and with a circular cross-section with radius $\rho = 0.003$. Both beams are of initial length $L = 0.05$. To pronounce the bias we use only two first-order geometrically exact beam elements for each beam. Four integration points are used for each contact element. The load is gradually increased in 7 quasi-static steps to the final magnitude of 42.

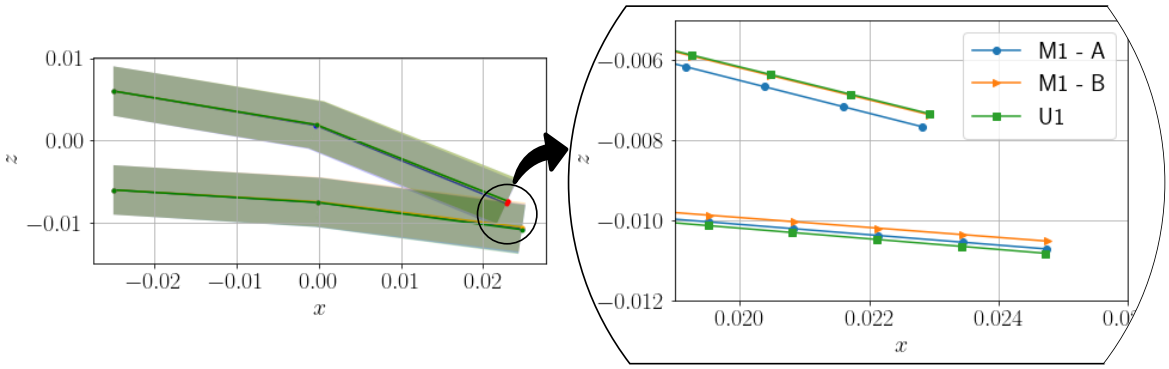


Figure 4.11: Objectivity: Final deformed state of the dual-beam system computed using the mortar and the unbiased formulation.

This test is conducted using the mortar and the unbiased line-to-line formulation twice (runs A and B), with the contact sides toggled in the two runs. Displacements of the top beam's free end are given in Table 4.3. Comparison of the displacements after changing the sides in the contact algorithm confirms that the new formulation is free from bias, unlike the mortar method which shows a certain amount of discrepancies. Deformation of the entire setup is illustrated in Figure 4.11.

Table 4.3: Objectivity: Displacements of the free end of the top beam

	M1		U1	
	$x_{\text{top end}}$	$z_{\text{top end}}$	$x_{\text{top end}}$	$z_{\text{top end}}$
A	0.02281696	−0.00766457	0.02293061	−0.0073395
B	0.02292918	−0.00736453	0.02293061	−0.0073395
B-A	0.00011222	0.00030005	3.469×10^{-18}	4.337×10^{-18}

4.2.3 Rotating beams

Two cantilever beams are placed one above the other at a certain angle. The bottom beam is collinear with the x-axis and completely fixed. The top beam lays initially in the plane x-y at an angle α around the z axis like shown in Figure 4.12. The beams are initially touching, so if the position of the top beam on z-axis is at H, the contact radius of each beam is $\rho = H/2$. The displacement of the free end of the top beam is free only in the z-axis. The rotations of the free end are not constrained. The free end is loaded with a point force in the z-axis. The appropriate size of the point force is such that if the contact was not considered, the two centrelines would intersect in the middle as shown in Figure 4.13. The size of the force is computed using the same setup only without contact constraints. This ensures maximum possible penetration for a given setup. Material parameters are: $EA = GA = 1$ and $GI = EI = 10$.

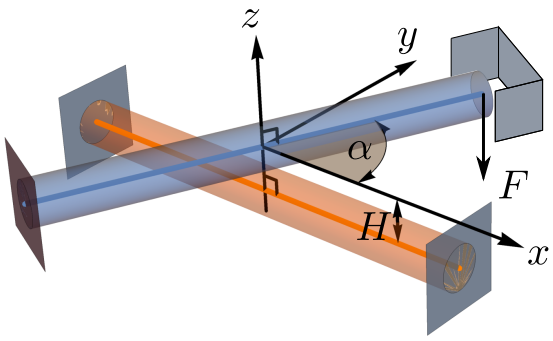


Figure 4.12: Rotating beams: setup

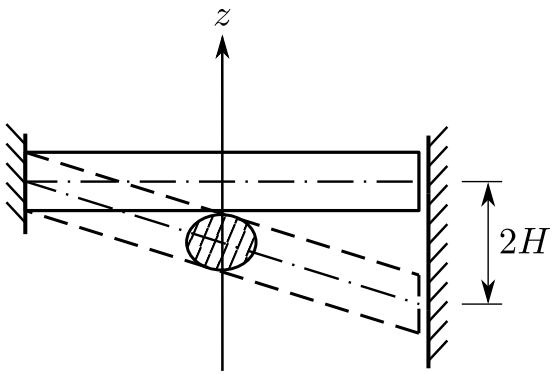


Figure 4.13: Rotating beams: maximum penetration

Each beam consists of only a single linear element with 3 integration points, which reduces the total number of degrees of freedom to 5 (4 for movement of the only free node and 1 for its Lagrange multiplier). The contact radius is being varied in order to determine the smallest possible ratio between the thickness of an element and its length, at which the contact is still detectable by the mortar method.

The test has been first analysed analytically. Evaluating equation (3.98) for the deformed configuration without contact detection as shown in Figure 4.13 yields the following gap function

$$g(s_{\text{top}}) = \sqrt{(1-2s_{\text{top}})^2(H^2+\sin^2(\alpha))}. \tag{4.2}$$

This gap is integrated according to (3.142) for the free node giving the following expression

$$\frac{1}{4}\left(-2H+\sqrt{H^2+\sin^2(\alpha)}\right) < 0. \tag{4.3}$$

When this expression equals zero, the condition is just before switching the Lagrange-multiplier degree of freedom. Solved for H, it defines the limit at which the contact will be detected for a given configuration, which is

$$H = \frac{\sin(\alpha)}{\sqrt{3}}. \tag{4.4}$$

The test has been conducted also numerically for values of α from 0 to π with a step $\pi/60$ and H for values 0.06 to 1.99 with a step 0.06. This produces a matrix of results which can be seen in Figure 4.14. A unified measure $(\rho^{(1)}+\rho^{(2)})/L^{(1)}$ is introduced, where index (1) is associated with the non-mortar element (blue in Figure 4.12) and (2) with the mortar element (orange in the same figure). Successfully converged results with a contact detected are coloured green while the rest remains white. Some results can converge to an unphysical solution, but this can be averted by introducing the load in several load steps and is as such not a concern of this test case. The contour is cross-plotted with the analytical solution for the minimal detectable radius (4.4). Numerical and analytical results match as expected.

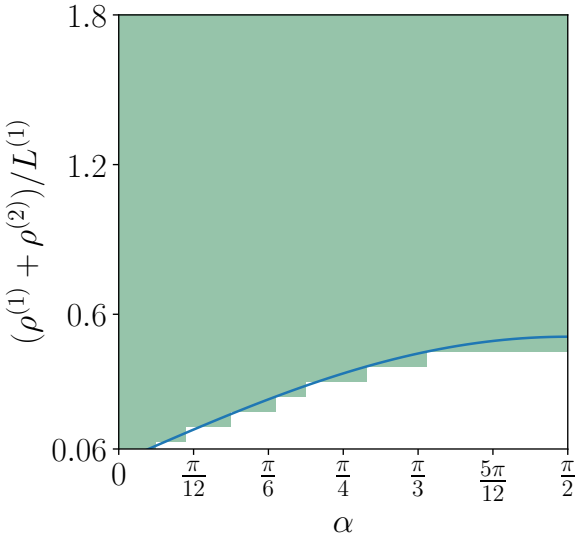


Figure 4.14: Rotating beams: minimal radius still to detect contact

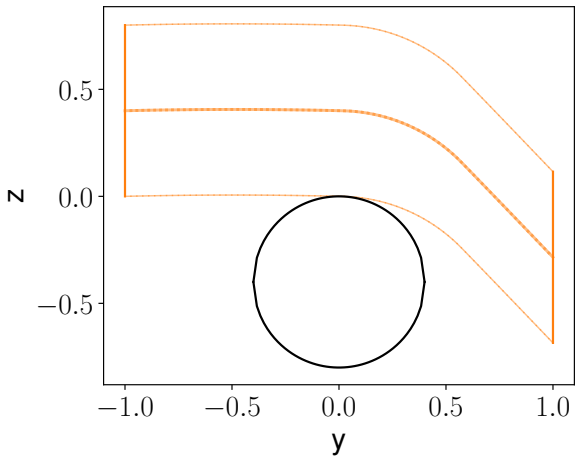


Figure 4.15: Rotating beams: deformed configuration at $\alpha = \pi/2$ with 81 elements

In a general situation, a contact can happen anywhere on the element. But if all nodes can become active, having the contact point exactly in the middle of the element is weighted by the shape functions the least and moving it towards some node would result in a quicker detection of a contact. So in some special cases, the contact can be detected even with smaller ratio of $(\rho^{(1)} + \rho^{(2)})/L^{(1)}$. In general, the worst case scenario must govern the choice for the size of the elements, which is predicted by this example.

Finally, the convergence of the mortar method for $\alpha = \pi/2$ and $H = 1$ has been tested. The load size is computed as above but is now added in 4 equal steps. Only the meshes with an odd number of the first-order elements have been tested to avoid a direct nodal contact. The final deformed configuration for the example using 81 elements is shown in Figure 4.15. Lagrange-multiplier field convergence is presented in Figure 4.16. The force distribution converges towards a step function with some boundary effects. A silhouette of the bottom-beam's cross section is overlaid to better illustrate the effective width of the contact. The corresponding gap functions can be seen in Figure 4.17. The gap is flattening to the same width as observed for the force. Some penetration is present for coarser meshes.

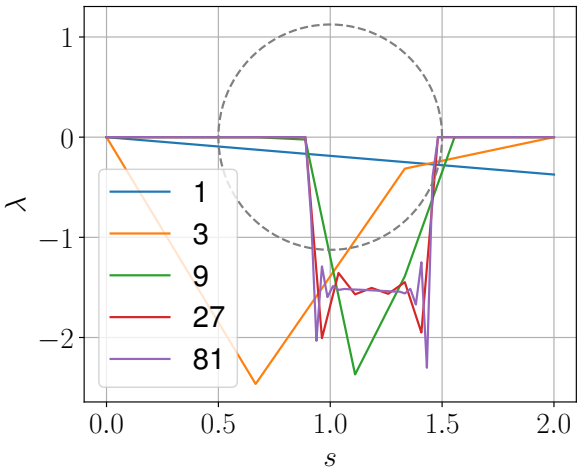


Figure 4.16: Rotating beams: Lagrange-multiplier distribution for $\alpha = \pi/2$ for a different number of elements

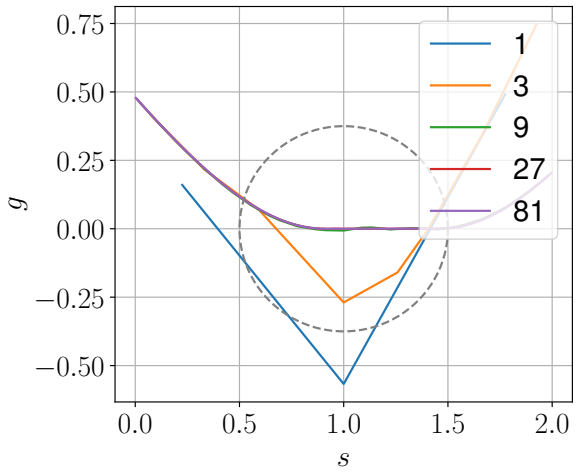


Figure 4.17: Rotating beams: gap distribution for $\alpha = \pi/2$ for a different number of elements

4.2.4 Cantilever test

This test is similar to Example 4.2 from [Bosten et al., 2022] and simulates a beam to rigid body contact. A cantilever beam is placed above another beam with all the degrees of freedom fixed, simulating a rigid body. Both beams have material parameters: $EA = 6.28 \times 10^6$, $GA =$

0.242×10^6, $GJ_t = 0.12$ and $EI = 0.16$. Radius of the circular cross-section is 0.001. The top beam is discretised using 8 quadratic elements while the bottom beam consists of a single linear element. A distributed force $q = 10$ is applied to the top beam in 30 equal load steps. Geometry of the setup is given in Figure 4.18. All contact elements use 4 integration points.

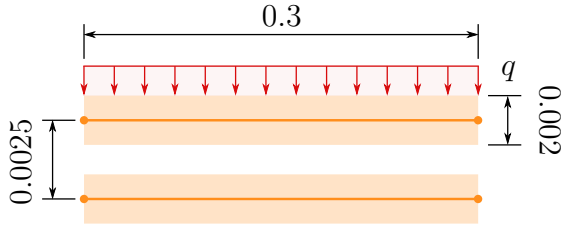


Figure 4.18: Cantilever: setup

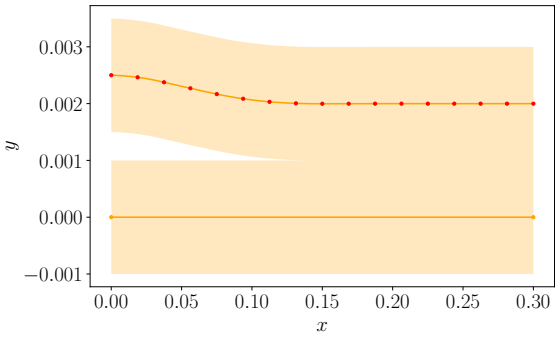
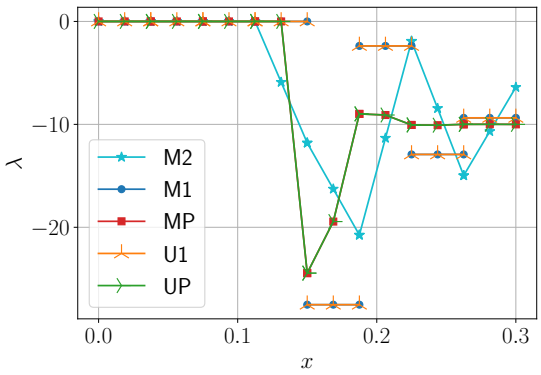


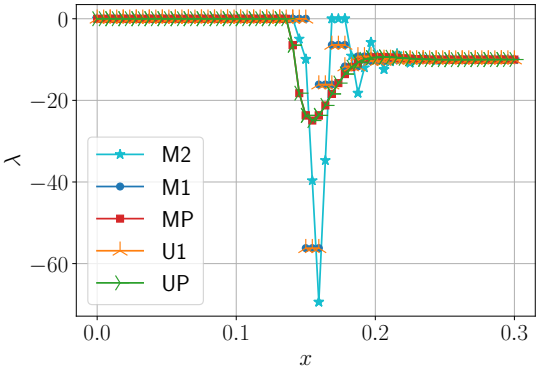
Figure 4.19: Cantilever: deformed state

Figure 4.19 shows the final deformed state obtained by the unbiased line-to-line method with a single Lagrange multiplier per element. The aspect ratio of the figure is such that the deformations are clearly visible. Figures 4.20a and 4.20b show the final contact pressure in the tested formulations. Qualitatively the contact pressure agrees with the benchmark solution presented in [Bosten et al., 2023]. Elements sharing the contact-force model (the same Lagrange-multiplier interpolation order or the same penalty parameter) produce closely related results. This is also observed for the convergence rate in Figure 4.20c, which shows the penalty parameter for which the contact is successfully evaluated. Figure 4.20a shows the final contact force in the tested formulations. Elements sharing the contact-force model (the same Lagrange-multiplier interpolation order or the same penalty parameter) produce closely related results. This is also observed for the convergence rate in Figure 4.20c, which shows the penalty parameter for which the contact is successfully evaluated.

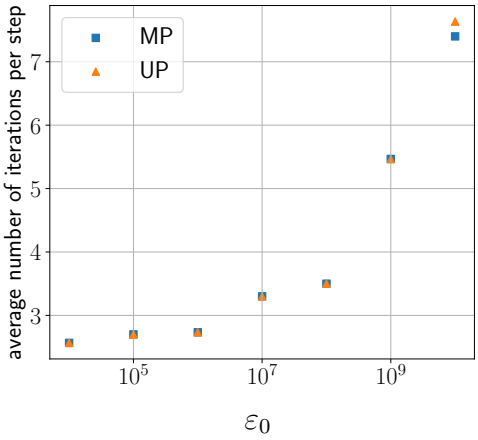
In Figure 4.22 we can see how the results for final contact force distribution from [Bosten et al., 2022] compare to the present study. The beam is discretized using 64 linear or quadratic elements and the full load is applied in 240 steps. The main difference originates from the selection of the beam element. In [Bosten et al., 2022] the authors use $SE(3)$ beam elements. Our first-order elements have a significantly different distribution of the Lagrange-multiplier field while our second-order elements produce comparable results to the reference $SE(3)$ solution. It is also interesting to compare how the second-order interpolation of the Lagrange multipliers affects the oscillations on the boundary of the contact region. In [Bosten et al., 2022], the authors test only the first-order interpolation of the Lagrange multipliers which leads to smaller oscillations



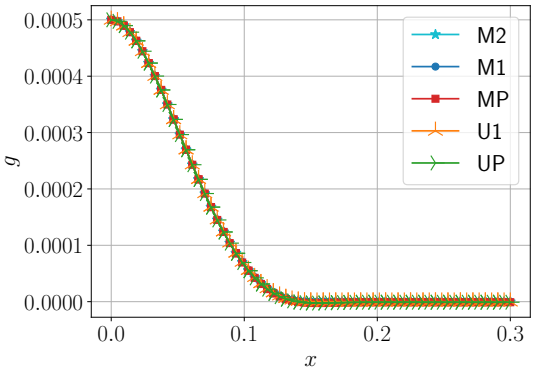
(a) Contact pressure using 8 elements per beam, $\varepsilon_0 = 10^7$



(b) Contact pressure using 32 elements per beam, $\varepsilon_0 = 10^7$



(c) Convergence with respect to the penalty parameter



(d) Gap function using 32 elements per beam

Figure 4.20: Cantilever: convergence of Lagrange multipliers with refinement of beam mesh

compared to the second-order interpolation. Also an interesting artefact of the second-order interpolation is a partially negative Lagrange-multiplier field, which is permitted by the constraint enforcement since only the nodal values are checked. The differences are not attributed to the difference in the solution technique (augmented-Lagrangian vs Lagrange-multipliers method) but to the difference in elements and interpolation of the Lagrange-multiplier field.

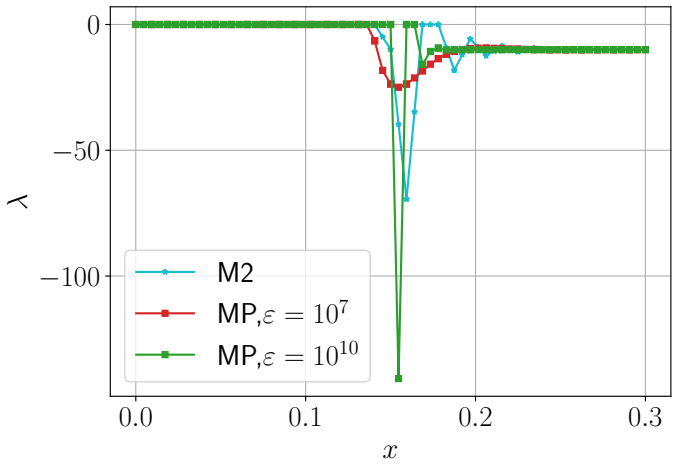


Figure 4.21: Cantilever: Contact pressure using 32 elements per beam; comparison of M2 and MP using two different penalty parameters.

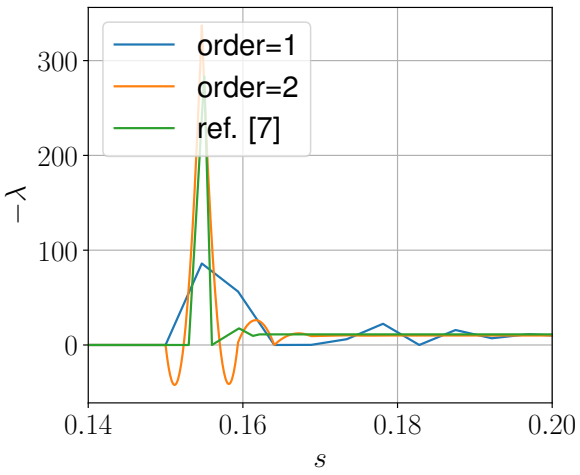


Figure 4.22: Cantilever: final Lagrange-multiplier field for different elements

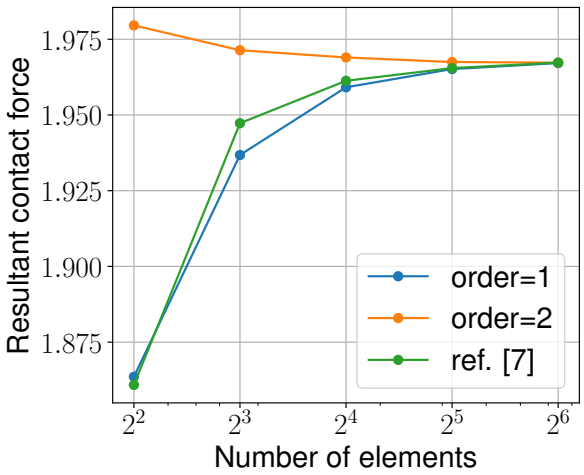


Figure 4.23: Cantilever: final contact force convergence for different elements

It is interesting to see in Figure 4.20a, that even a single-node interpolation results in some oscillations. The oscillations in the contact pressure strongly resemble the spectral leakage phenomenon well known in the signal processing community which is the result of a mismatch

between the actual frequency of the signal and the discrete frequencies of the Fourier transform [Carlson and Crilly, 2009]. The same reasoning can be applied here. The discontinuity in the pressure cannot be described exactly so it spreads to the neighbouring nodes within the contact region (the contact constraints do not permit the leakage to be distributed on both sides of the discontinuity). This happens because of the integration over the boundaries of the contact as the integration zone is not precisely computed. The penalty method (MP and UP) for lower penalty parameters on the other hand experiences a smooth transition as the contact conditions are strongly enforced. This is at the expense of a larger penetration at the contact boundary. When the penalty parameter is increased, similar oscillations appear - see Figure 4.21. Again, the discontinuity cannot be modelled exactly while, at the same time, the gap function cannot compensate with large enough penetration due to the increased stiffness, which manifests in the form of the oscillations.

Higher interpolation order means more discontinuities in the system due to the activation of Lagrange-multiplier degrees of freedom, which is reflected in a larger number of contact steps even though the convergence in a particular contact step is similar (observe total number of iterations vs. mean number of iterations per contact step in Table 4.4).

Table 4.4 shows the total number of iterations and normalised computation time for all the formulations, for the same number of number of integration points, tolerances, step-sizes and all other input parameters. For this example, the mortar element with the penalty method is the fastest, being 23 % faster than its Lagrange-multiplier counterpart, while the unbiased method is consistently slower than the mortar method (83 % difference between MP and UP and 67% M1 and U1). Since there are no large deflections in the test, the contact elements are all defined at the beginning and there is no need for contact search algorithm. The test, therefore, only measures the performance of the computation of projections and convergence. Since this test is easy to converge (approximately 2 iterations per load step), the difference in convergence rate is not dominant in the total computation time. Using the penalty method reduces the computational time although it increases the total number of iterations. This indicates an inferior convergence path. The total number of iterations strongly depends on the selected penalty parameter as can be seen in Figure 4.20c.

In Figure 4.23, spatial convergence of both elements compared to the reference solution is presented. Convergence speed is attributed to the interpolation order of the Lagrange-multiplier field. The present linear and $SE(3)$ elements both exhibit slower convergence than does the present quadratic element with the second-order Lagrange-multiplier interpolation.

Figure 4.24 shows in logarithmic scale how the gap is distributed for both orders of elements.

Table 4.4: Cantilever: formulation performance comparison

	M2	M1	MP ($\varepsilon_0 = 10^5$)	U1	UP ($\varepsilon_0 = 10^5$)
total no. of iter.	154	82	81	82	81
mean iter. per contact step	2.23	2.19	2.70	2.10	2.70
comp. time (ref.: M2)	1.00	0.52	0.40	0.87	0.73

The differences between them are small but one can observe that the value of the gap function is consistently closer to zero in the contact region with second-order interpolation.

The energy evolution for the example with 64 linear elements is shown in Figure 4.25. Energy difference $W_{ext} - E_P$ is relatively small for the entire duration of the simulation. In Figure 4.26 we can observe more closely the first 10 steps. The contact happens after step 2 which is visible in the change of $W_{ext} - E_P$ which is now different from zero. This behaviour is expected in non-smooth mechanics. The energy difference changes for the entire duration of the contact as is visible in Figure 4.27. There we can also see, that the second-order interpolation has a smaller difference in $W_{ext} - E_P$.

Median convergence in two iterations is observed as in [Bosten et al., 2022]. The average number of Newton-Raphson iterations is 1.3, but there are also 1.6 secondary contact loop iterations which brings to a total of 2.1 iterations per time step.

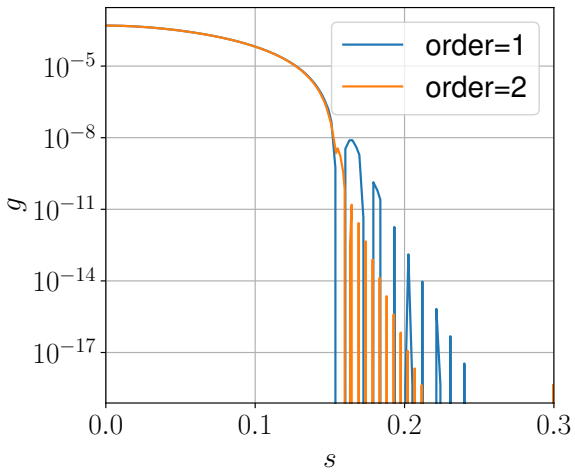


Figure 4.24: Cantilever: final gap for 64 lin. and quad. elements

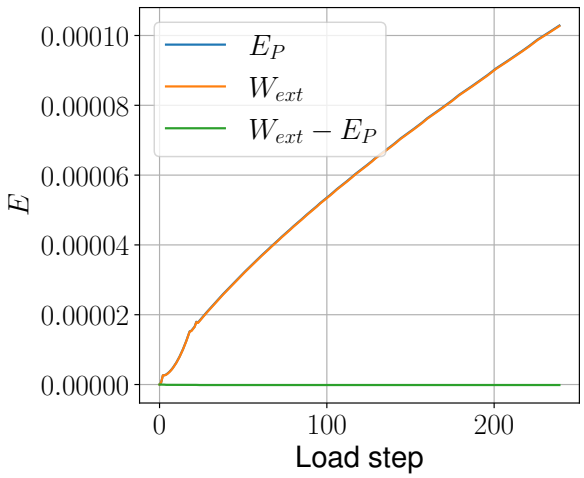


Figure 4.25: Cantilever: energies for entire simulation

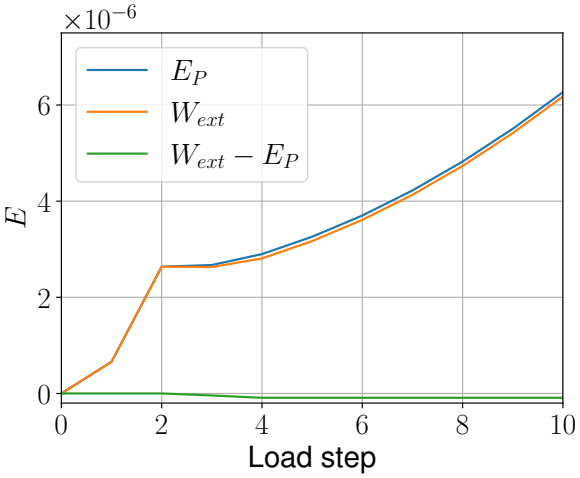


Figure 4.26: Cantilever: energies for only first ten steps

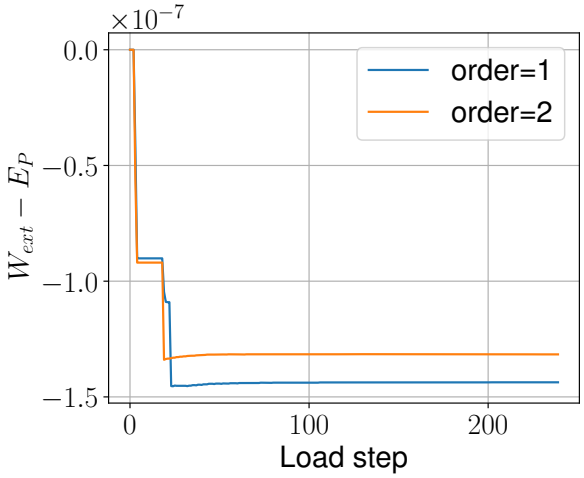


Figure 4.27: Cantilever: total energy in comparison for different interpolation

4.2.5 A Coupled bending

To analyse the contact algorithm stability under finite deformations, the following test utilises two deformable cantilever beams which are vertically aligned, see Figure 4.28. The beams are pressed together by a constant pressure $q = 1000$ defined in the material frame. In the second load step, a bending moment $M = 1$ is applied to the free end of the bottom beam so that both beams are bent upwards. Beams have the same material parameters as in Example 4.2.4, while the penalty parameter is set to 10^7 where applicable. Different discretisations of both beams and contact are tested, with n denoting the number of beam elements on each beam.

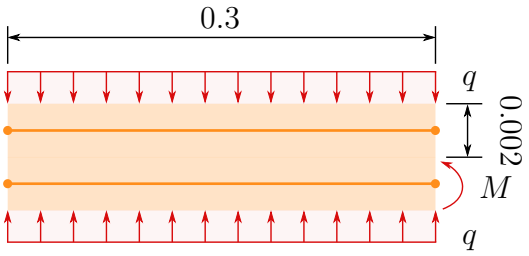


Figure 4.28: Coupled bending: setup

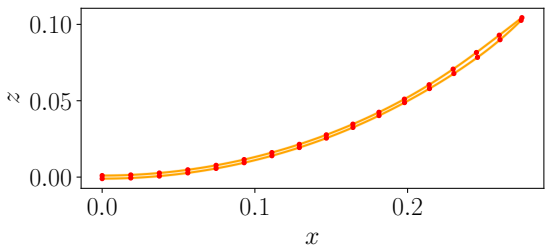


Figure 4.29: Coupled bending: deformed state computed with 16 elements per beam

Since first order beam elements result in a curve made of straight segments, the contact pressure is dominated by the geometrical constraints imposed by the interpolation. As this is a numerically difficult test case due to high compression loads, only second order beam elements

are tested.

Due to high bending deformation and axial stiffness, the top beam in the deformed configuration extends beyond the free end of the bottom beam. The high bending stiffness also induces some separation just before the tip. This effect is captured by all contact discretisations tested here and results in a pressure spike on the tip, Figure 4.30. A drop in the contact pressure is visible in the area just before that as the top beam deforms away from the bottom one. The separation causes oscillations in the contact pressure where the numerical procedure integrates over a discrete jump.

Using the U1 method, one can observe similar results as with M1, but there are also some short contact elements which do not contribute to the contact, even though they are well within the contact zone. This can be contributed to overconstraining as the larger neighboring elements do the heavy lifting. This short elements are a product of the segment-based integration and can cause potential instabilities as is shown in example 4.2.5 B.

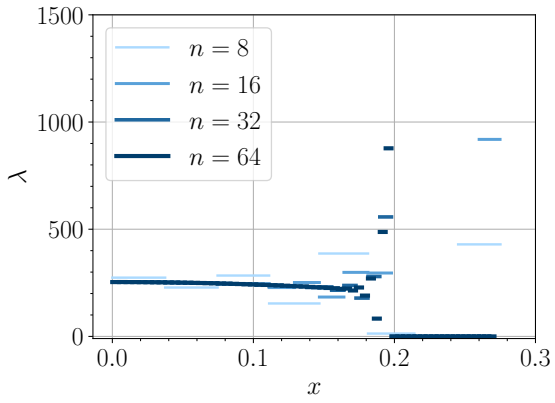
This oscillation between two constant values is called the checkerboard mode [Cheng and Ye, 2002] and can be potentially triggered by the piecewise constant interpolation in any mortar formulation, which is in our case methods U1 and M1. To stabilise it, one can penalise the pressure difference in adjacent contact elements [Puso et al., 2012].

4.2.5 B Coupled bending with sliding

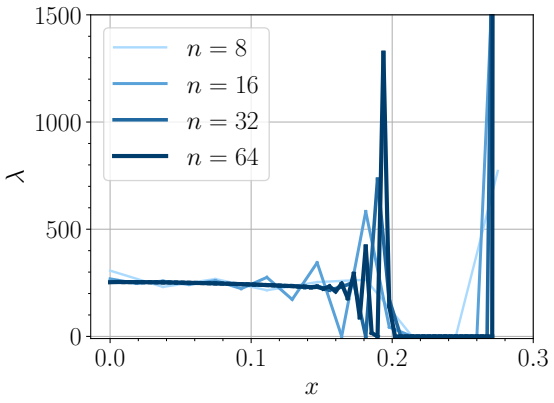
This is a continuation of example 4.2.5 A and tests the contact method during sliding with high deformation. After the bending, the bottom beam is displaced at its support in the direction of the x axis for a displacement u. The displacement is applied in increments of 0.0002. As the two beams separate their combined bending stiffness decreases and the deformation of both beams increases even further. The final deformed shape at $u = 0.04$ is shown in Figure 4.31.

The evolution of contact during sliding is shown in Figure 4.33. From the plots we can see, that the contact zone continuously evolves. The contact zone is shortening while also a new pressure jump appears at the beginning of the contact zone. This results in high oscillations for the mortar method, Figure 4.32, especially with higher number of elements. Nevertheless, the mortar method is stable and can solve the sliding problem without difficulties.

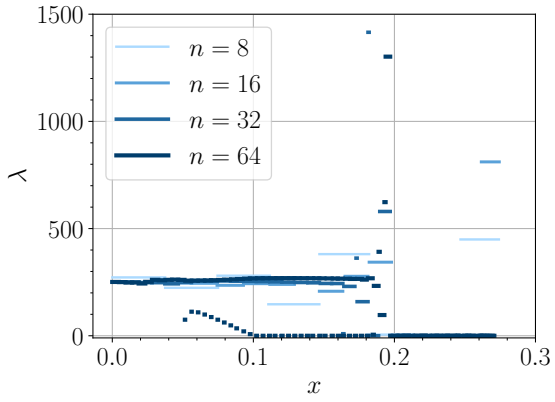
In contrast, the algorithm U1 experiences convergence problems during sliding. Since U1 uses segment-based integration as discussed in Section 3.5.5: Discretisation, this approach produces more degrees of freedom than mortar method which results in poorer conditioning of the numerical system and introduces more nonlinearities during sliding as the elements are created and destroyed. Interestingly, the penalty variant does not share the difficulties in this regard as



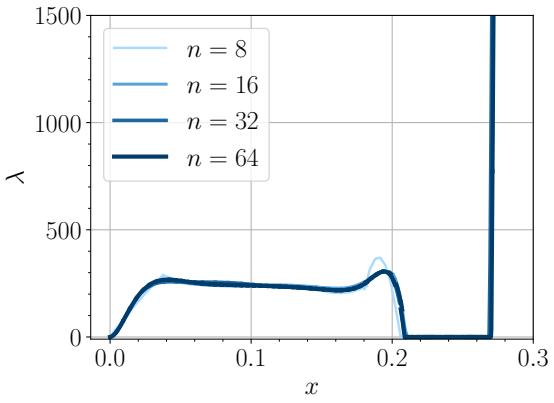
(a) Constant interpolation of Lagrange multipliers M1



(b) Linear interpolation of Lagrange multipliers M2



(c) Constant interpolation of Lagrange multipliers U1



(d) Penalty method UP

Figure 4.30: Coupled bending: convergence of Lagrange multipliers with refinement of beam mesh

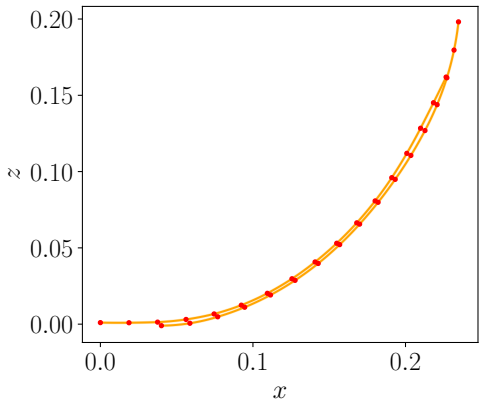


Figure 4.31: Coupled bending with sliding: deformed state computed with 16 elements per beam

it is not burdened with determining the active set of Lagrange multipliers, etc.

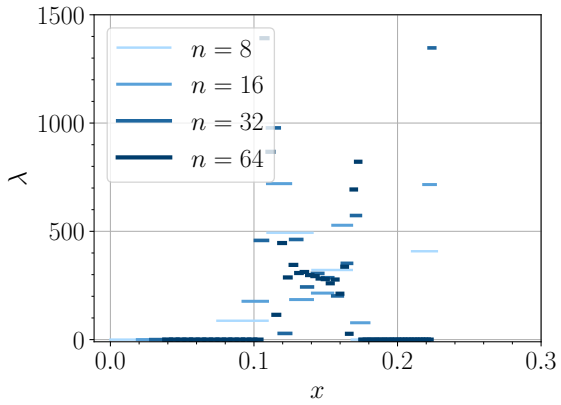
The penalty method again shows that the discontinuities present no problem and one obtains a continuous contact pressure. This is true for both MP and UP algorithms. As shown in the previous example in Figure 4.20c, the convergence highly depends on the penalty parameter which prevents higher values which would result in lower penetration.

Figure 4.34 shows the evolution of the gap function at different points of the simulation for the penalty formulation MP. Comparing it to the results obtained with M2 in Figure 4.33, the maximum penetration is similar, however, the penetration is present throughout the contact zone as is expected for the penalty formulation, as the contact force requires penetration to exist. The average penetration is closer to zero with the M2 method as it allows some positive gap in the middle.

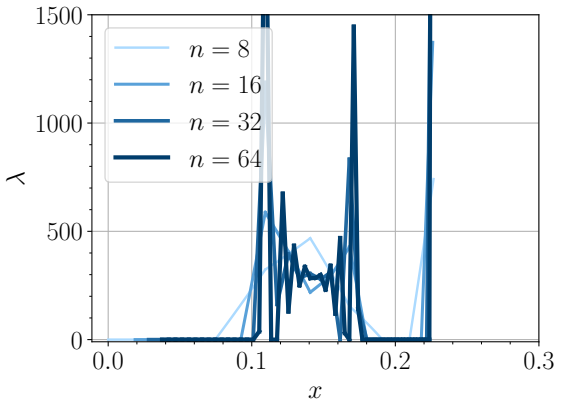
4.2.6 Twisting beams

Two parallel cantilever beams are placed one above the other and separated by an initial gap of 0.0005. The beams are of length 1 and material parameters $EA = 0.628$, $GA = 0.242$, $GJ_t = 0.12$ and $EI = 0.16$. The cross-section is circular with radius 0.001. The beams are fixed at one end, while the other is displaced in a circular fashion to complete a full revolution in 40 load steps. The nodes on the displaced end of the beams are allowed to freely rotate. Each beam is discretised using 16 quadratic elements. The iterative process is stopped when the energy norm 10^{-8} is achieved.

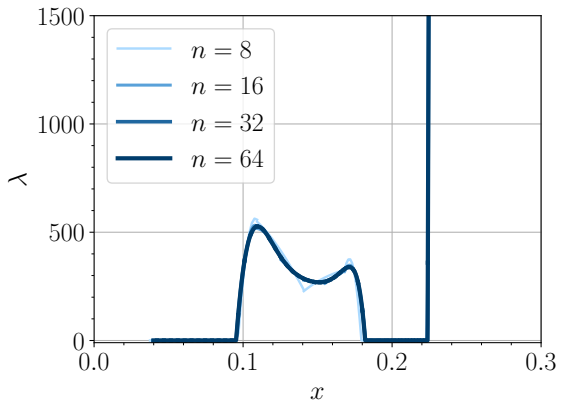
Figure 4.35 shows the deformed configuration of the beams. The results show some asymmetry due to different boundary conditions on each end as can be seen both in the gap and contact-force distribution in Figure 4.36a and 4.36b.



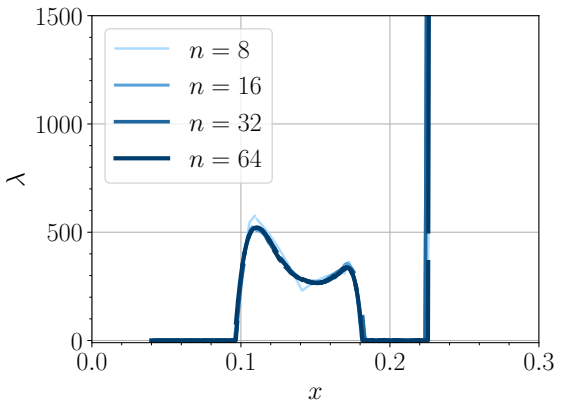
(a) Constant interpolation of Lagrange multipliers M1



(b) Linear interpolation of Lagrange multipliers M2



(c) Penalty method MP



(d) Penalty method UP

Figure 4.32: Coupled bending with sliding: Convergence of Lagrange multipliers with refinement of beam mesh

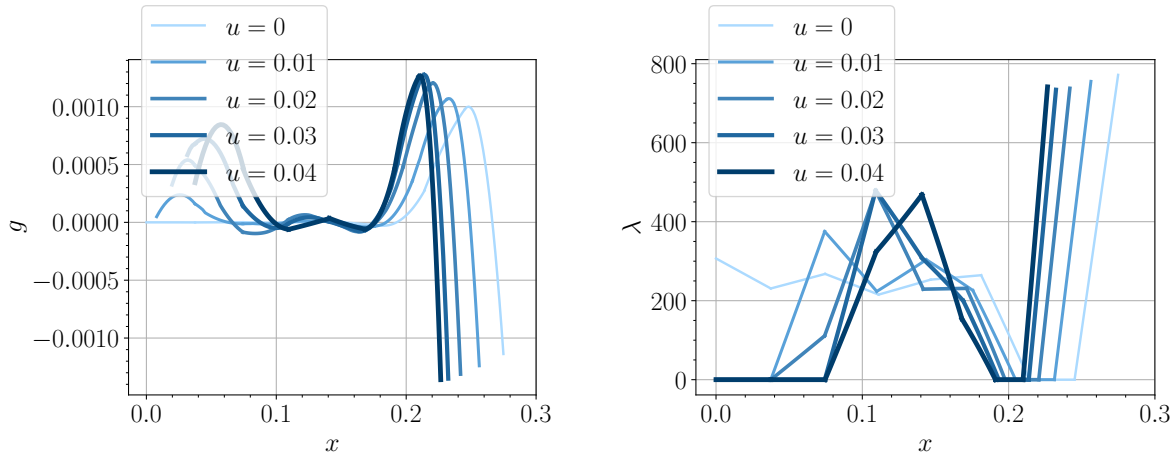


Figure 4.33: Coupled bending with sliding: evolution of the contact with sliding computed with M2 and 8 beam elements

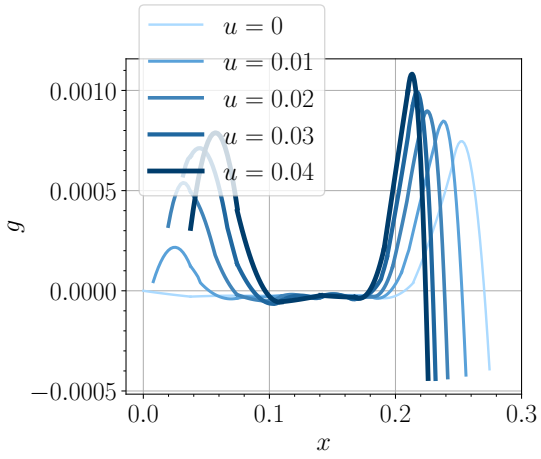


Figure 4.34: Coupled bending with sliding: evolution of the contact with sliding computed with MP and 8 beam elements

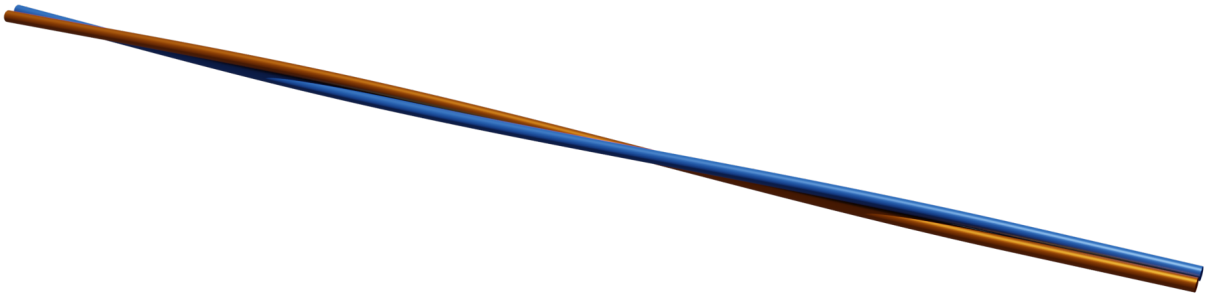
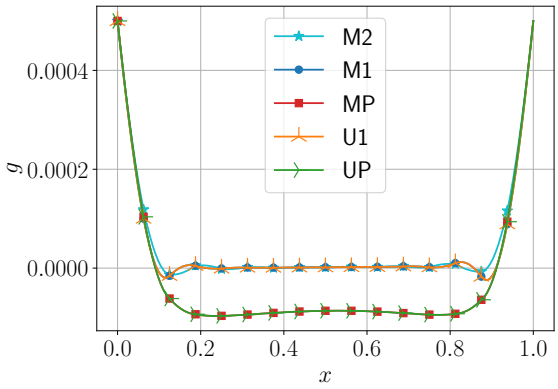


Figure 4.35: Twisting beams: final deformed configuration after twisting (cross-section is scaled by a factor of 5 for visualisation purposes)

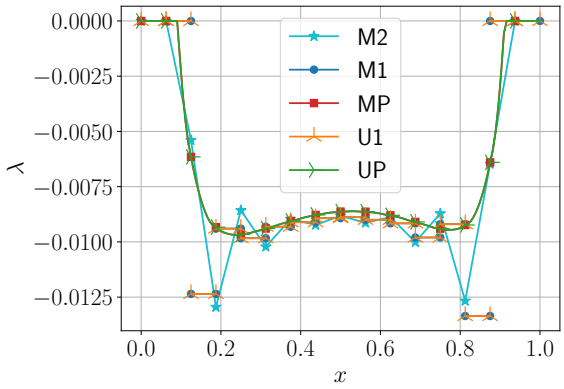
Comparison of the mortar and the unbiased method with the same contact-force model shows complete matching of the results in all respects: gap distribution, contact-force distribution and the number of iterations (see figures 4.36a, 4.36b and table 4.5). The unbiased method is slower because each element is computationally more expensive while also more elements need to be evaluated. The penalty method is computationally more efficient.

As in Section 4.2.4, we can observe some oscillations in the contact pressure at the boundaries of the contact. The penalty method numerically introduces some finite stiffness of the cross-section resulting in smooth gap and pressure fields. This example confirms the results of the example 4.2.4. The total number of iterations is the same for the mortar and the unbiased method, while the computational efficiency remains in favour of the mortar method. The computational cost of the penalty method varies with the penalty parameter. For a low value, e. g. $\varepsilon_0 = 100$, the method is the fastest among the tested, while admittedly the results are physically questionable as the penetration is around 10 % of the cross-section radius. For a larger value of the penalty parameter, e. g. $\varepsilon_0 = 400$, the computational cost increases as does the processing time, as visible in Table 4.5. The high number of iterations per step indicate that the load increment should be reduced for optimal performance. Dependency of the average number of iterations per step on the penalty parameter is plotted in Figure 4.37. Interestingly, with 40 steps, the increase is not monotonous which we contribute to undetermined local anomalies. Since this is a large time-step increment for this example, its decrease is expected to smoothen the trend. This is supported by an additional data set at 48 time steps.

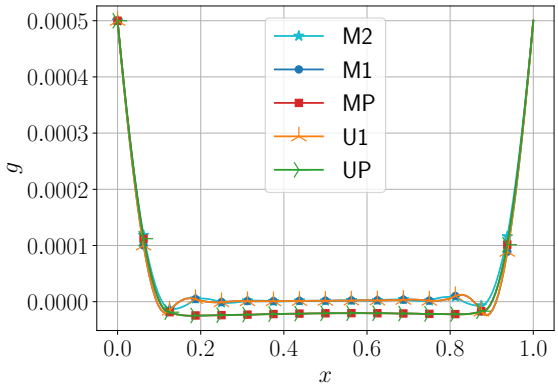
Following a very similar test case from [Tomec and Jelenić, 2022] we can also analyse the spatial convergence of the mortar method. The convergence of the final contact force distribution is shown in 4.38. An alternating solution can be identified with the overall amplitude becoming smaller as the number of elements increases. The contact force is converging towards a constant value of around -50 with the clear boundary effects explained later. The integration error has



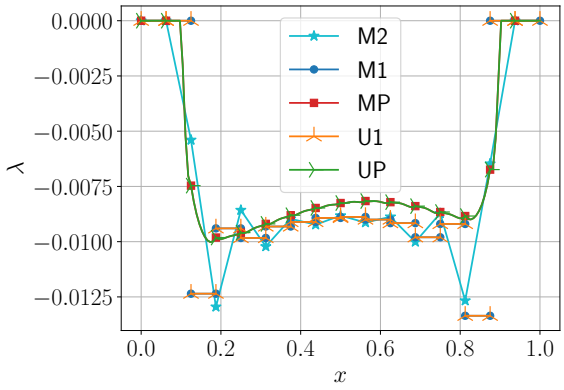
(a) Gap function, $\varepsilon_0 = 100$



(b) contact pressure, $\varepsilon_0 = 100$



(c) Gap function, $\varepsilon_0 = 400$



(d) contact pressure, $\varepsilon_0 = 400$

Figure 4.36: Twisting beams: comparison of the final state of the contact

Table 4.5: Twisting beams: formulation performance comparison

	M2	M1	$\text{MP}_{\varepsilon_0=100}$	$\text{MP}_{\varepsilon_0=400}$	U1	$\text{UP}_{\varepsilon_0=100}$	$\text{UP}_{\varepsilon_0=400}$
total iter.	113	108	92	604	108	92	504
mean iter.	2.05	2.0	2.3	15.1	2.0	2.3	12.6
rel. comp. (M2)	1.00	1.16	0.64	4.2	1.39	0.95	5.2

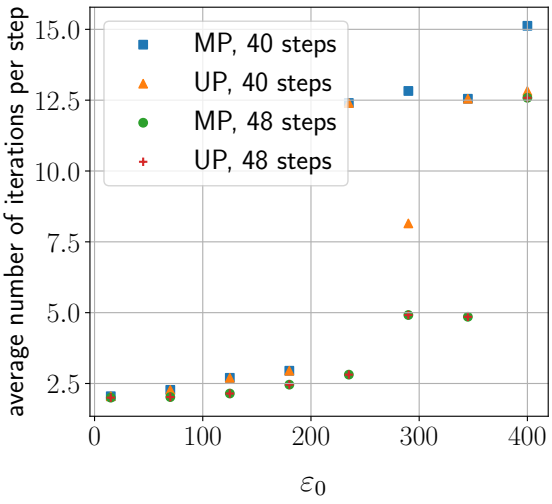


Figure 4.37: Twisting beams: convergence with respect to the penalty parameter

been tested using 8 elements in combination with 2, 3 and 8 contact integration points on each element. While all solutions successfully converge, it is necessary to use at least the number of nodes on an element for the number of contact integration points to eliminate significant integration error (see Figure 4.39). Higher order integration might also be necessary if strong C^1 discontinuities are present on the mortar side. This, however, has not been encountered in this example.

A comparative simulation with the definition of this example from [Bosten et al., 2022] has also been conducted. Four full turns are applied using 64 quadratic elements. Final integrated gap comparison can be seen in Figure 4.40. As expected, the mortar method fulfils the weighted gap constraint (3.142) in both cases. The difference shows in the Lagrange-multiplier field illustrated in Figure 4.41, which has a lot less oscillations for [Bosten et al., 2022]. This is attributed to the $SE(3)$ element, which can interpolate helix-like shape of the centreline much better than the polynomial interpolation used in our elements. Stable oscillations in the force in the most part of the beam show that this is indeed the converged solution which is required to force the beams into the helix-like shape.

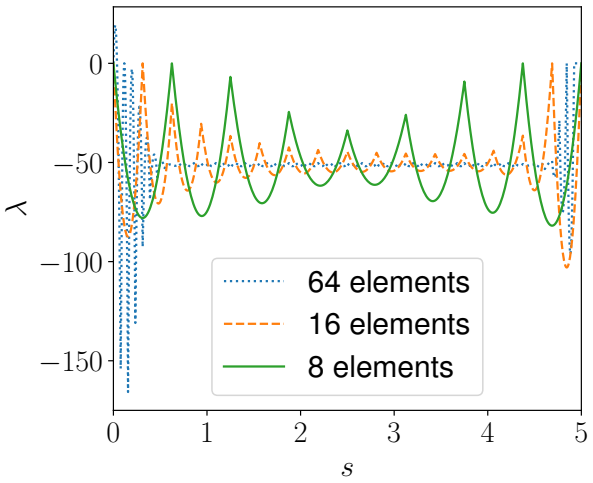


Figure 4.38: Twisting beams: convergence of the final contact force with respect to the number of elements per beam

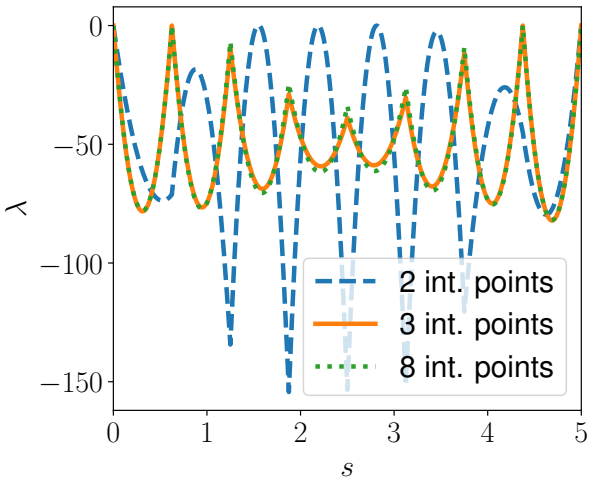


Figure 4.39: Twisting beams: convergence of the final contact force with respect to the number of integration points per element

4.2.7 A Twisting of a ring

A slender ring is being twisted using displacement control. This is achieved by clamping one point and providing rotation increments at the diametrically opposite point. The axis of rotation is defined by these two points. Due to large displacements, the ring experiences self contact and begins twisting around itself. The system is characterised by bifurcations as reported in [Van der Heijden et al., 2003, Goss et al., 2005, Gay Neto et al., 2015]. This is additionally supported by monitoring the eigenvalues of converged tangent stiffness matrices. The test servers to compare the robustness of the contact formulation in the presence of large displacements and small contact regions.

The ring is characterised by the following set of material properties: $EA = 6.52 \times 10^6$N, $GA = 2.18 \times 10^6$N, $GI_t = 30.6$Nm^2 and $EI = 40.7$Nm^2 which corresponds to an austenite nitinol wire [Gay Neto et al., 2015] with a 2mm cross-section diameter. The radius of the ring is 0.2m.

For the first 45 steps the step size is 0.1 radians bringing the system close to self-contact. Afterwards, the step size is reduced to 0.01 radians and remains constant. The test it continued until the loss of convergence. The ring is discretised using 500 second-order elements to achieve the slenderness of the contact $2\rho/(\text{initial element length}) = 0.8$ which is sufficient for reliable contact detection according to Figure 4.14. The evolution of the deformation of the ring is shown in Figure 4.42.

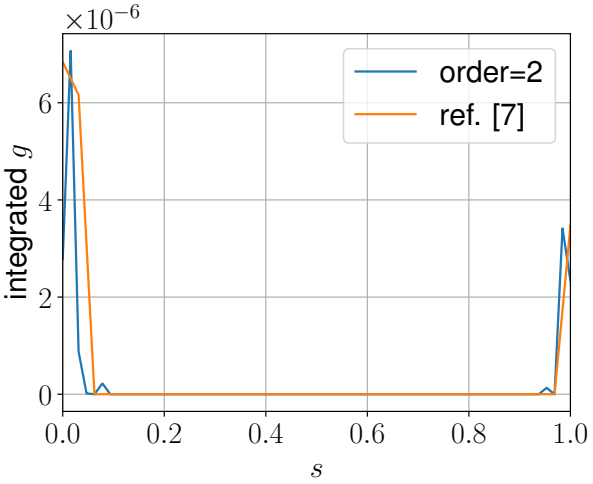


Figure 4.40: Twisting beams: integrated gap using second-order elements compared to $SE(3)$ elements [Bosten et al., 2022]

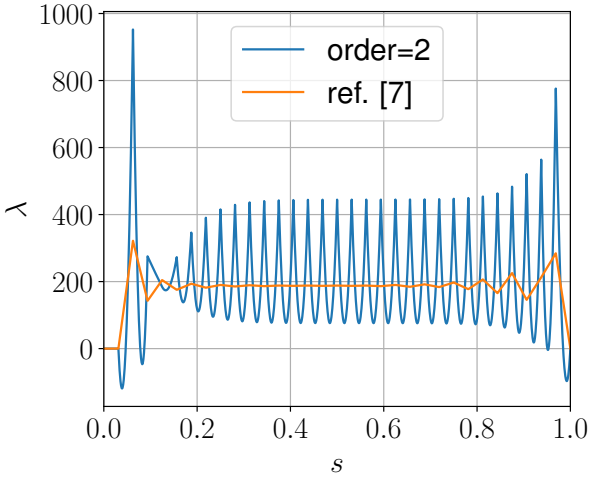


Figure 4.41: Twisting beams: Lagrange-multipliers using second-order elements compared to $SE(3)$ elements [Bosten et al., 2022]

Different formulations have been tested and the performance results are shown in Table 4.6. For the mortar method, a single Lagrange-multiplier per element converges better, achieving the final twist of 10.23 radians, while the linear interpolation of the Lagrange multipliers diverges soon after the first loop. The unbiased element U1 diverges later at 7.64 radians, but still sooner than the mortar element M1. This example is, because of the relatively large displacements between steps compared to the thickness of the cross-section, particularly sensitive to the stability of the contact algorithm. This means that switching nodes quickly results in switching between the bifurcation branches. For this reason, the M1 element is the most suitable as it has less nodes than M2 and simpler integration boundaries than U1. The latter is the main issue with the unbiased method - a simple error in the projection can quickly lead to incorrect determination of the integration boundaries and mark the element as not participating in the contact. The same projection error in the M1 element only affects the particular integration point as the boundaries are fixed.

Interestingly, the penalty method underperforms completely, not being able to properly detect contact and converge at all. We have tried setting a variety of penalty parameters ε_0 including $[10, 10^2, 10^3, 10^4, 10^5, 5 \times 10^5, 7 \times 10^5, 10^6, 10^7]$. If the parameter is smaller than 5×10^5, the beam slips right through itself without detecting contact, while larger parameters do not enable convergence. We have also implemented automated step halving but also could not pass the 5.18 radians of total rotation load, which is right after the contact. There might exist a combination

of parameters (step size and penalty parameter) for which the contact is successfully evaluated but are not easy to determine which is an argument in favour of Lagrange multipliers methods.

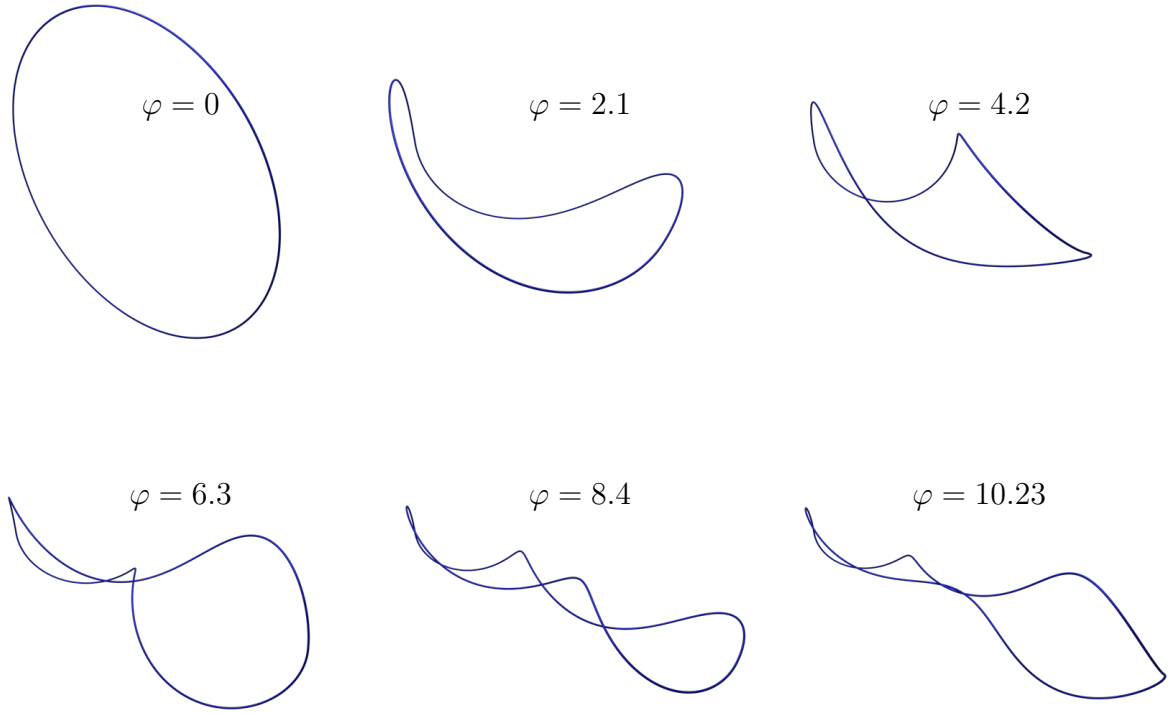


Figure 4.42: Twisting of a ring: deformed configuration using M1 at accumulated prescribed rotation φ

Table 4.6: Twisting of a ring: formulation performance comparison

	M2	M1	U1
total number of iterations	442	2338	2886
mean number of iterations per contact step	3.94	3.62	4.09
total rotation achieved	5.17	10.23	7.64

4.2.7 B Twisting of a ring

Self-contact is a special case of contact because, in general, both sides of the contact are carrying the mortar elements; in other words, both sides of the contact act simultaneously as master and slave side. This could in principle lead to stability issues due to under-determination of the system. To test this, we have considered another case, an example similar to [Chamekh et al., 2009]. A ring has a radius $R = 1$ and a circular cross-section with radius $\rho = 0.04\pi$.

The material parameters are selected in a way to avoid any bifurcation phenomena. The axial stiffness is $EA = 2.76461 \times 10^3$ the shear stiffness is $GA = 1.03924 \times 10^3$, the bending stiffness is $EI = 2764.52$ and the torsional stiffness is $GI_t = 2078.5$. The ring is being twisted by a moment instead of displacement control. The moment is applied in 14 load steps. First 9 load steps increase the moment magnitude by 700, after which the contact is expected and thus in the following steps the moment increase is reduced to 70. The total moment load accumulates to 6650. The final deformed shape with 32 second-order elements is presented in Figure 4.44.

First, the ring has been discretised with 8 second-order beam elements. Two cases have been analysed - test case A, where all 8 beam elements are also non-mortar and test case B, where only the first half of elements (4 elements between the fixing point and the concentrated moment) are non-mortar. No issues have arisen in any of the two cases and the resulting gap function in Figure 4.45 is matching completely in both cases. In test case A, the contact force in Figure 4.46 gets distributed equally between the non-mortar elements on both sides of the contact and is therefore half the size of the one in test case B.

Second, the convergence of the solution is tested by refining the mesh using 8 and 32 quadratic elements. Gap function evaluated at the integration points can be seen in Figure 4.47. It can be observed, that in the case with 8 elements, the curve is not defined in the entire domain, specifically where the gap should be larger. This is due to the divergence in the nearest point projection algorithm. As the contribution from these integration points is small, it does not affect the overall solution. Close to the contact, where the gap is smaller, the results almost match those from the finer mesh. The position of the contact is slightly shifted, probably due to the larger approximation error of the geometry itself. The contact force is forming a spike that is higher and narrower with increasing number of elements (Figure 4.48). The nature of the beam theory with rigid cross-section would suggest the expected result to be exactly a point force in the shape of the Dirac delta function. However, due to the weak character of the non-penetration condition, it is approximated by a distributed force.

Also due to the weak non-penetration condition employed in the mortar method, it has been expected from the start that if the ratio between cross-section radius and element length $(\rho^{(1)} + \rho^{(2)})/L^{(1)}$ is too large, it may pose a problem for contact detection when two beams intersect perpendicularly. This example has been used to test this property and it has been determined that at least 7 elements are required for contact to be detected in which case the ratio is $(\rho^{(1)} + \rho^{(2)})/L^{(1)} = 0.28$. This value agrees with Example 4.2.3, Figure 4.14 for an angle close to $\pi/4$.

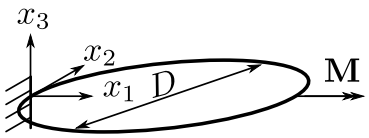


Figure 4.43: Twisting of a ring: setup

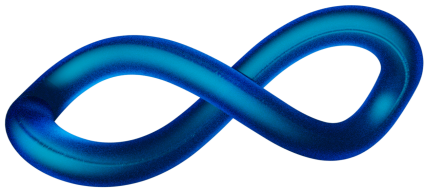


Figure 4.44: Twisting of a ring: Final deformed shape

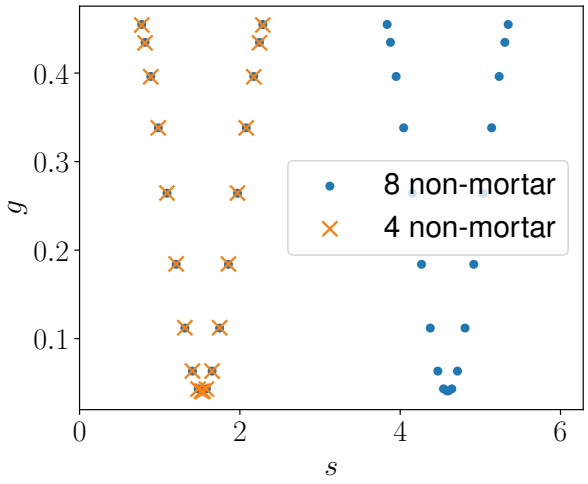


Figure 4.45: Twisting of a ring: Final gap with respect to the number of non-mortar elements

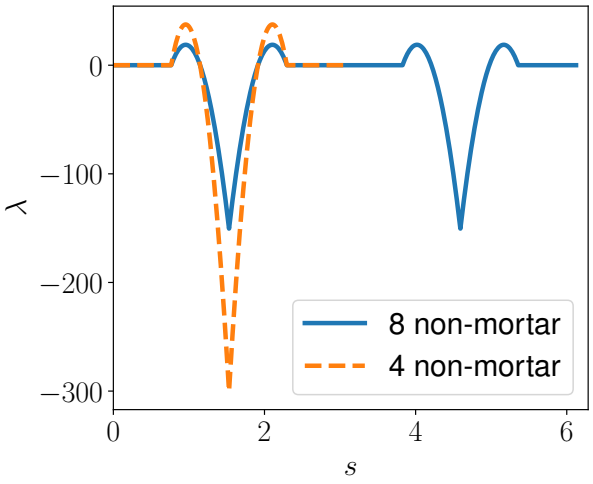


Figure 4.46: Twisting of a ring: Final contact force with respect to the number of non-mortar elements

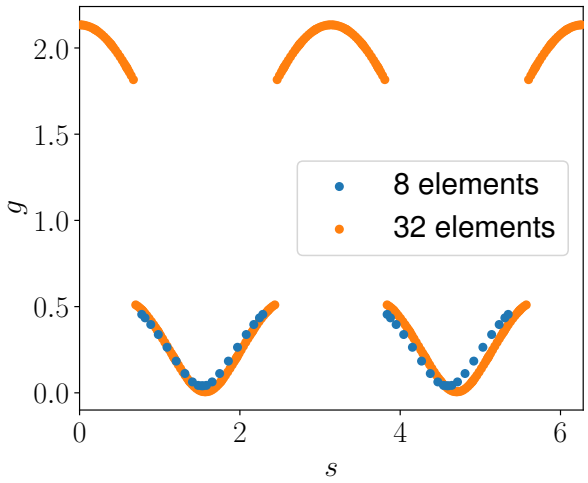


Figure 4.47: Twisting of a ring: Final gap with respect to the number of elements

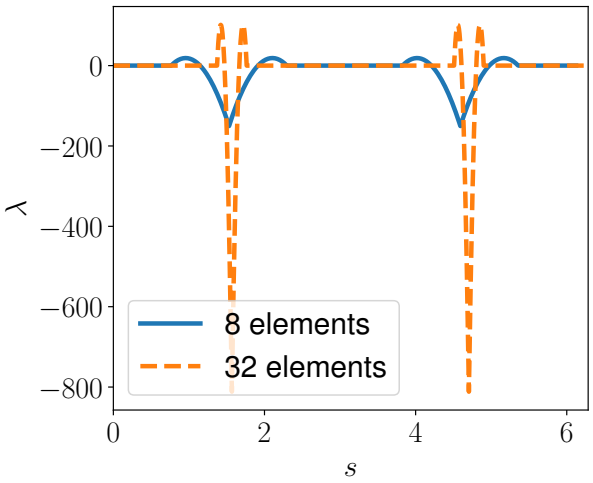


Figure 4.48: Twisting of a ring: Final contact force with respect to the number of elements

4.2.8 Braiding

This test is an extension of the Twisting beams test 4.2.6 and simulates a situation with more complicated contacts. Three fibres of diameter 0.0001 are simply supported on one end and have roller supports on the other to allow axial movement. Fibres are also unable to rotate around their axes by disabling rotational degree of freedom corresponding to the axial rotation in thee simple support. The material properties of the fibres are: $EA = 250$, $GA = 90$, $GI_t = 50$ and $EI = 60$. Fibres are positioned in a vertical plane with 0.0001 gap between each of them, as shown in Figure 4.49. Each fibre is 0.05 in length and is discretised using 25 beam elements.

Before starting the braiding motion, the fibres are tensed with an axial force of 10. Afterwards, the braiding process starts, which follows simple circular movement to switch the positions of the middle fibre end-point and the one of a side fibre, keeping an alternating pattern. The applied step size is $\pi/30$ requiring 60 steps to complete a full rotation of a fibre pair. The final deformed shape can be viewed in Figure 4.50.

The results cannot be obtained with the mortar method using either constant or linear interpolation of the Lagrange-multiplier field - the system does not converge after the onset of contact. The mortar element with the penalty method MP is able to continue and complete about one rotation of a fibre pair. The unbiased formulation in appears to be the better choice for this example with the penalty method element UP reaching 70 load steps and Lagrange-multiplier element U1 reaching almost two rotation as can be seen in Figure 4.50. As this example represents the most general contact of the ones presented in this thesis, this indicates that the mortar method might not always be the optimal formulation with respect to the robustness.

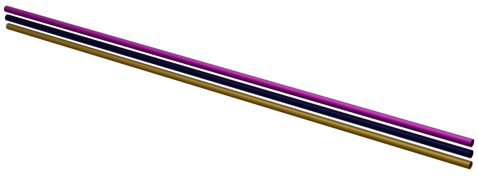


Figure 4.49: Braiding: undeformed

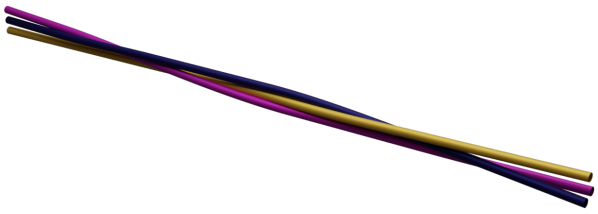


Figure 4.50: Braiding: deformed

Table 4.7: Braiding: formulation performance comparison

	M2	M1	MP ($\varepsilon_0 = 10^6$)	U1	UP ($\varepsilon_0 = 10^6$)
total load steps	17	17	61	111	70
mean iter.	/	/	2.78	2.33	2.20

Chapter 5

Discussion

5.1 Developments in beam dynamics and beam-to-beam contacts

5.1.1 Momentum and near-energy conserving beam element

We explore the feasibility of applying energy-momentum conservative principles to elements employing SE(3) interpolation. This interpolation method seamlessly links nodal values of position and rotation as elements of the SE(3) group, ensuring that all interpolated values remain within the same group. Furthermore, the resulting element is strain invariant and, since it is able to describe pure bending exactly, does not exhibit shear locking. This interpolation is implicitly defined for higher-order elements in (3.82), complemented by an explicit expression tailored for first-order elements in (3.81).

We are specifically interested in this interpolation as it is defined across the entire parametric domain of an element, which makes it suitable for direct implementation into contact models. Line-to-line contact formulations, like the ones in presented here, require a differentiable centreline position for their projections and closest point searches. Indeed, the contact formulation could feature an independent interpolation of the position, but this could lead to numerical artefacts where the two finite element centrelines resulting from different interpolations could be significantly apart and introduce an additional error in the process. It is thus a common practice to use suitable elements with the position field interpolated directly. As it is our long-term intention to develop a long-term stable and objective dynamic beam element capable also of contact interactions it is thus a good foundation to choose an interpolation with this aspect in mind.

Requirements of energy-momentum conservative algorithms

In a continuum, our model is derived from the geometrically exact beam theory using unified expression for the displacement and rotation. This allows an intuitive description of equilibrium equations in the fixed-pole frame, as shown in (3.45). To implement a Lie group time integrator with algorithmic conservation of momentum, we have derived a few necessary requirements

- usage of the Lie midpoint rule with appropriate algorithmic-midpoint quantities,
- sum of interpolation functions at every point in the domain is 1, as stated in (3.49),
- interpolation functions must be independent of the quantities they interpolate as defined in (3.47),
- generalised momentum must be approximated using the first-order Taylor series expansion in time (3.57).

These requirements are essential for momentum conservation and can be fulfilled in different ways, for example, like [Simo et al., 1995,Bottasso and Borri, 1998]. We provide an alternative by using independent trial and interpolation functions to achieve momentum conservation including the objective finite elements.

While momentum is a vectorial quantity and thus its conservation is attached to each degree of freedom, the mechanical energy is a scalar and cannot be considered in the same manner. To conserve energy, the internal and inertial midpoint forces are defined in a way to convey the change in elastic and kinetic energy in an element in a time step, see (3.57) and (3.71). To algorithmically conserve the mechanical energy, a formulation requires the following properties

- increments must be described using trial functions, i.e., they have the same interpolation,
- increments must be expressible through the midpoint-rule approximation, as in (3.64); numerical integration relaxes this constraint to hold only at integration points,
- strain increment must be consistent when computed using the update equation or using the interpolation.

Some of these requirements are in conflict with those for momentum conservation when implementing configuration interpolated elements - specifically the expression of the increment in terms of velocities and Lie group update, as noted in Remark 4. However, we have also shown that our algorithm tends to better conserve energy when a finer finite element mesh is used and is converging towards exact conservation.

Solution for configuration interpolated beam elements

The main novelty we have introduced to achieve the requirements set by energy-momentum conservative algorithm was to separately interpolate velocities. Instead of using interpolation of increments on the configuration, we have used it only for velocities as explained below (3.93). In this way, the configuration remains interpolated and thus differentiable along the element while the velocities become an implicitly defined field, known only in the integration points. Although these velocities no longer correspond to the instantaneous velocities defined by the time derivative of the interpolation of configuration, they still represent the motion of the beam and thus conform to the requirements of the finite element method.

As a bonus, our approach to energy conservation using a separate velocity field simplifies the expressions for inertial forces and makes the element even more attractive for implementation. The direct time derivatives of the implicit interpolation can become quite cumbersome otherwise, as has been noted also in other publications, e.g. [Sonneville and Géradin, 2023].

Finally, although the energy is not exactly conserved, we show that it approaches conservation with mesh refinement, while we also show that it conserves linear and angular momentum, which is important for stability of the time integration. Most importantly, the time integrator exhibits long-term stability shown using the numerical example of the flying spaghetti and retains the second-order convergence rate associated with the midpoint rule.

Such definition of the finite element allows to easily combine the presented beam elements with other finite elements since the degrees of freedom use common definitions and no special treatment is required apart from storing and updating at integration points. Its intended use is, however, strictly in dynamics as the formulation, when reduced to a static integrator, does not reflect the correct internal state any more: as is the case with most Lie midpoint formulations.

5.1.2 Mortar method applied to beam-to-beam contacts

In [Tomec and Jelenić, 2022] we investigate how can the mortar method, a well-established discretisation method for modelling interactions between mechanical bodies, be applied to beam-to-beam contacts and how it behaves in different types of contacts.

Mainly, the mortar method is a discretisation method for partial differential equations which are defined on non-overlapping finite element domains - the prime example being contacts. Although a contact is a geometrical constraint which could be in theory described algebraically, it can also be interpreted using a contact potential, which simply introduces a new potential energy contribution to the Lagrangian, see (3.101). The speciality of the contact potential is that it is conditional: it only applies when contact conditions (3.97) are fulfilled.

The contact potential is integrated strictly over the contact region (3.101). This is the domain where the bodies are physically in contact and contact forces appear repelling the two bodies and preventing penetration. When considering numerical modelling of the contact using the finite element method the challenge encountered is how to evaluate this integral. Namely, we want to be able to parametrise the contact region while considering the discretised nature of the connected domains, also noting that the contact region is not stationary. It is changing depending on the state of the system and can also completely disappear.

The mortar method tackles this problem by first arbitrarily designating the master and the slave side of the contact as depicted in Figure 3.5. These have different roles in the evaluation of the contact. The slave side serves as the fundamental parametric and discrete carrier of the contact integral. It means that in general the integration domain for the contact is equal to the boundary domain of the mechanical body. This, however, does not directly correspond to the contact region, which is why it is adjusted using algebraic constraints on the Lagrange multipliers, which model the contact force in the contact potential. The Lagrange multipliers are an unknown field defined also on the slave side and designed to correspond to the underlying finite element discretisation. The slave side, evidently, is named so as it does most of the heavy lifting. The master side mainly functions as a platform for the definition of the geometry and evaluation of the gap between the bodies.

To apply this method to beams, it must be somewhat adapted. Namely, the region must be parametrised differently as we do not work with surface (this would diminish the main advantage of the beam theory: one-dimensionality). The contact geometry must thus be changed to leverage the existing geometry of a beam. Furthermore, the cross-section is rigid which sometimes gets interpreted as inhibiting the natural tendency in movement of the beam and forcing unphysical behaviour. This is, however, simply debunked by considering the motion of the beam induced by the contact. Because a beam is a longitudinal object, the lateral deformations of the cross-section are not significant, and as such also neglected by the beam theory. The motion rather manifests in bending which presents significantly less rigidity for beam-like structures. The local behaviour of the contact is thus free to follow its natural tendency and should not present significant numerical disadvantages. If, however, the cross-sectional deformation is significant, an extended beam theory with an appropriate set of assumptions should be considered.

Contact geometry

Unlike 3D bodies, where integration over the boundary presents a lower-dimensional problem, beams are one-dimensional, and integrating over the boundary would significantly increase the

overall problem size. It is thus more convenient to exploit the fact that the cross-sections in beam theory are rigid. The integral over the boundary of the beam can be transported to the centreline, reducing the 2D problem to a single dimension. Furthermore, the domain directly corresponds to the length of the beam.

Usually, the cross-section is assumed to be circular, which significantly simplifies the evaluation of the contact. This assumption applies even to non-circular cross-sections because, as promoted already through beam theory, the radial dimension of the beam is less significant. A contact region between two beams is thus a line (or a point), and the Lagrange multipliers in the contact potential represent an integrated and still physically valid value with the units of force per unit length, sometimes called also contact pressure.

To evaluate the contact, we must be able to determine the gap between the bodies, as it determines the state of the contact. The gap is either positive, when the two beams do not touch, or zero, in the case of contact. In numerics, the gap can also be negative, representing penetration, although this is, in the converged state, usually insignificant. The gap is defined parametrically along the slave side to be simply plugged into the contact integral. To evaluate it at a particular point, we compute the shortest distance from the slave centreline to the master centreline and reduce it by the thickness of the beams (3.98). Doing so introduces some numerical error but it tends to correct itself in the equilibrium for most cases; see the discussion below (3.98).

Mortar method: Discretisation

Within the finite-element framework, all integrals are evaluated using the discretised domain, a mesh of finite elements. The mortar method utilises the underlying mesh on the slave side to discretise the field of Lagrange multipliers (3.137). The contact region is then defined using the active set strategy, which enables or disables degrees of freedom for Lagrange multipliers to effectively change the contact region. This method is described in section 3.5.6. Enforcing contact condition. This method does not address the boundaries of the contact region within the slave beam element - the method simply integrates over the discontinuities and thus introduces some numerical deficiencies. Namely, numerical artefacts appear in the form of oscillations in the contact force although these usually do not affect the convergence significantly.

In its simplest form, the Lagrange-multiplier field is interpolated using Lagrange polynomials. This introduces nodal Lagrange multipliers unknowns which is undesired because they represent secondary variables in the system. The algebraic system thus becomes less well-conditioned which can lead to larger numerical errors, slower convergence and, in the most extreme cases, loss of convergence. To remedy this, there exist dual mortar methods which introduce orthogonal

interpolation functions allowing to remove the Lagrange-multipliers from the system. Nevertheless, the additional contributions to the rest of the system can have effectively the same impact. This issue is thus inherent to the Lagrange-multiplier method and, as such, hard to avoid.

Mortar element

The contribution of the contact to the system can be summarised in a mortar element which is a "parasitic" element attached to a beam element on the slave side. The mortar element requires access to the underlying beam element to compute the gap function. Thus each mortar element encapsulates a single slave element and a pool of master elements, which are possible contact partners. The gap function must be computed to each master element in the pool and the smallest one is selected.

Contributions to the equilibrium are developed using standard variational techniques. To evaluate the integrals, numerical integration must be used (because of the non-linear gap function). Here, again, more options exist. We have decided to use the simplest and integrate over discontinuities, for example when the master beams change within the mortar element. The simple Gaussian quadrature rule is used, which turns out to be completely sufficient for the job. The theoretical contemplating of the order of integration seems to point towards a higher number of integration points, but a numerical examination reveals that the integral quickly converges because the non-linearities are generally low for a converged state (see Figure 4.38).

The interpolation of the Lagrange-multiplier field is optimal to be of first order. Higher orders tend to lead to better fulfilment of the contact conditions, but this comes at the expense of higher oscillations in the contact force. Furthermore, the number of non-linearities within the system becomes larger because more nodes need to be activated/deactivated. Higher order is sensible when it can lead to a reduced size of the system, i.e. coarser mesh. This is not possible within the mortar method as the number of mortar elements is determined by the number of beam elements, and thus only presents increased computational burden and less stability.

Finally, consider two equally long beams. The angle between the beams determines the size of the contact region. When parallel, the contact region is the entire domain of the slave beam. But when the angle increases towards 90 degrees, the contact region gradually reduces to a point. This is portrayed in Example: Rotating beams. The mortar method requires enough weight of the contact to an individual node so that the beams do not slip through with contact undetected. This can only be achieved with mesh refinement - longer elements harder detect these small contact regions. The condition is also theoretically derived. The resulting expression can be found in (4.4).

5.1.3 Objectivity of the contact formulation and correlation between the Lagrange multiplier method and the penalty method

While the mortar method provides a robust and stable contact formulation, it suffers from a non-symmetrically defined contact geometry and introduces bias via master-slave distinction. Section 3.5.5 discusses a way to avoid this using an objective contact method and applies the theoretical developments from both, mortar and objective contact, to Lagrange-multiplier and penalty method. This gives rise to four comparable contact formulations.

The idea of objective contact formulation is not new and is already present in the field of contact mechanics in the form of two-pass methods [Solberg et al., 2007], where the contact is evaluated two times. In the first pass, one side is master and the other is slave, and in the second pass, the reverse is used. We wanted to create an objective contact geometry from the start so that the double evaluation can be avoided. There are two main contributions to the contact bias which need to be rectified:

- integration over the contact region,
- projection used in the gap function.

Objective formulation in continuum

We started with an objective parametrisation of the contact region in continuum (3.125) and an objective projection function (3.126). The parametrisation is somewhat complicated as it needs to account for integration boundaries and direction of the parametrisation of individual beams. Our design goal has been to avoid computing the global contact regions so that contact can be applied to any pair of elements and evaluated correctly. The algorithm summarised in Box 1 has allowed us precisely this. As it turns out, using objective projection function (3.126) results in the same weak form (3.130) as for the mortar method, although with slightly different definitions of the fields and integration domain. The equation is also simplified to exclude the change in the integration variable ds as its contribution to the equilibrium is negligible. While the weak form remains the same, its linearisation (3.135) obtains a symmetric structure.

Unbiased discretisation

The main issue of the formulation arises when we introduce discontinuities to the beams through finite element discretisation. To integrate the contact region we utilise the segment-based integration and a corresponding segmentation of the domain into contact elements carrying the contact

force field. As the segmented domain is not necessarily connected, the contact force can only be defined on per element basis, which creates local variables and leaves us only with the option of a single degree of freedom per element and constant interpolation. Otherwise the increased number of unconnected Lagrange multipliers leads to instabilities due to under-constraint, as explained in section 3.5.5 Discretisation.

The resulting discretised formulation is thus not the most stable and further research should look into the ways to amend this. It nevertheless introduces a new way of achieving objective contact formulation and performs well in tests where there is not much relative sliding between beams (example 4.2.1 Patch test).

Lagrange-multiplier vs. penalty method

We also show how to convert a Lagrange-multipliers based contact formulation to the penalty method and all necessary transformations of the contact conditions to achieve this. The resulting formulation no longer enforces contact in the weak sense but strongly at integration points. The advantage of the penalty method is a (generally) reduced stiffness of the tangent stiffness matrix which can lead to better stability of the method. On the other hand, it can result in larger penetrations which can reduce stability when the beams try to cross over. Penalty method is computationally favourable, although its properties greatly depend on the selection of the penalty parameter.

Numerical tests show that the unbiased method indeed provides slightly different results (test 4.2.2) although we could not detect any major differences in convergence paths. The main drawback of the unbiased method is that it cannot be applied in problems with large relative sliding. In other cases it seems to provide comparable results to the mortar method, outperforming it only in the case of braiding (test 4.2.8), where three beams are in contact.

5.2 Meeting the goals

In this section, I provide a comprehensive overview of how each objective outlined in section 1.1 Objectives has been successfully met and addressed throughout the course of this study.

Investigation of the conservation of mechanical properties on Lie groups for beams

The conservation of beam's mechanical quantities is difficult to translate from the continuum to a discretised model, especially when we consider the correct Lie group kinematics of the beam. Through my research, I have showed that both, conservation of the generalised momentum

and conservation of mechanical energy, cannot be achieved when the interpolation of the beam configuration is explicitly given. By design I have decided to conserve momentum exactly and only achieve energy conservation convergence when the size of the finite element goes to zero.

Application of conservation properties to time integrator and dynamic beam element formulation

Building upon the foundational understanding of Lie group mechanics, I have proceeded to apply these properties to develop a novel time integrator and a dynamic beam element formulation. The velocity field of the proposed beam element is separately defined, connected to the configuration only through the incremental update equation. In this way, the velocity field no longer represents the time derivative of the configuration (instantaneous velocity), but rather a generic field, connected to the physical velocity in the weak sense. In this way, I have successfully derived a beam element which, using the Lie midpoint rule, conserves linear and angular momentum and quickly approaches energy conservation when its length is reducing.

Assessment of formulation's practical characteristics

I have shown the stability of this approach using a numerically challenging example to model: the flying spaghetti. This example demonstrates that even under extremely large rotations, vibrations and translations, the solution remains stable and converges with second order rate both in time and in space. Because of the simplified velocity field, the formulation remains numerically efficient in spite of the computationally demanding Lie group interpolation.

Combination of conserving time integrator with objective beam interpolation

I have been able to combine the conservative properties with the objective interpolation, which yields a strain-invariant, locking-free beam element. I have provided the arbitrary-order variant with an implicit definition of the interpolation function and a first-order explicit interpolation.

Understanding mechanics of constraints on beams

In addition to advancing the computational framework for beam analysis, my research also focused on understanding the mechanics of constraints on beams, including unilateral and bilateral contacts. Through theoretical analysis and numerical experimentation, I have gained valuable insights into the behaviour of constrained beam systems, which are essential for accurately modelling real-world structural interactions.

Development of robust contact handling algorithm

A key milestone of this project has been the development of a robust algorithm for handling contacts with unpredictable geometry. Leveraging insights from contact mechanics and computational geometry, I have devised an innovative approach that effectively manages con-

tact interactions in a wide range of scenarios, ensuring numerical stability and computational efficiency. I have developed several algorithms and compared them on a variety of numerical examples based on their numerical efficiency, robustness and stability.

Integration of contact element within finite element framework

Finally, I have successfully integrated the developed contact algorithms within the finite element framework, enabling seamless compatibility with existing beam element formulations. This integration ensures that my approach can be readily adopted by engineering practitioners using standard finite element analysis tools, facilitating the implementation of advanced contact modelling capabilities in structural simulations. The proposed algorithms have so far only been tested on static examples. Incorporating dynamics emerges as the next topic of research.

In summary, I have achieved all objectives outlined in the introduction, culminating in a comprehensive framework for dynamic beam analysis that leverages the principles of Lie group mechanics, objective beam interpolation, and robust contact handling algorithms. My research contributions not only advance the state-of-the-art in computational structural analysis but also hold significant promise for practical engineering applications in various fields.

5.3 Contextualisation

My research falls into the category of academic research for practical applications. This generally applies to novel theories, which are promising but not yet commercially viable. Within this framework, I have focused on developing state-of-the-art methods, which have so far only been tested on simple examples and are yet to be applied to large-scale computer models. In this regard, I have been able to simplify many of the algorithms and focus on the theoretical part.

Beam theory stands as a prime example of model reduction, a fundamental technique in numerical modelling. By utilising the right set of assumptions, we can drastically decrease the complexity of the model from the computational point of view, enabling faster and larger simulations. As the finite element method has positioned itself as the de facto analysis tool for solid mechanics, significant efforts are invested by the community to enhance its speed and robustness further.

New elements are continually being developed to address known issues with the existing elements. With advancements in computer technology, doors open for more advanced methods that trade some efficiency for potentially higher stability and robustness. These new methods aim to be used in more general cases where predicting exact outputs is challenging or where simulations are automated and expected to work across a wide range of scenarios. Following

these principles, advanced mathematical theories are applied to the existing framework or even change the fundamental understanding of the theory.

An exemplary instance is the application of Lie group theory to beam mechanics. Initially introduced for rotations within geometrically-exact beam theory by Simo in [Simo, 1985], this concept, now nearly 40 years old, has undergone extensive research and development. Because a beam can be treated as a collection of cross-sections with their positions and orientations, resembling a collection of infinitely thin rigid bodies, the robotics community has connected this concept with the mathematics of the SE(3) group, a group of rigid body motions (translation and rotation). This connection has led to innovations in interpolation within finite elements [Bottasso and Borri, 1998], and even more revolutionary thinking [Sonneville et al., 2014], where the entire beam theory has been developed based on the group SE(3). This approach demonstrates that all mechanical and kinematic quantities can be developed while treating the beam as a member of SE(3), showcasing an application with an objective finite element.

In subsequent research [Sonneville et al., 2017], higher-order elements were introduced, building upon the foundational work from [Sonneville et al., 2014]. Leveraging these advancements, I have devised a near energy- and momentum-conservative element while also simplifying its complexity. This element appears applicable directly to larger-scale models, and through preliminary testing on small cases, I anticipate it to exhibit strong numerical efficiency and robustness. My approach, guided by predetermined constraints on the element's properties and rooted in mechanical theory, minimizes arbitrary choices during the research process, reflecting the inherent nature of beam dynamics problems and suggesting favourable numerical behaviour.

When simulating real world examples, simulations should not only be able to encapsulate the numerical behaviour of a single body, but also interactions with its environment or between multiple bodies, sometimes thousands of them. An example of such complicated system is human hair, where each individual hair can be treated as a beam, constantly in contact with dozens of other hair. Some of these interactions can be perpendicular, for example crossing of hair strands, while others are longitudinal, like twisting or simply pressing two strands together.

Treating such examples requires robust techniques which are not really present yet in the mainstream industry. The development of new methods is thus constantly present, usually with a specific application in mind, for which this new method performs well. I have wanted to develop a more general tool where contact would be treated continuously and thus capable of handling both short and long contact regions. This has inspired me to translate the mortar method from 3D solid mechanics to beam mechanics.

This method seems to provide a reasonable contact model as long as the contact elements are

relatively short. Otherwise the beams can slip through each other, a problem not encountered in the rest of the contact mechanics. I analyse this problem and discuss the limits to which the mortar element still detects contact.

However, I have noted that the stability of the method is still not guaranteed in every case, e.g. test 4.2.7 A: Twisting of a ring, where the ring cannot be twisted more than one turn and even that only with specific material parameters. The main issue is that even refining the finite element mesh does not improve the results significantly. This has led me to believe that some fundamental changes are perhaps needed in the geometry of the contact.

As objective interpolation improves the stability of the beam element, I have hoped to achieve similar results with the objective definition of the contact. In the end I note that while the resulting formulation is objective, the necessary sacrifices in terms of discretisation might not justify the hardly observable improvements in the robustness. In this sense, further investigations into the possible discretisation techniques for objective contact formulations are still required. One possible approach is using the intermediate geometry, constructed as the average of the two beams in the region of the contact as proposed in [Durville, 2005].

5.4 Reflection and Outlook

In our recent work on dynamic beam finite elements, we have made notable strides in theory, particularly with the development of a dynamic beam finite element that prioritizes momentum and near-energy conservation, formulated using Lie groups. Our initial testing on a small-scale numerical example has shown promise, prompting us to consider practical applications in real-world scenarios for future research.

Looking forward, an important next step is to create the static counterpart of our dynamic beam element to cater to quasi-static simulations. This expansion will enhance the versatility of our theoretical contributions and make them more applicable in various structural analysis scenarios.

Our exploration of contact methods for beams has resulted in the mortar method, a robust approach for contact analysis. Despite its effectiveness, there is room for improvement. Future research should focus on refining the mortar method by adding features like friction, improving discretisation techniques, and integrating the lessons learned from the unbiased formulation.

Speaking of the unbiased formulation, while it shows promise in terms of contact geometry, it faces challenges due to underdeveloped discretisation. To unlock its full potential, we need to invest in refining the discretisation technique. If needed, we may have to explore alternative

approaches, even if they sacrifice some objectivity for practicality.

A notable gap in our current research is the lack of exploration of dynamic cases with contact. Our developed theories should not be limited to static scenarios alone; extending them to dynamic cases with contact presents an opportunity for future research. This would broaden the applicability of our work and provide insights into the dynamic aspects of structural mechanics.

As we outline these future research directions, our goal is to pragmatically build on our recent advancements and address practical challenges in structural mechanics. By focusing on these specific areas, we aim to contribute incrementally to the field, refining our theories and methodologies for more effective real-world applications.

Chapter 6

Conclusion

In concluding this PhD thesis, the research undertaken has been rooted in both a theoretical examination of beam dynamics and interactions between them, and a practical motivation reflecting the needs of industry. Within this work, we have provided a detailed overview of finite element approaches within geometrically-exact beam theory and specifically addressed the challenges of beam-to-beam contacts. The goals outlined in the introduction have been achieved, culminating in the publication of three papers that contribute significantly to the field: one on an energy-momentum conservative beam element, another on the mortar method for beam-to-beam contacts, and a third on an unbiased contact formulation for line-to-line contacts.

The research conducted and the numerous numerical tests performed have led to the development of formulations that represent the current state-of-the-art. The energy-momentum conservative beam element has shown particular promise, with no significant drawbacks identified so far. The mortar method has been validated as a robust and stable approach for certain beam-to-beam contact scenarios, while the unbiased formulation for line-to-line contacts has been recognized as innovative but in need of further refinement to achieve similar robustness.

This work not only meets the initial objectives but also provides a solid basis for future research in dynamic contact formulation. The findings and methodologies established in this thesis offer a starting point for further exploration and development in the field. By bridging theoretical concepts with practical applications, this research contributes to the ongoing evolution of engineering solutions to complex problems. As such, it lays the groundwork for future advancements in the study and application of beam interactions.

Appendix A

Solving implicit interpolation

Evaluation of implicit interpolation (3.81) at a certain point s requires the use of Newton-Raphson algorithm. The size of the resulting system of equations corresponds to the number of degrees of freedom - that is 6. The residual can be defined as

$$\boldsymbol{R}_H = \sum_{i=1}^{N} \psi_i \boldsymbol{\phi}_i = \mathbf{0}, \quad \exp(\hat{\boldsymbol{\phi}}_i)\widehat{\mathbf{T}(\tilde{\boldsymbol{\phi}}_i)\boldsymbol{\phi}'_i} = \exp(\hat{\boldsymbol{\phi}}_i)\hat{\boldsymbol{e}}. \tag{A.1}$$

Linearisation of the above expression goes as follows

$$\Delta \sum_{i=1}^{N} \psi_i(s)\boldsymbol{\phi}_i = \sum_{i=1}^{N} \psi_i(s)\Delta\boldsymbol{\phi}_i. \tag{A.2}$$

Let $\mathbf{H}(s) = \exp(\hat{\boldsymbol{u}})$. Then

$$\begin{aligned} \Delta \exp\left(\hat{\boldsymbol{\phi}}_i\right) &= \underline{\mathbf{H}}_i^{-1} \Delta \mathbf{H}(s), \\ \exp\left(\hat{\boldsymbol{\phi}}_i\right) \widehat{T\left(\tilde{\boldsymbol{\phi}}_i\right)\Delta\boldsymbol{\phi}_i} &= \underline{\mathbf{H}}_i^{-1} \exp(\hat{\boldsymbol{u}}) \widehat{T(\tilde{\boldsymbol{u}})\,\Delta\boldsymbol{u}}, \\ T\left(\tilde{\boldsymbol{\phi}}_i\right)\Delta\boldsymbol{\phi}_i &= T(\tilde{\boldsymbol{u}})\,\Delta\boldsymbol{u}. \end{aligned} \tag{A.3}$$

This leads to the following iteration

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k - \left[\sum_{i=1}^{N} \psi_i(s) T\left(\tilde{\boldsymbol{\phi}}_i^k\right)^{-1} T\left(\tilde{\boldsymbol{u}}^k\right)\right]^{-1} \left(\sum_{i=1}^{N} \psi_i(s)\boldsymbol{\phi}_i^k\right). \tag{A.4}$$

Appendix B

Energy conservation

1. Energy conservation is provided if the test functions

$$\delta\boldsymbol{\eta}(s) = \boldsymbol{\Psi}(s)\underline{\delta\boldsymbol{\eta}}$$

   are interpolated in the same way as the increments $\boldsymbol{\vartheta}$ over the time step. Our algorithm is not such.

2. Numerical evidence suggests that in our algorithm the energy-conservation error reduces with the element length. But can this to be proven?

3. To do so, let us limit our attention to the fist-order interpolation only, where

$$\delta\boldsymbol{\eta}(s) = \frac{L-s}{L}\underline{\delta\boldsymbol{\eta}}_1 + \frac{s}{L}\underline{\delta\boldsymbol{\eta}}_2$$
$$\mathbf{H}(s) = \underline{\mathbf{H}}_1 \exp\left(\frac{s}{L}\hat{\boldsymbol{d}}\right), \qquad \exp(\hat{\boldsymbol{d}}) = \underline{\mathbf{H}}_1^{-1}\underline{\mathbf{H}}_2.$$

   Take the assumption that L is small, such that s/L stays within the range $[0,1]$, but $\boldsymbol{d}$ becomes increasingly smaller, such that $\hat{\boldsymbol{d}}^2 \to \mathbf{0}$ (and also $\widehat{\boldsymbol{\vartheta}_2 - \boldsymbol{\vartheta}_1}^2 \to \mathbf{0}$). Can it be shown that under this assumption the above interpolation for $\mathbf{H}(s)$ is equivalent to

$$\boldsymbol{\vartheta}(s) = \frac{L-s}{L}\underline{\boldsymbol{\vartheta}}_1 + \frac{s}{L}\underline{\boldsymbol{\vartheta}}_2,$$

   where $\mathbf{H}_{n+1}(s) = \exp\left(\hat{\boldsymbol{\vartheta}}(s)\right)\mathbf{H}_n(s)$? If yes, this provides the proof under #2.

4. The actual interpolation for $\boldsymbol{\vartheta}(s)$ is

$$\begin{aligned}\exp(\hat{\boldsymbol{\vartheta}}) &= \mathbf{H}_{n+1}(s)\mathbf{H}_n^{-1}(s)\\ &= \mathbf{H}_{1,n+1}\exp\left(\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}\right)\exp\left(-\frac{s}{L}\hat{\boldsymbol{d}}_n\right)\mathbf{H}_{1,n}^{-1}\\ &= \exp(\hat{\boldsymbol{\vartheta}}_1)\mathbf{H}_{1,n}\sum_{i=0}^{\infty}\frac{1}{i!}\left(\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}\right)^i\sum_{j=0}^{\infty}\left(-\frac{s}{L}\hat{\boldsymbol{d}}_n\right)^j\mathbf{H}_{1,n}^{-1},\end{aligned}$$

where

$$\begin{aligned}\exp\left(\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}\right)\exp\left(-\frac{s}{L}\hat{\boldsymbol{d}}_n\right) &= \mathbf{I}+\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}-\frac{s}{L}\hat{\boldsymbol{d}}_n\\ &\quad+\frac{1}{2!}\left(\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}\right)^2-\left(\frac{s}{L}\right)^2\hat{\boldsymbol{d}}_{n+1}\hat{\boldsymbol{d}}_n+\frac{1}{2!}\left(\frac{s}{L}\hat{\boldsymbol{d}}_n\right)^2\\ &\quad+\frac{1}{3!}\left(\frac{s}{L}\hat{\boldsymbol{d}}_{n+1}\right)^3-\frac{1}{2!}\left(\frac{s}{L}\right)^3\hat{\boldsymbol{d}}_{n+1}^2\hat{\boldsymbol{d}}_n\\ &\quad+\frac{1}{2!}\left(\frac{s}{L}\right)^3\hat{\boldsymbol{d}}_{n+1}\hat{\boldsymbol{d}}_n^2-\frac{1}{3!}\left(\frac{s}{L}\hat{\boldsymbol{d}}_n\right)^3\\ &\quad+\dots\\ &= \mathbf{I}+\frac{s}{L}\left(\hat{\boldsymbol{d}}_{n+1}-\hat{\boldsymbol{d}}_n\right)\\ &\quad+\left(\frac{s}{L}\right)^2\left(\frac{1}{2}\hat{\boldsymbol{d}}_{n+1}^2-\hat{\boldsymbol{d}}_{n+1}\hat{\boldsymbol{d}}_n+\frac{1}{2}\hat{\boldsymbol{d}}_n^2\right)\\ &\quad+\frac{1}{2!}\left(\frac{s}{L}\right)^3\left(\frac{1}{3}\hat{\boldsymbol{d}}_{n+1}^3-\hat{\boldsymbol{d}}_{n+1}\hat{\boldsymbol{d}}_n-\hat{\boldsymbol{d}}_{n+1}\hat{\boldsymbol{d}}_n-\frac{1}{3}\hat{\boldsymbol{d}}_n^3\right)\\ &\quad+\dots\\ &= \mathbf{I}+\frac{s}{L}(\hat{\boldsymbol{d}}_{n+1}-\hat{\boldsymbol{d}}_n)+\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right).\end{aligned}$$

Likewise

$$\begin{aligned}\exp(-\hat{\boldsymbol{\vartheta}}_1)\exp\left(\hat{\boldsymbol{\vartheta}}(s)\right) &= \mathbf{I}-\hat{\boldsymbol{\vartheta}}_1+\hat{\boldsymbol{\vartheta}}(s)\\ &\quad+\frac{1}{2!}\hat{\boldsymbol{\vartheta}}_1^2-\hat{\boldsymbol{\vartheta}}_1\hat{\boldsymbol{\vartheta}}(s)+\frac{1}{2!}\left(\hat{\boldsymbol{\vartheta}}(s)\right)^2\\ &\quad-\frac{1}{3!}\hat{\boldsymbol{\vartheta}}_1^3+\frac{1}{2!}\hat{\boldsymbol{\vartheta}}_1^2\hat{\boldsymbol{\vartheta}}(s)-\frac{1}{2!}\hat{\boldsymbol{\vartheta}}_1\left(\hat{\boldsymbol{\vartheta}}(s)\right)^2+\frac{1}{3!}\left(\hat{\boldsymbol{\vartheta}}(s)\right)^3\\ &\quad+\dots\\ &= \mathbf{I}+\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1+\mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1)^2\right).\end{aligned}$$

5. Hence

$$\mathbf{I}+\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1+\mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1)^2\right)=\mathbf{I}+\frac{s}{L}\widehat{\tilde{\mathbf{H}}_{1,n}(\boldsymbol{d}_{n+1}-\boldsymbol{d}_n)}+\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right),$$

i.e.

$$\hat{\boldsymbol{\vartheta}}(s)=\hat{\boldsymbol{\vartheta}}_1+\frac{s}{L}\widehat{\tilde{\mathbf{H}}_{1,n}(\boldsymbol{d}_{n+1}-\boldsymbol{d}_n)}+\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right)+\mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1)^2\right)$$

where, according to (3.53),

$$\tilde{\mathbf{H}}_{1,n}(\boldsymbol{d}_{n+1}-\boldsymbol{d}_n)=\boldsymbol{\vartheta}_2-\boldsymbol{\vartheta}_1.$$

From here,

$$\begin{aligned}\hat{\boldsymbol{\vartheta}}(s) &= \hat{\boldsymbol{\vartheta}}_1+\frac{s}{L}\widehat{\boldsymbol{\vartheta}_2-\boldsymbol{\vartheta}_1}+\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right)+\mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1)^2\right)\\ &= \frac{L-s}{L}\hat{\boldsymbol{\vartheta}}_1+\frac{s}{L}\hat{\boldsymbol{\vartheta}}_2+\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right)+\mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s)-\hat{\boldsymbol{\vartheta}}_1)^2\right). \qquad (*)\end{aligned}$$

6. Clearly, therefore, the actual interpolation

$$\exp\left(\hat{\boldsymbol{\vartheta}}(s)\right) = \mathbf{H}_{n+1}(s)\mathbf{H}_n^{-1}(s)$$

from which equation $(*)$ is derived, is equivalent to

$$\boldsymbol{\vartheta}(s) = \frac{L-s}{L}\hat{\boldsymbol{\vartheta}}_1 + \frac{s}{L}\hat{\boldsymbol{\vartheta}}_2$$

if

$$\mathcal{O}\left(\hat{\boldsymbol{d}}^2\right) + \mathcal{O}\left((\hat{\boldsymbol{\vartheta}}(s) - \hat{\boldsymbol{\vartheta}}_1)^2\right) = \mathbf{0},$$

which is true following the initial assumption. Hence, for small elements, same interpolation is used for the test and the trial functions leading to increased energy conservation as the mesh is refined.

Appendix C

Useful identities

For $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{3 \text{ or } 6}$ and $\mathbf{Z} \in SO(3)$ or $SE(3)$:

$$\exp\left(\hat{\boldsymbol{x}}\right) = \sum_{i=0}^{\infty} \frac{\hat{\boldsymbol{x}}^i}{i!} \tag{C.1}$$

$$\exp\left(\tilde{\boldsymbol{x}}\right) = \widetilde{(\exp(\hat{\boldsymbol{x}}))} \tag{C.2}$$

$$\exp(\alpha\hat{\boldsymbol{x}})\boldsymbol{x} = \boldsymbol{x} \tag{C.3}$$

$$\mathbf{T}(\alpha\tilde{\boldsymbol{x}})\boldsymbol{x} = \boldsymbol{x} \tag{C.4}$$

$$\mathbf{T}^{-1}(\tilde{\boldsymbol{x}}) - \mathbf{T}^{-1}(-\tilde{\boldsymbol{x}}) = \hat{\boldsymbol{x}} \tag{C.5}$$

$$\exp(\tilde{\boldsymbol{x}}) = \mathbf{I} + \mathbf{T}(-\tilde{\boldsymbol{x}})\tilde{\boldsymbol{x}} \tag{C.6}$$

$$\mathbf{T}(-\tilde{\boldsymbol{x}})\,\mathbf{T}(\tilde{\boldsymbol{x}}) = \mathbf{T}(\tilde{\boldsymbol{x}})\,\mathbf{T}(-\tilde{\boldsymbol{x}}) \tag{C.7}$$

$$\mathbf{T}(-\tilde{\boldsymbol{x}})^{-1}\,\mathbf{T}(\tilde{\boldsymbol{x}}) = \mathbf{T}(\tilde{\boldsymbol{x}})\,\mathbf{T}(-\tilde{\boldsymbol{x}})^{-1} \tag{C.8}$$

$$\mathbf{T}(-\tilde{\boldsymbol{x}})\,\mathbf{T}(\tilde{\boldsymbol{x}})^{-1} = \mathbf{T}(\tilde{\boldsymbol{x}})^{-1}\,\mathbf{T}(-\tilde{\boldsymbol{x}}) \tag{C.9}$$

$$\mathbf{T}(-\tilde{\boldsymbol{x}})^{-1}\,\mathbf{T}(\tilde{\boldsymbol{x}})^{-1} = \mathbf{T}(\tilde{\boldsymbol{x}})^{-1}\,\mathbf{T}(-\tilde{\boldsymbol{x}})^{-1} \tag{C.10}$$

$$\exp(\tilde{\boldsymbol{x}}) = \mathbf{T}(-\tilde{\boldsymbol{x}})\,\mathbf{T}^{-1}(\tilde{\boldsymbol{x}}) \tag{C.11}$$

$$\mathrm{d}\exp\left(\hat{\boldsymbol{x}}\right) = \exp\left(\hat{\boldsymbol{x}}\right)\widehat{\mathbf{T}(\tilde{\boldsymbol{x}})\mathrm{d}\boldsymbol{x}} = \widehat{\mathbf{T}(-\tilde{\boldsymbol{x}})\mathrm{d}\boldsymbol{x}}\exp\left(\hat{\boldsymbol{x}}\right) \tag{C.12}$$

$$\frac{\mathrm{d}T\left(s\tilde{\boldsymbol{x}}\right)}{\mathrm{d}s} = \exp\left(-\tilde{\boldsymbol{x}}(s)\right) - \mathbf{T}(\tilde{\boldsymbol{x}}) = \frac{\mathbf{I}}{s} - \mathbf{T}(s\tilde{\boldsymbol{x}})\left(\frac{\mathbf{I}}{s} + \tilde{\boldsymbol{x}}\right) \tag{C.13}$$

$$\exp(\widehat{\mathbf{Z}\boldsymbol{x}}) = \mathbf{Z}\exp(\boldsymbol{x})\mathbf{Z}^{-1} \tag{C.14}$$

$$\mathbf{T}(\widetilde{\mathbf{Z}\boldsymbol{x}}) = \mathbf{Z}\,\mathbf{T}(\tilde{\boldsymbol{x}})\mathbf{Z}^{-1} \tag{C.15}$$

$$\widehat{\mathbf{Z}\boldsymbol{x}} = \mathbf{Z}\hat{\boldsymbol{x}}\mathbf{Z}^{-1} \tag{C.16}$$

$$\widehat{\hat{\boldsymbol{x}}\boldsymbol{y}} = \hat{\boldsymbol{x}}\hat{\boldsymbol{y}} - \hat{\boldsymbol{y}}\hat{\boldsymbol{x}} \tag{C.17}$$

Appendix D

Nearest point projection

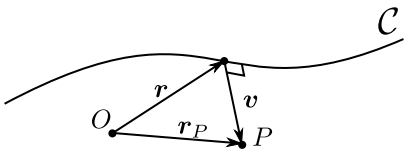


Figure D.1: Nearest point projection

The shortest distance from a curve $\mathcal{C}$ to a point P (Figure D.1) is the magnitude of the vector perpendicular to the curve. The curve could potentially have multiple points $\boldsymbol{r}_i$, where the normal is going through P, however, this algorithm is restricted to finding the local minimum, close to the initial guess. The curve is defined parametrically by the coordinate s and its position is denoted by $\boldsymbol{r}$. Point P has the position vector denoted by $\boldsymbol{r}_P$. The following set of equations needs to be solved

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{f}\left(\begin{Bmatrix}\boldsymbol{v}\\ s\end{Bmatrix}\right) = \begin{Bmatrix}\boldsymbol{r}(s)+\boldsymbol{v}\\ \boldsymbol{r}'(s)\boldsymbol{v}^{\mathrm{T}}\end{Bmatrix}, \tag{D.1}$$

$$\boldsymbol{f}(\boldsymbol{x}_n) = \begin{Bmatrix}\boldsymbol{r}_P\\ 0\end{Bmatrix}, \tag{D.2}$$

where $\boldsymbol{x}$ denotes the combined vector of unknowns and $\boldsymbol{x}_n$ its converged value satisfying equation (D.2). Since the system is nonlinear for non-straight curves $\mathcal{C}$, Newton-Raphson iterative algorithm is employed

$$\boldsymbol{f}(\boldsymbol{x}_n) - \boldsymbol{f}(\boldsymbol{x}_i) = \nabla\boldsymbol{f}(\boldsymbol{x}_i)\Delta\boldsymbol{x}_i \tag{D.3}$$

with the residual vector

$$\boldsymbol{f}(\boldsymbol{x}_n) - \boldsymbol{f}(\boldsymbol{x}_i) = \begin{Bmatrix}\boldsymbol{r}_P\\ 0\end{Bmatrix} - \begin{Bmatrix}\boldsymbol{r}(s)+\boldsymbol{v}\\ \boldsymbol{r}'(s)^{\mathrm{T}}\boldsymbol{v}\end{Bmatrix} \tag{D.4}$$

and the tangent matrix

$$\nabla \boldsymbol{f}(\boldsymbol{x}_i) = \begin{bmatrix} \mathbf{I} & \boldsymbol{r}'(s) \\ \boldsymbol{r}'(s)^{\mathrm{T}} & \boldsymbol{r}''(s)^{\mathrm{T}}\boldsymbol{v} \end{bmatrix}, \tag{D.5}$$

where $\mathbf{I}$ is the identity matrix of rank 3.

Appendix E

Tangent matrix for the mortar method

Final expression for the linearization of the virtual contact work following from (3.101), (3.118) and (3.108)

$$
\begin{aligned}
\Delta\left(\delta\Pi_N^e\right) = \int_0^{L^{(1)}} \Bigg\{ & \delta\boldsymbol{x}^{(1)\mathrm{T}}\boldsymbol{n}^{(2)}\Delta\lambda + \delta\lambda\boldsymbol{n}^{(2)\mathrm{T}}\Delta\boldsymbol{x}^{(1)} \\
& - \delta\boldsymbol{x}^{(1)\mathrm{T}}\frac{\lambda}{\|\boldsymbol{v}\|}\left[\left(\widehat{\boldsymbol{n}}^{(2)}\right)^2 + \frac{1}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\right]\Delta\boldsymbol{x}^{(1)}\Bigg\}\,\mathrm{d}s^{(1)} \\
+ \int_0^{L^{(1)}} \Bigg\{ & \delta\boldsymbol{x}^{(1)\mathrm{T}}\frac{\lambda}{\|\boldsymbol{v}\|}\left[\left(\widehat{\boldsymbol{n}}^{(2)}\right)^2 + \frac{1}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\right]\Delta\boldsymbol{x}^{(2)} \\
& - \delta\boldsymbol{x}^{(1)\mathrm{T}}\frac{\lambda}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{n}^{(2)\mathrm{T}}\Delta\boldsymbol{x}'^{(2)} - \delta\lambda\boldsymbol{n}^{(2)\mathrm{T}}\Delta\boldsymbol{x}^{(2)}\Bigg\}\,\mathrm{d}s^{(1)} \\
+ \int_0^{L^{(1)}} \Bigg\{ & \delta\boldsymbol{x}^{(2)\mathrm{T}}\frac{\lambda}{\|\boldsymbol{v}\|}\left[\left(\widehat{\boldsymbol{n}}^{(2)}\right)^2 + \frac{1}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\right]\Delta\boldsymbol{x}^{(1)} \\
& - \delta\boldsymbol{x}^{(2)\mathrm{T}}\boldsymbol{n}^{(2)}\Delta\lambda - \delta\boldsymbol{x}'^{(2)\mathrm{T}}\frac{\lambda}{S_c}\boldsymbol{n}^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\Delta\boldsymbol{x}^{(1)}\Bigg\}\,\mathrm{d}s^{(1)} \\
+ \int_0^{L^{(1)}} \Bigg\{ & -\delta\boldsymbol{x}^{(2)\mathrm{T}}\frac{\lambda}{\|\boldsymbol{v}\|}\left[\left(\widehat{\boldsymbol{n}}^{(2)}\right)^2 + \frac{1}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\right]\Delta\boldsymbol{x}^{(2)} \\
& + \delta\boldsymbol{x}^{(2)\mathrm{T}}\frac{\lambda}{S_c}\boldsymbol{x}'^{(2)}\boldsymbol{n}^{(2)\mathrm{T}}\Delta\boldsymbol{x}'^{(2)} + \delta\boldsymbol{x}'^{(2)\mathrm{T}}\frac{\lambda}{S_c}\boldsymbol{n}^{(2)}\boldsymbol{x}'^{(2)\mathrm{T}}\Delta\boldsymbol{x}^{(2)} \\
& - \delta\boldsymbol{x}'^{(2)\mathrm{T}}\frac{\lambda}{S_c}\boldsymbol{n}^{(2)}\boldsymbol{v}^{\mathrm{T}}\Delta\boldsymbol{x}'^{(2)}\Bigg\}\,\mathrm{d}s^{(1)}. \qquad \text{(E.1)}
\end{aligned}
$$

This expression can be condensed by using the following substitutions

$$\mathbf{A} = \lambda \frac{1}{\|\boldsymbol{v}\|} \left[\left(\widehat{\boldsymbol{n}}^{(2)} \right)^2 + \frac{1}{S_c} \boldsymbol{x}'^{(2)} \boldsymbol{x}'^{(2)\mathrm{T}} \right] \tag{E.2}$$

$$\mathbf{B} = \lambda \frac{1}{S_c} \boldsymbol{x}'^{(2)} \boldsymbol{n}^{(2)\mathrm{T}} \tag{E.3}$$

$$\mathbf{C} = \lambda \frac{1}{S_c} \boldsymbol{n}^{(2)} \boldsymbol{v}^{(2)\mathrm{T}}, \tag{E.4}$$

resulting in

$$\begin{aligned}
\Delta\left(\delta\Pi_N^e\right) = & \int_0^{L^{(1)}} \left(\delta\boldsymbol{x}^{(1)\mathrm{T}} \boldsymbol{n}^{(2)} \Delta\lambda + \delta\lambda \boldsymbol{n}^{(2)\mathrm{T}} \Delta\boldsymbol{x}^{(1)} - \delta\boldsymbol{x}^{(1)\mathrm{T}} \mathbf{A} \Delta\boldsymbol{x}^{(1)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \left(\delta\boldsymbol{x}^{(1)\mathrm{T}} \mathbf{A} \Delta\boldsymbol{x}^{(2)} - \delta\boldsymbol{x}^{(1)\mathrm{T}} \mathbf{B} \Delta\boldsymbol{x}'^{(2)} - \delta\lambda \boldsymbol{n}^{(2)\mathrm{T}} \Delta\boldsymbol{x}^{(2)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \left(\delta\boldsymbol{x}^{(2)\mathrm{T}} \mathbf{A} \Delta\boldsymbol{x}^{(1)} - \delta\boldsymbol{x}^{(2)\mathrm{T}} \boldsymbol{n}^{(2)} \Delta\lambda - \delta\boldsymbol{x}'^{(2)\mathrm{T}} \mathbf{B}^{\mathrm{T}} \Delta\boldsymbol{x}^{(1)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \left(-\delta\boldsymbol{x}^{(2)\mathrm{T}} \mathbf{A} \Delta\boldsymbol{x}^{(2)} + \delta\boldsymbol{x}^{(2)\mathrm{T}} \mathbf{B} \Delta\boldsymbol{x}'^{(2)} + \delta\boldsymbol{x}'^{(2)\mathrm{T}} \mathbf{B}^{\mathrm{T}} \Delta\boldsymbol{x}^{(2)} - \delta\boldsymbol{x}'^{(2)\mathrm{T}} \mathbf{C} \Delta\boldsymbol{x}'^{(2)} \right) \mathrm{d}s^{(1)}.
\end{aligned} \tag{E.5}$$

Each row describes the interaction of a different combination of nodes belonging to beam 1 and beam 2. Inserting interpolation functions (3.81) for individual node leads to

$$\begin{aligned}
\Delta\left(\delta\Pi_N^e\right)_{ijkl} = & \int_0^{L^{(1)}} \left(\delta\boldsymbol{X}_i^{(1)\mathrm{T}} N_i^{(1)} \boldsymbol{n}^{(2)} \Delta\Lambda_j \Phi_j^{(1)} + \delta\Lambda_i \Phi_i^{(1)} \boldsymbol{n}^{(2)\mathrm{T}} \Delta\boldsymbol{X}_j^{(1)} N_j^{(1)} - \delta\boldsymbol{X}_i^{(1)\mathrm{T}} N_i^{(1)} \mathbf{A} \Delta\boldsymbol{X}_j^{(1)} N_j^{(1)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \left(\delta\boldsymbol{X}_i^{(1)\mathrm{T}} N_i^{(1)} \mathbf{A} \Delta\boldsymbol{X}_l^{(2)} N_l^{(2)} - \delta\boldsymbol{X}_i^{(1)\mathrm{T}} N_i^{(1)} \mathbf{B} \Delta\boldsymbol{X}_l^{(2)} N_l'^{(2)} - \delta\Lambda_i \Phi_i^{(1)} \boldsymbol{n}^{(2)\mathrm{T}} \Delta\boldsymbol{X}_l^{(2)} N_l^{(2)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \left(\delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k^{(2)} \mathbf{A} \Delta\boldsymbol{X}_j^{(1)} N_j^{(1)} - \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k^{(2)} \boldsymbol{n}^{(2)} \Delta\Lambda_j \Phi_j^{(1)} - \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k'^{(2)} \mathbf{B}^{\mathrm{T}} \Delta\boldsymbol{X}_j^{(1)} N_j^{(1)} \right) \mathrm{d}s^{(1)} \\
& + \int_0^{L^{(1)}} \Bigg(- \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k^{(2)} \mathbf{A} \Delta\boldsymbol{X}_l^{(2)} N_l^{(2)} + \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k^{(2)} \mathbf{B} \Delta\boldsymbol{X}_l^{(2)} N_l'^{(2)} \\
& \qquad + \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k'^{(2)} \mathbf{B}^{\mathrm{T}} \Delta\boldsymbol{X}_l^{(2)} N_l^{(2)} - \delta\boldsymbol{X}_k^{(2)\mathrm{T}} N_k'^{(2)} \mathbf{C} \Delta\boldsymbol{X}_l^{(2)} N_l'^{(2)} \Bigg) \mathrm{d}s^{(1)}.
\end{aligned} \tag{E.6}$$

Rewriting this in the matrix form finally reveals (3.119)

$$
\begin{aligned}
\Delta\left(\delta\Pi_N^e\right)_{ijkl} = & \begin{Bmatrix} \delta\boldsymbol{X}_i^{(1)} \\ \delta\Lambda_i \end{Bmatrix}^{\mathrm{T}} \int_0^{L^{(1)}} \begin{bmatrix} -N_i^{(1)}\mathbf{A}N_j^{(1)} & N_i^{(1)}\boldsymbol{n}^{(2)}\Phi_j^{(1)} \\ \Phi_i^{(1)}\boldsymbol{n}^{(2)\mathrm{T}}N_j^{(1)} & 0 \end{bmatrix} \mathrm{d}s^{(1)} \begin{Bmatrix} \Delta\boldsymbol{X}_j^{(1)} \\ \Delta\Lambda_j \end{Bmatrix} \\
& + \begin{Bmatrix} \delta\boldsymbol{X}_i^{(1)} \\ \delta\Lambda_i \end{Bmatrix}^{\mathrm{T}} \int_0^{L^{(1)}} \begin{bmatrix} N_i^{(1)}\mathbf{A}N_l^{(2)} - N_i^{(1)}\mathbf{B}N_l'^{(2)} & \mathbf{0} \\ -\Phi_i^{(1)}\boldsymbol{n}^{(2)\mathrm{T}}N_l^{(2)} & 0 \end{bmatrix} \mathrm{d}s^{(1)} \left\{\Delta\boldsymbol{X}_l^{(2)}\right\} \\
& + \left\{\delta\boldsymbol{X}_k^{(2)}\right\}^{\mathrm{T}} \int_0^{L^{(1)}} \begin{bmatrix} N_k^{(2)}\mathbf{A}N_j^{(1)} - N_k'^{(2)}\mathbf{B}^{\mathrm{T}}N_j^{(1)} & N_k^{(2)}\boldsymbol{n}^{(2)}\Phi_j^{(1)} \\ \mathbf{0}^{\mathrm{T}} & 0 \end{bmatrix} \mathrm{d}s^{(1)} \begin{Bmatrix} \Delta\boldsymbol{X}_j^{(1)} \\ \Delta\Lambda_j \end{Bmatrix} \\
& + \left\{\delta\boldsymbol{X}_k^{(2)}\right\}^{\mathrm{T}} \int_0^{L^{(1)}} \left[-N_k^{(2)}\mathbf{A}N_l^{(2)} + N_k^{(2)}\mathbf{B}N_l'^{(2)} + N_k'^{(2)}\mathbf{B}^{\mathrm{T}}N_l^{(2)} - N_k'^{(2)}\mathbf{C}N_l'^{(2)}\right] \mathrm{d}s^{(1)} \left\{\Delta\boldsymbol{X}_l^{(2)}\right\}
\end{aligned} \tag{E.7}
$$

Appendix F

Unbiased projection algorithm

To obtain s_1 and s_2 from (3.125) and (3.126) we define

$$\boldsymbol{f}(s_1, s_2) = \begin{Bmatrix} (\boldsymbol{x}_1' \ + \boldsymbol{x}_2')^T (\boldsymbol{x}_1 \ - \boldsymbol{x}_2) \\ \frac{1}{2}(s_1 + s_2) - s \end{Bmatrix},$$

and look for the solution of the equation $\boldsymbol{f}(s_1, s_2) = \boldsymbol{0}$. The first-order Taylor series expansion then leads

$$\boldsymbol{f}\left(s_1 + h_1, s_2 + h_2\right) = \boldsymbol{f}(s_1, s_2) + \boldsymbol{\nabla}_{\{s_1, s_2\}} \boldsymbol{f}(s_1, s_2) \begin{Bmatrix} h_1 \\ h_2 \end{Bmatrix} = \boldsymbol{0}, \tag{F.1}$$

where

$$\boldsymbol{\nabla}_{\{s_1, s_2\}} \boldsymbol{f}(s_1, s_2) = \begin{bmatrix} \boldsymbol{p}^T \boldsymbol{x}_1'' + \boldsymbol{t}^T \boldsymbol{x}_1' & \boldsymbol{p}^T \boldsymbol{x}_2'' - \boldsymbol{t}^T \boldsymbol{x}_2' \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

Equation (F.1) is then solved iteratively in the standard Newton-Raphson way.

List of Figures

List of Tables

Bibliography

[Argyris, 1982] Argyris, J. (1982). An excursion into large rotations. *Computer Methods in Applied Mechanics and Engineering*, 32(1):85–155.

[Arnold and Brüls, 2007] Arnold, M. and Brüls, O. (2007). Convergence of the generalized-α scheme for constrained mechanical systems. *Multibody System Dynamics*, 85:187–202.

[Bauchau et al., 1995] Bauchau, O. A., Damilano, G., and Theron, N. J. (1995). Numerical integration of non-linear elastic multi-body systems. *International Journal for Numerical Methods in Engineering*, 38:2727–2751.

[Borri and Bottasso, 1994a] Borri, M. and Bottasso, C. (1994a). An intrinsic beam model based on a helicoidal approximation-part i: formulation. *International Journal for Numerical Methods in Engineering*, 37:2267–2289.

[Borri and Bottasso, 1994b] Borri, M. and Bottasso, C. (1994b). An intrinsic beam model based on a helicoidal approximation—part ii: Linearization and finite element implementation. *International Journal for Numerical Methods in Engineering*, 37:2291–2309.

[Bosten et al., 2022] Bosten, A., Cosimo, A., Linn, J., and Brüls, O. (2022). A mortar formulation for frictionless line-to-line beam contact. *Multibody System Dynamics*, 54(1):31–52.

[Bosten et al., 2023] Bosten, A., Denoël, V., Cosimo, A., Linn, J., and Brüls, O. (2023). A beam contact benchmark with analytic solution. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 103(11):e202200151.

[Bottasso and Borri, 1998] Bottasso, C. L. and Borri, M. (1998). Integrating finite rotations. *Computer Methods in Applied Mechanics and Engineering*, 164(3):307–331.

[Cardona and Géradin, 1989] Cardona, A. and Géradin, M. (1989). Time integration of the equations of motion in mechanism analysis. *Computers & Structures*, 33(4):801–820.

[Carlson and Crilly, 2009] Carlson, A. B. and Crilly, P. B. (2009). *Communication Systems: An Introduction to Signals and Noise in Electrical Communication*. McGraw-Hill Higher Education.

[Chamekh et al., 2009] Chamekh, M., Mani-Aouadi, S., and Moakher, M. (2009). Modeling and numerical treatment of elastic rods with frictionless self-contact. *Computer Methods in Applied Mechanics and Engineering*, 198(47):3751–3764.

[Cheng and Ye, 2002] Cheng, X.-l. and Ye, X. (2002). Analysis of the checkerboard mode and inf–sup condition. *Communications in Numerical Methods in Engineering*, 18(6):451–458.

[Crisfield and Jelenić, 1999] Crisfield, M. A. and Jelenić, G. (1999). Objectivity of strain measures in the geometrically exact three-dimensional beam theory and its finite-element implementation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 455(1983):1125–1147.

[Durville, 2005] Durville, D. (2005). Numerical simulation of entangled materials mechanical properties. *Journal of Materials Science*, 40:5941–5948.

[Durville, 2012] Durville, D. (2012). Contact-friction modeling within elastic beam assemblies: an application to knot tightening. *Computational Mechanics*, 49(6):687–707.

[Farah et al., 2015] Farah, P., Popp, A., and Wall, W. A. (2015). Segment-based vs. element-based integration for mortar methods in computational contact mechanics. *Computational Mechanics*, 55:209–228.

[Gay Neto et al., 2015] Gay Neto, A., Pimenta, P. M., and Wriggers, P. (2015). Self-contact modeling on beams experiencing loop formation. *Computational Mechanics*, 55(1):193–208.

[Goss et al., 2005] Goss, V. G., van der Heijden, G. H., Thompson, J. M., and Neukirch, S. (2005). Experiments on snap buckling, hysteresis and loop formation in twisted rods. *Experimental Mechanics*, 45(2):101–111.

[Hilber et al., 1977] Hilber, H. M., Hughes, T. J. R., and Taylor, R. L. (1977). Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5(3):283–292.

[Hughes et al., 1976] Hughes, T. J., Taylor, R. L., Sackman, J. L., Curnier, A., and Kanoknukulchai, W. (1976). A finite element method for a class of contact-impact problems. *Computer Methods in Applied Mechanics and Engineering*, 8(3):249–276.

[Ibrahimbegovic, 1997] Ibrahimbegovic, A. (1997). On the choice of finite rotation parameters. *Computer Methods in Applied Mechanics and Engineering*, 149(1):49–71. Containing papers presented at the Symposium on Advances in Computational Mechanics.

[Ibrahimbegovic and Mamouri, 2002] Ibrahimbegovic, A. and Mamouri, S. (2002). Energy conserving/decaying implicit time-stepping scheme for nonlinear dynamics of three-dimensional beams undergoing finite rotations. *Computer Methods in Applied Mechanics and Engineering*, 191(37):4241–4258.

[Jelenić and Crisfield, 1999] Jelenić, G. and Crisfield, M. (1999). Geometrically exact 3d beam theory: implementation of a strain-invariant finite element for statics and dynamics. *Computer Methods in Applied Mechanics and Engineering*, 171(1):141–171.

[Kawa and Litewka, 1999] Kawa, O. and Litewka, P. (1999). Contact with friction between 3d beams with deformable circular cross sections. *Engineering Transactions*, 63(4):439–462.

[Litewka, 2005] Litewka, P. (2005). The penalty and lagrange multiplier methods in the frictional 3d beam-to-beam contact problem. *Civil and Environmental Engineering Reports*, No. 1(1):189–207.

[Litewka, 2007] Litewka, P. (2007). Hermite polynomial smoothing in beam-to-beam frictional contact. *Computational Mechanics*, 40(5):815–826.

[Litewka, 2015a] Litewka, P. (2015a). Frictional beam-to-beam multiple-point contact finite element. *Computational Mechanics*, 56(2):243–264.

[Litewka, 2015b] Litewka, P. (2015b). Frictional beam-to-beam multiple-point contact finite element. *Computational Mechanics*, 56(2):243–264.

[Meier et al., 2018] Meier, C., Grill, M. J., Wall, W. A., and Popp, A. (2018). Geometrically exact beam elements and smooth contact schemes for the modeling of fiber-based materials and structures. *International Journal of Solids and Structures*, 154:124–146. Multiscale Modelling of Fibrous and Textile Materials.

[Meier et al., 2016] Meier, C., Popp, A., and Wall, W. A. (2016). A finite element approach for the line-to-line contact interaction of thin beams with arbitrary orientation. *Computer Methods in Applied Mechanics and Engineering*, 308:377–413.

[Meier et al., 2017] Meier, C., Wall, W. A., and Popp, A. (2017). A unified approach for beam-to-beam contact. *Computer Methods in Applied Mechanics and Engineering*, 315:972–1010.

[Neto et al., 2014] Neto, A. G., Pimenta, P. M., and Wriggers, P. (2014). Contact between rolling beams and flat surfaces. *International Journal for Numerical Methods in Engineering*, 97(9):683–706.

[Neto et al., 2015] Neto, A. G., Pimenta, P. M., and Wriggers, P. (2015). Self-contact modeling on beams experiencing loop formation. *Computational Mechanics*, 55:193–208.

[Popp et al., 2009] Popp, A., Gee, M., and Wall, W. (2009). A finite deformation mortar contact formulation using a primal–dual active set strategy. *International Journal for Numerical Methods in Engineering*, 79:1354–1391.

[Puso et al., 2012] Puso, M. A., Sanders, J., Settgast, R., and Liu, B. (2012). An embedded mesh method in a multiple material ale. *Computer Methods in Applied Mechanics and Engineering*, 245-246:273–289.

[Reissner, 1972] Reissner, E. (1972). On one-dimensional finite-strain beam theory: The plane problem. *Zeitschrift für angewandte Mathematik und Physik ZAMP*, 23(5):795–804.

[Romero and Armero, 2002] Romero, I. and Armero, F. (2002). An objective finite element approximation of the kinematics of geometrically exact rods and its use in the formulation of an energy–momentum conserving scheme in dynamics. *International Journal for Numerical Methods in Engineering*, 54(12):1683–1716.

[Simo, 1985] Simo, J. (1985). A finite strain beam formulation. the three-dimensional dynamic problem. part i. *Computer Methods in Applied Mechanics and Engineering*, 49(1):55–70.

[Simo et al., 1992] Simo, J., Tarnow, N., and Wong, K. (1992). Exact energy-momentum conserving algorithms and symplectic schemes for nonlinear dynamics. *Computer Methods in Applied Mechanics and Engineering*, 100(1):63–116.

[Simo and Vu-Quoc, 1986] Simo, J. and Vu-Quoc, L. (1986). A three-dimensional finite-strain rod model. part II: Computational aspects. *Computer Methods in Applied Mechanics and Engineering*, 58(1):79–116.

[Simo and Vu-Quoc, 1988] Simo, J. and Vu-Quoc, L. (1988). On the dynamics in space of rods undergoing large motions — a geometrically exact approach. *Computer Methods in Applied Mechanics and Engineering*, 66(2):125–161.

[Simo et al., 1995] Simo, J. C., Tarnow, N., and Doblare, M. (1995). Non-linear dynamics of three-dimensional rods: Exact energy and momentum conserving algorithms. *International Journal for Numerical Methods in Engineering*, 38(9):1431–1473.

[Solberg et al., 2007] Solberg, J. M., Jones, R. E., and Papadopoulos, P. (2007). A family of simple two-pass dual formulations for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 196(4):782–802.

[Sonneville et al., 2017] Sonneville, V., Brüls, O., and Bauchau, O. A. (2017). Interpolation schemes for geometrically exact beams: A motion approach. *International Journal for Numerical Methods in Engineering*, 112(9):1129–1153.

[Sonneville et al., 2014] Sonneville, V., Cardona, A., and Brüls, O. (2014). Geometrically exact beam finite element formulated on the special euclidean group se(3). *Computer Methods in Applied Mechanics and Engineering*, 268:451–474.

[Sonneville and Géradin, 2023] Sonneville, V. and Géradin, M. (2023). Two-field formulation of the inertial forces of a geometrically-exact beam element. *Multibody System Dynamics*, 59(2):239–254.

[Spurrier, 1978] Spurrier, R. A. (1978). Comment on " singularity-free extraction of a quaternion from a direction-cosine matrix". *Journal of Spacecraft and Rockets*, 15(4):255–255.

[Tomec and Jelenić, 2022] Tomec, J. and Jelenić, G. (2022). Analysis of static frictionless beam-to-beam contact using mortar method. *Multibody System Dynamics*, 55(3):293–322.

[Tomec and Jelenić, 2024a] Tomec, J. and Jelenić, G. (2024a). Frictionless line-to-line contact: Comparison of mortar and unbiased formulation in combination with lagrange-multiplier and penalty method. *International Journal of Solids and Structures*, 286-287:112586.

[Tomec and Jelenić, 2024b] Tomec, J. and Jelenić, G. (2024b). Momentum and near-energy conserving/decaying time integrator for beams with higher-order interpolation on se(3). *Computer Methods in Applied Mechanics and Engineering*, 419:116665.

[Van der Heijden et al., 2003] Van der Heijden, G. H., Neukirch, S., Goss, V. G., and Thompson, J. M. (2003). Instability and self-contact phenomena in the writhing of clamped rods. *International Journal of Mechanical Sciences*, 45(1):161–196.

[Wriggers and Simo, 1985] Wriggers, P. and Simo, J. C. (1985). Note on Tangent Stiffness for Fully Nonlinear Contact Problems. *Communications in Numerical Methods in Engineering*, 1(5):199–203.

[Wriggers and Zavarise, 2004] Wriggers, P. and Zavarise, G. (2004). *Computational Contact Mechanics*, chapter 6. John Wiley & Sons, Ltd.

[Zavarise and De Lorenzis, 2009] Zavarise, G. and De Lorenzis, L. (2009). A modified node-to-segment algorithm passing the contact patch test. *International Journal for Numerical Methods in Engineering*, 79(4):379–416.

[Zavarise and De Lorenzis, 2011] Zavarise, G. and De Lorenzis, L. (2011). *The Contact Patch Test for Linear Contact Pressure Distributions in 2D Frictionless Contact*, pages 79–100. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Zienkiewicz et al., 2005] Zienkiewicz, O. C., Taylor, R. L., and Zhu, J. Z. (2005). *The Finite Element Method: Its Basis and Fundamentals, Sixth Edition*. Butterworth-Heinemann, 6 edition.

[Zupan and Zupan, 2018] Zupan, E. and Zupan, D. (2018). On conservation of energy and kinematic compatibility in dynamics of nonlinear velocity-based three-dimensional beams. *Nonlinear Dynamics*, 95(2):1379–1394.