Interpolation of Finite Rotations in Flexible Multibody Dynamics Simulations*

Olivier A. Bauchau, Alexander Epple, and SeunDo Heo
Daniel Guggenheim School of Aerospace Engineering,
Georgia Institute of Technology, Atlanta, GA, USA.

Abstract

The representation and manipulation of finite rotations and the modeling of flexible bodies are at the heart of flexible multibody systems dynamics simulations. When dealing with finite element formulations, the interpolation of finite rotation fields becomes a critical issue. Objectivity, *i.e.* the invariance of interpolated strain fields to the addition of a rigid body motion, has been show to be an important problem. Furthermore, the manipulation of finite rotations and the unavoidable associated singularities must be dealt with in a rational manner. The various approaches to these two problems are reviewed in this paper. Simple algorithms that achieve objectivity are proposed. Furthermore, these algorithms are able to deal with finite rotations of arbitrary large magnitude through a simple rescaling operation. The choice of unknowns, total versus incremental, is also discussed as it affects the performance of the computational process. Numerical examples are presented that demonstrate the ability of the proposed approach to handle finite rotations of arbitrary magnitudes in complex, flexible multibody systems.

1 Introduction

Rigid multibody systems are characterized by two distinguishing features: the rigid body components of the system undergo finite relative rotations and these components are connected by mechanical joints that impose restrictions on their relative motion. When dealing with flexible multibody systems, each component of the system could be flexible, adding to the complexity and nonlinearity of the problem. The dynamic equations of motion of a rigid body were first derived by Euler [1, 2] and are know as Euler's Laws; Euler also provided a geometric interpretation of finite rotations stated in Euler's theorem on finite rotations [3]; finally, the analysis of the elastic deformation of a slender beam was first treated by Euler and is known as Euler's Elastica [4]. These towering contributions clearly make Euler the father of flexible multibody dynamics.

The representation and manipulation of finite rotations, and the modeling of flexible bodies are at the heart of flexible multibody systems dynamics simulations: these two topics are the focus of the present paper. To achieve computational efficiency, flexible components are often idealized as thin structures, such as as beams or shells, which are often modeled based on Cosserat curve or surface approaches, respectively. The kinematics of these problems are then described in terms of two fields, a displacement field and a rotation field.

The exact treatment of finite rotations is particularly important in multibody dynamics because finite rotations associated with the finite relative motions of the system's components

Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics, **222**(K4), pp 353-366, 2008.

are combined with the finite elastic motions of the flexible components. Consider, for instance, the motion of a helicopter rotor blade in which elastic deformations of the blade are superimposed onto the rigid body rotation of the entire rotor. Finite rotations do not form a linear space: Rodrigues [5] was the first to study how two rotations are combined into a single rotation. In this paper the expression "composition of finite rotation" is used to denote the combination of rotations to underline the fact these quantities are not additive.

The elastic deformation of a slender beam was first treated by Euler [4]. Major advances in beam theory came with the work of Reissner [6, 7, 8], who considered problems involving finite strains and spatially curved members. Simo and coworkers [9, 10] then developed the theory that is now referred to as "geometrically exact beam theory." The kinematics of the problem are based on the Euler-Bernoulli assumptions, but the strain-displacement equations are exact for arbitrarily large displacements and rotations. In most applications of this theory, strains are assumed to remain small, and hence, linear constitutive laws are used.

While classical formulations of flexible multibody systems are based on the floating frame of reference approach [11, 12], other approaches have been proposed such as the co-rotational formulation [13]; a comprehensive review of the various methods in use is given by Shabana [14]. More recently, the finite element method has found increasing use in the analysis of flexible multibody systems, Belytschko and Hsieh [15], Cardona *et al.* [16], or Bauchau [17]; a textbook by Géradin and Cardona [18] is devoted to this topic. At the heart of this numerical method is the interpolation of displacement fields within each element. Interpolation is a linear operation that has been used for decades to interpolate displacement fields, which form a linear space.

Application of the same, linear interpolation technique to finite rotation fields has been the subject of controversy, because finite rotation fields do not form a linear space. Crisfield and Jelenić [19] were the first to point out a major deficiency of this interpolation technique: its lack of objectivity. By definition, a rigid body motion generates no strains; hence, the strain field which is generated by a given displacement field has to be unaffected by the addition of a rigid body motion to the displacement field. If a computational scheme satisfies this condition, it is said to be "objective." Crisfield and Jelenić [19, 20] showed that classical interpolation formulæ applied to finite rotation fields violate the objectivity criterion. They prove the non-objectivity of the direct interpolation of total rotations [21], incremental rotations [22] and iterative rotations [10]. Crisfield and Jelenić argue that "all of these formulations can be regarded as stemming from the same family, for which the following is valid: the interpolation is applied to the rotation between a particular reference configuration and the current configuration. With hindsight, the nature of this interpolation is bound to make all of these formulations non-objective. The rotations interpolated in this way in general include rigid body rotations, so that the error, introduced by the interpolation, makes the resulting strain measures dependent on the rigid body rotation." They also point out, however, that while the errors in the computed strain field are small and decrease with mesh p- or h-refinement, lack of objectivity persists if rotation increments or Newton-Raphson updates are interpolated. Crisfield and Jelenić proposed a novel interpolation technique that guarantees objectivity by splitting rotations into rigid and elastic components: the sole elastic component is interpolated. This approach is akin to the co-rotational formulation [13], but retains the fully nonlinear strain-configuration equations, rather than their linearized counterparts.

Betsch and Steinmann [23] proposed an alternative approach to achieving objectivity: instead of interpolating finite rotation parameters, they interpolate the unit vectors forming the columns of the finite rotation tensor and proved that this approach also satisfies the objectivity criterion. Linear interpolation of unit vectors, however, does not yield unit vectors, nor does it preserve their orthogonality. Special procedures were developed to guarantee that the interpolated results lead to orthogonal rotation tensors. Numerical examples were shown that demonstrate the accuracy of numerical predictions. Romero *et al.* [24, 25] presented a comparison of different interpolation methods including the direct interpolation of finite rotations, the interpolation method proposed by Crisfield and Jelenić [19], and two new approaches, based on 1) the non-orthogonal interpolation of rotations with modification of geometrically exact beam

theory and 2) the isoparametric interpolation of rotations followed by orthogonalization using polar decomposition. Numerical tests of all four methods showed that with the exception of the direct interpolation of finite rotations, the other methods are objective, path-independent and preserve the orthogonality of the rotation tensor. However, the proposed interpolation approaches were shown to soften structural response, and could converge to erroneous solutions. They recommend the use of the interpolation approach of Crisfield and Jelenić. Finally, Ibrahimbegović and Taylor [26] also proposed interpolation techniques that satisfy the objectivity criterion for geometrically exact structural models. Update formulæ are based on an incremental approach and rely on the representation of finite rotations based on quaternion quantities, which must be stored at each node of the model. Special attention was paid to the implementation details for applied support rotations and the corresponding modifications of the residual vector and tangent matrix introduced by the follower forces and moments.

Because of the many difficulties associated with the treatment of finite rotations, "rotation-less formulations" have appeared in recent years. For instance, in the absolute nodal coordinate formulation [27], absolute displacements and global slopes are used as nodal coordinates, bypassing the need for finite rotations. Betsch and Steinmann [28] have advocated the use of the direction cosine matrix to represent finite rotations. It should be noted, however, that these rotation-less formulations use more coordinates than the minimum set required to represent finite rotations, and hence, typically require more computational resources than their counterparts based on minimum set representations.

Clearly, the properties of finite rotations are key to their manipulation and interpolation. A geometric interpretation of finite rotation was provided by Euler's theorem on finite rotations [3], which states: *"any rigid motion of a body leaving one of its point fixed may be represented by a rotation about a suitable axis passing through that point."* This implies that any rotation can be described by a single rotation of magnitude ϕ about a unit vector $\bar{n}$, *i.e.* three parameters, forming a "minimal set." Euler himself introduced the Euler angles [29], a widely used parameterizations of finite rotations. Many other parameterizations were proposed later, and comprehensive reviews of the topic can be found in [30, 31, 32]. Stuelpnagel [33] provided a concise analysis of different parameterizations of finite rotations. He showed that the six parameter representation consisting of the first two columns of the rotation tensor yields a set of linear differential equations for the motion of a rigid body. Furthermore, he proved that a minimum of five parameters is required to obtain a bijective mapping of the rotation group. This parametrization yields a set of nonlinear equations of motion for a rigid body and is not recommended for practical applications. Stuelpnagel showed that four parameter representations, such as the quaternion representation [34, 35, 36], are singularity free. Finally, he proved that minimal set parameterizations always involve a singularity. Bauchau and Trainelli [37], however, have shown that a simple rescaling operation enables the use of a minimal set representation of finite rotation, while avoiding all singularities. This option is available for the vectorial parameterization of finite rotations that encompasses a number popular representations such as the rotation vector, Rodrigues parameters [5] or the Wiener-Milenković parameters [38, 39], among others.

In the present work, the problem of interpolation of finite rotations within the framework of geometrically exact structural elements is revisited. For computational efficiency, it is desirable to use a minimal set representation of finite rotations, *i.e.* three parameters only. While quaternions have been used in multibody dynamics simulations [40, 41], the computational costs of dealing with four parameters and the enforcement of the associated normality condition have limited their use. A rescaling operation [37] is systematically used to eliminate singularities associated with such minimal set representations. The rescaling operation is based on the observation that addition of a rotation of magnitude $\phi = \pm 2\pi$ to a finite rotation leaves the associated rotation tensor unchanged. While the concept of objectivity is based on the invariance of the strain field with respect to the addition of a rigid body motion to the rotation field, the concept of rescaling is based on the invariance of the rotation tensor with respect to the addition of a rotation of magnitude $\phi = \pm 2\pi$, *i.e.* $R(\phi, \bar{n}) = R(\phi \pm 2\pi, \bar{n})$. In turn, this

raises the question of invariance on the interpolation of finite rotation with respect to rescaling. It is shown that the basic interpolation algorithm proposed by Crisfield and Jelenić [19] to achieve objectivity, is also invariant with respect to rescaling operations.

This paper is structured in the following manner. Section 2 summarizes the salient properties of finite rotations that are relevant to the present investigation, and geometrically exact beam formulations are reviewed in section 3. Finite rotation interpolation techniques for finite element implementations are described in section 4, with special attention devoted to the impact of rescaling operations. Rescaling also impacts the choice of unknowns, as discussed in section 5, and a new algorithm is proposed for the interpolation of incremental quantities. Finally, numerical examples are discussed that demonstrated the simplicity and efficiency of the proposed approach when applied to complex, flexible multibody systems.

2 Parameterization of finite rotations

The kinematic description of beam and shell models based on Cosserat curve and surface approaches, respectively, is formulated in terms of two fields, a displacement field and a rotation field. Whereas the displacement field forms a linear space, the finite rotation field does not, creating challenges in its parameterization and its finite element interpolation. Fundamental facts about finite rotations are reviewed in this section. Finite rotations in three-dimensional space form the set of second-order orthogonal tensors with positive determinant which constitute the special orthogonal group $SO(3)$, *i.e.*

$$SO(3) = \left\{ R | R^{-1} = R^T \wedge \det(R) = +1 \right\}. \tag{1}$$

A more geometric interpretation of finite rotation is provided by Euler's theorem on finite rotations which implies that every rotation can described by a single rotation of magnitude ϕ about a unit vector $\bar{n}$. Simple geometric arguments [2] then yield the following expression for the rotation tensor, known as Euler's formula,

$$R = I + \sin\phi\ \tilde{n} + (1 - \cos\phi)\ \tilde{n}\tilde{n}, \tag{2}$$

where $\tilde{n}$ denotes the skew-symmetric tensor with an axial vector $\bar{n}$. More recently, the vectorial parameterization for finite rotations has been introduced [37] whereby rotations are described using three parameters, $\underline{p}$, which are defined as

$$\underline{p} = p(\phi)\ \bar{n}, \tag{3}$$

where the generating function, $p(\phi)$, is an odd function of ϕ such that $\lim_{\phi\to 0} p(\phi) = \phi$. The main advantage of this representation is that many commonly used parameterizations of finite rotation correspond to various choices of the generating function. For instance, the rotation vector, also known as the exponential map, corresponds to $p(\phi) = \phi$, Rodrigues parameters [5] to $p(\phi) = 2\tan(\phi/2)$, and the Wiener-Milenković parameters [38, 39] to $p(\phi) = 4\tan(\phi/4)$. As expected from the work of Stuelpnagel [33] all these parameterizations present singularities, as discussed by Bauchau and Trainelli [37]. The explicit expression of the rotation tensor in terms of the vectorial parameterization is

$$R(\underline{p}) = I + R_1(\phi)\ \tilde{p} + R_2(\phi)\ \tilde{p}\tilde{p}, \tag{4}$$

where $R_1 = \nu\cos\phi/2$ and $R_2 = \nu^2/2$ are even functions of ϕ, and $\nu = (2\sin\phi/2)/p$, $\varepsilon = (2\tan\phi/2)/p$. Another important operation in the manipulation of finite rotations is the computation of the angular velocity vector, $\underline{\omega}$, as $\underline{\omega} = H(\underline{p})\dot{\underline{p}}$, where $\dot{\underline{p}}$ indicates the time derivative of the vectorial parameters. Operator $H(\underline{p})$ is given by

$$H(\underline{p}) = 1/(\mathrm{d}p/\mathrm{d}\phi) + H_1(\phi)\ \tilde{p} + H_2(\phi)\ \tilde{p}\,\tilde{p}, \tag{5}$$

where $H_1 = R_2$ and $H_2 = \left(1/(\mathrm{d}p/\mathrm{d}\phi) - R_1(\phi)\right)/p^2$ are even functions of ϕ. Of course, similar relationships can be used to compute the curvature vector, $\underline{\kappa}$, as $\underline{\kappa} = H(\underline{p})\underline{p}'$, where $\underline{p}'$ indicates the spatial derivative of the vectorial parameters.

For specific choices of the generating function, the vectorial parameterization exhibits desirable features: 1) for the Rodrigues and Wiener-Milenković parameters, all expressions involved in the manipulations of finite rotations become purely algebraic, improving computational efficiency, and 2) the rotation vector and Wiener-Milenković parameters are singularity free representations for $|\phi| < 2\pi$, enabling the representation of all orientations. Such parameterizations, however, are not necessarily "worry free." Indeed, finite rotations are often used in incremental procedures where an incremental rotation is added to a finite rotation at each time step, for instance. In this case, angles of arbitrary magnitude, *i.e.* $|\phi| > 2\pi$, are routinely encountered; consider, for instance, a rotating shaft, or a satellite tumbling in space. In these cases, singularities will always appear as $|\phi|$ increases.

The range of validity of the Wiener-Milenković parameterization can be *extended* by using a *rescaling operation*. This operation is based on the observation that rotations of magnitudes ϕ and $\phi^\dagger = \phi \pm 2\pi$ about the same axis $\bar{n}$ correspond to the same final orientation. The norm of the Wiener-Milenković parameters is $\|\underline{p}\| = p \leq 4$ when $|\phi| \leq \pi$. Let $\underline{p}$ and $\underline{p}^\dagger$ be associated with the rotations ϕ and $\phi^\dagger$, respectively. The relationship between these two sets of parameters is

$$\underline{p}^\dagger = 4\bar{n}\tan\frac{\phi^\dagger}{4} = 4\bar{n}\tan(\frac{\phi}{4} \pm \frac{\pi}{2}) = -\frac{\nu\underline{p}}{1-\nu}. \tag{6}$$

It is then readily shown that $pp^\dagger = 16$. If $\pi < |\phi| < 2\pi$, $p > 4$, and hence $p^\dagger < 4$; in other words, the rescaling operation decreases the norm of the vector parameterization.

Another fundamental operation is the *composition of finite rotations*, which was first addressed by Rodrigues [5] in terms of Rodrigues parameters, but is readily generalized in terms of the vectorial parameterization. Let $\underline{p}$, $\underline{q}$, and $\underline{r}$ with rotation angles ϕ_p, ϕ_q, and ϕ_r, respectively, be the vectorial parameterization of three finite rotations such that $R(\underline{r}) = R(\underline{p})R(\underline{q})$: it is said that rotation $\underline{r}$ is the composition of rotations $\underline{p}$ and $\underline{q}$. The formulæ for composition of the vectorial parameterization [37] are

$$\cos\phi_r/2 = \nu_p\nu_q\left(1/\varepsilon_p\varepsilon_q - \underline{p}^T\underline{q}/4\right), \tag{7a}$$

$$\nu_r\underline{r} = \nu_p\nu_q\left(\underline{p}/\varepsilon_q + \underline{q}/\varepsilon_p + \tilde{p}\underline{q}/2\right). \tag{7b}$$

The first equation is used to compute ϕ_r and hence, ν_r. The second equation then yields $\underline{r}$. When dealing with Wiener-Milenković parameters, the composition formulæ simply reduce to

$$\underline{r} = 4\left(q_0\underline{p} + p_0\underline{q} + \tilde{p}\underline{q}\right)/(\Delta_1 + \Delta_2), \tag{8}$$

where $p_0 = 2 - \underline{p}^T\underline{p}/8$, $q_0 = 2 - \underline{q}^T\underline{q}/8$, $\Delta_1 = (4-p_0)(4-q_0)$, and $\Delta_2 = p_0q_0 - \underline{p}^T\underline{q}$.

Consider now the practical case of a dynamic simulation that proceeds in small time step increments. At each time step, let the rotations at a point of the system be denoted $\underline{p}_i$ and $\underline{p}_f$ at the beginning and end of the time step, respectively, whereas the incremental rotation is denoted $\underline{p}$. If all rotations are measured in the inertial system, $R(\underline{p}_f) = R(\underline{p})R(\underline{p}_i)$, and hence, the composition formulæ, eqs. (7), must be applied to find the final rotation knowing the initial configuration and the rotation increment. As the simulation proceeds, the norm, p_f, of the rotation parameters is likely to increase, and at some point $p_f > 4$; a rescaling operation, see eq. (6), then becomes necessary to avoid singularities. The two operations, composition and rescaling, are conveniently combined into a single operation, which, for Wiener-Milenković parameters, takes on a particularly simple form

$$\underline{r} = \begin{cases} 4\left(q_0\underline{p} + p_0\underline{q} + \tilde{p}\underline{q}\right)/(\Delta_1 + \Delta_2) & \text{if } \Delta_2 \geq 0, \\ -4\left(q_0\underline{p} + p_0\underline{q} + \tilde{p}\underline{q}\right)/(\Delta_1 - \Delta_2) & \text{if } \Delta_2 < 0. \end{cases} \tag{9}$$

It is interesting to note that the rescaling condition automatically selects the largest denominator, also guaranteeing the most accurate numerical evaluation of the composed rotation.

Although the combination of the composition and rescaling operations is theoretically possible for all vectorial parameterizations, it takes a simple, purely algebraic form for the Wiener-Milenković parameters.

In the remainder of this paper, the composition of finite rotations with optional rescaling will be indicated by the following notation

$$R(\underline{r}) = R(\underline{p})R(\underline{q}) \Longleftrightarrow \underline{r} = \underline{p} \oplus \underline{q}, \tag{10}$$

which implies that $\underline{r}$ is computed with the help of eq. (9) for the Wiener-Milenković parameterization. Note that composition operations such as $R(\underline{r}) = R^T(\underline{p})R(\underline{q})$ are also commonly encountered. In view of eq. (4), it is readily observed that $R^T(\underline{p}) = R(-\underline{p})$ and hence, the following notion is used

$$R(\underline{r}) = R^T(\underline{p})R(\underline{q}) \Longleftrightarrow \underline{r} = \underline{p}^- \oplus \underline{q}, \tag{11}$$

where the notation $\underline{p}^-$ indicates that the signs of the rotation parameters should be changed before using eq. (9). Note the simplicity of eq. (9) as compare to the direct application of the composition equation, $R(\underline{r}) = R(\underline{p})R(\underline{q})$, that implies a four step procedure for the evaluation of $\underline{r}$ knowing $\underline{p}$ and $\underline{q}$: 1) evaluate $R(\underline{p})$, 2) evaluate $R(\underline{q})$, 3) evaluate the matrix product $R(\underline{r}) = R(\underline{p})R(\underline{q})$, 4) extract the parameters $\underline{r}$ from $R(\underline{r})$ using specialized algorithms, such as those described by Klumpp [42] and Shepperd [43].

3 Geometrically exact beams

Consider a beam of length L with a cross-section Ω of arbitrary shape, as depicted in fig. 1. An inertial frame of reference, $\mathcal{I} = (\bar{\imath}_1, \bar{\imath}_2, \bar{\imath}_3)$, is used. Let $\underline{x}_0(\alpha_1)$ be the position vector of a point on the reference line of the beam and α_1 a curvilinear coordinate that measures length along this reference line. At any point along the reference line, a local frame is defined, $\mathcal{B}^0 = (\bar{e}_1, \bar{e}_2, \bar{e}_3)$; $\bar{e}_1$ is the unit vector tangent to the reference line, whereas unit vectors $\bar{e}_2$ and $\bar{e}_3$ define the plane of the beam's cross-section and α_2 and α_3 are material coordinates along those axes. The position vector of a material point of the beam now writes

$$\underline{x}(\alpha_1, \alpha_2, \alpha_3) = \underline{x}_0(\alpha_1) + \underline{s}(\alpha_1, \alpha_2, \alpha_3), \tag{12}$$

where $\underline{s}(\alpha_1, \alpha_2, \alpha_3) = \alpha_2\ \bar{e}_2(\alpha_1) + \alpha_3\ \bar{e}_3(\alpha_1)$.

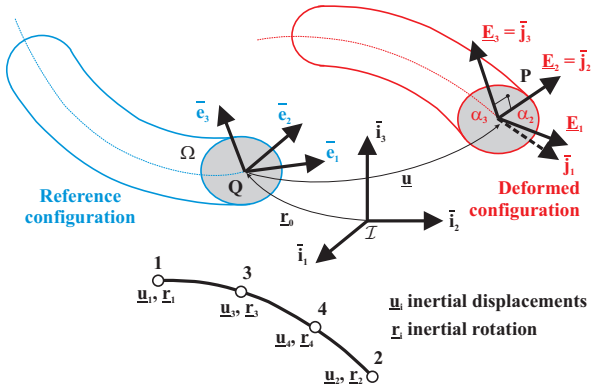

Figure 1: Curved beam in the reference and deformed configurations.

Coordinates α_1, α_2 and α_3 form a natural choice to represent the beam's configuration. Figure 1 also shows the deformed configuration of the beam. Two fundamental assumptions are made concerning the deformation of the beam: the cross-section does not deform in its own plane and the cross-section remains plane after deformation. Note that the plane of the cross-section is not assumed to remain normal to the reference line of beam, allowing for transverse

shearing deformations. These assumptions imply that each cross-section displaces and rotates like a rigid body. Consequently, material lines $\underline{E}_2$ and $\underline{E}_3$ remain mutually orthogonal unit vectors, and the following basis in constructed $\mathcal{B} = (\bar{\jmath}_1 = \tilde{\jmath}_2\bar{\jmath}_3, \bar{\jmath}_2 = \underline{E}_2, \bar{\jmath}_3 = \underline{E}_3)$. Note that material line $\underline{E}_1$ is not a unit vector, nor is it orthogonal to $\underline{E}_2$ and $\underline{E}_3$, as axial and transverse shearing strains develop during deformation. Let R_0 and $R(\alpha_1)$ be the rotation tensors that bring basis $\mathcal{I}$ to $\mathcal{B}^0$, and $\mathcal{B}^0$ to basis $\mathcal{B}$, respectively, both measured in basis $\mathcal{I}$. It then follows that the position vector of a material point of the deformed beam can written as

$$\underline{X}(\alpha_1, \alpha_2, \alpha_3) = \underline{x}_0(\alpha_1) + \underline{u}(\alpha_1) + R(\alpha_1)R_0\ \underline{s}(\alpha_2, \alpha_3), \tag{13}$$

where $\underline{u}(\alpha_1)$ is the reference line displacement vector.

Now that the kinematics of the problem are defined, the beam's sectional deformations, measured in basis $\mathcal{I}$, are given as

$$\underline{\Gamma} = (\underline{x}_0' + \underline{u}') - (RR_0)\bar{\imath}_1; \quad \tilde{\kappa} = R'R^T, \tag{14}$$

where the notation $(\cdot)'$ is used to denote a derivative with respect to α_1. The strain components measured in the convected material basis, $\mathcal{B}$, are denoted $\underline{\Gamma}^* = (RR_0)^T\underline{\Gamma}$ and consist of the sectional axial and shear strains. The curvature components measured in the convected material basis are denoted $\underline{\kappa}^* = (RR_0)^T\underline{\kappa}$ and consist of the sectional twisting and bending curvatures; $\underline{\kappa}$ is the axial vector of $\tilde{\kappa}$. The superscript $(\cdot)^*$ will be used here to indicate the components of vectors and tensors measured in the convected material frame, $\mathcal{B}$.

The principle of virtual work can be used to obtain the governing equations of the problem, which take following form

$$\dot{\underline{h}} - \underline{N}' = \underline{f}; \tag{15a}$$

$$\dot{\underline{g}} + \tilde{\dot{u}}\underline{h} - \underline{M}' - (\tilde{x}_0' + \tilde{u}')\underline{N} = \underline{m}, \tag{15b}$$

where $\underline{N}$ and $\underline{M}$ are the sectional forces and moments, respectively, $\underline{h}$ and $\underline{g}$ the sectional linear and angular momenta, respectively, and $\underline{f}$ and $\underline{m}$ the externally applied forces and moments per unit span of the beam, respectively, all measured in basis $\mathcal{I}$. The sectional constitutive law are

$$\begin{bmatrix} \underline{N}^* \\ \underline{M}^* \end{bmatrix} = \mathcal{C}^* \begin{bmatrix} \underline{\Gamma}^* \\ \underline{\kappa}^* \end{bmatrix}, \tag{16}$$

where $\mathcal{C}^*$ is the beam's 6×6 sectional stiffness matrix, and $\underline{N}^* = (RR_0)^T\underline{N}$ and $\underline{M}^* = (RR_0)^T\underline{M}$ the sectional forces and moment, respectively. Finally, the sectional linear and angular momenta are $\underline{h}^* = (RR_0)^T\underline{h}$ and $\underline{g}^* = (RR_0)^T\underline{g}$, respectively, and defined as

$$\begin{bmatrix} \underline{h}^* \\ \underline{g}^* \end{bmatrix} = \mathcal{M}^* \begin{bmatrix} (RR_0)^T\dot{\underline{u}} \\ \underline{\omega}^* \end{bmatrix}, \tag{17}$$

where $(RR_0)^T\dot{\underline{u}}$ and $\underline{\omega}^*$ denote the sectional linear and angular velocities measured in the convected material frame, and the sectional sectional mass matrix $\mathcal{M}^*$ is given by

$$\mathcal{M}^* = \begin{bmatrix} m & m\tilde{\eta}^{*T} \\ m\tilde{\eta}^* & \varrho^* \end{bmatrix} \tag{18}$$

where m is the mass per unit length, $\underline{\eta}^*$ the location of the cross-sectional center of mass, and ϱ^* the cross-sectional inertia tensor.

By definition, a rigid body motion is a motion that generates no strains. This implies that the following rigid body motion $\underline{u}(\alpha_1) = \underline{u}^R + (R^R - I)\underline{x}_0(\alpha_1)$, $R(\alpha_1) = R^R$, consisting of a translation, $\underline{u}^R$, and a rotation about the origin characterized by a rotation matrix, R^R, should generate no straining of the beam. It can be readily verified with the help of eqs. (14) that such rigid body motion results in $\underline{\Gamma} = 0$ and $\underline{\kappa} = 0$, as expected.

4 Finite element discretization

The interpolation of the displacement field within an element is at the heart of the finite element discretization procedure. Consider a simple, one dimensional beam element featuring N nodes, the displacement field and its spatial derivative are typically interpolated as

$$\underline{\hat{u}}(s) = h^k \underline{u}^k, \quad \text{and} \quad \underline{\hat{u}}'(s) = h^{k\prime} \underline{u}^k, \tag{19}$$

respectively, where $h^k(s)$, $k = 1, \dots N$ are the shape functions for the element that can be found in any finite element textbook, such as Bathe [44], for instance, $\underline{u}^k$ the component of the displacement vectors at the N nodes of the element, $\hat{(\cdot)}$ denotes interpolated quantities, and repeated superscripts, $(\cdot)^k$, imply a summation over the N nodes of the element. The local variable $s \in [-1, +1]$ measures non-dimensional length along the element. The notation $(\cdot)'$ indicates a derivative with respect to α_1, and $h^{k\prime}$ is computed with the help of the chain rule for derivatives as $h^{k\prime} = J^{-1} \mathrm{d}h^k/\mathrm{d}s$, where $J = \mathrm{d}\alpha_1/\mathrm{d}s$ is the determinant of the Jacobian of the variable transformation from α_1 to s. It is important to realize that interpolation is a linear operation, acting here on the displacement field which forms a linear space. Let $\underline{u}_i^k$, $\underline{u}^k$ and $\underline{u}_f^k$ be the nodal displacements at the beginning of a time step, the incremental nodal displacements, and the displacements at the end of a time step, respectively. Furthermore, let the displacement update at the nodes be written as $\underline{u}_f^k = \underline{u}_i^k + \underline{u}^k$, $k = 1, \dots N$. It then follows that

$$\underline{\hat{u}}_i(s) + \underline{\hat{u}}(s) = h^k(\underline{u}_i^k + \underline{u}^k) = h^k \underline{u}_f^k = \underline{\hat{u}}_f(s). \tag{20}$$

This important relationship implies that initial, final, and incremental fields can all three be interpolated with the same shape functions, and a simple update of the nodal values then guarantees compatibility of the interpolated displacement fields for all values of s.

When formulating beam and shell elements, the kinematic description of the problem also requires an interpolation of the rotation field and its derivative, written as

$$\underline{\hat{c}}(s) = h^k \underline{c}^k, \quad \text{and} \quad \underline{\hat{c}}'(s) = h^{k\prime} \underline{c}^k, \tag{21}$$

respectively, where $\underline{c}^k$ are the rotation parameters at the N nodes of the element. This interpolation simply provides an approximation to the rotation field within the element. Figure 2 shows the interpolated rotation field for a four-noded beam element using cubic interpolation polynomials. The rotations at the four nodes are defined by four rotation angles, $\phi_1 = 145°$, $\phi_2 = 160°$, $\phi_3 = 170°$, $\phi_4 = 181°$, and associated unit vectors

$$[\underline{n}_1, \underline{n}_2, \underline{n}_3, \underline{n}_4] = \begin{bmatrix} 0.50 & 0.52 & 0.55 & 0.60 \\ 1.00 & 1.03 & 1.05 & 1.06 \\ 1.20 & 1.10 & 1.25 & 1.50 \end{bmatrix}. \tag{22}$$

The interpolated rotation field was computed using eq. (21), and the first Wiener-Milenković parameter, $\hat{c}_1$, of the finite rotation is shown in fig. 2; similar results are obtained for the other parameters, c_2 and c_3. The curvature can be computed in a similar manner as $\underline{\hat{\kappa}}(s) = H\, \underline{\hat{c}}'(s)$, where operator $H(\underline{\hat{c}})$ is defined in eq. (5) and $\underline{\hat{c}}(s)$ and $\underline{\hat{c}}'(s)$ by eq. (21). Figure 3 shows the first component, $\hat{\kappa}_1$, of the curvature vector.

Although the interpolation procedure of eq. (21) looks reasonable considering the results shown in fig. 2, it suffers several serious drawbacks. First, let $\underline{c}_i^k$, $\underline{c}^k$ and $\underline{c}_f^k$ be the nodal rotations at the beginning of a time step, the incremental nodal rotations, and the rotations at the end of a time step, respectively. Proceeding as was done above for the displacement field implies that $\underline{\hat{c}}_f(s) = \underline{\hat{c}}_i(s) + \underline{\hat{c}}(s)$ if the nodal updates are selected as $\underline{c}_f^k = \underline{c}_i^k + \underline{c}^k$, $k = 1, \dots N$. Unfortunately, these relationships are not correct for finite rotations, which require $\underline{\hat{c}}_f(s) = \underline{\hat{c}}(s) \oplus \underline{\hat{c}}_i(s)$ and $\underline{c}_f^k = \underline{c}^k \oplus \underline{c}_i^k$, $k = 1, \dots N$. The nonlinear character of composition operations implies that $\underline{\hat{c}}_f(s) \neq \underline{\hat{c}}(s) \oplus \underline{\hat{c}}_i(s)$ if $\underline{c}_f^k = \underline{c}^k \oplus \underline{c}_i^k$. In other words, if the nodal rotations are updated using composition of finite rotations, the compatibility of the interpolated rotation fields cannot be guaranteed for all values of s.

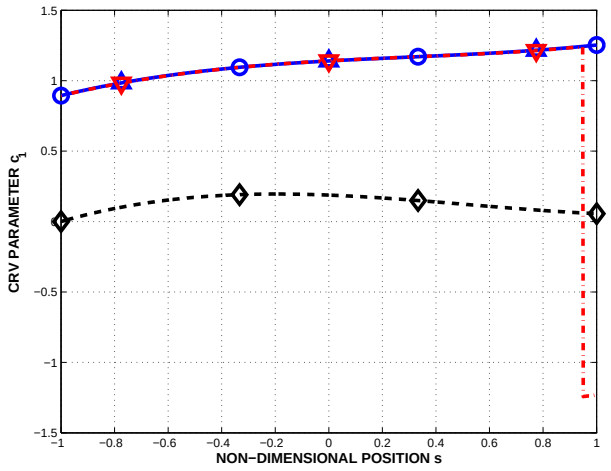

Figure 2: Wiener-Milenković parameter, $\hat{c}_1$, for the given rotation field; no rescaling is used at node 4. Nodal rotations: ($\circ$). Interpolation using eq. (21): solid line, corresponding Gauss point values: ($\triangle$). Relative nodal rotations: ($\diamondsuit$). Interpolation of relative rotations: dashed line. Interpolation computed by algorithm 1: dashed-dotted line, corresponding Gauss point values: ($\bigtriangledown$).

The second drawback becomes obvious once rescaling of finite rotations is taken into account. The finite rotation at the fourth node of the element is of magnitude $\phi_4 = 181^\circ > 180^\circ$, and hence, should be rescaled to avoid singularities. The Wiener-Milenković parameters of this node are $\underline{c}^4 = \lfloor 1.253, 2.214, 3.132 \rfloor$, $\|\underline{c}^4\| = 4.035 > 4$, whereas its rescaled parameters are $\underline{c}^{4\dagger} = \lfloor -1.231, -2.175, -3.078 \rfloor$, $\|\underline{c}^{4\dagger}\| = 3.965 < 4$, as expected. Figure 4 shows the rotation field interpolated using eq. (21) in the presence of rescaling. Note that the results presented in this figure should be identical to those shown in fig. 2 because they correspond to the interpolation of identical configurations: indeed, the rotation tensor at node 4 is uniquely defined, but represented by different rotation parameters, $\underline{c}^4$ and $\underline{c}^{4\dagger}$ due to rescaling. Clearly, the linear interpolation operation of eq. 21 is not invariant under the rescaling operation. The curvature field is shown in fig. 5 and clearly, in the presence of rescaling, the results are erroneous: without rescaling, the three Gauss point values of the first curvature component are κ_1 = -0.048, 0.230, and 0.322, respectively, as compared to κ_1 = -0.208, 0.308, and -5.0521, respectively, in the presence of rescaling.

Clearly, a more robust interpolation approach is necessary to deal with finite rotations in the presence of rescaling; the following algorithm was proposed by Crisfield and Jelenić [19].

Algorithm 1 (Finite rotation interpolation) *Interpolation of a finite rotation field defined by its rotation parameters, $\underline{c}^k$, at the N nodes of a finite element.*

Step 1. *Compute the nodal relative rotations, $\underline{r}^k$: remove the rigid body rotation, $\underline{c}^1$, from the finite rotation at each node, $\underline{r}^k = \underline{c}^{1-} \oplus \underline{c}^k$.*

Step 2. *Interpolate the relative rotation field: $\hat{\underline{r}}(s) = h^k \underline{r}^k$ and $\hat{\underline{r}}'(s) = h^{k\prime} \underline{r}^k$. Find the curvature field $\hat{\underline{\kappa}} = R(\underline{c}^1) H(\hat{\underline{r}})\, \hat{\underline{r}}'$.*

Step 3. *Restore the rigid body rotation removed in step 1: $\hat{\underline{c}}(s) = \underline{c}^1 \oplus \hat{\underline{r}}(s)$.*

Algorithm 1 removes the possible effects of rescaling from the interpolation procedure. In step 1, the relative rotations of the nodes with respect to node 1 are computed using the composition formula; note that the relative rotation field could be computed with respect to any of the nodes of the element, node 1 is simply a convenient choice. It is assumed here that the relative rotations within one single element are small enough that no rescaling is needed within the element, *i.e.* within the element, $|\phi_r^k| < \pi$. If this condition were not to be satisfied,

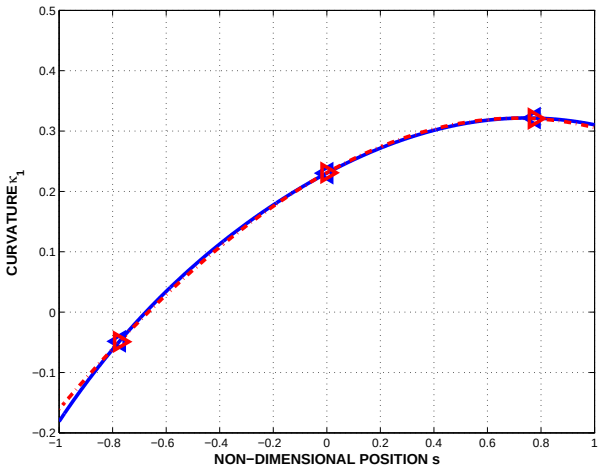

Figure 3: First component of the curvature vector, $\hat{\kappa}_1$, based on interpolation using eq. (21): solid line, corresponding Gauss point values: ($\triangleleft$). Curvatures computed by algorithm 1: dashed-dotted line, corresponding Gauss point values: ($\triangleright$).

a finer mesh would be required to limit the relative rotation within each element. Next, these relative rotations are interpolated using standard procedures. Finally, the interpolated relative rotation is composed with the rotation of node 1 to find the interpolated rotation field. Interpolations of nodal rotations computed by algorithm 1 were added to figs. 2 and 4. Since the nodal rotations presented in these figures only differ by the rescaling of node 4, the relative rotation fields are identical, the corresponding curvature fields are identical, as are the interpolated rotation fields. It should be note here that this interpolated rotation field seems to present a discontinuity at $s = 0.973$ in both figures: this is due to the rescaling operation in step 3 of algorithm 1, but does not affect the quality of the interpolation. In fact, the interpolation procedure of algorithm 1 is able to deal with the discontinuities inherent to the required rescaling operations. However, the presence of these discontinuities has implications for the linearization of the equations of motion as discussed in the next section of this paper.

The third drawback of interpolation based on eq. (21) is its lack of objectivity when computing strain components. As shown in section 3, the strain measures of geometrically exact beam theory are invariant with respect to the addition of a rigid body motion. The term "objectivity" was coined by Crisfield and Jelenić [19] and refers to the invariance of strain measures computed through interpolation to the addition of a rigid body motion. Since algorithm 1 is based on the interpolation of relative rotation, the addition of a rigid body motion is automatically filtered out from the interpolation step, ensuring the objectivity of the process. Jelenić and Crisfield [20] studied the lack of objectivity of interpolation schemes based on eq. (21) and concluded that "The non-invariance and path-dependence in these formulations decrease with both p-refinement and h-refinement and in practical applications cannot always be easily spotted." These conclusions are supported by the data presented here: in fig. 3, the curvatures computed based on eq. (21) (non-objective) are nearly identical to those computed with algorithm 1 (objective). In fact, at the Gauss points, the curvature components, κ_1, computed by the two approaches only differ by 0.16, -0.085 and 0.16%, respectively. These discrepancies are minute compared to the gross disparities observed in fig. 5 in the presence of rescaling. Clearly, the use of algorithm 1 provides objectivity of the strain measure, often only a small improvement in the quality of the interpolation, but is indispensable when dealing with rotation fields involving potential rescaling.

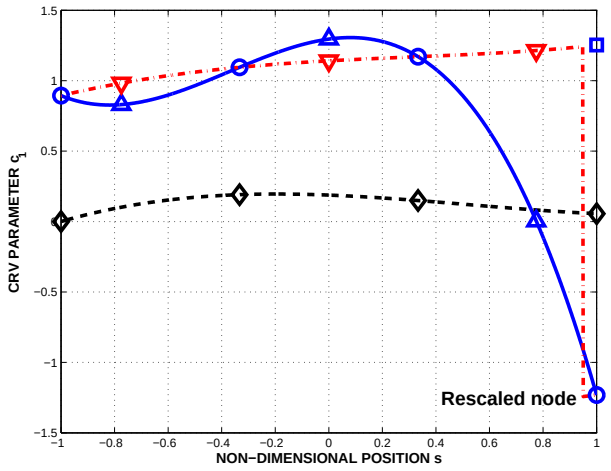

Figure 4: Wiener-Milenković parameter, $\hat{c}_1$, for the given rotation field; node 4 has been rescaled; for reference, the unscaled note 4 is indicated by ($\square$). Nodal rotations: ($\circ$). Interpolation using eq. (21): solid line, corresponding Gauss point values: ($\triangle$). Relative nodal rotations: ($\diamondsuit$). Interpolation of relative rotations: dashed line. Interpolation computed by algorithm 1: dashed-dotted line, corresponding Gauss point values: ($\triangledown$).

5 Total and incremental unknowns

Multibody simulations typically proceed in discrete time steps. Figure 6 shows the inertial frame of reference, the reference, *i.e.* unstressed, configuration of the beam at time $t = 0$, and its configurations at the beginning and end times of a typical time step, denoted t_i and t_f, respectively. Each frame is related to its parent frame by a finite motion characterized by a displacement and a finite rotation tensor, all measured in the inertial frame. It is assumed that the dynamic simulation has successfully proceeded up to time t_i, *i.e.* the corresponding displacement and rotation fields, denoted $\underline{u}_i$ and R_i, respectively, are known. Let $\underline{c}_i$ be a parameterization of the finite rotation tensor R_i.

To advance the solution from the initial to the final time of the time step, two sets of unknowns can be selected: the incremental displacements and rotations, denoted $\underline{u}$ and R, respectively, or the total displacements and rotations, denoted $\underline{u}_f$ and R_f, respectively. Let $\underline{c}$, and $\underline{c}_f$ be parameterizations of the finite rotation tensors R and R_f, respectively. From a kinematic viewpoint, both sets of unknowns are entirely equivalent. In typical dynamic simulations, however, small time steps must be selected to achieve convergence and guarantee the accuracy of the solution. Consequently, it can be assumed that incremental rotations will be of magnitude $|\phi| < \pi$; in fact, for most practical cases, $|\phi| \ll \pi$. Note that $|\phi| = \pi$ implies that within one single time step, a component of the system rotates by 180°. It cannot be assumed, however, that $|\phi_f|$, the rotation associated with rotation tensor R_f, is small, in fact, $|\phi_f| > \pi$ is likely to occur. The implication of these observations is clear: if total rotations are used as unknowns, some of the finite rotation parameters, $\underline{c}_f$, will be rescaled, as required, whereas if incremental rotations are used as unknowns, none of the unknown parameters, $\underline{c}$, will be rescaled. The interpolation algorithm developed in section 4 was shown to seamlessly handle rescaling, however, when dealing with dynamic simulations, additional considerations must be taken into account.

Spatial and time discretization algorithms typically transform the governing partial differential equations of complex multibody systems into a set of nonlinear algebraic equations, which are solved in an iterative manner using the Newton-Raphson method. Inherent to this approach is a linearization process that transforms the nonlinear algebraic equations into their linearized counterparts. Consider, for instance, the linearization of the curvature

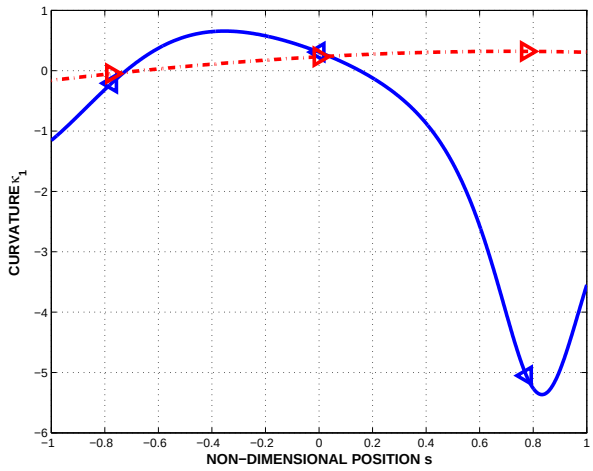

Figure 5: First component of the curvature vector, $\hat{\kappa}_1$, based on interpolation using eq. (21): solid line, corresponding Gauss point values: ($\triangleleft$). Curvatures computed by algorithm 1: dashed-dotted line, corresponding Gauss point values: ($\triangleright$).

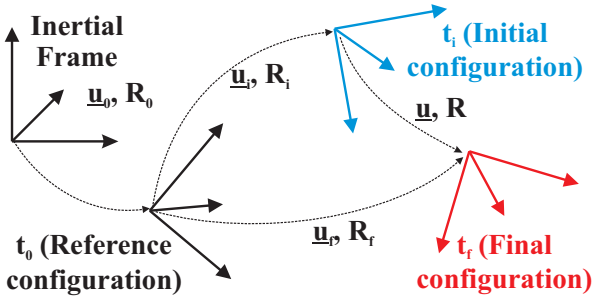

Figure 6: Configuration of the system at various instants in time.

vector, $\underline{\kappa} = H(\underline{c})\underline{c}'$, that will appear in the expression for the elastic forces of a beam element. Application of the linearization procedure leads to $\Delta\underline{\kappa} = H(\underline{c})\Delta\underline{c}' + D(\underline{c},\underline{c}')\Delta\underline{c}$, where $D(\underline{c},\underline{c}') = \partial(H(\underline{c})\underline{c}')/\partial\underline{c}$, and hence, operators $H(\underline{c})$ and $D(\underline{c},\underline{c}')$ will appear in the expression of the tangent stiffness matrix of the element. Let $\underline{c}$ and $\underline{c}^\dagger$ denote the parameters a finite rotation and their rescaled counterparts, respectively, as discussed in section 2. Clearly, $R(\underline{c}) = R(\underline{c}^\dagger)$ by construction of the rescaling operation, whereas it is easily verified that $H(\underline{c}) \neq H(\underline{c}^\dagger)$ and $D(\underline{c},\underline{c}') \neq D(\underline{c}^\dagger,\underline{c}^{\dagger\prime})$. In conclusion, whereas intrinsic quantities such as the rotation tensor, the curvature vector, or elemental elastic forces are invariant to rescaling, and whereas the interpolation operation can be made invariant to the same rescaling through the use of algorithm 1, the tangent stiffness matrix is not invariant to rescaling. The implications of this lack of invariance are easily understood by considering the situation depicted in fig. 4. At the Gauss points, which are used to evaluate the tangent stiffness matrix, no knowledge is available that node 4 was rescaled. The tangent stiffness matrix will be evaluated as if the rescaling of node 4 never took place, *i.e.* the equations are linearized about the wrong point. Hence, the search direction in the Newton-Raphson iteration process will be erroneous, which can ultimately cause failure of simulations.

In view of the above discussion, it is desirable to work with incremental rotations that remain small and do not require rescaling. The tangent stiffness matrix then always corresponds to the correct linearization of the problem. This contrasts with the choice of total rotations as unknowns for which these desirable features cannot be guaranteed. The choice of incremental nodal rotations as unknowns requires interpolation of the incremental rotation field to com-

pute the elemental elastic forces and tangent stiffness matrix. This task cannot be performed with the help of eq. (21): as already pointed out in section 2, the nonlinear nature of the composition operation is incompatible with the linear interpolation operation. An alternative approach is proposed for this operation.

Algorithm 2 (Incremental rotation interpolation) *Interpolation of the incremental rotation field between two configurations defined by nodal rotation parameters, $\underline{c}_i^k$ and $\underline{c}_f^k$, at the N nodes of a finite element.*

Step 1. *Use algorithm 1 to compute the interpolated rotation field, $\underline{c}_i(s)$, based on nodal values $\underline{c}_i^k$.*

Step 2. *Use algorithm 1 to compute the interpolated rotation field, $\underline{c}_f(s)$, based on nodal values $\underline{c}_f^k = \underline{c}^k \oplus \underline{c}_i^k$.*

Step 3. *Compute the incremental rotation field by composition: $\underline{c}(s) = \underline{c}_f(s) \oplus \underline{c}_i^-(s)$.*

Note that this approach is different from that proposed by Cardona and Géradin, who directly interpolated incremental rotations using eq. 21. It is also different from the objective algorithm proposed by Crisfield and Jelenić [19].

6 Numerical examples

A number of numerical examples are presented in this section to illustrate the various concepts discussed in the previous sections.

6.1 Total vs. incremental unknowns

At first, the use of total versus incremental unknowns will be contrasted, to underline the difficulties associated with the use of total rotations in the formulation of dynamic problems. Consider a free-free beam featuring the following physical properties: axial stiffness $S = 9.28$ kN, shearing stiffness $K_{22} = K_{33} = 3.57$ kN, torsional stiffness $J = 65.2$ N·m^2, bending stiffness $I_{22} = I_{33} = 32.6$ N·m^2, and mass per unit length $m = 0.35$ kg/m. The beam is modeled using a single cubic element and is subjected to two mutually orthogonal end bending moments Q_2 and Q_3, both acting in directions normal to the axis of the beam. Both bending moments have a triangular time history: starting from zero value at time $t = 0$, growing linearly to a maximum value of 0.3 N·m at $t = 0.5$ sec, linearly decreasing to a zero value at time $t = 1$ sec, and remaining zero at all subsequent times.

The dynamic response of the beam was computed using time step sizes $h = 1$ and 0.1 msec, with formulations using both total and incremental unknowns. Algorithms 1 and 2 were used to interpolate the total and incremental rotations, respectively. Figure 7 shows the third component of the conformal rotation vector at the beam's end opposite to the applied bending moments, for $h = 1$ msec; the formulations using total and incremental unknowns lead to nearly identical predictions. Figure 7 clearly shows the rescaling operation that occurs at time $t = 0.929$ sec. However, all four nodes of the element are rescaled simultaneously and the finite rotation interpolation procedure performs well with both total and incremental unknowns. Next, the time step size was reduced to $h = 0.1$ msec. In this case, due to the smaller time step size, the node at the free end of the beam is rescaled at time $t = 0.9284$ sec, while the other three nodes are not. As expected, the formulation using total unknowns fails to converge at that time step, whereas that using incremental unknowns converges.

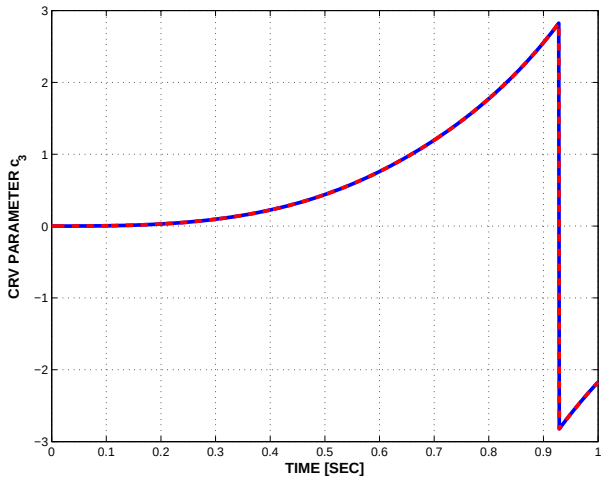

Figure 7: Time histories of third component of the conformal rotation vector at the end node: incremental formulation: solid line; total formulation: dashed line.

6.2 Convergence behavior of the incremental formulation

Next, consider a cantilevered beam rotating about an axis normal to its axis and passing at its root, as depicted in fig. 8. The beam's physical properties are identical to those used in the previous example and it is subjected to a transverse tip load, F, linearly increasing from 0 to 50 N in one second, and rotates at an angular speed, Ω, linearly increasing from 0 to 4 rad/sec in the same time. The system was simulated for 1.5 sec with a time step size $h = 0.01$ sec. In view of the results of the previous example, the investigation solely focuses on the formulation using incremental unknowns. However, the direct interpolation of rotation increments, eq. 21, will now be contrasted with the proposed interpolation approach, algorithm 2. Figures 9 shows the error in the beam's root shear force as a function of the number of linear elements used to mesh the beam. Figure 10 shows the corresponding results for quadratic elements. The reference solution for the error analysis was obtained using a 250 cubic element mesh for which convergence was established.

Figure 8: Rotating cantilevered beam subjected to transverse tip force.

For both linear and quadratic elements, direct rotation interpolation using eq. 21 leads to large errors when coarse meshes are used, but these errors decrease rapidly for both h- and p-refinement. Indeed, the errors observed for the quadratic element mesh are far smaller than those for the linear element mesh. When algorithm 2 is used to interpolate rotation increments, errors are reduced, although this reduction is less pronounced for finer meshes. Since the computational cost associated with the use of algorithm 2 is nearly identical to that of using eq. 21, the use of the former is advisable. Indeed, achieving a 0.01% error in root shear force with quadratic elements requires 5 elements with algorithm 2, but 16 elements for eq. 21; this will result in a nearly threefold gain in computational cost when using the proposed algorithm.

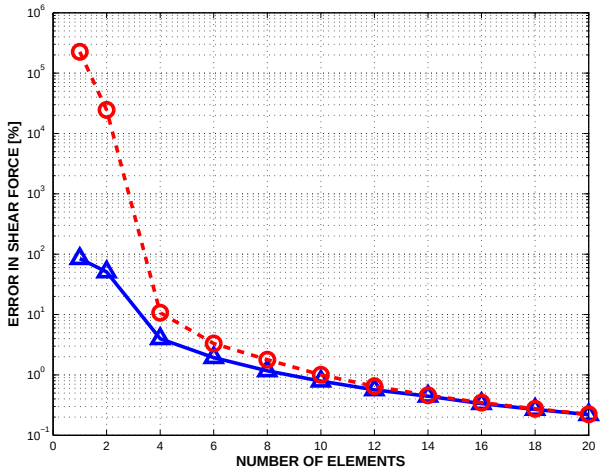

Figure 9: Beam root shear force error versus number of elements for linear element meshes. Interpolation using algorithm 2: solid line; direct interpolation using eq. 21: dashed line.

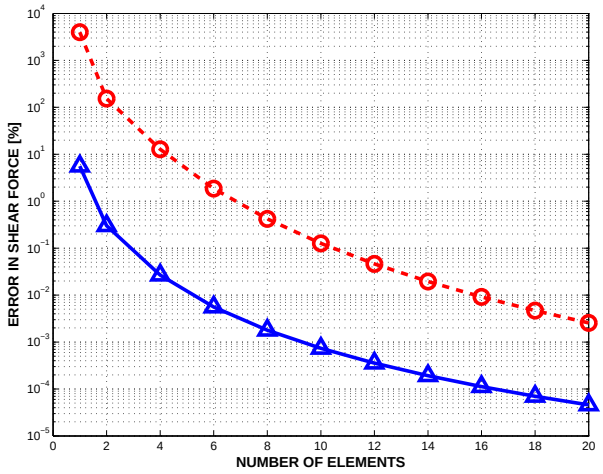

Figure 10: Beam root shear force error versus number of elements for quadratic element meshes. Interpolation using algorithm 2: solid line; direct interpolation using eq. 21: dashed line.

6.3 Rotorcraft tail rotor transmission

This last problem deals with the modeling of the supercritical tail rotor transmission of a helicopter. Fig. 11 shows the configuration of the problem. The aft part of the helicopter is modeled and consists of a 6 m fuselage section that connects at a 45 degree angle to a 1.2 m projected length tail section. This structure supports the transmission to which it is connected at points **M** and **T** by means of 0.25 m support brackets. The transmission is broken into two shafts, each connected to flexible couplings at either end. The flexible couplings are represented by flexible joints, consisting of concentrated springs. Shaft 1 is connected to a revolute joint at point **S**, and gear box 1 at point **G**. Shaft 2 is connected to gear box 1 and gear box 2 which in turn, transmits power to the tail rotor. The plane of the tail rotor is at a 0.3 m offset with respect to the plane defined by the fuselage and tail, and its hub is connected to gear box 2 by means of a short shaft. Each tail rotor blade has a length of 0.8 m and is connected to the rotor hub at point **H** through rigid root-attachments of length 0.2 m. The gear ratios for gear boxes 1 and 2 are 1:1 and 2:1, respectively.

The fuselage has the following physical characteristics: axial stiffness $S = 687$ MN, bending stiffnesses $I_{22} = 19.2$ and $I_{33} = 26.9$ MN·m^2, torsional stiffness $J = 8.77$ MN·m^2, and mass per

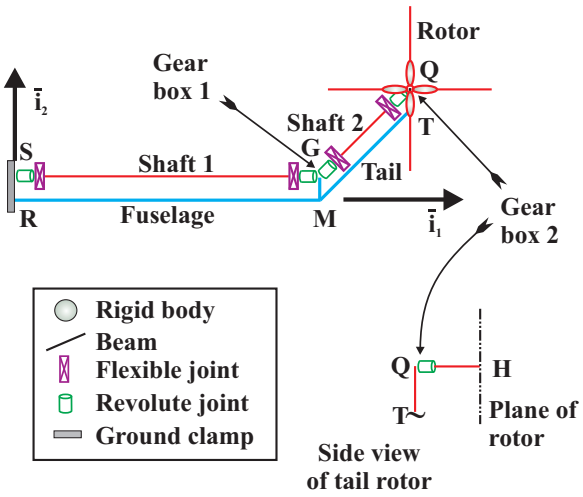

Figure 11: Configuration of a tail rotor transmission.

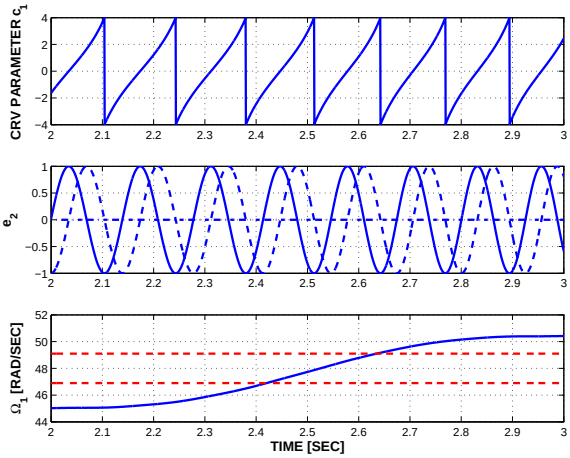

Figure 12: Time histories of a CRV parameter, a unit vector of the rotation tensor(dash-dotted line: $e_{2,1}$, solid line: $e_{2,2}$, dashed line: $e_{2,3}$) and the angular speed of shaft 1 mid-span.

unit span $m = 15.65$ kg/m. The properties of the tail are one third of those of the fuselage. Shafts 1 and 2 have the following physical characteristics: axial stiffness $S = 22.9$ MN, bending stiffnesses $I_{22} = 26.7$ and $I_{33} = 27.7$ kN·m^2, torsional stiffness $J = 22.1$ kN·m^2, and mass per unit span $m = 0.848$ kg/m. The center of mass of the shaft has a 1 mm offset with respect to the shaft reference line. The small difference in bending stiffnesses together with the center of mass offset are meant to represent an initial manufacturing imperfection or an unbalance in the shaft. The stiffness properties of the flexible couplings are as follows: axial stiffness 5.0 kN/m and damping 0.5 N·sec/m, transverse stiffnesses 1.0 MN/m, torsional stiffness 0.1 MN·m/rad, and bending stiffnesses 0.1 kN·m/rad. Finally, gear boxes 1 and 2 have a concentrated mass of 5.0 kg each, and the tail rotor a 15.0 kg mass with a polar moment of inertia of 3.0 kg·m^2.

At first, a static analysis of the system was performed for various constant angular velocities of the drive train. The natural frequencies of the system were computed about each equilibrium configuration. The two lowest natural frequencies of shaft 1 were found to be $\omega_1 = 46.9$ and $\omega_2 = 49.1$ rad/sec. According to linear theory, the system is stable when the shaft angular velocity is below ω_1 or above ω_2, but unstable between theses two speeds.

The system was loaded by a torque acting at the root of shaft 1, featuring the following

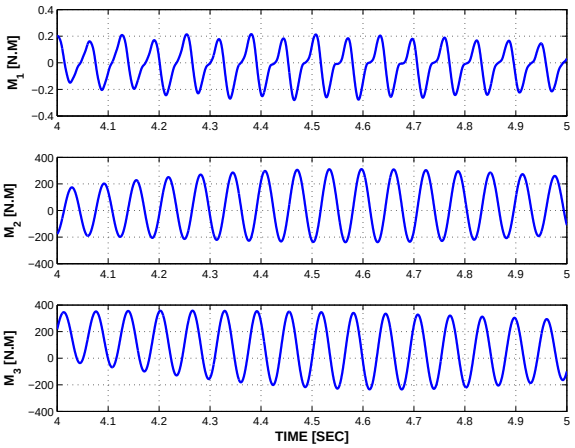

Figure 13: Time histories of the moments of shaft 1 mid-span.

time history

$$Q(t) = \begin{cases} 50\ (1 - \cos 2\pi t) & 0 < t < 1 \text{ sec}, \\ 0 & 1 < t < 2 \text{ sec}, \\ 6\ (1 - \cos 2\pi t) & 2 < t < 3 \text{ sec}, \\ 0 & 3 < t < 6 \text{ sec}. \end{cases}$$

After 1 sec, the angular velocity of shaft 1 stabilizes at about 45 rad/sec, below the critical speed. The torque applied for $2 < t < 3$ sec then accelerates the transmission through the critical zone to reach an angular velocity of 50.5 rad/sec. A constant time step size $h = 0.5$ msec was used for the entire simulation.

Fig. 12 shows the dynamic response at shaft 1's mid-span position for $2 < t < 3$ sec. The top portion of the figure shows the first component of the conformal rotation vector: a rescaling operation occurs for each complete revolution of the shaft. The middle portion of the figure shows the components of the unit vector $\bar{e}_2$, *i.e.* the second column of the rotation tensor. As expected, these quantities are continuous, as they do not "see" the rescaling operations. Finally, the bottom portion of the figure shows the angular velocity of the shaft. The horizontal dashed line indicate the unstable zone for the shaft. Clearly, the shaft goes through this critical zone fast enough to avoid the build up of lateral vibrations. Here again, the angular velocity is continuous, unaffected by the rotation rescaling operations. Figure 13 shows the torque, M_1, and the two bending moments, M_2 and M_3, at shaft 1's mid-span, for $4 < t < 5$ sec. Since the shaft has just passed through the critical zone, fairly large bending moments are observed. Here again all quantities are continuous, despite the multiple rescaling operations. This example clearly demonstrates the ability of the proposed approach to handle finite rotations of arbitrary magnitudes in complex, flexible multibody systems. The rescaling operations are applied at those nodes where they are required to avoid singularities in finite rotation representations. All other quantities, such as the rotation tensor, angular velocities, or bending moments are continuous and unaffected by the rescaling operations.

7 Conclusions

In summary, the following observations can be made. If the finite rotation field is interpolated with eq. (21) without ever rescaling the rotation parameters, the computation will proceed smoothly at first; although the interpolated strain field is not objective, errors remain small, specially if higher order elements are used with a fine mesh. During the simulation, rotation magnitudes will grow; no matter what parameterization is used to represent finite rotations, a

singularity will eventually be reached and the simulation will fail at that point. On the other hand, if the finite rotation field is interpolated with eq. (21) with rescaling of the rotation parameters, the computation will proceed smoothly at first, although the interpolated strain field is not objective. When the first node of the model is rescaled, the strain field computed in the elements connected to this node will be grossly erroneous, see fig. 5, and typically, convergence will not be reached for that time step at which rescaling occurs. Finally, if algorithm 1 is used for the interpolation of the strain field, the simulation is not affected by rescaling of the rotation parameters that takes place whenever required, and the computed strain field is objective. The rescaling operation becomes transparent to the computation process. However, evaluations of the tangent stiffness matrix based on interpolations of total unknowns computed with algorithm 1 can yield erroneous search directions in the Newton-Raphson process used to solve the nonlinear equations, which are inherent to time-stepping procedures. This can destabilize simulations. Therefore, the use incremental unknowns in conjunction with algorithm 2 is recommended. This method preserves the objectivity of geometrically exact formulations, yields tangent stiffness matrices and residual vectors, which are invariant to the rescaling of finite rotations, and, therefore, enables the use of geometrically exact structural models in multibody simulations.

References

[1] L. Euler. Découverte d'un nouveau principe de mécanique. *Mémoires de l'Académie des Sciences de Berlin*, 6(1752):185–217, 1750.

[2] L. Euler. Nova methodus motum corporum rigidorum determinandi. *Novi Commentari Academiae Scientiarum Imperialis Petropolitanae*, 20:208–238, 1775.

[3] L. Euler. Formulae generales pro translatione quacunque corporum rigidorum. *Novi Commentari Academiae Scientiarum Imperialis Petropolitanae*, 20:189–207, 1775.

[4] L. Euler. *Methodus Inveniendi Lineas Curvas Maximi Minimive Proprietate Gaudentes*. Bousquet, Lausanne and Geneva, 1744. Appendix I: De Curvis Elasticis.

[5] O. Rodrigues. Des lois géometriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. *Journal de Mathématiques Pures et Appliquées*, 5:380–440, 1840.

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

[7] E. Reissner. On one-dimensional large-displacement finite-strain beam theory. *Studies in Applied Mathematics*, 52:87–95, 1973.

[8] E. Reissner. On finite deformations of space-curved beams. *Zeitschrift für angewandte Mathematik und Physik*, 32:734–744, 1981.

[9] 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.

[10] 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.

[11] A.A. Shabana and R.A. Wehage. A coordinate reduction technique for dynamic analysis of spatial substructures with large angular rotations. *Journal of Structural Mechanics*, 11(3):401–431, March 1983.

[12] O.P. Agrawal and A.A. Shabana. Application of deformable-body mean axis to flexible multibody system dynamics. *Computer Methods in Applied Mechanics and Engineering*, 56(2):217–245, 1986.

[13] M.A. Crisfield. A consistent co-rotational formulation for non-linear, three-dimensional beam-elements. *Computer Methods in Applied Mechanics and Engineering*, 81:131–150, 1990.

[14] A.A. Shabana. Flexible multibody dynamics: Review of past and recent developments. *Multibody System Dynamics*, 1(2):189–222, June 1997.

[15] T. Belytschko and B.J. Hsieh. Nonlinear transient finite element analysis with convected coordinates. *International Journal for Numerical Methods in Engineering*, 7:255–271, 1973.

[16] A. Cardona and M. Géradin. Time integration of the equations of motion in mechanism analysis. *Computers & Structures*, 33(3):801–820, 1989.

[17] O.A. Bauchau. Computational schemes for flexible, nonlinear multi-body systems. *Multibody System Dynamics*, 2(2):169–225, 1998.

[18] M. Géradin and A. Cardona. *Flexible Multibody System: A Finite Element Approach*. John Wiley & Sons, New York, 2001.

[19] 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.

[20] G. Jelenić and M.A. Crisfield. Geometrically exact 3D beam theory: Implementation of a strain-invariant finite element for static and dynamics. *Computer Methods in Applied Mechanics and Engineering*, 171:141–171, 1999.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] I. Romero. The interpolation of rotations and its application to finite element models of geometrically exact rods. *Computational Mechanics*, 34(2):121–133, 2004.

[26] A. Ibrahimbegović and R.L. Taylor. On the role of frame-invariance in structural mechanics models at finite rotations. *Computer Methods in Applied Mechanics and Engineering*, 191:5159–5176, 2002.

[27] A.A. Shabana. Computer implementation of the absolute nodal coordinate formulation for flexible multibody dynamics. *Nonlinear Dynamics*, 16(3):293–306, 1998.

[28] P. Betsch and P. Steinmann. A DAE approach to flexible multibody dynamics. *Multibody System Dynamics*, 8:367–391, 2002.

[29] L. Euler. De motu corporum circa pumctum fixum mobilium. *Opera Mechanica et Astronomica*, 9(Series Secunda):413–441, 1776. Leonhardi Euleri Opera Omnia.

[30] T.R. Kane. *Dynamics*. Holt, Rinehart and Winston, Inc, New York, 1968.

[31] H. Cheng and K.C. Gupta. A historical note on finite rotations. *Journal of Applied Mechanics*, 56:139–145, 1989.

[32] M.D. Shuster. A survey of attitude representations. *Journal of the Astronautical Sciences*, 41(4):439–517, 1993.

[33] J. Stuelpnagel. On the parameterization of the three-dimensional rotation group. *SIAM Review*, 6(4):422–430, 1964.

[34] W.R. Hamilton. *Elements of Quaternions*. Cambridge University Press, Cambridge, 1899.

[35] A. Cayley. On certain results relating to quaternions. *Philosophical Magazine*, 26:141–145, 1845.

[36] K.W. Spring. Euler parameters and the use of quaternion algebra in the manipulation of finite rotations: A review. *Mechanism and Machine Theory*, 21:365–373, 1986.

[37] O.A. Bauchau and L. Trainelli. The vectorial parameterization of rotation. *Nonlinear Dynamics*, 32(1):71–92, 2003.

[38] T.F. Wiener. *Theoretical Analysis of Gimballess Inertial Reference Equipment Using Delta-Modulated Instruments*. PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1962. Department of Aeronautical and Astronautical Engineering.

[39] V. Milenković. Coordinates suitable for angular motion synthesis in robots. In *Proceedings of the Robot VI Conference, Detroit MI, March 2-4, 1982*, 1982. Paper MS82-217.

[40] R.A. Wehage. Quaternions and Euler parameters. A brief exposition. In E.J. Haug, editor, *Computer Aided Analysis and Optimization of Mechanical Systems Dynamics*, pages 147–180. Springer-Verlag, Berlin, Heidelberg, 1984.

[41] M. Géradin and A. Cardona. Kinematics and dynamics of rigid and flexible mechanisms using finite elements and quaternion algebra. *Computational Mechanics*, 4:115–135, 1989.

[42] A.R. Klumpp. Singularity-free extraction of a quaternion from a direction-cosine matrix. *Journal of Spacecraft and Rockets*, 13:754–755, December 1976.

[43] S.W. Shepperd. Quaternion from rotation matrix. *Journal of Guidance and Control*, 1:223–224, May-June 1978.

[44] K.J. Bathe. *Finite Element Procedures*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1996.