ABSTRACT

Title of dissertation: INTERPOLATION OF RIGID-BODY MOTION AND GALERKIN METHODS FOR FLEXIBLE MULTIBODY DYNAMICS

Shilei Han
Doctor of Philosophy, 2019

Dissertation directed by: Professor Olivier A. Bauchau
Department of Aerospace

Traditionally, flexible multibody dynamics problems are formulated as initial value problems: initial states of the system are given and solving for the equations of motion yields the dynamic response. Many practical problems, however, are boundary rather than initial value problems; two-point and periodic boundary problems, in particular, are quite common. For instance, the trajectory optimization of robotic arms and spacecrafts is formulated as a two-point boundary value problem; determination of the periodic dynamic response of helicopter and wind turbine blades is formulated as a periodic boundary value problem; the analysis of the stability of these periodic solutions is another important of problem.

The objective of this thesis is to develop a unified solution procedure for both initial and boundary value problems. Galerkin methods provide a suitable framework for the development of such solvers. Galerkin methods require interpolation schemes that approximate the unknown rigid-body motion fields. Novel interpolation schemes for rigid-body motions are proposed based on minimization

of weighted distance measures of rigid-body motions. Based on the proposed interpolation schemes, a unified continuous/discontinuous Galerkin solver is developed for the formulation of geometrically exact beams, for the determination of solutions of initial and periodic boundary value problems, for the stability analysis of periodic solutions, and for the optimal control/optimization problems of flexible multibody systems.

INTERPOLATION OF RIGID-BODY MOTION AND GALERKIN METHODS FOR FLEXIBLE MULTIBODY DYNAMIC

by

Shilei Han

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2019

Advisory Committee:
Professor Olivier A. Bauchau, Chair/Advisor
Professor Howard Elman
Professor Sung Lee
Professor Derek A. Paley
Professor Christine M. Hartzell

Acknowledgments

First and foremost I would like to thank my advisor, Professor Olivier A. Bauchau for giving me an opportunity to work on challenging and interesting projects over the past three years. He has always made himself available for help and advice for both of my research and life. It has been a great pleasure to work with and learn from such an extraordinary individual.

I would like to thank Professor Howard Elman, Professor Sung Lee, Professor Derek A. Paley, and Professor Christine M. Hartzell for agreeing to serve on my thesis committee and for sparing their invaluable time giving me advice on my thesis and reviewing the manuscript. I would also like to thank Professor Elman for teaching me iterative solvers and spending time to answer my questions. I would like to thank Professor Paley for teaching me analysis of nonlinear systems.

I would like to thank my colleagues, Valentin Sonneville and Alfonso Callejo for helping me on programming, Tyler Sinotte, Nishant Nemani, and Sheng-wei Wang for helping me getting familiar with the department and laboratory. I would also like to thank Tom Hurst and Matthew J. Sinclair for helping me understand the academic requirements of the Phd program.

I owe my deepest thanks to my family - my mother, father, and two elder sisters who have always stood by me and guided me through my career.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Motivation

Traditionally, flexible multibody dynamics problems are formulated as initial value problems: initial states (including configuration and velocities) of the system are given and solution of the equations of motion yields the dynamic response. Many practical problems, however, are boundary rather than initial value problems; two-point and periodic boundary problems, in particular, are quite common. Consider, for instance, the following problems:

- optimal control (or trajectory optimization) of robotic arms and spacecrafts, which involves determining the optimal control inputs that move the vehicle from a starting state to a specified terminal state and minimize a given cost function simultaneously;

- determination of the periodic dynamic response of helicopter and wind turbine blades, of rotating shafts and blades in power generation turbines, turboshaft engines, and jet engine, of internal combustion engines, and of many flexible mechanisms.

The motion of multibody systems is governed by differential-algebraic equa-

tions (DAEs). Accordingly, optimal control of flexible multibody dynamics is formulated as a DAE-constrained optimization problem, where the boundary values of DAEs are specified. In some cases, the terminal state can be partially fixed or totally free, which leads to mixed initial-boundary value problem. A closely related topic is the optimization problem, which involves determining the optimal geometric and material parameters such that an objective function is minimized. In most cases, only the initial values are specified in optimization problem.

The investigation of periodic boundary value problems involves two intertwined tasks: first, the determination of the periodic response of the system, and second, the analysis of the stability of this periodic solution. Different solvers have been developed to determine solutions of initial and periodic boundary value problems; the characterization of the stability of these periodic solutions is based on yet a different set of numerical tools.

The objective of this thesis is to develop a unified solution procedure for both of initial and boundary value problems. Galerkin methods provide a suitable framework for the development of such solvers. Galerkin methods require interpolation schemes that approximate the unknown rigid-body motion fields. Novel integration schemes for rigid-body motions are proposed based on minimization of the weighted distance measures of rigid-body motions. Based on the proposed interpolation schemes, a unified continuous/discontinuous Galerkin solver is developed for the formulation of geometrically exact beams, determination of solutions of initial and periodic boundary value problems, stability analysis of periodic solutions, and optimal control/optimization problems of flexible multibody systems.

1.2 Literature review

Prior research related to kinematics of rigid-body motion, interpolation, beam formulation, time integration schemes for initial value problems, solvers for periodic problems, stability analysis, optimal control and optimization problems are reviewed in this section.

1.2.1 Kinematics of rigid-body motion

Dual numbers were introduced in the 19th century by Clifford [1]. Typically, they are written as $\hat{a} = a + \epsilon\, b$, where a and b are referred to as the primal and dual parts, respectively, and parameter ϵ is such that $\epsilon^n = 0$ for $n \geq 2$.

Application of dual number to kinematics of rigid-body motion is now well established, see Yang and Freudenstein [2], Dimentberg [3], or the textbooks of Bottema and Roth [4] and McCarthy [5]. The geometric interpretation of the rather abstract concept of dual numbers is described by Angeles [6] and Pennestrì and Stefanelli [7] have explored the associated numerical algorithms. Their application to dynamics has been explored by Keler [8] and Brodsky and Shoham [9, 10]. A comprehensive review of the application of dual numbers to various fields is given by Fischer [11]. More recent presentations focusing on computational issues include those of Condurache and Burlacu [12], or Han and Bauchau [13].

Despite the efficient and elegant manner by which dual numbers deal with rigid-body motion, their use has remained limited to the field of kinematics. Although rigid-body motion is a key concept in rigid and flexible multibody dynamics,

dual numbers are rarely mentioned in these fields. Yet, the implementation of rigid multibody formulations requires extensive operations of motion and these operation can be simplified dramatically with the help of dual number algebra.

1.2.2 Interpolation of motion

Interpolation techniques in Euclidean space are well established. Difficulties arise when the same techniques are applied to rotation and motion fields. Indeed, rotations form the **S**pecial **O**rthogonal group, denoted SO(3), and motions form the **S**pecial **O**rthogonal group in $\mathbb{D}^{3\times 3}$, denoted $\mathcal{SO}(3)$, which is isomorphic to the **S**pecial **E**uclidean group SE(3).. Interpolation schemes for rotation and motion fall into four broad categories: the interpolation of rotation (motion) increments and vectors, embedding-based approaches, geodesic-based approaches, and minimization-based approaches.

Rotation increments and rotation parameter vectors live in the Euclidean space and interpolation scheme in Euclidean space can be applied to these quantities directly. Such schemes has been used in the early stages of application of the finite element method to geometrically exact beams [14–16]. As first noted by Crisfield and Jelenić [17], interpolation schemes should be *path-independent*, i.e., solutions are defined by the final configuration only, and *objective*, i.e., the strain measures are invariant under the superposition of a rigid-body motion. Interpolation of rotation increments results in path-dependent solutions, while interpolation of rotation parameter vectors is not objective. To remedy the situation, Crisfield and Jelenić [17]

proposed the interpolation of relative rotation parameter vectors and showed that this approach was both path-independent and objective. Similarly, interpolation of relative motion parameter vectors is both path-independent and objective [18].

Rotation matrix R and unit quaternion $\hat{e}$ are embedded in Euclidean spaces $\mathbb{R}^9$ and $\mathbb{R}^4$, respectively, and are subjected to six and one orthonormality constraints, respectively. The interpolated rotation is obtained by interpolating the rotation matrix or quaternions in Euclidean space $\mathbb{R}^9$ or $\mathbb{R}^4$, respectively, and mapping the result back onto the manifold via the closest point projection [18–21], which typically corresponds to a re-normalization operation. Motion interpolation algorithms are found by developing similar schemes operating on dual quantities [22, 23]. Embedding based approaches lead to path-independent and objective schemes in the framework of finite element methods. Several authors [24, 25] have noted that the re-normalization operation can be skipped altogether: constraints are enforced at the nodes only. Romero [20] pointed out that this practice leads to softer elements.

In geodesic-based approaches, the linear operation in Euclidean space is replaced by a geodesic operation on manifold SO(3) and $\mathcal{SO}(3)$, for rotation and motion, respectively. In the Euclidean space, the geodesic line between two points is the straight line joining these two points and interpolated quantities lie along this line. Similarly, the geodesic line on SO(3) is the arc of big circle passing through two rotations and interpolated rotations should lie along this line. Based on this idea, Shoemake [26] proposed the spherical linear interpolation (SLERP) for quaternions. He also proposed a higher-order interpolation scheme for quaternions on Bézier curves based on the recursive application of SLERP with the aid of de Casteljau's

algorithm. Park and Ravani [27] investigated Bézier curve interpolation on Riemannian manifold by means of matrix exponentials and logarithms.

These ideas were extended to the interpolation of motion by operating on dual quaternions, leading to the Screw Linear Interpolation (ScLERP) [13, 22]. SLERP and ScLERP produce constant-curvature interpolations, which have been used in geometrically exact beam formulations by Borri and Bottasso [28] and Sonneville *et al.* [29] to construct constant-strain elements. Alternatively, Merlini and Morandini [30, 31] have shown that interpolation scheme can be recast as vanishing of the weighted summation of the logarithm of the relative motion. Sonneville et al. [32] extended this interpolation technique by using an arbitrary motion parameter vector for relative motion.

In minimization-based approaches, the interpolation is recast to a minimization problem: minimizing the weighted summation of distance. Minimization problem can be used to derive interpolation or averaging schemes in any space provided that the selected distance function is a valid measured of distance in that space. Based on this idea, Pennec [33] formulated the rotation averaging problem as a minimization problem: he defined the objective function as the weighted summation of the geodesic distances from the average rotation to the rotations at the grid points. Buss and Fillmore [34] formulated the rotation interpolation problem as a minimization problem and also proved the existence and uniqueness of the minimum. Buss and Fillmore's approach leads to geodesic-based interpolation, i.e., SLERP and the interpolation scheme proposed by Merlini and Morandini [30, 31]. A similar approach was followed by Sander [35] to develop a geodesic-based finite element

method. Rotation averaging schemes based on other distance functions have been investigated by numerous authors [36–40]. Rotation interpolation schemes based on other distance functions, however, has not been explored.

1.2.3 Formulations of geometrically exact beam

Geometrically exact beam models were developed by Reissner [41] based on the rigid cross-section assumption. Simo [14, 42] generalized the formulation to beams undergoing large motion. When dealing with beams presenting complex sectional geometries and made of anisotropic laminated composite materials, sectional in- and out-of-plane warping have been shown [43–45] to alter stress distributions and sectional stiffness properties significantly and hence, the rigid-section assumption is no longer valid. For these problems, Hodges and his coauthors [44, 46, 47] have shown that the three-dimensional nonlinear problem decomposes into a nonlinear, one-dimensional analysis along the reference line and a linear, two-dimensional analysis over the cross-section. Their approach is based on variational asymptotic methods and on the decomposition of the rotation tensor. More recently, Bauchau and Han [48–54] have proposed a reduction procedure based on the Hamiltonian formalism that brings the three-dimensional, nonlinear elasticity problem to geometrically exact beam problems.

Because geometrically exact beams are Cosserat material lines, their kinematic description involves both displacement and rotation fields and it is customary to treat these two fields independently, following the footsteps of Simo [42] and of

numerous other researchers [15, 17, 24, 55]. The key to the rigorous description of geometrically exact beams is the treatment of finite rotation, a topic that has been the subject of intensive investigation [56–59]. Indeed, rotation tensors form the special orthogonal group SO(3), in contrast with the displacement field that forms an Euclidean space. As underlined by numerous authors [18, 20, 60], the traditional interpolation techniques of finite element methods cannot by used to interpolate the rotation tensor. Although suitable techniques have been developed based on the vectorial parameterization of rotation, the relationship between the strain components and rotation vectors is highly nonlinear, leading to complex equations of motion.

To avoid the complexity introduced by finite rotations, intrinsic formulations has been proposed by Hegemier and Nair [61] and Hodges [62, 63]. In intrinsic formulations, the unknowns are the sectional strains and velocities: displacement and rotation variables are eliminated. The resulting equations of motion exhibit low-order nonlinearities and space-time conservative schemes can be developed easily. With this formulation, however, the assembly of the beam elements is more complex because displacement and rotation variables do not show up explicitly. Zupan and Saje [64,65] developed a novel beam element based on the interpolation of the strain field, which is integrated to yield the displacement and rotation fields. The element is free of locking because shear and axial strains are interpolated directly. The strain based formulation, however, requires more computational effort than that based on displacement and rotation fields.

In recent years, a new approach to the description of the kinematics of beams

has been developed by Borri and Bottasso [28], McRobie and Lasenby [66], Merlini and Morandini [67], Sander [68], Sonneville et al. [69], and Demoures et al. [70]. In this approach, called the motion formalism, the displacement and rotation fields are treated as a unit that forms the special Euclidean group SE(3). This unified treatment of the displacement and rotation fields leads to simple governing equations presenting low-order algebraic nonlinearities and simplifies time integration for dynamic problems. As was the case for the interpolation of rotation, the interpolation of motion is a thorny issue that must be treated carefully, as discussed by numerous authors [18, 32, 67, 71]. In general, these schemes produce complicated expressions for the curvatures at the Gauss points and the evaluation of the elastic forces and stiffness matrix of beam elements becomes arduous. To alleviate this problem, many authors [24, 25, 72] simplify the expressions for the curvatures, leading to more compliant beam elements that although incompatible, converge under mesh refinement.

1.2.4 Initial value problem of multibody dynamics

Numerous time integration schemes have been applied to the solution of the differential-algebraic equations governing flexible multibody systems. Classical approaches to the solution of initial value problems include backward finite difference schemes [73] and implicit Runge-Kutta schemes [57, 74, 75]. In recent years, the modeling of flexible multibody systems has become based on the finite element method, as explained in the textbooks of Géradin and Cardona [76] or Bauchau [77], lead-

ing to far stiffer equations of motion. In this approach, asymptotic annihilation of the fictitious high-frequency modes resulting from the finite element discretization becomes indispensable, prompting the use of a different set of integrators presenting unconditional stability and asymptotic annihilation, such as the Hilber-Hughes-Taylor scheme (HHT) [76, 78, 79] and generalized-α scheme [80, 81]. Integrators that do not present these characteristics lead to divergent oscillations for the accelerations and Lagrange multipliers [82].

Discontinuous Galerkin method has been applied for the time integration of flexible multibody and it leads to unconditionally stable and asymptotically annihilating schemes. Hughes and Hulbert [83, 84] applied space-time finite element formulations to elasto-dynamics problems and used classical polynomial interpolation schemes in both space and time. Unfortunately, these classical schemes developed for linear fields cannot be used for the nonlinear configuration manifolds, such as finite rotation or rigid-body motion, that describe the kinematics of multibody systems. To bypass this difficulty, many authors [57,85–87] simply extended the existing schemes to accommodate the nonlinear rotation and motion fields. This approach leads to a reduction of accuracy of the schemes, which was later explained in a rigorous manner by Borri and Bottasso [88] who developed a general framework for interpreting finite element formulations in time. For instance, the time-discontinuous Galerkin scheme using two-node elements in time is third-order accurate for linear structural dynamics but reduces to second order for nonlinear multibody dynamics.

1.2.5 Periodic problem and stability analysis

All too often, the solution of periodic problems is obtained using solvers for initial value problems: starting from arbitrary initial conditions, time stepping algorithms are used to obtain the dynamic response of the system. Physical, or numerical energy dissipation, or both then bring the system to its periodic state, once the transient have died out. This approach suffers from serious drawbacks: first, it is not clear when a truly periodic solution is obtained and second, the method becomes ineffective for systems presenting low levels of damping, such as most systems found in aerospace applications.

Initial values solvers can also be used to find the solution of period problems via iteration, a process called the "shooting method." Unfortunately the shooting method is inefficient for flexible multibody dynamics: typically, a large number of iterations is required due to the high condition number of the system.

Periodic boundary value problems can be solved directly by using continuous or discontinuous Galerkin methods. Test and trial functions are chosen to be harmonic or periodic sinc functions, leading to the harmonic balance or Fourier collocation approach [23], respectively. These methods are computationally expensive because they involve Jacobian matrices of large bandwidth: indeed, for nonlinear problems, all harmonics of the system are coupled. Test and trial functions can also be selected as polynomials with a local support in time. This approach, often called the "finite element method in time" method, was pioneered by Bailey [89] and Leipholtz [90]. Applications involving finite element techniques were developed

shortly thereafter [85, 86, 91, 92] and lead to Jacobian matrices of much reduced bandwidth.

Two distinct approaches have been used for the analysis of the stability of periodic problems: Floquet's [93–98] and Hill's method [98–102]. As was the case for the harmonic balance method, Hill's method is based on a harmonic expansion of the solution, leading to a large eigenvalue problem, theoretically of infinite size. Furthermore, Hill's method requires the use of de-aliasing techniques, further increasing its computational cost. Floquet's method is based on the evaluation of the state transition matrix for one period. Stability characteristics of the system are related to the dominant eigenvalues of this state transition matrix, called the monodromy matrix. The monodromy matrix can be evaluated via time integration of the linearized governing equations [96] or by matrix elimination in Galerkin methods [85]. Bauchau et al. [103, 104] developed an approach, called the "implicit transition matrix approach," that evaluates the dominant eigenvalues of the monodromy matrix without its explicit evaluation.

1.2.6 Optimal control and optimization problem

Two fundamental approaches have been used for the DAE-constrained optimization problem: the nested iteration (or black-box) and simultaneous iteration (or all-at-once, direct transcription) methods.

In the nested iteration method [105–110], an existing DAE solver is embedded into the optimization loop. The DAE solver provides solutions of state variables,

gradients of state variables with respect to control inputs or design variables, and possibly seconder-order derivatives to the outer optimization solver. Typically, the gradients and seconder-order derivatives are evaluated using adjoint method or algorithmic differentiation. The optimization solver solves a reduced optimization problem where the state variables are eliminated numerically. Because the DAEs are nonlinear in general, each solving step for the state variables consists of several Newton iterations until the solutions converges. On the other hand, there is no need to satisfy constraint equations exactly in the early stages of optimization process when the state variables are far from their optimal values.

The simultaneous iteration method [111–113] solves the full size optimization problem directly, i.e., the state variables, control inputs and design variables are treated as independent variables. The method leads to a large scale nonlinear programming problem (NLP). Usually, second-order optimization algorithms such as sequential quadratic programming (SQP) are used to solve the resulted NLP. The simultaneous iteration method is more efficient because both of the DAEs and optimality condition are satisfied only at the final iteration.

1.3 Thesis contributions

The main objective this thesis is to develop an unified Galerkin solver for the beam formulation, initial and periodic boundary value problems, stability anlysis of periodic solutions, optimal control and optimization problems of flexible multibody dynamics. The thesis does contain several new developments in each topics it

focused.

Interpolation of rigid-body motion It it found that interpolation schemes in the Euclidean space can be recast as minimization problems for weighted distance metric. This observation allows the straightforward generalization of interpolation in the Euclidean space to interpolation on manifolds, provided that a metric of the manifold is defined. Four metrics of the motion manifold are proposed: the matrix, quaternion, vector, and geodesic metrics. For each of these metrics, the corresponding interpolation schemes are derived and their advantages and drawbacks are discussed.

Beam formulation The proposed motion interpolation schemes yield closed-form expressions for curvatures at the mesh nodes. Based on this fact, the spectral element formulations are proposed. The expressions for the internal forces and tangent stiffness matrices are simplified. Consequently, the proposed spectral element formulation is much easier to implement than its conventional counterpart.

Initial value problem The discontinuous Galerkin scheme is proposed based on the dual spherical linear interpolation (dual-SLERP). The proposed scheme is third-order accurate for both of rigid and flexible multibody dynamics. While the traditional approaches applied approximation on interpolation of rotations and motions and the approximation leads to a second-order scheme.

Periodic boundary value problem and stability analysis The continuous and discontinuous Galerkin methods are developed based on the dual-SLERP. It is found that the discontinuous method is third- and second-order accurate for non-stiff and stiff problems, respectively. For stability analysis, the monodromy matrix can be

constructed directly from the Jacobian matrix in both the continuous and discontinuous Galerkin methods. The discontinuous Galerkin method works well for stability analysis of flexible systems. While the continuous Galerkin method, without upwind mechanism, fails for stability analysis of flexible multibody systems.

Optimization and optimal control problems The simultaneous iteration method is used. Both of the continuous and discontinuous Galerkin methods are applied for discretization of the optimization problem. Exact Hessian matrices are derived to yield a decrease of the number of iterations and simultaneously a decrease of the overall computational time.

Related publications are listed

- Han, S.L. and Bauchau, O.A.: "Manipulation of Motion Via Dual Entities." *Nonlinear Dynamics*, 85(1): 509-524, 2016. (Materials of Chap. 2.)

- Han, S.L. and Bauchau, O.A.: "Spectral Methods for the Periodic Solution of Flexible Multibody Dynamics." *Nonlinear Dynamics*, 92(4): pp 1599-1618, 2018. (Materials of Chap 3).

- Han, S.L. and Bauchau, O.A.: "On the Global Interpolation of Motion." *Computer Methods in Applied Mechanics and Engineering*, 337: 352-386, 2018. (Materials of Chap. 3.)

- Han, S.L. and Bauchau, O.A.: "Spectral Elements Formulation for Geometrically Exact Beam." To appear in *AIAA Jounal*. (Materials of Chap. 4.)

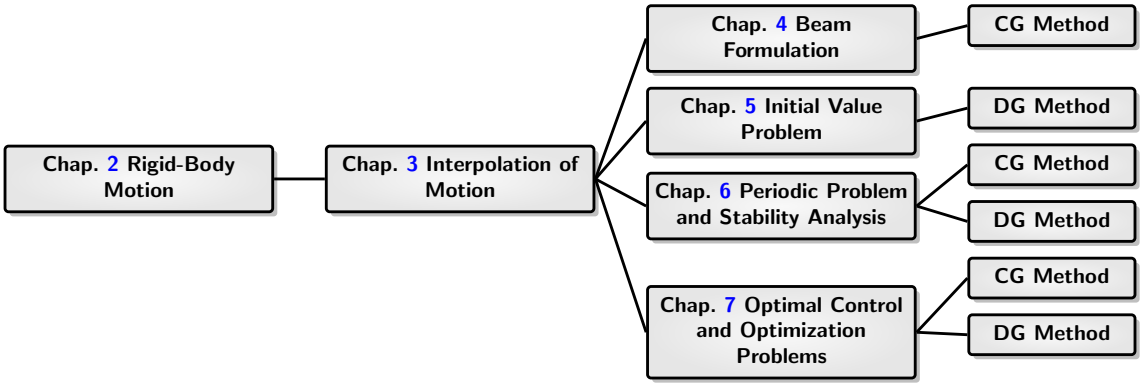

Figure 1.1: Content of the thesis.

1.4 Thesis organization

The contents of this thesis is summarized in table 1.1. The kinematics of rigid-body motion is investigated in Chap. 2. Chap. 3 focuses on the development of interpolation schemes for rigid-body motions. Chap. 4, 5, 6, and 7 focus on applications of the proposed interpolation schemes and Galerkin methods to four types of problem: *(1)* beam formulation, *(2)* initial value problem (or time integration of dynamics equations), *(3)* determination of periodic solutions and stability analysis of these solutions; *(4)* optimization and optimal control problems. Detailed content for each chapter are introduced as follows.

Rigid-Body Motion Dual numbers are introduced in section 2.1. The Plücker coordinates of lines and their dual-vector representations are introduced in section 2.2. The dual orthogonal matrix representation of rigid-body motions is discussed in section 2.3. Rodrigues' formula is proved in section 2.4; the formula relates the dual orthogonal matrix representation with Chasles' axis and dual angle. The dual quaternion representation of rigid-body motion is discussed in section 2.5; this

section also consists of composition rules of quaternions, trace identities that will be used in Chap. 3, and quaternion extraction algorithms. The generic vectorial representation of rigid-body motion is introduced in section 2.6; three specific vectorial parameterization: the Cartesian, linear, and Euler-Rodrigues motion parameters are introduced. For the application in dynamics and elasticity, derivatives, variation, and increments of rigid-body motion are introduced in section 2.7. Second-order derivatives of rigid-body motion and identities of commutativity are discussed in section 2.8. The tangent tensor, which relates the derivatives of rigid-body motion with derivatives of motion parameters, is introduced in section 2.9. The concept of tangent space, inner product and Riemann metric are introduced in section 2.10; inner product and Riemann metric are defined with the help of *extended notations*, which transform dual matrices and vectors of size 3 to real matrices and vectors of size 6. With the help of extended notations, gradient and Hessian of functions of rigid-body motions are investigated in section 2.11; it is shown that the Hessian is unsymmetric. Solving nonlinear equations and optimization problem by using Newton method is discussed in section 2.12.

Interpolation of Motion Staring with a review on interpolation schemes in Euclidean space, section 3.1, the interpolation problem is formulated as a minimization of weighted distance functions in section 3.2. Four types of distance between two rigid-body motions are defined in section 3.3. Interpolation schemes based on these four distance functions are investigated in sections 3.4, 3.5, 3.6, and 3.7. The motion increments and curvatures resulting from interpolations are investigated in section 3.8. Numerical examples are given in section 3.9.

Beam Formulation Kinematics of the problem is investigated in section 4.1. The proposed interpolation schemes is applied for beam problem in section 4.2. The weak and finite element formulations are obtained in section 4.3 and 4.4, respectively. Numerical examples are presented in section 4.5.

Initial Value Problem Lagrangian and Hamiltonian of a typical multibody system are obtained in section 5.1. In section 5.2, the weak formulations of continuous and discontinuous Galerkin methods for periodic and non-periodic problems are derived by using Hamiltonian variation principle. The discontinuous Galerkin method is applied for initial value problem. Introducing the test and trial functions in section 5.3 to the weak form leads to governing equations of initial value problem, as discussed in section 5.4. Numerical examples including time integration of rigid and flexible body dynamics are presented in section 5.5.

Periodic Problem and Stability Analysis This chapter starts with preliminaries of periodic problem in section 6.1, which consists of Floquet's theorem, Floquet's and Hill's methods for stability analysis, and a comparison of Floquet's and Hill's method for multi-dimensional Mathieu equations. The continuous Galerkin method is applied for determination of periodic solutions in section 6.2 and stability analysis in section 6.3. Post processing for continuous Galerkin method is investigated in section 6.4. The discontinuous Galerkin method is applied for determination of periodic solutions in section 6.5 and stability analysis in section 6.6. Post processing for discontinuous Galerkin method is investigated in section 6.7. Numerical examples are presented in section 6.8.

Optimal Control and Optimization Problems This chapter starts with a math-

ematical statement for optimal control and optimization problems in section 7.1. Both of continuous and discontinuous Galerkin method are applied for discretization in section 7.2. The first-order optimality condition are obtained in sections 7.3 and 7.4, for discontinuous and continuous Galerkin methods respectively. Newton method is applied to solve the nonlinear equations resulting from optimality condition and a solving strategy is proposed to deal with the structure matrix in section 7.5. Hessian matrices for both the continuous and discontinuous Galerkin methods are obtained in section 7.6. Numerical examples are presented in section 7.7.

Chapter 2: Preliminary of rigid-body motion

This chapter provides definitions, notations, and identities for rigid-body motions that will be used in the following chapters. Sections 2.1 to 2.9 focus on kinematic of rigid-body motions. Sections 2.10 to 2.12 provide materials used in static and dynamics.

2.1 Dual numbers

The classical notation for dual scalars is

$$\boldsymbol{a} = a + \epsilon\, a^{\circ}, \tag{2.1}$$

where a and a° are the primal and dual parts of the dual scalar. The domain of dual numbers is denoted $\mathbb{D}$ and hence, $\boldsymbol{a} \in \mathbb{D}$. Bookkeeping parameter ϵ is such that $\epsilon^n = 0$ for $n \geq 2$. The n^{th} power of a dual scalar is obtained easily

$$\boldsymbol{a}^n = a^n + \epsilon\, a^{\circ} n a^{n-1}. \tag{2.2}$$

A function of a dual variable is itself a dual scalar written as $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{a})$, or more explicitly, $f = f(a, a^{\circ})$ and $f^{\circ} = f^{\circ}(a, a^{\circ})$. The dual functions to be used here

are required to be analytic [114], which implies that they can be written as

$$\boldsymbol{f}(\boldsymbol{a}) = \sum_{n=0}^{\infty} c_n(\boldsymbol{a} - \boldsymbol{a}_0)^n, \tag{2.3}$$

for any $\boldsymbol{a}_0$. Using eq. (2.2) to express the powers of the dual scalar leads to $f = \sum_{n=0}^{\infty} c_n(a - a_0)^n$, which implies that $f = f(a)$ is a real analytic function of variable a only and $f^\circ = (a^\circ - a_0^\circ)\sum_{n=0}^{\infty} nc_n(a - a_0)^{n-1} = (a^\circ - aa_0^\circ)f'$, for any a_0°, which implies $f^\circ = a^\circ f'$, where notation $(\cdot)'$ indicates a derivative with respect to a.

In summary, analytic dual functions must present the following form

$$\boldsymbol{f}(\boldsymbol{a}) = f(a) + \epsilon\, a^\circ f'(a). \tag{2.4}$$

Two important observations can be made: *(1)* the primal part of an analytic function depends on the primal part of its dual variable only and *(2)* the dual part of an analytic function is a linear function of the dual part of its dual variable.

2.2 Representing lines by dual vectors

A straight line, denoted $\mathcal{L} = (\underline{x}_P, \bar{\ell})$, is defined by the position vector, $\underline{x}_P$, of an arbitrary point $\mathbf{P}$ on the line, and the unit vector, $\bar{\ell}$, along the direction of the line. Alternatively, a line can be represented by a unit dual vector

$$\bar{\boldsymbol{p}} = \bar{p} + \epsilon\, \underline{p}^\circ. \tag{2.5}$$

Dual vector $\bar{\boldsymbol{p}}$ is also referred to the *Plücker coordinates* or a line. Unit vector $\bar{p} = \bar{\ell}$ provides the orientation of the line and vector $\underline{p}^\circ = \tilde{x}_P \bar{\ell}$. Note that $\|\bar{p}\| = 1$ and $\bar{p}^T \underline{p}^\circ = 0$, as expected for unit dual vectors. Note that $\underline{p}^\circ$ is not a unit vector. The point on the line that is at the shortest distance from the origin of the reference frame is $\underline{x}_S = \tilde{p}\underline{p}^\circ$.

Consider two lines, $\mathcal{L}_P = (\underline{x}_P, \bar{p})$ and $\mathcal{L}_Q = (\underline{x}_Q, \bar{q})$, and their dual unit vector representations, denoted $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$, respectively. The scalar product of two unit vectors defines the cosine of the angle, α, between the two vectors. Similarly, the scalar product of dual vectors, illustrated in fig. 2.1, is such that $\bar{\boldsymbol{p}}^T \bar{\boldsymbol{q}} = \bar{p}^T \bar{q} + \epsilon(\bar{p}^T \tilde{x}_Q \bar{q} + \bar{q}^T \tilde{x}_P \bar{p})$. The primal part of this dual scalar is $\bar{p}^T \bar{q} = \cos\alpha$, where α is the angle between the two lines and its dual part is $\bar{p}^T \tilde{x}_Q \bar{q} + \bar{q}^T \tilde{x}_P \bar{p} = -\alpha^\circ \sin\alpha$, where α° is the shortest distance between the two lines. Defining dual scalar $\boldsymbol{a}$ such that

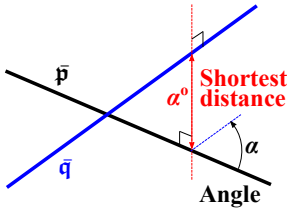

Figure 2.1: The shortest distance and angle between two lines

$$\boldsymbol{a} = \alpha + \epsilon\, \alpha^\circ, \tag{2.6}$$

then $\cos \boldsymbol{a}$ is an analytic dual function. In summary,

$$\bar{\boldsymbol{p}}^T \bar{\boldsymbol{q}} = \cos \boldsymbol{a}. \tag{2.7}$$

Similarly, if the scalar product of two unit dual vectors vanishes, i.e., if $\bar{\boldsymbol{p}}^T \bar{\boldsymbol{q}} = \boldsymbol{0}$, lines $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$ are mutually orthogonal, intersecting lines. The null dual scalar is

denoted $\mathbf{0} = 0 + \epsilon\ 0$.

Let $\underline{\boldsymbol{p}}$ and $\underline{\boldsymbol{q}}$ be two dual vectors; the vector product of two dual vectors as

$$\underline{u} = \tilde{\boldsymbol{p}}\underline{\boldsymbol{q}} = \tilde{p}\underline{q} + \epsilon\left(\tilde{p}\underline{q}^{\circ} + \tilde{p}^{\circ}\underline{q}\right), \tag{2.8}$$

The vector product of dual vectors is a dual vector.

The vector product of two unit vectors defines the sine of the angle between the two vectors, $\tilde{p}\bar{q} = \sin\alpha\,\bar{n}$, where $\bar{n}$ is the unit vector normal to vectors $\bar{p}$ and $\bar{q}$ and oriented according to the right hand rule. Similarly, the vector product of two unit dual vectors, illustrated in fig. 2.2, is

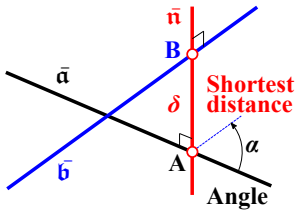

Figure 2.2: The vector product of two lines.

$$\tilde{\boldsymbol{p}}\bar{\boldsymbol{q}} = \sin\boldsymbol{a}\,\bar{\boldsymbol{n}} \tag{2.9}$$

where $\sin\boldsymbol{a}$ is an analytic dual function, and dual scalar $\boldsymbol{a}$ is defined by eq. (2.6). Because $\bar{\boldsymbol{n}}^T\bar{\boldsymbol{p}} = \bar{\boldsymbol{n}}^T\bar{\boldsymbol{q}} = \underline{0}$, line $\bar{\boldsymbol{n}}$ is normal unit vectors $\bar{p}$ and $\bar{q}$ and intersects lines $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$. This means that line $\bar{\boldsymbol{n}}$ joins the point of lines $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$ that are at the shortest distance from each other, denoted points P and Q in fig. 2.2.

2.3 Motion of a rigid body

A rigid-body motion is defined as the transformation that brings inertial frame $\mathcal{F} = [\mathbf{O}, \mathcal{I} = (\bar{\imath}_1, \bar{\imath}_2, \bar{\imath}_3)]$ to material frame $\mathcal{F}_b = [\mathbf{B}, \mathcal{B} = (\bar{b}_1, \bar{b}_2, \bar{b}_3)]$, as shown in figure 2.3. Rigid-body motion can be represented by motion tensor

$$\boldsymbol{R} = R + \epsilon\, \tilde{u} R, \tag{2.10}$$

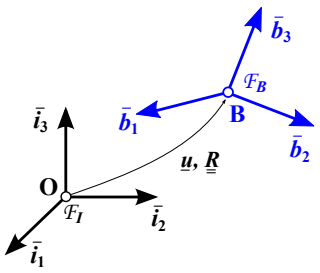

Figure 2.3: Two frames with a relative displacement, $\underline{u}$, and a relative rotation, R.

where rotation tensor R brings inertial basis $\mathcal{I}$ to material basis $\mathcal{B}$, and vector $\underline{u}$ is the relative position vector of reference point $\mathbf{B}$ with respect to the origin, $\mathbf{O}$. The motion tensor is an special orthogonal dual matrix: $\boldsymbol{R}^T\boldsymbol{R} = R^TR + 2\epsilon\, R^T(\tilde{u} - \tilde{u})R = \boldsymbol{I}$ and $\det(R) = 1$. The set of motion tensors form the **S**pecial **O**rthogonal group in $\mathbb{D}^{3\times3}$, denoted $\mathcal{SO}(3)$, which is isomorphic to the **S**pecial **E**uclidean group SE(3).

As shown in fig. 2.3, the three orthogonal lines passing through vector bases $\bar{\imath}_k$ and $\bar{b}_k$, $k = 1, 2, 3$, are denoted as

$$\begin{aligned} \bar{\boldsymbol{\imath}}_k &= \bar{\imath}_k + \epsilon\, \tilde{0}\bar{\imath}_k = \bar{\imath}_k + \epsilon\, \underline{0}, \\ \bar{\boldsymbol{b}}_k &= \bar{b}_k + \epsilon\, \tilde{u}\bar{b}_k, \end{aligned} \tag{2.11}$$

respectively. It follows that

$$\bar{\boldsymbol{b}}_i = \boldsymbol{R}\,\bar{\boldsymbol{\imath}}_i, \tag{2.12}$$

which indicates that the motion tensor transforms a line in inertial frame $\mathcal{F}$ to $\mathcal{F}_b$.

2.4 Rodrigues' formula

Chasles' theorem [115] states that the most general motion of a rigid body consists of a translation along a line followed by a rotation about the same line. Hence, a general motion is characterized by its Chasles' line of Plücker coordinates $\bar{\boldsymbol{n}} = \bar{n} + \epsilon\, \underline{n}^{\circ}$ and the magnitudes of the rotation and intrinsic displacement, denoted ϕ and ϕ°, respectively, for a total of six parameters. The scalar characteristics of the motion form a dual angle,

$$\boldsymbol{\phi} = \phi + \epsilon\, \phi^{\circ}. \tag{2.13}$$

In this section, the basic formulæ required for the manipulation of motion are expressed in terms of geometric entities $(\bar{\boldsymbol{n}}, \boldsymbol{\phi})$.

The Plücker coordinates of a material line of the body before and after it undergoes the specified motion are denoted $\bar{\boldsymbol{a}}$ and $\bar{\boldsymbol{b}}$, respectively, as shown in fig. 2.4. Vector product $\tilde{\boldsymbol{n}}\bar{\boldsymbol{a}} = \sin \boldsymbol{a}\ \bar{\boldsymbol{v}}$ defines line $\bar{\boldsymbol{v}}$ that is perpendicular to and intersects lines $\bar{\boldsymbol{n}}$ and $\bar{\boldsymbol{a}}$ at points **O** and **A**, respectively. Dual scalar $\boldsymbol{a} = \alpha + \epsilon\, \lambda$ defines the angle α between the lines and their shortest distance, λ, see fig 2.4. Next, vector product $\tilde{\boldsymbol{v}}\bar{\boldsymbol{n}} = \bar{\boldsymbol{u}}$ defines the last line of the canonical frame of the motion, $\mathcal{F} = (\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}, \bar{\boldsymbol{n}})$. Figure 2.4 shows a cylinder of radius λ and axis $\bar{\boldsymbol{n}}$ coincident with Chasles' line. Line $\bar{\boldsymbol{a}}$ is in the plane tangent to this cylinder at point **A**. During motion, line $\bar{\boldsymbol{a}}$

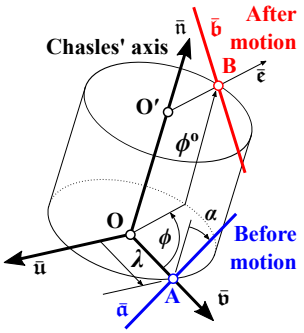

Figure 2.4: Material line of a body before and after motion

rotates around the cylinder by an angle ϕ and translates along line $\bar{\boldsymbol{n}}$ by a distance ϕ°. At the end of the motion, material point **A** has moved to point **B** and line $\bar{\boldsymbol{b}}$ is in the plane tangent to the cylinder at point **B**.

Because lines $\bar{\boldsymbol{n}}$ and $\bar{\boldsymbol{a}}$ are material lines of the body, their distance and relative orientation remain unchanged, i.e., $\bar{\boldsymbol{n}}^T\bar{\boldsymbol{a}} = \bar{\boldsymbol{n}}^T\bar{\boldsymbol{b}} = \cos \boldsymbol{a}$. For the same reasons, $\tilde{\boldsymbol{n}}\bar{\boldsymbol{b}} = \sin \boldsymbol{a}\ \bar{\boldsymbol{e}}$, where line $\bar{\boldsymbol{e}}$ is in the plane normal to $\bar{\boldsymbol{n}}$ at a distance ϕ° from point **O**, i.e., $\bar{\boldsymbol{e}} = \cos \boldsymbol{\phi}\bar{\boldsymbol{v}} - \sin \boldsymbol{\phi}\bar{\boldsymbol{u}}$. It follows that $\tilde{\boldsymbol{n}}\bar{\boldsymbol{b}} = \sin \boldsymbol{a}(\cos \boldsymbol{\phi}\bar{\boldsymbol{v}} - \sin \boldsymbol{\phi}\bar{\boldsymbol{u}})$, a vector-product equation whose solution is given as

$$\bar{\boldsymbol{b}} = \boldsymbol{\mu}\bar{\boldsymbol{n}} - \sin \boldsymbol{a}\ \tilde{\boldsymbol{n}}(\cos \boldsymbol{\phi}\ \bar{\boldsymbol{v}} - \sin \boldsymbol{\phi}\ \bar{\boldsymbol{u}}). \tag{2.14}$$

Dual scalar $\boldsymbol{\mu}$ is found to be $\boldsymbol{\mu} = \bar{\boldsymbol{n}}^T\bar{\boldsymbol{b}} = \bar{\boldsymbol{n}}^T\bar{\boldsymbol{a}}$, leading to $\bar{\boldsymbol{b}} = \bar{\boldsymbol{n}}\bar{\boldsymbol{n}}^T\bar{\boldsymbol{a}} + \sin \boldsymbol{\phi}\ \tilde{\boldsymbol{n}}\bar{\boldsymbol{a}} - \cos \boldsymbol{\phi}\ \tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}}\bar{\boldsymbol{a}}$. Finally, identity (A.1b) yields the desired result,

$$\bar{\boldsymbol{b}} = \boldsymbol{R}\bar{\boldsymbol{a}}, \tag{2.15}$$

where

$$\boldsymbol{R}(\bar{\boldsymbol{n}}, \boldsymbol{\phi}) = \boldsymbol{I} + \sin \boldsymbol{\phi}\ \tilde{\boldsymbol{n}} + (\boldsymbol{1} - \cos \boldsymbol{\phi})\tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}}, \tag{2.16}$$

is the motion tensor, which is fully defined by geometric entities $(\bar{\boldsymbol{n}}, \boldsymbol{\phi})$. The primal part of the motion tensor, R, relates the orientations of the lines $\bar{\boldsymbol{a}}$ and $\bar{\boldsymbol{b}}$ as $\bar{b} = R\,\bar{a}$, where $R = \exp(\phi\tilde{n}) = I + \sin\phi\tilde{n} + (1 - \cos\phi)\tilde{n}\tilde{n}$.

2.5 Dual unit quaternions

Rigid-body motion can also be represented by unit dual quaternions

$$\hat{\boldsymbol{q}} = \hat{q} + \epsilon\, \hat{q}^{\circ}, \tag{2.17}$$

where $\hat{q} = \{\eta, q_1, q_2, q_3\} = \{\eta, \underline{q}\}$ and $\hat{q}^{\circ} = \{\eta^{\circ}, q_1^{\circ}, q_2^{\circ}, q_3^{\circ}\} = \{\eta^{\circ}, \underline{q}^{\circ}\}$ are the primal and dual parts, respectively. Because $\hat{\boldsymbol{q}}^T\hat{\boldsymbol{q}} = \hat{q}^T\hat{q} + \epsilon\, 2\hat{q}^T\hat{q}^{\circ} = \mathbf{1}$, it follows that $\hat{q}$ is unit and $\hat{q}^{\circ}$ is orthogonal to $\hat{q}$. The following notation is introduced to define the dual scalar part, $\boldsymbol{\eta}$, and dual vector part, $\underline{\boldsymbol{q}}$, of the dual quaternion,

$$\boldsymbol{\eta} = \mathrm{scal}(\hat{\boldsymbol{q}}) = \eta + \epsilon\, \eta^{\circ}, \tag{2.18a}$$

$$\underline{\boldsymbol{q}} = \mathrm{vec}(\hat{\boldsymbol{q}}) = \underline{q} + \epsilon\, \underline{q}^{\circ}. \tag{2.18b}$$

Unit dual quaternions are related to Chasle's line $\bar{\boldsymbol{n}}$ and dual angle $\boldsymbol{\phi}$, such that dual vector $\underline{\boldsymbol{q}} = \mathrm{vec}(\hat{\boldsymbol{q}}) = \bar{\boldsymbol{n}} \sin \boldsymbol{\phi}/2$ and dual scalar $\boldsymbol{\eta} = \mathrm{scal}(\hat{\boldsymbol{q}}) = \cos \boldsymbol{\phi}/2$. The motion tensor can be expressed in terms of the unit dual quaternions,

$$\begin{aligned} \boldsymbol{R}(\hat{\boldsymbol{q}}) &= \boldsymbol{I} + 2\boldsymbol{\eta}\tilde{\boldsymbol{q}} + 2\tilde{\boldsymbol{q}}\tilde{\boldsymbol{q}} \\ &= (\boldsymbol{\eta}^2 - \underline{\boldsymbol{q}}^T\underline{\boldsymbol{q}})\boldsymbol{I} + 2\boldsymbol{\eta}\tilde{\boldsymbol{q}} + 2\underline{\boldsymbol{q}}\,\underline{\boldsymbol{q}}^T, \end{aligned} \tag{2.19}$$

where identity (A.1b) is introduced to yield the second equality.

Consider three unit dual quaternions $\hat{\boldsymbol{q}}$, $\hat{\boldsymbol{q}}_1$, and $\hat{\boldsymbol{q}}_2$ such that $\boldsymbol{R}(\hat{\boldsymbol{q}}) = \boldsymbol{R}(\hat{\boldsymbol{q}}_1)\boldsymbol{R}(\hat{\boldsymbol{q}}_2)$.

The composition rule for dual quaternions is

$$\hat{\boldsymbol{q}} = \boldsymbol{A}(\hat{\boldsymbol{q}}_1)\hat{\boldsymbol{q}}_2 = \boldsymbol{B}(\hat{\boldsymbol{q}}_2)\hat{\boldsymbol{q}}_1, \tag{2.20}$$

where matrix operators $\boldsymbol{A}$ and $\boldsymbol{B}$ are defined as

$$\boldsymbol{A}(\underline{\boldsymbol{q}}) = \begin{bmatrix} \eta & \underline{\boldsymbol{q}}^T \\ -\underline{\boldsymbol{q}} & \eta I + \tilde{\boldsymbol{q}} \end{bmatrix}, \quad \boldsymbol{B}(\underline{\boldsymbol{q}}) = \begin{bmatrix} \eta & \underline{\boldsymbol{q}}^T \\ -\underline{\boldsymbol{q}} & \eta I - \tilde{\boldsymbol{q}} \end{bmatrix}. \tag{2.21}$$

It is verified easily that

$$\boldsymbol{A}^T(\hat{\boldsymbol{q}})\boldsymbol{A}(\hat{\boldsymbol{q}}) = \boldsymbol{B}^T(\hat{\boldsymbol{q}})\boldsymbol{B}(\hat{\boldsymbol{q}}) = \boldsymbol{I}, \tag{2.22a}$$

$$\boldsymbol{A}(\hat{\boldsymbol{q}})\boldsymbol{B}^T(\hat{\boldsymbol{q}}) = \boldsymbol{B}^T(\hat{\boldsymbol{q}})\boldsymbol{A}(\hat{\boldsymbol{q}}) = \mathrm{diag}[1, \boldsymbol{R}(\hat{\boldsymbol{q}})]. \tag{2.22b}$$

Example 2.5.1. ***Trace of a product of matrices***

Consider the matrix product $\boldsymbol{R}^T\boldsymbol{G}$*, where* $\boldsymbol{R}$ *is an orthogonal dual matrix and* $\boldsymbol{G}$ *an arbitrary dual matrix. In view of identity* (2.19)*, the trace of dual matrix product is then* $\mathrm{tr}(\boldsymbol{R}^T\boldsymbol{G}) = \mathrm{tr}[(\boldsymbol{\eta}^2 - \underline{\boldsymbol{q}}^T\underline{\boldsymbol{q}})\boldsymbol{G} - 2\eta\tilde{\boldsymbol{q}}\boldsymbol{G} + 2\underline{\boldsymbol{q}}\,\underline{\boldsymbol{q}}^T\boldsymbol{G}]$*. Trace identities* (A.3e) *and* (A.3f) *imply* $\mathrm{tr}(\tilde{\boldsymbol{q}}\boldsymbol{G}) = 2\underline{\boldsymbol{q}}^T\mathrm{axial}(\boldsymbol{G})$ *and* $\mathrm{tr}(\underline{\boldsymbol{q}}\,\underline{\boldsymbol{q}}^T\boldsymbol{G}) = \underline{\boldsymbol{q}}^T\mathrm{symm}(\boldsymbol{G})\,\underline{\boldsymbol{q}}$*, respectively, leading to* $\mathrm{tr}(\boldsymbol{R}^T\boldsymbol{G}) = (\boldsymbol{\eta}^2 - \underline{\boldsymbol{q}}^T\underline{\boldsymbol{q}})\mathrm{tr}(\boldsymbol{G}) + 4\eta\underline{\boldsymbol{q}}^T\mathrm{axial}(\boldsymbol{G}) + 2\underline{\boldsymbol{q}}^T\mathrm{symm}(\boldsymbol{G})\,\underline{\boldsymbol{q}}$ *and finally,*

$$\mathbf{1} + \mathrm{tr}(\boldsymbol{R}^T\boldsymbol{G}) = \mathbf{1} + \mathrm{tr}(\boldsymbol{G}^T\boldsymbol{R}) = \hat{\boldsymbol{q}}^T\boldsymbol{W}(\boldsymbol{G})\,\hat{\boldsymbol{q}}, \tag{2.23}$$

where symmetric matrix $\boldsymbol{W}(\boldsymbol{G})$ *is*

$$\boldsymbol{W}(\boldsymbol{G}) = \begin{bmatrix} \mathbf{1} + \mathrm{tr}(\boldsymbol{G}) & 2\mathrm{axial}^T(\boldsymbol{G}) \\ 2\mathrm{axial}(\boldsymbol{G}) & [1 - \mathrm{tr}(\boldsymbol{G})]\boldsymbol{I} + 2\mathrm{symm}(\boldsymbol{G}) \end{bmatrix}. \tag{2.24}$$

If matrix $\boldsymbol{G}$ *is itself orthogonal, i.e., if it represents a motion, matrix* $\boldsymbol{W}$ *becomes*

$$\boldsymbol{W}(\boldsymbol{G}) = 4\hat{\boldsymbol{g}}\,\hat{\boldsymbol{g}}^T, \tag{2.25}$$

where unit dual quaternion $\hat{\boldsymbol{g}}$ *represents motion tensor* $\boldsymbol{G}$*. Finally, if* $\boldsymbol{G} = \boldsymbol{R}$*, eq. (2.23) becomes* $\mathbf{1} + \mathrm{tr}(\boldsymbol{R}^T\boldsymbol{R}) = \hat{\boldsymbol{q}}^T[4\hat{\boldsymbol{q}}\,\hat{\boldsymbol{q}}^T]\hat{\boldsymbol{q}} = 4(\hat{\boldsymbol{q}}^T\hat{\boldsymbol{q}})^2 = 4$*; because* $\mathrm{tr}(\boldsymbol{R}^T\boldsymbol{R}) = \mathrm{tr}(\boldsymbol{I}) = 3$*, this relationship is satisfied.*

Example 2.5.2. ***Transformation of matrix W***

Let $\hat{\boldsymbol{q}}_1$ *and* $\hat{\boldsymbol{q}}_2$ *be two unit dual quaternions representing two motions. Composition rule (2.20) now implies that the unit dual quaternions representing composed motion* $\boldsymbol{R}(\hat{\boldsymbol{q}}) = \boldsymbol{R}_1\boldsymbol{R}_2$ *is* $\hat{\boldsymbol{q}} = \boldsymbol{A}(\hat{\boldsymbol{q}}_1)\hat{\boldsymbol{q}}_2 = \boldsymbol{B}(\hat{\boldsymbol{q}}_2)\hat{\boldsymbol{q}}_1$*. Equation (2.23) yields* $\mathbf{1} + \mathrm{tr}[\boldsymbol{R}^T\boldsymbol{G})] = \mathbf{1} + \mathrm{tr}[(\boldsymbol{R}_1\boldsymbol{R}_2)^T\boldsymbol{G})] = \hat{\boldsymbol{q}}^T\boldsymbol{W}(\boldsymbol{G})\hat{\boldsymbol{q}} = \hat{\boldsymbol{q}}_2^T[\boldsymbol{A}^T(\hat{\boldsymbol{q}}_1)\boldsymbol{W}(\boldsymbol{G})\boldsymbol{A}(\hat{\boldsymbol{q}}_1)]\hat{\boldsymbol{q}}_2$*. The same operation can be recast as* $\mathbf{1} + \mathrm{tr}[R_2^T(R_1^T\boldsymbol{G})] = \hat{e}_2^T\boldsymbol{W}(\boldsymbol{R}_1^T\boldsymbol{G})\hat{\boldsymbol{q}}_2$ *and because quaternion* $\hat{e}_2$ *can be selected arbitrarily, the following identity results,*

$$\boldsymbol{W}(\boldsymbol{R}_1^T\boldsymbol{G}) = \boldsymbol{A}^T(\hat{\boldsymbol{q}}_1)\boldsymbol{W}(\boldsymbol{G})\boldsymbol{A}(\hat{\boldsymbol{q}}_1). \tag{2.26}$$

Equation (2.20) expresses the motion tensor in terms of unit dual quaternions. In many applications, the inverse operation is also required, i.e., given the motion

tensor, find the unit dual quaternions. For an orthogonal tensor, the symmetric matrix defined be eq. (2.25) becomes

$$\boldsymbol{W} = 4\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}^T = 4\begin{bmatrix} \boldsymbol{q}_0^2 & \boldsymbol{q}_0\boldsymbol{q}_1 & \boldsymbol{q}_0\boldsymbol{q}_2 & \boldsymbol{q}_0\boldsymbol{q}_3 \\ \boldsymbol{q}_0\boldsymbol{q}_1 & \boldsymbol{q}_1^2 & \boldsymbol{q}_1\boldsymbol{q}_2 & \boldsymbol{q}_1\boldsymbol{q}_3 \\ \boldsymbol{q}_0\boldsymbol{q}_2 & \boldsymbol{q}_1\boldsymbol{q}_2 & \boldsymbol{q}_2^2 & \boldsymbol{q}_2\boldsymbol{q}_3 \\ \boldsymbol{q}_0\boldsymbol{q}_3 & \boldsymbol{q}_1\boldsymbol{q}_3 & \boldsymbol{q}_2\boldsymbol{q}_3 & \boldsymbol{q}_3^2 \end{bmatrix}, \tag{2.27}$$

where $\boldsymbol{q}_0 = \boldsymbol{\eta}$. According to Klumpp [116] and Shepperd [117], the most accurate results will be obtained by extracting unit dual quaternions from the column of $\boldsymbol{W}$ which presents the largest diagonal term for the primal part. It can be readily shown that

$$\max\left(W_{00}, W_{11}, W_{22}, W_{33}\right) = \max\left(\mathrm{tr}(R), R_{11}, R_{22}, R_{33}\right). \tag{2.28}$$

If m is the index corresponding to the column with the maximum diagonal term, the desired Euler motion parameters then follow

$$\boldsymbol{q}_i = \frac{\boldsymbol{T}_{im}}{2\sqrt{\boldsymbol{T}_{mm}}}, \quad i = 0, 1, 2, 3. \tag{2.29}$$

2.6 Vectorial parameterization of motion

The vectorial parameterization of motion [118] is more general and defined as

$$\underline{\boldsymbol{p}} = \boldsymbol{p}(\boldsymbol{\phi})\bar{\boldsymbol{n}}, \tag{2.30}$$

where $\boldsymbol{p}(\boldsymbol{\phi})$ is a dual function of dual scalar $\boldsymbol{\phi}$, called the "generating function," for short. The primal part of the generating function is the generating function for the vectorial parameterization of rotation, $p(\phi)$. Because the generating dual function is selected to be analytic, see eq. (2.4), the generating dual function is of the form $\boldsymbol{p} = p(\phi) + \epsilon\ \phi^{\circ} p'(\phi)$, where notation $(\cdot)'$ indicates a derivative with respect to ϕ. A more explicit expression of the motion parameter vector becomes

$$\underline{\boldsymbol{p}} = \underline{p} + \epsilon\ \underline{p}^{\circ} = p(\phi)\bar{n} + \epsilon\ \left[\phi^{\circ} p'(\phi)\bar{n} + p(\phi)\underline{n}^{\circ}\right]. \tag{2.31}$$

Clearly, vector $\underline{\boldsymbol{q}}$ gathers all the information about the motion: Chasles' line, $\bar{\boldsymbol{n}}$, and dual scalar $\boldsymbol{\phi}$.

Two dual scalar functions play an important role in the vectorial parameterization of motion,

$$\boldsymbol{\nu} = \frac{\sin \boldsymbol{\phi}/2}{\boldsymbol{p}}, \tag{2.32a}$$

$$\boldsymbol{\varepsilon} = \frac{2\tan \boldsymbol{\phi}/2}{\boldsymbol{p}} = \frac{\boldsymbol{\nu}}{\boldsymbol{\eta}}. \tag{2.32b}$$

Equation (2.16) implies

$$\boldsymbol{R}\underline{\boldsymbol{p}} = \underline{\boldsymbol{p}}, \tag{2.33}$$

i.e., motion parameter vector $\underline{\boldsymbol{p}}$ is an eigenvector of the motion tensor associated with its unit eigenvalue.

Introducing the vectorial parameterization of motion into eq. (2.16) yields the

expression for the motion tensor,

$$\boldsymbol{R} = \boldsymbol{I} + \boldsymbol{\zeta}_1 \tilde{\boldsymbol{p}} + \boldsymbol{\zeta}_2 \tilde{\boldsymbol{p}} \tilde{\boldsymbol{p}}, \tag{2.34}$$

where

$$\boldsymbol{\zeta}_1(\boldsymbol{\phi}) = \frac{\sin \boldsymbol{\phi}}{\boldsymbol{p}} = \frac{\boldsymbol{\nu}^2}{\boldsymbol{\varepsilon}}, \quad \boldsymbol{\zeta}_2(\boldsymbol{\phi}) = \frac{\mathbf{1} - \cos \boldsymbol{\phi}}{\boldsymbol{p}^2} = \frac{\boldsymbol{\nu}^2}{2}. \tag{2.35}$$

Three specific vectorial parameterizations of motion are presented: the Cartesian, linear, and Euler-Rodrigues motion parameters. The naming of the various parameterizations presented here is mnemonic. The generating function and coefficients for these three parameterizations are listed in tab. 2.1.

	$\boldsymbol{p}(\boldsymbol{\phi})$	$\boldsymbol{\zeta}_1$	$\boldsymbol{\zeta}_2$	$\boldsymbol{\chi}_0$	$\boldsymbol{\chi}_1$
Cartesian	$\boldsymbol{\phi}$	$(\sin \boldsymbol{\phi})/\boldsymbol{\phi}$	$(\mathbf{1} - \cos \boldsymbol{\phi})/\boldsymbol{\phi}^2$	$\mathbf{1}$	$(\mathbf{1} - \mathbf{1}/\boldsymbol{\varepsilon})/\boldsymbol{\phi}^2$
Linear	$\sin(\boldsymbol{\phi})$	$\mathbf{1}$	$\boldsymbol{\varepsilon}/2$	$(2 - \boldsymbol{\varepsilon})/\boldsymbol{\varepsilon}$	$-\boldsymbol{\varepsilon}/4$
Euler-Rodrigues	$\sin(\boldsymbol{\phi}/2)$	$\boldsymbol{\eta}$	$\mathbf{1}/2$	$\boldsymbol{\eta}$	$\boldsymbol{\chi}_2$

Table 2.1: Coefficients of the vectorial parameterization.

For the Cartesian parameter vector, expanding the trigonometric functions in eq. (2.35) with infinite series yields

$$\boldsymbol{R}(\underline{\boldsymbol{p}}) = \sum_{k=0}^{\infty} \tilde{\boldsymbol{p}}^k = \exp(\tilde{\boldsymbol{p}}), \tag{2.36}$$

which is the exponential map. The inverse operation, logarithmic map, is defined as $\tilde{\boldsymbol{p}} = \log(\boldsymbol{R}) = \sum_{k=1}^{\infty} (-1)^{k-1}/k (\boldsymbol{R} - \boldsymbol{I})^k$.

2.7 Velocities, curvatures, variations, and increments of motion

Let $\boldsymbol{R}(t) = R(t) + \epsilon\, \tilde{u}(t)R(t)$ be the time-dependent motion tensor that brings inertial frame $\mathcal{F}_I = (\bar{\boldsymbol{i}}_1, \bar{\boldsymbol{i}}_2, \bar{\boldsymbol{i}}_3)$ to material frame $\mathcal{F}(t) = (\bar{\boldsymbol{b}}_1, \bar{\boldsymbol{b}}_2, \bar{\boldsymbol{b}}_3)$. Because the motion tensor is an orthogonal dual matrix, a time derivative yields $(\boldsymbol{R}^T\dot{\boldsymbol{R}})^T + \boldsymbol{R}^T\dot{\boldsymbol{R}} = \boldsymbol{0}$, which shows that the dual matrix in the parentheses must be a skew-symmetric dual matrix

$$\tilde{\boldsymbol{v}} = \boldsymbol{R}^T\dot{\boldsymbol{R}} = -\dot{\boldsymbol{R}}^T\boldsymbol{R} = \tilde{\omega} + \epsilon\, \widetilde{R^T\underline{\dot{u}}}, \tag{2.37}$$

where $\tilde{\omega} = R^T\dot{R}$. Dual vector $\underline{\boldsymbol{v}}$ stores the components of the velocity vector resolved in the material frame

$$\underline{\boldsymbol{v}} = \tilde{\omega} + \epsilon\, R^T\underline{\dot{u}}, \tag{2.38}$$

The primal part of the dual velocity vector is the angular velocity vector, while its dual part is the linear velocity of the rigid body. This quantity can be interpreted as the linear velocity of the point of the rigid body that instantaneously coincides with the origin of the reference frame, point **O**.

Similarly, let $\boldsymbol{R}(s)$ be the space-dependent motion tensor that brings inertial frame $\mathcal{F}_I = (\bar{\boldsymbol{i}}_1, \bar{\boldsymbol{i}}_2, \bar{\boldsymbol{i}}_3)$ to frame $\mathcal{F}(s) = (\bar{\boldsymbol{b}}_1, \bar{\boldsymbol{b}}_2, \bar{\boldsymbol{b}}_3)$. A spatial derivative of the motion tensor

$$\tilde{\boldsymbol{k}} = \boldsymbol{R}^T\boldsymbol{R}' = -(\boldsymbol{R}^T)'\boldsymbol{R} = \tilde{\kappa} + \epsilon\, \widetilde{R^T\underline{u}'}. \tag{2.39}$$

where $\tilde{k} = R^TR'$. Dual vector $\underline{\boldsymbol{k}}$ stores the components of the curvature vector

resolved in the material frame.

$$\underline{\boldsymbol{k}} = \underline{\kappa} + \epsilon\ R^T \underline{u}', \tag{2.40}$$

The primal part of the dual curvature vector is the curvature vector.

The velocity and curvature vectors can be expressed in terms of geometric entities $(\bar{\boldsymbol{n}}, \boldsymbol{\phi})$ and their time or spatial derivatives. Introducing eq. (2.16) into eqs. (2.37) and (2.39) leads to

$$\underline{\boldsymbol{v}} = \dot{\boldsymbol{\phi}}\ \bar{\boldsymbol{n}} + \sin\boldsymbol{\phi}\ \dot{\bar{\boldsymbol{n}}} - (\mathbf{1} - \cos\boldsymbol{\phi})\tilde{\boldsymbol{n}}\dot{\bar{\boldsymbol{n}}}, \tag{2.41a}$$

$$\underline{\boldsymbol{k}} = \boldsymbol{\phi}'\ \bar{\boldsymbol{n}} + \sin\boldsymbol{\phi}\ \bar{\boldsymbol{n}}' - (\mathbf{1} - \cos\boldsymbol{\phi})\tilde{\boldsymbol{n}}\bar{\boldsymbol{n}}', \tag{2.41b}$$

The definition of the *virtual* and *incremental motion vector*, $\underline{\delta\boldsymbol{u}}$, is analogous to that of the differential motion vector,

$$\tilde{\delta\boldsymbol{u}} = \boldsymbol{R}^T\delta\boldsymbol{R} = -\delta\boldsymbol{R}^T\boldsymbol{R} = \tilde{\delta\psi} + \epsilon\ R^T\delta\underline{u}, \tag{2.42a}$$

$$\tilde{\Delta\boldsymbol{u}} = \boldsymbol{R}^T\Delta\boldsymbol{R} = -\Delta\boldsymbol{R}^T\boldsymbol{R} = \tilde{\Delta\psi} + \epsilon\ R^T\Delta\underline{u}, \tag{2.42b}$$

where $\tilde{\delta\psi} = R^T\delta R$ and $\tilde{\Delta\psi} = R^T\Delta R$. Explicit expressions for the components of the virtual and incremental motion vector are

$$\underline{\delta\boldsymbol{u}} = \underline{\delta\psi} + \epsilon\ R^T\delta\underline{u}, \tag{2.43a}$$

$$\underline{\Delta\boldsymbol{u}} = \underline{\Delta\psi} + \epsilon\ R^T\Delta\underline{u}. \tag{2.43b}$$

2.8 Commutativity of second order derivatives

Consider the the velocity and curvatures vectors resolved in the material frame as defined in eqs. (2.37) and (2.39). Taking a spatial of eq. (2.37) and a time derivative of eq. (2.39) leads to

$$(\tilde{\boldsymbol{v}})' = \boldsymbol{R}'^T\dot{\boldsymbol{R}} + \boldsymbol{R}^T(\dot{\boldsymbol{R}})', \tag{2.44a}$$

$$(\tilde{\boldsymbol{k}})^{\cdot} = \dot{\boldsymbol{R}}^T\boldsymbol{R}' + \boldsymbol{R}^T(\boldsymbol{R}')^{\cdot}. \tag{2.44b}$$

The second derivatives of the motion tensor must commute, i.e., $(\dot{\boldsymbol{R}})' = (\boldsymbol{R}')^{\cdot}$, because it is a continuous function of both temporal and spatial variables. Subtracting eq. (2.44b) from eq. (2.44a) then yields $(\tilde{\boldsymbol{v}})' - (\tilde{\boldsymbol{k}})^{\cdot} = \boldsymbol{R}'^T\dot{\boldsymbol{R}} - \dot{\boldsymbol{R}}^T\boldsymbol{R}' = (\boldsymbol{R}'^T\boldsymbol{R})(\boldsymbol{R}^T\dot{\boldsymbol{R}}) - (\dot{\boldsymbol{R}}^T\boldsymbol{R})(\boldsymbol{R}^T\boldsymbol{R}') = \tilde{\boldsymbol{k}}^T\tilde{\boldsymbol{v}} - \tilde{\boldsymbol{v}}^T\tilde{\boldsymbol{k}} = \tilde{\boldsymbol{v}}\tilde{\boldsymbol{k}} - \tilde{\boldsymbol{k}}\tilde{\boldsymbol{v}}$. Finally, introducing identity $\tilde{\boldsymbol{v}}\tilde{\boldsymbol{k}} - \tilde{\boldsymbol{k}}\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}\underline{\boldsymbol{k}}$ leads to $\underline{\boldsymbol{v}}' - \dot{\underline{\boldsymbol{k}}} = \tilde{\boldsymbol{v}}\underline{\boldsymbol{k}}$.

The following results are obtained in a similar manner

$$\underline{\boldsymbol{v}}' = \dot{\underline{\boldsymbol{k}}} + \tilde{\boldsymbol{v}}\underline{\boldsymbol{k}}, \tag{2.45a}$$

$$\delta\underline{\boldsymbol{v}} = \dot{\underline{\delta\boldsymbol{u}}} + \tilde{\boldsymbol{v}}\underline{\delta\boldsymbol{u}}, \tag{2.45b}$$

$$\delta\underline{\boldsymbol{k}} = \underline{\delta\boldsymbol{u}}' + \tilde{\boldsymbol{k}}\underline{\delta\boldsymbol{u}}. \tag{2.45c}$$

Equation (2.45a) are known as the *compatibility equations*. Equations (2.45b) and (2.45c) are known as the *transpositional relationships* [119–121]. Although the transpositional relationships and compatibility equations bear different names, they are all

consequences of the commutativity of the second derivatives of the motion tensor.

2.9 Tangent tensor and identities of composition

Taking a time derivative of the matrix operator $\boldsymbol{A}(\hat{\boldsymbol{q}})$ and left-multiplying by $\boldsymbol{A}^T(\hat{\boldsymbol{q}})$ yield

$$\begin{aligned}\boldsymbol{A}^T(\hat{\boldsymbol{q}})\boldsymbol{A}(\dot{\hat{\boldsymbol{q}}}) &= \begin{bmatrix} \eta & \underline{\boldsymbol{q}}^T \\ -\underline{\boldsymbol{q}} & \eta\boldsymbol{I} - \tilde{\boldsymbol{q}} \end{bmatrix} \begin{bmatrix} \dot{\eta} & -\dot{\underline{\boldsymbol{q}}}^T \\ \dot{\underline{\boldsymbol{q}}} & \dot{\eta}I + \dot{\tilde{\boldsymbol{q}}} \end{bmatrix} \\ &= \begin{bmatrix} \hat{\boldsymbol{q}}^T\dot{\hat{\boldsymbol{q}}} & \dot{\eta}\underline{\boldsymbol{q}}^T - \eta\dot{\underline{\boldsymbol{q}}}^T + \underline{\boldsymbol{q}}^T\dot{\tilde{\boldsymbol{q}}} \\ \eta\dot{\underline{\boldsymbol{q}}} - \dot{\eta}\underline{\boldsymbol{q}} - \tilde{\boldsymbol{q}}\dot{\underline{\boldsymbol{q}}} & \underline{\boldsymbol{q}}\,\dot{\underline{\boldsymbol{q}}}^T + (\eta\boldsymbol{I} - \tilde{\boldsymbol{q}})(\dot{\eta}\boldsymbol{I} + \dot{\tilde{\boldsymbol{q}}}) \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} \boldsymbol{0} & -\underline{\boldsymbol{v}}^T \\ \underline{\boldsymbol{v}} & \tilde{\boldsymbol{v}} \end{bmatrix} = \frac{1}{2}\boldsymbol{A}(\hat{\boldsymbol{v}}),\end{aligned} \tag{2.46}$$

where $1/2\tilde{\boldsymbol{v}} = 1/2\boldsymbol{R}^T\dot{\boldsymbol{R}} = \underline{\boldsymbol{q}}\,\dot{\underline{\boldsymbol{q}}}^T + (\eta\boldsymbol{I} - \tilde{\boldsymbol{q}})(\dot{\eta}\boldsymbol{I} + \dot{\tilde{\boldsymbol{q}}})$ by using the second identity in eq. (2.19), and $1/2\underline{\boldsymbol{v}} = \eta\dot{\underline{\boldsymbol{q}}} - \dot{\eta}\underline{\boldsymbol{q}} - \tilde{\boldsymbol{q}}\dot{\underline{\boldsymbol{q}}}$ by using identity (A.1a), (A.1b), and $\eta^2 + \underline{\boldsymbol{q}}^T\underline{\boldsymbol{q}} = \boldsymbol{1}$. The augmented velocity vector is defined as $\hat{\boldsymbol{v}} = \{\boldsymbol{0}, \underline{\boldsymbol{v}}^T\}^T$. Identity $1/2\underline{\boldsymbol{v}} = \eta\dot{\underline{\boldsymbol{q}}} - \dot{\eta}\underline{\boldsymbol{q}} - \tilde{\boldsymbol{q}}\dot{\underline{\boldsymbol{q}}}$ is rewritten to

$$\dot{\hat{\boldsymbol{q}}} = \frac{1}{2}\boldsymbol{A}(\hat{\boldsymbol{q}})\hat{\boldsymbol{v}} = \frac{1}{2}\boldsymbol{B}(\hat{\boldsymbol{v}})\hat{\boldsymbol{q}}. \tag{2.47}$$

Taking a time derivative of the motion parameter vector yields $\dot{\underline{\boldsymbol{p}}} = \boldsymbol{p}'\dot{\boldsymbol{\phi}}\bar{\boldsymbol{n}} + \boldsymbol{p}\,\dot{\bar{\boldsymbol{n}}}$. Identity (A.1a) leads to $\tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}}\dot{\underline{\boldsymbol{p}}} = \boldsymbol{p}\tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}}\dot{\bar{\boldsymbol{n}}} = \boldsymbol{p}[\bar{\boldsymbol{n}}\,\bar{\boldsymbol{n}}^T - \boldsymbol{I}]\dot{\bar{\boldsymbol{n}}} = -\boldsymbol{p}\,\dot{\bar{\boldsymbol{n}}} = \boldsymbol{p}'\dot{\boldsymbol{\phi}}\bar{\boldsymbol{n}} - \dot{\underline{\boldsymbol{p}}}$, because $\bar{\boldsymbol{n}}$ is a unit vector. It follows that $\dot{\boldsymbol{\phi}}\bar{\boldsymbol{n}} = (\boldsymbol{I} + \tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}})\dot{\underline{\boldsymbol{p}}}/\boldsymbol{p}'$ and introducing these results

into eqs. (2.41a) then leads to

$$\dot{\underline{\boldsymbol{p}}} = \boldsymbol{T}^{-1}(\underline{\boldsymbol{p}})\underline{\boldsymbol{v}}, \tag{2.48}$$

where the inverse of *tangent tensor* is found as

$$\boldsymbol{T}^{-1}(\underline{\boldsymbol{p}}) = \boldsymbol{\chi}_0\boldsymbol{I} + \frac{1}{2}\tilde{\boldsymbol{p}} + \boldsymbol{\chi}_2\tilde{\boldsymbol{p}}\tilde{\boldsymbol{p}}. \tag{2.49}$$

Two scalar functions were defined,

$$\boldsymbol{\chi}_0 = \boldsymbol{p}', \tag{2.50a}$$

$$\boldsymbol{\chi}_2 = \frac{\mathbf{1}}{\boldsymbol{p}^2}\left(\boldsymbol{p}' - \frac{\mathbf{1}}{\boldsymbol{\varepsilon}}\right). \tag{2.50b}$$

Tangent tensor $\boldsymbol{T}$ enjoy the following remarkable propertie,

$$\boldsymbol{R}(\underline{\boldsymbol{p}}) = \boldsymbol{T}(-\underline{\boldsymbol{p}})\,\boldsymbol{T}^{-1}(\underline{\boldsymbol{p}}) = \boldsymbol{T}^{-1}(\underline{\boldsymbol{p}})\boldsymbol{T}(-\underline{\boldsymbol{p}}). \tag{2.51}$$

Consider three motion tensors $\boldsymbol{R}_1$, $\boldsymbol{R}_2$ and $\boldsymbol{R}$, such that $\boldsymbol{R} = \boldsymbol{R}_1^T\boldsymbol{R}_2$, i.e., $\boldsymbol{R}$ are the relative motion of motion $\boldsymbol{R}_2$ with respect to motion $\boldsymbol{R}_1$. A time derivative of this relationship yields $\boldsymbol{R}^T\dot{\boldsymbol{R}} = \boldsymbol{R}^T\dot{\boldsymbol{R}}_1^T\boldsymbol{R}_2 + \boldsymbol{R}^T\boldsymbol{R}_1^T\dot{\boldsymbol{R}}_2$ and hence, $\underline{\boldsymbol{v}} = -\boldsymbol{R}^T\underline{\boldsymbol{v}}_1 + \underline{\boldsymbol{v}}_2$, where $\tilde{\boldsymbol{v}}_1 = \boldsymbol{R}_1^T\dot{\boldsymbol{R}}_2$, and $\tilde{\boldsymbol{v}}_2 = \boldsymbol{R}_2^T\dot{\boldsymbol{R}}_2$. Suppose that $\hat{\boldsymbol{q}}_1$, $\hat{\boldsymbol{q}}_2$, and $\hat{\boldsymbol{q}}_3$ are the unit dual quaternions associated with motion tensor $\boldsymbol{R}_1$, $\boldsymbol{R}_2$, and $\boldsymbol{R}$, respectively. Introducing eq. (2.47) into yields $\boldsymbol{A}^T(\hat{\boldsymbol{q}})\dot{\hat{\boldsymbol{q}}} = -\boldsymbol{R}^T\hat{\boldsymbol{v}}_1 + \hat{\boldsymbol{v}}_2$ and finally, identities (2.22) leads to

$$\dot{\hat{\boldsymbol{q}}} = -\boldsymbol{B}(\hat{\boldsymbol{q}})\hat{\boldsymbol{v}}_1 + \boldsymbol{A}(\hat{\boldsymbol{q}})\hat{\boldsymbol{v}}_2. \tag{2.52}$$

Similarly, the following identities hold for variations and increments

$$\delta\hat{\boldsymbol{q}} = -\boldsymbol{B}(\hat{\boldsymbol{q}})\delta\hat{\boldsymbol{u}}_1 + \boldsymbol{A}(\hat{\boldsymbol{q}})\delta\hat{\boldsymbol{u}}_2, \tag{2.53a}$$

$$\Delta\hat{\boldsymbol{q}} = -\boldsymbol{B}(\hat{\boldsymbol{q}})\delta\hat{\boldsymbol{u}}_1 + \boldsymbol{A}(\hat{\boldsymbol{q}})\delta\hat{\boldsymbol{u}}_2, \tag{2.53b}$$

where $\delta\hat{\boldsymbol{u}}_k = \{\boldsymbol{0}, \delta\underline{\boldsymbol{u}}_k^T\}^T$ and $\delta\hat{\boldsymbol{u}}_k = \{\boldsymbol{0}, \underline{\Delta\boldsymbol{u}}_k^T\}^T$ for $k = 1, 2$.

Suppose that $\underline{\boldsymbol{p}}_1$, $\underline{\boldsymbol{p}}_2$, and $\underline{\boldsymbol{p}}_3$ are the vector parameters associated with motion tensor $\boldsymbol{R}_1$, $\boldsymbol{R}_2$, and $\boldsymbol{R}$, respectively. Introducing eq. (2.48) yields $\boldsymbol{T}(\underline{\boldsymbol{p}})\dot{\underline{\boldsymbol{p}}} = -\boldsymbol{R}^T\boldsymbol{T}(\underline{\boldsymbol{p}}_1)\dot{\underline{\boldsymbol{p}}}_1 + \boldsymbol{T}(\underline{\boldsymbol{p}}_2)\dot{\underline{\boldsymbol{p}}}_2$ and finally, identity (2.51) leads to

$$\dot{\underline{\boldsymbol{p}}} = -\boldsymbol{T}^{-1}(-\underline{\boldsymbol{p}})\underline{\boldsymbol{v}}_1 + \boldsymbol{T}^{-1}(\underline{\boldsymbol{p}})\underline{\boldsymbol{v}}_2. \tag{2.54}$$

Similarly, the following identities hold for variations and increments

$$\delta\underline{\boldsymbol{p}} = -\boldsymbol{T}^{-1}(-\underline{\boldsymbol{p}})\underline{\delta\boldsymbol{u}}_1 + \boldsymbol{T}^{-1}(\underline{\boldsymbol{p}})\underline{\delta\boldsymbol{u}}_2, \tag{2.55a}$$

$$\Delta\underline{\boldsymbol{p}} = -\boldsymbol{T}^{-1}(-\underline{\boldsymbol{p}})\underline{\Delta\boldsymbol{u}}_1 + \boldsymbol{T}^{-1}(\underline{\boldsymbol{p}})\underline{\Delta\boldsymbol{u}}_2. \tag{2.55b}$$

The Euler-Rodrigues parameter vector is the vector part of the unit dual quaternion, i.e., $\underline{\boldsymbol{p}} = \underline{\boldsymbol{q}} = \sin(\phi/2)\bar{\boldsymbol{n}} = \mathrm{vec}(\hat{\boldsymbol{q}})$. Accordingly, the inverse of tangent tensor of Euler-Rodrigues parameter vector is the lower-left 3×3 submatrix of matrix operator $\boldsymbol{A}(\hat{\boldsymbol{q}})$, denoted as

$$\boldsymbol{T}^{-1}(\underline{\boldsymbol{p}}) = \boldsymbol{T}^{-1}(\underline{\boldsymbol{q}}) = \mathrm{vec}[\boldsymbol{A}(\hat{\boldsymbol{q}})] = \eta\boldsymbol{I} + \tilde{\boldsymbol{q}}. \tag{2.56}$$

2.10 Tangent space, Riemann metric, and the extended notation

The tangent space at an element $\boldsymbol{R}$ on the manifold $\mathcal{SO}(3)$, is $\mathrm{T}_{\boldsymbol{R}}\mathcal{SO}(3) = \{\boldsymbol{R}\,\tilde{\boldsymbol{s}} | \tilde{\boldsymbol{s}} = -\tilde{\boldsymbol{s}}^T\}$. Clearly, the velocities, curvatures, variations, and increments of rigid-body motions live in the tangent space at identity $\boldsymbol{I}$. In a neighborhood of an element $\bar{\boldsymbol{R}}$, the rigid body motion can be represented by vector $\underline{\boldsymbol{s}}$ as

$$\boldsymbol{R} = \bar{\boldsymbol{R}} \exp(\tilde{\boldsymbol{s}}), \quad \underline{\boldsymbol{s}} = \underline{s} + \epsilon \underline{s}^{\circ}, \quad \|\underline{s}\| < \delta, \tag{2.57}$$

where $\exp(\cdot)$ is the exponential map and $\|\underline{s}\| < \delta$ indicates that the rotation angle should be in a small range.

Suppose $\underline{\boldsymbol{s}} \in \mathrm{T}_{\boldsymbol{R}}\mathcal{SO}(3)$ and $\underline{\boldsymbol{r}} \in \mathrm{T}^*_{\boldsymbol{R}}\mathcal{SO}(3)$, where superscript $(\cdot)^*$ indicates the dual of a space. For instance, let $\underline{\boldsymbol{s}} = \underline{\delta \boldsymbol{u}} = \underline{\delta\psi} + \epsilon\, R^T \delta \underline{u}$ define the virtual motion of the rigid body, where $\underline{\delta\psi}$ and $R^T\delta\underline{u}$ are the virtual rotation and motion of the rigid body, respectively. Furthermore, let $\underline{\boldsymbol{r}} = \underline{m} + \epsilon\, \underline{f}$ define the generalized forces, where $\underline{m}$ and $\underline{f}$ are the moment and force applied to a rigid body, respectively. The inner product operation in dual number domain $\underline{\boldsymbol{s}}^T \underline{\boldsymbol{r}} = \underline{\delta\psi}^T \underline{m} + \epsilon\, (\underline{\delta\psi}^T \underline{f} + \delta \underline{u}^T R\, \underline{m})$ does not yield the expected virtual work: while the primal part of this dual scalar does indeed represent the virtual work done by the moment, its dual part has no physical meaning; in fact, its units are not correct and are not even consistent. Dimentberg [3] underlined the problems discussed in the previous paragraph; he advised: *"as a result, many dynamics and statics problems must be solved on the basis of general screw theory with the screw expressed by means of six Plücker coordinates."*

Brodsky and Shoham [9, 10] have shown that the use of dual numbers to handle dynamics problems is quite complicated and requires the introduction of dual inertia operators. This approach will not be pursued here. Following Dimentberg's advice, the *"extended notation"* is introduced to tackle dynamics problems. The inner product in tangent space is defined as

$$\langle \underline{\boldsymbol{s}}, \underline{\boldsymbol{r}} \rangle = \underline{s}^T \underline{r} = \underline{s}^T \underline{r} + (\underline{s}^\circ)^T \underline{r}^\circ. \tag{2.58}$$

Similarly, the Riemann metric is on defined as

$$\langle \underline{\boldsymbol{s}}, \underline{\boldsymbol{s}} \rangle_M = \underline{s}^T M \, \underline{s} = \begin{Bmatrix} \underline{s}^\circ \\ \underline{s} \end{Bmatrix}^T \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix} \begin{Bmatrix} \underline{s}^\circ \\ \underline{s} \end{Bmatrix}, \tag{2.59}$$

where $M \in \mathbb{R}^{6\times 6}$ is the Riemann metric tensor. For instance, $\underline{\boldsymbol{s}} = \underline{\boldsymbol{v}}$ represents the velocity of a rigid body, M is the mass matrix, and $1/2 \underline{s}^T M \, \underline{s}$ represents the kinematic energy. Therein, bold symbols indicates the extended notation that recast dual vectors and matrices in $\mathbb{D}^3$ to entities in $\mathbb{R}^6$, i.e.,

$$\underline{\boldsymbol{s}} = \underline{s} + \epsilon \, \underline{s}^\circ \rightarrow \underline{s} = \begin{Bmatrix} \underline{s}^\circ \\ \underline{s} \end{Bmatrix}, \quad \boldsymbol{S} = S + \epsilon \, S^\circ \rightarrow \mathcal{S} = \begin{bmatrix} S & S^\circ \\ 0 & S \end{bmatrix}. \tag{2.60}$$

The extended notations that will be used in the following part of this paper are listed in table. 2.2.

	Dual number notation	**Extended notation**
Velocity vector	$\underline{\boldsymbol{v}}$	$\underline{v}$
Curvature vector	$\underline{\boldsymbol{k}}$	$\underline{\mathcal{K}}$
Variation of motion	$\underline{\delta \boldsymbol{u}}$	$\underline{\delta u}$
Increment of motion	$\underline{\Delta \boldsymbol{u}}$	$\underline{\Delta u}$
Motion tensor	$\boldsymbol{R}$	$\mathcal{R}$
Tangent tensor	$\boldsymbol{T}$	$\mathcal{T}$

Table 2.2: The dual entity and extended notations.

2.11 Gradient and Hessian on $\mathcal{SO}_3$

Consider a differentiable function $f : \mathcal{SO}_3 \to \mathbb{R}$. The gradient of function f at $\boldsymbol{R}$, denoted as $\nabla_{\mathcal{R}} f$, is defined as

$$\underline{s}^T(\nabla_{\mathcal{R}} f) = \frac{\mathrm{d} f[\mathcal{R}\exp(\varepsilon\tilde{s})]}{\mathrm{d}\varepsilon}|_{\varepsilon=0}. \tag{2.61}$$

Clearly, $\nabla_{\mathcal{R}} f$ lives in the dual of the tangent space. The second order derivatives of f at $\boldsymbol{R}$, are not unique because that the matrix multiplication is not commutative. The seconder order derivatives are defined as

$$\underline{s}_L^T(\nabla^2_{\mathcal{R}} f)\underline{s}_R = \frac{\partial^2 f[\mathcal{R}\exp(\varepsilon_R\tilde{s}_R)\exp(\varepsilon_L\tilde{s}_L)]}{\partial\varepsilon_L\partial\varepsilon_R}|_{\varepsilon_L,\varepsilon_R=0}, \tag{2.62a}$$

$$\underline{s}_L^T(\bar{\nabla}^2_{\mathcal{R}} f)\underline{s}_R = \frac{\partial^2 f[\mathcal{R}\exp(\varepsilon_L\tilde{s}_L)\exp(\varepsilon_R\tilde{s}_R)]}{\partial\varepsilon_L\partial\varepsilon_R}|_{\varepsilon_L,\varepsilon_R=0}. \tag{2.62b}$$

It is verified easily that

$$\nabla^2_{\mathcal{R}} f = (\bar{\nabla}^2_{\mathcal{R}} f)^T. \tag{2.63}$$

The Taylor expansion of function f in a neighborhood of $\mathcal{R}$ is

$$\begin{aligned} f[\bar{\mathcal{R}}\exp(\tilde{\underline{s}})] &= f(\mathcal{R}) + \underline{s}^T(\nabla_{\mathcal{R}} f) + \frac{1}{2}\underline{s}^T(\nabla^2_{\mathcal{R}} f)\underline{s} \\ &= f(\mathcal{R}) + \underline{s}^T(\nabla_{\mathcal{R}} f) + \frac{1}{2}\underline{s}^T(\bar{\nabla}^2_{\mathcal{R}} f)\underline{s}. \end{aligned} \tag{2.64}$$

Suppose that $\nabla^2_{\mathcal{R}} f + \bar{\nabla}^2_{\mathcal{R}} f$ is positive definite, then $\bar{\mathcal{R}}$ is a local minimal of function f if the gradient $\nabla_{\mathcal{R}} f$ vanishes.

2.12 Solving nonlinear equation and optimization problems on $\mathcal{SO}(3)$

Consider a set of nonlinear equations formulated on $\mathcal{SO}_3$

$$\underline{r}(\mathcal{R}) = \underline{0}, \tag{2.65}$$

where vector valued function $\underline{r} : \mathcal{SO}_3 \to \mathbb{R}^6$. The nonlinear equation is solved by using Newton method

Algorithm 1: Newton method on $\mathcal{SO}_3$. Therein, $\underline{\Delta\psi}$ the primal part of $\underline{\Delta u}$ and δ is radius of the trust region.

1: Initial guess $\mathcal{R}$
2: **while** $\|\underline{r}\| > \varepsilon$ **do**
3: Evaluate Jacobin $\nabla_{\mathcal{R}}\underline{r}$ and residual $\underline{r}$
4: Solve $\underline{\Delta u} = -(\nabla_{\mathcal{R}}\underline{r}) \setminus \underline{r}$
5: $\underline{\Delta u} = \min\{\frac{\delta}{\|\underline{\Delta\psi}\|}, 1\}\underline{\Delta u}$
6: Update $\mathcal{R} = \mathcal{R}\exp(\tilde{\Delta u})$
7: **end while**

Consider an optimization problem

$$\begin{aligned} &\min\ J(\mathcal{R}, \underline{\theta}) \\ &\text{s.t. } \underline{r}(\mathcal{R}, \underline{\theta}) = \underline{0} \end{aligned} \tag{2.66}$$

where J represent the cost functional, vector $\underline{\theta}$ stores the design parameters, and $\underline{r} : \mathcal{R} \to \mathbb{R}^m$, $1 \leq n < 6$, represents the constraint. The augmented Lagrangian $L = J(\mathcal{R}, \underline{\theta}) + \underline{\nu}^T \underline{r}(\mathcal{R}, \underline{\theta})$ is introduced, where $\underline{\nu} \in \mathbb{R}^n$ stores the Lagrange multipliers. Vanishing of the first order variation, $\delta L = 0$, leads to the first-order optimality condition

$$\delta\underline{\theta} : \nabla_\theta J + (\nabla_\theta \underline{r})^T \underline{\nu} = \underline{0} \tag{2.67a}$$

$$\underline{\delta u} : \nabla_{\mathcal{R}} J + (\nabla_{\mathcal{R}} \underline{r})^T \underline{\nu} = \underline{0} \tag{2.67b}$$

$$\delta\underline{\nu} : \underline{r} = \underline{0} \tag{2.67c}$$

The nonlinear equations resulting from first-order condition are solved by using Newton method. An linearization of eq. (2.67) yields

$$\begin{bmatrix} \nabla^2_\theta L & \nabla^2_{\theta\mathcal{R}} & (\nabla_\theta \underline{r})^T \\ \nabla^2_{\mathcal{R}\theta} L & \nabla^2_{\mathcal{R}} L & (\nabla_{\mathcal{R}} \underline{r})^T \\ \nabla_\theta \underline{r} & \nabla_{\mathcal{R}} \underline{r} & \end{bmatrix} \begin{Bmatrix} \Delta\underline{\theta} \\ \underline{\Delta u} \\ \Delta\underline{\nu} \end{Bmatrix} = -\text{residuals}. \tag{2.68}$$

In most cases, the main effort of solving optimization problem is to derive the second order derivatives $\nabla^2_{\mathcal{R}} L$.

Chapter 3: Interpolation of motion

In this chapter, the interpolations of rotation and motion are treated. For simplicity of the discussion, the motion field is assumed to depend on a single variable η only. All the interpolation schemes presented in the chapter generalize easily to multiple dimensions.

3.1 Interpolation in Euclidean spaces

Interpolation techniques in Euclidean space are well established. Given a set of vectors, $\underline{x}_k \in \mathbb{R}^m$, located at grid points η_k, $k = 0, 1, \ldots, N$, classical interpolation schemes define the interpolated vector, $\underline{x}(\eta)$, as

$$\underline{x}(\eta) = \sum_{k=0}^{N} l_k(\eta)\underline{x}_k, \tag{3.1}$$

where $l_k(\eta)$ are the shape functions that are required to satisfy the property of partition of unity,

$$\sum_{k=0}^{N} l_k(\eta) = 1. \tag{3.2}$$

Interpolation scheme (3.1) represents the weighted average of the vectors at the grid points: shape function $ell_k(\eta)$ is the weight associated with vector $\underline{x}_k$. In the finite

element methods, it is customary to use the Lagrangian polynomial.

Interpolation scheme (3.1) can be recast as the following minimization problem,

$$\underline{x}(\eta) = \arg\min_{\underline{x}\in\mathbb{R}^m} J(\underline{x}, \underline{x}_k), \text{ where } J(\underline{x}, \underline{x}_k) = \sum_{k=0}^{N} l_k(\eta)\text{dist}_E^2(\underline{x}, \underline{x}_k). \tag{3.3}$$

The distance between points $\underline{x}$ and $\underline{x}_k$ in the Euclidean space is defined as $\text{dist}_E(\underline{x}, \underline{x}_k) = \sqrt{(\underline{x} - \underline{x}_k)^T(\underline{x} - \underline{x}_k)}$. The solution of minimization problem (3.3) is found by imposing the stationarity of objective function $J(\underline{x}, \underline{x}_k)$, leading to $\delta\underline{x}^T[\sum_{k=0}^{N} l_k(\eta)(\underline{x} - \underline{x}_k)] = \delta\underline{x}^T[\underline{x} - \sum_{k=0}^{N} l_k(\eta)\underline{x}_k] = 0$, which then implies interpolation scheme (3.1). Because the Hessian at the stationary point, I, is positive-definite, the stationary point is a minimum. Minimization problem (3.3) can be used to derive interpolation schemes in any space provided that the distance function, $\text{dist}(\cdot, \cdot)$, is a valid measure of distance in that space. Clearly, interpolation schemes can be derived in any space once a distance or metric of the space is defined.

3.2 Interpolation of motion fields

Application of classical interpolation scheme (3.1) to rigid-body motion fields leads to

$$\boldsymbol{R}(\eta) = \sum_{k=0}^{N} l_k(\eta)\boldsymbol{R}_k, \tag{3.4}$$

where $\boldsymbol{R}_k$ are the nodal motion tensors. The interpolated tensor does not represent a rigid-body motion because it is not orthogonal. This stems from the fact that motion fields do not form a linear space, in contrast with displacement fields that

do.

Interpolation scheme (3.1) cannot extended, as is, to interpolation on a manifold because linear operations are not valid on the manifold. Due to the special structure of the Euclidean space, scheme (3.1) provides a valid approach to interpolation in that space only. A natural question arises: is it possible to reformulate classical interpolation scheme (3.1) in a manner that becomes independent of the properties of the Euclidean space?

Figure 3.1 depicts the motion interpolation problem in a schematic manner. The motion tensor that brings reference frame $\mathcal{F}_I$ to the frame at grid point η_k is denoted $\boldsymbol{R}_k$ and the corresponding dual quaternion is denoted $\hat{\boldsymbol{q}}_k$. The desired interpolated motion tensor at point η is denoted $\boldsymbol{R}$ and the corresponding dual quaternion is denoted $\hat{\boldsymbol{q}}$. Nodal and interpolated motions are resolved in the inertial frame. For clarity, the dual quaternions that define the various motions, $\hat{\boldsymbol{q}}_k$, $k = 0, \ldots, N$ and $\hat{\boldsymbol{q}}$, are indicated in fig. 3.1. Parameter η can be interpreted as a spatial or a temporal parameter leading to the spatial or temporal interpolation problems, respectively.

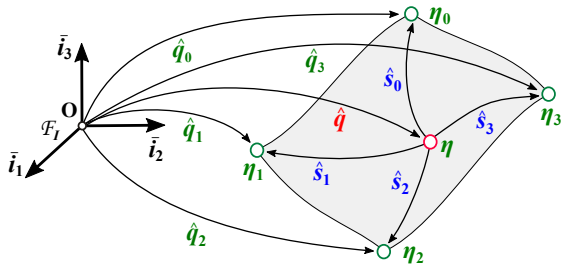

Figure 3.1: Interpolation of a motion field.

Next, the components of the relative motion tensors from point η to points η_k, all resolved in interpolated frame $\boldsymbol{R}$, are denoted $\boldsymbol{S}_k = \boldsymbol{R}^T \boldsymbol{R}_k$. The corresponding

dual quaternions and motion parameter vectors are denoted $\hat{\boldsymbol{s}}_k = \boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{q}}_k$ and $\underline{\boldsymbol{s}}_k$, respectively. The following notation is introduced

$$\boldsymbol{G} = \sum_{k=0}^{N} l_k(\eta)\boldsymbol{R}_k, \quad \boldsymbol{S} = \sum_{k=0}^{N} l_k(\eta)\boldsymbol{S}_k = S + \epsilon\, S^\circ, \quad \boldsymbol{S} = \boldsymbol{R}^T\boldsymbol{G}, \tag{3.5}$$

where the last equation is a direct consequence of the previous definitions. The corresponding relationships for the dual quaternions are

$$\hat{\boldsymbol{g}} = \sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{q}}_k, \quad \hat{\boldsymbol{s}} = \sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{s}}_k = \hat{s} + \epsilon\, \hat{s}^\circ, \quad \hat{\boldsymbol{s}} = \boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}. \tag{3.6}$$

Interpolation on $\mathcal{SO}(3)$ is obtained easily as an extension of eq. (3.3)

$$\boldsymbol{R} = \arg\min_{\boldsymbol{R}\in\mathcal{SO}(3)} \boldsymbol{J}(\boldsymbol{R}, \boldsymbol{R}_k), \text{ with } \boldsymbol{J}(\boldsymbol{R}, \boldsymbol{R}_k) = \sum_{k=0}^{N} l_k \mathrm{dist}^2(\boldsymbol{R}, \boldsymbol{R}_k). \tag{3.7}$$

Different metrics leads to different interpolation schemes. Minimization of a dual function is defined as following

Definition 3.2.1 (Minimization of a dual function)**.** *The minimization of a dual function of dual variables implies the satisfaction of two conditions:* (1) *the variation of the function vanishes and* (2) *the primal part of the function achieves a minimum.*

3.3 The distance between two motions

A metric or distance is a function, denoted $\mathrm{dist}(x_1, x_2)$, that defines a measure of distance between two elements, denoted x_1 and x_2, of set U. In gen-

eral, metrics must satisfy the following four conditions: *(1)* the metric is non-negative, i.e., $\text{dist}(x_1, x_2) \geq 0$, *(2)* the metric vanishes for identical elements only, i.e., $\text{dist}(x_1, x_2) = 0$ if and only if $x_1 = x_2$, *(3)* the metric is a symmetric function of its arguments, i.e., $\text{dist}(x_1, x_2) = \text{dist}(x_2, x_1)$, and *(4)* the triangular inequality is satisfied, i.e., $\text{dist}(x_1, x_2) \leq \text{dist}(x_1, x_3) + \text{dist}(x_3, x_2)$ for any element $x_3 \in U$.

These four conditions do not define the metric uniquely. In practice, metrics are selected to be geometrically meaningful and easy to handle mathematically. Usually, the distance between two vectors of the Euclidean space, $\underline{x}_1, \underline{x}_2 \in \mathbb{R}^m$, is defined as $\|\underline{x}_1 - \underline{x}_2\| = \sqrt{(\underline{x}_1 - \underline{x}_2)^T(\underline{x}_1 - \underline{x}_2)}$.

Motions are defined in the domain of dual numbers and the associated metrics are dual number functions, selected to be analytic. If $\text{dist}(\boldsymbol{R}_1, \boldsymbol{R}_2)$ is an analytic function, its primal part is $\text{dist}(R_1, R_2)$, where rotation tensors R_1 and R_2 are the primal parts of motion tensors $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$, respectively. Because the norm of dual numbers is not defined, dual numbers cannot be compared. Consequently, dual number metrics must satisfy the following four conditions: *(1)* the primal part of the metric is non-negative, i.e., $\text{dist}(R_1, R_2) \geq 0$, *(2)* the metric vanishes for identical elements only, i.e., $\text{dist}(\boldsymbol{R}_1, \boldsymbol{R}_2) = \boldsymbol{0}$ if and only if $\boldsymbol{R}_1 = \boldsymbol{R}_2$, *(3)* the metric is a symmetric function of its arguments, i.e., $\text{dist}(\boldsymbol{R}_1, \boldsymbol{R}_2) = \text{dist}(\boldsymbol{R}_2, \boldsymbol{R}_1)$, and *(4)* the triangular inequality is satisfied for the primal part of the metric, i.e., $\text{dist}(R_1, R_2) \leq \text{dist}(R_1, R_3) + \text{dist}(R_3, R_2)$ for any rotation R_3.

With this definition, any metric in $\mathbb{R}$ for rotation can be extended to a corresponding metric in $\mathbb{D}$ for motion simply by replacing the operation on real numbers by the corresponding operation on dual numbers, as expected from the principle

of transference. Metrics for rotation have been proposed based on various types of representations of rotation: the rotation tensor [36, 37, 39, 40, 122–125] (defined in $\mathbb{R}^{3\times3}$), unit quaternions [36, 39, 40, 124, 125] (defined in $\mathbb{R}^4$), or rotation parameter vectors [39,125,126] (defined in $\mathbb{R}^3$). Although rotation metrics can be defined based on Euler angles [125, 126], these metrics are not invariant because Euler angles do not form a vectorial representation of rotation. The next four sections present four metrics of motion based on different types of representations of motion.

3.3.1 The matrix based metric

The first metric to be proposed is based on the expression of the motion tensor: the distance between two motions, $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$, is defined as the Euclidean distance between the matrix representations in $\mathbb{D}^{3\times3}$, leading to the *matrix metric*,

$$\begin{aligned} \mathrm{dist}^2_{\mathrm{m}}(\boldsymbol{R}_1, \boldsymbol{R}_2) &= \|\boldsymbol{R}_1 - \boldsymbol{R}_2\|^2_F = \mathrm{tr}[(\boldsymbol{R}_1 - \boldsymbol{R}_2)^T(\boldsymbol{R}_1 - \boldsymbol{R}_2)] \\ &= 6 - 2\mathrm{tr}(\boldsymbol{R}_{12}), \end{aligned} \tag{3.8}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and $\boldsymbol{R}_{12} = \boldsymbol{R}_1^T\boldsymbol{R}_2$ denotes the relative motion. Let $\boldsymbol{\phi}_{12} = \theta_{12} + \epsilon\, d_{12}$ be the dual angle associated with relative motion $\boldsymbol{R}_{12}$. The trace of the relative motion now becomes $\mathrm{tr}(\boldsymbol{R}_{12}) = 1 + 2\cos(\boldsymbol{\phi}_{12})$ and introducing this result into eq. (3.8) yields

$$\mathrm{dist}^2_{\mathrm{m}}(\boldsymbol{R}_1, \boldsymbol{R}_2) = 4(1 - \cos\boldsymbol{\phi}_{12}) = 4(1 - \cos\theta_{12}) + \epsilon\, 4d_{12}\sin\theta_{12}. \tag{3.9}$$

Because the relative rotation angle $\theta_{12} \in [-\pi, \pi)$, the primal part of matrix metric $\mathrm{dist_m}(R_1, R_2) \in [0, 2\sqrt{2}]$.

3.3.2 The quaternion based metric

If motions $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ are represented by dual quaternions $\hat{\boldsymbol{q}}_1$ and $\hat{\boldsymbol{q}}_2$, respectively, the distance between the motions can be measured by the Euclidean distance between the dual quaternion in $\mathbb{D}^4$, leading to the *quaternion metric*,

$$\begin{aligned} \mathrm{dist}_{\mathrm{q}}^2(\boldsymbol{R}_1, \boldsymbol{R}_2) &= \|\hat{\boldsymbol{q}}_1 - \hat{\boldsymbol{q}}_2\|^2 = \|\hat{\mathbf{1}} - \hat{\boldsymbol{q}}_{12}\|^2 = 2 - 2\mathrm{scal}(\hat{\boldsymbol{q}}_{12}) \\ &= 2(1 - \cos \boldsymbol{\phi}_{12}/2), \end{aligned} \tag{3.10}$$

where notation $\|\cdot\|$ indicates the Euclidean norm and $\hat{\boldsymbol{q}}_{12} = \boldsymbol{A}^T(\hat{\boldsymbol{q}}_1)\hat{\boldsymbol{q}}_2$ is the dual quaternion representing the relative motion. The second equality of eq. (3.10) results from the fact that matrix $\boldsymbol{A}$ is orthogonal for unit dual quaternions. Because quaternions $+\hat{\boldsymbol{q}}$ and $-\hat{\boldsymbol{q}}$ represent the same motion, it is always possible to select the signs of quaternions $\hat{\boldsymbol{q}}_1$ and $\hat{\boldsymbol{q}}_2$ to render the primal part of $\mathrm{scal}(\hat{\boldsymbol{q}}_{12})$ positive. The primal part of matrix metric $\mathrm{dist_q}(R_1, R_2)$ is in the range of $[0, \sqrt{2}]$.

3.3.3 The motion parameter vector based metric

Let $\underline{\boldsymbol{p}}_{12}$ be the motion parameter vector representing relative motion $\boldsymbol{R}_1^T\boldsymbol{R}_2$. The Euclidean norm of the relative motion parameter vector in $\mathbb{D}^3$ leads to the *vector metric*,

$$\mathrm{dist}_{\mathrm{v}}^2(\boldsymbol{R}_1, \boldsymbol{R}_2) = \|\underline{\boldsymbol{p}}_{12}\|^2. \tag{3.11}$$

Any of the vectorial parameterizations of motion presented in section 2.6 can be used to define the distance. Selecting, for instance, the Euler-Rodrigues parameterization, see section 2.6, leads to

$$\text{dist}_{\text{v}}^2(\boldsymbol{R}_1, \boldsymbol{R}_2) = \|\underline{\boldsymbol{p}}_{12}\|^2 = 4\sin^2(\boldsymbol{\phi}_{12}/2) = 2(1 - \cos\boldsymbol{\phi}_{12}), \tag{3.12}$$

which is half of the matrix metric defined by eq. (3.10). If the motion parameter vector is selected as $\underline{\boldsymbol{p}} = 4\sin(\boldsymbol{\phi}/4)\bar{\boldsymbol{n}}$, the vector metric becomes

$$\text{dist}_{\text{v}}^2(\boldsymbol{R}_1, \boldsymbol{R}_2) = \|\underline{\boldsymbol{p}}_{12}\|^2 = 16\sin^2(\boldsymbol{\phi}_{12}/4) = 8(1 - \cos\boldsymbol{\phi}_{12}/2), \tag{3.13}$$

which is four times of the quaternion metric defined in eq. (3.10). Other choices of the generating function will lead to other definitions of the metric.

3.3.4 The geodesic based metric

If motion parameter vector $\underline{\boldsymbol{p}} = \boldsymbol{\phi}\bar{\boldsymbol{n}}$ is used, see section 2.6, vector metric (3.11) becomes

$$\text{dist}_{\text{g}}^2(\boldsymbol{R}_1, \boldsymbol{R}_2) = \|\text{axial}[\log(\boldsymbol{R}_1^T\boldsymbol{R}_2)]\|^2 = \boldsymbol{\phi}_{12}^2. \tag{3.14}$$

The geodesic metric corresponds to the arc-length along the geodesic line on manifold $\mathcal{SO}(3)$ starting from motion $\boldsymbol{R}_1$ and ending at motion $\boldsymbol{R}_2$, as discussed in section [71]. The primal part of matrix metric $\text{dist}_{\text{g}}(R_1, R_2)$ is in the range of $[0, \pi]$.

3.3.5 Properties of the four metrics

For reference, the definitions of the matrix, quaternion, and geodesic metrics are listed in table 3.1. It is verified easily that all metrics satisfy the four conditions required for a proper metric. The distance measures obtained with the matrix and quaternion metrics are the same as those obtained with the vector metric based on different choices of the motion parameter vectors within a multiplicative constant.

This does not mean, however, that the vector metric is well suited for global motion interpolation. Indeed, the rotation parts of motion parameter vectors represent rotation with three parameters only, and hence, all encounter singularities, as shown by Stuelpnagel [127]. In contrast, the matrix metric, which is based on a nine-parameter representation of motion, is free of singularity over the entire range of motions. Because a two-to-one correspondence exists between dual quaternions and motion, the quaternion metric also encounters problems for global interpolation of periodic motion.

Metric type	**Definition**	**Value**
Matrix metric	$\text{dist}_\text{m} = \|\boldsymbol{R}_1 - \boldsymbol{R}_2\|_F$	$\sqrt{4(1-\cos\boldsymbol{\phi}_{12})}$
Quaternion metric	$\text{dist}_\text{q} = \|\hat{\boldsymbol{q}}_1 - \hat{\boldsymbol{q}}_2\|^2$	$\sqrt{2(1-\cos\boldsymbol{\phi}_{12}/2)}$
Vector metric	$\text{dist}_\text{v} = \|\underline{\boldsymbol{p}}_{12}\|$	$\begin{cases}\sqrt{2(1-\cos\boldsymbol{\phi}_{12})},\\ \sqrt{8(1-\cos\boldsymbol{\phi}_{12}/2)},\end{cases}$
Geodesic metric	$\text{dist}_\text{g} = \|\text{axial}[\log(\boldsymbol{R}_1^T\boldsymbol{R}_2)]\|$	$\boldsymbol{\phi}_{12}$

Table 3.1: List of proposed metrics on $\mathcal{SO}(3)$. The two values listed for the vector metric correspond to two choices of generating function: $\underline{\boldsymbol{p}} = 2\sin(\boldsymbol{\phi}/2)\bar{\boldsymbol{n}}$ and $\underline{\boldsymbol{p}} = 4\sin(\boldsymbol{\phi}/4)\bar{\boldsymbol{n}}$.

A metric is objective and tensorial if

$$\mathrm{dist}(\boldsymbol{R}\,\boldsymbol{R}_1, \boldsymbol{R}\,\boldsymbol{R}_2) = \mathrm{dist}(\boldsymbol{R}_1, \boldsymbol{R}_2), \quad \text{(objective)} \tag{3.15a}$$

$$\mathrm{dist}(\boldsymbol{R}_1\,\boldsymbol{R}, \boldsymbol{R}_2\,\boldsymbol{R}) = \mathrm{dist}(\boldsymbol{R}_1, \boldsymbol{R}_2), \quad \text{(tensorial)} \tag{3.15b}$$

for all $\boldsymbol{R}, \boldsymbol{R}_1, \boldsymbol{R}_2 \in \mathcal{SO}(3)$, respectively. Objectivity implies that the distance remains unchanged under the superposition of a rigid-body motion. If the distance remains unchanged when choosing a different body-fixed frame, it is tensorial, i.e., the distance is a tensor of order zero. All four metrics defined in table 3.1 are objective because they depend on the relative motion only, $\boldsymbol{R}_1^T\boldsymbol{R}_2 = (\boldsymbol{R}\,\boldsymbol{R}_1)^T(\boldsymbol{R}\,\boldsymbol{R}_2)$. It is verified easily that the primal parts all four metrics are also tensorial although their dual parts are not.

Under the superposition of rigid-body motion $\boldsymbol{R}_g$, the motion tensors at the grid points become $\boldsymbol{R}_g\,\boldsymbol{R}_k$. If the metric is left-invariant, $\mathrm{dist}(\boldsymbol{R}, \boldsymbol{R}_k) = \mathrm{dist}(\boldsymbol{R}_g\,\boldsymbol{R}, \boldsymbol{R}_g\,\boldsymbol{R}_k)$, and hence, objective function (3.7) remains invariant under the superposition of rigid-body motions. Therefore, interpolation scheme (3.7) is objective for any of the four metrics listed in table 3.1.

3.4 Interpolation based on the matrix metric

Using matrix metric (3.8), the objective function of minimization problem (3.7) becomes

$$\boldsymbol{J}(\boldsymbol{R}, \boldsymbol{R}_k) = \sum_{k=0}^{N} l_k(\eta)[6 - 2\mathrm{tr}(\boldsymbol{R}^T \boldsymbol{R}_k)] = 6 - 2\mathrm{tr}(\boldsymbol{R}^T \boldsymbol{G}) = 6 - 2\mathrm{tr}(\boldsymbol{S}). \tag{3.16}$$

According to definition (3.2.1), the minimization of this objective function requires the satisfaction of stationarity condition $\delta\boldsymbol{J} = \boldsymbol{0}$, leading to $\delta\boldsymbol{J} = -2\mathrm{tr}(\delta\boldsymbol{S}) = 2\mathrm{tr}(\delta\tilde{\boldsymbol{u}}\,\boldsymbol{S}) = \underline{\boldsymbol{0}}$, where $\delta\tilde{\boldsymbol{u}} = \boldsymbol{R}^T\delta\boldsymbol{R}$ is the virtual motion vector. Introducing trace identity (A.3e) then yields $\delta\boldsymbol{J} = -4\underline{\delta\boldsymbol{u}}^T\mathrm{axial}(\boldsymbol{S}) = 0$, which implies that dual matrix $\boldsymbol{S}$ must be symmetric. Definition (3.2.1) also requires the minimization of the primal part of the objective function. The second variation of the objective function is $\delta^2\boldsymbol{J} = -4\underline{\delta\boldsymbol{u}}^T\mathrm{axial}(\delta\boldsymbol{S}) = 4\underline{\delta\boldsymbol{u}}^T\mathrm{axial}(\delta\tilde{\boldsymbol{u}}\,\boldsymbol{S})$ and identity (A.3b) then yields $\delta^2\boldsymbol{J} = 2\underline{\delta\boldsymbol{u}}^T[\mathrm{tr}(\boldsymbol{S})\boldsymbol{I} - \boldsymbol{S}]\underline{\delta\boldsymbol{u}}$; the Hessian of the problem is $[\mathrm{tr}(\boldsymbol{S})\boldsymbol{I} - \boldsymbol{S}]$ and its primal part, $[\mathrm{tr}(S)I - S]$, must be positive-definite if the primal part of the objective function is to be a minimum.

3.4.1 The implicit interpolation scheme

The solution of minimization problem (3.7) implies the vanishing of the axial part of matrix $\boldsymbol{S}$. Using eq. (3.5) now yields $\mathrm{axial}(\boldsymbol{S}) = \mathrm{axial}(\sum l_k\boldsymbol{R}^T\boldsymbol{R}_k) =$

$\sum l_k \text{axial}(\boldsymbol{S}_k) = \underline{0}$ and finally

$$\sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{s}}_k(\eta) = \underline{0}, \tag{3.17}$$

where $\underline{\boldsymbol{s}}_k(\eta) = \text{axial}(\boldsymbol{S}_k(\eta)) = \sin\boldsymbol{\phi}_k(\eta)\bar{\boldsymbol{n}}_k(\eta)$ is the linear motion parameter vector, see section 2.6, associated with relative motion matrix $\boldsymbol{S}_k(\eta) = \boldsymbol{R}^T(\eta)\boldsymbol{R}_k$. Because relative motion parameter vector $\underline{\boldsymbol{s}}_k(\eta)$ is a nonlinear function of the unknown interpolated motion tensor $\boldsymbol{R}(\eta)$, interpolation scheme (3.17) is an implicit interpolation scheme.

3.4.2 The polar decomposition approach

As implied by eq. (3.5), the interpolated motion is such that $\boldsymbol{G} = \boldsymbol{RS}$, where dual matrix $\boldsymbol{R}$ is orthogonal and dual matrix $\boldsymbol{S}$ is symmetric; furthermore, matrix $[\text{tr}(S)I - S]$ must be positive-definite. The polar decomposition theorem (A.5.1) provides a unique solution to this problem. This approach has been proposed by a number of authors [13,23,128] and is a natural extension of the polar decomposition of the deformation gradient tensor used in continuum mechanics.

3.4.3 The quaternion approach

The quaternion-based approach for the minimization of objective function (3.16) was proposed by Davenport [129,130]. As shown in the paragraphs above, the minimization of $\boldsymbol{J}$ is equivalent to the maximization of $\text{tr}(\boldsymbol{S})$. In view of equation (2.23),

the maximization of $\mathrm{tr}(\boldsymbol{S})$ is stated as

$$\begin{aligned}\max_{\hat{\boldsymbol{q}}\in\{\mathbb{D}^4|\hat{\boldsymbol{q}}^T\hat{\boldsymbol{q}}=\mathbf{1}\}} \mathrm{tr}\left[\boldsymbol{R}^T(\hat{\boldsymbol{q}})\boldsymbol{G}\right] &= \max_{\hat{\boldsymbol{q}}\in\{\mathbb{D}^4|\hat{\boldsymbol{q}}^T\hat{\boldsymbol{q}}=\mathbf{1}\}} \hat{\boldsymbol{q}}^T\boldsymbol{W}(\boldsymbol{G})\,\hat{\boldsymbol{q}} \\ &= \max_{\hat{\boldsymbol{q}}\in\mathbb{D}^4,\boldsymbol{\lambda}\in\mathbb{D}} \left[\hat{\boldsymbol{q}}^T\boldsymbol{W}(\boldsymbol{G})\,\hat{\boldsymbol{q}} + \boldsymbol{\lambda}(\mathbf{1}-\hat{\boldsymbol{q}}^T\hat{\boldsymbol{q}})\right],\end{aligned} \tag{3.18}$$

where symmetric operator matrix $\boldsymbol{W}(\boldsymbol{G})$ is defined by eq. (2.24) and $\boldsymbol{\lambda}$ is the Lagrange multiplier used to enforce the normality constraint for dual quaternion $\hat{\boldsymbol{q}}$. Variation with respect to $\boldsymbol{\lambda}$ gives the normality condition for dual quaternion $\hat{\boldsymbol{q}}$ and variation with respect to $\hat{\boldsymbol{q}}$ yields $\boldsymbol{W}(\boldsymbol{G})\,\hat{\boldsymbol{q}} = \boldsymbol{\lambda}\hat{\boldsymbol{q}}$, which also implies $\boldsymbol{\lambda} = \hat{\boldsymbol{q}}^T\boldsymbol{W}(\boldsymbol{G})\,\hat{\boldsymbol{q}}$. In summary, the maximization problem requires $\boldsymbol{\lambda}$ to be the eigenvalue of matrix $\boldsymbol{W}$ with the largest primal part.

The dual eigenvalue problem is recast as $[W(G) + \epsilon\ W^\circ(G^\circ)](\hat{q} + \epsilon\ \hat{q}^\circ) = (\lambda + \epsilon\ \lambda^\circ)(\hat{q} + \epsilon\ \hat{q}^\circ)$, which expands to

$$W(G)\hat{q} = \lambda\hat{q}, \tag{3.19a}$$

$$W(G)\hat{q}^\circ + W^\circ(G^\circ)\hat{q} = \lambda\hat{q}^\circ + \lambda^\circ\hat{q}. \tag{3.19b}$$

Solution of eigenvalue problem (3.19a) yields $\lambda_{\max}$, the largest eigenvalue of W, and the associated eigenvector $\hat{q}$. Combining eq. (3.19b) with normality condition $\hat{q}^T\hat{q}^\circ = 0$ yields a linear system for the remaining unknowns

$$\begin{bmatrix} W(G) - \lambda_{\max} I_4 & -\hat{q} \\ -\hat{q}^T & 0 \end{bmatrix} \begin{Bmatrix} \hat{q}^\circ \\ \lambda^\circ \end{Bmatrix} = - \begin{Bmatrix} W^\circ(G^\circ)\hat{q} \\ 0 \end{Bmatrix}. \tag{3.20}$$

If dual matrix $\boldsymbol{G}$ happens to represent a motion, i.e., is orthogonal, eq. (2.25) implies $\boldsymbol{W}(\boldsymbol{G}) = 4\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}^T - \boldsymbol{I}$ and the algorithm degenerates into the determination of the unit dual quaternion associated with an orthogonal dual matrix. The process is singularity free and yields a uniquely defined unit dual quaternion. The primal part of this problem corresponds to the scheme proposed by Klumpp and Shepperd [116, 117] for the determination of the unit quaternion associated with an orthogonal tensor.

3.4.4 Relationship between the two approaches

Let $\boldsymbol{G} = \boldsymbol{R}(\hat{\boldsymbol{q}}_\mathrm{p})\boldsymbol{S}$ denote the solution of the minimization problem obtained from the polar decomposition approach. Identity (2.26) now yields the following result

$$\boldsymbol{W}(\boldsymbol{G}) = \boldsymbol{A}^T(\hat{\boldsymbol{q}}_\mathrm{p})\boldsymbol{W}(\boldsymbol{S})\boldsymbol{A}(\hat{\boldsymbol{q}}_\mathrm{p}). \tag{3.21}$$

Because matrix $\boldsymbol{A}$ is orthogonal, eq. (3.21) represents a similarity transformation and hence, the eigenvalues of matrices $\boldsymbol{W}(\boldsymbol{S})$ and $\boldsymbol{W}(\boldsymbol{G})$ are identical.

Equation (2.24) provides the explicit expression of $\boldsymbol{W}(\boldsymbol{S})$ as

$$\boldsymbol{W}(\boldsymbol{S}) = \begin{bmatrix} \mathrm{tr}(\boldsymbol{S}) & \underline{\boldsymbol{0}} \\ \underline{\boldsymbol{0}}^T & 2\boldsymbol{S} - \mathrm{tr}(\boldsymbol{S})\boldsymbol{I} \end{bmatrix}.$$

The eigenvalues of the primal part, $W(S)$, can be found easily as [130–132] $\mathrm{tr}(S) = \sqrt{\lambda_3}+\sqrt{\lambda_2}+\eta\sqrt{\lambda_1}$, $\sqrt{\lambda_3}-\sqrt{\lambda_2}-\eta\sqrt{\lambda_1}$, $-\sqrt{\lambda_3}+\sqrt{\lambda_2}-\eta\sqrt{\lambda_1}$, and $-\sqrt{\lambda_3}-\sqrt{\lambda_2}+\eta\sqrt{\lambda_1}$, where $\sqrt{\lambda_1} \leq \sqrt{\lambda_2} \leq \sqrt{\lambda_3}$ are the singular values of $\boldsymbol{G}$, and $\eta = \pm 1$ depends on

the sign of $\det(G)$, as discussed in section A.4. Clearly, the largest eigenvalue of matrix $W(S)$ is $\mathrm{tr}(S)$, the corresponding eigenvector is the identity quaternion, $\hat{1}$. In view of the similarity transformation, the largest eigenvalue of matrix $W(G)$ is $\hat{1}$ and the corresponding eigenvector is $A(\hat{q}_\mathrm{p})\hat{1} = \hat{q}_\mathrm{p}$. Clearly, the solutions of the polar decomposition and quaternion approaches are identical.

The polar decomposition and quaternion approaches have been presented independently in the literature and the previous two sections show that both approaches provide solutions of the same minimization problem. The present section proves that these two solutions are identical, as expected.

Discussion 3.4.1. *The matrix metric based interpolation can be summarized as follows:* (1) *interpolate the matrix representations in Euclidean space* $\mathbb{D}^9$ *at the grid points and* (2) *map the result back onto the manifold using the minimization procedure. The second step of the process can be understood as a projection onto the manifold via a "closest point projection," the meaning of which depends on the selected metric. Because the interpolation scheme operates on a set of redundant variables (nine dual numbers), all singularities are avoided. On the other hand, the projection operation is computationally expensive: indeed, both polar decomposition and quaternion approaches require the solution of an eigenvalue problem.*

Matrix metric based interpolation has been investigated by numerous authors for rotation [18–21], and motion [18, 23]; both computer vision and finite element applications were targeted. In the finite element method, the closest point projection adds to the complexity of the formulation and hence, some authors [24, 25, 72] simply

ignore this step, leading to simpler formulations. In spectral methods, the closest point projection vanishes from the formulation naturally; it only arises as a post-processing step if the interpolated field must be evaluated at intermediate locations between the grid points [23].

Averaging of rotation have received considerable attention in many areas of engineering. This problem is very similar to interpolation: in the case of averaging, the weight functions $l_k(\eta)$ in eq. (3.7) are all set to unity. Matrix metric based averaging has been investigated for rotation [36, 37, 39, 40, 124] and for motion [133]. The estimation of the orientation of spacecraft, crystals, or other objects from measured data leads to a rotation averaging problem, known as Wahba's problem [134]. Approaches to the solution of this problem fall into two categories: (1) *approaches based on the polar decomposition theorem, or equivalently, on the singular value decomposition [37, 124, 130, 135–138] and* (2) *approaches based on quaternion algebra [129, 132, 138–140]. Both approaches can be extended easily from rotation to motion by using dual entities. Clearly, the same concepts and tools are used for interpolation and averaging.*

The multiplicative decomposition of the deformation gradient tensor, G, into a rotation tensor, R, and a stretch tensor, S, as $G = R\,S$, is a fundamental tool of continuum mechanics [141, 142]. This decomposition is unique and is provided by the polar decomposition theorem. This operation can be though of as finding rotation tensor R that is the "closest point projection" of the deformation gradient tensor.

3.5 Interpolation based on the quaternion metric

Using quaternion metric (3.10), the objective function of minimization problem (3.7) becomes

$$\boldsymbol{J}(\hat{\boldsymbol{q}}) = \sum_{k=0}^{N} l_k(\eta)[2 - 2\text{scal}(\hat{\boldsymbol{s}}_k)] = 2 - 2\text{scal}[\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}] = 2 - 2\hat{\boldsymbol{q}}^T\hat{\boldsymbol{g}}, \tag{3.22}$$

where the second equality results from eq. (3.6) and the last from the definition of matrix $\boldsymbol{A}$ in eq. (2.21). In general, dual quaternion $\hat{\boldsymbol{g}}$ is not unit.

In view of definition (3.2.1), the minimization of this objective function requires the satisfaction of stationarity condition $\delta\boldsymbol{J} = \boldsymbol{0}$, leading to $\delta\boldsymbol{J} = -2\delta\hat{\boldsymbol{q}}^T\hat{\boldsymbol{g}} = -\underline{\delta\boldsymbol{u}}^T[\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}] = \hat{\boldsymbol{0}}$. Note that the scalar part of $\underline{\delta\boldsymbol{u}}$ always vanishes and hence, the stationarity condition implies $\text{vec}[\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}] = \text{vec}(\hat{\boldsymbol{s}}) = \underline{\boldsymbol{0}}$, i.e., $\underline{\boldsymbol{s}} = \underline{\boldsymbol{0}}$. Next, definition (3.2.1) requires the minimization of the primal part of the objective function, $J = 2 - 2\hat{q}^T\hat{g}$.

Stationarity condition $\underline{\boldsymbol{s}} = \underline{\boldsymbol{0}}$ can be solved explicitly. Indeed, $\text{vec}[\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}] = \underline{\boldsymbol{0}}$ implies $\underline{\boldsymbol{s}} = -\boldsymbol{\pi}\underline{\boldsymbol{q}} + \boldsymbol{\eta}\underline{\boldsymbol{g}} - \tilde{\boldsymbol{q}}\underline{\boldsymbol{g}} = \underline{\boldsymbol{0}}$, where $\boldsymbol{\eta} = \text{scal}(\hat{\boldsymbol{q}})$, $\underline{\boldsymbol{q}} = \text{vec}(\hat{\boldsymbol{q}})$, $\boldsymbol{\pi} = \text{scal}(\hat{\boldsymbol{g}})$, and $\underline{\boldsymbol{g}} = \text{vec}(\hat{\boldsymbol{g}})$. The solution of equation $-\boldsymbol{\pi}\underline{\boldsymbol{q}} + \boldsymbol{\eta}\underline{\boldsymbol{g}} - \tilde{\boldsymbol{q}}\underline{\boldsymbol{g}} = \underline{\boldsymbol{0}}$ implies that vectors $\underline{\boldsymbol{q}}$ and $\underline{\boldsymbol{g}}$ are parallel, i.e., $\boldsymbol{\eta}\underline{\boldsymbol{g}} = \boldsymbol{\pi}\underline{\boldsymbol{q}}$. Multiplying this relationship by itself yields $\boldsymbol{\pi}^2\|\underline{\boldsymbol{q}}\|^2 = \boldsymbol{\eta}^2\|\underline{\boldsymbol{g}}\|^2 = (\boldsymbol{1} - \|\underline{\boldsymbol{q}}\|^2)\|\underline{\boldsymbol{g}}\|^2$ and solving for $\|\underline{\boldsymbol{q}}\|^2$ leads to $\|\underline{\boldsymbol{q}}\|^2 = \|\underline{\boldsymbol{g}}\|^2/(\boldsymbol{\pi}^2 + \|\underline{\boldsymbol{g}}\|^2) = \|\underline{\boldsymbol{g}}\|^2/\|\hat{\boldsymbol{g}}\|^2$. Because $\underline{\boldsymbol{q}}$ and $\underline{\boldsymbol{g}}$ are parallel, $\|\underline{\boldsymbol{q}}\|^2 = \|\underline{\boldsymbol{g}}\|^2/\|\hat{\boldsymbol{g}}\|^2$ implies $\underline{\boldsymbol{q}} = \underline{\boldsymbol{g}}/\|\hat{\boldsymbol{g}}\|$. Introducing $\underline{\boldsymbol{q}} = \underline{\boldsymbol{g}}/\|\hat{\boldsymbol{g}}\|$ into identity $\boldsymbol{\eta}\underline{\boldsymbol{g}} = \boldsymbol{\pi}\underline{\boldsymbol{q}}$ yields to $\boldsymbol{\eta} = \boldsymbol{\pi}/\|\hat{\boldsymbol{g}}\|$. Combining the

scalar and vector parts leads to

$$\hat{\boldsymbol{q}} = \frac{\hat{\boldsymbol{g}}}{\|\hat{\boldsymbol{g}}\|} = \frac{\sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{q}}_k}{\|\sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{q}}_k\|}. \tag{3.23}$$

Because $\boldsymbol{A}^T(\hat{\boldsymbol{q}})$ is orthogonal, the norms of relative dual quaternion $\hat{\boldsymbol{s}}$ becomes $\|\hat{\boldsymbol{s}}\|^2 = [\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}]^T[\boldsymbol{A}^T(\hat{\boldsymbol{q}})\hat{\boldsymbol{g}}] = \|\hat{\boldsymbol{g}}\|^2$. Because the vector part of $\hat{\boldsymbol{s}}$ vanishes, it follows that $\hat{\boldsymbol{s}} = \|\hat{\boldsymbol{g}}\|\hat{\mathbf{1}}$. The only singularity that can arise in eq. (3.23) is the vanishing of the denominator, i.e., $\sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{q}}_k = \underline{\mathbf{0}}$; this hardly ever happens in practical problems.

To guaranty that the primal part of the objective function, $J(\hat{q}) = 2 - 2\hat{q}^T\hat{g}$, reaches its minimum, its Hessian should be positive-definite. Taking the second-order variation of $J(\hat{q})$ leads to

$$\begin{aligned} \delta^2 J &= -\underline{\delta\psi}^T \mathrm{vec}[A^T(\delta\hat{q})\underline{g}] = -\underline{\delta\psi}^T \mathrm{vec}[A^T(\delta\hat{q})A(\hat{q})\hat{s}] \\ &= \underline{\delta\psi}^T \mathrm{vec}[A(\hat{\delta\psi}/2)\hat{s}] = 1/2\ \sigma\underline{\delta\psi}^T\underline{\delta\psi}, \end{aligned} \tag{3.24}$$

where $\sigma = \mathrm{scal}(\hat{s})$. As discussed in section 3.3.2, it is always possible to select the sign of $\hat{\boldsymbol{q}}$ to render σ non-negative and hence, the primal part of the Hessian, σI, is positive semi-definite; the stationary point is a minimum, as required.

In summary, the closed-form solution of the quaternion-metric based interpolation scheme is

$$\hat{\boldsymbol{q}} = \frac{\hat{\boldsymbol{g}}}{\boldsymbol{\gamma}}, \tag{3.25}$$

where

$$\boldsymbol{\gamma} = \|\hat{\boldsymbol{g}}\| = \|\hat{\boldsymbol{s}}\| = \sqrt{\boldsymbol{\pi}^2 + \|\underline{\boldsymbol{g}}\|^2}. \tag{3.26}$$

The solution of the minimization problem was established by Sonneville et al. [32]. Although the quaternion-metric based interpolation scheme was formulated as a minimization problem, it can also be stated as

$$\underline{\boldsymbol{s}} = \sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{s}}_k = \underline{\mathbf{0}}, \tag{3.27a}$$

$$\hat{\boldsymbol{s}} = \sum_{k=0}^{N} l_k(\eta)\hat{\boldsymbol{s}}_k = \boldsymbol{\gamma}\hat{\mathbf{1}}. \tag{3.27b}$$

Discussion 3.5.1. *The quaternion metric based interpolation can be summarized as follows:* (1) *interpolate the quaternion representations in Euclidean space* $\mathbb{D}^4$ *at the grid points and* (2) *map the result back onto the manifold using the minimization procedure. The second step of the process is a projection onto the manifold via a "closest point projection." For the quaternion metric, the closest point projection corresponds to a simple orthonormalization of the interpolated dual quaternion, as implied by eq.* (3.23).

Because the interpolation scheme operates on a set of redundant variables (4 dual numbers), singularities are avoided. This does not imply, however, that quaternion metric based interpolation can be used in all cases: the two-to-one correspondence between unit dual quaternion and motions may cause the interpolation of a periodic motion to be either discontinuous or non-periodic. Section 3.9.2 presents examples of these problems.

Quaternion metric based interpolation has been investigated by numerous authors for rotation [18, 19] and motion [18, 22, 32, 143]. Furthermore, quaternion metric based averaging has been investigated for rotation [36, 40, 124]. The scheme can be described simply as "the interpolated frame is the normalized weighted average of the nodal bi-quaternions; the weighting factors are the classical polynomial shape functions." Although the interpolated frame can be written in an explicit manner, the interpolated motion is a nonlinear function of the nodal motions.

3.6 Interpolation based on the vector metric

Using vector metric (3.11), the objective function of minimization problem (3.7) becomes

$$\boldsymbol{J}(\underline{\boldsymbol{s}}) = \sum_{k=0}^{N} l_k(\eta) \|\underline{\boldsymbol{p}}_k\|^2, \tag{3.28}$$

where $\underline{\boldsymbol{p}}_k = \boldsymbol{p}_k(\boldsymbol{\phi}_k)\bar{\boldsymbol{n}}_k$ denotes an arbitrary motion parameter vector corresponding to relative motion tensor $\boldsymbol{R}^T\boldsymbol{R}_k$. In view of definition (3.2.1), the minimization of this objective function requires the satisfaction of stationarity condition $\delta\boldsymbol{J} = \boldsymbol{0}$, leading to

$$\begin{aligned} \delta\boldsymbol{J} &= 2\sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{p}}_k^T \delta\underline{\boldsymbol{p}}_k = -2\sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{p}}_k^T \boldsymbol{T}^{-1}(-\underline{\boldsymbol{p}}_k)\delta\underline{\boldsymbol{u}} \\ &= -2\delta\underline{\boldsymbol{u}}^T \sum_{k=0}^{N} l_k(\eta)\frac{\mathrm{d}\boldsymbol{p}_k}{\mathrm{d}\boldsymbol{\phi}_k}\underline{\boldsymbol{p}}_k = \boldsymbol{0}, \end{aligned} \tag{3.29}$$

where eq. (2.55a) and the expression of tangent tensor (2.49) were used to obtain the second and third equalities, respectively. The stationarity of the objective function

requires

$$\sum_{k=0}^{N} l_k(\eta) \frac{\mathrm{d}\boldsymbol{p}_k}{\mathrm{d}\boldsymbol{\phi}_k} \underline{\boldsymbol{p}}_k = \underline{\mathbf{0}}, \tag{3.30}$$

where $\boldsymbol{\phi}_k$ is the dual angle associated with relative motion parameter vector $\underline{\boldsymbol{p}}_k$.

Consider, for instance, motion parameter vector $\underline{\boldsymbol{p}}_k = m \ \sin(\boldsymbol{\phi}_k/m)\bar{\boldsymbol{n}}_k$. Elementary trigonometric identities yield $(\mathrm{d}\boldsymbol{p}_k/\mathrm{d}\boldsymbol{\phi}_k)\ \underline{\boldsymbol{p}}_k = m/2 \ \sin(2\boldsymbol{\phi}_k/m)\bar{\boldsymbol{n}}_k$, which amounts to using a different motion parameter vector, $\underline{\boldsymbol{s}}_k = (\mathrm{d}\boldsymbol{p}_k/\mathrm{d}\boldsymbol{\phi}_k)\ \underline{\boldsymbol{p}}_k$. Equation (3.30) becomes

$$\sum_{k=0}^{N} l_k(\eta) \underline{\boldsymbol{s}}_k = \underline{\mathbf{0}}. \tag{3.31}$$

This equation defines the interpolated motion implicitly and hence, it must be solved numerically, typically via Newton iteration, to determine the interpolated motion. This approach yields good results for as long as the motion parameter vector does not encounter singularities.

The minimization of the objective function $\boldsymbol{J}$ also requires its primal part, $J = \sum_{k=0}^{N} \underline{s}_k^T \underline{s}_k$, to achieves its minimum, which implies that the Hessian of J should be positive-definite. Taking the second-order variation of J leads to

$$\delta^2 J = 2\underline{\delta\psi}^T \sum_{k=0}^{N} l_k(\eta) T^{-1}(-\underline{s}_k) \underline{\delta\psi}, \tag{3.32}$$

where $\underline{s}_k$ is the primal part of $\underline{\boldsymbol{s}}_k$. Introducing the tangent tensor defined by eq. (2.49) and identity (3.2) into eq. (3.32), yields

$$\delta^2 J = 2\underline{\delta\psi}^T \left[\sum_{k=0}^{N} l_k(\eta)(a_{0,k} I + a_{2,k} \tilde{s}_k \tilde{s}_k) \right] \underline{\delta\psi} = \underline{\delta\psi}^T H \, \underline{\delta\psi}. \tag{3.33}$$

In general, the Hessian of the problem, H is not positive-definite and hence, existence of the solution of minimization problem cannot be guaranteed. For small relative rotation motions $\underline{s}_k \to 0$ and $H \to 2I$; the Hessian now becomes positive-definite and the minimization problem can be solved.

Discussion 3.6.1. *The vector metric based interpolation have been used by numerous authors in the framework of the finite element method [13, 16, 20, 144], where interpolation of the rotation field within an element is required. As the size of the element decreases, the relative rotation within the element also decreases. Singularities are avoided and a unique solution of implicit interpolation scheme* (3.31) *exists. Similarly, interpolation schemes based on motion parameter vectors provide suitable schemes for the interpolation of motion [13].*

Interpolation scheme (3.31) *is akin to that proposed by Merlini and Morandini [31, 145] who used the Cartesian motion vector presented in section 2.6 for the interpolation. It can be described as follows: "the weighted average of the relative nodal motion parameter vectors vanishes; the weighting factors are the classical polynomial shape functions." Clearly, the scheme depends on the choice of a motion parameter vector; different choices lead to slightly different interpolated motions. The quaternion- and vector-metric based interpolation schemes are identical when the latter uses the Euler-Rodrigues motion parameter vector.*

3.7 Interpolation based on the geodesic metric

The geodesic metric is obtained by using motion parameter vector $\underline{\boldsymbol{p}} = \boldsymbol{\phi}\bar{\boldsymbol{n}}$ and hence, is a particular case of the vector metric presented in section 3.6. All the developments presented in that section apply to the geodesic metric. In particular, the stationarity condition of the objective function becomes

$$\sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{s}}_k = \underline{\mathbf{0}}, \tag{3.34}$$

which implies that the weighted sum of the relative motion vectors vanishes. The Hessian of the problem is

$$H = 2\left[I + \sum_{k=0}^{N} l_k(\eta)\chi_{2,k}\tilde{s}_k\tilde{s}_k\right]. \tag{3.35}$$

As was the case for the vector metric, the Hessian is not positive-definite, in general. When the relative motions remain small, the Hessian becomes positive-definite.

Motion parameter vector $\boldsymbol{\phi}_k\bar{\boldsymbol{n}}$ is singularity free in the range $\theta_k \in [-\pi, \pi)$ and Newton iteration provides an efficient solution of implicit interpolation scheme (3.34) for $|\theta_k| < \pi$. As $|\theta_k|$ approaches π, Newton iteration fails to converge. Buss and Fillmore [34] proposed a fixed point iteration method for rotation interpolation that was later extended by Kavan et al. [22] for motion interpolation. These approaches also fail to converge as $|\theta_k|$ approaches π, a situation commonly encoutered in global interpolation.

Discussion 3.7.1. *As an example of interpolation scheme* (3.34)*, consider interpolation between two grid points using linear shape functions,* $(1-\eta)/2\text{axial}[\log(\boldsymbol{R}^T\boldsymbol{R}_1)]+(1+\eta)/2\text{axial}[\log(\boldsymbol{R}^T\boldsymbol{R}_2)] = \underline{\boldsymbol{0}}$*. It is verified easily that the explicit interpolation formula is*

$$\boldsymbol{R} = \boldsymbol{R}_1 \exp[\tilde{\boldsymbol{k}}(1+\eta)/2] = \boldsymbol{R}_2 \exp[-\tilde{\boldsymbol{k}}(1-\eta)/2], \tag{3.36}$$

where $\underline{\boldsymbol{q}}_{12} = \text{axial}[\log(\boldsymbol{R}_1^T\boldsymbol{R}_2)]$*. The interpolation leads to the geodesic on* $\mathcal{SO}_3$ *passing through* $\boldsymbol{R}_1$ *and* $\boldsymbol{R}_2$*.*

The rotation portion of interpolation scheme (3.36) *is the spherical linear interpolation (SLERP) first proposed by Shoemake [26] for computer animation applications. When expressed in terms of dual quaternion, interpolation scheme* (3.36) *becomes the Screw Linear Interpolation (ScLERP) developed by Kavan et al. [22, 143, 146]. Because it is limited to local interpolation, scheme* (3.34) *has been used in the finite element framework. Borri and Bottasso [28] and Ghosh and Roy [60] developed constant-strain element based on SLERP. Merlini and Morandini [67, 147] proposed motion interpolation scheme* (3.34) *for beams, plates, and shells. The same approached was followed by Sander [68, 148] and Sonneville et al. [29, 69]. Buss and Fillmore [34] were the first to recast rotation interpolation schemes as minimization problems using the geodesic metric. More recently, Sander [68, 148] proposed geodesic metric based finite elements for which the motion interpolation was obtained by solving optimization problem* (3.28) *directly. Numerous authors [33, 37, 39, 40] have also cast the problem of averaging of rotations through a minimization approach.*

3.8 The incremental motion and curvature fields

Sections 3.4, 3.5, 3.6, and 3.7 have focused on the interpolation of motion based on the matrix, quaternion, vector, and geodesic metrics, respectively. It is remarkable that all these approaches to the interpolation of motion can be recast in the form of an implicit interpolation formula

$$\sum_{k=0}^{N} l_k(\eta)\underline{\boldsymbol{s}}_k(\eta) = \underline{0}. \tag{3.37}$$

For the matrix, quaternion, and geodesic metric based interpolation schemes, relative motion parameter vectors $\underline{\boldsymbol{s}}_k$ are the linear, Euler-Rodrigues, and Cartesian motion parameter vectors associated with relative motion tensors $\boldsymbol{S}_k(\eta) = \boldsymbol{R}^T(\eta)\boldsymbol{R}_k$, respectively, see section 2.6. A linearization of eq. (3.37) leads to

$$\underline{\delta\boldsymbol{u}} = [\sum_{i=0}^{N} l_i \boldsymbol{T}^{-1}(-\underline{\boldsymbol{s}}_i)]^{-1} \sum_{k=0}^{N} l_k \underline{\delta\boldsymbol{u}}_k, \tag{3.38}$$

where identity (3.2) is introduced. Similarly, taking a derivative of eq. (3.37) leads to

$$\underline{\boldsymbol{k}} = [\sum_{i=0}^{N} l_i \boldsymbol{T}^{-1}(-\underline{\boldsymbol{s}}_i)]^{-1} \sum_{k=0}^{N} l'_k \underline{\boldsymbol{s}}_k. \tag{3.39}$$

The interpolations in eqs. (3.38) and (3.39) can be simplified in three cases listed in the following paragraphs.

- The dual SLERP in eq. (3.36). Taking a linearization and derivative with

eq. (3.36) leads to

$$\underline{\Delta \boldsymbol{u}}(\eta) = \left[\boldsymbol{I}_3 - \frac{1+\eta}{2}\boldsymbol{T}(-\underline{\boldsymbol{s}}_0)\boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_{01}) \quad \frac{1+\eta}{2}\boldsymbol{T}(-\underline{\boldsymbol{s}}_0)\boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_{01})\right]\underline{\Delta \hat{\boldsymbol{u}}} \overset{\text{def}}{=} \boldsymbol{L}_u \underline{\Delta \hat{\boldsymbol{u}}}, \tag{3.40a}$$

$$\underline{\boldsymbol{k}}(\eta) = \frac{1}{2}\underline{\boldsymbol{s}}_{01}, \tag{3.40b}$$

where $\underline{\Delta \hat{\boldsymbol{u}}}^T = \{\underline{\Delta \boldsymbol{u}}_0^T, \ldots, \underline{\Delta \boldsymbol{u}}_N^T\}$ and identity (2.55b) is used to yield the equality. Linearization of eq. (3.40b) yields

$$\Delta\underline{\boldsymbol{k}}(\eta) = \frac{1}{2}\left[-\boldsymbol{T}^{-1}(-\underline{\boldsymbol{s}}_{01}) \quad \boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_{01})\right]\underline{\Delta \hat{\boldsymbol{u}}} \overset{\text{def}}{=} \boldsymbol{L}_v \underline{\Delta \hat{\boldsymbol{u}}}. \tag{3.41}$$

- Quaternion metric based interpolation

$$\underline{\Delta \boldsymbol{u}}(\eta) = \frac{1}{\boldsymbol{\sigma}}\sum_{k=0}^{N} l_k(\eta)\boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_k)\underline{\Delta \hat{\boldsymbol{u}}} \overset{\text{def}}{=} \boldsymbol{L}_u \underline{\Delta \hat{\boldsymbol{u}}}, \tag{3.42a}$$

$$\underline{\boldsymbol{k}}(\eta) = \frac{1}{\boldsymbol{\sigma}}\sum_{k=0}^{N} l'_k(\eta)\underline{\boldsymbol{s}}_k = \sum_{k=0}^{N} l'_k \underline{\boldsymbol{r}}_k, \tag{3.42b}$$

where $\boldsymbol{\sigma} = \sum_{k=0}^{N} l_k(\eta)\boldsymbol{s}_{0k}$, $\boldsymbol{s}_{0k} = \sqrt{\boldsymbol{1} - \underline{\boldsymbol{s}}_k^T \underline{\boldsymbol{s}}_k/4}$, and $\underline{\boldsymbol{r}}_k = \underline{\boldsymbol{s}}_k/\boldsymbol{\sigma}$. Linearization of eq. (3.42b) yields

$$\begin{aligned}\Delta\underline{\boldsymbol{k}}(\eta) = &-(\frac{1}{\boldsymbol{\sigma}}\sum_{k=0}^{N} l'_k \boldsymbol{s}_{0k} I - \frac{1}{2}\tilde{\boldsymbol{k}})\boldsymbol{L}_u \underline{\Delta \hat{\boldsymbol{u}}} + \frac{1}{\boldsymbol{\sigma}}\sum_{k=0}^{N}[l'_k \boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_k) + \frac{1}{4} l_k \underline{\boldsymbol{k}}\underline{\boldsymbol{s}}_k^T]\underline{\Delta \boldsymbol{u}}_k \\ &\overset{\text{def}}{=} \boldsymbol{L}_v \underline{\Delta \hat{\boldsymbol{u}}}.\end{aligned} \tag{3.43}$$

- In the spectral formulation, the grid points are co-located with quadrature

points, as discussed in appendix (B.4). For this type of problem, the curvature vector at the quadrature points points η_j, $j = 0, 2, \ldots, N$ have simple expressions

$$\underline{\boldsymbol{k}}(\eta_j) = \sum_{k=0}^{N} l'_k(\eta_j)\underline{\boldsymbol{s}}_{jk}. \tag{3.44}$$

Linearization of eq. (3.44) yields

$$\Delta\hat{\underline{\boldsymbol{k}}} = \boldsymbol{B}\,\Delta\underline{\hat{\boldsymbol{u}}}, \tag{3.45}$$

where $\hat{\underline{\boldsymbol{k}}}^T = \{\underline{\boldsymbol{k}}_0^T, \ldots, \underline{\boldsymbol{k}}_N^T\}$ and matrix $\boldsymbol{B}$ is composed of $(N+1)\times(N+1)$ sub-matrices of size 6×6. Notation $[\cdot]_{jk}$ indicates the sub-matrix of size 6×6 at location (j,k), $j,k = 0,\ldots,N$,

$$[\boldsymbol{B}]_{jk} = \begin{cases} \tilde{\boldsymbol{k}}_j - \sum_{i=0}^{N} l'_i(\eta_j)\ \boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_{ji}), & \text{for } k = j, \\ l'_k(\eta_j)\boldsymbol{T}^{-1}(\underline{\boldsymbol{s}}_{jk}), & \text{for } k \neq j. \end{cases} \tag{3.46}$$

For all the first two cases, the following extended notations are introduced when used in static or dynamic problem

$$\underline{\Delta u} = \mathcal{L}_u\hat{\underline{\Delta}}\underline{u}, \tag{3.47a}$$

$$\Delta\underline{\kappa} = \mathcal{L}_v\hat{\underline{\Delta}}\underline{u}, \tag{3.47b}$$

where each dual submatrix of size 3×3 in $\boldsymbol{L}_u$ and $\boldsymbol{L}_v$ are recast to submatrix of size 6×6 according to transformation (2.60). Similarly, extended notation of eq. (3.45)

in spectral formulation is

$$\Delta\hat{\underline{\kappa}} = \mathcal{B}\,\hat{\Delta}\underline{u}. \tag{3.48}$$

3.9 Numerical examples

A set of numerical examples will be presented to validate the proposed interpolation approaches. The use of Chebyshev spectral functions, Fourier spectral functions, and cubic B-spline interpolation functions will be demonstrated. Many of the interpolation schemes described in sections 3.4 to 3.7 have been used for finite element applications. Within that framework, local interpolation is sufficient and typically, Lagrange's polynomials are used as basis functions [149, 150]. Numerous numerical examples can be found in the literature cited in sections 3.4 to 3.7 and will not repeated here.

3.9.1 Interpolation of non-periodic motion

To validate the various interpolation schemes, a simple example is presented. Figure 3.2 shows a three-dimensional curve described by parametric equation $\underline{r}(\theta) = \rho(\theta)\cos\theta\bar{\imath}_1 + \rho(\theta)\sin\theta\bar{\imath}_2 + p(\theta)\theta\bar{\imath}_3$, where $\rho(\theta) = 1 + 1/3[2\theta/T - 1]^2 + 1/5\ [1 - \cos(11\ \theta/T)]$ and $p(\theta) = 2 + 2/11\ \theta/T + 1/7\ \sin(7\ \theta/T)$ are the radius and pitch of the curve, respectively, and $T = 3\pi$. At an arbitrary point $\mathbf{B}$ of the curve, Frenet-Serret's orthonormal triad $\mathcal{B} = (\bar{t}, \bar{n}, \bar{b})$ is defined by the unit tangent, normal, and binormal vectors denoted $\bar{t}$, $\bar{n}$, and $\bar{b}$, respectively. Rotation tensor $R(\theta)$ brings inertial basis $\mathcal{I} = (\bar{\imath}_1, \bar{\imath}_2, \bar{\imath}_3)$ to $\mathcal{B}$. Point $\mathbf{B}$ and basis $\mathcal{B}$ define a frame $\mathcal{F}_B = [\mathbf{B}, \mathcal{B}]$.

Motion tensor $\boldsymbol{R}(\eta)$ brings inertial frame $\mathcal{F}_I = [\mathbf{O}, \mathcal{I}]$ to $\mathcal{F}_B$. The components of the tangent and curvature vectors resolved in basis $\mathcal{B}$ are $(R^T \underline{r}')^T = \{\|\underline{r}'\|, 0, 0\}$ and $\underline{\omega} = \text{axial}(R^T R') = \{\eta, 0, \kappa\}^T$, where notation $(\cdot)'$ indicates a derivative with respect to θ.

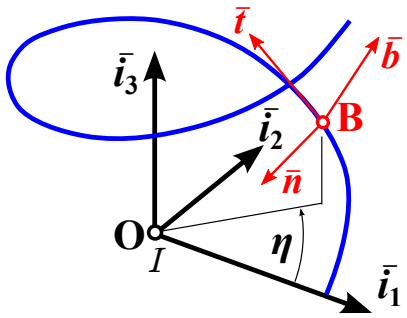

Figure 3.2: Configuration of the curve.

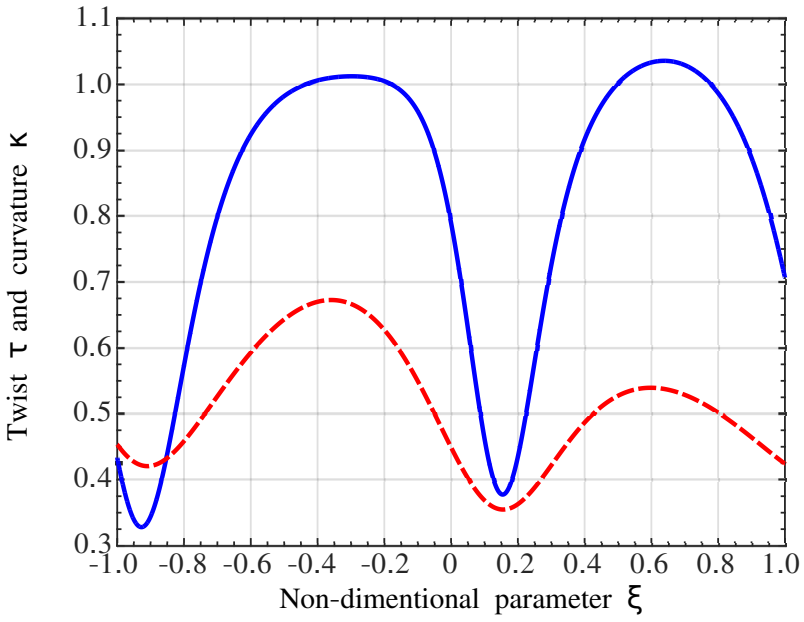

Figure 3.3: The twist (solid line) and curvature (dashed line) of the helix.

The twist η and curvature κ of the curve, defined as

$$\eta = \|\underline{r}'\| \frac{\underline{r}'^T \tilde{r}'' \underline{r}'''}{\|\tilde{r}' \underline{r}''\|^2}, \quad \text{and} \kappa = \frac{\|\tilde{r}' \underline{r}''\|}{\|\underline{r}'\|},$$

are shown in fig. 3.3 a functions of non-dimensional parameter $\eta = \theta/T$.

Motion field $\boldsymbol{R}(\theta)$ was interpolated based on the proposed schemes using Chebyshev spectral functions, see appendix B.6, with an increasing number of grid points, $N = 8, 16, 32, 64, 128$. To assess the accuracy of the interpolation, each grid interval $[\eta_k,\ \eta_{k+1}]$ was divided into 50 subintervals and notation $(\cdot)_j, j = 0, 1, \cdots, 50N$ indicates quantities evaluated at the sub-grid points. Interpolated rotations and curvatures were evaluated and compared to their exact counterpart, denoted $(\cdot)^{\mathrm{e}}$. The following measures were selected to quantify errors in orientation and curvature,

$$E_R = \sum_{j=0}^{50N} \frac{\mathrm{dist}_{\mathrm{m}}(R_j - R_j^{\mathrm{e}})}{50N}, \quad E_\omega = \sum_{k=0}^{50N} \frac{\|\underline{\omega}_j - \underline{\omega}_j^{\mathrm{e}}\|}{50N}. \tag{3.49}$$

Figure 3.4 shows the orientation and curvature error measures defined by eq. (3.49) versus the number of grid points on a logarithmic plot for two interpolation schemes: interpolation scheme (3.7) with matrix metric (3.8) and interpolation scheme (3.7) with quaternion metric (3.10).

Next, the position and tangent vector errors were quantified using the following measures,

$$E_r = \sum_{j=0}^{50N} \frac{\|\underline{r}_j - \underline{r}_j^{\mathrm{e}}\|}{50N}, \quad E_v = \sum_{k=0}^{50N} \frac{\|\underline{r}' - \underline{r}_j^{\mathrm{e}\prime}\|}{50N}. \tag{3.50}$$

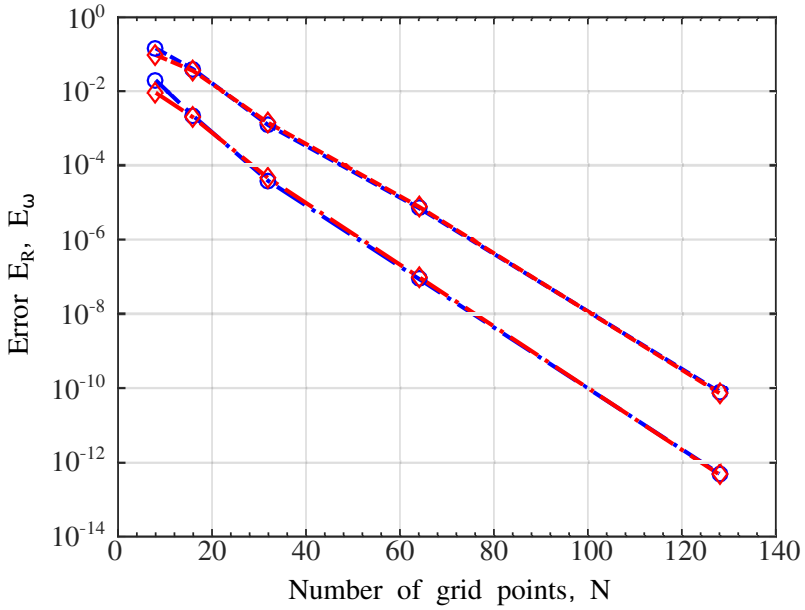

Figure 3.4: Chebyshev spectral interpolation error for orientation and curvature vector; dist_m: ($\circ$), dist_q: ($\diamond$). Orientation error, E_R, dashed-dotted lines; curvature error, E_ω, dashed lines.

Figure 3.5 shows the position and tangent vector error measures defined by eq. (3.50) versus the number of grid points on a logarithmic plot for the same two interpolation schemes. Figures 3.4 and 3.5 show that all error measures converge exponentially, as expected of spectral interpolation. The curvature based interpolation is more accurate than the matrix and quaternion metric based approaches, whose accuracies are comparable.

Next, the same problem was treated using the cubic B-spline interpolation functions described in appendix B.8. An increasing number of grid points was used, $N = 8, 16, 32, 64, 128$. Figure 3.6 shows the orientation and curvature error measures defined by eq. (3.49) versus the number of grid points on a logarithmic plot for the same two interpolation schemes. Figure 3.7 shows the corresponding position and tangent vector error measures defined by eq. (3.50).

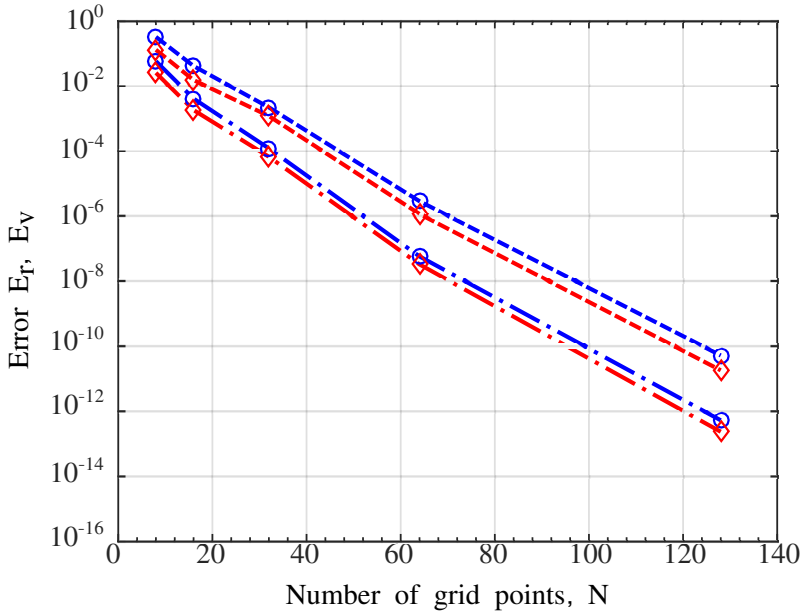

Figure 3.5: Chebyshev spectral interpolation error for position and tangent vectors; dist_m: ($\circ$), dist_q: ($\diamond$). Position vector error, E_r, dashed-dotted line; tangent vector error, E_v, dashed line.

Because the cubic B-spline interpolation functions are local, the exponential convergence property observed with the Chebyshev spectral functions no longer holds. Figures 3.6 and 3.7 show that the convergence rates are 4.5 for the orientation and position vector errors and 3.5 for curvature and tangent vector errors, for both matrix and quaternion metric based approaches. Convergence rates are one order higher when the curvature based interpolation is used. The accuracies of the matrix and quaternion metric based approaches are comparable.

3.9.2 Interpolation of periodic motion

Many flexible multibody systems feature a periodic response. One way of obtaining these periodic solutions is to implement Fourier spectral methods that impose periodicity of the solution through the use of Fourier basis functions. A pre-

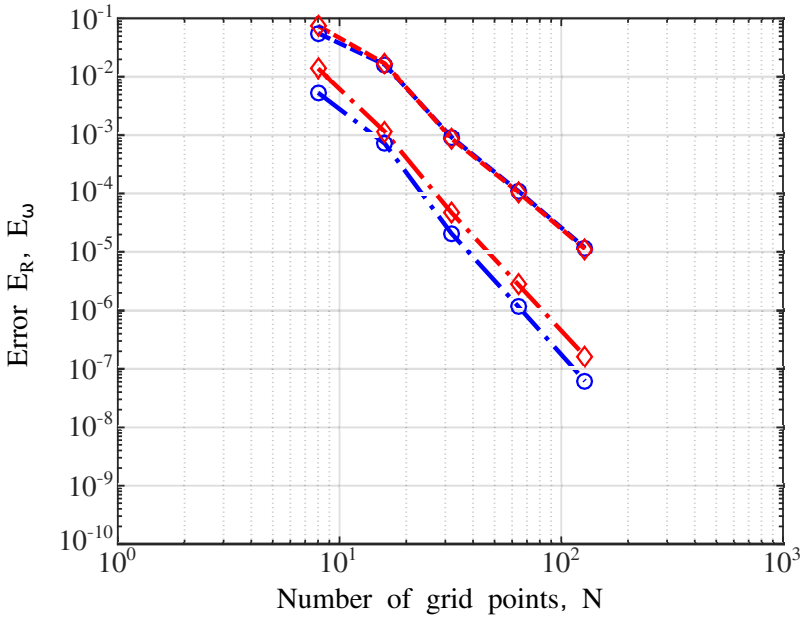

Figure 3.6: Cubic B-spline interpolation error for orientation and curvature vector; dist_m: ($\circ$), dist_q: ($\diamond$). Orientation error, E_R, dashed-dotted line; curvature error, E_ω, dashed line.

requisite of this approach is the ability to interpolate the solution over one complete period based on its value at grid points. To achieve the exponential convergence promised by Fourier spectral methods, the interpolated field must be continuous and periodic.

Clearly, vector metric based interpolation schemes are not suitable for this problem because all motion parameter vectors present singularities when interpolating large rotations [127]. Although quaternions present no singularities for large rotations, quaternion metric based interpolation schemes are equally unsuitable for periodic problems. Indeed, the two-to-one mapping between quaternion representations and motion produce interpolated fields that are either continuous or periodic, but not both. The two-to-one mapping arises from the fact that quaternions $+\hat{e}$ and $-\hat{e}$ represent the same motion.

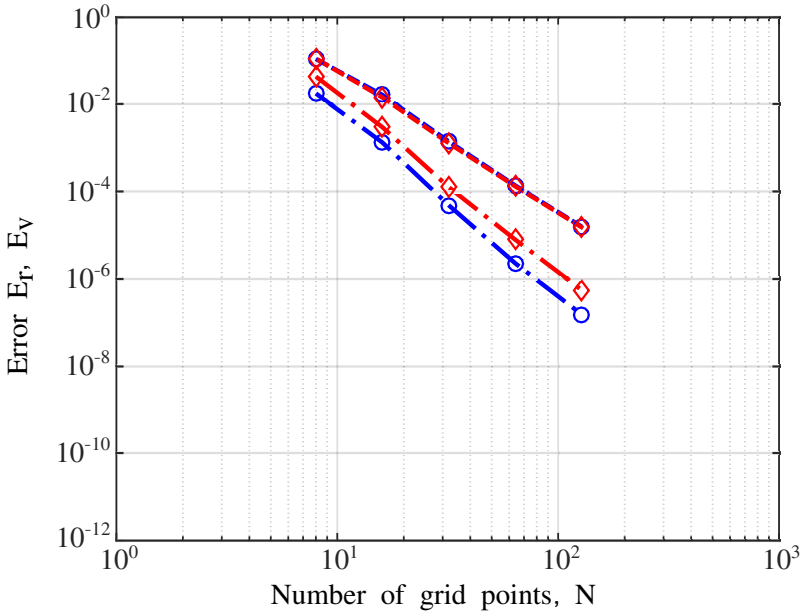

Figure 3.7: Cubic B-spline interpolation error for position and tangent vectors; dist_m: ($\circ$), dist_q: ($\diamond$). Position vector error, E_r, dashed-dotted line; tangent vector error, E_v, dashed line.

To illustrate this insidious problem, consider the periodic rotation field described by the following time functions of the Euler angles: precession $\phi = \Omega t$, nutation $\theta = \pi/13 \sin(2\Omega t)$, and spin $\psi = \pi/17\ [1 - \cos(3\Omega t)]$, where $\Omega = 9$ rad/s. The algorithm of Klumpp and Shepperd [116, 117] was used to extract the quaternion representation of this rotation field and fig. 3.8 shows that components $e_2(t)$ and $e_4(t)$ present a discontinuity at $\phi \approx 3\pi/2$. This discontinuity stems from the extraction algorithm: the sign of the extracted quaternion is selected to avoid singularities, not discontinuities. Note that components $e_2(t)$ and $e_4(t)$ are discontinuous but periodic.

In an attempt to overcome this problem, Klumpp and Shepperd's algorithm can be modified slightly to prevent the change of sign of the quaternion, resulting in the second set of results shown in fig. 3.8. Discontinuities are eliminated but

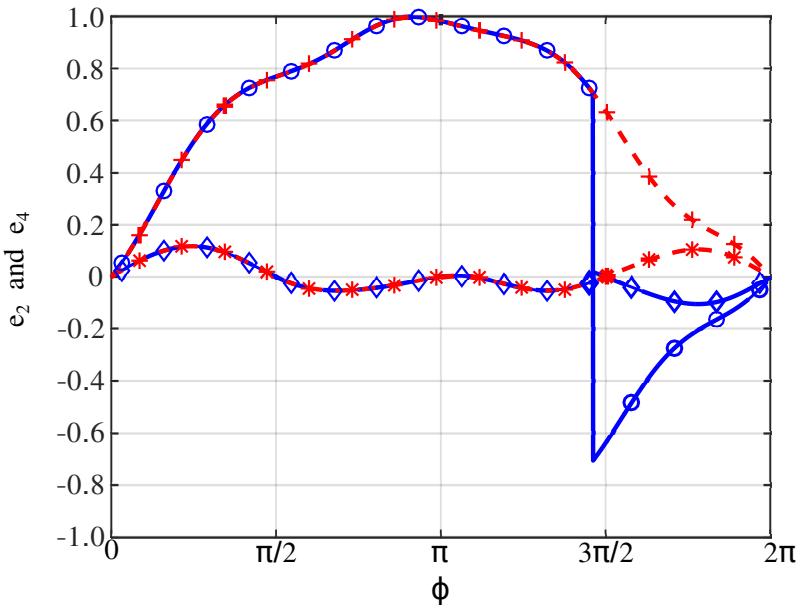

Figure 3.8: Quaternion components e_2 and e_4 extracted from a periodic rotation field. Discontinuous e_2: ($\diamond$), and e_4: ($\circ$). Non-periodic e_2: ($*$), and e_4: ($+$).

quaternion components $e_2(t)$ and $e_4(t)$ now become non-periodic because their time derivatives at $\phi = 0$ and 2π differ.

This lack of periodicity stems the definition of quaternions, $e_0(t) = \text{scal}(\hat{e}) = \cos\theta(t)/2$ and $\underline{e}(t) = \text{vec}(\hat{e}) = \bar{n}\sin\theta(t)/2$: although angle $\theta(t)$ is a periodic function of period T, functions $\cos\theta(t)/2$ and $\sin\theta(t)/2$ are periodic functions but of period $2T$. This observation hints at another potential solution: the period of the problem is taken to be $2T$ rather that T, leading to interpolated fields that are continuous and periodic.

Figure 3.9 shows interpolated quaternion component $e_4(t)$ when using different interpolation strategies with 63 grid points. First, when the non-periodic quaternion field is interpolated, a sharp peak occurs near $\phi = 2\pi$, resulting from the discontinuity of the velocity field. Second, when the discontinuous quaternion field

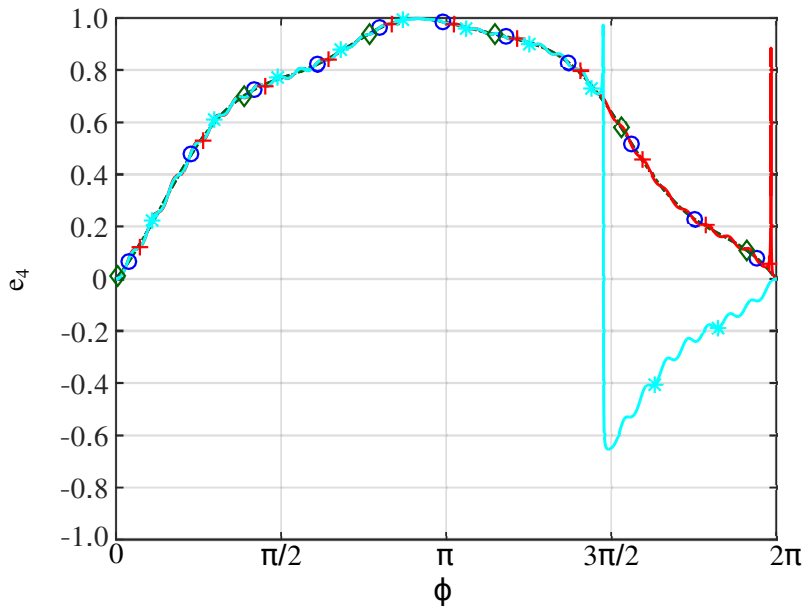

Figure 3.9: Quaternion component e_4. Exact solution: ($\circ$); interpolation of non-periodic field on T: ($+$); interpolation of discontinuous field on T: ($*$); interpolation on $2T$: ($\diamond$).

is interpolated, the interpolation scheme capture the discontinuity, but oscillations result. Finally, if the interpolation scheme is applied to the problem with a period of $2T$, the interpolated field matches its exact counterpart closely.

Figure 3.10 shows the corresponding results for angular velocity of precession. For the interpolation of non-periodic or discontinuous quaternions, Gibbs' phenomenon arises resulting in violent oscillations in the velocity field. If the interpolation scheme is applied to the problem with a period of $2T$, a smooth velocity field is obtained.

While the interpolation based on a period of $2T$ seems to yield reasonable results, it comes with a considerable decrease in accuracy. Figure 3.11 illustrates this point: the convergence of the matrix and quaternion metric based interpolations are contrasted, showing the clear superiority of the former approach.

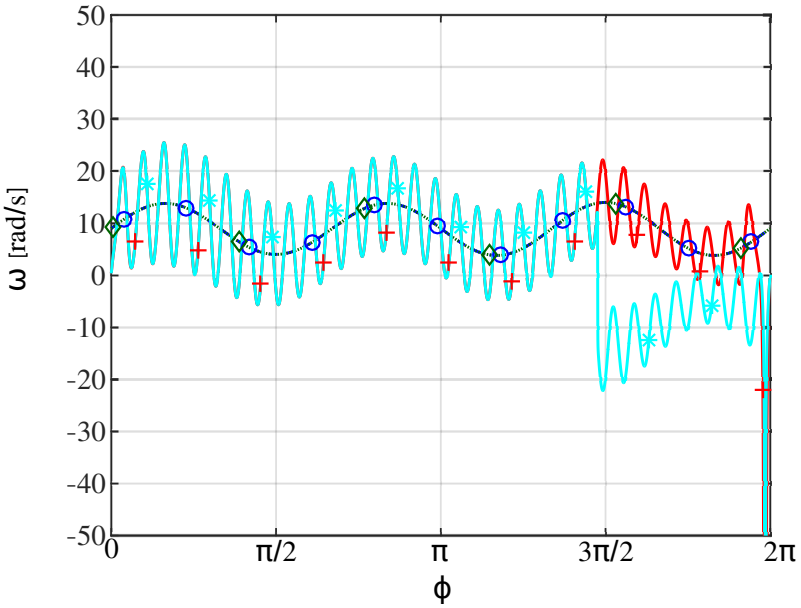

Figure 3.10: Velocity component $\mathcal{R}^T\underline{\dot{r}}$. Exact solution: ($\circ$); interpolation of non-periodic field on T: ($+$); interpolation of discontinuous field on T: ($*$); interpolation on $2T$: ($\diamond$).

Clearly, the best scheme for the interpolation of periodic rotation fields is the matrix metric based scheme. To further illustrate approach, a periodic motion field is interpolated. The rotation is described by the following time functions of the Euler angles: precession $\phi = t$, nutation $\theta = \pi/11\sin(5t)$, and spin $\psi = \pi/7[1-\cos(3t)]$. The position vector is described by its three components, $r_1(t) = \cos(t)+1/7\sin(3t)$, $r_2(t) = \sin t + 1/9[1-\cos(5t)]$ and $r_3(t) = 1/7\sin(7t) + 1/9[1-\cos(9t)]$.

First, this periodic motion was interpolated using Fourier spectral basis functions with an increasing number of grid points, $N = 4, 8, 16, 32, 64, 128$. Figure 3.12 shows the orientation and angular velocity error measures defined by eq. (3.49) versus the number of grid points on a logarithmic plot for interpolation scheme (3.7) with matrix metric (3.8). Figure 3.13 shows the corresponding results for the position and velocity vectors error measures defined by eq. (3.50).

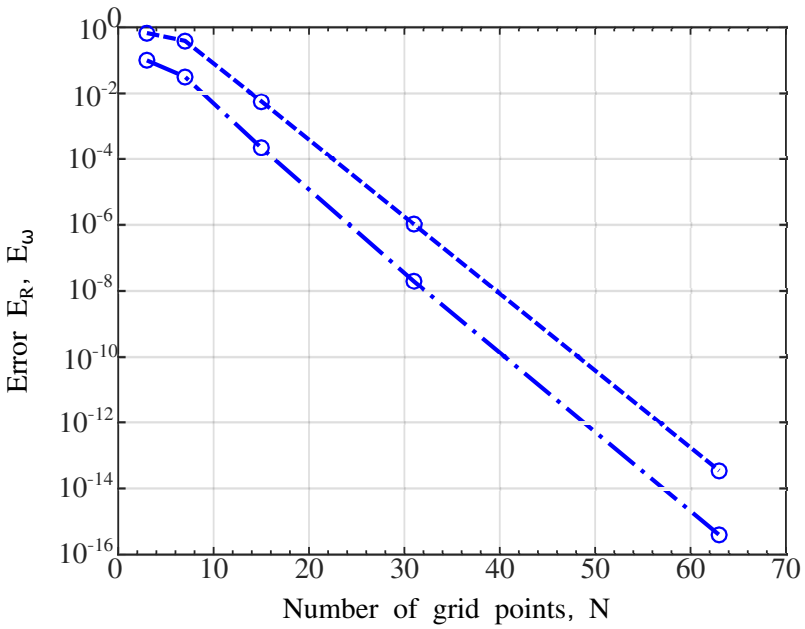

Figure 3.11: Fourier spectral interpolation error; dist_m: ($\circ$), dist_q: ($\diamond$); error of rotation E_R: dashed-dotted line, error of angular velocity E_ω: dashed line.

Next, the periodic motion was interpolated using cubic B-spline basis functions with an increasing number of grid points, $N = 4, 8, 16, 32, 64, 128$. Figure 3.14 shows the orientation and angular velocity error measures for the same interpolation schemes. Figure 3.15 shows the corresponding results for the position and velocity vectors error measures. As was observed for non-periodic problems, a convergence rates of 4.5 is observed for the orientation and displacement errors whereas a convergence rate of 3.5 is achieved for the linear and angular velocity vectors when using the matrix metric approach. Convergence rates are one order higher when the velocity based interpolation is used.

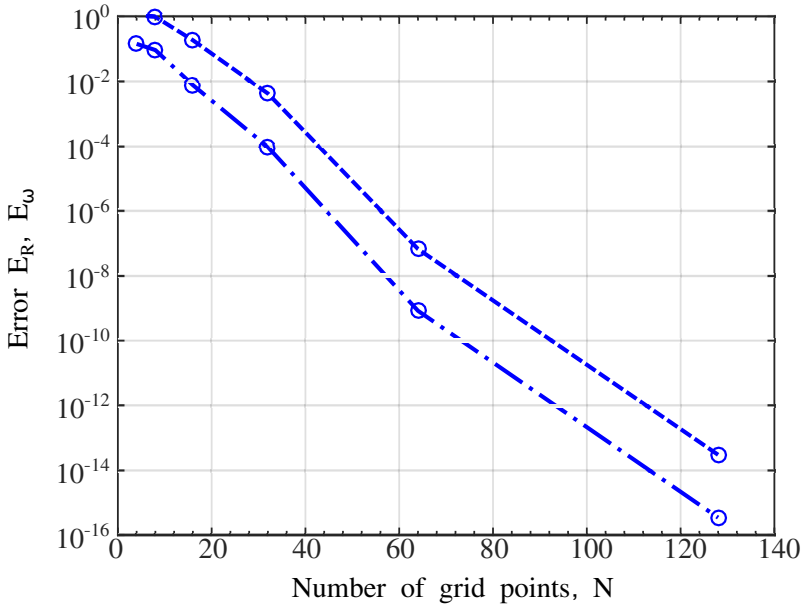

Figure 3.12: Fourier spectral interpolation error for orientation and angular velocity. Error of rotation E_R: dashed-dotted line, error of angular velocity E_ω: dashed line.

3.10 Summaries and conclusions

The classical interpolation schemes formulated for the Euclidean space have been recast as the minimization of weighted distance measures. It then becomes possible to use the same schemes for the motion manifold, provided that adequate metrics of this manifold are defined. Four metrics were defined: the matrix, quaternion, vector, and geodesic metrics. The advantages and drawbacks of each scheme were discussed and the relationship of the derived schemes with schemes that have appeared in the literature were clarified.

The problem of global interpolation was given special attention. Global interpolation schemes must be able to handle motions of arbitrary magnitude, in contrast with local interpolation schemes that can deal with small relative motions only. Schemes based on the matrix metric and on the interpolation of derivatives

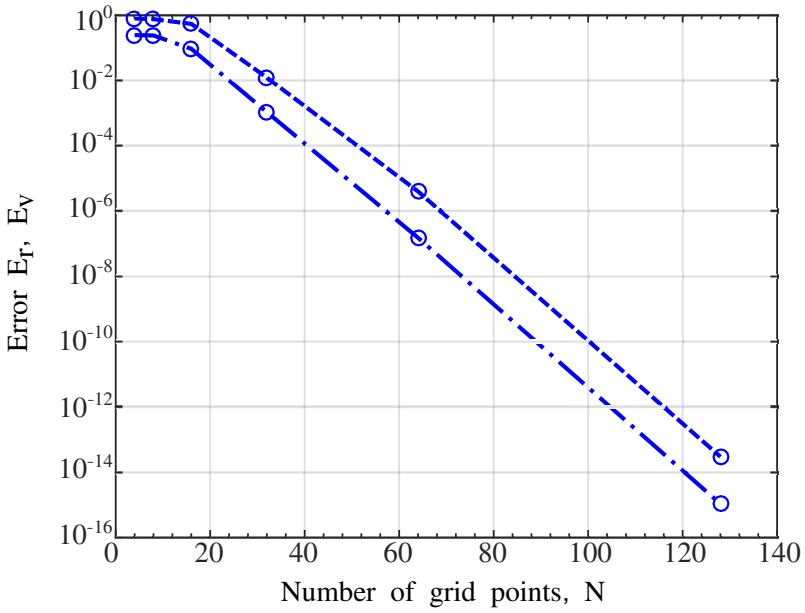

Figure 3.13: Fourier spectral interpolation error for position and velocity vectors. Error of position E_r: dashed-dotted line, error of velocity E_v: dashed line.

were found to be suitable for global interpolation problems; because they encounter singularities in the presence of large motions, the other schemes failed to provide reasonable solutions. Schemes based on the quaternion metric were found to be suitable for interpolation of non-periodic motion but not for that of periodic motion. The two-to-one mapping between quaternions and motions produces interpolated fields that are either continuous or periodic, but not both. Although this issue can be resolved by interpolating on a double period, this practice results in a severe reduction of accuracy. All the schemes presented in this paper are suitable for local interpolation problems, such as those found in the finite element method. The accuracy of the schemes based on the four metric were found to be similar, provided that the same basis functions and grid points are used.

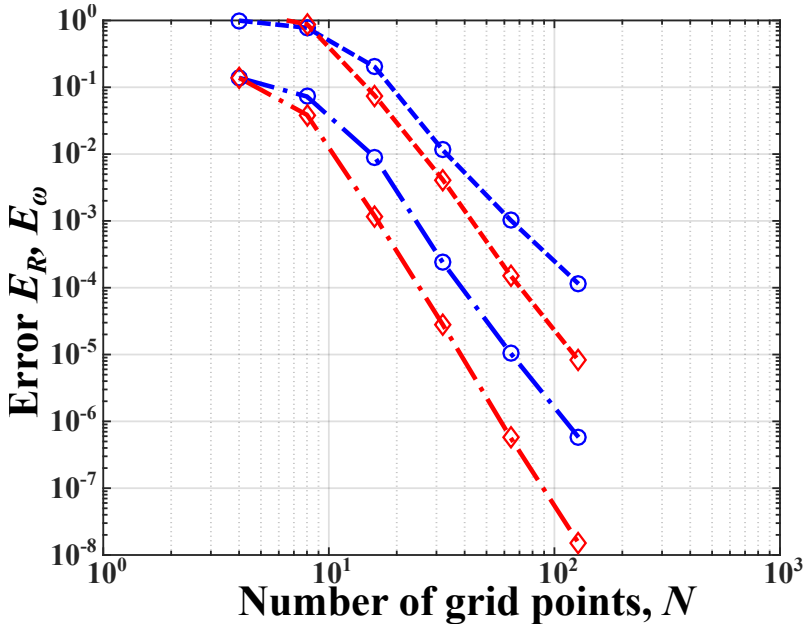

Figure 3.14: Cubic B-spline interpolation error for orientation and angular velocity. Orientation error, E_R, dashed-dotted line; angular velocity error, E_ω, dashed line.

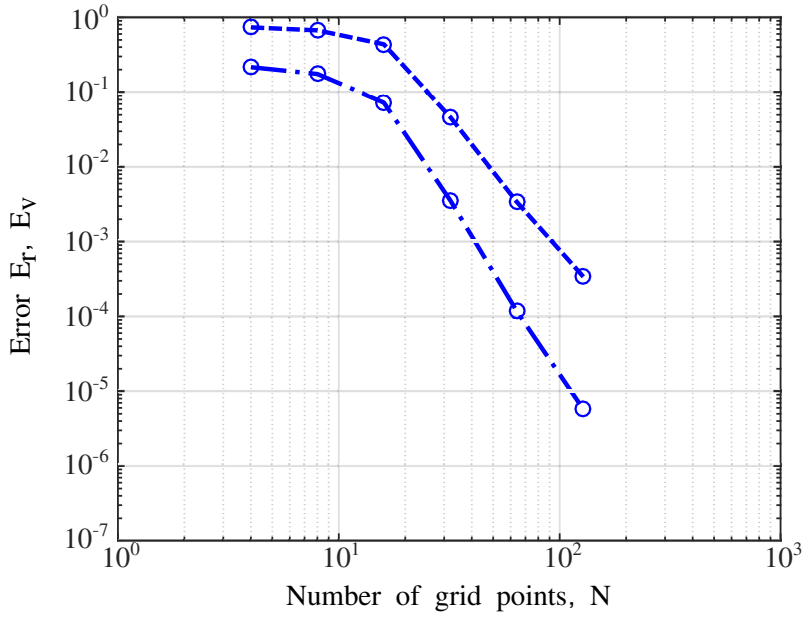

Figure 3.15: B-spline interpolation error for position and velocity vectors. Position error, E_r, dashed-dotted line; velocity error, E_v, dashed line.

Chapter 4: Galerkin Method in Spatial Domain: Beam Formulation

This chapter focuses on the formulation of geometrically exact beam, which is an application of the continuous Galerkin method to a two-point boundary value problem in spatial domain.

4.1 Kinematics of the problem

Figure 4.1 depicts an initially curved and twisted beam with a cross-section of arbitrary shape. The volume of the beam is generated by sliding the cross-section along the reference line of the beam, which is defined by an arbitrary curve in space denoted $\mathcal{C}$. Curvilinear coordinate s defines the arc-length of $\mathcal{C}$. Point $\mathbf{B}$ is located at the intersection of the reference line with the plane of the cross-section.

Frame $\mathcal{F}_0 = \left[\mathbf{B}, \mathcal{B}_0 = (\bar{b}_1, \bar{b}_2, \bar{b}_3)\right]$ defines the cross-section in the reference configuration. The plane of the cross-section is determined by two mutually orthogonal unit vectors, $\bar{b}_2$ and $\bar{b}_3$. The reference point and orientation of the cross-section change as it slides along curve $\mathcal{C}$ and hence, frame $\mathcal{F}_0$ is a function of arc-length coordinate s. In the deformed configuration, the cross-section is defined by frame $\mathcal{F} = \left[\mathbf{B}, \mathcal{B} = (\bar{B}_1, \bar{B}_2, \bar{B}_3)\right]$. The motion tensors that bring frame $\mathcal{F}_I$ to $\mathcal{F}_0$ and $\mathcal{F}_I$

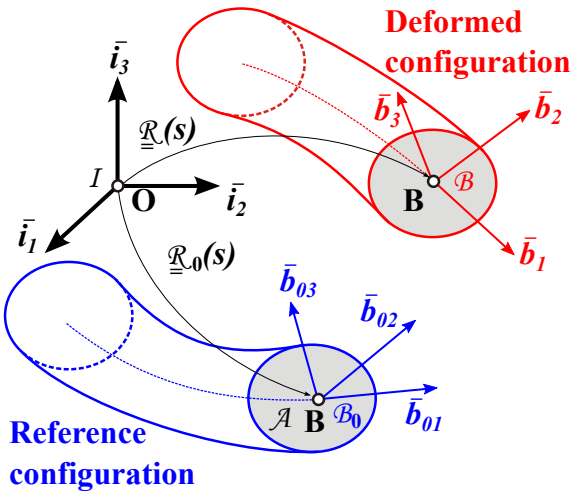

Figure 4.1: Configurations of a geometrically exact beam.

to $\mathcal{F}$ are

$$\boldsymbol{R}_0(s) = R_0 + \epsilon\tilde{r}_0 R_0, \tag{4.1a}$$

$$\boldsymbol{R}(s) = R + \epsilon\tilde{r} R, \tag{4.1b}$$

respectively. Therein $\underline{r}_0$ and $\underline{r}$ are the position vector of material point **B** in the initial and deformed configurations; rotation tensor R_0 and R bring basis $\mathcal{I}$ to basis $\mathcal{B}_0$ and to basis $\mathcal{B}$, respectively.

The beam's dual curvature vector in its initial and deformed configuration are

$$\tilde{\boldsymbol{k}}_0 = \boldsymbol{R}_0^T \boldsymbol{R}_0' = \tilde{k}_0 + \epsilon\tilde{t}_0, \tag{4.2a}$$

$$\tilde{\boldsymbol{k}} = \boldsymbol{R}^T \boldsymbol{R}' = \tilde{k} + \epsilon\tilde{t}, \tag{4.2b}$$

respectively; Therein notation $(\cdot)'$ indicates a derivative with respect to coordinate

s; $\underline{t}_0 = R_0^T \underline{r}_0'$ and $\underline{k}_0 = \text{axial}(R_0 R_0')$ are the tangent and curvature vector in the initial configuration, respectively; $\underline{t} = R^T \underline{r}'$ and $\underline{k} = \text{axial}(RR')$ are the tangent and curvature vector in the deformed configuration, respectively. The sectional strain measures [49] of the beam are defined as the differences between the curvature vectors in the deformed and reference configurations

$$\underline{\boldsymbol{e}} = \underline{\boldsymbol{k}} - \underline{\boldsymbol{k}}_0. \tag{4.3}$$

4.2 Interpolation of rigid-body motion

Consider a beam element with a non-dimensional coordinate $\eta \in [-1, 1]$ along its axis; the end points of the beam are located at $\eta = \pm 1$. In the spectral formulation, the nodes are located at the GLL points, $\eta_k^{GLL}, k = 0, \ldots, N$. The N Gauss points are denoted as $\mu_i, i = 0, \ldots, N-1$. The configuration of the beam is defined by the nodal values of the motion tensors, denoted as $\boldsymbol{R}_k$, $k = 0, \ldots, N$. The interpolated motion tensor at point η is denoted as $\boldsymbol{R}$, as indicated in fig. 4.2. The relative motion tensor from η to η_k^{GLL} and from η_j^{GLL} to η_k^{GLL} are $\boldsymbol{R}^T \boldsymbol{R}_k$ and $\boldsymbol{R}_j^T \boldsymbol{R}_k$, $j, k = 0, \ldots, N$, respectively. The motion parameter vectors associated with relative motion tensors, $\boldsymbol{R}^T \boldsymbol{R}_k$ and $\boldsymbol{R}_j^T \boldsymbol{R}_k$, are denoted as $\underline{\boldsymbol{s}}_k$ and $\underline{\boldsymbol{s}}_{jk}$, respectively; the linear, Euler-Rodrigues, and Cartesian motion parameter vectors will be used in this work. Let arrays $\underline{\Delta \hat{\boldsymbol{u}}}^T = \{\underline{\Delta \boldsymbol{u}}_0^T, \ldots, \underline{\Delta \boldsymbol{u}}_N^T\}$ and $\Delta \hat{\underline{\boldsymbol{k}}}^T = \{\Delta \underline{\boldsymbol{k}}_0^T, \ldots, \Delta \underline{\boldsymbol{k}}_N^T\}$ store the nodal incremental motion and generalized curvature vectors, respectively.

As discussed in chapter 3, the curvature vector resulting from the interpolation

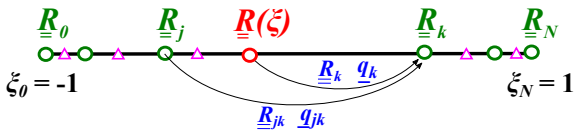

Figure 4.2: Interpolation of rigid-body motion in a spectral element, GLL points: (∘), Gauss points: (△).

can be found as

$$\underline{\boldsymbol{k}}(\xi) = \left[\sum_{k=0}^{N} l_k(\xi)\boldsymbol{T}^{-1}(-\underline{\boldsymbol{q}}_k)\right]^{-1} \sum_{k=0}^{N} l'_k(\xi)\underline{\boldsymbol{q}}_k. \tag{4.4}$$

The curvature vector at the nodes (the GLL points) is

$$\underline{\boldsymbol{k}}_j = \frac{1}{J_j}\sum_{k=0}^{N} l'_k(\eta_j)\underline{\boldsymbol{s}}_{jk}, \tag{4.5}$$

where l_k denote Lagrange's polynomial based on Gauss-Lobatto abscissæ, see appendix (B.4), and J_j is the Jacobian associated with the transformation from the arc-length coordinate s to the non-dimensional coordinate η. The curvature field over one element is constructed by interpolating the nodal curvatures. This approach was pioneered by Bathe and Dvorkin [151]: the nodes, or GLL points, are used as tying points. The following assumed curvature field is introduced

$$\underline{\boldsymbol{k}}(\eta) = \sum_{i=0}^{N-1} \bar{\ell}_i(\eta)\left[\sum_{j=0}^{j=N} l_j(\eta_j^G)\underline{\boldsymbol{k}}_j\right], \tag{4.6}$$

where $\bar{\ell}_i(\eta)$, $i = 0, 1, \ldots, N-1$ are Lagrangian polynomials based on Gauss-Legendre abscissæ defined in appendix (B.4), and $\underline{\boldsymbol{k}}_j$ are the curvatures at the GLL points defined in eq. (4.5). Clearly, the assumed curvature fields are polynomials of degree

$N-1$. Because the Gauss points are located at the zeros of the Legendre polynomial of degree $P_{N+1}(\eta)$, the assumed curvatures equal the true curvatures at the Gauss points if the true curvature (4.4) are polynomials of degree N. Because quantities $d_{j,k}$ and $l_j(\eta_j^G)$ are numerical values, the linearization of assumed curvature field only requires the linearization of curvature expressions at the nodes, $\underline{\boldsymbol{k}}_j$.

4.3 Governing equations

The governing equations of the problem will be derived from Hamilton's principle. Inertial effects due to sectional warping can be ignored for beams undergoing low frequency motion, i.e., frequencies whose associated wave lengths are much longer than the dimensions of the cross-section [53, 152]. After integration over the cross-section of the beam, the kinetic energy can be found as

$$K = \frac{1}{2}\int \underline{v}^T M\, \underline{v}\, \mathrm{d}s = \frac{1}{2}\int \underline{v}^T \underline{p}\, \mathrm{d}s, \tag{4.7}$$

where array $\underline{p} = M\,\underline{v}$ stores the components of the momentum vector resolved in the material frame. The sectional mass matrix, M, is defined as

$$M = \begin{bmatrix} mI_3 & m\tilde{q}_c^T \\ m\tilde{q}_c & \varrho_B \end{bmatrix}, \tag{4.8}$$

where m is the sectional mass per unit span, vector $\underline{q}_c$ is the position vector of the sectional center of mass with respect to reference point **B**, and tensor ϱ_B, of size 3×3, is the sectional mass moment of inertia per unit span computed with respect

to point **B**. Taking a variation of the kinetic energy expressed by eq. (4.7) gives $\delta K = \int \delta \underline{v}^T M \, \underline{v} \, \mathrm{d}s$. Using compatibility relationships (2.45b) and integrating by parts then leads to

$$\delta K = -\int \underline{\delta u}^T \left(\underline{\dot{p}} - \tilde{v}^T \underline{p} \right) \, \mathrm{d}s. \tag{4.9}$$

Sectional warping leads to strain components of the same order as those due to rigid-section motion and hence, warping effects must be taken into account when evaluating the strain energy. Hodges et al. [44, 46, 47] have shown that the three-dimensional beam problem can be decomposed into a linear, two-dimensional analysis over the cross-section, and a nonlinear, one-dimensional analysis along the beam's span. Those authors used the variational asymptotic method to reach this conclusion. More recently, the same conclusion was reached by Bauchau and Han [48, 49] using the Hamiltonian formalism. A byproduct of the two-dimensional sectional analysis is the sectional stiffness matrix, D, of size 6×6, which takes into account the warping effects due to geometric complexity and material heterogeneity of the cross-section. The strain energy can be found as

$$V = \frac{1}{2} \int \underline{e}^T D \, \underline{e} \, \mathrm{d}s = \frac{1}{2} \int \underline{e}^T \underline{f} \, \mathrm{d}s, \tag{4.10}$$

where vector $\underline{f} = D \, \underline{e}$, of size 6×1, stores the sectional stress resultants resolved in the material frame and vector $\underline{e}$, of size 6×1, stores the components of the sectional deformation measures resolved in the material frame as defined by eq. (4.3).

Taking a variation of the strain energy expressed by eq. (4.10) gives $\delta V =$

$\int \delta \underline{e}^T D\, \underline{e}\ \mathrm{d}s$. Using the compatibility relationships of eq. (2.45c), variation of the strain energy becomes

$$\delta V = \frac{1}{2}\int \left(\underline{\delta u}'^T \underline{\mathcal{F}} + \underline{\delta u}^T \tilde{\underline{\kappa}}^T \underline{f}\right)\ \mathrm{d}s = -\int \underline{\delta u}^T \left(\underline{f}' - \tilde{\underline{\kappa}}^T \underline{f}\right) \mathrm{d}s, \tag{4.11}$$

where the second equality results from integrating by parts.

For the problem at hand, the virtual work done by the externally applied forces is expressed as

$$\delta W = \int \underline{\delta u}^T \underline{l}\ \mathrm{d}s, \tag{4.12}$$

where vector $\underline{l}^T = \{\underline{n}^T\ \underline{m}^T\}$, of size 6×1, stores the components of the externally applied force, $\underline{n}$, and moment vector, $\underline{m}$, per unit span of the beam, respectively, resolved in the material frame.

The principle of virtual work states that $\delta V - \delta K - \delta W = 0$, introducing eqs. (4.9), (4.11) and (4.12) leads to $\int [\underline{\delta u}^T (\underline{\dot{p}} - \tilde{v}^T \underline{p} + \tilde{\underline{\kappa}}^T \underline{f} - \underline{l}) + \underline{\delta u}^T \underline{f}]\ \mathrm{d}s = 0$ and the weak form of the governing equations of motion then result from integration by parts

$$\int \underline{\delta u}^T \left[\underline{\dot{p}} - \tilde{v}^T \underline{p} - \underline{f}' + \tilde{\underline{\kappa}}^T \underline{f} - \underline{l}\right]\ \mathrm{d}s = 0. \tag{4.13}$$

Because the virtual motion vector is arbitrary, the strong form of the governing equations becomes

$$\left(\underline{\dot{p}} - \tilde{v}^T \underline{p}\right) - \left(\underline{f}' - \tilde{\underline{\kappa}}^T \underline{f}\right) = \underline{l}, \tag{4.14}$$

where the first and second terms on the left-hand side are the contributions of inertial and elastic forces, respectively.

4.4 Finite element formulation

The virtual motion and velocity vectors are interpolated within the elements as

$$\underline{\delta u} = \sum_{k=0}^{N} l_k(\eta)\underline{\delta u}_k = L(\eta)\,\hat{\underline{\delta u}}, \tag{4.15a}$$

$$\underline{v} = \sum_{k=0}^{N} l_k(\eta)\underline{v}_k = L(\eta)\,\hat{\underline{v}}, \tag{4.15b}$$

where $l_k(\eta)$ denote Lagrange's polynomial based on Gauss-Lobatto abscissæ, $L(\eta) = [l_0 I_6, \ldots, l_N I_6]$ stacks all the shape functions, $\hat{\underline{\delta u}}^T = \{\underline{\delta u}_0^T, \ldots, \underline{\delta u}_N^T\}$ and $\hat{\underline{\boldsymbol{v}}}^T = \{\underline{v}_0^T, \ldots, \underline{v}_N^T\}$ store the nodal values for motion increments and velocities, respectively. Introducing the interpolation into the weak form of the governing equations (4.13) and using quadrature rules will yield the discretized governing equations

$$\bar{M}\,\dot{\hat{\underline{v}}} - \hat{\underline{f}}^{\text{iner}} + \hat{\underline{f}}^{\text{int}} = \hat{\underline{f}}^{\text{ext}}, \tag{4.16}$$

where the mass matrix $\bar{M}$, gyroscopic force $\hat{\underline{f}}^{\text{iner}}$, elastic force $\hat{\underline{f}}^{\text{int}}$, and external force $\hat{\underline{f}}^{\text{ext}}$ are defined as

$$\begin{aligned}
\bar{M} &= \int_{-1}^{1} JL^T M\, L \, \mathrm{d}\eta, \\
\hat{\underline{f}}^{\text{iner}} &= \int_{-1}^{1} JL^T \tilde{\underline{v}}^T M\, \underline{v} \, \mathrm{d}\eta, \\
\hat{\underline{f}}^{\text{int}} &= \int_{-1}^{1} \left[L'^T D\, \underline{e} + JL^T \tilde{\underline{\kappa}}^T(\eta) D\, \underline{e} \right] \, \mathrm{d}\eta, \\
\hat{\underline{f}}^{\text{ext}} &= \int_{-1}^{1} JL^T \underline{f}^{\text{ext}} \, \mathrm{d}\eta,
\end{aligned} \tag{4.17}$$

where $J = \mathrm{d}s/\mathrm{d}\eta$ is the Jacobian associated with the coordinate transformation.

A linearization of gyroscopic force $\hat{\underline{f}}^{\text{iner}}$ yields

$$\Delta\hat{\underline{f}}^{\text{iner}} = \int_{-1}^{1} JL^T(\tilde{\underline{v}}^T M + \underline{p}^U) \, \mathrm{d}\eta \; \Delta\hat{\underline{\boldsymbol{v}}} = G\, \Delta\hat{\underline{v}}, \tag{4.18}$$

where notation $(\bullet)^U$, a linear map between a vector of size 6×1 and a matrix of size 6×6, is defined as $\tilde{\underline{\kappa}}^T \underline{p} = (\underline{p})^U \underline{\kappa}$. Similarly, eq. (3.48) is introduced to linearize the elastic forces, leading to

$$\Delta\hat{\underline{f}}^{\text{int}} = \int_{-1}^{1} \left[L'^T D\, L + JL^T(\tilde{\underline{\kappa}}^T D + \underline{f}^U) L \right] \mathcal{B} \, \mathrm{d}\eta \; \hat{\underline{\Delta u}} = K\, \hat{\underline{\Delta u}}. \tag{4.19}$$

Finally, the linearized governing equations are found as

$$M\Delta\dot{\hat{\underline{v}}} - G\Delta\hat{\underline{v}} + K\, \hat{\underline{\Delta u}} = \hat{\underline{r}}, \tag{4.20}$$

where $\hat{\underline{r}}$ is the residual.

To evaluate the integration in eqs. (4.17), (4.18), and (4.19), two types of quadrature rules are considered: the reduced Gauss-Lobatto and Gauss rules. The force vectors and matrices based on Gauss-Lobatto quadrature rule (B.7b) are found as

$$\begin{aligned}
\hat{\underline{f}}^{\text{iner}} &= \text{diag}(\tilde{v}_k^T) M\, \hat{\underline{v}}, \\
\hat{\underline{f}}^{\text{int}} &= \left[N + \text{diag}(J_k w_{\nu,k} \tilde{\kappa}_k^T) \right] \text{diag}(D)\, \hat{\underline{e}}, \\
M &= \text{diag}(J_k w_{\nu,k} M), \\
G &= \text{diag}(\tilde{v}_k^T) M + \text{diag}(J_k w_{\nu,k} D\, \underline{e}_k), \\
K &= \left[\left[N + \text{diag}(J_k w_{\nu,k} \tilde{\kappa}_k^T) \right] \text{diag}(D) + \text{diag}(J_k w_{\nu,k} [D\, \underline{e}_k)^U] \right] B.
\end{aligned}$$

Derivative matrix N is defined as $N = [d^T \text{diag}(w_k)] \otimes I_6$, where matrix d is defined by eq. (B.16), notation $\otimes$ indicates a Kronecker product, $w_{\nu,k}$ are the weights at the GLL points, and vector $\hat{\underline{e}}^T = \{\underline{e}_0^T, \ldots, \underline{e}_N^T\}$ stacks strain components at the GLL points. The external force vector is $(\hat{\underline{f}}^{\text{ext}})^T = \{J_0 w_0 (\underline{f}^{\text{ext}})^T(\eta_0), \ldots, J_N w_N (\underline{f}^{\text{ext}})^T(\eta_N)\}$. Because the nodes and quadrature points are collocated, the mass matrix becomes diagonal and the expression for the elastic forces is simpler than that resulting from Gaussian quadrature in conventional finite element formulations.

The force vectors and matrices based on Gauss quadrature rule (B.7) are found

as

$$
\begin{aligned}
\underline{\hat{f}}^{\text{iner}} &= \sum_{i=0}^{N-1} J(\mu_i) w_{\mu,i} L^T(\mu_i) \tilde{\underline{v}}^T(\mu_i) M\, \underline{v}(\mu_i), \\
\underline{\hat{f}}^{\text{int}} &= \sum_{i=0}^{N-1} \left[w_{\mu,i} L'^T(\mu_i) D\, \underline{e}(\mu_i) + J(\mu_i) w_{\mu,i} L^T(\mu_i) \tilde{\underline{\kappa}}^T(\mu_i) D\, \underline{e}(\mu_i) \right], \\
\underline{\hat{f}}^{\text{ext}} &= \sum_{i=0}^{N-1} J(\mu_i) w_{\mu,i} L^T(\mu_i) \mathcal{L}(\mu_i) \\
M &= \sum_{i=0}^{N-1} J(\mu_i) w_{\mu,i} L^T(\mu_i) M\, L(\mu_i), \\
G &= \sum_{i=0}^{N-1} J(\eta_i) w_{\mu,i} L^T(\mu_i) \left[\tilde{\underline{v}}^T(\mu_i) M + (M\, \underline{v}(\mu_i))^U \right] L(\mu_i), \\
K &= \sum_{i=0}^{N-1} \left[w_{\mu,i} L'^T(\mu_i) D + J(\mu_i) w_{\mu,i} L^T(\mu_i) \left[\tilde{\underline{\kappa}}^T(\mu_i) D + (D\, \underline{e}(\mu_i))^U \right] \right] L(\mu_i) \mathcal{B}.
\end{aligned}
$$

Therein, $w_{\mu,k}$ are the weights at the Gauss points. The curvature and strain components at the Gauss points are evaluated through interpolations of the nodal quantities, i.e., $\underline{\kappa}(\mu_i) = \sum_{k=0}^{N} l_k(\mu_i) \underline{\kappa}_k$ and $\underline{e}(\mu_i) = \sum_{k=0}^{N} l_k(\mu_i) (\underline{\kappa}_k - \underline{\kappa}_{0k})$.

4.5 Numerical examples

To validate the proposed approach, a set of numerical examples will be presented. In all cases, the sectional mass and stiffness matrices of the beam are computed using *SectionBuilder*, a finite element based tool for the analysis of cross-sections of beams of arbitrary configuration made of anisotropic materials [48,49,51]. Reference solutions will be provided by *Dymore 4*, a finite element based, flexible multibody system analysis tool that uses the classical description of kinematics, i.e., the displacement and rotation fields are treated separately. Predictions based on

Gauss-Lobatto and Gauss quadrature rules are compared in the first examples; for the remaining examples, only the predictions based on Gauss quadrature rules are presented.

4.5.1 Cantilevered beam with a 45-degree bend

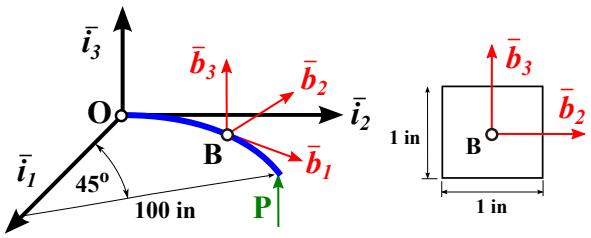

Figure 4.3: Cantilevered beam with a 45-degree bend.

In this example, the response of the 45-degree bend cantilevered beam shown in fig. 4.3 is investigated. The beam is cantilevered at point **O** and subjected to a static tip load, $P = 600$ lb, acting along unit vector $\bar{\imath}_3$. The initial curvature of the beam about unit vector $\bar{\imath}_3$ is $k_3 = -0.01$ in^{-1}. The cross-section of the beam is $1{\times}1$ in^2. The beam is made of isotropic material with the following properties: Young's modulus $E = 10^7$ psi, and Poisson's ratios $\nu = 0.0$. The non-vanishing entries of sectional stiffness matrix D predicted by *SectionBuilder* are $\mathcal{D}_{11} = 1.00\ 10^7$ lb, $\mathcal{D}_{22} = \mathcal{D}_{33} = 4.17\ 10^6$ lb, $\mathcal{D}_{44} = 7.03\ 10^5$ lb·in^2, $\mathcal{D}_{55} = \mathcal{D}_{66} = 8.33\ 10^5$ lb·in^2, and $\mathcal{D}_{16} = -8.33\ 10^3$ lb·in.

	No. of elements	u_1 (in)	u_2 (in)	u_3 (in)
Proposed, Gauss-Lobatto	8 (3-node element)	-13.670	-23.712	53.421
Proposed, Gauss	8 (3-node element)	-13.731	-23.818	53.607
Ibrahimbegović [55]	8 (3-node element)	-13.729	-23.814	53.605
Bathe and Bolourchi [153]	8	-13.4	-23.5	53.4

Table 4.1: Displacement components of the free tip.

Table 4.1 lists the present predictions based on the Gauss-Lobatto and Gauss quadrature rules, those of Ibrahimbegović [55], and those of Bathe and Bolourchi [153]. Figure 4.4 and 4.5 show the error measure, $\|\underline{u} - \underline{u}_r\|/\|\underline{u}_r\|$, for the tip displacement of the beam versus the number of spectral elements on a logarithmic plot, for the Gauss-Lobatto and Gauss quadrature rules, respectively. The reference solution, $\underline{u}_r$, is obtained using 128 spectral element of order $N = 6$. The figures shows the error measures for spectral elements with $N = 2$, 3, 4, and 5 (corresponding to 3, 4, 5, and 6 nodes in each element). The predictions based on Gauss-Lobatto rules are quite inaccurate compared with those based on Gauss rules. For the Gauss rules based approach, the elements based on quaternion and geodesic metric interpolations are slightly more accurate than those based on the matrix metric interpolation. For the three interpolation schemes, the convergence rate is between $2N - 0.5$ to $2N$. For 3-node elements, the quaternion and geodesic metric based interpolations have the same order of accuracy as the 3-node elements based on classical kinematics implemented in *Dymore 4*. For 4-node elements, the proposed approach become more accurate than those based on classical kinematics when the number of elements is larger than 8.

For the solutions of 8 4-node elements, figures 4.6 and 4.7 show the axial and shear strains, respectively, obtained from the interpolated and assumed strain fields defined by eqs. (4.4) and 4.6, respectively, over the root element of the beam. The interpolated strain field behaves as a polynomial of degree 3 and presents 3 zeros. The interpolated and assumed strain fields are nearly identical at the Gauss points. The beam elements based Gauss-Lobatto quadrature rules suffer from locking be-

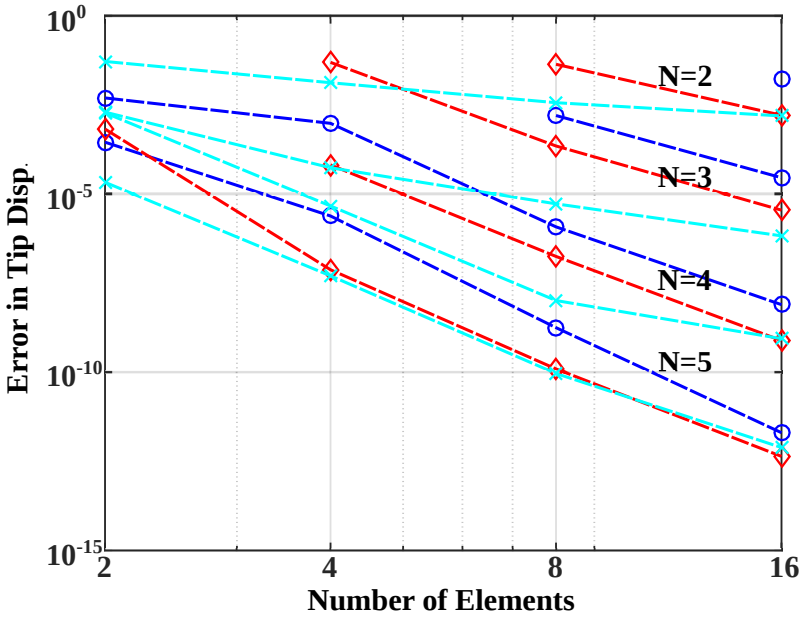

Figure 4.4: Tip displacement error versus number of elements and degree of the Lagrange polynomial for elements based on Gauss-Lobatto quadrature rules. Matrix metric: ($\circ$), quaternion metric: ($\diamond$), geodesic metric: ($\times$).

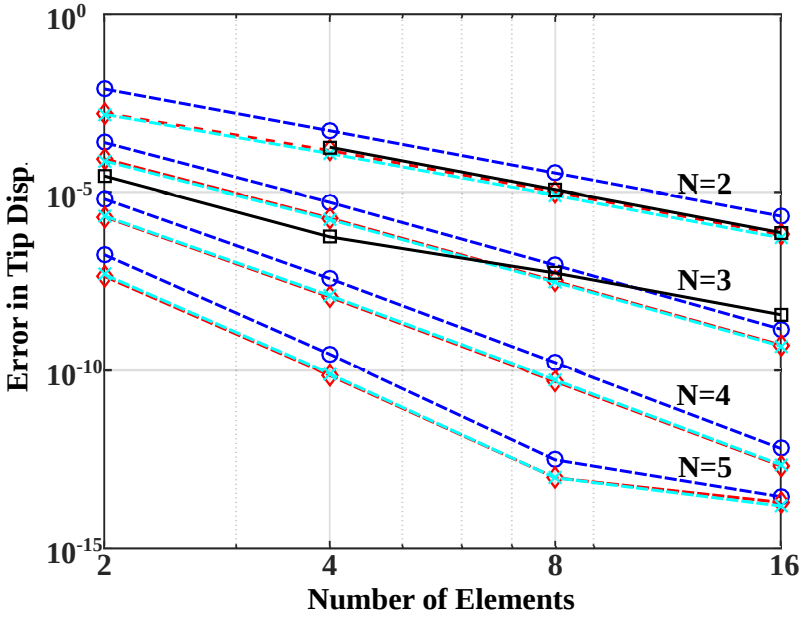

Figure 4.5: Tip displacement error versus number of elements and degree of the Lagrange polynomial for elements based on Gauss quadrature rules. Matrix metric: ($\circ$), quaternion metric: ($\diamond$), geodesic metric: ($\times$), *Dymore 4*: ($\square$).

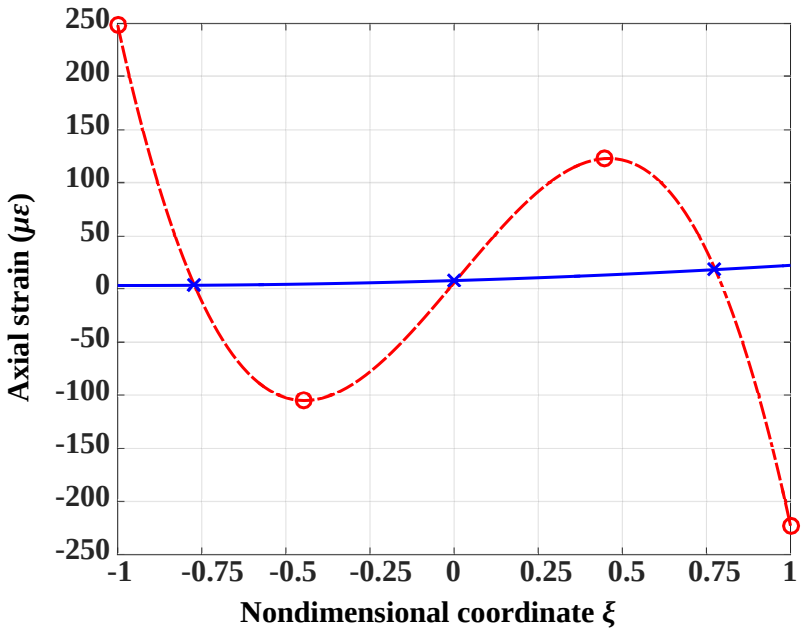

Figure 4.6: Axial strain ε_1 in the root element. Interpolated strain (eq. (4.4)): dashed line; assumed-strain (eq. 4.6): solid line. Gauss points: ($\times$), Gauss-Lobatto points: ($\circ$).

cause the axial and shear strains interpolated at the GLL quadrature points are very inaccurate and hence, the strain energy associated with axial and shear deformation is grossly overestimated. The assumed strains is a polynomial of degree 2, which is consistent with the motion field interpolated via polynomials of order 3. Consequently, beam elements based on assumed strain and Gauss quadrature rules are locking free.

4.5.2 Post-buckling of a circular arch

Figure 4.8 depicts a circular arch of radius $R = 100$, hinged at one end and clamped at another, subjected to a vertical force, P, applied at mid-span. The post-buckling behavior of this structure is investigated. The beam's sectional stiffness

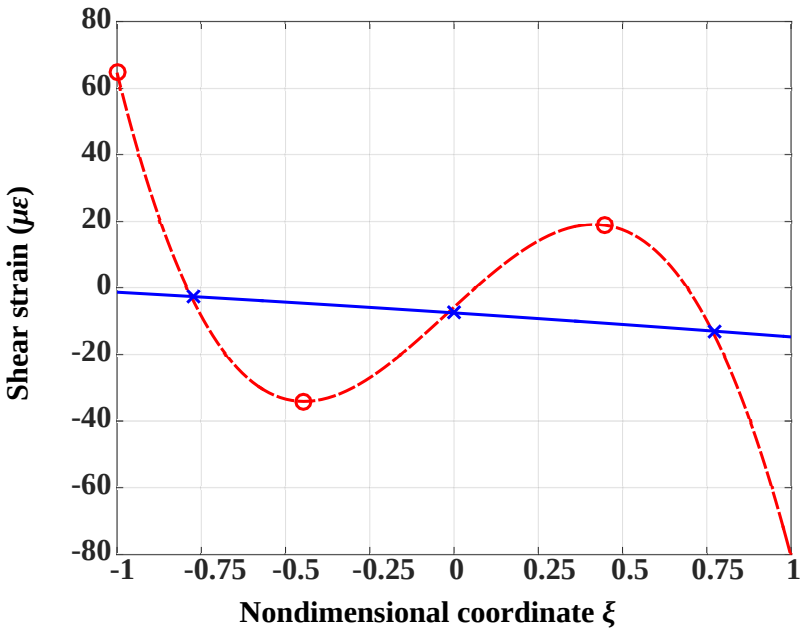

Figure 4.7: Shear strain γ_{12} in the root element. Interpolated strain (eq. (4.4)): dashed line; assumed-strain (eq. 4.6): solid line. Gauss points: ($\times$), Gauss-Lobatto points: ($\circ$).

matrix is $D = \text{diag}(10^8, 10^8, 10^8, 10^6, 10^6, 10^6, 10^6)$. This example was first described by Crisfield [154], who provides all input data in non-dimensional form.

The mesh consists of 12 4-node elements and the arc-length method of Crisfield [154] is used to trace the buckling and post buckling behavior of the structure. Figure 4.9 shows the applied load as a function of the magnitude of the displacement vector of the mid-span point; the configurations of the arch for $P = 897.9$ and - 17.8, labeled as curves 1 and 2, respectively, are shown in fig. 4.10. For the present approach, all the three metrics provide the same critical buckling load $P_{\text{cr}} = 897.9$, to four significant digits. This compares favorably with the buckling load $P_{\text{cr}} = 897$ reported by DaDeppo and Schmidt [155].

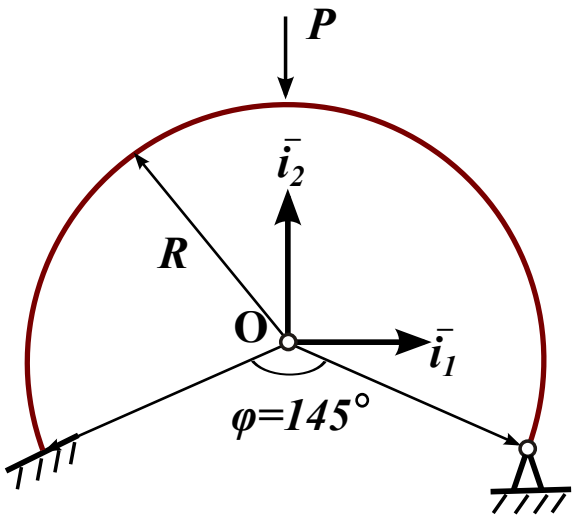

Figure 4.8: Configuration of a clamped-hinged circular arch.

4.6 Summaries and conclusions

A novel formulation was proposed for geometrically exact beams. It combines the spectral method with motion interpolation schemes. Motion interpolation schemes based on matrix, quaternion, and geodesic metrics yield simple expressions for the sectional strains and linearized strain-motion relationships at the nodes. Beam elements based on Gauss-Lobatto and Gauss quadrature rules were investigated. Gauss-Lobatto rules only requires summation over the nodes of the elements. Gauss quadrature rules requires curvatures at the Gauss points that are evaluated through an interpolation of the nodal curvatures. In both cases, the expressions for the internal forces and tangent stiffness matrices are simplified. Consequently, the proposed spectral element formulation is much easier to implement than its conventional counterpart.

Numerical examples have demonstrated the Gauss-Lobatto rules based elements suffer from axial and shear locking, while the Gauss rules based elements

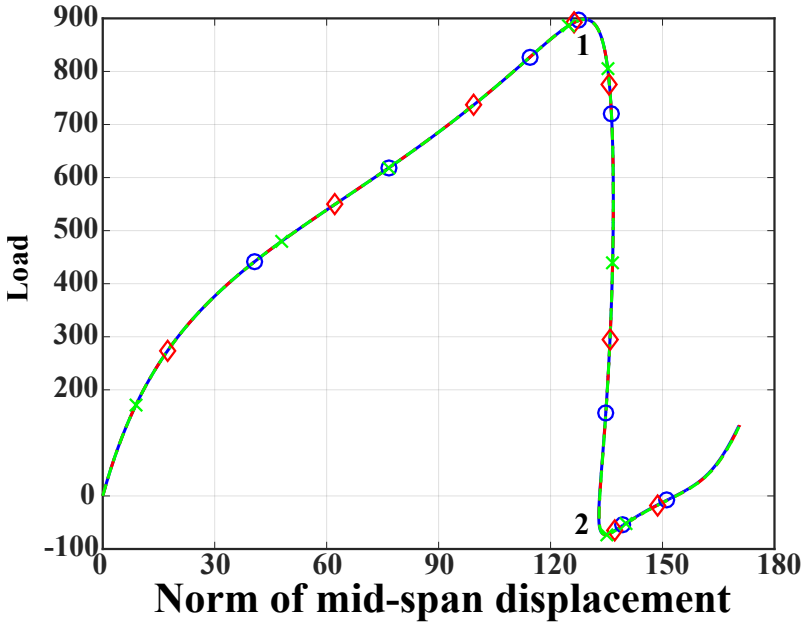

Figure 4.9: Load-displacement curve, matrix metric: (∘), quaternion metric: (⋄), geodesic metric: (×).

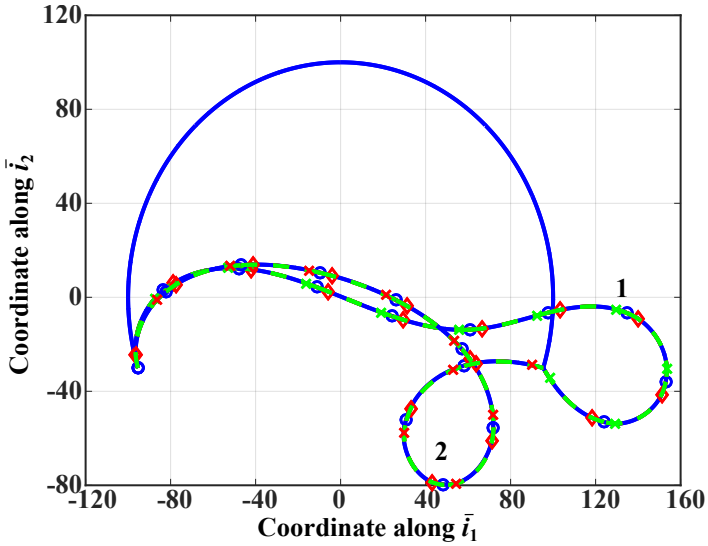

Figure 4.10: Buckling modes at the critical points, matrix metric: (∘), quaternion metric: (⋄), geodesic metric: (×).

are locking free. The convergence rate of a $(N+1)$-node spectral element based on Gauss rules is about $2N - 0.5$ to $2N$ for all the three interpolation schemes. For the static problem investigated, the matrix metric based interpolation is less accurate than the quaternion and geodesic metric based interpolation. As the number of elements increase, the Gauss rules base formulation becomes more accurate than the conventional beam element in *Dymore 4*. Excellent agreement is observed between the various motion interpolation strategies and the prediction of classical formulations.

Chapter 5: Galerkin Methods in Time Domain: the Initial Value Problem

This chapter focuses on the application of Galerkin's method for initial value problems. Weak formulations for both of the continuous and discontinuous Galerkin methods are derived. The time integration schemes resulting from discontinuous Galerkin method are unconditionally stable and suitable for initial value problems.

5.1 Lagrangian and Hamiltonian of a multibody system

Consider a flexible multibody system composed of rigid bodies, kinematic joints, and flexible components such as beams. The finite element method is used to discretize the problem in spatial dimensions, leading to n structural nodes. The kinematics of each structural node is represented by motion tensor $\mathcal{R}^i$, $i = 1, 2, \ldots, n$. Let the n structural nodes be subjected to m holonomic constraints, $\underline{g}(\mathcal{R}^1, \ldots, \mathcal{R}^n, t) = \underline{0}$. The augmented Lagrangian of the system is

$$\mathcal{L}(\mathcal{R}^1, \ldots, \mathcal{R}^n, \underline{\hat{v}}, \underline{\lambda}, t) = 1/2\, \underline{\hat{v}}^T M\, \underline{\hat{v}} - \mathcal{V}(\mathcal{R}^1, \ldots, \mathcal{R}^n) - \underline{\lambda}^T\, \underline{g}, \tag{5.1}$$

where matrix $M \in \mathbb{R}^{6n\times 6n}$ is the mass matrix of the system, array $\hat{\underline{v}}^T = \{\underline{v}^{1T}, \ldots, \underline{v}^{nT}\}$ stacks the velocity for all structural nodes, $\mathcal{V}(\mathcal{R}^1, \ldots, \mathcal{R}^n)$ denotes the potential energy of the system, and $\underline{\lambda} \in \mathbb{R}^m$ are the Lagrange multipliers. The momentum is readily found as $\hat{\underline{p}} = \partial\mathcal{L}/\partial\hat{\underline{v}} = M\,\hat{\underline{v}}$. The augmented Hamiltonian of the system is found from the Lagrangian via Legendre's transformation,

$$\begin{aligned}\mathcal{H}(\mathcal{R}^1, \ldots, \mathcal{R}^n, \hat{\underline{p}}, \underline{\lambda}, \underline{t}) &= \max_{\hat{\underline{v}}} \left[\hat{\underline{v}}^T \hat{\underline{p}} - \mathcal{L}(\mathcal{R}^1, \ldots, \mathcal{R}^n, \hat{\underline{v}}, \underline{\lambda}, t)\right] \\ &= 1/2\,\hat{\underline{p}}^T M^{-1}\,\hat{\underline{p}} + \mathcal{V}(\mathcal{R}^1, \ldots, \mathcal{R}^n) + \underline{\lambda}^T \underline{g} \\ &= 1/2\,\hat{\underline{w}}^T M\,\hat{\underline{w}} + \mathcal{V}(\mathcal{R}^1, \ldots, \mathcal{R}^n) + \underline{\lambda}^T \underline{g},\end{aligned} \tag{5.2}$$

where independent state variables, $\hat{\underline{w}}$, are introduced to represent the velocities. Theoretically, independent velocity vector $\hat{\underline{w}}$ should satisfy the constraints at the velocity level, i.e., $\dot{\underline{g}} = G\,\hat{\underline{w}} + \underline{g}_t = \underline{0}$, where $\underline{g}_t = \partial\underline{g}/\partial\underline{t}$. Constraints at the velocity level can be enforced by adding a term $\underline{\mu}^T\dot{\underline{g}}$ into the Hamiltonian and application of Hamiltonian variation principle will leads to an index-2 formulation. In this work, constraints at the velocity level are omitted for simplicity.

5.2 Weak formulations of the continuous and discontinuous Galerkin methods

Suppose the entire time interval $[0, T]$ is divided to K 2-node elements denoted as $[t^+_{I-1}, t^-_I]$, where $I = 1, \ldots, K$, $t^b_0 = 0$, and $t^b_K = T$. In

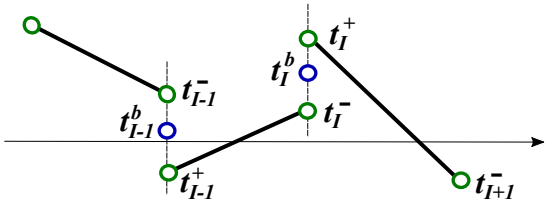

Figure 5.1: Illustration of discontinuous Galerkin method.

general, quantities evaluated at t_I^-, t_I^b,

and t_I^+ are different in Galerkin meth-

ods. For the initial value problems, the equations of motion are solved element by element. For periodic boundary value problems, the unknowns at all the time grids of an entire period are solved for concurrently. Consider the Ith element $[t_{I-1}^+, t_I^-]$. A non-dimensional coordinate $\eta = 2(t - t_{I-1}^+)/(t_I^+ - t_{I-1}^-) - 1 \in [-1, 1]$ is introduced for convenience. Let notation $(\cdot)'$ denotes a derivative with respect to the non-dimensional time η. Clearly, $(\cdot)' = \mathrm{d}t/\mathrm{d}\eta(\dot{\cdot}) \overset{\text{def}}{=} J(\dot{\cdot})$, where $J = (t_I^- - t_{I-1}^+)/2$ is the the Jacobian associated with the coordinate transformation.

The action of the system in the Ith element $[t_{I-1}^+, t_I^-]$ is

$$\mathcal{S} = \int_{t_{I-1}^+}^{t_I^-} \mathcal{L}\,\mathrm{d}t = \int_{t_{I-1}^+}^{t_I^-} \left(\underline{\hat{v}}^T M\, \underline{\hat{w}} - \mathcal{H}\right)\,\mathrm{d}t, \tag{5.3}$$

which depends on both of the primal variables $\mathcal{R}^i$, $i = 1, 2, \ldots, n$ and dual variables $\underline{\hat{w}}$. The Hamiltonian variational principle states

$$\begin{aligned}\delta\mathcal{S} + \int_{t_{I-1}^+}^{t_I^-} \underline{\hat{\delta u}}^T \underline{f}^{\text{ext}}\mathrm{d}t = {} & \underline{\hat{\delta u}}^T(t_I^-) M\, \underline{\hat{w}}(t_I^b) - \underline{\hat{\delta u}}^T(t_{I-1}^+) M\, \underline{\hat{w}}(t_{I-1}^b) \\ & + \delta\underline{\hat{w}}^T(t_I^-) M\, [\![\underline{\hat{q}}]\!]_I - \delta\underline{\hat{w}}^T(t_{I-1}^+) M\, [\![\underline{\hat{q}}]\!]_{I-1},\end{aligned} \tag{5.4}$$

where $\underline{\delta u}^T = \{\underline{\delta u}^{1T}, \ldots, \underline{\delta u}^{nT}\}$ stacks the virtual motion for all the structural nodes, $\underline{f}^{\text{ext}} \in \mathbb{R}^{6n}$ denotes the externally applied force, and $\underline{\hat{q}}$ represent the jump of motion

tensor at the boundaries, defined as

$$[\![\hat{\underline{q}}]\!]_{I-1} = \begin{Bmatrix} \log_{\mathrm{v}}[\mathcal{R}^{1T}(t^b_{I-1})\mathcal{R}^1(t^+_{I-1})] \\ \vdots \\ \log_{\mathrm{v}}[\mathcal{R}^{nT}(t^b_{I-1})\mathcal{R}^n(t^+_{I-1})] \end{Bmatrix}, \quad \text{and} \quad [\![\hat{\underline{q}}]\!]_I = \begin{Bmatrix} \log_{\mathrm{v}}[\mathcal{R}^{1T}(t^b_I)\mathcal{R}^1(t^-_I)] \\ \vdots \\ \log_{\mathrm{v}}[\mathcal{R}^{nT}(t^b_I)\mathcal{R}^n(t^-_I)] \end{Bmatrix}.$$

Term $\delta\hat{\underline{w}}^T M\,[\![\hat{\underline{q}}]\!]$ does not appear in the classic Hamiltonian variational principle [77, 156]; it is introduced here to account for the discontinuity of primal variable R^i at the element boundaries. The logarithm of relative motion tensor is used to measure the jump of rigid-body motions at element boundaries. If the primal variables were vectors in Euclidean space, the jumps would simply be $[\![\hat{\underline{q}}]\!]_{I-1} = \hat{\underline{q}}(t^+_{I-1}) - \hat{\underline{q}}(t^b_{I-1})$ [84, 88, 157].

Introducing the expression of action (5.3) to eq. (5.4) leads to the weak form of Hamilton equations

$$\begin{aligned}
&\int_{t^+_{I-1}}^{t^-_I} \left[\delta\hat{\underline{w}}^T M\,\hat{\underline{v}} + \delta\hat{\underline{v}}^T M\,\hat{\underline{w}} - \delta\mathcal{H} + \hat{\underline{\delta u}}^T \underline{f}^{\mathrm{ext}}\right] \mathrm{d}t \\
&= \int_{t^+_{I-1}}^{t^-_I} \left[\delta\hat{\underline{w}}^T M(\hat{\underline{v}} - \hat{\underline{w}}) + \dot{\hat{\underline{\delta u}}}^T M\,\hat{\underline{w}} - \hat{\underline{\delta u}}^T[\mathrm{diag}(\tilde{w}^i)^T M\,\hat{\underline{w}} + \underline{f}^{\mathrm{int}} + G^T\underline{\lambda} - \underline{f}^{\mathrm{ext}}] + \delta(\underline{\lambda}^T\underline{g})\right] \mathrm{d}t \\
&= \hat{\underline{\delta u}}^T(t^-_I) M\,\hat{\underline{w}}(t^b_I) - \hat{\underline{\delta u}}^T(t^+_{I-1}) M\,\hat{\underline{w}}(t^b_{I-1}) + \delta\hat{\underline{w}}^T(t^-_I) M\,[\![\hat{\underline{q}}]\!]_I - \delta\hat{\underline{w}}^T(t^+_{I-1}) M\,[\![\hat{\underline{q}}]\!]_{I-1},
\end{aligned} \tag{5.5}$$

where eq. (2.45b) is introduced to yield the first equality; linearization of the potential energy leads to the internal forces $\delta\mathcal{V}(\mathcal{R}^1, \ldots, \mathcal{R}^n) = \hat{\underline{\delta u}}^T \underline{f}^{\mathrm{int}}$; linearization of the constraints leads to the Jacobian associated with constraints $\delta\underline{g} = G\,\hat{\underline{\delta u}}$.

The boundary term in eq. (5.6) can be dealt with in three alternative manners.

- The Bi-discontinuous Galerkin (bi-DG) formulation, in which the quantities at $(\cdot)^-$, $(\cdot)^b$, and $(\cdot)^+$ are differ. The bi-discontinuous Galerkin method of primal and dual variables has been proposed by Borri and his coauthors in refs. [88, 158]. Integrating the term $\dot{\hat{\delta}}\underline{u}^T M\, \underline{\hat{w}}$ by parts yield

$$\begin{aligned}
&\int_{t_{I-1}^+}^{t_I^-} \begin{Bmatrix} M\delta\underline{\hat{w}} \\ \hat{\underline{\delta u}} \\ \delta\underline{\lambda} \end{Bmatrix}^T \begin{Bmatrix} \underline{\hat{w}} - \underline{\hat{v}} \\ M\,\dot{\underline{\hat{w}}} - \mathrm{diag}(\tilde{w}^i)^T M\,\underline{\hat{w}} + \underline{f}^{\mathrm{int}} + G^T\underline{\lambda} - \underline{f}^{\mathrm{ext}} \\ \underline{g} \end{Bmatrix} \mathrm{d}t \\
&= \begin{Bmatrix} M\,\delta\underline{\hat{w}}(t_{I-1}^+) \\ \hat{\underline{\delta u}}(t_{I-1}^+) \end{Bmatrix}^T \begin{Bmatrix} [\![\underline{\hat{q}}]\!]_{I-1} \\ M[\![\underline{\hat{w}}]\!]_{I-1} \end{Bmatrix} \\
&\quad - \begin{Bmatrix} M\,\delta\underline{\hat{w}}(t_I^-) \\ \hat{\underline{\delta u}}(t_I^-) \end{Bmatrix}^T \begin{Bmatrix} [\![\underline{\hat{q}}]\!]_{I} \\ M[\![\underline{\hat{w}}]\!]_{I} \end{Bmatrix},
\end{aligned} \tag{5.6}$$

where $[\![\underline{\hat{w}}]\!]_{I-1} = \underline{\hat{w}}(t_{I-1}^b) - \underline{\hat{w}}(t_{I-1}^+)$ and $[\![\underline{\hat{w}}]\!]_{I} = \underline{\hat{w}}(t_{I}^b) - \underline{\hat{w}}(t_{I}^-)$.

- The Discontinuous Galerkin (DG) method, in which both the motion tensors and velocities evaluated at times t_I^- and t_I^b are identical. The weak form (5.6) then become

$$\begin{aligned}
&\int_{t_{I-1}^+}^{t_I^-} \begin{Bmatrix} M\delta\underline{\hat{w}} \\ \hat{\underline{\delta u}} \\ \delta\underline{\lambda} \end{Bmatrix}^T \begin{Bmatrix} \underline{\hat{w}} - \underline{\hat{v}} \\ M\,\dot{\underline{\hat{w}}} - \mathrm{diag}(\tilde{w}^i)^T M\,\underline{\hat{w}} + \underline{f}^{\mathrm{int}} + G^T\underline{\lambda} - \underline{f}^{\mathrm{ext}} \\ \underline{g} \end{Bmatrix} \mathrm{d}t \\
&= \begin{Bmatrix} M\,\delta\underline{\hat{w}}(t_{I-1}^+) \\ \hat{\underline{\delta u}}(t_{I-1}^+) \end{Bmatrix}^T \begin{Bmatrix} [\![\underline{\hat{q}}]\!]_{I-1} \\ M[\![\underline{\hat{w}}]\!]_{I-1} \end{Bmatrix}.
\end{aligned} \tag{5.7}$$

- The Continuous Galerkin (CG) method, in which the quantities evaluated at time $(\cdot)^-$, $(\cdot)^b$, and $(\cdot)^+$ are identical and the jump of motion term $\hat{\underline{q}}$ vanishes. Finally, the weak Hamiltonian formulation (5.5) becomes

$$\begin{Bmatrix} \delta M\, \hat{\underline{w}} \\ \dot{\hat{\underline{\delta u}}} \\ \hat{\underline{\delta u}} \\ \delta \underline{\lambda} \end{Bmatrix}^T \begin{Bmatrix} \hat{\underline{w}} - \hat{\underline{v}} \\ -M\, \hat{\underline{w}} \\ -\mathrm{diag}(\tilde{w}^i)^T M\, \hat{w} + \underline{f}^{\mathrm{int}} + G^T \underline{\lambda} - \underline{f}^{\mathrm{ext}} \\ \underline{g} \end{Bmatrix} \mathrm{d}t = \underline{\hat{\delta u}^T(t_{I-1}^+) M\, \hat{\underline{w}}(t_{I-1}^b) - \hat{\delta u}^T(t_I^-) M\, \hat{\underline{w}}(t_I^b)}. \tag{5.8}$$

The underlined terms in eq. (5.8) are of opposite signs in adjacent elements and vanish when these elements are assembled. For periodic problem, the boundary nodes satisfy periodicity conditions $\mathcal{R}^i(0^b) = \mathcal{R}^i(T^b)$ and $\hat{\underline{w}}(0^b) = \hat{\underline{w}}(T^b)$. The first condition is enforced by using identical nodes for times $t_0 = 0$ and $t_K = T$; the second condition is enforced by the corresponding assembly process. Finally, the weak Hamiltonian formulation (5.5) for periodic problem becomes

$$\sum_{I=1}^{K} \int_{t_{I-1}^+}^{t_I^-} \begin{Bmatrix} \delta M\, \hat{\underline{w}} \\ \dot{\hat{\underline{\delta u}}} \\ \hat{\underline{\delta u}} \\ \delta \underline{\lambda} \end{Bmatrix}^T \begin{Bmatrix} \hat{\underline{w}} - \hat{\underline{v}} \\ -M\, \hat{\underline{w}} \\ -\mathrm{diag}(\tilde{w}^i)^T M\, \hat{w} + \underline{f}^{\mathrm{int}} + G^T \underline{\lambda} - \underline{f}^{\mathrm{ext}} \\ \underline{g} \end{Bmatrix} \mathrm{d}t = 0. \tag{5.9}$$

The DG method is suitable for the initial value problem and will be investigated in

the following sections.

5.3 Test and trial functions

The test functions are polynomials of the first degree, i.e., $\delta\hat{\underline{w}}$, $\hat{\underline{\delta u}} \in \text{span}(\underline{l} \otimes I_{6n})$ and $\delta\underline{\lambda} \in \text{span}(\underline{l} \otimes I_m)$, where $\underline{l} = \{l_0(\eta), l_1(\eta)\}$ and l_k, $k = 0, 1$, are Lagrange polynomials of the first degree. The independent velocity and Lagrange multipliers are interpolated linearly, $\hat{\underline{w}} = \sum_{k=0}^{1} l_k \hat{\underline{w}}_k$ and $\underline{\lambda} = \sum_{k=0}^{1} l_k \underline{\lambda}_k$. The motion tensors are interpolated using the dual-SLERP, as investigated in chapter (3). Let

$$\hat{\underline{q}}_{01} = \begin{Bmatrix} \log_{\text{v}}[\mathcal{R}^{1T}(t_{I-1}^{+})\mathcal{R}^{1}(t_I^{-})] \\ \vdots \\ \log_{\text{v}}[\mathcal{R}^{nT}(t_{I-1}^{+})\mathcal{R}^{n}(t_I^{-})] \end{Bmatrix}, \quad \hat{\underline{\Delta u}}(t) = \begin{Bmatrix} \underline{\Delta u}^{1}(t) \\ \vdots \\ \underline{\Delta u}^{n}(t) \end{Bmatrix}, \quad \text{and} \quad \underline{\Delta \boldsymbol{u}}_I = \begin{Bmatrix} \hat{\underline{\Delta u}}(t_{I-1}^{+}) \\ \hat{\underline{\Delta u}}(t_I^{-}) \end{Bmatrix},$$

stack the relative motion parameter vectors of each structural nodes from time t_{I-1}^{+} to t_I^{-}, incremental motion vectors of each structural node at arbitrary time t, and all the structure nodes at time t_{I-1}^{+} and t_I^{-}, respectively. The dual-SLERP (3.40a), (3.40b), and (3.41) for the n structural nodes can be written as

$$\hat{\underline{\Delta u}} = \mathbb{L}_{uI} \underline{\Delta \boldsymbol{u}}_I, \tag{5.10a}$$

$$\hat{\underline{v}} = \frac{1}{2J} \hat{\underline{q}}_{01}, \tag{5.10b}$$

$$\Delta\hat{\underline{v}} = \mathbb{L}_{vI} \underline{\Delta \boldsymbol{u}}_I, \tag{5.10c}$$

where

$$\mathbb{L}_{uI} = \begin{bmatrix} \mathcal{L}^1_{uI,1-6} & & & \mathcal{L}^1_{uI,7-12} & & \\ & \ddots & & & \ddots & \\ & & \mathcal{L}^n_{uI,1-6} & & & \mathcal{L}^n_{uI,7-12} \end{bmatrix},$$

$$\mathbb{L}_{vI} = \begin{bmatrix} \mathcal{L}^1_{vI,1-6} & & & \mathcal{L}^1_{vI,7-12} & & \\ & \ddots & & & \ddots & \\ & & \mathcal{L}^n_{vI,1-6} & & & \mathcal{L}^n_{vI,7-12} \end{bmatrix}.$$

Therein superscripts represent the label of structural nodes, subscripts $1-6$ and $7-12$ represent the corresponding columns of matrices. In view of identity (2.55b), a linearization of the jump of motion leads to

$$\begin{aligned} \Delta[\![\underline{\hat{q}}]\!]_{I-1} &= \mathbb{T}_{I-1,-}\,\underline{\hat{\Delta u}}(t^-_{I-1}) + \mathbb{T}_{I-1,+}\,\underline{\hat{\Delta u}}(t^+_{I-1}) \\ &= (\underline{1}^T_1 \otimes \mathbb{T}_{I-1,-})\underline{\Delta \boldsymbol{u}}_{I-1} + (\underline{1}^T_0 \otimes \mathbb{T}_{I-1,+})\underline{\Delta \boldsymbol{u}}_I, \end{aligned} \tag{5.11}$$

where

$$\mathbb{T}_{I-1,-} = \begin{bmatrix} -\mathcal{T}^{-1}(-[\![\underline{q}^1]\!]_{I-1}) & & \\ & \ddots & \\ & & -\mathcal{T}^{-1}(-[\![\underline{q}^n]\!]_{I-1}) \end{bmatrix},$$

$$\mathbb{T}_{I-1,+} = \begin{bmatrix} \mathcal{T}^{-1}([\![\underline{q}^1]\!]_I) & & \\ & \ddots & \\ & & \mathcal{T}^{-1}([\![\underline{q}^n]\!]_I) \end{bmatrix}.$$

The following notations are introduced

$$\underline{1}_0 = \begin{Bmatrix} 1 \\ \\ 0 \end{Bmatrix}, \quad \underline{1}_1 = \begin{Bmatrix} 0 \\ \\ 1 \end{Bmatrix}, \quad \underline{\imath} = \left(\int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \underline{l} \, \mathrm{d}t \right)^{-1} \underline{1}_0 = \frac{1}{J} \begin{Bmatrix} 2 \\ \\ -1 \end{Bmatrix}, \tag{5.12}$$

where $\underline{1}_0$ and $\underline{1}_1$ are introduced to indicate matrix or vectors with non-vanishing block at the first and last rows.

5.4 Governing equations

For convenience, matrices C, K, and N, all of size $6n \times 6n$, coming from the linearization of the gyroscopic, internal, and constraint forces, respectively, are introduced

$$\Delta[\mathrm{diag}(\tilde{w}_i)^T M \, \underline{\hat{w}}] = C \, \Delta\underline{\hat{w}}, \tag{5.13a}$$

$$\Delta\underline{f}^{\mathrm{int}} = K \, \hat{\underline{\Delta u}}, \tag{5.13b}$$

$$\Delta(G^T \underline{\lambda}) = N \, \hat{\underline{\Delta u}} + G^T \Delta\underline{\lambda}. \tag{5.13c}$$

Introducing the test and trial functions in section 5.3 into the weak form of DG method (5.7) leads to

$$\underline{\boldsymbol{w}}_I - \underline{\boldsymbol{v}}_I = \underline{\imath} \otimes [\![\underline{\hat{q}}]\!]_{I-1}, \tag{5.14a}$$

$$\mathbb{M}\underline{\dot{\boldsymbol{w}}}_I + \underline{\boldsymbol{f}}_I^{\mathrm{gyro}} + \underline{\boldsymbol{f}}_I^{\mathrm{int}} + \mathbb{G}_{I1}^T \underline{\boldsymbol{\lambda}}_I = \underline{\boldsymbol{f}}_I^{\mathrm{ext}} + \underline{1}_0 \otimes (M \, [\![\underline{\hat{w}}]\!]_{I-1}), \tag{5.14b}$$

$$\underline{\boldsymbol{g}}_I = \underline{0}, \tag{5.14c}$$

where $\underline{\boldsymbol{w}}_I^T = \{\underline{\hat{w}}^T(t_{I-1}^+),\ \underline{\hat{w}}^T(t_I^-)\}$ and $\underline{\boldsymbol{\lambda}}_I^T = \{\underline{\lambda}^T(t_{I-1}^+),\ \underline{\lambda}^T(t_I^-)\}$ stack the independent velocities and Lagrange multipliers at the two nodes, respectively;

$$
\begin{aligned}
\underline{\boldsymbol{v}}_I &= \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \otimes \frac{\hat{\underline{q}}_{01}}{2J}, & \mathbb{M} &= \frac{1}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \otimes M, \\
\underline{\boldsymbol{f}}_I^{\text{ext}} &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes \underline{f}^{\text{ext}}\ \mathrm{d}t, & \underline{\boldsymbol{f}}_I^{\text{int}} &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes \underline{f}^{\text{int}}\ \mathrm{d}t, \\
\underline{\boldsymbol{f}}_I^{\text{gyro}} &= -\int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes [\mathrm{diag}(\tilde{w}^i)^T M\, \underline{\hat{w}}]\ \mathrm{d}t, & \underline{\boldsymbol{g}}_I &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes \underline{g}\ \mathrm{d}t, \\
\mathbb{G}_{I1} &= \int_{t_{I-1}^+}^{t_I^-} (l^T l) \otimes G\ \mathrm{d}t.
\end{aligned}
$$

Therein, subscript I indicates quantities of the Ith element. The subscript of matrix $\mathbb{M}$ is omitted because it remains constant for all the elements. All the integrals are evaluated with Gaussian quadrature rules by using two quadrature points.

Linearization of eq. (5.14a) leads to

$$
\begin{aligned}
\Delta\underline{\boldsymbol{w}}_I &= [\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \otimes \mathbb{L}_{vI} + (\underline{\imath}\,\underline{1}_0^T) \otimes \mathbb{T}_{I-1,+}]\underline{\Delta\boldsymbol{u}}_I + [(\underline{\imath}\,\underline{1}_1^T) \otimes \mathbb{T}_{I-1,-}]\underline{\Delta\boldsymbol{u}}_{I-1} \\
&\overset{\text{def}}{=} \mathbb{L}_{I1}\underline{\Delta\boldsymbol{u}}_I + \mathbb{L}_{I0}\underline{\Delta\boldsymbol{u}}_{I-1},
\end{aligned}
\tag{5.15}
$$

where identities (5.10c) and (5.11) are introduced to yield the equality. Linearization of eqs. (5.14b) and (5.14b) leads to

$$
(\mathbb{M} + \mathbb{C}_I)\Delta\underline{\boldsymbol{w}}_I + (\mathbb{K}_{\mathbb{I}} + \mathbb{N}_I)\underline{\Delta\boldsymbol{u}}_I + \mathbb{G}_{I1}^T\Delta\underline{\boldsymbol{\lambda}}_I/s = \underline{\boldsymbol{r}}_f + \underline{1}_0 \otimes M\Delta[\![\underline{\hat{w}}]\!]_{I-1}, \tag{5.16a}
$$

$$
\mathbb{G}_{I2}\underline{\Delta\boldsymbol{u}}_I = \underline{\boldsymbol{r}}_g, \tag{5.16b}
$$

where $\underline{\boldsymbol{r}}_{fI}$ and $\underline{\boldsymbol{r}}_{gI}$ are residuals and

$$
\begin{aligned}
\mathbb{C}_I &= -\int_{t_{I-1}^+}^{t_I^-} (\underline{l}^T \underline{l}) \otimes C \, \mathrm{d}t, \\
\mathbb{K}_I &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes K \, \mathbb{L}_{uI} \, \mathrm{d}t, \\
\mathbb{N}_I &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes N \, \mathbb{L}_{uI} \, \mathrm{d}t, \\
\mathbb{G}_{I2} &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes G \, \mathbb{L}_{uI} \, \mathrm{d}t.
\end{aligned}
$$

Introducing eq. (5.15) into eq. (5.16a) leads to

$$
\begin{bmatrix} -(\underline{1}_0 \underline{1}_1^T) \otimes M \mathbb{L}_{(I-1)0} & \end{bmatrix} \begin{Bmatrix} \underline{\Delta \boldsymbol{u}}_{I-2} \\ \Delta \underline{\boldsymbol{\lambda}}_{I-2}/s \end{Bmatrix} + \\
\begin{bmatrix} [(\underline{1}_0 \underline{1}_0^T) \otimes M + \mathbb{M} + \mathbb{C}_I] \mathbb{L}_{I0} - (\underline{1}_0 \underline{1}_1^T) \otimes M \mathbb{L}_{(I-1)1} & \end{bmatrix} \begin{Bmatrix} \underline{\Delta \boldsymbol{u}}_{I-1} \\ \Delta \underline{\boldsymbol{\lambda}}_{I-1}/s \end{Bmatrix} + \\
\begin{bmatrix} [(\underline{1}_1 \underline{1}_1^T) \otimes M + \mathbb{M}_I + \mathbb{C}_I] \mathbb{L}_{I1} + \mathbb{K}_I + \mathbb{N}_I & s \mathbb{G}_{I1}^T \\ s \mathbb{G}_{I2} & \end{bmatrix} \begin{Bmatrix} \underline{\Delta \boldsymbol{u}}_I \\ \Delta \underline{\boldsymbol{\lambda}}_I/s \end{Bmatrix} = \begin{Bmatrix} \underline{\boldsymbol{r}}_{fI} \\ s \underline{\boldsymbol{r}}_{gI} \end{Bmatrix}, \tag{5.17}
$$

where a scale factor s is introduced to balance the order of magnitude of equilibrium and constraint equations. In initial value problems, terms of $\underline{\boldsymbol{\Delta u}}_{I-2}$ and $\underline{\boldsymbol{\Delta u}}_{I-1}$ do not appear because velocity $\hat{\underline{w}}(t_{I-1}^-)$ and motion tensors $\mathcal{R}^i(t_{I-1}^-)$ are given as constants when solving the equations of the Ith element. The total number of degrees of freedom is $2(6n + m)$ for initial value problems.

5.5 Numerical examples

Numerical examples are presented to validate the proposed formulation. To measure the accuracy of the solver, the L_∞ error are introduced as

$$e_u = \frac{\|\underline{r}(T) - \underline{r}_e(T)\|}{\|\underline{r}_e(T)\|}, \quad e_R = \frac{1}{\sqrt{2}}\|\log[R_e^T(T)R(T)]\|_F,$$
$$e_v = \frac{\|\underline{\dot{r}}(T) - \underline{\dot{r}}_e(T)\|}{\|\underline{\dot{r}}_e(T)\|}, \quad e_\omega = \frac{\|\underline{\omega}(T) - \underline{\omega}_e(T)\|}{\|\underline{\omega}_e(T)\|},$$

where subscript e indicates reference solutions

5.5.1 Dynamic problem of a rigid-body

A rigid-body undergoes the motion defined by the following position vector and 3-1-3 Euler angles

$$\underline{r}_e = \begin{Bmatrix} \sin 2t \\ \cos 2t - 1 \\ 0.5t - \sin 4t \end{Bmatrix} \text{ m}, \quad \underline{\phi}_e = \begin{Bmatrix} 2t \\ 2\sin t \\ 3\cos t - 3 \end{Bmatrix} \text{ rad.}$$

The mass matrix of the rigid body resolved in material basis is

$$M = \begin{bmatrix} mI_3 & m\tilde{\eta}^T \\ m\tilde{\eta} & \underline{\underline{\varrho}} \end{bmatrix}, \quad \underline{\eta} = \begin{Bmatrix} 0.25 \\ -0.03 \\ 0.12 \end{Bmatrix} \text{ m}, \quad \underline{\underline{\varrho}} = \begin{bmatrix} 1.25 & 0.57 & -0.23 \\ 0.57 & 1.56 & 0.34 \\ -0.23 & 0.34 & 2.54 \end{bmatrix} \text{ kg} \cdot \text{m}^2,$$

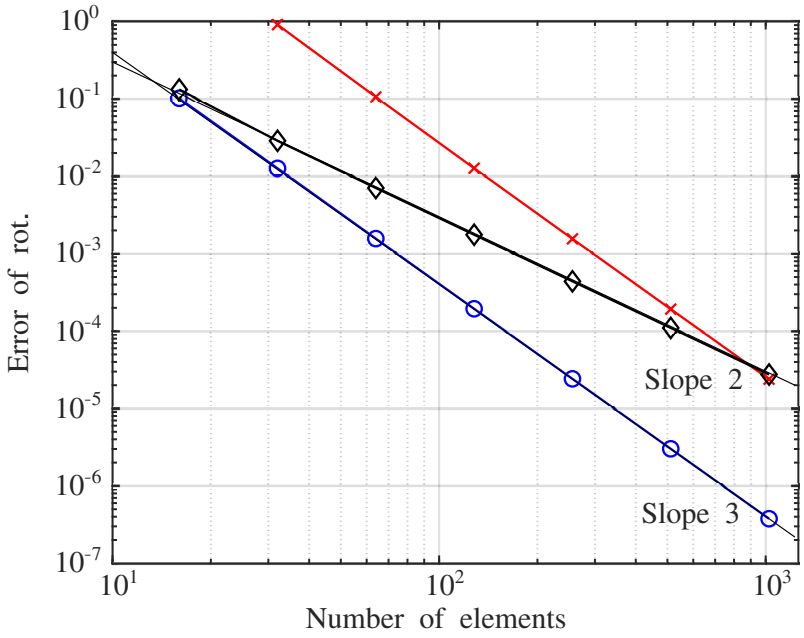

Figure 5.2: Error of rotation versus number of elements for the rigid-body problem. Present: ($\circ$); Radau IIA: ($\times$); generalized-α with $\rho_\infty = 1$: ($\diamond$).

and $m = 6.0$ kg. External forces and moments can then be calculated by the equations of motion as $\underline{f}^{\text{ext}} = M\,\underline{\dot{v}}_e - \tilde{v}_e^T M\,\underline{v}_e$.

Three schemes are used to integrate the dynamic equations over a time period of $T = 4$ s: the generalized-α with $\rho_\infty = 1$ [81], Radau IIA [57], and the proposed discontinuous Galerkin method. The convergence plots for rotation, displacement, angular velocity, and velocity are shown in figs. 5.2, 5.3, 5.4, and 5.5, respectively. The generalized-α scheme is second-order accurate while both Raudau IIA and the proposed approach are third-order accurate. The Raudau IIA scheme is less accurate than the generalized-α scheme for large time steps and becomes superior as time steps decrease. The proposed approach is far more accurate than both of Radau IIA and generalized-α schemes for all considered times step sizes.

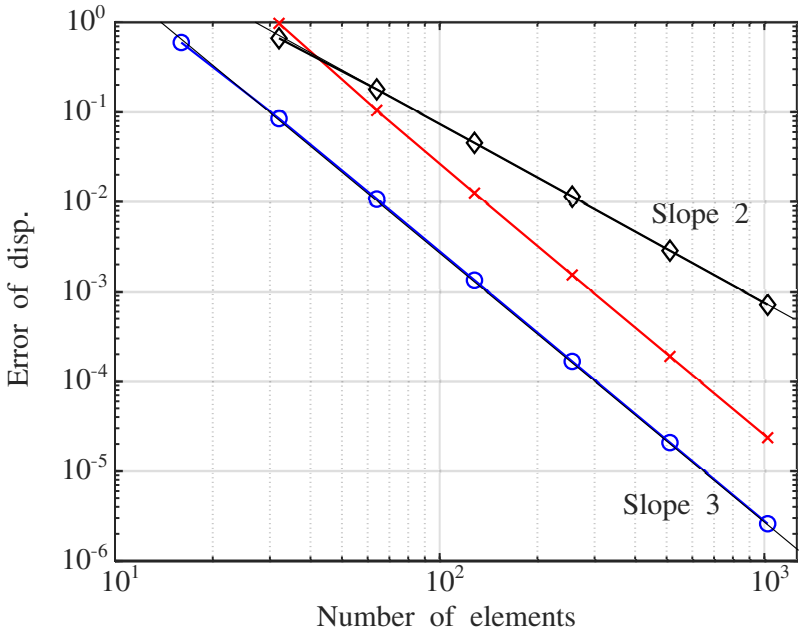

Figure 5.3: Error of displacement versus number of elements for the rigid-body problem. Present: ($\circ$); Radau IIA: ($\times$); generalized-α with $\rho_\infty = 1$: ($\diamond$).

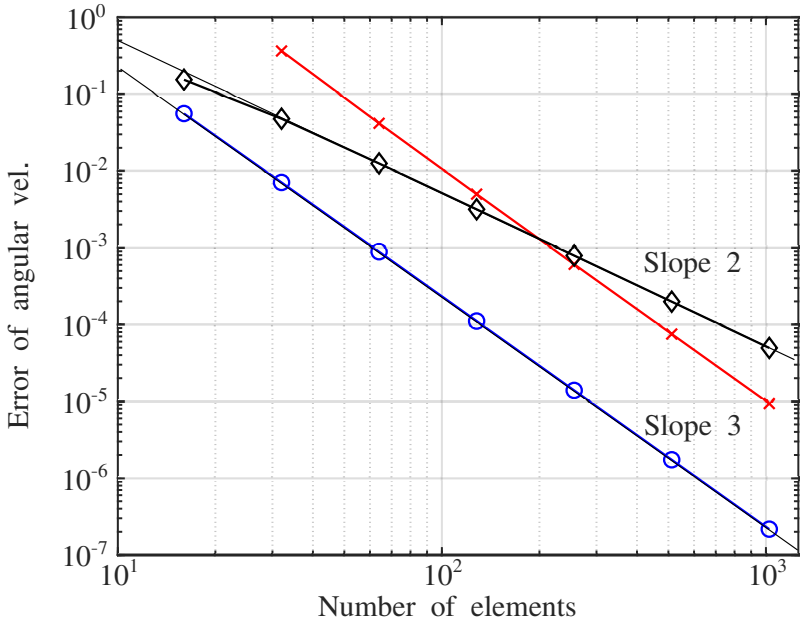

Figure 5.4: Error of angular velocity versus number of elements, rigid-body problem. Present: ($\circ$); Radau IIA: ($\times$); generalized-α with $\rho_\infty = 1$: ($\diamond$).

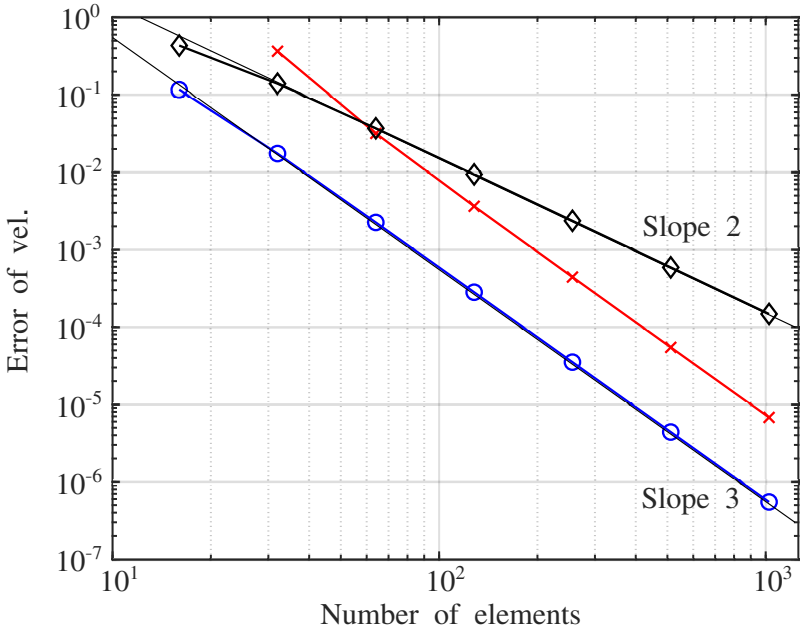

Figure 5.5: Error of velocity versus number of elements for the rigid-body problem. Present: ($\circ$); Radau IIA: ($\times$); generalized-α with $\rho_\infty = 1$: ($\diamond$).

5.5.2 Dynamic problem of a hinged beam

Consider a uniform straight beam, of length 2.4 m, hinged at the root so as to allow rotation about the $\bar{\imath}_3$ axis and free at the tip. The cross-section properties of the beam were given in ref. [157]. The applied loading consists of a tip load

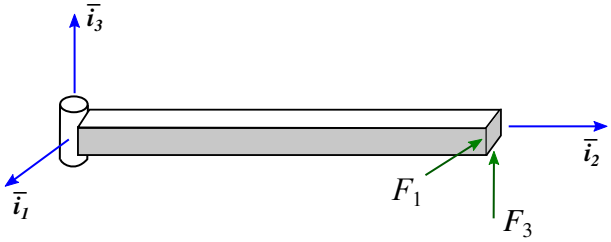

Figure 5.6: The sketch of a flexible beam.

$$\underline{F} = P \begin{Bmatrix} -1 \\ 0 \\ 1 \end{Bmatrix} [1 + \cos(2\pi \frac{t}{T_0} - \pi)], \quad t \leq T_0,$$

$$\underline{F} = \underline{0}, \quad t > T_0,$$

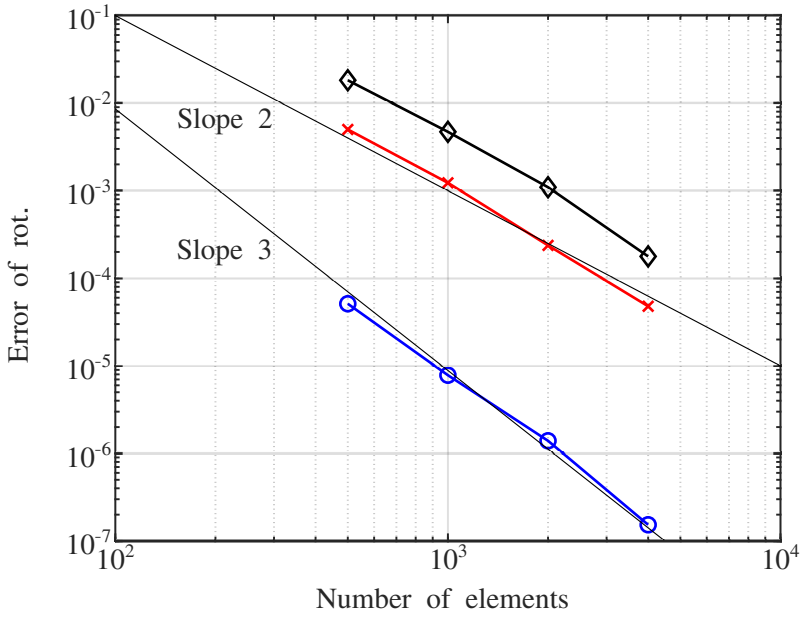

Figure 5.7: Error of rotation versus number of element, hinged beam problem. Present: ($\circ$); generalized-α, $\rho_\infty = 0.5$: ($\times$); generalized-α, $\rho_\infty = 0$: ($\diamond$).

where $P = 1$ kN and $T_0 = 0.022$ s. The beam is initially at rest with its reference line along the $\bar{\imath}_2$ axis. The beam is meshed to 12 two-node finite element for discretization in space.

The generalized-α with $\rho_\infty = 0,\ 0.5$ and proposed DG method are used to integrate the system over a time period of $T = 0.25$ s. For both schemes, the predictions for 8000 time steps (or number of elements), corresponding to a time step length of 3.125 10^{-5} s, are used as reference solutions. The converge plots for rotation and displacement are shown in fig. 5.7 and 5.8, respectively. As expected, generalized-α and DG method are second- and third-order accurate, respectively.

The convergence plots for angular velocity and velocity are shown in figs. 5.9 and 5.10, respectively. While both schemes are second-order accurate, the DG method is far more accurate than generalized-α scheme for the same number of time steps. Also note that the number of unknowns for the DG method is twice

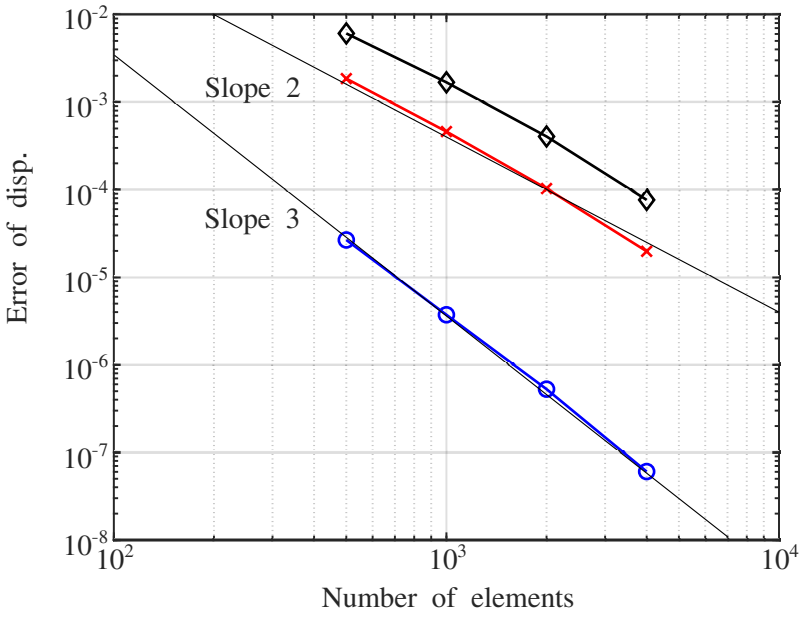

Figure 5.8: Error of displacement versus number of element, hinged beam problem. Present: ($\circ$); generalized-α, $\rho_\infty = 0.5$: ($\times$); generalized-α, $\rho_\infty = 0$: ($\diamond$).

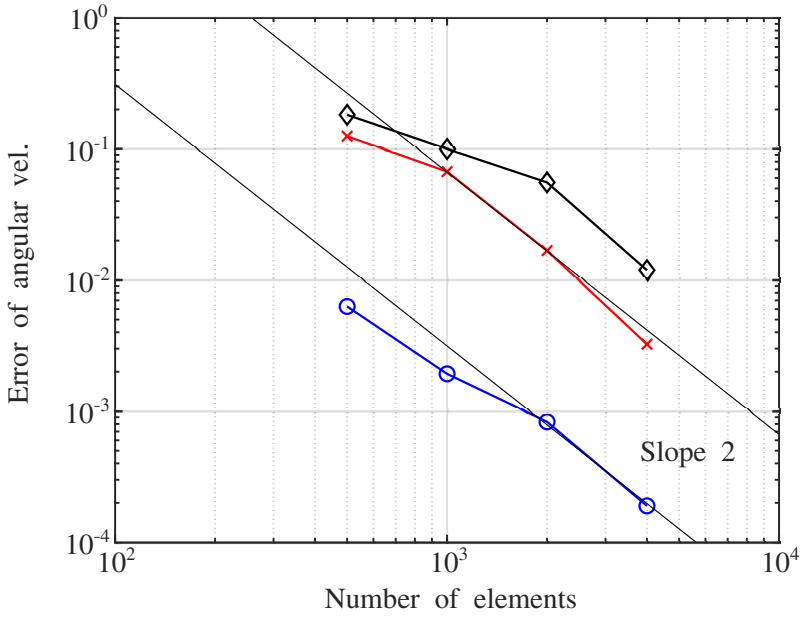

Figure 5.9: Error of angular velocity versus number of element, hinged beam problem. Present: ($\circ$); generalized-α, $\rho_\infty = 0.5$: ($\times$); generalized-α, $\rho_\infty = 0$: ($\diamond$).

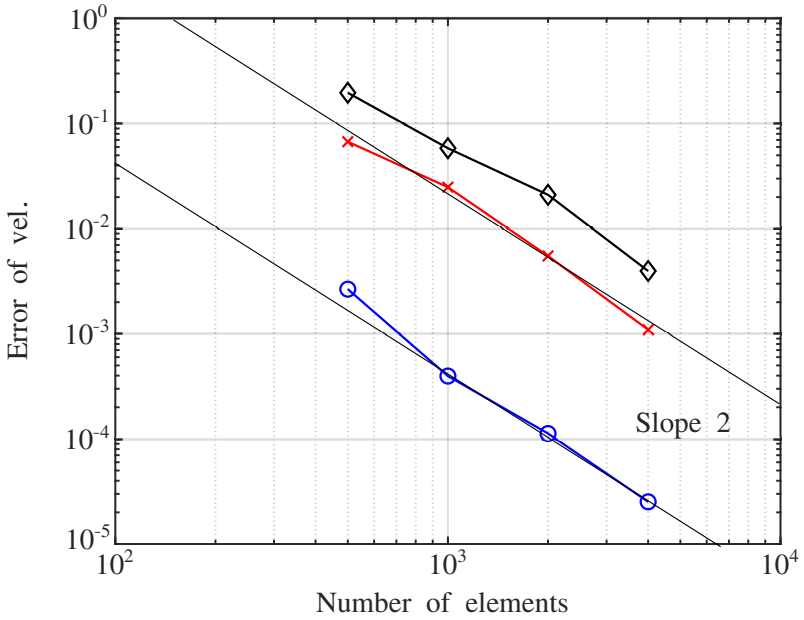

Figure 5.10: Error of velocity versus number of element, hinged beam problem. Present: ($\circ$); generalized-α, $\rho_\infty = 0.5$: ($\times$); generalized-α, $\rho_\infty = 0$: ($\diamond$).

that for the generalized-α scheme. For a fair comparison, the error of displacement by using 10^3 steps is about 1.5 10^{-6} in DG method; the error of using 2 10^3 steps is about 3 10^{-4} in generalized-α scheme, as shown in fig. 5.8. Indeed, the proposed DG method is more efficient than the generalized-α scheme.

5.6 Summary and conclusions

Three types of weak formulations for flexible multibody dynamics are derived based on Hamilton's principle: the *(1)* Bi-discontinuous, *(2)* discontinuous, and *(3)* continuous Galerkin methods. The discontinuous Galerkin method allows jumps of motion and velocity across element boundaries and leads to unconditionally stable schemes for the time integration of dynamics problems. The dual-SLERP is used for interpolation of motion and leads to two-node elements in the time domain.

The proposed scheme is validated by numerical examples. The predicted displacements and rotations are third-order accurate in rigid-body and flexible beam problems; while the velocities and angular velocities are third-order accurate in rigid-body dynamics and second-order accurate in flexible beam dynamics, respectively.

Chapter 6: Galerkin Methods in Time Domain: Periodic Problem and Stability Analysis

The proposed continuous and discontinuous Galerkin formulations presented in Chap. (5) are applied to determination of periodic solutions and stability analysis of these solutions.

6.1 Preliminary on periodic problems

The stability of a periodic solution is determined by the Floquet multipliers of the periodic linear system resulting from the linearization of the original system along this periodic solution. This section reviews Floquet's and Hill's methods for linear periodic systems. The importance of numerical damping in Floquet's method and de-aliasing techniques in Hill's method is illustrated in numerical examples.

6.1.1 Floquet's theorem

Consider a first-order, nonlinear dynamic system $\underline{f}(\dot{\underline{q}}, \underline{q}, t) = \underline{0}$, where $\underline{f}, \underline{q} \in \mathbb{R}^n$. Suppose a $T-$periodic solution $\underline{q}^*(t)$ exists, i.e., $\underline{q}^* = \underline{q}^*(t+T)$. A perturbation about the periodic solution is denoted $\underline{q}(t) = \underline{q}^*(t) + \underline{x}(t)$. Introducing this expression into the governing equation and linearizing about $\underline{q}^*$ then leads to a first-order, linear

periodic system

$$\dot{\underline{x}} = A(t)\underline{x}, \tag{6.1}$$

where matrix $A = (\partial \underline{f}/\partial \dot{\underline{q}})^{-1}\partial \underline{f}/\partial \underline{q}|_{\underline{q}=\underline{q}^*}$ is also $T-$periodic. The fundamental solution matrix, or the state transition matrix, of linear periodic system (6.1) is denoted as $X(t)$, $X(0) = I_n$. Floquet's theorem [159, 160] states that the fundamental solution matrix can be expressed as

$$X(t) = Q(t)\exp(R\,t), \tag{6.2}$$

where $Q(t+T) = Q(t)$ is periodic and matrix R, a function of t_0, is defined as

$$R = \frac{1}{T}\log[X(t_0)^{-1}X(t_0+T)] = \frac{1}{T}\log[C(t_0)].$$

Matrix $C(t_0)$ is known as the monodromy matrix and is $T-$periodic with respect to t_0. For $t_0 = 0$, the monodromy matrix becomes $C(0) = X(0)^{-1}X(T) = X(T)$, which is the state transition matrix from time 0 to time T. In the sequel, $C(0)$ is abbreviated as C .

The eigenvalues of matrices R and C are referred to as the Floquet exponents and multipliers, respectively. Let σ_i and $\lambda_i = \exp(T\sigma_i)$, $i = 1, 2, \ldots, n$ denote the Floquet exponents and multipliers, respectively. The spectral decompositions of matrices C and R are

$$R = V\,\Sigma\,V^{-1}, \quad C = V\,\Lambda\,V^{-1}, \tag{6.3}$$

where $\Sigma = \text{diag}(\sigma_i)$ and $\Lambda = \exp(\Sigma T) = \text{diag}(\lambda_i)$. Introducing the spectral decomposition into eq. (6.2) leads to $X(t)V = Q(t)V \exp(\Sigma t)$. Suppose that $\underline{x}_0 = \underline{v}$, i.e., an arbitrary column of matrix V. The solution of the periodic system becomes

$$\underline{x}(t) = X(t)\underline{v} = Q(t)\underline{v}\exp(\sigma t). \tag{6.4}$$

Clearly, the stability of periodic solutions is determined by the Floquet exponents or multipliers: the solution of the periodic system is stable if $\mathfrak{Re}(\sigma_i) \leq 0$ or $|\lambda_i| \leq 1$ and instable if $\mathfrak{Re}(\sigma_i) > 0$ or $|\lambda_i| > 1$. The approaches to stability analysis fall into two broad categories: Floquet's and Hill's methods.

6.1.2 Floquet's method

Floquet's method is related to the discretiation of periodic system (6.1) in the time domain. Suppose the time span $[pT, (p+1)T]$ is partitioned to K intervals, denoted $[pT + t_{I-1}, pT + t_I]$, where $I = 1, \ldots, K$, $t_0 = 0$, and $t_K = T$. The $p-$period shift vector is denoted $\hat{\underline{x}}_p^T = \{\underline{x}^T(pT+t_0), \underline{x}^T(pT+t_1), \ldots, \underline{x}^T(pT+t_{K-1})\}$.

Discretizing differential equation (6.1) over the entire time span yields

$$\begin{bmatrix} \ddots & \ddots & \ddots & & & & \\ & K_{-2-1} & K_{-1-1} & K_{-10} & & & \\ & & K_{0-1} & K_{00} & K_{01} & & \\ & & & K_{10} & K_{11} & K_{12} & \\ & & & & \ddots & \ddots & \ddots \end{bmatrix} \begin{Bmatrix} \vdots \\ \underline{\hat{x}}_{-2} \\ \underline{\hat{x}}_{-1} \\ \underline{\hat{x}}_{0} \\ \underline{\hat{x}}_{1} \\ \underline{\hat{x}}_{2} \\ \vdots \end{Bmatrix} = \underline{0}, \tag{6.5}$$

where matrices $K_{pq} \in \mathbb{R}^{nK \times nK}$ result from the time discretization process, which could be based on finite difference, finite element, or spectral collocation schemes. Matrices K_{pp-1} and K_{pp+1} represent the coupling of period p with periods $p-1$ and $p+1$, respectively. In view of eq. (6.4), the solution of the periodic system satisfies $\underline{\hat{x}}_p = \lambda^{p-q}\underline{\hat{x}}_q$. Introducing this relationship into the third row of eq. (6.5) leads to quadratic eigenvalue problem

$$(K_{0-1} + \lambda K_{00} + \lambda^2 K_{01})\underline{\hat{x}}_{-1} = \underline{0}, \tag{6.6}$$

where the eigenvalues are the Floquet multipliers. Because eq. (6.6) is a quadratic eigenvalue problem, its solution is arduous. In practice, matrices K_{0-1}, K_{00}, and K_{01} are sparse and taking this sparsity into account leads to more efficient approaches.

Suppose the K intervals are all of equal size, $t_I - t_{I-1} = T/K$. Consider conditionally and unconditionally stable time discretization: the second-order central

and backward difference schemes

$$\begin{aligned}\dot{\underline{x}}(I\Delta t) &= \frac{\underline{x}[(I+1)\Delta t] - \underline{x}[(I-1)\Delta t]}{2\Delta t},\\ \dot{\underline{x}}(I\Delta t) &= \frac{3\underline{x}(I\Delta t)}{2\Delta t} - \frac{2\underline{x}[(I-1)\Delta t]}{\Delta t} + \frac{\underline{x}[(I-2)\Delta t]}{2\Delta t},\end{aligned} \tag{6.7}$$

respectively. A discretized version of the third row of eqs. (6.5) is obtained by introducing these schemes into eq. (6.1) to find

$$\frac{1}{2\Delta t}\left[\begin{array}{c|cccc|c} I_n & 2\Delta t A(t_0) & -I_n & & & \\ & I_n & 2\Delta t A(t_1) & \ddots & & \\ & & \ddots & \ddots & -I_n & \\ & & & I_n & 2\Delta t A(t_{K-1}) & -I_n \end{array}\right] \left\{\begin{array}{c} \underline{x}(t_{-1}) \\ \underline{x}(t_0) \\ \vdots \\ \underline{x}(t_{K-1}) \\ \underline{x}(t_K) \end{array}\right\} = \underline{0}, \tag{6.8}$$

and

$$\frac{1}{2\Delta t}\left[\begin{array}{cc|ccccc} -I_n & 4I_n & 2\Delta t A(t_0) - 6I_n & & & & \\ & -I_n & 4I_n & 2\Delta t A(t_1) - 6I_n & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -I_n & 4I_n & 2\Delta t A(t_{K-1}) - 6I_n \end{array}\right] \left\{\begin{array}{c} \underline{x}(t_{-2}) \\ \underline{x}(t_{-1}) \\ \underline{x}(t_0) \\ \vdots \\ \underline{x}(t_{K-1}) \end{array}\right\} = \underline{0}, \tag{6.9}$$

for the central and backward difference, respectively.

Elimination of intermediate variables $\underline{x}(t_1), \ldots, \underline{x}(t_{K-2})$ yields

$$\begin{Bmatrix} \underline{x}(t_0) \\ \underline{x}(t_{K-1}) \end{Bmatrix} = \begin{bmatrix} P_{00} & P_{0T} \\ P_{T0} & P_{TT} \end{bmatrix} \begin{Bmatrix} \underline{x}(t_{-1}) \\ \underline{x}(t_K) \end{Bmatrix}, \tag{6.10}$$

and

$$\begin{Bmatrix} \underline{x}(t_{K-2}) \\ \underline{x}(t_{K-1}) \end{Bmatrix} = P \begin{Bmatrix} \underline{x}(t_{-2}) \\ \underline{x}(t_{-1}) \end{Bmatrix}, \tag{6.11}$$

for the central and backward difference schemes, respectively. Eq. (6.10) can be rewritten to

$$\begin{Bmatrix} \underline{x}(t_K) \\ \underline{x}(t_{K-1}) \end{Bmatrix} = \begin{bmatrix} -P_{0T}^{-1}P_{00} & P_{0T}^{-1} \\ P_{T0} - P_{TT}P_{0T}^{-1}P_{00} & P_{TT}P_{0T}^{-1} \end{bmatrix} \begin{Bmatrix} \underline{x}(t_0) \\ \underline{x}(t_{-1}) \end{Bmatrix} = P \begin{Bmatrix} \underline{x}(t_0) \\ \underline{x}(t_{-1}) \end{Bmatrix}. \tag{6.12}$$

Clearly, eqs. (6.12) and (6.11) represent the transition relationships from $[\underline{x}(t_{-1}), \underline{x}(t_0)]$ to $[\underline{x}(t_{K-1}), \underline{x}(t_K)]$ and from $[\underline{x}(t_{-2}), \underline{x}(t_{-1})]$ to $[\underline{x}(t_{K-2}), \underline{x}(t_{K-1})]$, respectively. In view of eq. (6.4), the eigenvalues of matrix P are Floquet's multipliers λ.

6.1.3 Hill's method

Hill's method is related to the discretization of periodic system (6.1) in the frequency domain. Introducing a Fourier expansion of periodic term $Q(t)\underline{v}$ into governing equation (6.4) yields

$$\underline{x}(t) = \sum_{k=-\infty}^{\infty} \underline{y}_k \exp[(\sigma + \mathbb{i}k\Omega)t], \tag{6.13}$$

where σ is the Floquet exponent, $\Omega = 2\pi/T$ is the fundamental frequency, and $\underline{y}_k$ are the coefficients of Fourier series. Periodic matrix is also expanded to $A(t) = \sum_{k=-\infty}^{\infty} \hat{A}_k \exp(\mathbb{i} k\Omega t)$. Introducing these expansions into the periodic system (6.1) yields

$$\left(\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \iddots \\ \cdots & \hat{A}_0 + \mathbb{i}\Omega I_n & \hat{A}_{-1} & \hat{A}_{-2} & \cdots \\ \cdots & \hat{A}_1 & \hat{A}_0 & \hat{A}_{-1} & \cdots \\ \cdots & \hat{A}_2 & \hat{A}_1 & \hat{A}_0 - \mathbb{i}\Omega I_n & \cdots \\ \iddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} - \sigma I \right) \begin{Bmatrix} \vdots \\ \underline{y}_{-1} \\ \underline{y}_0 \\ \underline{y}_1 \\ \vdots \end{Bmatrix} = \underline{0}. \tag{6.14}$$

A finite truncation of eq. (6.14) leads to an eigenvalue problem for the Floquet exponents.

The closed-form expressions for the Fourier series of matrix $A(t)$ exist for special cases only, where matrix $A(t)$ involves polynomial or trigonometric functions. In practice, Fourier series are evaluated numerically via discrete Fourier transformation, causing the introduction of aliasing. Suppose the entire period $[0, T]$ is partitioned to n_T intervals of equal length $\Delta t = T/n_T$. Let $t_k = kT/n_T$, $k = 0, 1, \ldots, n_T - 1$ denotes the time grid points. Let $\hat{\underline{x}}^T = \{\underline{x}^T(0), \underline{x}^T(\Delta t), \ldots, \underline{x}^T(T - \Delta t)\}$. The truncated Fourier series expansion of $\hat{\underline{x}}$ in the real domain is

$$\underline{x}(t) = \exp(\sigma t) \left(\underline{y}_0 + \sum_{k=0}^{n_H} [\underline{y}_{kc} \cos(k\Omega t) + \underline{y}_{ks} \sin(k\Omega t)] \right), \tag{6.15}$$

where $n_H = \lceil (n_T - 1)/2 \rceil$ and $\underline{y}_{n_H s}$ vanishes for even n_T. Let

$$\hat{\underline{y}}^T = \{\underline{y}_0^T, \underline{y}_{1c}^T, \underline{y}_{1s}^T, \ldots, \underline{y}_{n_H c}^T, \underline{y}_{n_H s}^T\},$$

it follows

$$\hat{\underline{y}} = (F_{0,n_T} \otimes I_n)\hat{\underline{x}}, \tag{6.16}$$

where $F_{0,n_T} \in \mathbb{R}^{n_T \times n_T}$ is the discrete Fourier transformation operator defined in eq. (B.33), and notation $\otimes$ represents the Kronecker product. If the periodic term $Q(t)\underline{v}$ is composed of harmonics up to the n_Hth, i.e., $Q(t)\underline{v} \in \text{span}[1, \exp(\mathbb{i}\Omega t), \ldots, \exp(\mathbb{i} n_H \Omega t)]$, the coefficients of discretized Fourier transformation coincide with those of infinite expansions, i.e., $\underline{y}_{kc} = (\underline{y}_k + \underline{y}_{-k})/2$ and $\underline{y}_{ks} = (\underline{y}_k - \underline{y}_{-k})/(2\mathbb{i})$. In general, these two sets of coefficients are different. Aliasing is introduced in discrete Fourier transform (6.16) if $Q(t)\underline{v}$ involves harmonics of frequency higher than $n_H\Omega$. Coefficients $\underline{y}_{kc}$ and $\underline{y}_{ks}$ of the low-frequency harmonics are polluted by these high-frequency harmonics [161]. The simplest de-aliasing technique is to set coefficients of higher-order harmonic to zero, i.e., setting $y_{kc} = 0$ and $u_{ks} = 0$ for $k > n_D$, where $n_D < n_H$ is problem dependent.

Introducing truncated Fourier expansion (6.16) into periodic system (6.1) yields

$$\left[(F_{0,n_T} \otimes I_n)\text{diag}[A(t_k)](F_{0,n_T}^{-1} \otimes I_n) - \Omega \otimes I_n - \sigma I\right] \hat{\underline{y}} = \underline{0}, \tag{6.17}$$

where

$$\Omega = \Omega\,\mathrm{diag}\left(0, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \ldots, \begin{bmatrix} 0 & n_H \\ -n_H & 0 \end{bmatrix}\right),$$

for odd n_T and

$$\Omega = \Omega\,\mathrm{diag}\left(0, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \ldots, \begin{bmatrix} 0 & n_H - 1 \\ -n_H + 1 & 0 \end{bmatrix}, 0\right),$$

for even n_T, respectively. Solving eigenvalue problem (6.17) yields the Floquet exponents.

6.1.4 Comparison of Floquet's and Hill's methods: multi-dimensional Mathieu equation

Consider the following multi-dimensional Mathieu equation

$$\ddot{\underline{x}} + \mathrm{diag}[k^4\omega^2(1 - 2\mu/k^2\cos(\Omega t))]\underline{x} = \underline{0}, \tag{6.18}$$

where $\underline{x} \in \mathbb{R}^n$ and $k = 1, 2, \ldots, n$, ω and Ω are the natural and excitation frequencies, respectively. Fig. 6.1 shows the dominant Floquet multiplier for the case of $\mu = 0.15$ and $n = 4$. Four approaches are used: the second-order central and backward difference schemes both with $K = 50$ and Hill's method with $n_T = 13$ ($n_H = 7$) with and without de-aliasing. The dominant Floquet multipliers predicted by the central-difference method is of magnitude 10^{10} because the scheme itself introduces spurious eigenvalues larger than unity for each time step. They are of magnitude

$1e10$ and are not shown in the figure. The dominant eigenvalues blows up quickly in the process of elimination of intermediate variables to form the final state transition matrix in eq. (6.10). The backward-difference method yields accurate predictions for the present problem because this scheme is absolutely stable and introduces numerical dissipation for the high-frequency modes, i.e., for $\omega > \Omega$, see fig. 6.1. Hill's method predicts the theoretical instabilities at $\Omega/(2\omega) = 1/3, 1/2, 1$ but also introduce fictitious instabilities. To avoid aliasing, Fourier coefficients above the $3rd$ harmonics were set to zero; Hill's method then provides accurate predictions for the present problem.

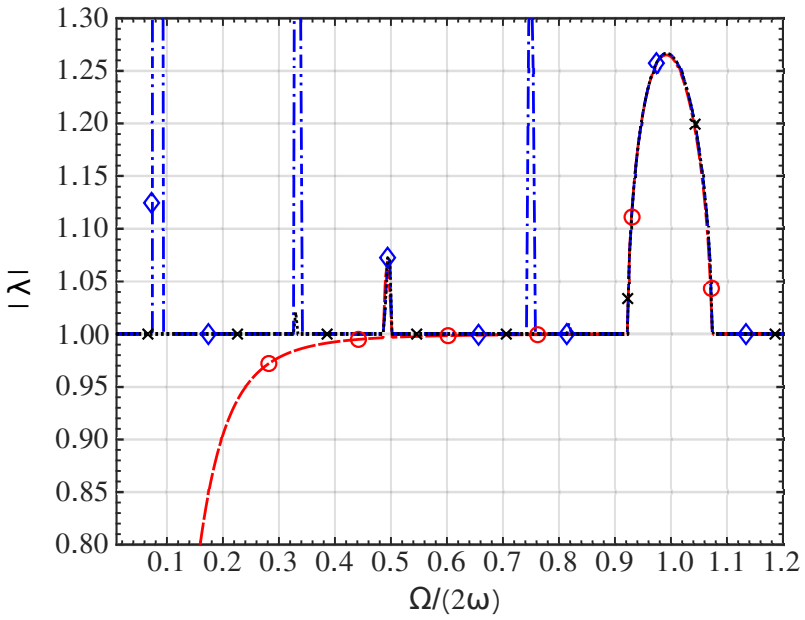

Figure 6.1: The dominant Floquet multiplier for the Multi-dimensional Mathieu equation, backward finite difference: ($\circ$), Hill's method without de-aliasing: ($\diamond$), Hill's method with de-aliasing: ($\times$).

6.2 Continuous Galerkin method

The test and trial functions are the same as those in section 5.3. Introducing the test and trial functions into the weak Hamiltonian formulation (5.8) leads to

$$\underline{\boldsymbol{w}}_I - \underline{\boldsymbol{v}}_I = \underline{0}, \tag{6.19a}$$

$$\mathbb{A}_{I=1}^{K}\left(-\mathbb{M}\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{I1}\underline{\boldsymbol{\lambda}}_I\right) = \mathbb{A}_{I=1}^{K}\underline{\boldsymbol{f}}_I^{\text{ext}}, \tag{6.19b}$$

$$\mathbb{A}_{I=1}^{K}\,\boldsymbol{g}_I = \underline{0}, \tag{6.19c}$$

where notation $\mathbb{A}_{I=1}^{K}$ represents the assembly operator; $\underline{\boldsymbol{w}}_I$, $\underline{\boldsymbol{v}}_I$, $\underline{\boldsymbol{f}}_I^{\text{gyro}}$, $\underline{\boldsymbol{f}}_I^{\text{int}}$, $\mathbb{G}_{I1}$, $\underline{\boldsymbol{f}}_I^{\text{ext}}$, and $\underline{\boldsymbol{g}}_I$ are the same as those in 5.4; matrix $\mathbb{M}$, different with that in chapter 5, is defined as

$$\mathbb{M} = \frac{1}{2}\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix} \otimes M.$$

The independent velocity can be chosen to either continuous or discontinuous across element boundaries. The latter choice is used because it allows a elimination of $\underline{\boldsymbol{w}}_I$ on a element level. After a element-wise elimination, eq. (6.19) becomes

$$\mathbb{A}_{I=1}^{K}(-\mathbb{M}\underline{\boldsymbol{v}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{I1}^{T}\underline{\boldsymbol{\lambda}}_I/s) = \mathbb{A}_{I=1}^{K}\underline{\boldsymbol{f}}_I^{\text{ext}}, \tag{6.20a}$$

$$\mathbb{A}_{I=1}^{K}\,\boldsymbol{g}_I = \underline{0}, \tag{6.20b}$$

Linearization of eq. (6.20) yields

$$\begin{aligned}
&\mathbb{A}_{I=1}^{K}\begin{bmatrix} -\mathbb{M}+\mathbb{C}_I+\mathbb{K}_I+\mathbb{N}_I & s\mathbb{G}_{I1}^T \\ s\mathbb{G}_{I2} & \end{bmatrix}\left(\mathbb{A}_{I=1}^{K}\begin{Bmatrix} \underline{\Delta \boldsymbol{u}}_I \\ \Delta\underline{\boldsymbol{\lambda}}_I/s \end{Bmatrix}\right) \\
&= \begin{bmatrix} -\mathbb{M}_T+\mathbb{C}_T+\mathbb{K}_T+\mathbb{N}_T & s\mathbb{G}_{T1}^T \\ s\mathbb{G}_{T2} & \end{bmatrix}\begin{Bmatrix} \underline{\Delta \boldsymbol{u}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T/s \end{Bmatrix} = \begin{Bmatrix} \underline{\boldsymbol{r}}_{fT} \\ s\underline{\boldsymbol{r}}_{gT} \end{Bmatrix},
\end{aligned} \tag{6.21}$$

where matrices $\mathbb{C}_I$, $\mathbb{K}_I$, $\mathbb{N}_I$, $\mathbb{G}_{I1}$, and $\mathbb{G}_{I2}$ are the same as those defined in section 5.4; notation $(\cdot)_T$ indicates an assembly of the corresponding elements. The increments after assembly are

$$\underline{\boldsymbol{\Delta u}}_T = \begin{Bmatrix} \underline{\hat{\Delta u}}(t_0) \\ \underline{\hat{\Delta u}}(t_1) \\ \vdots \\ \underline{\hat{\Delta u}}(t_{K-1}) \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{\lambda}}_T = \begin{Bmatrix} \Delta\underline{\lambda}(t_0) \\ \Delta\underline{\lambda}(t_1) \\ \vdots \\ \Delta\underline{\lambda}(t_{K-1}) \end{Bmatrix}.$$

Because quantities evaluated at t_I^- and t_I^+ are identical in the continuous Galerkin method, t_I is used for clarity. Clearly, the total number of degrees of freedom is $K(6n+m)$ for periodic boundary value problem.

The sparse patterns of the global Jacobian matrix is shown in fig. 6.2. Note that each block retain its sparsity pattern resulting from the spatial topology of the multibody system.

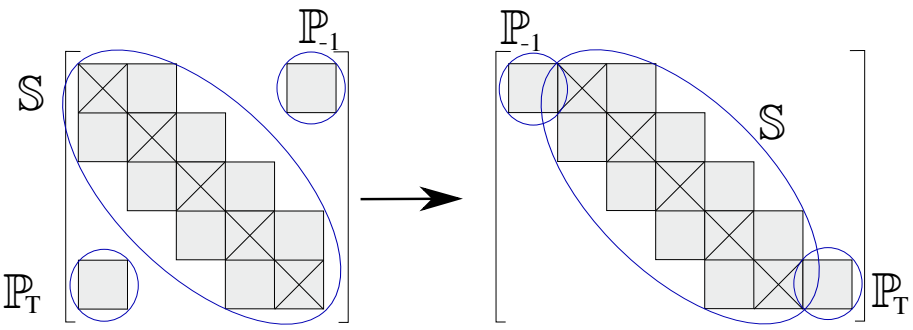

Figure 6.2: The sparse pattern in CG method. Left: Jacobian matrix for solving periodic boundary value problems; right: state transition matrix from time $(t_{-1}, t_0)$to (t_{K-1}, t_K). For this illustration, the entire period was divided into four 2-node elements. The assembled blocks are indicated by $(\times)$.

6.3 Stability analysis of periodic solutions, CG method

The stability analysis is based on linearized governing equation (6.21). To construct the state transition matrix from time t_0 to t_K, the periodic loop is cut at the nodes of these time grids. The periodic loop is cut also because the solution of a linearized periodic system is not periodic, as discussed in section 6.1.1. The linearized governing equation (6.21) is reformulated as

$$\mathbb{S}\begin{Bmatrix} \underline{\boldsymbol{\Delta u}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T/s \end{Bmatrix} = -\begin{bmatrix} \mathbb{P}_{-1} \\ 0 \end{bmatrix}\begin{Bmatrix} \underline{\hat{\Delta u}}(t_{-1}) \\ \Delta\underline{\lambda}(t_{-1})/s \end{Bmatrix} - \begin{bmatrix} 0 \\ \mathbb{P}_T \end{bmatrix}\begin{Bmatrix} \underline{\hat{\Delta u}}(t_K) \\ \Delta\underline{\lambda}(t_K)/s \end{Bmatrix}, \tag{6.22}$$

where the residuals $\underline{\boldsymbol{r}}_{fT}$ and $\underline{\boldsymbol{r}}_{gT}$ vanish because the linearization is about the periodic solution; matrix $\mathbb{S}$, of size $K(6n+m) \times K(6n+m)$, is the block diagonal of the Jacobian matrix in eq. (6.21); matrices $\mathbb{P}_{-1}$ and $\mathbb{P}_T$, both of size $(6n+m) \times (6n+m)$, are the non-vanishing upper-right and lower-left corner blocks of the Jacobian matrix. The construction of state transition matrix is similar to the process used in

the central difference method, see section 6.1.2.

Solving matrix equation. (6.22) and eliminating the intermediate variables lead to

$$\begin{Bmatrix} \hat{\underline{\Delta u}}(t_0) \\ \hat{\underline{\Delta u}}(t_{K-1}) \end{Bmatrix} = \begin{bmatrix} S_{00} & S_{0T} \\ S_{T0} & S_{TT} \end{bmatrix} \begin{Bmatrix} \hat{\underline{\Delta u}}(t_{-1}) \\ \hat{\underline{\Delta u}}(t_K) \end{Bmatrix}, \tag{6.23}$$

where matrices S_{00}, S_{0T}, S_{T0}, and S_{TT}, all of size $(6n+m)\times(6n+m)$, result from the solution of matrix equations. Note the components of incremental motion $\hat{\underline{\Delta u}}(t)$ are not independent because they should satisfy constraint equations $G(t)\,\hat{\underline{\Delta u}}(t) = \underline{0}$. It follows that $\hat{\underline{\Delta u}}(t) = N(t)\underline{p}(t)$, where $N \in \mathbb{R}^{6n\times(6n-m)}$, $G(t)N(t) = 0$, and $N^T N = I$ is the null space of G, and $\underline{p} \in \mathbb{R}^{(6n-m)}$ represents the independent increments. Equation (6.23) then becomes

$$\begin{Bmatrix} \underline{p}(t_0) \\ \underline{p}(t_{K-1}) \end{Bmatrix} = \begin{bmatrix} P_{00} & P_{0T} \\ P_{T0} & P_{TT} \end{bmatrix} \begin{Bmatrix} \underline{p}(t_{-1}) \\ \underline{p}(t_K) \end{Bmatrix}, \tag{6.24}$$

where

$$\begin{bmatrix} P_{00} & P_{0T} \\ P_{T0} & P_{TT} \end{bmatrix} = \begin{bmatrix} N(t_0) & \\ & N(t_{K-1}) \end{bmatrix}^T \begin{bmatrix} S_{00} & S_{0T} \\ S_{T0} & S_{TT} \end{bmatrix} \begin{bmatrix} N(t_{K-1}) & \\ & N(t_0) \end{bmatrix}.$$

Note that the null space at $t_{\mathrm{mod}(i,K)}$ are the same as that at t_i. Finally, the transition matrix of the independent state variables from time (t_{-1}, t_0) to time (t_{K-1}, t_K) can

be found as

$$\begin{Bmatrix} \underline{p}(t_K) \\ \underline{p}(t_{K-1}) \end{Bmatrix} = \underbrace{\begin{bmatrix} -P_{0T}^{-1}P_{00} & P_{0T}^{-1} \\ P_{T0} - P_{TT}P_{0T}^{-1}P_{00} & P_{TT}P_{0T}^{-1} \end{bmatrix}}_{\text{denoted } P} \begin{Bmatrix} \underline{p}(t_0) \\ \underline{p}(t_{-1}) \end{Bmatrix}. \tag{6.25}$$

The eigenvalues of matrix P are Floquet multipliers.

6.4 Post processing of CG method

As shown in eq. (6.19a), the independent velocity vectors at the nodes are $\underline{\hat{w}}(t_{I-1}^+) = \underline{\hat{w}}(t_I^-) = \underline{\hat{q}}_{01}/(2J)$, and hence, velocities remain constant within each element, i.e., $\underline{\hat{w}}(t) = \underline{\ell} \otimes I_{6n}\underline{\boldsymbol{w}}_I = \underline{\hat{q}}_{01}/(2J)$. This fact indicates that velocities converge with order 1. To increase the order of convergence, a re-interpolation is introduced. Let $\underline{\hat{w}}^*$ denote the corrected velocity and it is determined by interpolation $\underline{\hat{w}}^* = l \otimes I_{6n}\underline{\boldsymbol{w}}_I^*$, where $\underline{\boldsymbol{w}}_I^*$ stacks the corrected nodal velocities. The corrected nodal velocity is determined by

$$\mathbb{A}_{I=1}^K(\int_{t_{I-1}^+}^{t_I^-} \underline{l}^T\underline{l}\,\mathrm{d}t\otimes\underline{\boldsymbol{w}}_I) = \mathbb{A}_{I=1}^K(\int_{t_{I-1}^+}^{t_I^-} \underline{l}^T\underline{l}\mathrm{d}t\otimes\underline{\boldsymbol{w}}_I^*) = (\mathbb{A}_{I=1}^K \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T\underline{l}\,\mathrm{d}t\otimes I_{6n})\underline{\boldsymbol{w}}_T^*, \tag{6.26}$$

where subscript T represent a collection of nodal quantities in one period. Finally, the velocity field inside each element is $\underline{\hat{w}}^*(t) = (\underline{l}\otimes I_{6n})\underline{\boldsymbol{w}}_I^*$. The corrected velocities converge with order 2, as will be shown in numerical examples.

6.5 Discontinuous Galerkin method

An assembly of governing equations of each element (5.14) leads to the equations for periodic problems

$$\underline{\boldsymbol{w}}_I - \underline{\boldsymbol{v}}_I = \underline{\imath} \otimes [\![\underline{\hat{q}}]\!]_{I-1}, \tag{6.27a}$$

$$\mathbb{A}_{I=1}^{K}\left[\mathbb{M}\underline{\dot{\boldsymbol{w}}}_I + \underline{\boldsymbol{f}}_I^{\mathrm{gyro}} + \underline{\boldsymbol{f}}_I^{\mathrm{int}} + \mathbb{G}_{I1}^T\underline{\boldsymbol{\lambda}}_I\right] = \mathbb{A}_{I=1}^{K}\left[\underline{\boldsymbol{f}}_I^{\mathrm{ext}} + \underline{1}_0 \otimes (M\,[\![\underline{\hat{w}}]\!]_{I-1})\right], \tag{6.27b}$$

$$\mathbb{A}_{I=1}^{K}\underline{\boldsymbol{g}}_I = \underline{0}. \tag{6.27c}$$

An assembly of the linearized governing equations (5.17) of each element leads to the linearization of eqs. (6.27)

$$\left(\begin{bmatrix} \mathrm{diag}_{-2}\left(-(\underline{1}_0\underline{1}_1^T)\otimes M\mathbb{L}_{(I-1)0}\right) \\ \end{bmatrix} + \right.$$
$$\begin{bmatrix} \mathrm{diag}_{-1}\left([(\underline{1}_0\underline{1}_0^T)\otimes M + \mathbb{M} + \mathbb{C}_I]\mathbb{L}_{I0} - (\underline{1}_0\underline{1}_1^T)\otimes M\mathbb{L}_{(I-1)1}\right) \\ \end{bmatrix} +$$
$$\left.\begin{bmatrix} \mathrm{diag}\left([(\underline{1}_1\underline{1}_1^T)\otimes M + \mathbb{M}_I + \mathbb{C}_I]\mathbb{L}_{I1} + \mathbb{K}_I + \mathbb{N}_I\right) & \mathrm{diag}\left(s\mathbb{G}_{I1}^T\right) \\ \mathrm{diag}\left(s\mathbb{G}_{I2}\right) & \end{bmatrix}\right)\begin{Bmatrix} \underline{\Delta\boldsymbol{u}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T/s \end{Bmatrix} = \begin{Bmatrix} \underline{\boldsymbol{r}}_{fT} \\ s\underline{\boldsymbol{r}}_{gT} \end{Bmatrix}, \tag{6.28}$$

where block matrices $\mathrm{diag}(\bullet_I)$, $\mathrm{diag}_{-1}(\bullet_I)$, and $\mathrm{diag}_{-2}(\bullet_I)$ are defined as

$$\mathrm{diag}(\bullet_I) = \begin{bmatrix} \bullet_1 & & \\ & \ddots & \\ & & \bullet_K \end{bmatrix}, \quad \mathrm{diag}_{-1}(\bullet_I) = \left[\begin{array}{ccc|c} & & & \bullet_1 \\ \hline \bullet_2 & & & \\ & \ddots & & \\ & & \bullet_K & \end{array}\right],$$

$$\mathrm{diag}_{-2}(\bullet_I) = \left[\begin{array}{ccc|c} & & & \bullet_1 \\ & & & \bullet_2 \\ \hline \bullet_3 & & & \\ & \ddots & & \\ & & \bullet_K & \end{array}\right].$$

The sparsity pattern of the Jacobian matrix after assembly is shown in fig. 6.3.

The increments with subscript T are defined as

$$\underline{\boldsymbol{\Delta u}}_T = \begin{Bmatrix} \underline{\hat{\Delta u}}(t_0^+) \\ \underline{\hat{\Delta u}}(t_1^-) \\ \vdots \\ \underline{\hat{\Delta u}}(t_{K-1}^+) \\ \underline{\hat{\Delta u}}(t_K^-) \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{\lambda}}_T = \begin{Bmatrix} \Delta\underline{\lambda}(t_0^+) \\ \Delta\underline{\lambda}(t_1^-) \\ \vdots \\ \Delta\underline{\lambda}(t_{K-1}^+) \\ \Delta\underline{\lambda}(t_K^-) \end{Bmatrix}. \tag{6.29}$$

The total number of degrees of freedom is $2K(6n+m)$ for periodic boundary value problems.

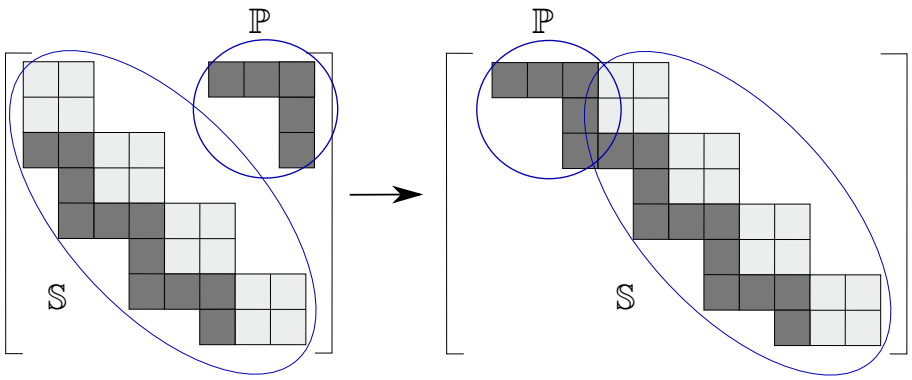

Figure 6.3: The sparsity pattern in the DG method. Block diagonal matrix diag$(\cdot)$: grey; block lower sub-diagonal matrix $\mathrm{diag}_{-1}(\cdot)$ and $\mathrm{diag}_{-2}(\cdot)$: dark grey. Left: Jacobian matrix for solving periodic boundary value problem; right: state transition matrix from time $(t_{-1}^-, t_{-1}^+, t_0^-)$ to $(t_{K-1}^-, t_{K-1}^+, t_K^-)$. The entire period is divided to 4 2-node elements.

6.6 Stability analysis of periodic solutions, DG method

The stability analysis is based on the linearized governing equation (6.28). Similarly as in CG formulation, the periodic loop is cut at the nodes of time grids t_{K-1}^-, t_{K-1}^+, and t_K^-. For convenience, increments with subscript $(p,-)$ are introduced

$$\underline{\boldsymbol{\Delta u}}_{p,-} = \begin{Bmatrix} \underline{\hat{\Delta u}}(pT + t_{-1}^-) \\ \underline{\hat{\Delta u}}(pT + t_{-1}^+) \\ \underline{\hat{\Delta u}}(pT + t_0^-) \end{Bmatrix}, \quad \text{and} \quad \Delta\underline{\boldsymbol{\lambda}}_{p,-} = \begin{Bmatrix} \Delta\underline{\lambda}(pT + t_{-1}^-) \\ \Delta\underline{\lambda}(pT + t_{-1}^+) \\ \Delta\underline{\lambda}(pT + t_0^-) \end{Bmatrix}.$$

Linearized governing equation (6.28) is reformulated as

$$\mathbb{S} \begin{Bmatrix} \underline{\boldsymbol{\Delta u}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T / s \end{Bmatrix} = - \begin{bmatrix} \mathbb{P} \\ 0 \end{bmatrix} \begin{Bmatrix} \underline{\boldsymbol{\Delta u}}_{0,-} \\ \Delta\underline{\boldsymbol{\lambda}}_{0,-} / s \end{Bmatrix}, \tag{6.30}$$

where the residuals vanish because the linearization is about the periodic solution; matrix $\mathbb{S}$, of size $2K(6n+m) \times 2K(6n+m)$, is composed of the diagonal and sub-diagonal blocks of the Jacobian matrix in eq. (6.28); matrix $\mathbb{P}$, of size $3(6n+m) \times 3(6n+m)$, is the non-vanishing upper-right corner blocks of the Jacobian matrix. The construction of state transition relationship is similar to the process used in the backward difference method discussed in section 6.1.2. Elimination of the intermediate variables yields

$$S \begin{Bmatrix} \underline{\boldsymbol{\Delta u}}_{1,-} \\ \Delta \underline{\boldsymbol{\lambda}}_{1,-}/s \end{Bmatrix} = P \begin{Bmatrix} \underline{\boldsymbol{\Delta u}}_{0,-} \\ \Delta \underline{\boldsymbol{\lambda}}_{0,-}/s \end{Bmatrix}. \tag{6.31}$$

The eigenvalues of matrix $S \setminus P$ are the Floquet multipliers.

Because only the dominant eigenvalue of the state transition matrix is evaluated for stability analysis, the Arnoldi process can be used efficiently. It is not necessary to evaluate matrices S and$P$ in eq. (6.31) explicitly. The only computational expensive operation in algorithm (6.6) is the factorization of matrix $\mathbb{S}$.

6.7 Post processing of DG method

To construct continuous kinematic solutions, the re-interpolation in section 6.4 can be used. Another approach is to use the Radau correction. In view of property of Radau polynomials, identity (B.21b) in appendix B.5, the jump terms can be

Algorithm 2: Arnoldi iteration

Factorize matrix $\mathbb{S}$
Choose an arbitrary starting vector $\underline{q}_0 \in \mathbb{R}^{3(6n+m)}$
$\underline{q}_1 = \underline{q}_0 / \|\underline{q}_0\|$
for $j = 1$ to ℓ **do**
    $\underline{v} = \mathbb{P}\underline{q}_j$
    Solve $\mathbb{S}\,\underline{w} = -\begin{Bmatrix} \underline{v} \\ \underline{0} \end{Bmatrix}$ with a forward and backward substitution, where $\underline{0} \in \mathbb{R}^{(2K-3)(6n+m)}$
    Select the last $3(6n+m)$ entries from $\underline{w}$, denote as $\underline{u}$, and let $\underline{r} - \underline{u}$
    for $i = 1$ to j **do**
        $h_{i,j} = \underline{q}_i^T \underline{u}$
        $\underline{r} = \underline{r} - h_{i,j}\underline{q}_i$
    end for
    $h_{j+1,i} = \underline{r}$
    $\underline{q}_{j+1} = \underline{r} / h_{j+1,j}$
    Solve dominant eigenvalue of matrix $H_j \underline{s} = \lambda \underline{s}$
    if $\|h_{j+1,j} s_j\| \leq$ error bound **then**
        return λ
    end if
end for

transformed to a integral over a element

$$\underline{1}_0 \otimes [\![\underline{\hat{q}}]\!]_{I-1} = -\int_{t_{I-1}^+}^{t_I^-} \frac{1}{J}\underline{l}^T \bar{r}_2' \,\mathrm{d}t \otimes [\![\underline{\hat{q}}]\!]_{I-1}, \tag{6.32a}$$

$$\underline{1}_0 \otimes [\![\underline{\hat{w}}]\!]_{I-1} = -\int_{t_{I-1}^+}^{t_I^-} \frac{1}{J}\underline{l}^T \bar{r}_2' \,\mathrm{d}t \otimes [\![\underline{\hat{w}}]\!]_{I-1}, \tag{6.32b}$$

where $\bar{r}_2$ is the right Radau polynomial of degree 2. Introducing these two identities into eq. (5.14b) leads to

$$\begin{aligned}\int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes M(\frac{1}{J}\underline{l}' \otimes I_{6n}\underline{\boldsymbol{w}}_I - \frac{1}{J}\bar{r}_2'[\![\underline{\hat{w}}]\!]_{I-1})\mathrm{d}t &= \int_{t_{I-1}^+}^{t_I^-} \underline{l}^T \otimes (M\,\dot{\underline{\hat{w}}}^*)\,\mathrm{d}t \\ &= \underline{\boldsymbol{f}}_I^{\text{ext}} + \underline{\boldsymbol{f}}_I^{\text{gyro}} - \underline{\boldsymbol{f}}_I^{\text{int}} - \mathbb{G}_{I1}^T\underline{\boldsymbol{\lambda}}_I,\end{aligned} \tag{6.33}$$

where the corrected velocity $\underline{\hat{w}}^*$ is

$$\underline{\hat{w}}^*(t) = \underline{l} \otimes I_{6n}\underline{\boldsymbol{w}}_I - \bar{r}_2[\![\underline{\hat{w}}]\!]_{I-1}. \tag{6.34}$$

Clearly, jump term $[\![\underline{\hat{w}}]\!]_{I-1}$ adds a correction to the velocity field inside each element. The velocity field becomes continuous across element boundaries after correction, i.e., $\underline{\hat{w}}^*(t_{I-1}^+) = \underline{\hat{w}}(t_{I-1}^+) - [\underline{\hat{w}}(t_{I-1}^+) - \underline{\hat{w}}(t_{I-1}^-]$.

Introducing identities (6.32) into eq. (5.14a) and splitting the equation for each structural nodes lead to

$$\int_{t_{I-1}^+}^{t_I^-} \underline{\ell}^T \otimes (\frac{1}{2J}\underline{q}_{01}^i - \frac{1}{J}\bar{r}_2'[\![\underline{q}^i]\!]_{I-1})\,\mathrm{d}t = \int_{t_{I-1}^+}^{t_I^-} \underline{\ell}^T\underline{\ell}\,\mathrm{d}t \otimes \underline{\boldsymbol{w}}_I^i. \tag{6.35}$$

Clearly, motion tensors $\mathcal{R}^{i*}$, $i = 1, 2, \ldots, n$, would be the corrected motion if they

satisfying differential equation

$$(\mathcal{R}^{i*})^T\dot{\mathcal{R}}^{i*} = \frac{1}{2J}\tilde{q}^i_{01} - \frac{1}{J}\bar{r}'_2[\![\tilde{q}^i]\!]_{I-1}, \tag{6.36}$$

with boundary condition $\mathcal{R}^{i*}(t^+_{I-1}) = \mathcal{R}^i(t^-_{I-1})$ and $\mathcal{R}^{i*}(t^-_I) = \mathcal{R}^i(t^-_I)$. However, such solutions do not exist in general and there is no Radau corrections for motion tensors. If the unknowns are vectors rather than motion tensors, eq. (6.36) becomes $\dot{\underline{q}}^{i*} = (\underline{q}^i_1 - \underline{q}^i_0)/2J - \bar{r}'_2/J\bar{r}_2[\![\underline{q}^i]\!]_{I-1}$, and the corrected solutions are

$$\underline{q}^*_i = \underline{l} \otimes I_6\underline{q}^i_I - \bar{r}_2[\![\underline{q}^i]\!]_{I-1}. \tag{6.37}$$

6.8 Numerical examples

Numerical examples are presented to validate the proposed formulation. The first three are periodic boundary value problems, and the last three are stability analyses. To measure the accuracy of the solver, the L_2 error are introduced as

$$e_u = \left(\frac{\int_0^T \|\underline{r} - \underline{r}_e\|^2 \,\mathrm{d}t}{\int_0^T \|\underline{r}_e\|^2 \,\mathrm{d}t}\right)^{1/2}, \quad e_R = \left(\frac{1}{2T}\int_0^T \|\log(R_e^T R)\|_F^2 \,\mathrm{d}t\right)^{1/2},$$
$$e_v = \left(\frac{\int_0^T \|\dot{\underline{r}} - \dot{\underline{r}}_e\|^2 \,\mathrm{d}t}{\int_0^T \|\dot{\underline{r}}_e\|^2 \,\mathrm{d}t}\right)^{1/2}, \quad e_\omega = \left(\frac{\int_0^T \|\underline{\omega} - \underline{\omega}_e\|^2 \,\mathrm{d}t}{\int_0^T \|\underline{\omega}_e\|^2 \,\mathrm{d}t}\right)^{1/2}.$$

Therein, subscript e indicates reference solutions and Gaussian quadrature is used to evaluate the integrals.

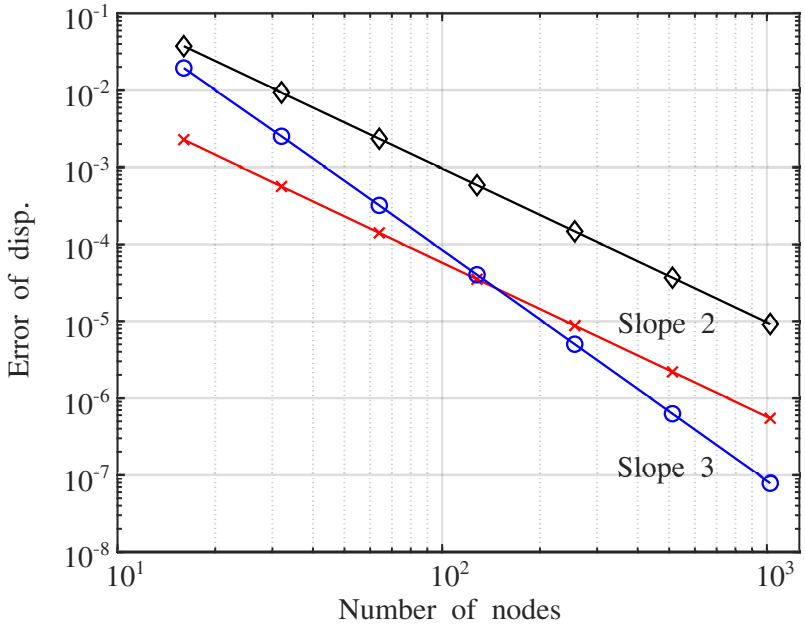

Figure 6.4: Error of displacement versus number of nodes, non-stiff problem, $(\Omega/\omega)^2 = 2$. CG: ($\times$); DG: ($\diamond$); DG with Radau correction: ($\circ$).

6.8.1 Periodic problem of a mass-spring system

Consider a simple mass-spring system governed by equation $\ddot{x}+\Omega^2 x = f\cos(\omega t)$. Two cases are considered: *(1)* $(\Omega/\omega)^2 = 2$ and *(2)* $(\Omega/\omega)^2 = 2\ 10^7$. The latter case is introduced to investigate the behavior of stiff problems. Two types of corrections are used: *(1)* re-interpolation using eq. (6.26) and *(2)* Radau correction for velocity and displacement, see eqs. (6.34) and (6.37), respectively.

Figs. 6.4 and 6.5 show the convergence behavior for the displacement and velocity fields of the non-stiff problem, respectively. In the CG method, the displacement field exhibit second-order convergence because of the linear interpolation scheme; the velocity field exhibits first-order convergence and re-interpolation leads to second-order convergence. In the DG method, both of displacement and velocity fields exhibit second- and third-order convergence, without and with Radau

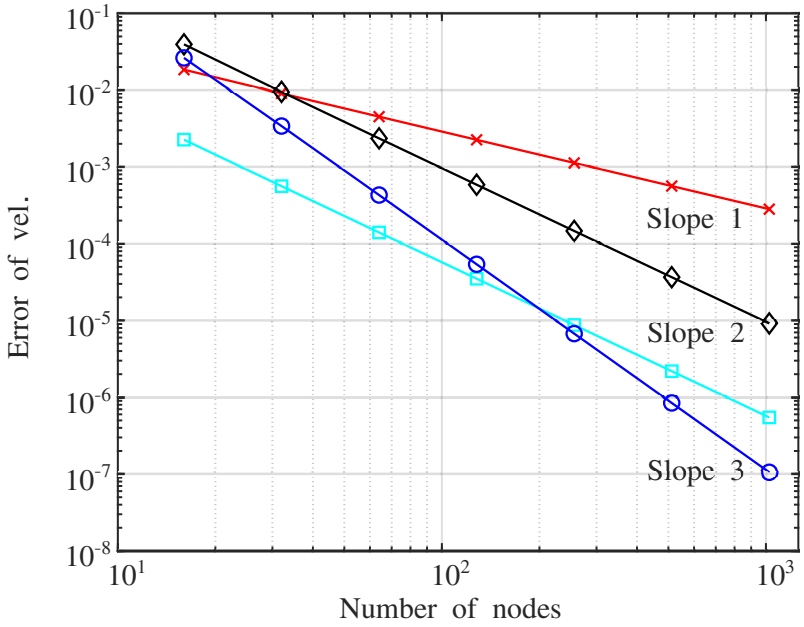

Figure 6.5: Error of velocity versus number of nodes, non-stiff problem, $(\Omega/\omega)^2 = 2$. CG: ($\times$); CG with re-interpolation: ($\square$); DG : ($\diamond$); DG with Radau correction: ($\circ$).

correction, respectively.

The convergence behavior for stiff problems is shown in figs. (6.6) and (6.7). The CG method exhibits the same order of convergence as for non-stiff problems, for both displacement and velocity fields. In the DG method, the displacement field exhibits second-order convergence and Radau corrections do not improve the convergence rate; the velocity field exhibits first-order convergence and while Radau corrections improve the accuracy slightly, the convergence order is not increased; re-interpolation increases the convergence rate from first to second order for the velocity field. As shown in figs. (6.6) and (6.7), the CG method is more accurate than its DG counterpart for stiff problem, when an equal number of nodes are used.

Figs. (6.8) and (6.9) show the non-dimensional velocity field predicted using eight 2-node elements for the non-stiff and stiff problems, respectively. For non-stiff problems, the jumps at element boundaries are not significant and Radau correc-

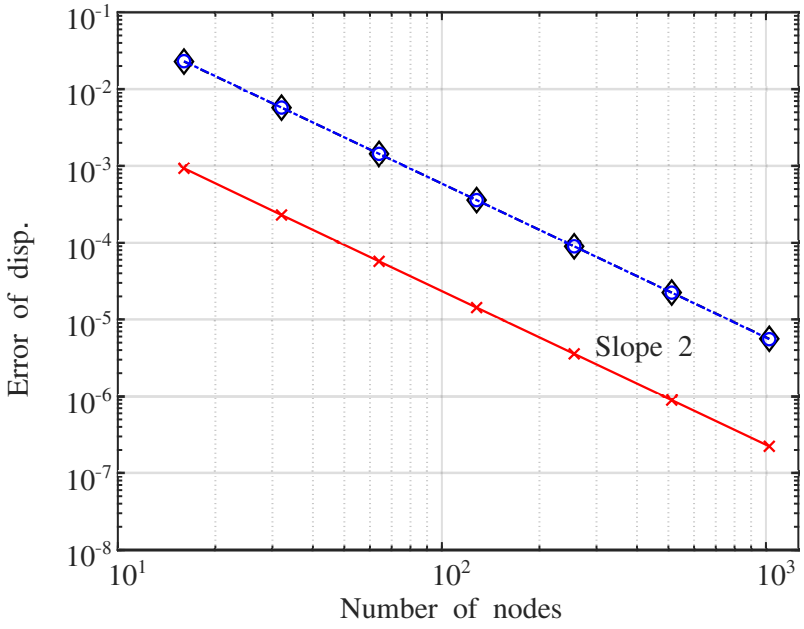

Figure 6.6: Error of displacement versus number of nodes, stiff problem, $(\Omega/\omega)^2 = 2\ 10^7$. CG: ($\times$); DG: ($\diamond$); DG with Radau correction: ($\circ$).

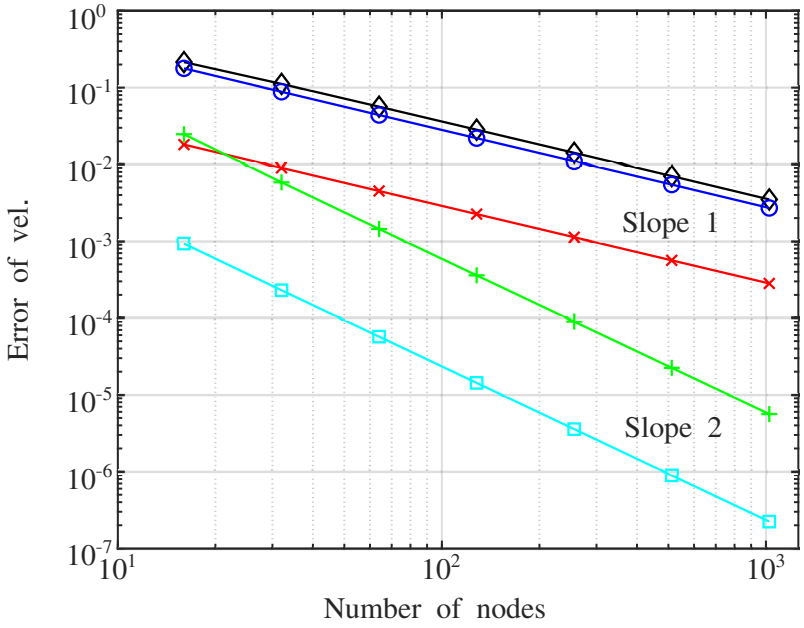

Figure 6.7: Error of velocity versus number of nodes, stiff problem, $(\Omega/\omega)^2 = 2\ 10^7$. CG: ($\times$); CG with re-interpolation: ($\square$); DG : ($\diamond$); DG with Radau correction: ($\circ$); DG with re-interpolation: ($+$).

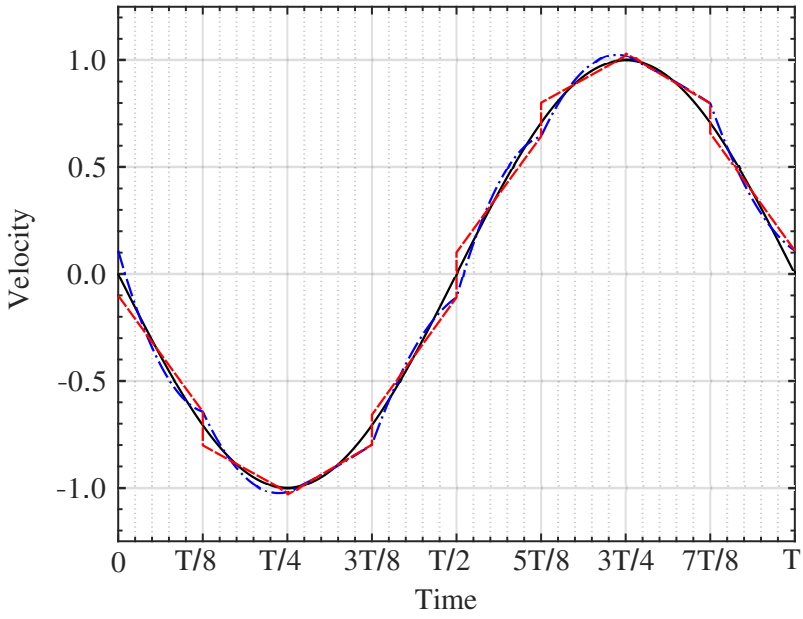

Figure 6.8: The velocity over one period, non-stiff problem, $(\Omega/\omega)^2 = 2$. Analytic: solid line; DG: dashed line; DG with Radau correction: dashed-dotted line.

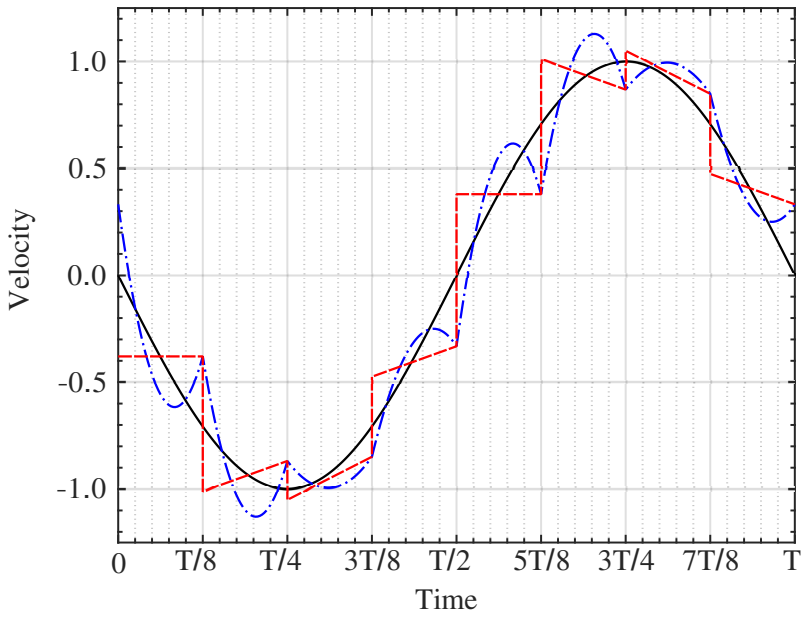

Figure 6.9: The velocity over one period, stiff problem, $(\Omega/\omega)^2 = 2\ 10^7$. Analytic: solid line; DG: dashed line; DG with Radau correction: dashed-dotted line.

tions improve the predictions dramatically. For stiff problems, however, the jump magnitudes are far larger than those observed for non-stiff problems and Radau corrections do not improve the predictions.

6.8.2 Periodic problem of a rigid body

Consider a rigid-body undergoing the periodic motion described by the following position vector and 3-1-3 Euler angles

$$\underline{r}_e = \begin{Bmatrix} \sin 2t \\ \cos 4t \\ \sin 6t \end{Bmatrix} \text{ m}, \quad \text{and} \quad \underline{\phi}_e = \begin{Bmatrix} \cos t \\ 2\sin t \\ 3\cos t \end{Bmatrix} \text{ rad},$$

respectively. The mass matrix of the rigid body is given in section 5.5.1. External forces and moments can be calculated from the equations of motion. These external forces and moments are now applied to the rigid body and the proposed schemes are used to predict its dynamic response.

The convergence plots for rotation, displacement, angular velocity, and velocity are shown in figs. 6.10, 6.11, 6.12 and 6.13, respectively. Because no stiffness is involved in this problem, the DG method and Radau corrections are expected to behave well. The CG method exhibit second-order convergence for both of rotation and displacement fields; the same order of convergence is also observed for the angular velocity and velocity fields when re-interpolations corrections are used. In DG method, order-two convergence is observed for both of rotation and displacement

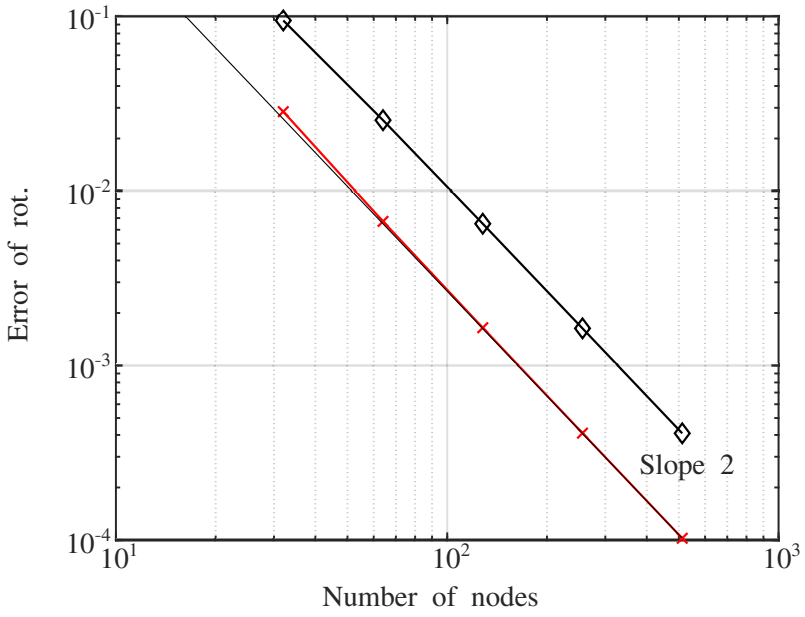

Figure 6.10: Error of rotation versus number of nodes, periodic rigid-body problem. CG: ($\times$); DG: ($\diamond$).

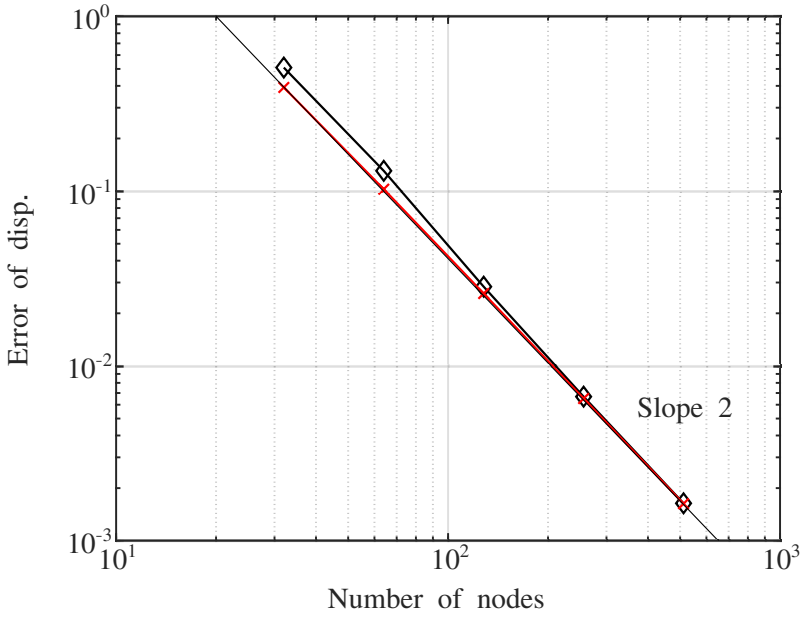

Figure 6.11: Error of displacement versus number of nodes, periodic rigid-body problem. CG: ($\times$); DG: ($\diamond$).

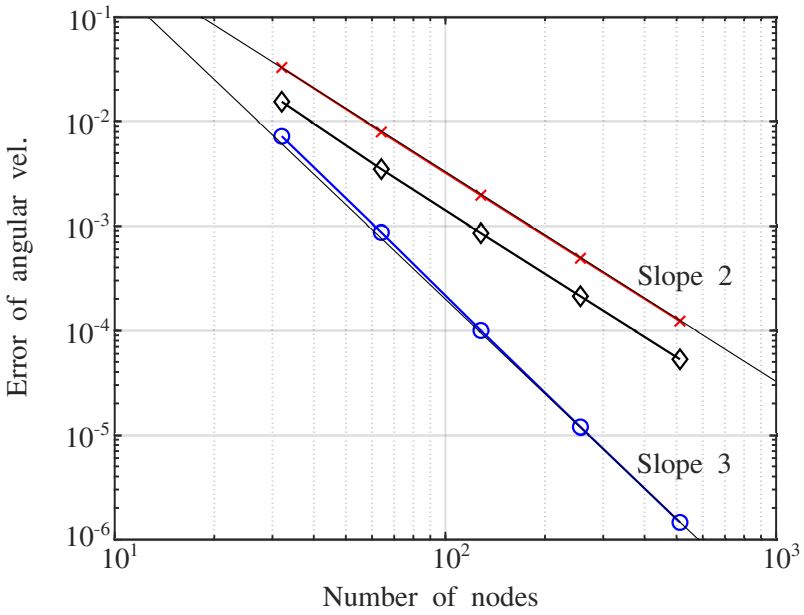

Figure 6.12: Error of angular velocity versus number of nodes, periodic rigid-body problem. CG with re-interpolation: ($\times$); DG: ($\diamond$); DG with Radau correction: ($\circ$).

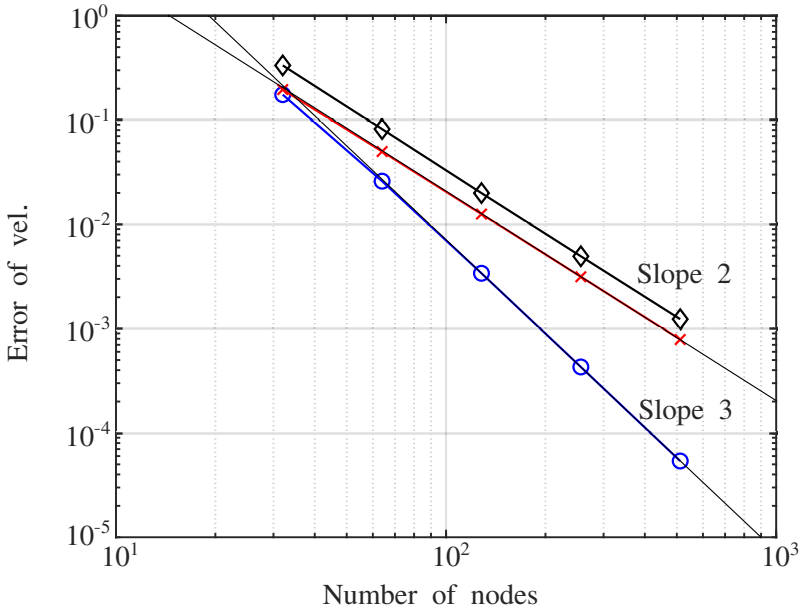

Figure 6.13: Error of velocity versus number of nodes, periodic rigid-body problem. CG with re-interpolation: ($\times$); DG: ($\diamond$); DG with Radau correction: ($\circ$).

fields; Radau corrections do not affect the predictions, as discussed in section 6.7; angular velocity and velocity fields exhibit second- and third-order accuracy without and with Radau corrections, respectively.

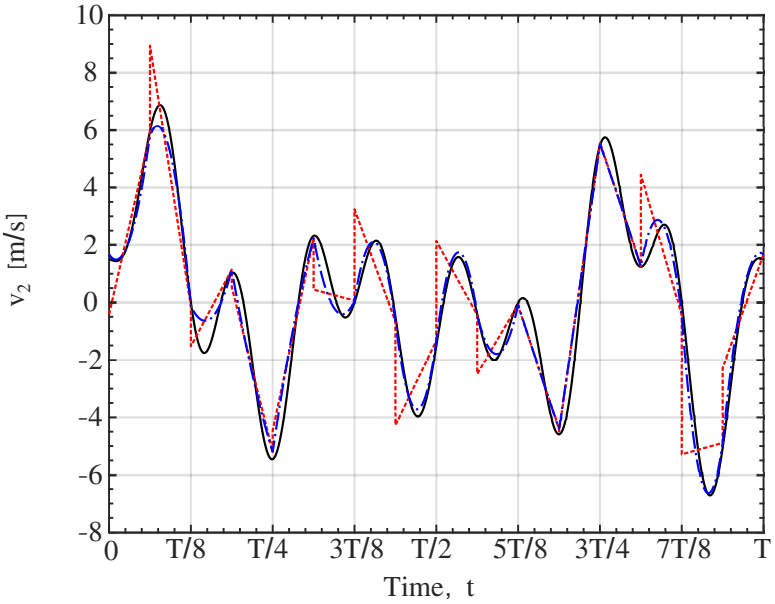

Figure 6.14: The second components of velocity vector $R^T \underline{\dot{r}}$. Reference: solid line; DG: dotted line; DG with Radau correction: dashed-dotted line.

Fig. 6.14 depicts the second velocity component predicted by DG method by using 16 2-node elements. Indeed, the Radau correction (6.34) improves the prediction of velocities dramatically.

6.8.3 Periodic problem of a spatial four-bar mechanism

This example deals with the flexible, spatial four-bar mechanism depicted in fig. 6.15. Bar 1 is connected to the ground via a revolute joint at point **A** and to bar 2 by means of a spherical joint at point **B**. In turns, bar 2 is connected to bar 3 via a universal joint at point **C** and finally, bar 3 is connected to the ground via a revolute joint at point **D**. At point **D**, the rotation of bar 3 is prescribed as

$\theta = \omega t + [1 - \cos(2\omega t)]/3 + \sin(5\omega t)]/5 - [1 - \cos(3\omega t)]/5$ rad, where $\omega = 0.3$ rad/s. The problem is periodic with a period $T = 2\pi/\omega = 20.944$ s. The geometric and material properties of the system are presented in ref. [23] and will not be repeated here.

Bar 1 and 3 is meshed using 1 four-node beam elements where bar 2 is meshed to 2 four-node elements. The periodic solution of the problem is obtained using three approaches: the Fourier collocation method [23] and the proposed CG and DG methods. The Fourier collocation method uses the solution for 99 grid points as its reference solution, while the CG and DG methods use the solution of 128 elements as their reference solutions.

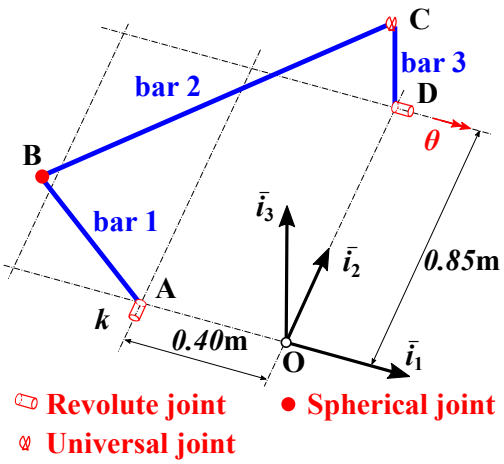

Figure 6.15: Configuration of a spatial four-bar mechanism.

Figs. 6.16 and 6.17 show the convergence rates for rotation and displacement of the material point **B**. Both CG and DG methods exhibit second-order convergence, as expected. The Fourier collocation method is more accurate than the CG and DG methods as the number of grid points or nodes becomes greater than 20. The Fourier collocation method is computationally expensive because the bandwidth of the Jacobian matrix is proportional to the number of time grids while the bandwidth remains constant in CG and DG methods as the number of elements increases.

The convergence plots for angular velocity and velocity of the material point **P** are shown in figs. 6.18 and 6.19, respectively. A re-interpolation is used in CG

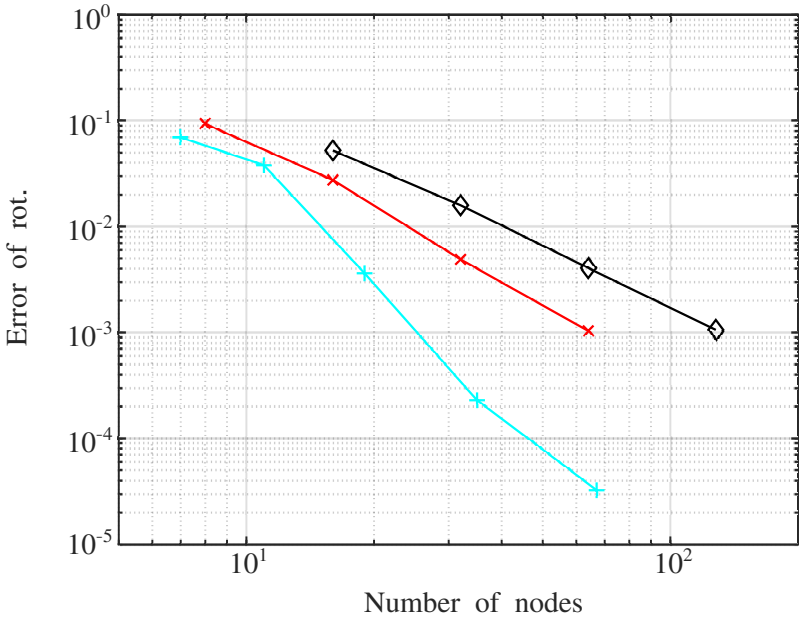

Figure 6.16: Error of rotation versus number of time grid points, spatial four-bar problem. DG: ($\diamond$); CG: ($\times$); Fourier collocation: ($+$).

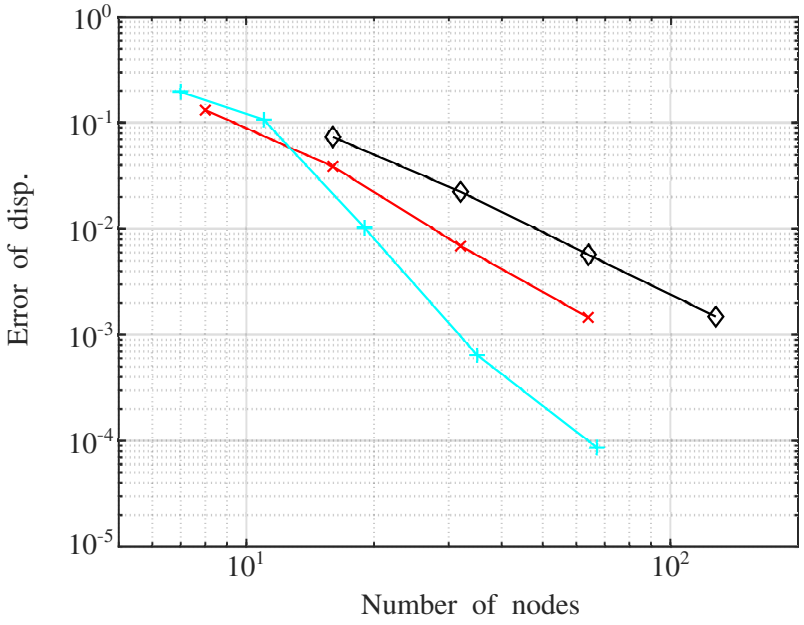

Figure 6.17: Error of displacements versus number of time grid points, spatial four-bar problem. DG: ($\diamond$); CG: ($\times$); Fourier collocation: ($+$).

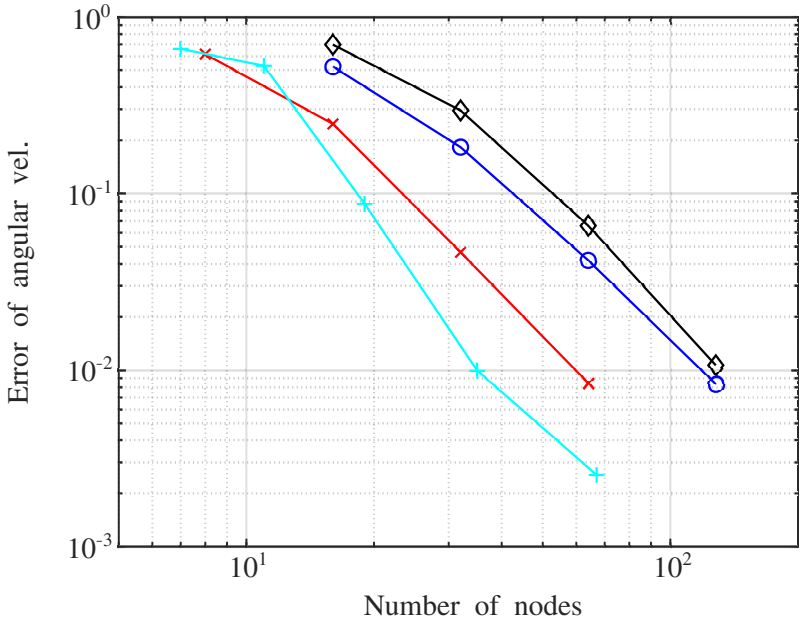

Figure 6.18: Error of angular velocity versus number of time grid points, spatial four-bar problem. DG with re-interpolation: ($\diamond$); DG with Radau correction and re-interpolation: ($\circ$); CG with re-interpolation: ($\times$); Fourier collocation: ($+$).

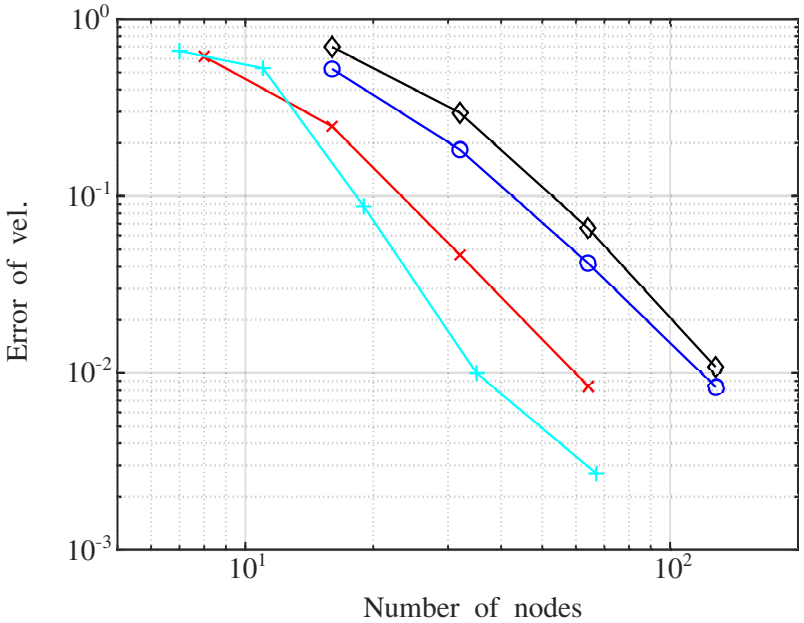

Figure 6.19: Error of velocity versus number of time grid points, spatial four-bar problem. DG with re-interpolation: ($\diamond$); DG with Radau correction and re-interpolation: ($\circ$); CG with re-interpolation: ($\times$); Fourier collocation: ($+$).

method. A combination of corrections is considered in DG method, i.e., the angular velocity and velocity are first corrected by eq. (6.34) then re-interpolated, see section 6.4. The corrected angular velocity and velocity fields exhibit second-order accuracy and Radau corrections improve the accuracy marginally. In summary, because the current problem is stiff, the DG method behaves like the stiff mass-spring system and is less accurate than the CG method.

6.8.4 Parametric excitation of a simply supported beam

Consider a simply supported beam of length $L = 2.0$ m, subjected to an end compressive load of harmonically varying amplitude, $P\cos(\Omega t)$, as depicted in fig. 6.20. The physical properties of the beam are: axial stiffness, $EA = 1.275\ 10^8$ N; shear stiffness about $\bar{\imath}_2$ and $\bar{\imath}_3$, $GA = 4.17\ 10^7$ N; torsional stiffness about $\bar{\imath}_1$, $GJ = 4.80\ 10^3$ N·m^2; bending stiffness about $\bar{\imath}_2$ and $\bar{\imath}_3$, $EI = 6.25\ 10^3$ N·m^2; mass per unit span, $m = 4.80$ kg/m; moment of inertial about $\bar{\imath}_2$ and $\bar{\imath}_3$, $J = 2.35\ 10^{-4}$ kg·m^2.

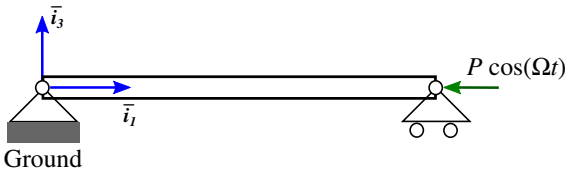

Figure 6.20: The parametric excitation of a simply supported beam.

For this problem, the equations of motion can be transformed to the multi-dimensional Mathieu equation in section (6.1.4) via the modal coordinate transfor-

mation method. The following notations are introduced for convenience: the fundamental natural frequency $\omega = \pi^2/L^2\sqrt{EI/m}$, static bucking load $P_{cr} = \pi^2 EI/L^2$, and $\mu = 1/2\, P/P_{cr}$. The load magnitude is chosen to $P = 0.15P_{cr}$, i.e., $\mu = 0.3$.

In the present analysis, the beam is modeled by using twelve 2-noded elements in the spatial dimension. To find the periodic solution and perform stability analysis, twelve 2-noded elements in the time domain are used. Fig. 6.21 shows the predicted Floquet multipliers. The Floquet multipliers predicted by the CG method are very large in the entire frequency range; this method cannot be used for the present problem. The predictions of DG method are in good agreement with the analytical solution. The DG method introduces numerical damping for high frequency modes, i.e., $\Omega/(2\omega) \leq 0.7$, as shown in fig. 6.21.

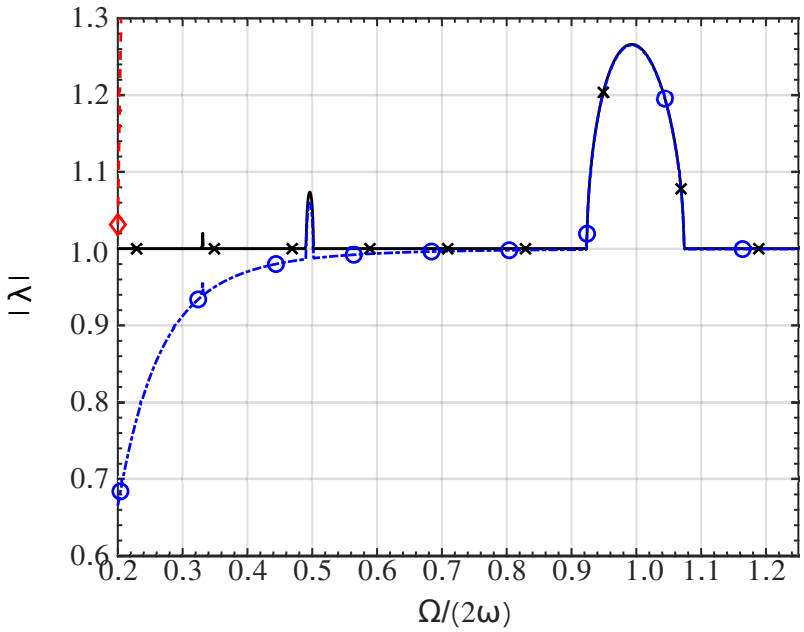

Figure 6.21: Floquet multipliers versus excitation frequency, for $\mu = 0.15$. Analytic: ($\times$); CG: ($\diamond$); DG: ($\circ$).

6.8.5 Ground resonance of a rotor model

A rigid body of mass $m_0 = 20$ kg is connected to the ground by a spring of stiffness $k_0 = 12$ kN/m. Four blades and are connected to the rigid body by revolute joints and root retentions. Each blade and retention are of length $L = 4.25$ m and $e = 0.25$ m, respectively. The retentions are driven to rotate about the rigid body at a constant angular velocity Ω. The lead-lag joints are connected by a torsional spring of stiffness $k_\theta = 2760$ N·m/rad. Both the blades and retentions are modeled as uniform beams with the same sectional properties: mass per unit span $m = 3$ kg/m. Three cases for the in-plane bending stiffness are considered: *(1)* rigid $EI_3 = \infty$ *(2)* stiff $EI_3 = 40$, and *(3)* soft $EI = 7.5$ kN·m^2.

Each root retention and beam is modeled with one and two cubic spatial beam elements in the spatial domain, respectively. To determine the periodic solution and analyze its stability, twelve 2-node elements in the time domain are used. The system exhibits central symmetry and its stability can also be analyzed via the Coleman transformation [162]. Under the Coleman transformation, the periodic

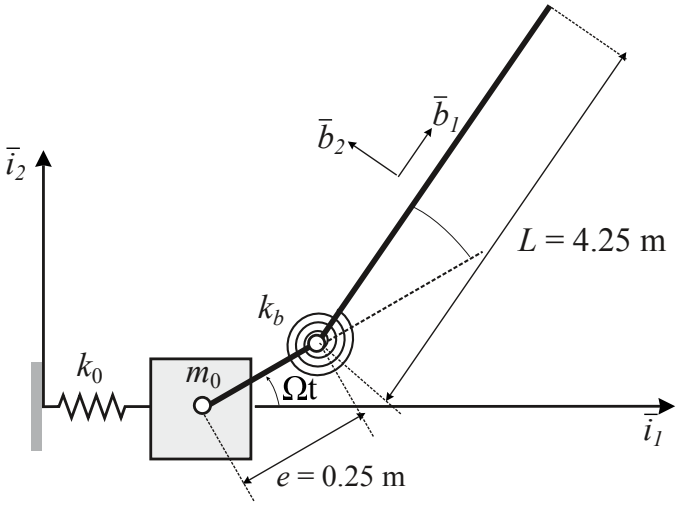

Figure 6.22: Schematic of the ground resonance problem. For clarity, a single blade of the system is shown.

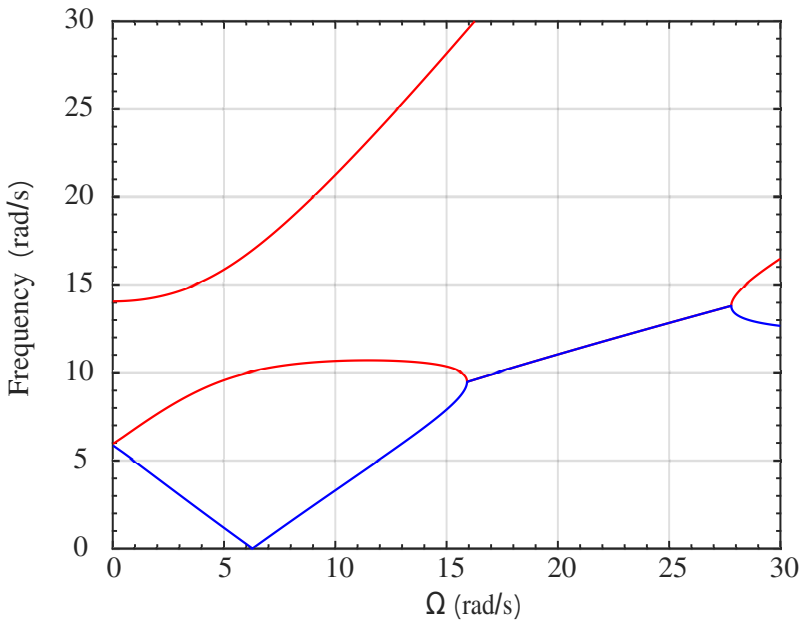

Figure 6.23: Coalescence of the frequencies.

system can be transformed into a time invariant system. For the rigid-blade case, the three frequencies of the transformed system is shown in fig. 6.23. As the driving frequency in the range of $[15, 27]$ rad/s, two frequencies coalesce and the system becomes unstable.

The instabilities predicted by the proposed Floquet's method are shown in fig. 6.24. The CG method works only for the rigid-blade case because it does not have numerical damping, while the DG method works for all the tree cases. The predictions of Coleman transformation and the proposed method agree well. The instability zone in flexible blade case is larger than that in the rigid-blade case and the instability zone becomes even larger as the blade becomes softer.

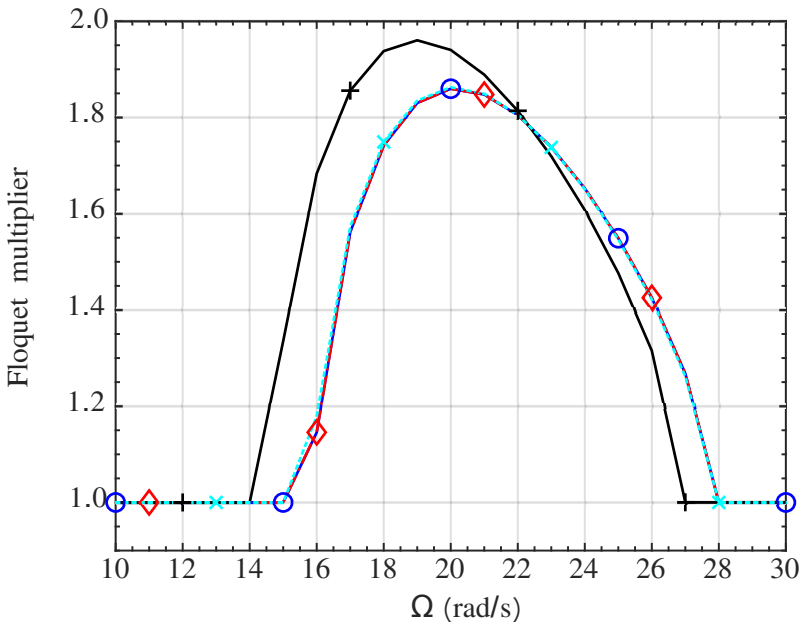

Figure 6.24: Floquet multiplier of the ground resonance problem. Coleman transformation: (○); CG, rigid: (◇); DG, stiff: (×); DG, soft: (+).

6.8.6 Stability of a wind turbine model

This example deals with the ground resonance of a four-bladed wind turbine model, as depicted in fig. 6.25. The tower is of height $H = 6.0$ m and represented by a beam. The nacelle is attached at the tip of the tower, projects $S = 0.8$ m forward, and is represented by a beam rigid-connected with a rigid body at its left tip. The

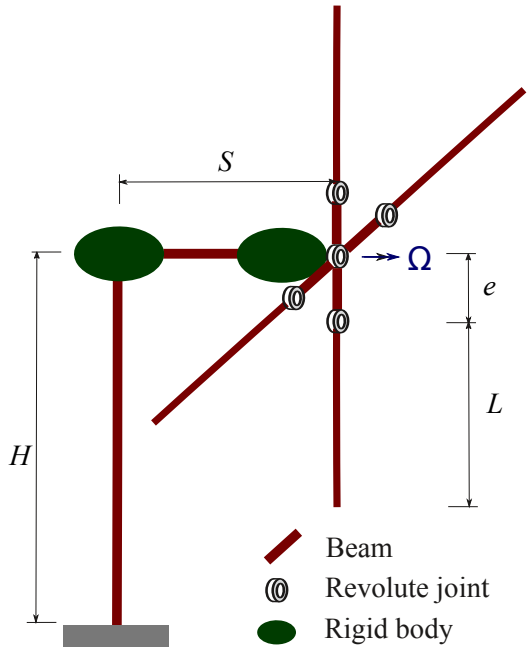

Figure 6.25: Schematic of a wind turbine model.

rotor hub is located at the right tip of the nacelle and is represented by a rigid body. Each blade is of length $L = 4.25$ and connected to the hub by a revolute joint and a root retention of length $e = 0.25$ m. The rotor are driven to rotating about the hub with a constant angular velocity Ω. The mass and stiffness properties are listed in the following tables.

Body	m (kg or kg/m)	$I_{22} = I_{33} = 1/2I_{11}$ (kg·m^2 or kg·m)
Nacelle-rigid-body	50	5
Hub	30	4
Tower	12.72	0.01
nacelle		
blade	3	0.278 10^{-3}
retention		

Table 6.1: Mass properties of the wind turbine model.

Body	Extension (MPa)	Shear (MPa)	Bending (kN·m^2)	Torsion (kN·m^2)
Tower	811	258	387	297
nacelle				
blade	81.1	25.8	7.51	4.87
retention				

Table 6.2: Stiffness properties of the wind turbine model.

The tower and each blade are meshed to six 2-noded beam elements and each root retention are meshed to one 2-noded beam element. To find the periodic solution and perform stability analysis, twenty 2-noded element in the time domain are used. The predicted Floquet multiplier is shown in fig. 6.26.

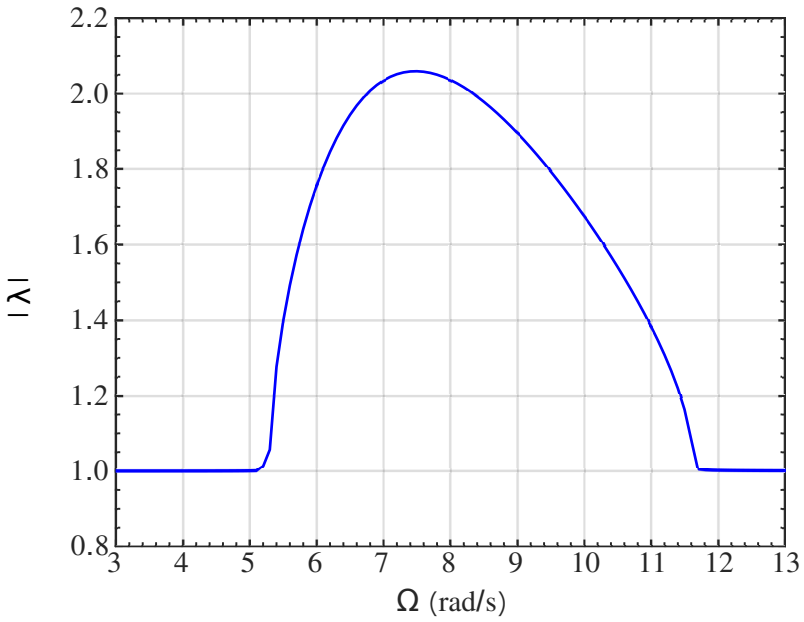

Figure 6.26: Floquet multiplier predicted by DG Hamiltonian method.

6.9 Summary and conclusions

Floquet's and Hill's methods are investigated for the stability analysis of periodic solutions. The advantages and limitations of these two methods are illustrated by the analysis of a multi-dimensional Mathieu equation. Hill's method introduces fictitious instabilities resulting from aliasing. Floquet's method provides reliable predictions when unconditionally stable integration schemes are used but fails when using conditionally stable schemes.

The continuous and discontinuous Galerkin formulations are applied to the determination of periodic solutions and their stability analysis. In both formulations, the monodromy matrix is constructed by manipulating the Jacobian matrix resulting from a linearization about the periodic solution. Numerical examples of increasing complexity are presented to validate the proposed methods. The continuous

Galerkin method is second-order accurate for both stiff or non-stiff problems while the discontinuous Galerkin method is third- and second-order accurate for non-stiff and stiff problems, respectively. The continuous Galerkin method works for stability analysis of rigid systems only, because it is conditionally stable. The discontinuous Galerkin method works well for the stability analysis of flexible systems.

Chapter 7: Galerkin Methods in Time Domain: Optimal Control and Optimization Problems

Continuous and discontinuous Galerkin methods are applied to optimal control and optimization problems.

7.1 The problem of optimal control and optimization

Consider a flexible multibody system composed of n structural nodes and the kinematics of each nodes is represented by motion tensors $\mathcal{R}^i$, $i = 1, 2, \ldots, n$. The nodes subjected to m holonomic constraints, i.e., $\underline{g}(\mathcal{R}^i, t) = \underline{0}$. The optimal control problem is formulated as

$$\min \quad \phi[\mathcal{R}^i(T), \hat{\underline{w}}(T)] + \int_0^T L(\mathcal{R}^i, \hat{\underline{w}}, \underline{\theta}, t)\mathrm{d}t, \tag{7.1a}$$

subject to

$$\hat{\underline{w}} - \hat{\underline{v}} = \underline{0}, \tag{7.1b}$$

$$M\,\dot{\hat{\underline{w}}} - \mathrm{diag}(\tilde{w}^i)^T M\,\hat{\underline{w}} + \underline{f}^{\mathrm{int}} + G_{\mathrm{p}}^T\underline{\lambda} + G_{\mathrm{a}}^T\underline{\theta} = \underline{f}^{\mathrm{ext}}, \tag{7.1c}$$

$$\underline{g}(\mathcal{R}^i, t) = \underline{0}, \tag{7.1d}$$

$$\underline{\psi}[\mathcal{R}^i(T), \hat{\underline{w}}(T)] = \underline{0}, \tag{7.1e}$$

where the cost functional consists of a terminal term denoted by ϕ and of an integral term; the first three constraint equations (7.1b), (7.1c), and (7.1d) are the kinematic, equilibrium, and constraint equations of the multibody system; $\underline{\theta} \in \mathbb{R}^l$ denotes the control torques; $G_\mathrm{p} \in \mathbb{R}^{m\times 6n} = \nabla_{\mathcal{R}}\underline{g}$ and $G_\mathrm{a} \in \mathbb{R}^{l\times 6n}$ denote the constraint Jacobian of the passive and actuated joints, respectively; $\underline{\psi} \in \mathbb{R}^r$ denotes terminal constraints on the state variables.

Similarly, the optimization problem is formulated as

$$\min \quad \phi[\mathcal{R}^i(T), \underline{\hat{w}}(T)] + \int_0^T L(\mathcal{R}^i, \underline{\hat{w}}, \underline{\theta}, t)\mathrm{d}t, \tag{7.2a}$$

subject to

$$\underline{\hat{w}} - \underline{\hat{v}} = \underline{0}, \tag{7.2b}$$

$$M(\underline{\theta})\,\dot{\underline{\hat{w}}} - \mathrm{diag}(\tilde{\hat{w}}^i)^T M(\underline{\theta})\,\underline{\hat{w}} + \underline{f}^{\mathrm{int}}(\underline{\theta}) + G_\mathrm{p}^T(\underline{\theta})\underline{\lambda} = \underline{f}^{\mathrm{ext}}, \tag{7.2c}$$

$$\underline{g}(\mathcal{R}^i, \underline{\theta}, t) = \underline{0}, \tag{7.2d}$$

$$\underline{\psi}[\mathcal{R}^i(T), \underline{\hat{w}}(T)] = \underline{0}, \tag{7.2e}$$

where $\underline{\theta} \in \mathbb{R}^l$ denotes the control design parameters; mass matrix M, internal force vector $\underline{f}^{\mathrm{int}}$, and kinematic constraints $\underline{g}$ may depend explicitly on the design parameters.

7.2 CG and DG methods for discretization

Both of continuous and discontinuous Galerkin methods are applied to the discretization of optimal control problem (7.1) and optimization problem (7.2).

Suppose time span $[0, T]$ is divided to K two-node elements, denoted $[t_{I-}^{+}, t_{I}^{-}]$, $I = 1, 2, \ldots, K$. Clearly, $t_0^+ = 0$, and $t_K^- = T$. The choice of test and trial functions has been discussed in section 5.3. The discretized kinematic, equilibrium, and constraint equations, (7.1b), (7.1c), and (7.1d) in optimal control problems, are rewritten here for convenience.

- DG method

$$\underline{\boldsymbol{w}}_I - \underline{\boldsymbol{v}}_I = \underline{\imath} \otimes [\![\underline{\hat{q}}]\!]_{I-1}, \tag{7.3a}$$

$$\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{\text{p}I1}^T \underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}_{\text{a}I}^T \underline{\boldsymbol{\theta}}_I} = \underline{\boldsymbol{f}}_I^{\text{ext}} + \underline{1}_0 \otimes (M [\![\underline{\hat{w}}]\!]_{I-1}), \tag{7.3b}$$

$$\underline{\boldsymbol{g}}_I = \underline{0}, \tag{7.3c}$$

  where the term $\mathbb{G}_{aI}^T \underline{\boldsymbol{\theta}}_I$ resulting from the discretization of actuation forces.

- CG method

$$\underline{\boldsymbol{w}}_I - \underline{\boldsymbol{v}}_I = \underline{0}, \tag{7.4a}$$

$$-\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{\text{p}I1}^T \underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}_{\text{a}I}^T \underline{\boldsymbol{\theta}}_I} = \underline{\boldsymbol{f}}_I^{\text{ext}} + \underline{1_1 \otimes (M\, \underline{\hat{w}}(t_K))} - \underline{\underline{1_0 \otimes (M\, \underline{\hat{w}}(t_0))}}, \tag{7.4b}$$

$$\underline{\boldsymbol{g}}_I = \underline{0}, \tag{7.4c}$$

  where the underlined and double-underlined terms are active in the last and first elements, respectively; the boxed term is active in optimal control problems.

In both of the CG and DG methods, independent velocities $\underline{\boldsymbol{w}}_I$ can be eliminated from eqs. (7.3a) or (7.4), respectively. Consequently, these two equations do not enter into the final governing equations explicitly.

When the DG method is applied, the discretized optimal control problem (7.1) becomes

$$\min \quad \phi(\mathcal{R}^i_{I,N}, \underline{\hat{w}}_{I,N}) + \sum_{I=1}^{K} L_I, \tag{7.5a}$$

subject to

$$\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}^{\text{gyro}}_I + \underline{\boldsymbol{f}}^{\text{int}}_I + \mathbb{G}^T_{\text{p}I1}\underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}^T_{\text{a}I}\underline{\boldsymbol{\theta}}_I} = \underline{\boldsymbol{f}}^{\text{ext}}_I + \underline{1}_0 \otimes (M[\![\underline{\hat{w}}]\!]_{I-1}), \tag{7.5b}$$

$$\underline{\boldsymbol{g}}_I = \underline{0}, \tag{7.5c}$$

$$\underline{\psi}(\mathcal{R}^i_{K,N}, \underline{\hat{w}}_{K,N}) = \underline{0}, \tag{7.5d}$$

where $L_I = \int_{t^+_{I-1}}^{t^+_I} L \, \mathrm{d}t$. When CG method is applied, the discretized optimal control

problem (7.1) becomes

$$\min \quad \phi(\mathcal{R}^i_{I,N}, \underline{\hat{w}}_{I,N}) + \sum_{I=1}^{K} L_I, \tag{7.6a}$$

subject to

$$-\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}^{\text{gyro}}_I + \underline{\boldsymbol{f}}^{\text{int}}_I + \mathbb{G}^T_{\text{p}I1}\underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}^T_{\text{a}I}\underline{\boldsymbol{\theta}}_I} = \underline{\boldsymbol{f}}^{\text{ext}}_I + \underline{1_1 \otimes (M\underline{\hat{w}}(t_K))} - \underline{1_0 \otimes (M\underline{\hat{w}}(t_0))}, \tag{7.6b}$$

$$\underline{\boldsymbol{g}}_I = \underline{0}, \tag{7.6c}$$

$$\underline{\psi}(\mathcal{R}^i_{K,N}, \underline{\hat{w}}_{K,N}) = \underline{0}. \tag{7.6d}$$

7.3 First order optimality condition, DG method

If the DG method is applied for the discritization, the following element-wise augmented functional are introduced

$$\begin{aligned}\mathcal{L}_I = L_I + \underline{\boldsymbol{\nu}}^T_I [\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}^{\text{gyro}}_I + \underline{\boldsymbol{f}}^{\text{int}}_I + \mathbb{G}^T_{\text{p}I1}\underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}^T_{\text{a}I}\underline{\boldsymbol{\theta}}_I} - \underline{\boldsymbol{f}}^{\text{ext}}_I \\ - \underline{1_0 \otimes (M[\![\underline{\hat{w}}]\!]_{I-1})}] + \underline{\boldsymbol{\xi}}^T_I \underline{\boldsymbol{g}}_I + \underline{\phi + \underline{\pi}^T \underline{\psi}},\end{aligned} \tag{7.7}$$

where the underlined term is active only for the last element, i.e.,$I = K$. Therein, $\underline{\boldsymbol{\nu}}_I \in \mathbb{R}^{6n(N+1)}$, $\underline{\boldsymbol{\xi}}_I \in \mathbb{R}^{m(N+1)}$, and $\underline{\pi} \in \mathbb{R}^r$ are Lagrange multipliers. For convenience,

the following notation was introduced

$$\delta\underline{\boldsymbol{p}}_I \overset{\text{def}}{=} \begin{Bmatrix} \delta\underline{\boldsymbol{u}}_{I-2} \\ \delta\underline{\boldsymbol{\lambda}}_{I-2} \\ \delta\underline{\boldsymbol{u}}_{I-1} \\ \delta\underline{\boldsymbol{\lambda}}_{I-1} \\ \delta\underline{\boldsymbol{u}}_I \\ \delta\underline{\boldsymbol{\lambda}}_I \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{p}}_I \overset{\text{def}}{=} \begin{Bmatrix} \Delta\underline{\boldsymbol{u}}_{I-2} \\ \Delta\underline{\boldsymbol{\lambda}}_{I-2} \\ \Delta\underline{\boldsymbol{u}}_{I-1} \\ \Delta\underline{\boldsymbol{\lambda}}_{I-1} \\ \Delta\underline{\boldsymbol{u}}_I \\ \Delta\underline{\boldsymbol{\lambda}}_I \end{Bmatrix}, \quad \delta(\text{or } \Delta)\underline{\boldsymbol{d}}_I \overset{\text{def}}{=} \delta(\text{or } \Delta) \begin{Bmatrix} \underline{\boldsymbol{\nu}}_{I-2} \\ \underline{\boldsymbol{\xi}}_{I-2} \\ \underline{\boldsymbol{\nu}}_{I-1} \\ \underline{\boldsymbol{\xi}}_{I-1} \\ \underline{\boldsymbol{\nu}}_I \\ \underline{\boldsymbol{\xi}}_I \end{Bmatrix},$$

$$\delta\underline{\boldsymbol{p}}_T \overset{\text{def}}{=} \begin{Bmatrix} \delta\underline{\boldsymbol{u}}_T \\ \delta\underline{\boldsymbol{\lambda}}_T \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{p}}_T \overset{\text{def}}{=} \begin{Bmatrix} \Delta\underline{\boldsymbol{u}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T \end{Bmatrix}, \quad \delta(\text{or } \Delta)\underline{\boldsymbol{d}}_I \overset{\text{def}}{=} \delta(\text{or } \Delta) \begin{Bmatrix} \underline{\boldsymbol{\nu}}_T \\ \underline{\boldsymbol{\xi}}_T \end{Bmatrix},$$

where subscript T indicates a stack of quantities in all time grids in an ascent order.

Vanishing of the first order variation of functional $\sum_{I=1}^{K} \delta\mathcal{L}_I = 0$ leads to

$$\sum_{I=1}^{K} \left(\begin{Bmatrix} \delta\underline{\boldsymbol{\theta}}_I \\ \delta\underline{\boldsymbol{p}}_I \\ \delta\underline{\boldsymbol{d}}_I \\ \delta\underline{\pi} \end{Bmatrix}^T \begin{Bmatrix} \nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I \\ \nabla_{\underline{\pi}}\mathcal{L}_I \end{Bmatrix} \right) = \begin{Bmatrix} \delta\underline{\boldsymbol{\theta}}_T \\ \delta\underline{\boldsymbol{p}}_T \\ \delta\underline{\boldsymbol{d}}_T \\ \delta\underline{\pi} \end{Bmatrix}^T \left(\mathbb{A}_{I=1}^{K} \begin{Bmatrix} \nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I \\ \nabla_{\underline{\pi}}\mathcal{L}_I \end{Bmatrix} \right) = 0, \tag{7.8}$$

notation $\mathbb{A}$ indicates an assembly over elements. Because all variables are free,

eq. (7.8) yields to the first-order optimality condition

$$\mathbb{A}_{I=1}^{K}\begin{Bmatrix}\nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I\\ \nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I\\ \nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I\\ \nabla_{\underline{\pi}}\mathcal{L}_I\end{Bmatrix}=\underline{0}_{2K[l+2(6n+m)+r]}, \tag{7.9}$$

where

$$\nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I=\nabla_{\underline{\boldsymbol{\theta}}_I}L_I+\mathbb{G}_{\mathrm{a}I}\underline{\boldsymbol{\nu}}_I+\underline{1}_1(\nabla_{\underline{\theta}}\phi)+\boxed{\boxed{\mathbb{F}_I^T\underline{\boldsymbol{\nu}}_I}},$$

$$\nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I=\begin{bmatrix}\mathbb{K}_{I2}^T\\ \mathbb{K}_{I1}^T\\ \mathbb{K}_{I0}^T\end{bmatrix}\begin{Bmatrix}\underline{\boldsymbol{\nu}}_I\\ \underline{\boldsymbol{\xi}}_I/s\end{Bmatrix}+\underline{\begin{Bmatrix}\underline{0}_{2(6n+m)}\\ \underline{1}_1\otimes[\nabla_{\mathcal{R}}\phi+(\nabla_{\mathcal{R}}\psi)^T\underline{\pi}]\\ \underline{0}_{2m}\end{Bmatrix}},$$

$$\nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I=\begin{Bmatrix}\underline{0}_{2\times(6n+m)}\\ \mathbb{M}\,\underline{\boldsymbol{w}}_I+\underline{\boldsymbol{f}}_I^{\mathrm{gyro}}+\underline{\boldsymbol{f}}_I^{\mathrm{int}}+\mathbb{G}_{\mathrm{p}I1}^T\underline{\boldsymbol{\lambda}}_I+\boxed{\mathbb{G}_{\mathrm{a}I}^T\underline{\boldsymbol{\theta}}_I}-\underline{1}_0\otimes(M[\![\underline{\hat{w}}]\!]_{I-1})\\ s\underline{\boldsymbol{g}}_I\end{Bmatrix},$$

$$\nabla_{\underline{\pi}}\mathcal{L}_I=\underline{\psi}.$$

Therein, double-boxed terms are active in optimization problems only and the fol-

lowing notation is defined

$$\mathbb{F}_I = \nabla_{\underline{\boldsymbol{\theta}}_I} \left[\mathbb{M}\, \underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{\text{p}I1}^T \underline{\boldsymbol{\lambda}}_I - \underline{1}_0 \otimes (M[\![\underline{\hat{w}}]\!]_{I-1}) \right],$$

$$\mathbb{K}_{I2} = \begin{bmatrix} -(\underline{1}_0 \underline{1}_1^T) \otimes M\mathbb{L}_{(I-1)0} \\ \\ \end{bmatrix}$$

$$\mathbb{K}_{I1} = \begin{bmatrix} [(\underline{1}_0 \underline{1}_0^T) \otimes M + \mathbb{M} + \mathbb{C}_I]\mathbb{L}_{I0} - (\underline{1}_0 \underline{1}_1^T) \otimes M\mathbb{L}_{(I-1)1} \\ \\ \end{bmatrix},$$

$$\mathbb{K}_{I0} = \begin{bmatrix} [(\underline{1}_1 \underline{1}_1^T) \otimes M + \mathbb{M}_I + \mathbb{C}_I]\mathbb{L}_{I1} + \mathbb{K}_I + \mathbb{N}_I & s\mathbb{G}_{I1}^T \\ s\mathbb{G}_{I2} & \end{bmatrix}.$$

7.4 First-order optimality condition, CG method

If the CG method is applied for the discretization, the following element-wise augmented functional is introduced

$$\begin{aligned} \mathcal{L}_I = L_I + \underline{\boldsymbol{\nu}}_I^T [& -\mathbb{M}\, \underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\text{gyro}} + \underline{\boldsymbol{f}}_I^{\text{int}} + \mathbb{G}_{\text{p}I1}^T \underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}_{\text{a}I}^T \underline{\boldsymbol{\theta}}_I} - \underline{\boldsymbol{f}}_I^{\text{ext}} \\ & + \underline{1}_1 \otimes (M\, \underline{\hat{\boldsymbol{w}}}(t_K)) - \underline{\underline{1}_0 \otimes (M \underline{\hat{\boldsymbol{w}}}(t_0))}] + \underline{\boldsymbol{\xi}}_I^T \underline{\boldsymbol{g}}_I + \underline{\phi + \underline{\pi}^T \underline{\psi}}, \end{aligned} \tag{7.10}$$

where the underlined term is active only for the last element, i.e., $I = K$. Therein, $\underline{\boldsymbol{\nu}}_I \in \mathbb{R}^{6n(N+1)}$, $\underline{\boldsymbol{\xi}}_I \in \mathbb{R}^{m(N+1)}$, and $\underline{\pi} \in \mathbb{R}^r$ are Lagrange multipliers. For convenience,

the following notations are introduced

$$\delta\underline{\boldsymbol{p}}_I \overset{\text{def}}{=} \begin{Bmatrix} \delta\underline{\boldsymbol{u}}_I \\ \delta\underline{\boldsymbol{\lambda}}_I \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{p}}_I \overset{\text{def}}{=} \begin{Bmatrix} \Delta\underline{\boldsymbol{u}}_I \\ \Delta\underline{\boldsymbol{\lambda}}_I \end{Bmatrix}, \quad \delta(\text{or } \Delta)\underline{\boldsymbol{d}}_I \overset{\text{def}}{=} \delta(\text{or } \Delta) \begin{Bmatrix} \underline{\boldsymbol{\nu}}_I \\ \underline{\boldsymbol{\xi}}_I \end{Bmatrix},$$
$$\delta\underline{\boldsymbol{p}}_T \overset{\text{def}}{=} \begin{Bmatrix} \bar{\delta\underline{\boldsymbol{u}}}_T \\ \delta\underline{\boldsymbol{\lambda}}_T \end{Bmatrix}, \quad \Delta\underline{\boldsymbol{p}}_T \overset{\text{def}}{=} \begin{Bmatrix} \bar{\Delta\underline{\boldsymbol{u}}}_T \\ \Delta\underline{\boldsymbol{\lambda}}_T \end{Bmatrix}, \quad \delta(\text{or } \Delta)\underline{\boldsymbol{d}}_I \overset{\text{def}}{=} \delta(\text{or } \Delta) \begin{Bmatrix} \underline{\boldsymbol{\nu}}_T \\ \underline{\boldsymbol{\xi}}_T \end{Bmatrix},$$

where subscript T indicates a stack of quantities in all time grids in ascent order. In the CG method, the motion tensors and velocities at the initial and terminal time t_0 and t_K are given. Accordingly, the increments of motion tensors are rearranged as

$$\bar{\Delta\underline{\boldsymbol{u}}}_T = \begin{Bmatrix} \Delta\hat{\underline{v}}(t_K) \\ \hat{\Delta\underline{u}}(t_1) \\ \vdots \\ \hat{\Delta\underline{u}}(t_{K-1}) \end{Bmatrix}.$$

Vanishing of the first-order variation of functional $\sum_{I=1}^{K} \delta\mathcal{L}_I = 0$ leads to the first-order optimality condition

$$\mathbb{A}_{I=1}^{K} \begin{Bmatrix} \nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I \\ \nabla_{\underline{\pi}}\mathcal{L}_I \end{Bmatrix} = \underline{0}_{K[l+2(6n+m)+r]-12n}, \tag{7.11}$$

where

$$\nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I = \nabla_{\underline{\boldsymbol{\theta}}_I}L_I + \mathbb{G}_{\mathrm{a}I}\underline{\boldsymbol{\nu}}_I + \underline{1}_1 \otimes (\nabla_{\underline{\theta}}\phi) + \boxed{\boxed{\mathbb{F}_I^T\underline{\boldsymbol{\nu}}_I}},$$

$$\nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I = \mathbb{K}_I^T \begin{Bmatrix} \underline{\boldsymbol{\nu}}_I \\ \underline{\boldsymbol{\xi}}_I/s \end{Bmatrix} + \underline{\begin{Bmatrix} \underline{0}_{2(6n+m)} \\ \underline{1}_1 \otimes [\nabla_{\mathcal{R}}\phi + (\nabla_{\mathcal{R}}\psi)^T\underline{\pi}] \\ \underline{0}_m \end{Bmatrix}},$$

$$\nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I = \begin{Bmatrix} \underline{0}_{2(6n+m)} \\ -\mathbb{M}\,\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\mathrm{gyro}} + \underline{\boldsymbol{f}}_I^{\mathrm{int}} + \mathbb{G}_{\mathrm{p}I1}^T\underline{\boldsymbol{\lambda}}_I + \boxed{\mathbb{G}_{\mathrm{a}I}^T\underline{\boldsymbol{\theta}}_I} + \underline{1}_1 \otimes (M\,\underline{\hat{\boldsymbol{w}}}(t_K)) - \underline{\underline{1}_0 \otimes (M\underline{\hat{\boldsymbol{w}}}(t_0))} \\ s\underline{\boldsymbol{g}}_I \end{Bmatrix},$$

$$\nabla_{\underline{\pi}}\mathcal{L}_I = \underline{\psi}.$$

Therein

$$\mathbb{F}_I = \nabla_{\underline{\boldsymbol{\theta}}_I}\left[-\mathbb{M}\underline{\boldsymbol{w}}_I + \underline{\boldsymbol{f}}_I^{\mathrm{gyro}} + \underline{\boldsymbol{f}}_I^{\mathrm{int}} + \mathbb{G}_{\mathrm{p}I1}^T\underline{\boldsymbol{\lambda}}_I - \underline{1}_1 \otimes (M\,\underline{\hat{\boldsymbol{w}}}(t_K)) + \underline{\underline{1}_0 \otimes (M\underline{\hat{\boldsymbol{w}}}(t_0))}\right],$$

$$\mathbb{K}_I = \begin{bmatrix} -\mathbb{M} + \mathbb{C}_I + \mathbb{K}_I + \mathbb{N}_I & s\mathbb{G}_{\mathrm{p}I1}^T \\ s\mathbb{G}_{\mathrm{p}I2} & \end{bmatrix}.$$

7.5 Newton method

The nonlinear equations resulting from the first-order optimality condition are solved using Newton method. The linearization of eqs. (7.9) and (7.9) have the same

form

$$\begin{aligned}
&\mathbb{A}_{I=1}^{K}\left(\begin{bmatrix}
\nabla^2_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I & \nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\boldsymbol{p}}_I}\mathcal{L}_I & \nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I & \\
(\nabla_{\underline{\boldsymbol{p}}_I,\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I)^T & \nabla^2_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I & \nabla_{\underline{\boldsymbol{p}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I & \nabla_{\underline{\boldsymbol{p}}_I,\underline{\pi}}\mathcal{L}_I \\
(\nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I)^T & (\nabla_{\underline{\boldsymbol{p}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I)^T & & \\
& (\nabla_{\underline{\boldsymbol{p}}_I,\underline{\pi}}\mathcal{L}_I)^T & &
\end{bmatrix}\right)\begin{Bmatrix}
\Delta\underline{\boldsymbol{\theta}}_T \\ \Delta\underline{\boldsymbol{p}}_T \\ \Delta\underline{\boldsymbol{d}}_T \\ \Delta\underline{\pi}
\end{Bmatrix} \\
&= -\mathbb{A}_{I=1}^{K}\begin{Bmatrix}
\nabla_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{p}}_I}\mathcal{L}_I \\ \nabla_{\underline{\boldsymbol{d}}_I}\mathcal{L}_I \\ \nabla_{\underline{\pi}}\mathcal{L}_I
\end{Bmatrix},
\end{aligned} \tag{7.12}$$

where the submatrices are defined in the next section.

The Hessian matrix in eq. (7.12) is composed of 4×4 major blocks, and each of them are sparse. Although the entire matrix can be factorized directly as a sparse matrix, a more convenient approach is to manipulate each blocks individually. For convenience, the linear system is rewritten as

$$\begin{bmatrix}
A & B^T & C^T & \\
B & J & K^T & G^T \\
C & K & & \\
& G & &
\end{bmatrix}\begin{Bmatrix}
\Delta\underline{\boldsymbol{\theta}} \\ \Delta\underline{\boldsymbol{p}} \\ \Delta\underline{\boldsymbol{d}} \\ \Delta\underline{\pi}
\end{Bmatrix} = \begin{Bmatrix}
\underline{r}_2 \\ \underline{r}_2 \\ \underline{r}_3 \\ \underline{r}_4
\end{Bmatrix}. \tag{7.13}$$

The first step is to obtain increments $\Delta\underline{\boldsymbol{p}}$ and $\Delta\underline{\boldsymbol{d}}$ from the second and third rows

$$\Delta\underline{\boldsymbol{p}} = K^{-1}(\underline{r}_3 - C\Delta\underline{\boldsymbol{\theta}}) = \underline{y} - T\Delta\underline{\boldsymbol{\theta}}, \tag{7.14a}$$

$$\Delta\underline{\boldsymbol{d}} = K^{-T}(\underline{r}_2 - J\Delta\underline{\boldsymbol{p}} - B\Delta\underline{\boldsymbol{\theta}} - G^T\Delta\underline{\pi}) = \underline{z} + (K^{-T}JT - L)\Delta\underline{\boldsymbol{\theta}} - S\Delta\underline{\pi}, \tag{7.14b}$$

where $T = K^{-1}C$, $L = K^{-T}B$, $S = K^{-T}G^T$, $\underline{y} = K^{-1}\underline{r}_3$, and $\underline{z} = K^{-T}(\underline{r} - J\underline{y})$. Matrix T represents the sensitivity of primal variables with respected to control inputs. Clearly, the system is controllable if and only if T has full-column rank. Introducing solutions (7.14) to the first and last rows of eqs. (7.13) leads to

$$\begin{bmatrix} A + T^TJT - B^TT - T^TB & -(GT)^T \\ -GT & \end{bmatrix} \begin{Bmatrix} \Delta\underline{\boldsymbol{\theta}} \\ \Delta\underline{\pi} \end{Bmatrix} = \begin{Bmatrix} \underline{z}_1 \\ \underline{z}_2 \end{Bmatrix}, \tag{7.15}$$

where $\underline{z}_1 = \underline{r}_1 - B^T\underline{y} - C^T\underline{z}$ and $\underline{z}_4 = \underline{r}_4 - G\underline{y}$.

In summary, solving the final linear systems (7.13) breaks down to the following steps

1. Factorize matrices K and K^T;

2. Compute auxiliary matrices $T = K^{-1}C$, $A + T^TJT$, GT, and vectors $K^{-1}\underline{r}_3$, $K^{-T}(\underline{r} - J - \underline{y})$, $K^{-T}\underline{r}_2$;

3. Factorize matrix

$$\begin{bmatrix} A + T^TJT - B^TT - T^TB & -(GT)^T \\ -GT & \end{bmatrix}$$

and solve reduced linear system (7.15).

7.6 Hessian matrices for CG and DG methods

In the DG method, the Hessian matrices in eq. (7.12) are defined as

$$\nabla^2_{\underline{\boldsymbol{\theta}}_I} \mathcal{L}_I = \nabla^2_{\underline{\boldsymbol{\theta}}_I} L_I + \underline{(\underline{1}_1 \underline{1}_1^T) \otimes (\nabla^2_{\underline{\theta}} \underline{\phi})} + \boxed{\nabla_{\underline{\boldsymbol{\theta}}_I} (\mathbb{F}_I^T \underline{\boldsymbol{\nu}}_I)}, \tag{7.16a}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I, \underline{\boldsymbol{p}}_I} \mathcal{L}_I = \begin{bmatrix} 0 & 0 & 0 & 0 & \boxed{\mathbb{H}_I} & 0 \end{bmatrix}, \tag{7.16b}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I, \underline{\boldsymbol{d}}_I} \mathcal{L}_I = \begin{bmatrix} 0 & 0 & 0 & 0 & \boxed{\mathbb{G}_{\mathrm{a}I}} + \boxed{\boxed{\mathbb{F}_I^T}} & 0 \end{bmatrix}, \tag{7.16c}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I, \underline{\pi}} \mathcal{L}_I = \underline{\underline{1}_1 \otimes (\nabla_{\underline{\theta}} \underline{\psi})^T}, \tag{7.16d}$$

$$\nabla_{\underline{\boldsymbol{p}}_I, \underline{\boldsymbol{d}}_I} \mathcal{L}_I = \begin{bmatrix} 0 & 0 & \mathbb{K}_{I2} \\ 0 & 0 & \mathbb{K}_{I1} \\ 0 & 0 & \mathbb{K}_{I0} \end{bmatrix}, \tag{7.16e}$$

$$\nabla_{\underline{\boldsymbol{p}}_I, \underline{\pi}} \mathcal{L}_I = \begin{bmatrix} 0 & 0 & \underline{(\underline{1}_1 \underline{1}_1^T) \otimes (\nabla_{\mathcal{R}} \underline{\psi})^T} \\ 0 & 0 & 0 \end{bmatrix}, \tag{7.16f}$$

where $\mathbb{H}_I = \int_{t_{I-1}^+}^{t_I^-} \ell^T \otimes H \mathbb{L}_{uI} \, \mathrm{d}t$ and matrix H, of size $6n \times 6n$, comes from the linearization of actuated constraint forces, i.e., $\Delta(G_{\mathrm{a}} \underline{\theta}) = G_{\mathrm{a}} \Delta \underline{\theta} + H \, \underline{\hat{\Delta} u}$. The Hessian matrix $\nabla^2_{\underline{\boldsymbol{p}}_I} \mathcal{L}_I$, of size $6(6n+m) \times 6(6n+m)$, is unsymmetric as discussed in section 2.11. In the CG method, the Hessian matrices in eq. (7.12) are defined

as

$$\nabla^2_{\underline{\boldsymbol{\theta}}_I}\mathcal{L}_I = \nabla^2_{\underline{\boldsymbol{\theta}}_I} L_I + \underline{(\underline{1}_1\underline{1}_1^T)\otimes(\nabla^2_{\underline{\theta}}\phi)} + \boxed{\nabla_{\underline{\boldsymbol{\theta}}_I}(\mathbb{F}_I^T\underline{\boldsymbol{\nu}}_I)}, \tag{7.17a}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\boldsymbol{p}}_I}\mathcal{L}_I = \left[\boxed{\mathbb{H}_I} \quad 0\right], \tag{7.17b}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I = \left[\boxed{\mathbb{G}_{\mathrm{a}I}} + \boxed{\boxed{\mathbb{F}_I^T}} \quad 0\right], \tag{7.17c}$$

$$\nabla_{\underline{\boldsymbol{\theta}}_I,\underline{\pi}}\mathcal{L}_I = \underline{\underline{1}_1\otimes(\nabla_{\underline{\theta}}\underline{\psi})^T}, \tag{7.17d}$$

$$\nabla_{\underline{\boldsymbol{p}}_I,\underline{\boldsymbol{d}}_I}\mathcal{L}_I = \mathbb{K}_I, \tag{7.17e}$$

$$\nabla_{\underline{\boldsymbol{p}}_I,\underline{\pi}}\mathcal{L}_I = \left[\begin{array}{c}\underline{(\underline{1}_1\underline{1}_1^T)\otimes(\nabla_{\mathcal{R}}\underline{\psi})^T}\\ \hline 0\end{array}\right]. \tag{7.17f}$$

7.7 Numerical example

Numerical examples are presented to validate the proposed formulation.

7.7.1 Optimal control of flexible robotic arms

Consider a robotic arm consisting of a rigid end-effector and three flexible beams connected by revolute joints. Bars 1, 2, and 3 are all of length 0.4 m. The mass and stiffness properties are listed in tables. 7.1 and 7.2, respectively.

Body	m (Kg or Kg/m)	$I_{22} = I_{33} = 1/2I_{11}$ (Kg·m^2 or Kg·m)
End-effector	2	0.1
Bar 1	19.5	4.0625 10^{-3}
Bar 2 and 3	2.5272	6.823 10^{-5}

Table 7.1: Mass properties of the robotic arm.

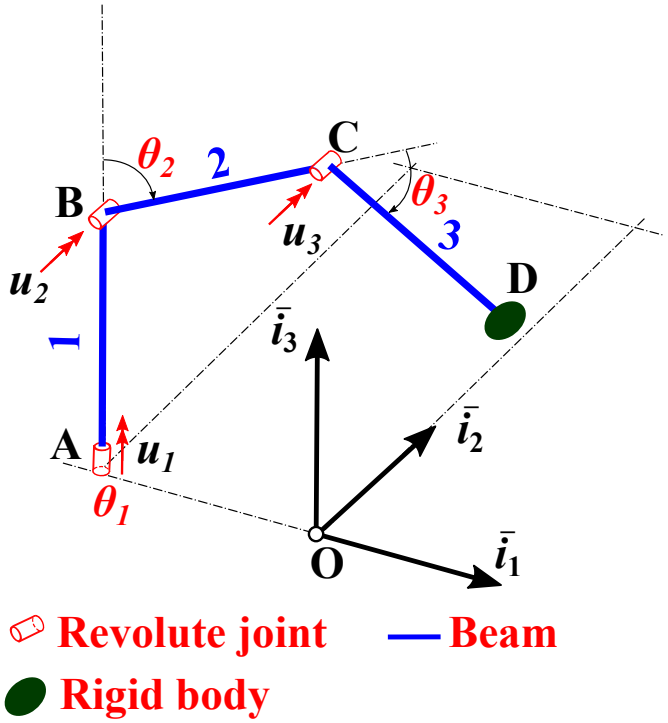

Figure 7.1: Configuration of the robotic arm.

Body	Extension, shear (MPa)	Torsion, bending (KN·m^2)
Bar 1	500, 159.4	67.66, 104.2
Bar 2 and 3	64.8, 20.65	1.137, 1.750

Table 7.2: Stiffness properties of the robotic arm.

The problem is to determine the control torques $u_i, i = 1, 2, 3$ that minimize the cost functional $J = \int_{t=0}^{0.5} u_1^2 + u_2^2 + u_3^2 \, \mathrm{d}t$, and move point point D from position $\underline{r}(0) = (0.7980, 0, 0.4566)$ m at $t = 0$ to $\underline{r}(0.5) = (0.3766, 0.5865, 0.4846)m$ at $t = 0.5$ s, from rest to rest.

The proposed DG method is applied for the problem. Each beam is modeled by 8 2-node elements. The time period $[0, 0.5]$ s is meshed to 20 2-node elements. The predicted control torques are shown in fig. 7.2.

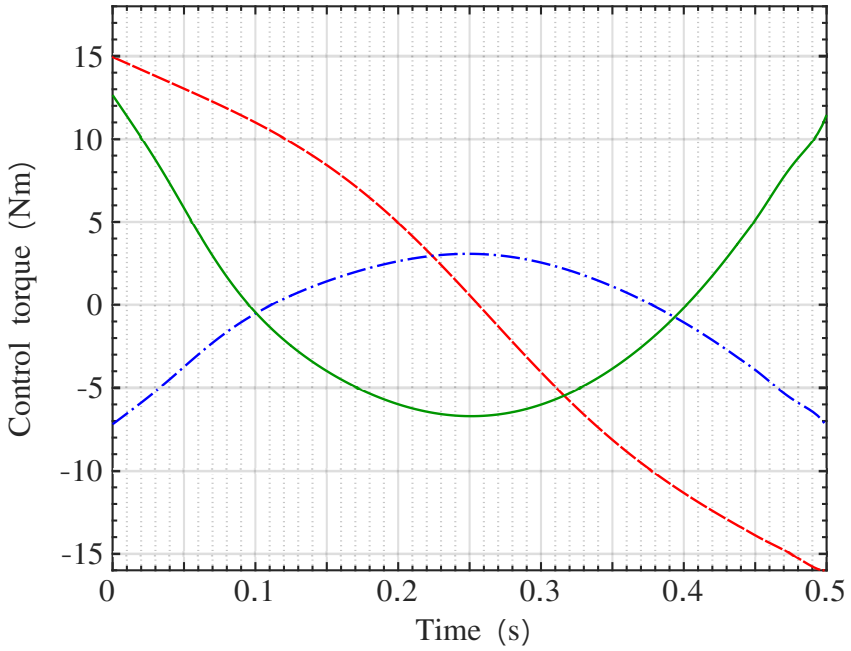

Figure 7.2: Control torques, u_1: dashed line, u_2: dashed-dotted line, u_3: solid line.

7.8 Summary and conclusions

The simultaneous iteration method is used for optimal control and optimization problems. Both of continuous and discontinuous Galerkin methods are applied to the discretization of the governing equations. A strategy is proposed for the factorization of the structured Hessian matrix. The proposed DG method is validated using the example of optimal control of a robotic arm.

Chapter 8: Summary and Conclusions

8.1 Summary and Conclusions of the thesis

In this thesis, a unified Galerkin solver is developed for solving the geometrically exact beam problem, time integration of dynamic equations, determination of periodic solutions, stability analysis of periodic solutions, optimal control and optimization problems.

The representations of rigid-body motion as motion tensor, dual unit quaternions, and dual parameter vectors are presented in Chap. 2. Derivatives and variations of rigid-body motions are defined. Of the central importance is the extended notation that relates the dual number quantities in kinematics and the real number quantities in statics and dynamics. With the help of the extended notion, gradients and Hessian are defined for functions of rigid-body motions. Newton's method for nonlinear equations and optimization problems formulated on manifold $\mathcal{SO}(3)$ are presented.

Galerkin methods require interpolation of rigid-body motion fields. A set of unified interpolation schemes for rigid-body motion is proposed in Chap. 3. The fundamental conclusion of this chapter is that the proposed interpolation schemes for rigid-body motions converge with the same rate as that of interpolation schemes

in the Euclidean space.

Application of the continuous Galerkin method to beam problems leads to the beam formulation presented in Chap. 4. Two fundamental strategies are introduced: *(1)* the curvatures at nodes (the Lobatto points), rather than at the Gauss points, are evaluated for simplicity, and *(2)* the assumed curvature field is constructed via interpolation of nodal curvatures. The strain energy is evaluated from the assumed curvature field thereby avoiding locking.

The discontinuous Galerkin method is developed and applied to the time integration of initial value problems in Chap. 5. The time integration scheme resulting from the dual-SLERP time interpolation is third-order accurate for both of displacements and rotations. The proposed scheme is more accurate than the generalized-α and Radau II-A schemes.

The proposed continuous and discontinuous Galerkin methods are applied to periodic boundary value problems in Chap. 6. The CG method is superior to DG method in determination of periodic solutions: the CG method converges with the expected order for both of stiff and non-stiff problems; the order of accuracy of the DG method, however, decreases for stiff problems. On the other hand, the DG method provides a reliable tool for the stability analysis of periodic solutions, whereas the CG method fails for such problems.

The proposed CG and DG methods are applied to optimal control and optimization problems in Chap. 7. Both of kinematic variables and control inputs are treated as optimization variables and this treatment leads to a simultaneous iteration method. The Hessian is derived analytically to decrease the number of

iterations.

8.2 Future research

The research in this thesis could be extended in the following directions.

In-extensible Kirchhoff beams/cables. The governing equations for geometric exact beams are stiff because the axial/shear and torsion/bending stiffness differ significantly. In fact, the high axial/shear stiffness is akin to a penalty formulation and the resulting deformation is negligible in most practical problems. Stiff equations lead to numerical difficulties. For instance, the convergence order of discontinuous Galerkin methods decreases for stiff periodic problems, as shown in the first example of Chap. 6. Two methods exist to enforce the vanishing of axial/shear strains: *(1)* enforcing the axial and shear strains to vanish in the geometrically exact beam formulation, and *(2)* enforcing the axial strain to vanish in the Kirchhoff beam formulation. Formulations of in-extensible Kirchhoff beams/cables could be investigated in the future.

Stable inversion based feedforward control of flexible multibody systems. The stable inverse dynamics of flexible multibody systems is formulated as an overdetermined two-point boundary value problem (TPBVP) of DAEs. The TPBVP can be discretized by using continuous Galerkin method and the overdetermined boundary conditions can be enforced weakly.

Optimal control and optimization problem with inequality constraints. The research on optimal control/optimization problem is far from complete. For instance,

only the discontinuous Galerkin method for optimal control problem was implemented and inequality constraints were not considered. In the future, the continuous Galerkin method could be implemented and optimizations problem should be investigated in detail. Inequality constraints could be enforced by using primal/dual interior point method.

Optimal control and optimization of periodic boundary value problems. Optimization of periodic solutions are of practical interest and these problems should be investigated in future. Mathematically, optimization of periodic boundary value problems is easier than optimization of initial value problems because no terminal condition is involved.

Appendix A: Fundamental Identities

A.1 Vector identities

Important vector identities will be used throughout this book. If $\underline{a}$, $\underline{b}$, and $\underline{c}$ are three arbitrary vectors, the following identities can be readily verified by painstakingly expanding the various products,

$$\widetilde{(\tilde{a}\,\underline{b})} = \tilde{a}\,\tilde{b} - \tilde{b}\,\tilde{a}, \tag{A.1a}$$

$$\tilde{a}\,\tilde{b} = \underline{b}\,\underline{a}^T - (\underline{a}^T\underline{b})I, \tag{A.1b}$$

$$\tilde{a}\,\tilde{b} - \tilde{b}\,\tilde{a} = \underline{b}\,\underline{a}^T - \underline{a}\,\underline{b}^T, \tag{A.1c}$$

$$\tilde{a}\,\tilde{b} - \underline{a}\,\underline{b}^T = \widetilde{(\tilde{a}\,\underline{b})} - (\underline{a}^T\underline{b})I, \tag{A.1d}$$

$$\widetilde{\tilde{a}\,\underline{b}}\,\underline{c} = (\underline{a}^T\underline{c})\underline{b} - (\underline{b}^T\underline{c})\underline{a}, \tag{A.1e}$$

$$\tilde{a}\,\tilde{b}\,\underline{c} = (\underline{a}^T\underline{c})\underline{b} - (\underline{a}^T\underline{b})\underline{c}, \tag{A.1f}$$

$$\underline{a}\,\underline{b}^T\underline{c} = (\underline{b}^T\underline{c})\underline{a}, \tag{A.1g}$$

$$\underline{a}^T\tilde{b}\,\underline{c} = \underline{b}^T\tilde{c}\,\underline{a} = \underline{c}^T\tilde{a}\,\underline{b}. \tag{A.1h}$$

If $\bar{n}$ is a unit vector and $\underline{a}$ an arbitrary vector, the following identities also hold

$$(\underline{a}^T\bar{n})\bar{n} = \underline{a} + \tilde{n}\tilde{n}\underline{a}, \tag{A.2a}$$

$$\tilde{n}\tilde{n}\tilde{n} = -\tilde{n}, \tag{A.2b}$$

$$\tilde{n}\dot{\tilde{n}}\tilde{n} = 0, \tag{A.2c}$$

where notation $(\cdot)^{\cdot}$ indicates a derivative with respect to time.

A.2 Identities for matrices and vectors

This section presents a set of useful identities that are used throughout this book. The identities involve a matrix A, of size 3×3, and vector $\underline{a}$, of size 3×1 and complement the vector identities presented in eqs. A.1.

$$\text{axial}(\tilde{a}A) = \frac{1}{2}[\text{tr}(A) - A]\underline{a}, \tag{A.3a}$$

$$\text{axial}(A\tilde{a}) = \frac{1}{2}[\text{tr}(A) - A^T]\underline{a}, \tag{A.3b}$$

$$\text{axial}(\tilde{b}\tilde{a}A) = \left[\frac{1}{2}\widetilde{A^T\underline{b}} - \text{axial}(A)\underline{b}^T\right]\underline{a}, \tag{A.3c}$$

$$\text{axial}(\tilde{b}A\tilde{a}) = \left[\frac{1}{2}A\tilde{b} - \underline{b}\,\text{axial}^T(A)\right]\underline{a}, \tag{A.3d}$$

$$\text{tr}(\tilde{a}A) = -2\underline{a}^T\text{axial}(A), \tag{A.3e}$$

$$\text{tr}(\underline{a}\,\underline{a}^T A) = \underline{a}^T\text{symm}(A)\underline{a}. \tag{A.3f}$$

These identities can be verified easily.

A.3 Solution of the vector-product equation

Consider the following statement, $\tilde{a}\underline{x} = \underline{b}$, that can be viewed as a linear system for unknown $\underline{x}$. Because $\tilde{a}\underline{a} = \underline{0}$, the system is singular and $\underline{a}$ forms the null space of the system; the solvability condition is $\underline{a}^T\underline{b} = 0$. The solution of the problem is $\underline{x} = \mu\underline{a} + \alpha\tilde{a}\underline{b}$, where μ is an arbitrary scalar and α a scalar to be solved for. Introducing the solution yields $\underline{b} = \tilde{a}(\mu\underline{a} + \alpha\tilde{a}\underline{b}) = \alpha\tilde{a}\tilde{a}\underline{b}$. Identity (A.1a) yields $\underline{b} = \alpha[\underline{a}\,\underline{a}^T - \|\underline{a}\|^2 I]\underline{b} = -\alpha\|\underline{a}\|^2\underline{b}$, where the solvability condition implies the second equality. Clearly, $\alpha = -1/\|\underline{a}\|^2$ and the solution is

$$\underline{x} = \mu\underline{a} - \frac{\tilde{a}\underline{b}}{\|\underline{a}\|^2}, \tag{A.4}$$

where scalar μ remains undetermined, as expected.

A.4 The polar decomposition theorem

The *polar decomposition theorem* can be stated as follows.

Theorem A.4.1 (Polar decomposition theorem)**.** *An invertible matrix, $G \in \mathbb{R}^{3\times3}$, can be decomposed into the product of a proper orthogonal matrix, $R \in \mathrm{SO}(3)$, by a symmetric matrix, S, as $G = R\,S$. Matrices R and S are defined uniquely if matrix $[\mathrm{tr}(S)I - S]$ is required to be positive-definite.*

Proof. Let the spectral decomposition of positive-definite matrix G^TG be $U^T\mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)U$, where positive eigenvalues, λ_i, $i = 1, 2, 3$, satisfy $\lambda_1 \leq \lambda_2 \leq \lambda_3$. In view of

identity $S^TS = (R^TG)^T(R^TG) = G^TG$, symmetric matrix S can be chosen as $U^T\text{diag}(\pm\sqrt{\lambda_1}, \pm\sqrt{\lambda_2}, \pm\sqrt{\lambda_3})U$. Of these eight choices, two only, $U^T\text{diag}(\pm\sqrt{\lambda_1}, \sqrt{\lambda_2}, \sqrt{\lambda_3})U$, render matrix $[\text{tr}(S)I - S]$ positive-definite. The sign of the lowest eigenvalue is determined by the sign of $\det(G) = \det(R)\det(S) = \det(S)$: choose the positive or negative sign if $\det(G)$ is positive or negative, respectively. □

Polar decomposition theorem (A.4.1) differs slightly from the traditional polar decomposition theorem used in continuum mechanics [142, 163]. The proof above shows that eight different symmetric matrices S satisfy multiplicative decomposition $G = R\,S$. The solution is made unique by imposing an additional condition: in the traditional and present versions of the theorem, matrices S and $[\text{tr}(S)I - S]$ are required to be positive-definite, respectively. When $\det(G) > 0$, the two theorems are identical.

A.5 The dual polar decomposition theorem

The polar decomposition theorem in section A.4 for matrices of size 3×3 is now generalized to dual matrices.

Theorem A.5.1 (Dual polar decomposition theorem)**.** *An invertible dual matrix, $\mathcal{G} \in \mathbb{D}^{3\times3}$, can be decomposed into the product of a dual orthogonal matrix, $\mathcal{R} \in \mathcal{SO}(3)$, by a symmetric dual matrix, $\mathcal{S}$, as $\mathcal{G} = \mathcal{R}\mathcal{S}$. Matrices $\mathcal{R}$ and $\mathcal{S}$ are defined uniquely if it is also required that matrix $[\text{tr}(S)I - S]$ be positive-definite, where S is the primal part of dual matrix $\mathcal{S}$.*

Proof. To prove the theorem, dual matrices $\mathcal{R}$ and $\mathcal{S}$ will be constructed and the

solution will be shown to be unique. First, dual identity $\mathcal{G} = \mathcal{R}\,\mathcal{S}$ is expanded as $(G + \epsilon\, G^\circ) = (R + \epsilon\, R^\circ)(S + \epsilon\, S^\circ)$, which implies

$$G = R\,S, \tag{A.5a}$$

$$G^\circ = R\,S^\circ + R^\circ S. \tag{A.5b}$$

Equation (A.5a) expresses theorem A.4.1, i.e., proper orthogonal tensor R and symmetric matrix S are defined uniquely. Equation (A.5b) implies $R^T G^\circ = S^\circ + (R^T R^\circ)S$, where matrix $R^T R^\circ = \tilde{z}$ is antisymmetric because motion tensor $\mathcal{R}$ is orthogonal. Because matrix S° must be symmetric, $\text{axial}(S^\circ) = \underline{0}$, and extracting the axial part of this equation yields $\text{axial}(R^T G^\circ) = \text{axial}[\tilde{z}S]$. Identity (A.3a) now yields $[\text{tr}(S)I - S]\underline{z} = 2\text{axial}(R^T G^\circ)$, a linear system that can be solved to find $\underline{z}$. The dual parts of orthogonal matrix $\mathcal{R}$ and symmetric matrix $\mathcal{S}$ are found as $R^\circ = R\tilde{z}$ and $S^\circ = R^T G^\circ - \tilde{z}S$, respectively. □

Appendix B: Interpolation Functions

Consider a continuous function $f(\alpha)$ defined in $\alpha \in [0, T]$. The parameter α can represent temporal or spatial coordinates. The function can be non-periodic or periodic with period T. The Chebyshev spectral interpolation is used for the approximation of non-periodic functions and coordinate transformation $\alpha = T(\eta + 1)/2$ is used. The Fourier spectral interpolation is used for the approximation of periodic functions and coordinate transformation $\alpha = T\eta/(2\pi)$ is used.

B.1 Legendre polynomials

Spectral methods are based on orthogonal polynomials originating from the solution of eigenvalue problems for ordinary differential equations, a class of problems known as "Sturm-Liouville problems." Legendre's polynomials, denoted as $P_n(\eta)$, $\eta \in [-1, 1]$, are the solutions of the following Sturm-Liouville problem,

$$\frac{\mathrm{d}}{\mathrm{d}\eta}\left[\left(1-\eta^2\right)\frac{\mathrm{d}P_n}{\mathrm{d}\eta}\right] + n(n+1)P_n = 0, \tag{B.1}$$

with boundary conditions $P_n(-1) = (-1)^n$ and $P_n(1) = 1$. The few lowest-order polynomials are $P_0(\eta) = 1$, $P_1(\eta) = \eta$, $P_2(\eta) = (3\eta^2 - 1)/2$, $P_3(\eta) = (5\eta^3 - 3\eta)/2$.

Alternatively, Legendre's polynomials are generated by the following recurrence relationship

$$(n+1)P_{n+1}(\eta) = (2n+1)\eta P_n(\eta) - nP_{n-1}(\eta), \quad n > 0. \tag{B.2}$$

Figure B.1 shows the six lowest-order Legendre polynomials. Numerous properties of these polynomial are stated in Abramowitz and Stegun [164] or derived in textbooks [165, 166].

The set of polynomials of degree less or equal to N forms a vector space of dimension $N+1$, denoted as $\mathbf{P}_N$. Clearly, the set of Legendre's polynomials up to the N^{th} degree, $\{P_0, \ldots, P_N\}$, forms an orthogonal basis of $\mathbf{P}_N$.

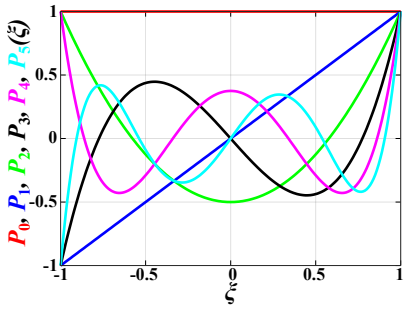

Figure B.1: The six lowest-order Legendre polynomials.

The Gauss-Legendre points or Gauss-Legendre abscissæ play an important role when dealing with quadrature problems; these points are denoted

$$\mu_k, k = 0, 1, \ldots, N-1, \tag{B.3}$$

and are the N real zeros of Legendre's polynomial $P_{N+1}(\eta)$. For all polynomial $p(\eta) \in \mathbf{P}_{2N-1}$, the following quadrature rule holds [165, 166],

$$\int_{-1}^{1} p(\eta)\ \mathrm{d}\eta = \sum_{k=0}^{N-1} p(\mu_k) w_{\mu,k}, \tag{B.4}$$

where $w_{\mu,k}$ are the weights of Gauss-Legendre quadrature rules. Numerical values for the Gauss-Legendre abscissæ and associated weights can be found in Abramowitz and Stegun [164].

Similarly, the Gauss-Lobatto points or Gauss-Lobatto abscissæ are denoted

$$\nu_0 = -1, \nu_N = +1, \text{ and } \nu_k, k = 1, \ldots, N-1, \tag{B.5}$$

where ν_k, $k = 1, \ldots, N-1$, are the $N-1$ real zeros of polynomial $P'_{N+1}(\eta)$ and notation $(\cdot)'$ indicates a derivative with respect to η. For all polynomial $p(\eta) \in \mathbf{P}_{2N-1}$, the following quadrature rule holds [165, 166],

$$\int_{-1}^{1} p(\eta)\ \mathrm{d}\eta = \sum_{k=0}^{N} p(\nu_k) w_{\nu,k}, \tag{B.6}$$

where $w_{\nu,k}$ are the weights of Gauss-Lobatto rule. Numerical values for the Gauss-Lobatto abscissæ and associated weights can be found in Abramowitz and Stegun [164].

For polynomials $p(\eta)$ of degree higher than $2N$, the Gauss-Legendre and Gauss-Lobatto quadrature rules, eqs. (B.4) and (B.6), respectively, become approximate [165, 166]

$$\int_{-1}^{1} p(\eta)\ \mathrm{d}\eta \approx \sum_{k=0}^{N-1} p(\mu_k) w_{\mu,k}, \tag{B.7a}$$

$$\int_{-1}^{1} p(\eta)\ \mathrm{d}\eta \approx \sum_{k=0}^{N} p(\nu_k) w_{\nu,k}. \tag{B.7b}$$

To illustrate quadrature rules B.7, integral $\int_{-1}^{+1} \cos x \, \mathrm{d}x = 2 \sin 1$ was computed numerically using the Gauss-Lobatto and Gauss-Legendre formulas, eqs. (B.7a) and (B.7b), respectively, with an increasing number of sampling points, $N = 2, 3, 4, 5$.

Table B.1 shows the results obtained with the two approaches and calls for the following comments. For the same number of sampling points, the Gauss-Legendre is far more accurate than the Gauss-Lobatto quadrature rule. Similar accuracies are obtained for the Gauss-Legendre and Gauss-Lobatto quadrature rules provided that one additional sampling point is used in the latter approach.

	Gauss-Legendre	**Gauss-Lobatto**
$N = 2$	-0.42	-36
$N = 3$	$3.7\ 10^{-03}$	0.63
$N = 4$	$-1.7\ 10^{-05}$	$-4.9\ 10^{-03}$
$N = 5$	$4.7\ 10^{-08}$	$2.1\ 10^{-05}$

Table B.1: Error in % for the Gauss-Legendre and Gauss-Lobatto quadrature rules.

This observation can be explained as follows: in Gauss-Legendre quadrature, the location of all sampling points is optimized to obtained the most accurate results. In contract, for the Gauss-Lobatto quadrature rule, the function is always sampled at the end points, $\nu_0 = -1$ and $\nu_N = +1$; the location of the remaining intermediate sampling points is then selected to obtained the most accurate results. Because of this superiority of the Gauss-Legendre quadrature rule, it is used extensively in finite element methods for the numerical evaluation of the mass and stiffness matrices, see textbook on the finite element methods [149, 150].

B.2 Lagrange polynomials

Lagrange's polynomials are used extensively to interpolate functions; the are defined as

$$l_n(\eta) = \prod_{k=0,\ k\neq n}^{N} \frac{\eta - \alpha_k}{\alpha_n - \alpha_k}, \tag{B.8}$$

where α_k, $k = 0, 1, \ldots, N$, are called the abscissæ of the interpolation. By construction, Lagrange's polynomials satisfy the following identity

$$l_n(\alpha_k) = \delta_{nk}, \tag{B.9}$$

where δ_{ij} denotes the Kronecker delta. If the discrete values of an arbitrary function are known at the abscissæ, this function is interpolated easily as

$$f(\eta) = \sum_{k=0}^{N} l_k(\eta) f(\alpha_k). \tag{B.10}$$

At the abscissæ, the interpolating function becomes $f(\alpha_j) = \sum_{k=0}^{N} l_k(\alpha_j) f(\alpha_k) = \sum_{k=0}^{N} \delta_{kj} f(\alpha_k) = f(\alpha_j)$. It is verified easily that Lagrange's polynomials satisfy the property of partition of unity

$$\sum_{k=0}^{N} l_k(\eta) = 1. \tag{B.11}$$

Various sets of abscissæ can be used, depending on the desired type of interpolation: equally spaced abscissæ, abscissæ located at the Gauss-Lobatto quadrature points, and abscissæ located at the Gauss-Legendre quadrature points will be illus-

trated below.

B.3 Lagrangian polynomials based on equally spaced abscissæ

In the finite method, Lagrangian polynomials based on equally spaced abscissæ are often used to interpolate the displacement fields within each finite element. In the finite element literature [149,150], these Lagrangian polynomials are often called "shape functions" of the element.

To illustrate the process, the displacement field of the element is assumed to be defined at three nodes along the element. The first two nodes, denoted nodes 1 and 2, are located at the end points of the element, and one additional node, denoted node 3, is inside the element. Let $\underline{u}_1$, $\underline{u}_2$, and $\underline{u}_3$ be the displacement vectors of nodes 1, 2, and 3, respectively. The displacement field of the element is now interpolated based on the displacement vectors at the nodes using *shape functions* denoted $h_1(\eta)$, $h_2(\eta)$, and $h_3(\eta)$,

$$\underline{u}(\eta) = h_1(\eta)\underline{u}_1 + h_2(\eta)\underline{u}_2 + h_3(\eta)\underline{u}_3. \tag{B.12}$$

Node 1, 2, and 3 are located at $\eta = -1$, $+1$, and 0, respectively, which are equally spaced abscissæ. The shape functions are

$$h_1(\eta) = -\frac{1}{2}\eta(1-\eta), \quad h_2(\eta) = \frac{1}{2}\eta(1+\eta), \quad h_3(\eta) = 1-\eta^2. \tag{B.13}$$

Although, the ordering is different, these shape functions are Lagrangian polynomials based on equally spaced abscissæ and could be written in the form given by

eq. (B.8).

The reasoning developed in the previous paragraphs can be repeated for elements presenting two, three, or four nodes, leading to linear, quadratic, or cubic polynomial shape functions, respectively. For elements featuring two nodes located at their end points, the two linear shape functions are

$$h_1(\eta) = \frac{1}{2}(1-\eta), \quad h_2(\eta) = \frac{1}{2}(1+\eta). \tag{B.14}$$

For elements with four nodes, two at their end points and two internal nodes located at $\eta = \mp 1/3$, the four cubic shape functions are

$$\begin{aligned} h_1(\eta) &= \frac{9}{16}(\eta^2 - \frac{1}{9})(1-\eta), \quad h_3(\eta) = -\frac{27}{16}(1-\eta^2)(\eta - \frac{1}{3}), \\ h_2(\eta) &= \frac{9}{16}(\eta^2 - \frac{1}{9})(1+\eta), \quad h_4(\eta) = \quad \frac{27}{16}(1-\eta^2)(\eta + \frac{1}{3}). \end{aligned} \tag{B.15}$$

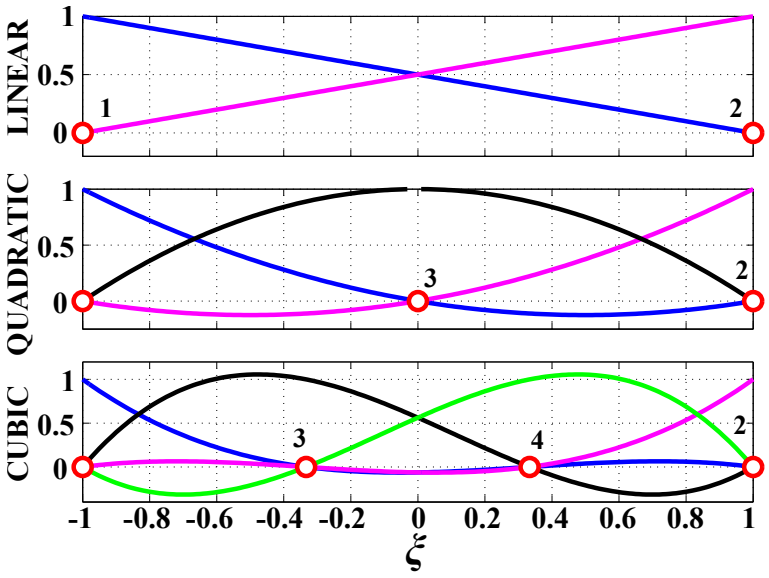

Figure B.2: Linear (top), quadratic (middle), and cubic (bottom) shape functions. Red circles indicate the node locations.

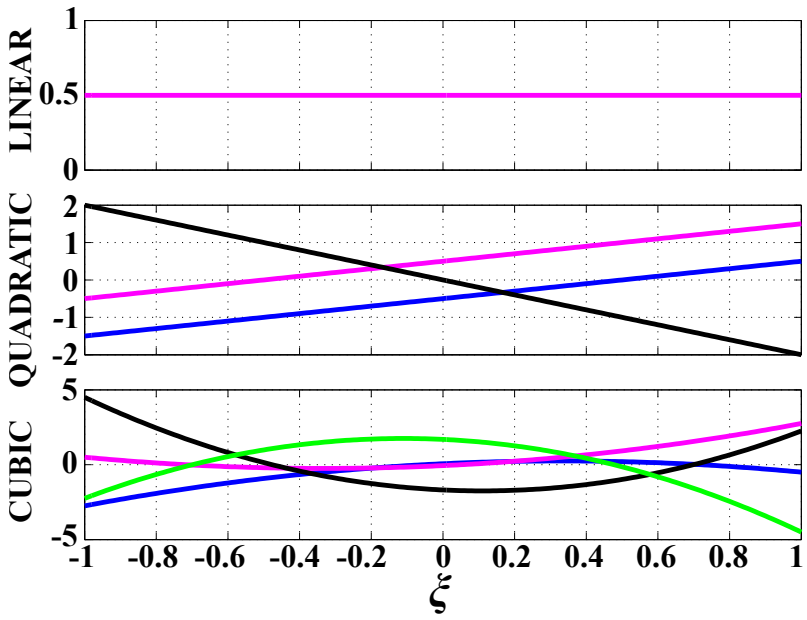

Figure B.3: Derivatives of the shape function. Linear: top figure. Quadratic: middle figure. Cubic: bottom figure.

The shape functions defined by eqs. (B.14), (B.13), and (B.15) are depicted in the top, middle, and bottom portions of fig. B.2, respectively. Derivatives of the shape functions with respect to variable η will also be necessary and are readily computed from eqs. (B.14), (B.13), and (B.15). Figure B.3 depicts these derivatives.

B.4 Lagrangian polynomials based on Gauss-Lobatto and Gauss-Legendre abscissæ

In the spectral [23, 167] and assumed strain [151] formulations of the finite element method, Lagrangian polynomials based on Gauss-Lobatto and Gauss-Legendre abscissæ, respectively, will be used.

Lagrange's polynomials based on the Gauss-Lobatto abscissæ are denoted $l(\eta)$; fig B.4 depicts the six Lagrange polynomials for the case $N = 5$. At the Gauss-

Lobatto quadrature points, the derivatives of Lagrange's polynomials $l_k(\eta)$ with respect to η become

$$d_{j,k} = \frac{\mathrm{d}l_k}{\mathrm{d}\eta}|_{\eta=\nu_j} = \begin{cases} -N(N+1)/4, & j = k = 0, \\ 0, & j = k \in [1, 2, \ldots, N-1], \\ N(N+1)/4, & j = k = N, \\ \dfrac{P_N(\nu_j)}{(\nu_j - \nu_k)P_N(\nu_k)}, & j \neq k. \end{cases} \tag{B.16}$$

These derivatives are stored in matrix d, of size $(N+1) \times (N+1)$, such that entry (j,k) of this matrix are $[d]_{jk} = d_{j,k}$.

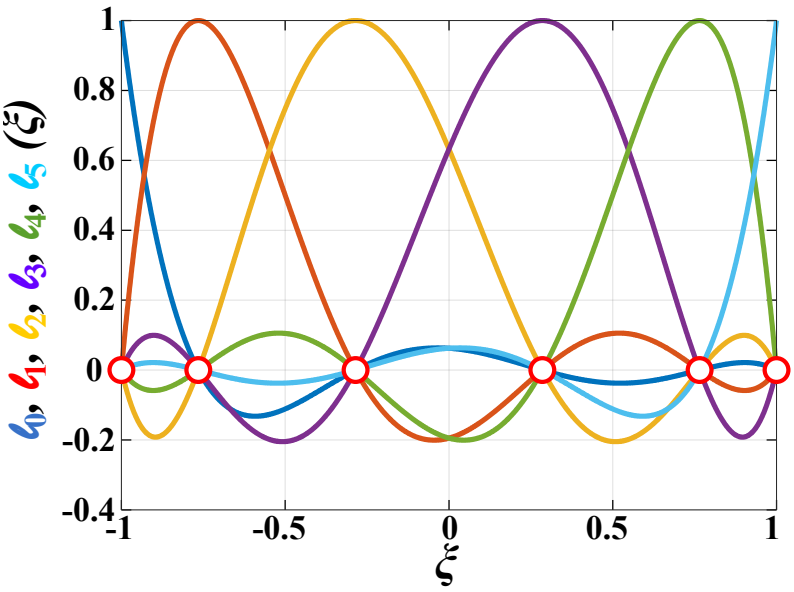

Figure B.4: Lagrange polynomials based on Gauss-Lobatto abscissæ.

Finally, Lagrange's polynomials based on the Gauss-Legendre abscissæ are denoted $m(\eta)$; fig B.5 depicts the six Lagrange polynomials for the case $N = 5$.

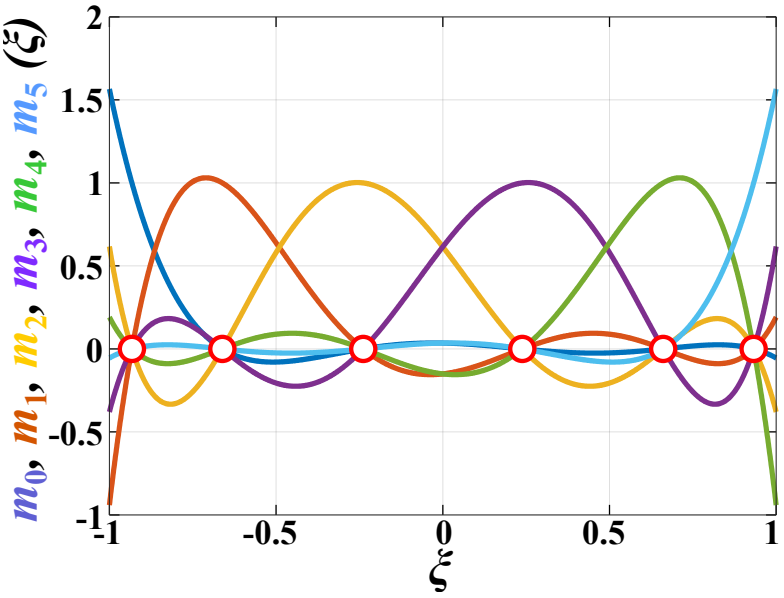

Figure B.5: Lagrange polynomials based on Gauss-Legendre abscissæ.

B.5 Lagrangian polynomials based on Gauss-Radau abscissæ: Radau polynomials

Legendre's polynomials of degree k, denoted as $P_k(\eta)$, $\eta \in [-1, 1]$, are orthogonal polynomials generated by the following recurrence relationship

$$(k+1)P_{k+1}(\eta) = (2k+1)\eta P_{k-1}(\eta) - kP_{k-2}(\eta), \quad k \geq 2. \tag{B.17}$$

The few lowest-order polynomials are $P_0(\eta) = 1$, $P_1(\eta) = \eta$, $P_2(\eta) = (3\eta^2 - 1)/2$, $P_3(\eta) = (5\eta^3 - 3\eta)/2$. The set of polynomials of degree less or equal to N forms a vector space of dimension $N + 1$, denoted as $\mathbf{P}_N$. Clearly, the set of Legendre's polynomials up to the Nth degree, $\{P_0, \ldots, P_N\}$, forms an orthogonal basis of $\mathbf{P}_N$.

The left and right Radau polynomials [125] of degree k are defined as

$$\bar{\ell}_k = \frac{1}{2}(P_k + P_{k-1}), \tag{B.18a}$$

$$\bar{r}_k = \frac{(-1)^k}{2}(P_k - P_{k-1}). \tag{B.18b}$$

The left and right Radau points are zeros of polynomials $\bar{\ell}_k$ and $\bar{r}_k$, respectively. Clearly, the right Radau polynomial $\bar{l}_k$ is orthogonal to any polynomial $p \in \mathbf{P}_{k-2}$. It is verified easily that

$$\bar{\ell}_k(-1) = 0, \quad \bar{\ell}_k(1) = 1, \tag{B.19}$$

$$\bar{r}_k(-1) = 1, \quad \bar{r}_k(1) = 0. \tag{B.20}$$

Consider a polynomial $p \in \mathbf{P}_{k-1}$, integration by parts leads to

$$\int_{-1}^{1} p\bar{\ell}_k' \,\mathrm{d}t = p\bar{\ell}_k\big|_{-1}^{1} - \int_{-1}^{1} p'\bar{\ell}_k \,\mathrm{d}t = p(1), \tag{B.21a}$$

$$\int_{-1}^{1} p\bar{r}_k' \,\mathrm{d}t = p\bar{r}_k\big|_{-1}^{1} - \int_{-1}^{1} p'\bar{r}_k \,\mathrm{d}t = -p(-1), \tag{B.21b}$$

because of identities (B.19) and $p' \in \mathbf{P}_{k-2}$.

B.6 Chebyshev interpolation

The Chebyshev polynomials of the first kind, $T_k(\eta)$, appear as a solution to the Sturm-Liouville problem. An explicit expression of Chebyshev polynomials is

$$T_k(\eta) = \cos(k \arccos \eta), \quad \eta \in [-1, 1]. \tag{B.22}$$

The lowest polynomials are $T_0(\eta) = 1$, $T_1(\eta) = \eta$, $T_2(\eta) = 2\eta^2 - 1$, $T_3(\eta) = 4\eta^3 - 3\eta$, etc. The Chebyshev polynomials can be generated from the following recurrence relationship

$$T_{k+1}(\eta) = 2\eta T_k(\eta) - T_{k-1}(\eta), \ k \geq 0. \tag{B.23}$$

The p^{th} order derivatives of Chebyshev polynomials are given by [168]

$$T_k^{(p)}(\eta) = 2^p k \sum_{l \geq 0, k-p-l \text{ even}}^{k-p} \begin{pmatrix} [k+p-l]/2 - 1 \\ [k-p-l]/2 \end{pmatrix} \begin{pmatrix} ([k+p+l]/2 - 1)! \\ ([k-p+l]/2 \quad)! \end{pmatrix} T_l(\eta), \tag{B.24}$$

where notation $(\cdot)^{(p)}$ indicates the p^{th} order derivatives with respect to η.

Considering the Chebyshev polynomials up to the N^{th} order, the Chebyshev-Gauss-Lobatto quadrature points are given by [165, 166]

$$\eta_i = -\cos(\pi i / N), \quad i \in [0, 1, \cdots, N], \tag{B.25}$$

which are also the extrema of $T_k(\eta)$. The Chebyshev polynomials T_k, $k \in [0, 1, \ldots, N]$,

satisfy the following discreet orthogonality relationships

$$\gamma_{kl} = \sum_{i=0}^{N} T_k(\eta_i) T_l(\eta_i) = \begin{cases} N, & k = l = 0, \\ N/2, & k = l \in [1, 2, \cdots, N], \\ 0, & k \neq l, \end{cases} \tag{B.26}$$

The discretized Chebyshev expansion for the function $f(\eta)$ is $f(\eta) = \sum_{l=0}^{N} f_l T_l(\eta)$, where the coefficients are found as $\gamma_{ll} f_l = \sum_{k=0}^{N} T_l(\eta_k) f(\eta_k)$ by using orthogonality condition (B.26). Introducing the coefficients f_l back to the expansion then yields the interpolation formula

$$f(\eta) = \sum_{k=0}^{N} l_k(\eta) f(\eta_k). \tag{B.27}$$

The spectral interpolation polynomials are

$$l_k(\eta) = \frac{(-1)^{N+k+1}(1-\eta^2) T'_N(\eta)}{c_k N^2 (\eta - \eta_k)}, \tag{B.28}$$

where $c_k = 2$ for $k = 0$ or N and otherwise $c_k = 1$.

B.7 Fourier spectral interpolation and discrete Fourier transformation

The complex exponential functions of period 2π is defined as $\exp(\mathbb{i} k\eta)$, where $\mathbb{i} = \sqrt{-1}$. Consider the complex exponential up to harmonic $N/2$ where N is even,

the quadrature points are equally spaced over a period 2π

$$\eta_k = 2\pi\eta/(N+1), \quad k \in [0, 1, \ldots, N]. \tag{B.29}$$

The complex exponential functions $\exp(ik\eta)$, $k \in [-N/2, \ldots, N/2]$, satisfy the discretized orthogonality

$$\gamma_{kl} = \sum_{j=0}^{N} \exp(\mathbb{i}k\eta_j)\exp(\mathbb{i}l\eta_j) = \begin{cases} N+1, & k = l = 0, \\ 0, & k \neq l. \end{cases} \tag{B.30}$$

The discreet Fourier expansion for the function $f(\eta)$ is $f(\eta) = \sum_{l=-N/2}^{N/2} \hat{f}_l \exp(\mathbb{i}l\eta)$, where the coefficients are found as $\gamma_{ll}\hat{f}_l = \sum_{k=0}^{N} \exp(\mathbb{i}l\eta_k) f(\eta_k)$ by using orthogonality condition (B.30). Introducing the coefficients $\hat{f}_l$ back into the expansion then yields the interpolation formula

$$f(\eta) = \sum_{k=0}^{N} f(\eta_k) \left(\frac{1}{N+1} \sum_{j=-N/2}^{N/2} e^{\mathbb{i}j(\eta-\eta_k)} \right) = \sum_{k=0}^{N} l_k(\eta) f(\eta_k). \tag{B.31}$$

The Fourier spectral interpolation functions, $l_k(\tau)$, are defined as

$$\begin{aligned} l_k(\eta) &= \frac{1}{N+1} \sum_{j=-N/2}^{N/2} e^{\mathbb{i}j(\eta-\eta_k)} = e^{-\mathbb{i}(N+1)x} \, \frac{1-\exp(\mathbb{i}2y)}{(N+1)[1-\exp(\mathbb{i}2x)]} \\ &= e^{-\mathbb{i}(N+1)x} \, \frac{\exp(\mathbb{i}y)}{\exp(\mathbb{i}x)} \, \frac{\exp(-\mathbb{i}y) - \exp(\mathbb{i}y)}{N[\exp(-\mathbb{i}x) - \exp(\mathbb{i}x)]} = \frac{\sin y}{(N+1)\sin x}, \end{aligned} \tag{B.32}$$

where $x = (\eta - \eta_k)/2$ and $y = (N+1)(\eta - \eta_k)/2$. The second equality in eq. (B.32) results from the summation formula for geometric series and the last equality comes

from the Euler formula for complex numbers.

The discrete Fourier transformation operation and its inverse, both of size of $(N+1)\times(N+1)$, are found as

$$F=\frac{2}{N+1}\begin{bmatrix} 1/2 & 1/2 & \cdots & 1/2 \\ \cos\eta_0 & \cos\eta_1 & \cdots & \cos\eta_N \\ \sin\eta_0 & \sin\eta_1 & \cdots & \sin\eta_N \\ \vdots & \vdots & \ddots & \vdots \\ \cos(N/2\eta_0) & \cos(N/2\eta_1) & \cdots & \cos(N/2\eta_N) \\ \sin(N/2\eta_0) & \sin(N/2\eta_1) & \cdots & \sin(N/2\eta_N) \end{bmatrix}, \tag{B.33}$$

$$F^{-1}=\begin{bmatrix} 1 & \cos\eta_0 & \sin\eta_0 & \cdots & \cos(N/2\eta_0) & \sin(N/2\eta_0) \\ 1 & \cos\eta_1 & \sin\eta_1 & \cdots & \cos(N/2\eta_1) & \sin(N/2\eta_1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cos\eta_N & \sin\eta_N & \cdots & \cos(N/2\eta_N) & \sin(N/2\eta_N) \end{bmatrix}.$$

The function values in time and frequency domains are related as

$$\begin{Bmatrix} \hat{f}_0 \\ \hat{f}_1+\hat{f}_{-1} \\ (Hf_1-\hat{f}_{-1})/\mathbb{i} \\ \vdots \\ \hat{f}_{N/2}+\hat{f}_{-N/2} \\ (Hf_{N/2}-\hat{f}_{-N/2})/\mathbb{i} \end{Bmatrix} = F\begin{Bmatrix} f(\eta_0) \\ f(\eta_1) \\ f(\eta_2) \\ \vdots \\ f(\eta_{N-1}) \\ f(\eta_N) \end{Bmatrix}, \quad \text{and} \quad \begin{Bmatrix} f(\eta_0) \\ f(\eta_1) \\ f(\eta_2) \\ \vdots \\ f(\eta_{N-1}) \\ f(\eta_N) \end{Bmatrix} = F^{-1}\begin{Bmatrix} \hat{f}_0 \\ \hat{f}_1+\hat{f}_{-1} \\ (Hf_1-\hat{f}_{-1})/\mathbb{i} \\ \vdots \\ \hat{f}_{N/2}+\hat{f}_{-N/2} \\ (Hf_{N/2}-\hat{f}_{-N/2})/\mathbb{i} \end{Bmatrix}. \tag{B.34}$$

B.7.1 Properties of the spectral interpolation

Both of the Chebyshev and Fourier interpolation functions satisfy

$$l_k(\eta_l) = \begin{cases} 1, & \text{for } k = l, \\ 0, & \text{for } k \neq l, \end{cases} \tag{B.35}$$

which guarantees the interpolation curve passing through the grid points. Furthermore, the interpolation functions satisfy the property of partition of unity (3.2).

Evaluating the derivatives of interpolation scheme (B.27) or (B.31) with respect to η yields

$$f'(\eta) = \sum_{k=0}^{N} l'_k(\eta) f(\eta_k), \tag{B.36}$$

At the grid points, the differentiation function can be represented by matrix I', of size $(N+1) \times (N+1)$. The entries of I' located at the l^{th} row and k^{th} column are denoted as $l'_{l,k} = l'_k(\eta_l)$. Entries $l'_{l,k}$ is evaluated as

$$l'_{l,k} = \frac{2}{T} \begin{cases} -(2N^2+1)/6, & l = k = 0, \\ -\dfrac{\eta_l}{2(1-\eta_l^2)}, & l = k \in [1, 2, \ldots, N-1], \\ (2N^2+1)/6, & l = k = N, \\ \dfrac{c_l(-1)^{l+k}}{c_k(\eta_l - \eta_k)}, & l \neq k, \end{cases} \tag{B.37}$$

for the Chebyshev spectral interpolation and

$$l'_{l,k} = \frac{2\pi}{T} \begin{cases} 0, & \text{for } k = l, \\ \dfrac{(-1)^{l-k}}{2\tan(\eta_l - \eta_k)/2}, & \text{for } k \neq l, \end{cases} \tag{B.38}$$

for the Fourier spectral interpolation.

B.8 B-splines interpolation

Let U be a set of $m+1$ non-decreasing numbers, $\eta_0 \leq \eta_1 \leq \eta_2 \leq \cdots \leq \eta_m$. The scalars η_i, $i = 0, 1, 2, \ldots, m$ are called knots, the set U the knot vector. For the k^{th} normalized B-spline basis function of degree p, the basis function $l_{k,p}(t)$ are defined by the Cox-de Coor recursive formulae

$$l_{k,0}(\eta) = \begin{cases} 1, & \eta \in [\eta_k, \eta_{k+1}), \\ 0, & \text{otherwise}, \end{cases} \tag{B.39}$$

for $p = 0$, and

$$l_{k,p}(\eta) = \frac{\eta - \eta_k}{\eta_{k+p-1} - \eta_k} l_{k,p-1}(\eta) + \frac{\eta_{k+p} - \eta}{\eta_{k+p} - \eta_{k+1}} l_{k+1,p-1}(\eta). \tag{B.40}$$

for $p > 0$ and $k = 0, 1, \ldots, N$. For all k and p, $l_{k,p}(\eta)$ is a non-zero polynomial on $[\eta_k, \eta_{k+p+1})$. The number of knots $m+1$, degree p, and basis functions $N+1$ satisfy $m = N + p + 1$. At a knot of multiplicity j, basis function $l_{k,p}$ is C^{p-j} continuous. Basis functions satisfy the property of partition of unity (3.2), i.e.,

$\sum_{k=0}^{N} l_{k,p}(\eta) = 1$ for $\eta \in [\eta_p, \eta_{N+1}]$. In the knot span $\eta \in [\eta_k, \eta_{k+1})$, the only non-vanishing basis functions are $l_{k-p,p}, \ldots, l_{k,p}$. Efficient algorithms can be developed for the evaluation of the basis functions [169].

The B-spline interpolation in (B.41) are valid for both non-periodic and periodic functions. In the non-periodic case, the so-called clamped B-spline curves is generated, in which the first and last knot are of multiplicity $p+1$,

$$U = \{\underbrace{\eta_0, \ldots, \eta_0}_{p+1 \text{ times}}, \eta_1, \ldots, \eta_{N-p}, \underbrace{\eta_{N-p+1}, \ldots \eta_{N-p+1}}_{p+1 \text{ times}}\},$$

where the knots $\eta_0, \eta_1, \ldots, \eta_{N-p+1}$ can be uniform or non-uniform. Given the basis functions of degree p, $l_{k,p}(\eta)$, $k = 0, 1, \ldots, N$ and corresponding $N+1$ control points g_k, $k = 0, 1, \ldots, N$, the B-spline function is set up as

$$f(\eta) = \sum_{k=0}^{N} l_{k,p}(\eta) g_k, \tag{B.41}$$

where $\eta \in [\eta_0, \eta_{N-p+1}]$.

In general, the B-spline function $f(\eta)$ does not pass through control points g_k. To construct function $f(\eta)$ passing through points $f(\bar{\eta}_0)$, $f(\bar{\eta}_1)$, $\ldots$, $f(\bar{\eta}_N)$, the first step is to determine the control point g_k. Introducing $f(\bar{\eta}_k)$, $k = 0, 1, \ldots, N$ into

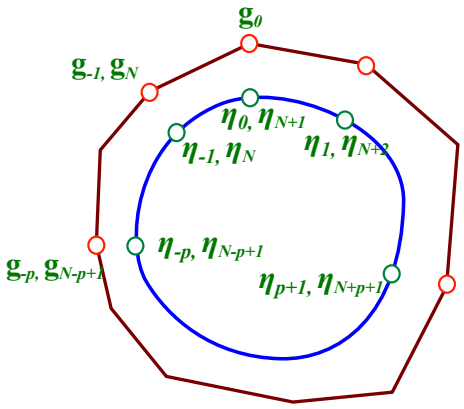

Figure B.6: Warping knots and control points.

eq. (B.41) yields a linear system of size $(N+1) \times (N+1)$

$$\begin{bmatrix} l_{0,p}(\bar{\eta}_0) & \cdots & l_{N,p}(\bar{\eta}_0) \\ \vdots & \ddots & \vdots \\ l_{0,p}(\bar{\eta}_N) & \cdots & l_{N,p}(\bar{\eta}_N) \end{bmatrix} \begin{Bmatrix} g_0 \\ \vdots \\ g_N \end{Bmatrix} = \begin{Bmatrix} f(\bar{\eta}_0) \\ \vdots \\ f(\bar{\eta}_N) \end{Bmatrix}, \tag{B.42}$$

where control points g_k are the $N+1$ unknowns. Evaluating the derivatives of the B-spline curve with respect to η yields $f'(\eta) = \sum_{k=0}^{N} l'_{k,p}(\eta) g(\eta_k)$, where the derivative of the basis functions are

$$l'_{k,p}(\eta) = \frac{l_{k,p-1}(\eta) + (\eta - \eta_k) l'_{k,p-1}(\eta)}{\eta_{k+p-1} - \eta_k} + \frac{(\eta_{k+p} - \eta) l'_{k+1,p-1}(\eta) - l_{k+1,p-1}(\eta)}{\eta_{k+p} - \eta_{k+1}}, \tag{B.43}$$

starting with $l'_{k,0} = 0$.

As shown in figure B.6, to construct a periodic (or closed) B-spline curve of degree p, the $N+1$ control points "wrap around" as $g_{-p}, \ldots, g_{-1}, g_0, \ldots, g_N$ where $g_{-1} = f_N, \ldots, g_{-p} = g_{N-p+1}$. The number of control points are $N + p + 1$. The knot vector also wraps around as $U = \{\eta_{-p}, \ldots, \eta_{-1}, \eta_0, \ldots, \eta_N, \eta_{N+1}, \ldots, \eta_{N+p+1}\}$,

where $\eta_{N+j+1} - \eta_{N+1} = \eta_j - \eta_0$, and $\eta_0 - \eta_{-j} = \eta_{N+1} - \eta_{N+1-j}$, $j = 1, 2, \ldots, p$. There are $m + 1 = (N + p + 1) + p + 1$ knots. Function $f(\eta), \eta \in [\eta_0, \eta_{N-p+1}]$, of degree p defined on the above constructed control points and knot sequence is a periodic function with C^{p-1} continuity at the all grid points f_k. For the periodic case, eq. (B.41) becomes

$$f(\eta) = \sum_{k=0}^{N} l_{k,p}(\eta) g_k = \sum_{k=0}^{N-p} l_{k,p}(\eta) g_k + \sum_{k=N-p+1}^{N} [l_{k,p}(\eta) + l_{k-N-1,p}(\eta)] g_k, \qquad \text{(B.44)}$$

where $\eta \in [\eta_0, \eta_{N+1}]$. Given the value of function $f(\eta)$ on a set of grid points, $f(\bar{\eta}_0)$, $f(\bar{\eta}_1)$, ..., and $f(\bar{\eta}_N)$, the control point g_k, $k = 0, 1, \ldots, N$ are then determined by

$$\begin{bmatrix} l_{0,p}(\bar{\eta}_0) & \cdots & l_{N-p,p}(\bar{\eta}_0) & l_{N-p+1,p}(\bar{\eta}_0) + l_{-p,p}(\bar{\eta}_0) & \cdots & l_{N,p}(\bar{\eta}_0) + l_{-1,p}(\bar{\eta}_0) \\ \vdots & \ddots & \vdots & & & \\ l_{0,p}(\bar{\eta}_N) & \cdots & l_{N-p,p}(\bar{\eta}_N) & l_{N-p+1,p}(\bar{\eta}_N) + l_{-p,p}(\bar{\eta}_N) & \cdots & l_{N,p}(\bar{\eta}_N) + l_{-1,p}(\bar{\eta}_N) \end{bmatrix} \begin{Bmatrix} g_0 \\ \vdots \\ g_N \end{Bmatrix} = \begin{Bmatrix} f(\bar{\eta}_0) \\ \vdots \\ f(\bar{\eta}_N) \end{Bmatrix}, \qquad \text{(B.45)}$$

where the control points g_k are the $N+1$ unknowns. To make the notations of basis functions $l_{k,p}(\eta)$ and the spectral functions in the previous section consistent, degree p subscript is omitted in this paper.

In Chebyshev and Fourier spectral interpolation, the evaluation of derivatives at one grid point requires the function values at all other grid points. In contrast, evaluation of the same derivatives for B-splines only require the function values at the grid points in span $[l - p, l + p]$, see eq. (B.43). When used for the solution differential equations, Chebyshev and Fourier spectral interpolations yield governing equations that couple the variables at all grid points and hence, the bandwidth of

the iteration matrix is $(N+1)m$, where m is the number of degrees of freedom at each grid point. In contrast, the use of B-spline interpolation functions leads to bandwidths of $2pm$. Clearly, the computational burden associated with the use of Chebyshev and Fourier spectral interpolations is far higher that associated with their B-spline counterparts. On the other hand, the further achieve exponential convergence whereas the latter do not.

Bibliography

[1] W. K. Clifford. Preliminary sketch of biquaternions. *Proceedings of the London Mathematical Society*, s1-4(1):381–395, 1871.

[2] A. T. Yang and F. Freudenstein. Application of dual-number quaternion algebra to the analysis of spatial mechanisms. *ASME Journal of Applied Mechanics*, 86:300–308, 1964.

[3] F. M. Dimentberg. The screw calculus and its applications. Technical Report AD 680993, Clearinghouse for Federal and Scientific Technical Information, Virginia, USA, april 1968.

[4] O. Bottema and B. Roth. *Theoretical Kinematics*. Dover Publications, Inc., New York, 1979.

[5] J. M. McCarthy. *An Introduction to Theoretical Kinematics*. The MIT Press, Cambridge, MA, 1990.

[6] J. Angeles. The application of dual algebra to kinematic analysis. In J. Angeles and E. Zakhariev, editors, *Computational Methods in Mechanical Systems*, volume 161, pages 3–31. Springer-Verlag, Heidelberg, 1998.

[7] E. Pennestrì and R. Stefanelli. Linear algebra and numerical algorithms using dual numbers. *Multibody System Dynamics*, 18:323–344, 2007.

[8] M. L. Keler. Kinematics and statics including friction in single-loop mechanisms by screw calculus and dual vectors. *Journal of Engineering for Industry*, 95(2):471–480, 1973.

[9] V. Brodsky and M. Shoham. The dual inertia operator and its application to robot dynamics. *ASME Journal of Mechanical Design*, 116:1089–1095, 1994.

[10] V. Brodsky and M. Shoham. Dual numbers representation of rigid body dynamics. *Mechanism and Machine Theory*, 34:975–991, 1999.

[11] I. S. Fischer. *Dual Number Methods in Kinematics, Statics and Dynamics*. CRC Press, Boca Raton, 1999.

[12] D. Condurache and A. Burlacu. Dual tensors based solutions for rigid body motion parameterization. *Mechanism and Machine Theory*, 74:390–412, 2014.

[13] S. L. Han and O. A. Bauchau. Manipulation of motion via dual entities. *Nonlinear Dynamics*, 85(1):509–524, July 2016.

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

[15] A. Cardona and M. Géradin. A beam finite element non-linear theory with finite rotation. *International Journal for Numerical Methods in Engineering*, 26:2403–2438, 1988.

[16] A. Ibrahimbegović, F. Frey, and I. Kozar. Computational aspects of vector-like parameterization of three-dimensional finite rotations. *International Journal for Numerical Methods in Engineering*, 38(21):3653–3673, 1995.

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

[18] O. A. Bauchau and S. L. Han. Interpolation of rotation and motion. *Multibody System Dynamics*, 31(3):339–370, 2014.

[19] E. B. Dam, M. Koch, and M. Lillholm. Quaternions, interpolation and animation. Technical Report DIKU 98/5, Institute of Computer Science, University of Copenhagen, 1998.

[20] I. Romero. The interpolation of rotations and its application to finite element models of geometrically exact rods. *Computational Mechanics*, 34(2):121–133, 2004.

[21] E. S. Gawlik and M. Leok. Embedding-based interpolation on the special orthogonal group. *SIAM Journal on Scientific Computing*, 2017. To appear.

[22] L. Kavan, S. Collins, C. O'Sullivan, and J. Žára. Dual quaternions for rigid transformation blending. Technical Report TCD-CS-2006-46, The University of Dublin, Trinity College, 2006.

[23] S. L. Han and O. A. Bauchau. Spectral collocation methods for the periodic solution of flexible multibody dynamics. *Nonlinear Dynamics*, 92(4):1599–1618, June 2018.

[24] P. Betsch and P. Steinmann. Frame-indifferent beam element based upon the geometrically exact beam theory. *International Journal for Numerical Methods in Engineering*, 54:1775–1788, 2002.

[25] I. Romero and F. Armero. 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*, 54:1683–1716, 2002.

[26] K. Shoemake. Animating rotation with quaternion curves. *SIGGRAPH Computer Graphics*, 19(3):245–254, 1985.

[27] F. C. Park and B. Ravani. Bézier curves on Riemannian manifolds and Lie groups with kinematics applications. *Journal of Mechanical Design*, 117(1):36–40, 1995.

[28] M. Borri and C. L. Bottasso. An intrinsic beam model based on a helicoidal approximation. Part I: Formulation. Part II: Linearization and finite element implementation. *International Journal for Numerical Methods in Engineering*, 37:2267–2309, 1994.

[29] V. Sonneville, A. Cardona, and O. Brüls. Geometric interpretation of a non-linear beam finite element on the Lie group SE(3). *Archive of Mechanical Engineering*, 64(2):305–329, August 2014.

[30] T. Merlini and M. Morandini. The helicoidal modeling in computational finite elasticity. Part I: Variational formulation. *International Journal of Solids and Structures*, 41(18-19):5351–5381, 2004.

[31] T. Merlini and M. Morandini. The helicoidal modeling in computational finite elasticity. Part II: Multiplicative interpolation. *International Journal of Solids and Structures*, 41(18-19):5383–5409, 2004.

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

[33] X. Pennec. Computing the mean of geometric features – Application to the mean rotation. Technical Report RR-3371, INRIA, 1998.

[34] S. R. Buss and J. P. Fillmore. Spherical averages and applications to spherical splines and interpolation. *ACM Transactions on Graphics*, 20(2):95–126, 2001.

[35] O. Sander. Geodesic finite elements on simplicial grids. *International Journal for Numerical Methods in Engineering*, 92:999–1025, 2012.

[36] C. Gramkow. On averaging rotations. *Journal of Mathematical Imaging and Vision*, 15(1):7–16, 2001.

[37] M. Moakher. Means and averaging in the group of rotations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):1–16, 2002.

[38] K. H. Strobl and G. Hirzinger. Optimal hand-eye calibration. In *Robots and Systems, 2006 IEEE/RSJ International Conference*, pages 4647–4653, 2006.

[39] I. Sharf, A. Wolf, and M. B. Rubin. Arithmetic and geometric solutions for average rigid-body rotation. *Mechanism and Machine Theory*, 45(9):1239–1251, 2010.

[40] R. Hartley, J. Trumpf, Y. C. Dai, and H. D. Li. Rotation averaging. *International Journal of Computer Vision*, 103(3):267–305, 2013.

[41] E. Reissner. On one-dimensional large-displacement finite-strain beam theory. *Studies in Applied Mathematics*, 52:87–95, 1973.

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

[43] V. Giavotto, M. Borri, P. Mantegazza, G. Ghiringhelli, V. Carmaschi, G. C. Maffioli, and F. Mussi. Anisotropic beam theory and applications. *Computers & Structures*, 16(1-4):403–413, 1983.

[44] D. H. Hodges. *Nonlinear Composite Beam Theory*. AIAA, Reston, Virginia, 2006.

[45] E. Carrera, G. Gaetano, and M. Petrolo. *Beam Structures: Classical and Advanced Theories*. John Wiley & Sons, New York, 2011.

[46] D. A. Danielson and D. H. Hodges. A beam theory for large global rotation, moderate local rotation, and small strain. *Journal of Applied Mechanics*, 55(1):179–184, 1988.

[47] A. R. Atilgan, D. H. Hodges, and M. V. Fulton. Nonlinear deformation of composite beams: Unification of cross-sectional and elastica analyses. *Applied Mechanics Reviews*, 44(11):S9–S15, November 1991.

[48] O. A. Bauchau and S. L. Han. Three-dimensional beam theory for flexible multibody dynamics. *Journal of Computational and Nonlinear Dynamics*, 9(4):041011 (12 pages), 2014.

[49] S. L. Han and O. A. Bauchau. Nonlinear three-dimensional beam theory for flexible multibody dynamics. *Multibody System Dynamics*, 34(3):211–242, July 2015.

[50] S. L. Han and O. A. Bauchau. On the solution of Almansi-Michell's problem. *International Journal of Solids and Structures*, 75-76(1):156–171, December 2015.

[51] S. L. Han and O. A. Bauchau. On Saint-Venant's problem for helicoidal beams. *Journal of Applied Mechanics*, 83(2):021009 (14 pages), 2016.

[52] S. L. Han and O. A. Bauchau. On the analysis of thin-walled beams based on Hamiltonian formalism. *Computers & Structures*, 170(1):37–48, July 2016.

[53] S. L. Han and O. A. Bauchau. Nonlinear, three-dimensional beam theory for dynamic analysis. *Multibody System Dynamics*, 41(2):173–200, 2017.

[54] S. L. Han and O. A. Bauchau. On the nonlinear extension-twist coupling of beams. *European Journal of Mechanics / A Solids*, 2018. To appear.

[55] A. Ibrahimbegović. On finite element implementation of geometrically nonlinear Reissner's beam theory: Three-dimensional curved beam elements. *Computer Methods in Applied Mechanics and Engineering*, 122(1-2):11–26, 1995.

[56] J. Argyris. An excursion into large rotations. *Computer Methods in Applied Mechanics and Engineering*, 32(1-3):85–155, 1982.

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

[58] P. Betsch, A. Menzel, and E. Stein. On the parameterization of finite rotations in computational mechanics. A classification of concepts with application to smooth shells. *Computer Methods in Applied Mechanics and Engineering*, 155(3-4):273–305, 1998.

[59] O. A. Bauchau and L. Trainelli. The vectorial parameterization of rotation. *Nonlinear Dynamics*, 32(1):71–92, 2003.

[60] S. Ghosh and D. Roy. Consistent quaternion interpolation for objective finite element approximation of geometrically exact beam. *Computer Methods in Applied Mechanics and Engineering*, 198(3-4):555–571, 2008.

[61] G. A. Hegemier and S. Nair. A nonlinear dynamical theory for heterogeneous, anisotropic, elastic rods. *AIAA Journal*, 15(1):8–15, January 1977.

[62] D. H. Hodges. A mixed variational formulation based on exact intrinsic equations for dynamics of moving beams. *International Journal of Solids and Structures*, 26(11):1253–1273, March 1990.

[63] D. H. Hodges. Geometrically exact, intrinsic theory for dynamics of curved and twisted anisotropic beams. *AIAA Journal*, 41(6):1131–1137, June 2003.

[64] D. Zupan and M. Saje. Finite-element formulation of geometrically exact three-dimensional beam theories based on interpolation of strain measures. *Computer Methods in Applied Mechanics and Engineering*, 192:5209–5248, 2003.

[65] P. Češarek, M. Saje, and D. Zupan. Dynamics of flexible beams: Finite-element formulation based on interpolation of strain measures. *Finite Elements in Analysis and Design*, 72:47–63, 2013.

[66] F. A. McRobie and J. Lasenby. Simo-Vu Quoc rods using Clifford algebra. *International Journal for Numerical Methods in Engineering*, 45(4):377–398, 1999.

[67] T. Merlini and M. Morandini. The helicoidal modeling in computational finite elasticity. Part I: Variational formulation. Part II: Multiplicative interpolation. *International Journal of Solids and Structures*, 41(18-19):5351–5409, 2004.

[68] O. Sander. Geodesic finite elements for Cosserat rods. *International Journal for Numerical Methods in Engineering*, 82(13):1645–1670, 2010.

[69] V. Sonneville, A. Cardona, and O. Brüls. Geometrically exact beam finite element formulated on the special Euclidean group SE(3). *Computer Methods in Applied Mechanics and Engineering*, 268(1):451–474, 2014.

[70] F. Demoures, F. Gay-Balmaz, M. Kobilarov, and T. S. Ratiu. Multisymplectic Lie group variational integrator for a geometrically exact beam in R^3. *Communications in Nonlinear Science and Numerical Simulation*, 19(10):3492–3512, 2014.

[71] S. L. Han and O. A. Bauchau. On the global interpolation of motion. *Computer Methods in Applied Mechanics and Engineering*, 337(10):352–386, 2018.

[72] S. R. Eugster, C. Hesch, P. Betsch, and Ch. Glocker. Director-based beam finite elements relying on the geometrically exact beam theory formulated in skew coordinates. *International Journal for Numerical Methods in Engineering*, 97(2):111–129, 2014.

[73] C. W. Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliff, N.J., 1971.

[74] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II : Stiff and Differential-Algebraic Problems*. Springer, Berlin, 1996.

[75] J. Wang, J. Rodriguez, and R. Keribar. Integration of flexible multibody systems using Radau IIA algorithms. *Journal of Computational and Nonlinear Dynamics*, 5(4):041008–14 pages, 2010.

[76] M. Géradin and A. Cardona. *Flexible Multibody System: A Finite Element Approach*. John Wiley & Sons, New York, 2001.

[77] O. A. Bauchau. *Flexible Multibody Dynamics*. Springer, Dordrecht, Heidelberg, London, New-York, 2011.

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

[79] L. O. Jay and D. Negrut. Extensions of the HHT-α method to differential algebraic equations in mechanics. *Electronic Transactions on Numerical Analysis*, 26(1):190–208, 2007.

[80] J. Chung and G. M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, 60:371–375, 1993.

[81] M. Arnold and O. Brüls. Convergence of the generalized-α scheme for constrained mechanical systems. *Multibody System Dynamics*, 18(2):185–202, 2007.

[82] A. Cardona and M. Géradin. Time integration of the equations of motion in mechanism analysis. *Computers & Structures*, 33(3):801–820, 1989.

[83] T. R. J. Hughes and M. Hulbert. Space-time finite element formulations for elasto-dynamics: Formulation and error estimates. *Computer Methods in Applied Mechanics and Engineering*, 66:339–363, 1988.

[84] G.M. Hulbert. Time finite element methods for structural dynamics. *International Journal for Numerical Methods in Engineering*, 33(2):307–331, 1992.

[85] M. Borri. Helicopter rotor dynamics by finite element time approximation. *Computers & Mathematics with Applications*, 12(1, Part A):149–160, 1986.

[86] O. A. Bauchau and C. H. Hong. Nonlinear response and stability analysis of beams using finite elements in time. *AIAA Journal*, 26(9):1135–1142, September 1988.

[87] M. Borri, C. L. Bottasso, and L. Trainelli. Integration of elastic multibody systems by invariant conserving/dissipating algorithms. Part I: Formulation. *Computer Methods in Applied Mechanics and Engineering*, 190:3669–3699, 2001.

[88] M. Borri and C. L. Bottasso. A general framework for interpreting time finite element formulations. *Computational Mechanics*, 13:133–142, 1993.

[89] C. D. Bailey. Application of Hamilton's law of varying action. *AIAA Journal*, 13:1154–1157, September 1975.

[90] H. H. E. Leipholtz. Space-time formulation of Hamilton's law. *Mechanics Research Communications*, 9:117–323, May 1982.

[91] L.J. Hou and D.A. Peters. Periodic trim solutions with hp-version finite elements in time. *Mathematical and Computer Modelling*, 17(3):29–46, 1993.

[92] P.K. Gudla and R. Ganguli. Discontinuous Galerkin finite element in time for solving periodic differential equations. *Computer Methods in Applied Mechanics and Engineering*, 196(1):682–696, 2006.

[93] G. Floquet. Sur les équations différentielles linéaires à coefficients périodiques. *Annales scientifiques de l'École Normale Supérieure*, 12:47–88, 1883.

[94] T. R. Kane and D. Sobala. A new method for attitude stabilization. *AIAA Journal*, 1(6):1365–1367, 1963.

[95] J. Dugundji and J. H. Wendell. Some analysis methods for rotating systems with periodic coefficients. *AIAA Journal*, 21(6):890–897, 1983.

[96] P. P. Friedmann. Numerical methods for determining the stability and response of periodic systems with applications to helicopter rotor dynamics and aeroelasticity. *Computers and Mathematics with Applications*, 12A:131–148, 1986.

[97] G. H. Gaonkar and D. A. Peters. Review of Floquet theory in stability and response analysis of dynamic systems with periodic coefficients. In *R.L. Bisplinghoff Memorial Symposium Volume on Recent Trends in Aeroelasticity, Structures and Structural Dynamics, Feb 6-7, 1986*, pages 101–119. University Press of Florida, Gainesville, 1986.

[98] N. M. Wereley. *Analysis and control of linear periodically time varying systems.* PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 1991.

[99] G. W. Hill. On the part of the motion of the lunar perigee which is a function of the mean motions of the sun and moon. *Acta Mathematica*, 8(1):1–36, 1886.

[100] P. W. Likins and K. G. Lindh. Infinite determinant methods for stability analysis of periodic-coefficient differential equations. *AIAA Journal*, 8(4):680–686, 1970.

[101] S. T. Noah and G. R. Hopkins. A generalized Hill's method for the stability analysis of parametrically excited dynamic systems. *Journal of Applied Mechanics*, 49(1):217–223, 1982.

[102] G. Von Groll and D. J. Ewins. The harmonic balance method with arc-length continuation in rotor/stator contact problems. *Journal of Sound and Vibration*, 241(2):223–233, 2001.

[103] O. A. Bauchau and Y. G. Nikishkov. An implicit Floquet analysis for rotorcraft stability evaluation. *Journal of the American Helicopter Society*, 46:200–209, 2001.

[104] O. A. Bauchau and J. L. Wang. Stability analysis of complex multibody systems. *Journal of Computational and Nonlinear Dynamics*, 1(1):71–80, January 2006.

[105] E. J. Haug and J. S. Arora. *Applied Optimal Design: Mechanical and Structural Systems*. John Wiley & Sons, New York, 1979.

[106] E. J. Haug. Design sensitivity analysis of dynamic systems. In *Computer aided optimal design: structural and mechanical systems*, pages 705–755. Springer, Berlin, Germany, 1987.

[107] D. Bestle and P. Eberhard. Analyzing and optimizing multibody systems. *Journal of Structural Mechanics*, 20(1):67–92, 1992.

[108] Yitao Zhu, Daniel Dopico, Corina Sandu, and Adrian Sandu. Dynamic response optimization of complex multibody systems in a penalty formulation using adjoint sensitivity. *Journal of Computational and Nonlinear Dynamics*, 10(3):1–9, May 2015.

[109] Alfonso Callejo, Javier García de Jalón, Pablo Luque, and D. A. Mántaras. Sensitivity-based, multi-objective design of vehicle suspension systems. *Journal of Computational and Nonlinear Dynamics*, 10(3):031008, 2015.

[110] K. Nachbagauer, S. Oberpeilsteiner, K. Sherif, and W. Steiner. The use of the adjoint method for solving typical optimization problems in multibody dynamics. *Journal of Computational and Nonlinear Dynamics*, 10(6):061011, 2015.

[111] C.L. Bottasso and A. Croce. Optimal control of multibody systems using an energy preserving direct transcription method. *Multibody System Dynamics*, 12(1):17–45, 2004.

[112] O. Brüls, E. Lemaire, P. Duysinx, and P. Eberhard. Optimization of multibody systems and their structural components. In *Multibody Dynamics*, pages 49–68. Springer, 2011.

[113] E. Tromme, O. Brüls, J. Emonds-Alt, M. Bruyneel, G. Virlez, and P. Duysinx. Discussion on the optimization problem formulation of flexible components in multibody systems. *Structural and Multidisciplinary Optimization*, 48(6):1189–1206, 2013.

[114] F. B. Hildebrand. *Advanced Calculus for Applications*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, second edition, 1976.

[115] M. Chasles. Note sur les propriétés générales du système de deux corps semblables entre eux et placés d'une manière quelconque dans l'espace; et sur le déplacement fini, ou infiniment petit d'un corps solide libre. *Bulletin des Sciences Mathématiques de Férussac*, 14:321–326, 1830.

[116] A. R. Klumpp. Singularity-free extraction of a quaternion from a direction-cosine matrix. *Journal of Spacecraft and Rockets*, 13:754–755, December 1976.

[117] S. W. Shepperd. Quaternion from rotation matrix. *Journal of Guidance and Control*, 1:223–224, May 1978.

[118] O. A. Bauchau and J. Y. Choi. The vector parameterization of motion. *Nonlinear Dynamics*, 33(2):165–188, 2003.

[119] J. I. Neimark and N. A. Fufaev. *Dynamics of Nonholonomic Systems*. American Mathematical Society, Providence, Rhode Island, 1972.

[120] J. G. Papastavridis. *Analytical Mechanics: A Comprehensive Treatise on the Dynamics of Constrained systems; for Engineers, Physicists and Mathematician*. Oxford University Press, Oxford, UK, 2002.

[121] D. T. Greenwood. *Advanced Dynamics*. Cambridge University Press, Cambridge, 2003.

[122] B. Ravani and B. Roth. Motion synthesis using kinematic mappings. *Journal of Mechanisms, Transmissions, and Automation in Design*, 105(3):460–467, 1983.

[123] W. D. Curtis, A. L. Janin, and K. Zikan. A note on averaging rotations. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, pages 377–385, Seattle, WA, September 1993.

[124] M. Humbert, N. Gey, J. Muller, and C. Esling. Determination of a mean orientation from a cloud of orientations. Application to electron back-scattering pattern measurements. *Journal of Applied Crystallography*, 29(6):662–666, 1996.

[125] H. T. Huynh. Collocation and Galerkin time-stepping methods. In *19th AIAA Computational Fluid Dynamics*. American Institute of Aeronautics and Astronautics, 2009.

[126] J. J. Kuffner. Effective sampling and distance metrics for 3D rigid body path planning. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA '04)*, volume 4, pages 3993–3998, New Orleans, LA, 2004.

[127] J. Stuelpnagel. On the parameterization of the three-dimensional rotation group. *SIAM Review*, 6(4):422–430, 1964.

[128] D. Condurache. Dual algebra solutions to the extended Wahba problem. *Romanian Journal of Mechanics*, 1(1):3–44, 2017.

[129] P. Davenport. A vector approach to the algebra of rotations with applications. Technical Report X-546-65-437, NASA, November 1965.

[130] F. L. Markley and D. Mortari. Quaternion attitude estimation using vector observations. *Journal of the Astronautical Sciences*, 48(2/3):359–380, 2000.

[131] N. Mackey. Hamilton and Jacobi meet again: Quaternions and the eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 16(2):421–435, 1995.

[132] C. Bouby, D. Fortuné, W. Pietraszkiewicz, and C. Vallée. Direct determination of the rotation in the polar decomposition of the deformation gradient by maximizing a Rayleigh quotient. *Journal of Applied Mathematics and Mechanics*, 85(3):155–162, 2005.

[133] K. Daniilidis. Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286–298, 1999.

[134] G. Wahba. A least squares estimate of satellite attitude. *SIAM Review*, 7(3):409–409, 1965.

[135] L. C. Martins and P. Podio-Guidugli. A variational approach to the polar decomposition theorem. *Rendiconti Accademia Nazionale dei Lincei, Serie VIII*, 66:487–493, 1979.

[136] L. C. Martins and P. Podio-Guidugli. An elementary proof of the polar decomposition theorem. *The American Mathematical Monthly*, 87(4):288–290, 1980.

[137] B. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, April 1987.

[138] R. R. Hartley and R. P. Behringer. Logarithmic rate dependence of force networks in sheared granular materials. *Nature*, 421:928–931, 2003.

[139] J. Lu and P. Papadopoulos. On the direct determination of the rotation tensor from the deformation gradient. *Mathematics and Mechanics of Solids*, 2(1):17–26, 1997.

[140] A. Morawiec. A note on mean orientation. *Journal of Applied Crystallography*, 31(5):818–819, 1998.

[141] G. Grioli. Una proprietà di minimo nella cinematica delle deformazioni finite. *Bollettino dell'Unione Matematica Italiana*, 2:252–255, 1940.

[142] L. E. Malvern. *Introduction to the Mechanics of a Continuous Medium*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1969.

[143] L. Kavan and J. Žára. Spherical blend skinning: A real-time deformation of articulated models. In *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games*, I3D '05, pages 9–16, New York, NY, USA, 2005.

[144] O. A. Bauchau, A. Epple, and S. D. Heo. Interpolation of finite rotations in flexible multibody dynamics simulations. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, 222(K4):353–366, 2008.

[145] T. Merlini and M. Morandini. Computational shell mechanics by helicoidal modeling, II: Shell element. *Journal of Mechanics of Materials and Structures*, 6(5):693–728, 2011.

[146] L. Kavan, S. Collins, J. Žára, and C. O'Sullivan. Geometric skinning with approximate dual quaternion blending. *ACM Transactions on Graphics*, 27(4):105:1–105:23, 2008.

[147] T. Merlini and M. Morandini. Computational shell mechanics by helicoidal modeling, I: Theory. II: Shell element. *Journal of Mechanics of Materials and Structures*, 6(5):659–728, 2011.

[148] O. Sander. Geodesic finite elements of higher order. *IMA Journal of Numerical Analysis*, 36(1):238–266, 2016.

[149] T. J. R. Hughes. *The Finite Element Method*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1987.

[150] K. J. Bathe. *Finite Element Procedures*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1996.

[151] K. J. Bathe and E. N. Dvorkin. A formulation of general shell elements - The use mixed interpolation of tensorial components. *International Journal for Numerical Methods in Engineering*, 22:697–722, 1986.

[152] V. V. Volovoi, D. H. Hodges, V. L. Berdichevsky, and V. G. Sutyrin. Dynamic dispersion curves for non-homogeneous, anisotropic beams with cross sections of arbitrary geometry. *Journal of Sound and Vibration*, 215:1101–1120, 1998.

[153] K. J. Bathe and S. Bolourchi. Large displacement analysis of three-dimensional beam structures. *International Journal for Numerical Methods in Engineering*, 14(7):961–986, 1979.

[154] M. A. Crisfield. A fast incremental/iterative solution procedure that handles "snap-through". *Computers & Structures*, 13(1-3):55–62, 1981.

[155] D. A. DaDeppo and R. Schmidt. Instability of clamped-hinged circular arches subjected to a point load. *Journal of Applied Mechanics*, 42:894–896, 1975.

[156] C. Lanczos. *The Variational Principles of Mechanics*. Dover Publications, Inc., New York, 1970.

[157] O. A. Bauchau and N. J. Theron. Energy decaying scheme for non-linear beam models. *Computer Methods in Applied Mechanics and Engineering*, 134(1-2):37–56, 1996.

[158] M. Borri, G.L. Ghiringhelli, M. Lanz, P. Mantegazza, and T. Merlini. Dynamic response of mechanical systems by a weak Hamiltonian formulation. *Computers & Structures*, 20(1):495–508, 1985.

[159] H. Hochstadt. *Differential Equations*. Dover Publications, Inc., New York, 1964.

[160] A. H. Nayfeh and D. T. Mook. *Nonlinear Oscillations*. John Wiley & Sons, New York, 1979.

[161] L.N. Trefethen. *Approximation Theory and Approximation Practice*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2012.

[162] R. P. Coleman. Theory of self-excited mechanical oscillations of hinged rotor blades. NACA Report WR-L-308, Langley Research Center, 1943.

[163] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, second edition, 1989.

[164] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions*. Dover Publications, Inc., New York, 1964.

[165] D. Funaro. *Polynomial Approximation of Differential Equations*. Springer-Verlag, Berlin, Germany, 1992.

[166] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral Methods for Time-Dependent Problems*. Cambridge University Press, Cambridge, 2007.

[167] S. L. Han and O. A. Bauchau. Spectral formulation for geometrically exact beams: A motion interpolation based approach. *AIAA Journal*, 2019. To appear.

[168] E. H. Doha. The coefficients of differentiated expansions and derivatives of ultraspherical polynomials. *Computers & Mathematics with Applications*, 21(2):115–122, 1991.

[169] L. Piegl and W. Tiller. *The Nurbs Book*. Springer-Verlag, Berlin, New Jersey, second edition, 1997.