Multibody Syst Dyn
DOI 10.1007/s11044-015-9451-1

A fast multi-obstacle muscle wrapping method using natural geodesic variations

Andreas Scholz[1] · Michael Sherman[2] · Ian Stavness[3] · Scott Delp[4] · Andrés Kecskeméthy[5]

Received: 28 July 2014 / Accepted: 2 March 2015

Abstract Musculoskeletal simulation has become an essential tool for understanding human locomotion and movement disorders. Muscle-actuated simulations require methods that continuously compute musculotendon paths, their lengths, and their rates of length change to determine muscle forces, moment arms, and the resulting body and joint loads. Musculotendon paths are often modeled as locally length minimizing curves that wrap frictionlessly over moving obstacle surfaces representing bone and tissue. Biologically accurate wrapping surfaces are complex, and a single muscle path may wrap around many obstacles. However, state-of-the-art muscle wrapping methods are either limited to analytical results for a pair of simple surfaces, or they are computationally expensive. In this paper, we introduce the Natural Geodesic Variation (NGV) method for the fast and accurate computation of a musculotendon's shortest path across an arbitrary number of general smooth wrapping surfaces, and an explicit formula for the path's exact rate of length change. The total path is regarded as a concatenation of straight-line segments between local surface geodesics, where each geodesic is naturally parameterized by its starting point, direction, and length.

✉ A. Scholz
scholz.andreas@uni-due.de

M. Sherman
msherman@stanford.edu

I. Stavness
ian.stavness@usask.ca

S. Delp
delp@stanford.edu

A. Kecskeméthy
andres.kecskemethy@uni-due.de

[1] Department of Mechanical Engineering, University of Duisburg-Essen, 47057 Duisburg, Germany

[2] Department of Bioengineering, Stanford University, Stanford, CA 94305, USA

[3] Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5A2, Canada

[4] Departments of Bioengineering and Mechanical Engineering, Stanford University, Stanford, CA 94305, USA

[5] Chair of Mechanics and Robotics, University of Duisburg-Essen, 47057 Duisburg, Germany

The shortest path is computed by finding the root of a global path-error constraint equation that enforces that the geodesics connect collinearly with adjacent straight-line segments. High computational speed is achieved using Newton's method to zero the path error, with an explicit, banded Jacobian that maps natural variations of the geodesic parameters to path-error variations. Three simulation benchmarks demonstrate that the NGV method computes high-precision solutions for path length and rate of length change, allows for wrapping over biologically accurate surfaces, and is capable of simulating muscle paths over hundreds of surfaces in real time. We thus believe the NGV method will facilitate the development of more accurate yet very efficient musculoskeletal models.

Keywords Muscle wrapping · Musculotendon path · Shortest path · Geodesic · Geodesic variation · Jacobi field

1 Introduction

Musculoskeletal simulations provide a quantitative means to predict internal body loads given a set of experimentally measured body kinematics and ground reaction forces. Therefore, musculoskeletal models are widely used to study pathological gait patterns [1–5], to predict the contribution of individual muscles to motion such as walking [6–10], running [11, 12], and shoulder motion [13, 14], to quantify bone-on-bone contact forces and joint loads [15–18], and for surgical planning [19]. Internal body loads depend on muscle forces and the paths of muscles inside the body. To better understand the force-generating capacity of anatomical muscle, researchers have studied isolated muscles and developed models of contraction dynamics [20–24]. It is now well-established that a muscle's ability to generate force depends on the path's current length and its rate of length change, yet the in-vivo measurement of muscle paths and forces is still extremely challenging. While it is possible to accurately measure the coordinates of muscle origin and insertion points from cadavers [25, 26], experimental data of muscle forces and paths has only been obtained in few cases using MRI [27, 28] and force transducers attached to tendons [29, 30].

Musculoskeletal models require algorithms that compute muscle paths, their lengths and their rates of length change to determine muscle forces and the contribution of individual muscles to motion. Muscles commonly wrap around multiple complex anatomical obstacles such as bones and neighboring tissue, thus most muscle paths cannot be represented adequately by straight lines. Therefore, a broad variety of muscle wrapping approaches has been reported in the literature [13, 28, 31–47]. Finite element algorithms [28, 36, 38, 44, 45] provide the highest level of detail because they allow for considering muscle deformations and using realistic bone geometry for wrapping, but they are computationally expensive. In the majority of musculoskeletal models, muscle paths are approximated by length-minimizing curves (geodesics) that transmit forces to the skeleton while wrapping around geometric obstacle surfaces representing bone and tissue [13, 31–35, 37, 39–43, 46, 47]. Such curved-line approaches have been widely used to simulate the upper limb [31, 39, 40], the lower limb [33, 35, 42], and the shoulder [13, 14].

Curved-line muscle wrapping approaches can roughly be subdivided into two groups: approaches using path or surface discretizations [33, 35, 37, 39, 40, 43] and approaches using smooth curves on smooth wrapping surfaces [31, 32, 37, 39, 41, 46]. Discretized surfaces such as bone meshes obtained from CT or MRI scans [33, 34, 48] provide generality and low computational costs, but cause nonsmooth rate of length change during path evolution and wrapping over surface edges. This can slow down variable step size integrators during simulation and introduces discontinuities in muscle force. Researchers have also applied nonlinear optimization to compute discretized shortest muscle paths [37, 39]. They minimized the energy of a series of lumped springs, given implicit surface equations as unilateral constraints. Their approach allows for using multiple implicit surfaces and provides good approximations of the exact shortest path when the level of path discretization is sufficiently high. However, computational costs grow significantly with the level of path discretization and the number of surfaces, and the method yields nonsmooth rate of length change.

Smooth curves and surfaces are necessary for muscle wrapping to avoid the nonsmooth behavior of discretizations. An early smooth wrapping approach was presented by van der Helm [13]. He used spheres, cylinders, and ellipsoids for single-object wrapping around the shoulder, approximating surface geodesics by planar curves. Garner and Pandy [31] introduced the Obstacle-Set method which computes the exact shortest (geodesic) path across a maximum of two spheres, cylinders, or a combination of both. Their method requires a series of case distinctions and does not generalize for more than two elementary surfaces. Stavness et al. [46] regarded the total path as a concatenation of straight-line segments between two geodesic segments on each surface, where each geodesic segment is assumed to emanate in the direction of its adjacent straight-line segment. They computed the shortest path across multiple implicit surfaces by iterating the origin-point positions of the geodesic segments such that the two geodesic segments on each surface connect collinearly at their closest points, and the adjacent straight-line segments are tangent to the surface. This approach is general and accurate but computationally slow as it relies on finite-differences Jacobians and requires nested loops for finding the closest points on each pair of local geodesic segments. Scholz et al. [47] used a single geodesic segment per surface and formulated the constraints for the shortest path solely at the transitions between the geodesic segments and their adjacent straight-line segments. In that work, each geodesic segment was parameterized by the coordinates of its boundary points on general parametric surfaces, and a system of local path-error constraints was introduced which enforces that the transitions between all geodesic segments and adjacent straight-line segments are collinear. That method allowed for the computation of shortest muscle paths across multiple surfaces by solving a system of nonlinear path-error constraints with an explicit Jacobian. Hence, that method is more efficient than [46], yet it still requires nested loops for computing geodesics between two points on a surface. Overall, previous approaches to smooth-surface wrapping did not address the explicit computation of the rate of length change of the muscle path, which is an important input for computing muscle forces using Hill-type muscle models [20–22, 24].

There is no muscle wrapping algorithm in the literature that computes a muscle's shortest path over multiple biologically realistic surfaces in real time, as well as the path's exact rate of length change. As a result, biomechanists face a trade-off between the computational speed and the accuracy of their models. In this paper, we introduce the Natural Geodesic Variation (NGV) method that allows for both the accurate and fast computation of a muscle's shortest path as it wraps across an arbitrary number of general smooth wrapping surfaces. Analogously to [47], the path is regarded as a concatenation of straight-line segments which have to connect collinearly to local geodesic segments on the surfaces. The collinearity conditions are used to state a nonlinear path-error constraint equation, whose root is computed iteratively to find the shortest muscle path. The new approach to solving the shortest-path problem presented here consists of (i) naturally parameterizing each surface geodesic by its starting point, direction, and length; (ii) introducing four natural, i.e., independent, variations of the parameters of each geodesic; and (iii) completely elaborating the differential geometric relationships between the natural geodesic variations and the resulting path-error variations. Thereby, the shortest-path problem can be solved very efficiently without nested loops by applying a gradient-based root-finding method, e.g., Newton's method, with an explicit banded Jacobian. In addition to solving the shortest-path problem, we present an explicit formula for the path's exact rate of length change which is independent of the formulation used to solve for the path.

The paper is structured as follows: Sect. 2 introduces the global path's constraint equations and the unknowns of the root-finding problem, i.e., the geodesic parameters. Section 3 elaborates the differential-geometric concepts which are required to compute the path-error Jacobian for the Newton iterations explicitly. Section 4 describes the structure of the constraint Jacobian and its assembly from the terms derived in Sect. 3. In Sect. 5, an explicit formula for the path's exact rate of length change is derived. Section 6 contains the results of three simulation benchmarks which were used to evaluate the speed and the accuracy of our method: (i) an accuracy benchmark that compares our method with a conventional nonlinear energy-minimization approach; (ii) a geometry benchmark that contains multiple complex wrapping obstacles including a surface patch fitted to a human ribcage; and (iii) a computational-speed benchmark to measure the computational costs when wrapping a single muscle over a variable number of moving obstacles.

2 Formulation of the shortest-path problem as a root-finding problem

Our goal is to compute the path of a massless taut string between an origin point O and an insertion point I that wraps frictionlessly over an ordered set of n obstacle surfaces $\mathcal{S}^i$ $(i = 1, \dots, n)$. At any instant of time, the string's path is a global geodesic, i.e., it minimizes the arc length with respect to all other neighboring curves connecting O and I. We assume here that the global path always touches all surfaces (no lift-off considered), thus that it consists of an alternating sequence of $n + 1$ straight-line segments and n local geodesic curves γ^i on the surfaces. We will further assume that the complete surface patch of interest (of contact) of surface $\mathcal{S}^i$ can be covered by a nonsingular differentiable parameterization $x^i(u^i, v^i) : \mathbb{R}^2 \mapsto \mathbb{R}^3$ with respect to a surface-fixed coordinate frame $\mathcal{K}^i_S$ in terms of two surface coordinates $(u^i, v^i) \in \mathbb{R}^2$. The surface coordinates are chosen so that the outward unit normal of $\mathcal{S}^i$ with respect to $\mathcal{K}^i_S$ is given by

$$N^i := \frac{x^i_u \times x^i_v}{\|x^i_u \times x^i_v\|}, \qquad x_{(\cdot)} := \frac{\partial x}{\partial(\cdot)}. \tag{1}$$

The parameterization $x(u, v)$ on $\mathcal{S}$ (index i omitted) allows for computing arc length parameterized geodesics $\gamma = x(u(s), v(s)) : [0, \ell_\gamma] \mapsto \mathcal{S}$ with length ℓ_γ by solving two ordinary differential equations [49]

$$u'' + \Gamma^1_{11}\left(u'\right)^2 + 2\Gamma^1_{12}u'v' + \Gamma^1_{22}\left(v'\right)^2 = 0, \tag{2}$$

$$v'' + \Gamma^2_{11}\left(u'\right)^2 + 2\Gamma^2_{12}u'v' + \Gamma^2_{22}\left(v'\right)^2 = 0, \tag{3}$$

where $(\cdot)' = \mathrm{d}(\cdot)/\mathrm{d}s$ denotes a derivative with respect to arc length s. For the Christoffel symbols in Eqs. (2) and (3) it holds

$$\Gamma^1_{11} = \frac{\frac{1}{2}\,\mathrm{E}_u\,\mathrm{G} - \mathrm{A}\,\mathrm{F}}{\mathrm{E}\,\mathrm{G} - \mathrm{F}^2}, \tag{4}$$

$$\Gamma^1_{12} = \frac{\mathrm{C}\,\mathrm{G} - \mathrm{D}\,\mathrm{F}}{\mathrm{E}\,\mathrm{G} - \mathrm{F}^2}, \tag{5}$$

$$\Gamma^1_{22} = \frac{\mathrm{B\,G} - \frac{1}{2}\mathrm{F\,G}_v}{\mathrm{E\,G} - \mathrm{F}^2}, \tag{6}$$

$$\Gamma^2_{11} = \frac{\frac{1}{2}\mathrm{E}_u\,\mathrm{F} - \mathrm{A\,E}}{\mathrm{F}^2 - \mathrm{E\,G}}, \tag{7}$$

$$\Gamma^2_{12} = \frac{\mathrm{C\,F} - \mathrm{D\,E}}{\mathrm{F}^2 - \mathrm{E\,G}}, \tag{8}$$

$$\Gamma^2_{22} = \frac{\mathrm{B\,F} - \frac{1}{2}\mathrm{E\,G}_v}{\mathrm{F}^2 - \mathrm{E\,G}}, \tag{9}$$

with $\mathrm{A} = x_{uu} \cdot x_v$, $\mathrm{B} = x_u \cdot x_{vv}$, $\mathrm{C} = x_u \cdot x_{uv}$, $\mathrm{D} = x_{uv} \cdot x_v$, $\mathrm{E} = x_u \cdot x_u$, $\mathrm{F} = x_u \cdot x_v$, $\mathrm{G} = x_v \cdot x_v$, $\mathrm{E}_u = 2\,x_{uu} \cdot x_u$, $\mathrm{G}_v = 2\,x_{vv} \cdot x_v$. Equations (2) and (3) can be regarded as the equations of motion of a freely moving particle on S. Their solution requires four initial conditions: the surface coordinates (u_P, v_P) of the geodesic's start point P (the particle's start point), and the derivatives (u'_P, v'_P) of these coordinates with respect to arc length, defining the curve's initial direction (the particle's initial velocity). We choose (u'_P, v'_P) such that γ is a unit-speed curve, that is,

$$\|x'\| = \|x_u u' + x_v v'\| = 1. \tag{10}$$

Without loss of generality, we assume that the initial length of each local geodesic γ^i is zero. Thus it can be parameterized by the following five parameters

$$q^i := \begin{bmatrix} u^i_P & v^i_P & u^{i\,\prime}_P & v^{i\,\prime}_P & \ell^i_\gamma \end{bmatrix}^{\mathrm{T}}, \tag{11}$$

of which, however, only four are independent due to the unit-speed constraint for x' (see Eq. (10)).

At the geodesic's start point $P^i : x^i_P = x^i(u^i(0), v^i(0))$ and end point $Q^i : x^i_Q = x^i(u^i(\ell^i_\gamma), v^i(\ell^i_\gamma))$, we define two orthonormal boundary-point trihedra (see Fig. 1)

$$\mathcal{K}^i_P := \left\{x^i_P,\ t^i_P,\ N^i_P,\ B^i_P\right\}, \tag{12}$$

$$\mathcal{K}^i_Q := \left\{x^i_Q,\ t^i_Q,\ N^i_Q,\ B^i_Q\right\}, \tag{13}$$

where $t = x'$ is the geodesic's tangent, N is the surface's normal (see Eq. (1)), and $B := t \times N$ is the respective binormal. We further define unit vectors e^i and e^{i+1} along the straight-line segments (see Fig. 1) given the positions of all geodesic boundary points on the surfaces and the positions of the points O and I.

At the path's solution configuration, all transitions between straight-line segments and adjacent geodesics are collinear. Thus the projections of the unit vectors of adjacent straight-line segments onto the normal and binormal vectors of the geodesics' boundary-point trihedra vanish. Figure 1 shows a collinear transition at P^i and a collinearity condition violation at Q^i. We use the collinearity conditions to measure the deviation of a candidate path from the solution path and define a local path-error function [47]

$$\varepsilon^i\left(q^{i-1}, q^i, q^{i+1}\right) := \begin{bmatrix} e^i & \cdot & N^i_P \\ e^i & \cdot & B^i_P \\ e^{i+1} & \cdot & N^i_Q \\ e^{i+1} & \cdot & B^i_Q \end{bmatrix} \tag{14}$$

on each surface S^i, where $\cdot$ is the scalar-product operator. Note that ε^i is a function of q^{i-1} only if $i > 1$, and a function of q^{i+1} only if $i < n$. Assembling the local path errors (Eq. (14)) as well as the geodesics' parameters (Eq. (11)) into global vectors

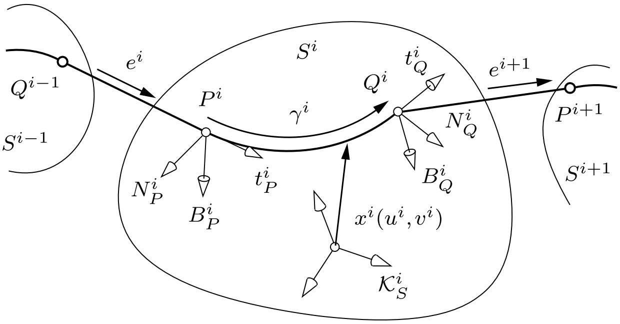

Fig. 1 Example of a collinear transition between the geodesic γ^i and the straight-line segment e^i at the geodesic's start point P^i, and a noncollinear transition between γ^i and e^{i+1} at the geodesic's end point Q^i. The deviation from collinearity of each transition is measured by projecting the unit vectors of adjacent straight-line segments onto the normal and binormal vectors of the geodesic's boundary-point trihedra, yielding a local path-error function ε^i for each geodesic

$$\varepsilon(q) := \begin{bmatrix} \varepsilon^1 \\ \varepsilon^2 \\ \vdots \\ \varepsilon^n \end{bmatrix} \in \mathbb{R}^{4n\times 1}, \qquad q := \begin{bmatrix} q^1 \\ q^2 \\ \vdots \\ q^n \end{bmatrix} \in \mathbb{R}^{5n\times 1}, \tag{15}$$

yields the global nonlinear constraint equation system for the global path error ε of the muscle path

$$\varepsilon(q) = 0. \tag{16}$$

Our approach to compute the locally shortest path between O and I is to apply a gradient-based root-finding method for finding all $5n$ unknowns in q that fulfill the $4n$ collinearity constraints in Eq. (16). Equation (16) must be solved at every simulation time step, thus it can be considered as independent of time for the root-finding iterations (points O and I as well as the surfaces S^i can be considered as fixed). Equation (16) contains n more unknowns than equations, where $2n$ unknowns are pairwise dependent on each other due to the unit-speed constraint Eq. (10). A direct solution of Eq. (16) for q would require adding the n unit-speed constraints (see Eq. (10)) to form a regular equation system. However, this would significantly increase the complexity of the solution approach, and also make it dependent on the choice of the surface coordinates (u, v). Instead, our approach consists in (i) introducing a minimal set of four natural geodesic variations of the five parameters defining each geodesic, which are independent both of each other and the surface coordinates; and (ii) computing the derivative of the global path error with respect to the natural variations to obtain the gradient of the path error. Note that one of the four natural geodesic variations will be nonholonomic. Nevertheless, for notational convenience, we will use the term derivative in the sense of a velocity-transmission coefficient relating output variations to corresponding input variations.

We introduce the following natural variations of the geodesic's parameters: (i) the infinitesimal displacement $\mathrm{d}s^i_P$ of the geodesic's start point P^i in direction of the tangent t^i_P; (ii) the infinitesimal displacement $\mathrm{d}\beta^i_P$ of P^i in direction of the binormal B^i_P; (iii) the infinitesimal clockwise rotation $\mathrm{d}\theta^i$ of the geodesic's initial direction about the normal N^i_P; and (iv) the infinitesimal length increment $\mathrm{d}\ell^i_\gamma$ of the geodesic at point Q^i when point P^i is fixed. For each geodesic, we obtain a vector of four independent natural geodesic variations

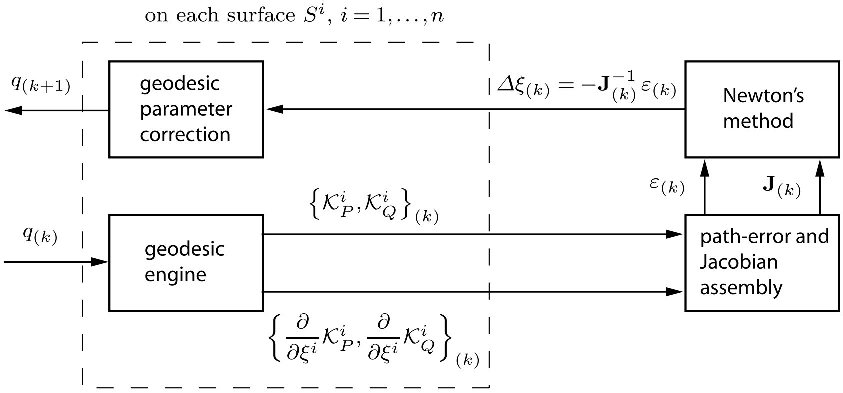
on each surface S^i, $i = 1,\dots,n$
$q_{(k+1)}$
geodesic parameter correction
$\Delta\xi_{(k)} = -\mathbf{J}^{-1}_{(k)}\varepsilon_{(k)}$
Newton's method
$\varepsilon_{(k)}$
$\mathbf{J}_{(k)}$
$\{\mathcal{K}^i_P, \mathcal{K}^i_Q\}_{(k)}$
$q_{(k)}$
geodesic engine
path-error and Jacobian assembly
$\left\{\frac{\partial}{\partial\xi^i}\mathcal{K}^i_P, \frac{\partial}{\partial\xi^i}\mathcal{K}^i_Q\right\}_{(k)}$

Fig. 2 General solution flow of a single Newton step k. For a given set of geodesic parameters $q_{(k)}$, a geodesic engine computes all local surface geodesics, the boundary-point trihedra and their derivatives with respect to the natural geodesic variations. From the spatial positions of the boundary point trihedra and their derivatives, the global path error and the Jacobian are assembled. According to Newton's method, a linear equation system is solved to compute the natural corrections $\Delta\xi_{(k)}$ to correct the positions of all geodesics. These corrections $\Delta\xi_{(k)}$ are mapped back to corrections $\Delta q_{(k)}$ of the geodesic parameters to obtain a new set $q_{(k+1)}$ of geodesic parameters for the next iteration step $k+1$

$$\mathrm{d}\xi^i := \begin{bmatrix}\mathrm{d}s^i_P & \mathrm{d}\beta^i_P & \mathrm{d}\theta^i & \mathrm{d}\ell^i_\gamma\end{bmatrix}^{\mathrm{T}}. \tag{17}$$

Accordingly, we obtain the global path-error Jacobian

$$\mathbf{J} := \frac{\partial\varepsilon}{\partial\xi} \in \mathbb{R}^{4n\times 4n}, \qquad \mathrm{d}\xi := \begin{bmatrix}\mathrm{d}\xi^1\\ \mathrm{d}\xi^2\\ \vdots\\ \mathrm{d}\xi^n\end{bmatrix} \in \mathbb{R}^{4n\times 1}, \tag{18}$$

mapping natural geodesic variations to path-error variations. Given a reasonably close initial guess for q, the Jacobian $\mathbf{J}$ can be used in a gradient-based root-finding method to compute finite corrections $\Delta\xi$ of the geodesics' initial conditions until Eq. (16) is fulfilled. Here we will use Newton's method to accomplish this task. Figure 2 gives an overview of the four main steps for a single path-correction iteration k: (i) shooting geodesics on all surfaces for a given set of parameters $q_{(k)}$ and computing the boundary-point trihedra as well as their derivatives; (ii) evaluating the global path error $\varepsilon_{(k)}$ and the path-error Jacobian $\mathbf{J}_{(k)}$; (iii) computing the natural path corrections $\Delta\xi_{(k)}$; and (iv) using the natural corrections $\Delta\xi_{(k)}$ to compute the new set of geodesic parameters $q_{(k+1)}$. In this paper, it will be assumed that the initial guess $q_{(0)}$ is sufficiently close to the solution, i.e., close enough so that Newton's method converges to the solution. During a simulation, this can be achieved by using the solution of a previous time step as the initial guess for the current time step.

The following Sect. 3 is concerned with computing the partial derivatives of the boundary-point trihedra with respect to the natural geodesic variations, which are required to compute the path-error Jacobian $\mathbf{J}$ explicitly. A detailed discussion of the structure of $\mathbf{J}$ as well as the mapping from finite natural corrections to corrections of the geodesics' parameters will be given in Sect. 4.

3 Derivatives of the boundary-point trihedra

The global path-error Jacobian **J** comprises the partial derivatives of the local path errors in Eq. (14) with respect to the natural geodesic variations in Eq. (17). Accordingly, we require eight partial derivatives per geodesic: the partial derivatives of the two boundary-point trihedra with respect to the four natural geodesic variations. This section thus contains four subsections, each describing the partial derivatives of the trihedra with respect to one natural variation. Note that we will only consider a generic geodesic and thus omit the index i throughout the whole section for notational convenience.

When the parameters of a local geodesic are varied by $\mathrm{d}\xi$, its boundary point trihedra $\mathcal{K}_P$ and $\mathcal{K}_Q$ move along particular curves on S. We are interested in the differential geometry of these curves to compute the variations $\mathrm{d}\mathcal{K}_P$ and $\mathrm{d}\mathcal{K}_Q$ in terms of the natural geodesic variations $\mathrm{d}\xi$. The previously defined trihedra $\mathcal{K}_P$ and $\mathcal{K}_Q$ can be regarded as moving Darboux trihedra on S. For a general Darboux trihedron $\{\tilde{x}, \tilde{t}, \tilde{N}, \tilde{B}\}$ that travels along some arc-length parameterized curve $\sigma(\cdot)$ on a surface, the following differential relationships hold [50]

$$\frac{\mathrm{d}}{\mathrm{d}(\cdot)}\begin{bmatrix}\tilde{x}\\ \tilde{t}\\ \tilde{N}\\ \tilde{B}\end{bmatrix} = \begin{bmatrix}1 & 0 & 0\\ 0 & \kappa_n & \kappa_g\\ -\kappa_n & 0 & -\tau\\ -\kappa_g & \tau & 0\end{bmatrix}^{\sigma}\begin{bmatrix}\tilde{t}\\ \tilde{N}\\ \tilde{B}\end{bmatrix}. \tag{19}$$

The tilde notation in Eq. (19) distinguishes vectors of the general trihedron from vectors of the trihedra at the geodesic's boundary points. In the above Eq. (19), κ_g^σ is the geodesic or tangential curvature, τ^σ is the geodesic torsion, and κ_n^σ is the normal curvature of the surface. The latter two are easily evaluated by the general formulas [49, 50]

$$\tau = \frac{(\mathrm{FL}-\mathrm{EM})\,\mathrm{d}u^2 + (\mathrm{GL}-\mathrm{EN})\,\mathrm{d}u\,\mathrm{d}v + (\mathrm{GM}-\mathrm{FN})\,\mathrm{d}v^2}{\sqrt{\mathrm{EG}-\mathrm{F}^2}(\mathrm{E}\,\mathrm{d}u^2 + 2\,\mathrm{F}\,\mathrm{d}u\,\mathrm{d}v + \mathrm{G}\,\mathrm{d}v^2)} \tag{20}$$

and

$$\kappa_n = \frac{\mathrm{L}\,\mathrm{d}u^2 + 2\,\mathrm{M}\,\mathrm{d}u\,\mathrm{d}v + \mathrm{N}\,\mathrm{d}v^2}{\mathrm{E}\,\mathrm{d}u^2 + 2\,\mathrm{F}\,\mathrm{d}u\,\mathrm{d}v + \mathrm{G}\,\mathrm{d}v^2}. \tag{21}$$

In the above Eqs. (20) and (21), $\mathrm{E} = x_u \cdot x_u$, $\mathrm{F} = x_u \cdot x_v$, $\mathrm{G} = x_v \cdot x_v$ and $\mathrm{L} = N \cdot x_{uu}$, $\mathrm{M} = N \cdot x_{uv}$, $\mathrm{N} = N \cdot x_{vv}$ are the coefficients of the first and second fundamental forms of S, respectively, where N is the surface normal. Both τ and κ_n only depend on a point on S and the direction of the curve passing through that point, thus these quantities can be evaluated easily at the geodesic's boundary points. However, evaluating the geodesic curvature κ_g requires more knowledge about the curve considered. In what follows, we will compute κ_g of the curves along which $\mathcal{K}_P$ and $\mathcal{K}_Q$ travel when the geodesic's initial conditions are varied.

3.1 Variation of the geodesic's start point position in tangential direction

Stating the global path-error Jacobian explicitly requires information about how the boundary point trihedra $\mathcal{K}_P$ and $\mathcal{K}_Q$ displace when the geodesic parameters are varied by $\mathrm{d}\xi$. This subsection describes how the geodesic's boundary point trihedra $\mathcal{K}_P$ and $\mathcal{K}_Q$ displace when the initial start point P is displaced in tangential direction t_P by $\mathrm{d}s_P$. For a tangential displacement $\mathrm{d}s_P$ of the geodesic's start point P, the trihedron $\mathcal{K}_P$ travels along the geodesic γ. Given constant length ℓ_γ of the geodesic, the end point Q displaces in direction of t_Q by the same amount. Let $\mathrm{d}s_Q$ and $\mathrm{d}\beta_Q$ be arc length elements on S at Q in direction of t_Q and B_Q, respectively. Then, the following simple relations hold:

$$\frac{\partial s_Q}{\partial s_P} = 1, \qquad \frac{\partial \beta_Q}{\partial s_P} = 0. \tag{22}$$

Since γ is a geodesic, it has zero geodesic curvature. Inserting $\kappa_g^\gamma = 0$ into Eq. (19) yields the derivatives $\partial\mathcal{K}_P/\partial s_P$ and $\partial\mathcal{K}_Q/\partial s_P$ expressed in $\mathcal{K}_P$ and $\mathcal{K}_Q$

$$\frac{\partial}{\partial s_P}\begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_{P,Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \kappa_n & 0 \\ -\kappa_n & 0 & -\tau \\ 0 & \tau & 0 \end{bmatrix}_{P,Q}^{\gamma} \begin{bmatrix} t \\ N \\ B \end{bmatrix}_{P,Q}. \tag{23}$$

3.2 Variation of the geodesic's start point position in binormal direction

For the evaluation of the variations $\mathrm{d}\mathcal{K}_P$ and $\mathrm{d}\mathcal{K}_Q$ with respect to a binormal displacement of P, we assume that the geodesic γ is parallel-transported along another geodesic $\gamma^\perp$ through P orthogonal to γ in direction of B_P (see Fig. 3). The parallel transport of γ along $\gamma^\perp$ ensures that the initial direction of γ is (locally) kept constant, since $\gamma^\perp$ has vanishing tangential curvature $\kappa_g^\perp = 0$.

At point P, the tangent of $\gamma^\perp$ is equal to the binormal of γ, the binormal of $\gamma^\perp$ is equal to the negative tangent of γ, and both surface normal vectors are equal. Thus for $\partial\mathcal{K}_P/\partial\beta_P$ expressed in $\mathcal{K}_P$ it holds

$$\frac{\partial}{\partial \beta_P}\begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_{P} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -\tau & 0 \\ \tau & 0 & -\kappa_n \\ 0 & \kappa_n & 0 \end{bmatrix}_{P}^{\perp} \begin{bmatrix} t \\ N \\ B \end{bmatrix}_{P}, \tag{24}$$

where $\kappa_{n,P}^\perp$ and $\tau_P^\perp$ are the normal curvature and the geodesic torsion of $\gamma^\perp$ at P in direction of B_P. Note that two orthogonal curves through one point have the same geodesic torsion but with a different sign. Thus it holds $\tau_P^\perp = -\tau_P^\gamma$, which can be shown by computing $\partial t_P/\partial\beta_P$ explicitly. By Young's theorem, it holds

$$\begin{aligned} \frac{\partial t_P}{\partial \beta_P} &= \frac{\partial}{\partial \beta_P}\left(\frac{\partial x_P}{\partial s_P}\right) \\ &= \frac{\partial}{\partial s_P}\left(\frac{\partial x_P}{\partial \beta_P}\right) = \frac{\partial}{\partial s_P}\left(B_P\right) = \tau_P^\gamma \, N_P. \end{aligned} \tag{25}$$

Comparing Eq. (25) with Eq. (24) confirms $\tau_P^\perp = -\tau_P^\gamma$.

We still require information about how $\mathrm{d}\beta_P$ transmits through the geodesic to a displacement of the end point trihedron $\mathcal{K}_Q$. This transmission is given by the variational vector field $x_{\beta_P}(s)$ along γ, for whose derivation we briefly review Jacobi fields. In general, variational vector fields $J(s) \in \mathbb{R}^3$ along surface geodesics fulfill the Jacobi equation (see [51])

$$\nabla^2 J(s) + K(s)\,(t(s) \times J(s)) \times t(s) = 0, \tag{26}$$

where

$$K = \frac{\mathrm{L}\,\mathrm{N} - \mathrm{M}^2}{\mathrm{E}\,\mathrm{G} - \mathrm{F}^2} \tag{27}$$

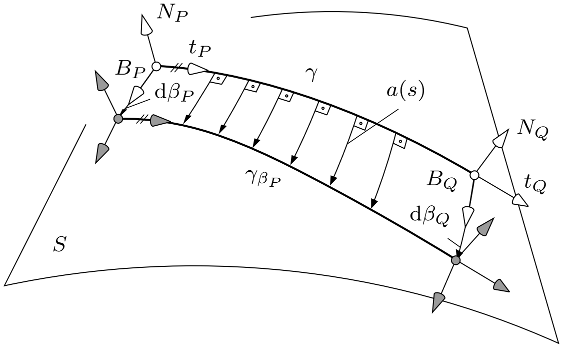

Fig. 3 A binormal displacement $\mathrm{d}\beta_P$ of the geodesic's start point trihedron $\mathcal{K}_P$ causes a binormal displacement $\mathrm{d}\beta_Q$ of the end point trihedron $\mathcal{K}_Q$. The transmission from $\mathrm{d}\beta_P$ to $\mathrm{d}\beta_Q$ is described by the scalar Jacobi differential Eq. (33)

is the Gaussian curvature (see [49, 52]) and ∇ denotes a covariant derivative (see [51]) with respect to arc length s. The covariant derivative is equal to the usual Euclidean derivative projected onto the tangent space of the surface. An important property of Jacobi fields is that they fulfill the following equation (see [53])

$$J(s) \cdot t(s) = J(0) \cdot t(0) + \big(J'(0) \cdot t(0)\big)\, s, \tag{28}$$

which relates the behavior of a Jacobi field at any point along the geodesic with its behavior at $s = 0$. For the considered binormal variation of point P, we are particularly interested in the behavior of $J(s) = x_{\beta_P}(s)$. At $s = 0$, by definition, $x_{\beta_P}(0) = B_P$ and $t(0) = t_P$, yielding

$$x_{\beta_P}(0) \cdot t(0) = 0. \tag{29}$$

Since $x_{\beta_P}(0) = B_P$ we have from Eq. (23) that $\partial B_P/\partial s_P = \tau_P^\gamma N_P$ and thus

$$x'_{\beta_P}(0) \cdot t(0) = 0. \tag{30}$$

Substituting $J(s) = x_{\beta_P}(s)$ in Eq. (28) and inserting Eqs. (29) as well as (30) yields

$$x_{\beta_P}(s) \cdot t(s) = 0, \qquad s \in [0, \ell_\gamma]. \tag{31}$$

Equation (31) shows that the sought variational vector field $x_{\beta_P}(s)$ is always orthogonal to the geodesic (see Fig. 3), or zero at points conjugate to P (see [51]). Accordingly, $x_{\beta_P}(s)$ is a multiple of some scalar $a(s)$ and the binormal $B(s)$

$$x_{\beta_P}(s) = a(s)\, B(s), \quad a(s) \in \mathbb{R},\ s \in [0, \ell_\gamma]. \tag{32}$$

Inserting $J(s) = a(s)\, B(s)$ into Eq. (26) and multiplying it by $B(s)$ simplifies the vector Jacobi Eq. (26) to a scalar Jacobi equation

$$a''(s) + K(s)\, a(s) = 0, \tag{33}$$

with

$$a(s) = x_{\beta_P}(s) \cdot B(s). \tag{34}$$

Note that $a'(s) = \nabla a(s)$ so we prefer the prime for notational convenience.

Solving the scalar Jacobi Eq. (33) requires the two initial conditions $a(0)$ and $a'(0)$. Since $x_{\beta_P}(0) = B_P$ we directly obtain $a(0) = 1$ by evaluating Eq. (34) at $s = 0$. Furthermore, we obtain by differentiating Eq. (34)

$$a'(s) = \nabla x_{\beta_P}(s) \cdot B(s) + x_{\beta_P}(s) \cdot \nabla B(s). \tag{35}$$

Since $B(s)$ is the parallel transport of B_P along γ, it holds $\nabla B(s) = 0$ and the latter term on the right hand side of Eq. (35) vanishes. Inserting $\nabla x_{\beta_P}(0) = \nabla B(0) = 0$ into Eq. (35) directly yields $a'(0) = 0$. Given the initial conditions of Eq. (33), we obtain the relation between binormal displacements of the start point P and the end point Q by evaluating Eq. (33) at $s = \ell_\gamma$

$$\frac{\partial \beta_Q}{\partial \beta_P} = a(\ell_\gamma) = a_Q. \tag{36}$$

We are now in a position to compute $\partial \mathcal{K}_Q / \partial \beta_P$ using the Darboux Eqs. (19). Let α denote the curve along which $\mathcal{K}_Q$ moves when point P is displaced in binormal direction. At Q, the tangent of α is equal to the binormal of γ, the binormal of α is equal to the negative tangent of γ, and both normals are equal. Thus it holds for $\partial \mathcal{K}_Q / \partial \beta_Q$ expressed in $\mathcal{K}_Q$

$$\frac{\partial}{\partial \beta_Q} \begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_Q = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -\tau & \kappa_g \\ \tau & 0 & -\kappa_n \\ -\kappa_g & \kappa_n & 0 \end{bmatrix}_Q^\alpha \begin{bmatrix} t \\ N \\ B \end{bmatrix}_Q . \tag{37}$$

The normal curvature $\kappa^\alpha_{n,Q}$ is easily computed using Eq. (21). For the geodesic torsion it holds $\tau^\alpha_Q = -\tau^\gamma_Q$, which is already known after evaluating Eq. (23) at Q. We still require the geodesic curvature $\kappa^\alpha_{g,Q}$ of α at Q. To derive it, we use the fact that $x(s, \beta_P)$ is an orthogonal surface parameterization along the geodesic γ. In the orthogonal surface parameterization $x(s, \beta_P)$, it holds $\mathrm{F}^{(s,\beta_P)} = x_s \cdot x_{\beta_P} = 0$, and the geodesic curvature κ^α_g is given by (see [51])

$$\kappa^\alpha_g = \frac{\mathrm{G}^{(s,\beta_P)}_s}{2\,\mathrm{G}^{(s,\beta_P)} \sqrt{\mathrm{E}^{(s,\beta_P)}}}. \tag{38}$$

Since $\mathrm{E}^{(s,\beta_P)} = x_s \cdot x_s = 1$ and $\mathrm{G}^{(s,\beta_P)} = a\,B \cdot a\,B = a^2$, Eq. (38) simplifies to

$$\kappa^\alpha_g = \frac{a'}{a}. \tag{39}$$

Note that κ^α_g vanishes for $\ell_\gamma = 0$ which is consistent with the previously made assumption that γ is parallel transported along a binormal geodesic $\gamma^\perp$. Inserting Eq. (39) into Eq. (37) and multiplying by $a_Q = \partial \beta_Q / \partial \beta_P$ yields the sought derivative $\partial \mathcal{K}_Q / \partial \beta_P$ expressed in $\mathcal{K}_Q$

$$\frac{\partial}{\partial \beta_P} \begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_Q = \begin{bmatrix} 0 & 0 & a \\ 0 & a\,\tau^\gamma & a' \\ -a\,\tau^\gamma & 0 & -a\,\kappa^\alpha_n \\ -a' & a\,\kappa^\alpha_n & 0 \end{bmatrix}_Q \begin{bmatrix} t \\ N \\ B \end{bmatrix}_Q , \tag{40}$$

where $\tau^\alpha = -\tau^\gamma$ was inserted.

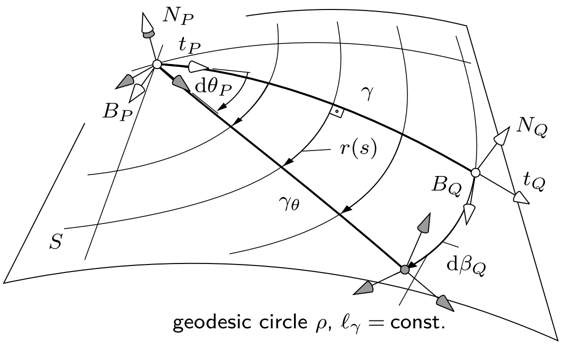

Fig. 4 A clockwise rotation $\mathrm{d}\theta$ of the geodesic's initial direction t_P about the surface normal N_P causes a binormal displacement $\mathrm{d}\beta_Q$ of the geodesic's end point trihedron $\mathcal{K}_Q$ along a geodesic circle ρ with its center at P. The relation between an arc length element $\mathrm{d}\beta_Q$ of ρ and the rotation $\mathrm{d}\theta$ is given by $r(s)$, the solution of the scalar Jacobi differential Eq. (46)

3.3 Variations of the geodesic's initial direction

When the initial direction of γ is rotated by an angular increment $\mathrm{d}\theta$, measured clockwise about the surface normal N_P, the following formulas hold for $\partial \mathcal{K}_P / \partial \theta$ (see Fig. 4)

$$\frac{\partial}{\partial \theta} \begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_P = \begin{bmatrix} 0 \\ B \\ 0 \\ -t \end{bmatrix}_P . \tag{41}$$

We are further interested in how a rotation $\mathrm{d}\theta$ affects the trihedron $\mathcal{K}_Q$. Again, we derive the necessary information about the respective variational vector field $x_\theta(s)$ from the vector Jacobi Eq. (26) and Eq. (28). First consider the behavior of x_θ at point P. By definition of the variation x_θ, it holds

$$x_\theta(0) = 0 . \tag{42}$$

Since $x'(0)$ is equal to the geodesic's tangent t_P at point P, it holds (see Eq. (41))

$$x'_\theta(0) = \frac{\partial t_P}{\partial \theta} = B_P . \tag{43}$$

Inserting Eqs. (42) and (43) into Eq. (28) yields

$$x_\theta(s) \cdot t(s) = 0, \qquad s \in [0, \ell_\gamma]. \tag{44}$$

Equation (44), known as the Lemma of Gauss, ensures that x_θ is orthogonal to the geodesic's tangent, or zero at P and points conjugate to P. Accordingly, all points on the geodesic keep a constant distance to the point P when the initial direction is rotated (see Fig. 4). The curves of constant geodesic length are called geodesic circles.

By the Lemma of Gauss (Eq. (44)), x_θ is always parallel to the geodesic's binormal, or zero. Thus x_θ can be written as

$$x_\theta(s) = r(s)\, B(s), \qquad r(s) \in \mathbb{R},\ s \in [0, \ell_\gamma], \tag{45}$$

which is analogous to Eq. (32). Inserting Eq. (45) into the vector Jacobi Eq. (26) and multiplying by $B(s)$ yields again a simplified scalar Jacobi equation

$$r''(s) + K(s)\, r(s) = 0 \tag{46}$$

analogous to Eq. (33). Solving this scalar Jacobi Eq. (46) requires the two initial conditions $r(0)$ and $r'(0)$. Noting that $\nabla B(s) = 0$ (see discussion of Eq. (35)), we have

$$r(s) = x_\theta(s) \cdot B(s), \tag{47}$$

$$r'(s) = \nabla x_\theta(s) \cdot B(s), \tag{48}$$

which, together with Eqs. (42) and (43), yield the sought initial values $r(0) = 0$ and $r'(0) = 1$. The relationship between a rotation $\mathrm{d}\theta$ and the displacement $\mathrm{d}\beta_Q$ of Q along the geodesic circle with $\ell_\gamma = \text{const.}$ is obtained by evaluating Eq. (46) at $s = \ell_\gamma$

$$\frac{\partial \beta_Q}{\partial \theta} = r(\ell_\gamma) = r_Q. \tag{49}$$

We can now compute $\partial \mathcal{K}_Q / \partial \theta$ analogously to the derivative $\partial \mathcal{K}_Q / \partial \beta_P$ described in Sect. 3.2. Let ρ denote the geodesic circle along which $\mathcal{K}_Q$ travels when the initial direction of γ is rotated. The parameterization $x(s, \theta)$ is orthogonal, thus the geodesic curvature of ρ is given by

$$\kappa_g^\rho = \frac{\mathrm{G}_s^{(s,\theta)}}{2\,\mathrm{G}^{(s,\theta)}\sqrt{\mathrm{E}^{(s,\theta)}}} = \frac{r'}{r}. \tag{50}$$

For the derivative $\partial \mathcal{K}_Q / \partial \theta$ expressed in $\mathcal{K}_Q$ it holds

$$\frac{\partial}{\partial \theta} \begin{bmatrix} x \\ t \\ N \\ B \end{bmatrix}_Q = \begin{bmatrix} 0 & 0 & r \\ 0 & r\,\tau^\gamma & r' \\ -r\,\tau^\gamma & 0 & -r\,\kappa_n^\rho \\ -r' & r\,\kappa_n^\rho & 0 \end{bmatrix}_Q \begin{bmatrix} t \\ N \\ B \end{bmatrix}_Q . \tag{51}$$

Note that for $\ell_\gamma = 0$ it holds $P = Q$. Inserting the Jacobi field's initial conditions $r(0) = 0$ and $r'(0) = 1$ into Eq. (51) confirms Eq. (41).

There is an intuitive interpretation of the quantities κ_n^ρ, κ_g^ρ, and r_Q of the geodesic circle ρ with radius ℓ_γ. While κ_n^ρ is the normal curvature of ρ required such that $\mathcal{K}_Q$ stays on the surface, κ_g^ρ is the tangential curvature of ρ required such that $\mathcal{K}_Q$ keeps a constant distance to P, i.e., stays on the geodesic circle. The quantity $r_Q = \partial \beta_Q / \partial \theta$ determines the speed of $\mathcal{K}_Q$ in terms of the rotational speed of the geodesic's initial direction.

3.4 Variation of the geodesic's length

Recall that $\mathrm{d}\ell_\gamma$ is a length increment of the geodesic for a fixed start point P. Accordingly, it holds $\mathrm{d}\ell_\gamma = \mathrm{d}s_Q$ and it follows

$$\frac{\partial \mathcal{K}_Q}{\partial \ell_\gamma} = \frac{\partial \mathcal{K}_Q}{\partial s_Q}. \tag{52}$$

4 Assembly

This section describes the structure of the global path-error Jacobian in Eq. (18), and contains the derivatives of the straight-line unit vectors in Eq. (14), which are required to complete the Jacobian. Moreover, it describes the mapping from natural geodesic corrections to a new set of geodesic parameters for the Newton steps.

4.1 Structure of the global path-error Jacobian

Due to the dependency of local path errors on the parameters of directly neighboring geodesics, the global path-error Jacobian $\mathbf{J} = \partial\varepsilon/\partial\xi$ has band structure with five lower scalar diagonals and three upper scalar diagonals

$$\mathbf{J} = \begin{bmatrix} \frac{\partial\varepsilon^1}{\partial\xi^1} & \frac{\partial\varepsilon^1}{\partial\xi^2} & & & & & \\ \frac{\partial\varepsilon^2}{\partial\xi^1} & \frac{\partial\varepsilon^2}{\partial\xi^2} & \frac{\partial\varepsilon^2}{\partial\xi^3} & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & \frac{\partial\varepsilon^i}{\partial\xi^{i-1}} & \frac{\partial\varepsilon^i}{\partial\xi^i} & \frac{\partial\varepsilon^i}{\partial\xi^{i+1}} & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \frac{\partial\varepsilon^{n-1}}{\partial\xi^{n-2}} & \frac{\partial\varepsilon^{n-1}}{\partial\xi^{n-1}} & \frac{\partial\varepsilon^{n-1}}{\partial\xi^n} \\ & & & & & \frac{\partial\varepsilon^n}{\partial\xi^{n-1}} & \frac{\partial\varepsilon^n}{\partial\xi^n} \end{bmatrix}. \tag{53}$$

A general block-row in $\mathbf{J}$ comprises three submatrices: two coupling matrices $\partial\varepsilon^i/\partial\xi^{i-1}$ if $i > 1$, $\partial\varepsilon^i/\partial\xi^{i+1}$ if $i < n$, and local path-error Jacobians $\partial\varepsilon^i/\partial\xi^i$. The jth columns ($j = 1, \dots, 4$) of the ith local path-error Jacobian and the respective coupling matrices are given by

$$\frac{\partial\varepsilon^i}{\partial\xi^i_j} = \begin{bmatrix} \frac{\partial N^i_P}{\partial\xi^i_j} \cdot e^i + N^i_P \cdot \frac{\partial e^i}{\partial\xi^i_j} \\ \frac{\partial B^i_P}{\partial\xi^i_j} \cdot e^i + B^i_P \cdot \frac{\partial e^i}{\partial\xi^i_j} \\ \frac{\partial N^i_Q}{\partial\xi^i_j} \cdot e^{i+1} + N^i_Q \cdot \frac{\partial e^{i+1}}{\partial\xi^i_j} \\ \frac{\partial B^i_Q}{\partial\xi^i_j} \cdot e^{i+1} + B^i_Q \cdot \frac{\partial e^{i+1}}{\partial\xi^i_j} \end{bmatrix}, \tag{54}$$

$$\frac{\partial\varepsilon^i}{\partial\xi^{i-1}_j} = \begin{bmatrix} N^i_P \cdot \frac{\partial e^i}{\partial\xi^{i-1}_j} & B^i_P \cdot \frac{\partial e^i}{\partial\xi^{i-1}_j} & 0 & 0 \end{bmatrix}^{\mathrm{T}}, \tag{55}$$

$$\frac{\partial\varepsilon^i}{\partial\xi^{i+1}_j} = \begin{bmatrix} 0 & 0 & N^i_Q \cdot \frac{\partial e^{i+1}}{\partial\xi^{i+1}_j} & B^i_Q \cdot \frac{\partial e^{i+1}}{\partial\xi^{i+1}_j} \end{bmatrix}^{\mathrm{T}}, \tag{56}$$

where the derivatives of N^i_P, B^i_P and N^i_Q, B^i_Q with respect to ξ^i are given in Sect. 3. We still require explicit derivatives of the straight-line unit vectors e^i and e^{i+1} with respect to ξ^{i-1}, ξ^i, and ξ^{i+1} to complete the Jacobians and the coupling matrices.

4.2 Derivatives of the unit vectors along the straight-line segments

The unit vectors e^i and e^{i+1} along the straight-line segments between two surfaces are defined as

$$e^i = \frac{x^i_P - x^{i-1}_Q}{\|x^i_P - x^{i-1}_Q\|}, \tag{57}$$

$$e^{i+1} = \frac{x^{i+1}_P - x^i_Q}{\|x^{i+1}_P - x^i_Q\|}, \tag{58}$$

where x needs to be evaluated with respect to some common coordinate system. Note that for $i = 1$ and $i = n$, the origin point O and the insertion point I are used to compute the unit vectors. The nonzero derivatives of the unit vectors in Eqs. (57) and (58) with respect to ξ^{i-1}, ξ^i, and ξ^{i+1} are given by

$$\frac{\partial e^i}{\partial \xi^i_j} = \frac{\frac{\partial x^i_P}{\partial \xi^i_j} - e^i \left(e^i \cdot \frac{\partial x^i_P}{\partial \xi^i_j} \right)}{\ell^i}, \tag{59}$$

$$\frac{\partial e^i}{\partial \xi^{i-1}_j} = \frac{-\frac{\partial x^{i-1}_Q}{\partial \xi^{i-1}_j} + e^i \left(e^i \cdot \frac{\partial x^{i-1}_Q}{\partial \xi^{i-1}_j} \right)}{\ell^i}, \tag{60}$$

$$\frac{\partial e^{i+1}}{\partial \xi^i_j} = \frac{-\frac{\partial x^i_Q}{\partial \xi^i_j} + e^{i+1} \left(e^{i+1} \cdot \frac{\partial x^i_Q}{\partial \xi^i_j} \right)}{\ell^{i+1}}, \tag{61}$$

$$\frac{\partial e^{i+1}}{\partial \xi^{i+1}_j} = \frac{\frac{\partial x^{i+1}_P}{\partial \xi^{i+1}_j} - e^{i+1} \left(e^{i+1} \cdot \frac{\partial x^{i+1}_P}{\partial \xi^{i+1}_j} \right)}{\ell^{i+1}}, \tag{62}$$

where ℓ^i is the length of the ith straight-line segment. Equations (59)–(62) can be evaluated using the derivatives of the geodesics' boundary-point trihedra given in Sect. 3. Note that Eq. (62) is equal to Eq. (59) up to the index and thus does not need to be recomputed explicitly.

4.3 Updating the geodesics' initial conditions

At a given iteration step k, the iterative Newton solver computes a global vector

$$\Delta\xi_{(k)} = -\mathbf{J}^{-1}_{(k)}\, \varepsilon_{(k)} \tag{63}$$

of finite natural path corrections (see Fig. 2) to adapt the initial conditions of all local geodesics. For each geodesic γ^i, the corrections are given by

$$\Delta\xi^i_{(k)} = \begin{bmatrix} \Delta s^i_P & \Delta\beta^i_P & \Delta\theta^i & \Delta\ell^i_\gamma \end{bmatrix}^{\mathrm{T}}_{(k)}, \tag{64}$$

which are used to compute the start-point coordinates $(u_P, v_P)^i_{(k+1)}$, the direction $(u'_P, v'_P)^i_{(k+1)}$, and the length $\ell^i_{\gamma,(k+1)}$ of the new geodesic $\gamma^i_{(k+1)}$. Note that the derivatives $(u'_P, v'_P)^i_{(k+1)}$ need to be normalized such that $\gamma^i_{(k+1)}$ has unit speed, i.e., $\|x'\|^i_{(k+1)} = 1$ (see Eq. (10)). In the following, we will discuss the parameter correction of a single geodesic, thus we will omit the index i again for notational simplicity.

The mapping from natural coordinate corrections $(\Delta s_P, \Delta\beta_P)_{(k)}$ to corrections $(\Delta u_P, \Delta v_P)_{(k)}$ requires the general coordinate transformation $\partial(u, v)/\partial(s, \beta)$. To derive it, we consider a differential dx and express it both in the surface basis $\{x_u, x_v\}$ and the natural basis $\{t, B\}$ along the geodesic, yielding

$$\mathrm{d}x = \begin{bmatrix} x_u & x_v \end{bmatrix} \begin{bmatrix} \mathrm{d}u \\ \mathrm{d}v \end{bmatrix} = \begin{bmatrix} t & B \end{bmatrix} \begin{bmatrix} \mathrm{d}s \\ \mathrm{d}\beta \end{bmatrix}. \tag{65}$$

Pairwise multiplication of Eq. (65) with x_u and x_v and solving for (du, dv) yields the coordinate transformation

$$\begin{bmatrix} \mathrm{d}u \\ \mathrm{d}v \end{bmatrix} = \underbrace{\begin{bmatrix} \mathrm{E} & \mathrm{F} \\ \mathrm{F} & \mathrm{G} \end{bmatrix}^{-1} \begin{bmatrix} t \cdot x_u & B \cdot x_u \\ t \cdot x_v & B \cdot x_v \end{bmatrix}}_{\partial(u,v)/\partial(s,\beta)} \begin{bmatrix} \mathrm{d}s \\ \mathrm{d}\beta \end{bmatrix}. \tag{66}$$

Thus for the coordinates $(u_P, v_P)_{(k+1)}$ of the new start point $P_{(k+1)}$ it holds

$$\begin{bmatrix} u_P \\ v_P \end{bmatrix}_{(k+1)} = \begin{bmatrix} u_P \\ v_P \end{bmatrix}_{(k)} + \frac{\partial(u_P, v_P)}{\partial(s_P, \beta_P)}_{(k)} \begin{bmatrix} \Delta s_P \\ \Delta\beta_P \end{bmatrix}_{(k)}. \tag{67}$$

The correction of the geodesic's initial direction requires three steps. The first step consists in rotating the tangent vector $t_{P,(k)}$ at point $P_{(k)}$ by the computed angular difference $\Delta\theta_{(k)}$. For the rotated vector $\hat{t}_{P,(k)}$ at $P_{(k)}$ it holds

$$\hat{t}_{P,(k)} = t_{P,(k)} \cos\Delta\theta_{(k)} + B_{P,(k)} \sin\Delta\theta_{(k)}. \tag{68}$$

In the second step, $\hat{t}_{P,(k)}$ is parallel transported to the new point $P_{(k+1)}$ with a constant orientation in space according to the Levi-Civita parallelism [54]. At the new point, $\hat{t}_{P,(k)}$ is not tangential to the surface anymore, thus it has two tangential components with magnitudes μ, ν, and one normal component with magnitude λ

$$\hat{t}_{P,(k)} = \begin{bmatrix} x_u & x_v & N \end{bmatrix}_{P,(k+1)} \begin{bmatrix} \mu \\ \nu \\ \lambda \end{bmatrix}. \tag{69}$$

Scalar multiplication of Eq. (69) with $(x_u)_{P,(k+1)}$ and $(x_v)_{P,(k+1)}$ yields

$$\begin{bmatrix} \hat{t}_{P,(k)} \cdot (x_u)_{P,(k+1)} \\ \hat{t}_{P,(k)} \cdot (x_v)_{P,(k+1)} \end{bmatrix} = \begin{bmatrix} E & F \\ F & G \end{bmatrix}_{P,(k+1)} \begin{bmatrix} \mu \\ \nu \end{bmatrix} \tag{70}$$

and thus for the tangential components μ and ν of $\hat{t}_{P,(k)}$ at $P_{(k+1)}$ it holds

$$\begin{bmatrix} \mu \\ \nu \end{bmatrix} = \begin{bmatrix} E & F \\ F & G \end{bmatrix}^{-1}_{P,(k+1)} \begin{bmatrix} \hat{t}_{P,(k)} \cdot (x_u)_{P,(k+1)} \\ \hat{t}_{P,(k)} \cdot (x_v)_{P,(k+1)} \end{bmatrix}. \tag{71}$$

The third step consists in normalizing μ and ν such that $\gamma_{(k+1)}$ has unit speed. The new components $u'_{P,(k+1)}$ and $v'_{P,(k+1)}$ that yield a unit-speed geodesic starting at $P_{(k+1)}$ are given by

$$\begin{bmatrix} u'_P \\ v'_P \end{bmatrix}_{(k+1)} = \left(\frac{1}{\mathrm{E}\,\mu^2 + 2\,\mathrm{F}\,\mu\,\nu + \mathrm{G}\,\nu^2} \right)_{P,(k+1)} \begin{bmatrix} \mu \\ \nu \end{bmatrix}. \tag{72}$$

Finally, the computation of the new length $\ell_{\gamma,(k+1)}$ is straightforward

$$\ell_{\gamma,(k+1)} = \ell_{\gamma,(k)} + \Delta\ell_{\gamma,(k)}. \tag{73}$$

5 Rate of length change

In the previous sections, it was assumed that all surfaces S^i as well as the origin point O and insertion point I are fixed. In this section, we also consider the spatial motion of all objects and derive an explicit formula for the path's exact rate of length change. To this end, we assume that the global path error is exactly zero so $\varepsilon = 0$ at any instant of time, i.e., the considered path is always a global geodesic between O and I.

The total muscle path is a concatenation of $n + 1$ straight-line segments of length ℓ^i and n geodesic segments of length ℓ^i_γ. Hence its total length is given by

$$L = \sum_{i=1}^{n+1} \ell^i + \sum_{i=1}^{n} \ell^i_\gamma. \tag{74}$$

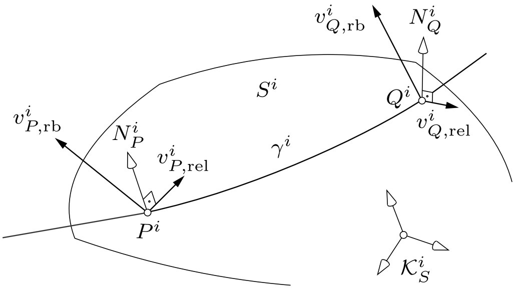

Fig. 5 Decomposition of the absolute boundary-point velocities v^i_P and v^i_Q into two components: absolute rigid-body velocities $v^i_{P,\mathrm{rb}}$, $v^i_{Q,\mathrm{rb}}$ of the surface at P^i and Q^i and relative boundary-point velocities $v^i_{P,\mathrm{rel}}$, $v^i_{Q,\mathrm{rel}}$ of the geodesic's boundary points with respect to the surface

Accordingly, the rate of length change $\dot{L}$ of the total path is obtained as the sum of the rates $\dot{\ell}^i$ and $\dot{\ell}^i_\gamma$ of the single elements, where the dot notation denotes a time derivative. For a straight-line segment, the rate of length change is the projection of its absolute end-point velocities v^{i-1}_Q and v^i_P, i.e., measured with respect to an inertially fixed frame, on the unit vector e^i

$$\dot{\ell}^i = e^i \cdot \left[v^i_P - v^{i-1}_Q\right]. \tag{75}$$

For a geodesic segment, we obtain the rate of length change from the first variation of its arc length. Let $x(s,\epsilon)$ be a geodesic variation such that $\gamma : x(s,0)$ is a geodesic (index i omitted here). The first variation of arc length of γ is given by

$$\delta\ell_\gamma = \left[x' \cdot \delta x_\epsilon\right]_0^{\ell_\gamma} - \int_0^{\ell_\gamma} x'' \cdot \delta x_\epsilon \,\mathrm{d}s, \tag{76}$$

where δx_ϵ are arbitrary vectors tangential to S. Since the variation is a geodesic, $x''(s,0)$ is normal to S and the integral in Eq. (76) vanishes, yielding for the remaining terms

$$\delta\ell_\gamma = x'(\ell_\gamma) \cdot \delta x_\epsilon(\ell_\gamma) - x'(0) \cdot \delta x_\epsilon(0). \tag{77}$$

Substituting $\delta\ell_\gamma = \mathrm{d}\ell_\gamma$, $x'(\ell_\gamma) = t_Q$, $\delta x_\epsilon(\ell_\gamma) = \mathrm{d}x_Q$, $x'(0) = t_P$, $\delta x_\epsilon(0) = \mathrm{d}x_P$, and dividing by $\mathrm{d}t$ gives an explicit expression for the rate of length change of the ith geodesic segment

$$\dot{\ell}^i_\gamma = t^i_Q \cdot v^i_{Q,\mathrm{rel}} - t^i_P \cdot v^i_{P,\mathrm{rel}}. \tag{78}$$

Here, $v^i_{P,\mathrm{rel}}$ and $v^i_{Q,\mathrm{rel}}$ are the boundary-point velocities relative to the surface S^i, i.e., measured with respect to the surface frame $\mathcal{K}^i_S$ (see Fig. 5). The relation between the absolute boundary-point velocities in Eq. (75) and the relative components in Eq. (78) is given by

$$v^i_P = v^i_{P,\mathrm{rb}} + v^i_{P,\mathrm{rel}}, \tag{79}$$

$$v^i_Q = v^i_{Q,\mathrm{rb}} + v^i_{Q,\mathrm{rel}}, \tag{80}$$

where $v^i_{P,\mathrm{rb}}$ and $v^i_{Q,\mathrm{rb}}$ are the absolute rigid-body velocities of the surface beneath the geodesic segment at P^i and Q^i (see Fig. 5).

After inserting Eqs. (75) and (78) into the time derivative of Eq. (74), also using Eqs. (79) and (80) as well as the relations $t^i_P = e^i$ and $t^i_Q = e^{i+1}$ (which hold for the solution path with $\varepsilon = 0$), all terms comprising the relative velocities $v^i_{P,\mathrm{rel}}$ and $v^i_{Q,\mathrm{rel}}$ cancel pairwise out. Consequently, the total rate of length change becomes

$$\dot{L} = -e^1 \cdot v_O + \left(\sum_{i=1}^{n} e^i \cdot v^i_{P,\mathrm{rb}} - e^{i+1} \cdot v^i_{Q,\mathrm{rb}}\right) + e^{n+1} \cdot v_I, \tag{81}$$

where v_O and v_I are the absolute velocities of the origin point O and the insertion point I of the muscle.

Equation (81) shows that the rate of length change $\dot{L}$ depends solely on the rigid-body velocities of the surface points beneath the geodesics' boundary points and the velocities of the origin and insertion points. Equation (81) is independent of the formulation used to compute the path itself and computationally inexpensive, making a numerical differentiation of path length unnecessary.

6 Implementation and computational benchmarks

We have implemented the Natural Geodesic Variation (NGV) method in the C++ multibody simulation environment Mobile [55]. Despite the extensive use of differential geometry, the method's algorithmic implementation is surprisingly straightforward. It comprises

1. A set of smooth surface objects, each capable of calculating elementary differential-geometric entities such as the surface normal vector, first and second fundamental forms, normal curvature, geodesic torsion, and Gaussian curvature;
2. A geodesic shooter that computes one geodesic and the two associated Jacobi fields on each surface given the geodesic's starting point, direction, and length. The shooter also computes the Darboux trihedra and their derivatives at the geodesic's boundary points with respect to the four natural geodesic variations;
3. A global equation assembler that computes (a) the straight-line segments to evaluate the path error using the Darboux trihedra, and (b) the banded path-error Jacobian explicitly using the derivatives of the Darboux trihedra and the derivatives of the straight-line unit vectors;
4. An iterative solver which computes a set of natural geodesic corrections to correct the parameters of all geodesics until the path error vanishes and the shortest path is found; and
5. An explicit evaluator for the path's rate of length change, if required.

Accuracy, ability to wrap around complex geometry, and computational speed of the method have been tested in three simulation benchmarks which cover scalable and nontrivial applications. The benchmarks were performed on an Intel® Core™ i7-2700K @3.50 GHz with 8 GB RAM running Windows 7 Professional, 64 bit. We used an LSodar integrator with relative and absolute tolerances of 1.0E-9 to numerically solve the geodesic Eqs. (2), (3) and the Jacobi field Eqs. (33), (46). Lapack's band matrix routine DGBSV was used to solve the linear equation system for the Newton iterations. In all benchmark simulations, we used the solution parameters of a previous time step as the initial guess for the current time step for the Newton iterations, with manually determined initial conditions at the start (see Sect. 7).

The first benchmark evaluates the accuracy of our NGV method and compares it with a nonlinear energy-minimization approach. The second benchmark contains multiple complex wrapping obstacles including a surface patch fitted to a human ribcage to evaluate our method's ability to wrap over biologically accurate surfaces. In the third benchmark, we measured the computational costs of our method with respect to the number of wrapping surfaces.

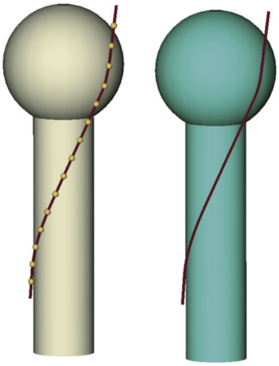

Fig. 6 Simple upper limb model: (*left*) the path was computed by a nonlinear optimization routine which minimizes the sum of squared energies of the path segments (here: $m = 15$ vertices); (*right*) the path was computed by our NGV method. It consists of three straight lines and two local surface geodesics

6.1 Accuracy benchmark

We simulated a widely used sphere-cylinder upper-limb model (see, e.g., [39]) to evaluate the accuracy of our algorithm when computing muscle paths, their lengths, and their rates of length change. In our benchmark, the model consists of a sphere with radius 8.0 cm, a cylinder with radius 4.0 cm, and a muscle path which wraps over both surfaces (Fig. 6). The cylinder is rigidly attached to the sphere such that the sphere's center lies in the symmetry axis of the cylinder. The model performed spatial sinusoidal rotations about the center of the sphere while the path's origin was fixed with respect to ground and the path's insertion point had a constant position with respect to the cylinder. A constant output sampling step size of 0.02 s was used. The final simulation time was 10.0 s.

For comparison, we implemented an established energy minimization approach, e.g., used by [37, 39]. It consists in discretizing the whole muscle path by a variable number m of lumped elastic elements and minimizing the sum of squared energies subject to unilateral surface constraint equations. The shortest path is computed by finding the $3m$ coordinates of all m path vertices such that the path's energy is minimal, and no vertex penetrates any surface. We provided both the gradient of the path energy as well as the gradient of the surface constraints explicitly. The optimization routine used was NAG e04unc (tolerance set to 1.0E-9) with disabled command window outputs.

In this benchmark, we applied exactly two Newton iterations per time step resulting in a maximal path error of 2.04E-9 with an average of 1.45E-10. When a single Newton step was used, the maximal path error was 1.31E-5 with an average of 4.69E-6. The example shows that our method solves the shortest path problem with a very high precision in a few iterations. Accordingly, it is well-suited to compute accurate and smooth solutions for the muscle's path length (Fig. 7(a)) and its rate of length change (Fig. 7(c)) using Eq. (81). In contrast, the energy minimization approach shows approximate and oscillatory behavior (see Fig. 7(b, d)) as it approximates the exact shortest path by a polyline. The energy minimization method requires a fine path discretization to compute accurate solutions for the path's length and its rate of length change (see Fig. 7(b, d)). Independent of the level of path discretization, energy minimization yields oscillations in the path's rate of length change which needs to be computed from finite differences of path length (see Fig. 7(d)). In our implementation, the energy minimization method was orders of magnitudes slower than the NGV method, depending on the level of path discretization. Further details on the computational speed of our method are given in Sect. 6.3.

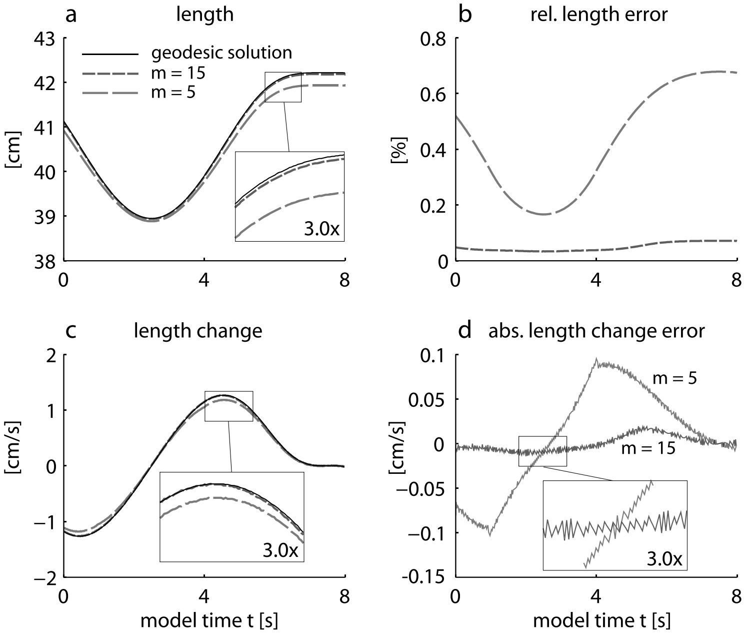

Fig. 7 Qualitative comparison between our NGV method and the energy minimization method using $m = 5$ and $m = 15$ path vertices. Our method computes smooth and accurate solutions for the path's length and the rate of length change. The errors in path length and rate of length change of the energy minimization method decrease with increasing level of path discretization but the method yields nonsmooth behavior of the path's rate of length change independently of the level of discretization

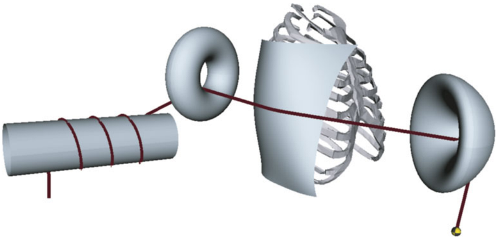

Fig. 8 Force-driven wrapping over nonsimple surfaces, including a surface patch fitted to a ribcage. Our method provides solution continuity as demonstrated by wrapping the frictionless path multiple times over the cylinder, and allows for wrapping over biologically accurate surfaces

6.2 Complex geometry benchmark

The second benchmark is a force-driven dynamic simulation including four (nonsimple) wrapping surfaces: a simple cylinder, a torus, a polynomial surface patch, and an elliptic torus (see Fig. 8) with high curvature. The surface patch was fitted to the geometry of human a ribcage using a MATLAB surface fitting routine to demonstrate our method's ability to wrap over anatomically realistic, general surfaces. The simulation time was 5.0 s with a constant output sampling step size of 0.02 s. At the path's end point, a freely moving point mass is attached. During simulation, the path was continuously wrapped three times over the cylinder to demonstrate that our method ensures solution continuity. Recall here that we assumed a frictionless contact between the path and the objects. The torus was moved vertically up and down to slide the path across the ribcage patch (see Fig. 8).

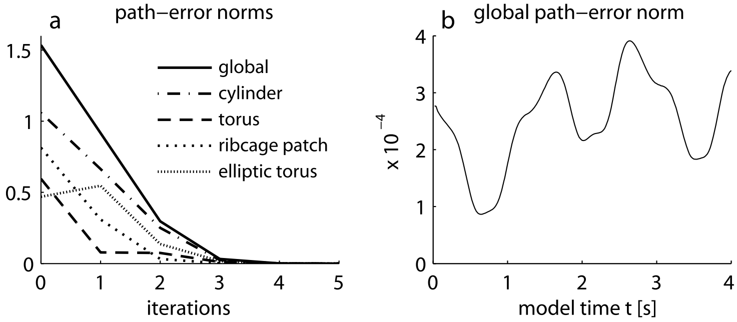

Fig. 9 (*Left*) Convergence behavior within the first five iterations at initialization of the complex geometry benchmark. All local path errors converge to zero, with quadratic convergence behavior near the solution. (*Right*) Time history of the global path error norm during the simulation, where the path-error tolerance was set to 1.0E-3. During the simulation, the maximal path error was 3.93E-4 with an average of 2.87E-4

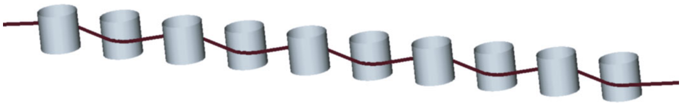

Fig. 10 Wrapping over a variable number of cylinders. Our method computes high-precision solutions for shortest paths over a large number of wrapping surfaces in real time

Before simulation start, we chose unfeasible initial conditions resulting in a global path-error norm of $\|\varepsilon\| = 1.53$. Seven Newton iterations were used to solve for a feasible path (until $\|\varepsilon\| = 8.72$E-14) at the beginning of the simulation. Figure 9(a) shows the convergence behavior of the global path error, and the local path error norm of each surface within the first five iterations at initialization. Newton's method ensures that the path-error converges quadratically to zero near the solution. For the rest of the simulation, the path-error tolerance was set to 1.0E-3. Figure 9(b) displays the resulting time history of the global path-error norm with a maximum of 3.93E-4 and an average of 2.87E-4.

6.3 Computational speed benchmark

Simulating anatomically complex parts of the body such as the spine or the hand may require a large number of wrapping surfaces. In this benchmark, we evaluated our method's computational speed with respect to the number of wrapping surfaces. To this end, we wrapped a single path over a variable number of cylinders which performed spatial sinusoidal translations and rotations while the path's origin and insertion points were fixed. The total simulation time was 40.0 s with a constant output sampling step size of 0.02 s. Figure 10 shows the benchmark setup for $n = 10$ cylinders.

The computational costs of the NGV method majorly depend on solving the geodesic Eqs. (2), (3) together with the Jacobi field Eqs. (33), (46), on computing the nonzero elements of the path-error Jacobian, and on solving the linear equation system for the Newton iterations. The computational costs for the individual subtasks of our method grow linearly with the number of wrapping obstacles, making it well suited for simulating muscle paths over many surfaces. On simple surfaces such as for the cylinders presented in this benchmark, the geodesic equations and the Jacobi field equations can be solved analytically. We evaluated the method's computational costs for both solving these equations numerically

Table 1 Computational costs of the NGV method vs. the number of wrapping surfaces, when analytic and numeric solutions for the geodesics and the Jacobi field equations are used

Number of cylinders	Real-time factor	
	Analytic	Numeric
100	0.05	0.12
200	0.11	0.24
300	0.16	0.36
400	0.22	0.49
500	0.29	0.62
1000	0.61	1.34

and analytically. Table 1 displays the computational costs for both cases vs. the number of surfaces, showing that (a) the computational costs of our method grow linearly with the number of surfaces, and (b) our method allows for wrapping over a large number of surfaces in real time. In our current implementation, the numeric solution is about two times slower than the analytic solution.

For this benchmark, we applied a single Newton iteration per simulation time step. For 1000 cylinders using analytic solutions for the geodesics, the maximal path-error norm during the whole simulation was 5.4E-4, showing that the path stays close to the exact solution even when many surfaces are involved.

7 Discussion

The NGV method is based on the modeling assumption that muscles and tendons behave like one-dimensional massless strings that take the locally shortest path from origin to insertion. This premise makes it necessary to lump the masses of muscles with body segments, which causes errors when simulating the dynamics of musculoskeletal systems, as shown by Pai [56]. Despite their limitations, however, such curved-line muscles are used today in the majority of models. Even though they are relatively simple, they can adequately recreate experimentally measured moment arms, as shown, for instance, by Gatti et al. [40].

Since the NGV method is based on Newton's method, it requires an initialization before simulation start. The goal of this initialization is to find a reasonable set of geodesics such that the resulting candidate path converges to the solution. For the benchmark simulations shown in Sect. 6, we performed the initialization manually. This is a straightforward process which has to be performed only once per model, but an automated initialization that takes user-defined wrapping directions into account remains a desirable future extension. After successful initialization it suffices to use the solution of a previous time step as the initial guess for the current time step. This procedure was used for all benchmarks in this paper as it leads to a one to two-step convergence at every simulation time step. Future extensions might also incorporate velocity constraints for the shortest path to reduce the number of iterations further.

Muscle paths computed by the NGV method are locally length minimizing, but no information about their global optimality is provided. This limitation, however, does not restrict the method's use in musculoskeletal models. Quite the contrary, the local convergence together with the feedback of previous solutions ensures that simulated muscle paths evolve continuously as the musculoskeletal system moves (see Fig. 8). A further limitation of the current version of the NGV method is that it does not permit path lift-off and touch-down events. Though such contact events are rare in biological systems due to connective tissue and small relative motion between muscles and wrapping objects, they can happen in musculoskeletal models that contain thin curved-line muscles. We are currently working on an extension of the NGV method which allows for simulating lift-off and touch-down.

Though all derivations within this paper are based on parametric surfaces, the NGV method allows for incorporating implicit surfaces as well. This is possible because all intrinsic surface quantities, such as Gaussian curvature, normal curvature, or geodesic torsion, can also be evaluated explicitly on implicit surfaces. In this setting, a comparison between both surface representations remains a desirable goal for future investigations.

This paper contains the results of three simulation benchmarks that were designed to allow for an assessment of the method's generality, solution quality, and absolute computational speed. These benchmarks indicate that the NGV method is very fast, yet they do not yield a one-to-one performance comparison with other muscle wrapping methods, such as the energy minimization method which we implemented for the accuracy benchmark in Sect. 6.2. This would require carefully chosen benchmark setups and well developed software implementations of the compared methods in the same software package. For this reason, this paper contains no performance comparison.

8 Conclusions

This paper introduced the Natural Geodesic Variation (NGV) method for the very efficient and fast computation of shortest paths between two points across multiple nonsimple geometric objects, such as curved-line muscle paths in musculoskeletal models. The method is based on finding the root of a global path-error constraint equation with a gradient-based root-finding method using an explicitly determined banded Jacobian. Additionally, a general formula to compute the shortest path's exact rate of length change was introduced, making an approximate computation from finite differences unnecessary.

In contrast to existing methods, the NGV method is both general and fast. It does not rely on special-case solutions, has no nested loops, and can handle a large number of nonsimple wrapping surfaces, such as surface patches fitted to bone geometry, in real time. It computes time-continuous, geometrically smooth, high-precision solutions for the shortest path, its length, and its rate of length change, which lie within the accuracy of machine precision. The excellent computational efficiency is achieved by completely taking advantage of the intrinsic differential geometry of surfaces and geodesics, which allows for computing the path-error Jacobian for the root-finding method explicitly.

According to the extensive use of differential geometry, the theoretical derivations of the method are more involved compared to existing methods, yet the final method's implementation into biomechanics software packages is surprisingly straightforward. Since this paper focuses on the method's mathematical foundations, we have employed standard solvers for the ODEs and the Newton iterations. We expect that the efficiency can be further improved by using more advanced numerical tools for these tasks.

Because of its speed and its generality, we believe that the NGV method will become a useful tool for solving muscle wrapping problems irrespectively of the number and the shape of the contact surfaces. It will allow biomechanists to develop more accurate, yet fast musculoskeletal models.

Acknowledgements The authors gratefully acknowledge the support of Leonidas Guibas, Adrian Butscher, and Justin Solomon when discussing Jacobi fields; Matthew Millard for his valuable feedback on the method, the manuscript and the figures; and Francisco Geu Flores for his support in benchmarking.

References

1. Arnold, A.S., Blemker, S.S., Delp, S.L.: Evaluation of a deformable musculoskeletal model for estimating muscle-tendon lengths during crouch gait. Ann. Biomed. Eng. **29**, 263–274 (2001)
2. Kerr, G.H., Selber, P.: Musculoskeletal aspects of cerebral palsy. J. Bone Jt. Surg.—Br. Vol. **85**(2), 157–166 (2003)
3. Arnold, A.S., Anderson, F.C., Pandy, M.G., Delp, S.L.: Muscular contributions to hip and knee extension during the single limb stance phase of normal gait: a framework for investigating the causes of crouch gait. J. Biomech. **38**, 2181–2189 (2005)
4. Hicks, J.L., Schwartz, M.H., Arnold, A.S., Delp, S.L.: Crouched postures reduce the capacity of muscles to extend the hip and knee during the single-limb stance phase of gait. J. Biomech. **41**, 960–967 (2008)
5. Steele, K.M., Seth, A., Hicks, J.L., Schwartz, M.S., Delp, S.L.: Muscle contributions to support and progression during single-limb stance in crouch gait. J. Biomech. **43**, 2099–2105 (2010)
6. Neptune, R.R., Kautz, S.A., Zajac, F.E.: Contributions of the individual ankle plantar flexors to support, forward progression and swing initiation during walking. J. Biomech. **34**, 1387–1398 (2001)
7. Zajac, F.E., Neptune, R.R., Kautz, S.A.: Biomechanics and muscle coordination of human walking. Part 1: introduction to concepts, power transfer, dynamics and simulations. Gait Posture **16**, 215–232 (2002)
8. Zajac, F.E., Neptune, R.R., Kautz, S.A.: Biomechanics and muscle coordination of human walking. Part 2: lessons from dynamical simulations and clinical implications. Gait Posture **17**, 1–17 (2003)
9. Liu, M.Q., Anderson, F.C., Pandy, M.G., Delp, S.L.: Muscles that support the body also modulate forward progression during walking. J. Biomech. **39**, 2623–2630 (2006)
10. van der Krogt, M.M., Delp, S.L., Schwartz, M.H.: How robust is human gait to muscle weakness? Gait Posture **36**, 113–119 (2012)
11. Neptune, R.R., Sasaki, K.: Ankle plantar flexor force production is an important determinant of the preferred walk-to-run transition speed. J. Exp. Biol. **208**, 799–808 (2005)
12. Hamner, S.R., Seth, A., Delp, S.L.: Muscle contributions to propulsion and support during running. J. Biomech. **43**, 2709–2716 (2010)
13. van der Helm, F.C.T.: The shoulder mechanism: a dynamic approach. Ph.D. Thesis. Delft University of Technology (1991)
14. Yu, J., Ackland, D.C., Pandy, M.G.: Shoulder muscle function depends on elbow joint position: an illustration of dynamic coupling in the upper limb. J. Biomech. **44**, 1859–1868 (2011)
15. Sasaki, K., Neptune, R.R.: Individual muscle contributions to the axial knee joint contact force during normal walking. J. Biomech. **43**, 2780–2784 (2010)
16. Lin, Y.-C., Walter, J.P., Banks, S.A., Pandy, M.G., Fregly, B.J.: Simultaneous prediction of muscle and contact forces in the knee during gait. J. Biomech. **43**, 945–952 (2010)
17. Winby, C.R., Lloyd, D.G., Besier, T.F., Kirk, T.B.: Muscle and external load contribution to knee joint contact loads during normal gait. J. Biomech. **42**, 2294–2300 (2009)
18. Moissenet, F., Chèze, L., Dumas, R.: A 3D lower limb musculoskeletal model for simultaneous estimation of musculo-tendon, joint contact, ligament and bone forces during gait. J. Biomech. (2013)
19. Giat, Y., Mizrahl, J., Levine, W.S., Chen, J.: Simulation of distal tendon transfer of the biceps brachii and the brachialis muscles. J. Biomech. **27**(8), 1005–1014 (1994)
20. Hill, A.V.: The mechanics of active muscle. Proc. - Royal Soc., Biol. Sci. **141**, 104–117 (1953)
21. Gordon, A.M., Huxley, A.F., Julian, F.J.: The variation in isometric tension with sarcomere length in vertebrae muscle fibers. J. Physiol. **184**, 170–192 (1966)
22. Bahler, A.S., Fales, J.T., Zierler, K.L.: The dynamic properties of mammalian skeletal muscle. J. Gen. Physiol. **51**, 369–384 (1968)
23. Zajac, F.E.: Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Crit. Rev. Biomed. Eng. **17**(4), 359–411 (1989)
24. Millard, M., Uchida, T., Seth, A., Delp, S.L.: Flexing computational muscle: modeling and simulation of musculotendon dynamics. J. Biomech. Eng. **135**, 021004-1–021004-11 (2013)
25. Brand, R.A., Crowninshield, R.D., Wittstock, C.E., Pedersen, D.R., Clark, C.R., van Krieken, F.M.: A model of lower extremity muscular anatomy. J. Biomech. Eng. **104**(4), 304–310 (1982)
26. Ackland, D.C., Pandy, M.G.: Lines of action and stabilizing potential of the shoulder musculature. J. Anat. **215**, 184–197 (2009)
27. Blemker, S.S., Asakawa, D.S., Gold, G.E., Delp, S.L.: Image-based musculoskeletal modeling: applications, advances, and future opportunities. J. Magn. Reson. Imaging **25**, 441–451 (2007)
28. Webb, J.D., Blemker, S.S., Delp, S.L.: 3D finite element models of shoulder muscles for computing lines of actions and moment arms. Comput. Methods Biomech. Biomed. Engin., 1–9 (2012)
29. An, K.-A., Berglund, L., Cooney, W.P., Chao, E.Y.S., Kovacevic, N.: Direct in vivo tendon force measurement system. J. Biomech. **23**(12), 1269–1271 (1990)

30. Schuind, F., Garcia-Elias, M., Cooney, W.P. III, An, K.-N.: Flexor tendon forces: in vivo measurements. J. Hand Surg. **17**(2), 291–298 (1992)
31. Garner, B.A., Pandy, M.G.: The obstacle-set method for representing muscle paths in musculoskeletal simulations. Comput. Methods Biomech. Biomed. Eng. **3**, 1–30 (1999)
32. Charlton, I.W., Johnson, G.R.: Application of spherical and cylindrical wrapping algorithms in a musculoskeletal model of the upper limb. J. Biomech. **34**, 1209–1216 (2001)
33. Gao, F., Damsgaard, M., Rasmussen, J., Christensen, S.T.: Computational method for muscle-path representation in musculoskeletal models. Biol. Cybern. **87**, 199–210 (2002)
34. Marai, G.E., Laidlaw, D.H., Demiralp, C., Andrews, S., Grimm, C.M., Crisco, J.J.: Estimating joint contact areas and ligaments lengths from bone kinematics and surfaces. IEEE Trans. Biomed. Eng. **51**(5), 790–799 (2004)
35. Carman, A.B., Milburn, P.D.: Dynamic coordinate data for describing muscle-tendon paths: a mathematical approach. J. Biomech. **38**, 943–951 (2005)
36. Blemker, S.S., Delp, S.L.: Rectus femoris and vastus intermedius fiber excursions predicted by three-dimensional muscle models. J. Biomech. **39**, 1383–1391 (2006)
37. Marsden, S.P., Swailes, D.C., Johnson, G.R.: Algorithms for exact multi-object muscle wrapping and application to the deltoid muscle wrapping around the humerus. Proc. Inst. Mech. Eng., H J. Eng. Med. **222**(7), 1081–1095 (2008)
38. Röhrle, O., Davidson, J.B., Pullan, A.J.: Bridging scales: a three-dimensional electromechanical finite element model of skeletal muscle. SIAM J. Sci. Comput. **30**(6), 2882–2904 (2008)
39. Audenaert, A., Audenaert, E.: Global optimization method for combined spherical-cylindrical wrapping in musculoskeletal upper limb modeling. Comput. Methods Programs Biomed. **92**, 8–19 (2008)
40. Gatti, C.J., Hughes, R.E.: Optimization of muscle wrapping objects using simulated annealing. Ann. Biomed. Eng. **37**(7), 1342–1347 (2009)
41. Vasavada, A.N., Lasher, R.A., Meyer, T.E., Lin, D.C.: Defining and evaluating wrapping surfaces for MRI-derived spinal muscle paths. J. Biomech. **41**, 1450–1457 (2008)
42. Arnold, E.M., Ward, S.R., Lieber, R.L., Delp, S.L.: A model of the lower limb for analysis of human movement. Ann. Biomed. Eng. **38**(2), 269–279 (2010)
43. Esat, I.I., Ozada, N.: Articular human joint modeling. Robotica **28**, 321–339 (2010)
44. Favre, P., Gerber, C., Snedeker, J.G.: Automated muscle wrapping using finite element detection. J. Biomech. **43**, 1931–1940 (2010)
45. Spyrou, L.A., Aravas, N.: Muscle-driven finite element simulation of human foot movements. Comput. Methods Biomech. Biomed. Eng. **5**(9), 925–934 (2012)
46. Stavness, I., Sherman, M., Delp, S.L.: A general approach to muscle wrapping over multiple surfaces. Florida, USA, 2012. Proc. Amer. Soc. Biomech. (2012)
47. Scholz, A., Stavness, I., Sherman, M., Delp, S.L., Kecskeméthy, A.: Improved muscle wrapping algorithms using explicit path-error Jacobians. Barcelona, Spain, 2012. Comput. Kinematics. (2012)
48. Desailly, E., Sardain, P., Khouri, N., Yepremian, D., Lacouture, P.: The convex wrapping algorithm: a method for identifying muscle paths using the underlying bone mesh. J. Biomech. **43**, 2601–2607 (2010)
49. Struik, D.J.: Lectures on Classical Differential Geometry. Dover, New York (1988)
50. Strubecker, K.: Differentialgeometrie Band 3: Theorie der Flächenkrümmung. de Gruyter, Berlin (1969)
51. do Carmo, M.P.: Differential Geometry of Curves and Surfaces. Prentice Hall, New York (1976)
52. Pressley, A.: Elementary Differential Geometry. Springer, Berlin (2010)
53. do Carmo, M.P.: Riemannian Geometry. Birkhäuser, Basel (1993)
54. Strubecker, K.: Differentialgeometrie Band 2: Theorie der Flächenmetrik. de Gruyter, Berlin (1969)
55. Kecskeméthy, A., Hiller, M.: An object-oriented approach for an effective formulation of multibody dynamics. Comput. Methods Appl. Math. **115**(3–4), 287–314 (1994)
56. Pai, D.K.: Muscle mass in musculoskeletal models. J. Biomech. **43**(11), 2093–2098 (2010)