Contact detection between curved fibres: high order makes a difference

Octave Crespel, Emile Hohnadel, Thibaut Métivet, Florence Bertails-Descoubes

Contact detection between curved fibres: high order makes a difference

OCTAVE CRESPEL*, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, France
EMILE HOHNADEL*, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, France
THIBAUT MÉTIVET, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, France
FLORENCE BERTAILS-DESCOUBES, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, France

Fig. 1. A wavy hair wisp made of 2025 super-helices is virtually combed, yielding a hundred thousand tight frictional contacts ($\mu = 0.1$). In this paper we show that during such a sliding motion, the force response is highly dependent on the proxy elements used for *collision detection*. In particular, we find that even in the case of a rod model with a C^1-smooth centreline, segment-based collision detection yields artificial noise in the contact force response, which increases with the curvature of the rod at each contact point. We introduce a new detection scheme between two smooth curves that fixes this issue. Compared to a classical segment-based detection scheme (right plot, black curve), our method retrieves force profiles that integrate much less noise (yellow curve), hence increasing the accuracy of force prediction.

Computer Graphics has a long history in the design of effective algorithms for handling contact and friction between solid objects. For the sake of simplicity and versatility, most methods rely on low-order primitives such as line segments or triangles, both for the detection and the response stages. In this paper we carefully analyse, in the case of fibre systems, the impact of such choices on the retrieved contact forces. We highlight the presence of artifacts in the force response that are tightly related to the low-order geometry used for contact detection. Our analysis draws upon thorough comparisons between the high-order super-helix model and the low-order discrete elastic rod model. These reveal that when coupled to a low-order, segment-based detection scheme, both models yield spurious jumps in the contact force profile. Moreover, these artifacts are shown to be all the more visible as the geometry of fibres at contact is curved. In order to remove such artifacts we develop an accurate high-order detection scheme between two smooth curves, which relies on an efficient adaptive pruning strategy. We use this algorithm to detect contact between super-helices at high precision, allowing us to recover, in the range of wavy to highly curly fibres, much smoother force profiles during sliding motion than with a classical segment-based strategy. Furthermore, we show that our approach offers better scaling properties in terms of efficiency vs. precision compared to segment-based approaches, making it attractive for applications where accurate and reliable forces are desired. Finally, we demonstrate the robustness and accuracy of our fully high-order approach on a challenging hair combing scenario.

CCS Concepts: • **Computing methodologies** → **Animation**; **Physical simulation**.

Additional Key Words and Phrases: High-order contact detection, Dry frictional contact, Kirchhoff thin elastic rod, Super-Helices

ACM Reference Format:
Octave Crespel, Emile Hohnadel, Thibaut Métivet, and Florence Bertails-Descoubes. 2024. Contact detection between curved fibres: high order makes a difference. *ACM Trans. Graph.* 43, 4, Article 132 (July 2024), 23 pages. https://doi.org/10.1145/3658191

*Both authors contributed equally to this research.

Authors' addresses: Octave Crespel, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000, Grenoble, France, octave.crespel@inria.fr; Emile Hohnadel, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000, Grenoble, France, emile.hohnadel@inria.fr; Thibaut Métivet, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000, Grenoble, France, thibaut.metivet@inria.fr; Florence Bertails-Descoubes, Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000, Grenoble, France, florence.descoubes@inria.fr.

0730-0301/2024/7-ART132
https://doi.org/10.1145/3658191

1 INTRODUCTION

In physics-based animation, accounting for contact and friction between moving solid bodies is essential to convey visual realism. Computer Graphics has dedicated more than thirty years of research to devising effective methods for both contact detection [Teschner et al. 2005] and contact response [Baraff 1994; Bridson et al. 2002; Daviet et al. 2011; Goldenthal et al. 2007; Kaufman et al. 2008], which are nowadays well-settled in the film and video game industries.

Essentially driven by visual applications, the community has often favoured the development of low-order and position-based simulation primitives in order to accelerate the process of collision detection and develop simple and fast response strategies. However, with the recent fostering of new applications mixing virtual and real scenarios, such as special effects [Museth 2020], sound generation [An et al. 2012; Schweickart et al. 2017], virtual try-on [Bartle et al. 2016; Clegg et al. 2015; Erickson et al. 2017], or virtual prototyping [Martínez et al. 2019; Tricard et al. 2020], there is nowadays an increasing demand for high realism in the predictions made by the physics simulators, not only regarding the resulting geometry – important for the *visual* evaluation of an object – but also regarding the resulting contact forces – essential for the *feeling, stability, or feasibility* of a physical system. These new considerations call for a profound revision of past choices merely focused on producing virtual objects with a plausible appearance.

In this paper, we show that provided a high-order collision detection strategy is set up, high-order fibre models yield a better response in terms of both geometry and forces than low-order models when coupled to an accurate frictional contact solver. To support this claim, our analysis proceeds in two steps. First, relying upon thorough comparisons between the (high-order) super-helix model [Bertails et al. 2006] and the (low-order) discrete elastic rod model [Bergou et al. 2008], we show that when coupled to a low-order detection scheme, both models generate spurious artifacts in the contact force profile that amplify as the fibre curvature at contact increases. We then develop a geometrically exact helix-helix collision detection scheme, which allows us to recover smoother and more physically accurate force profiles when coupled to the high-order super-helix model. We showcase the advantages of this new fully high-order pipeline by validating the computed forces in a controlled curved three point bending scenario before ramping up the complexity by considering the combing of realistic hair wisps with various natural curvatures, which highlights the robustness of our approach.

2 RELATED WORK

Visually correct collision detection is key to the realism of physics simulations. In both interactive and baked scenarios, preventing interpenetration gives a sense of concreteness to the objects involved. Frictional contacts are also crucial to achieving immersive experiences, both in terms of geometry and dynamics, by including physical dissipation into interactions. Collisions must therefore be handled accordingly, and significant efforts have gone towards making reliable collision detection computationally tractable.

Collision detection pipelines usually require three components [Ericson 2004]:

(1) A *broad phase*, which first prunes out non-colliding primitive pairs with simple space partitioning and region overlap tests. As such, this phase drastically decreases the computational load when large scenes are involved.
(2) A *representation* of the geometry under consideration, approximated at the appropriate level of detail.
(3) A *narrow phase*, which computes the contact regions between two shapes, along with all the required information (e.g. normals, penetration depth, local coordinates).

When high precision is required, these last two points are in practice responsible for most of the algorithmic complexity of the overall collision detection, and must therefore carefully balance the ratio between cost and accuracy to maintain robust yet tractable simulations.

Contact detection between surfaces. In the case of contacting surfaces, triangle meshes are often used to parametrise the geometry, as they provide convenience and versatility. The same representation, although at varying levels of detail, can be seamlessly used for simulation, rendering, and collision detection. Parametric or implicit shape descriptions are more rarely used in Computer Graphics, but techniques do exist to handle these (see e.g. [Teschner et al. 2005] for a review). For example, Von Herzen et al. [1990] use kd-trees in parametric space to find collisions between two surfaces, and accelerate the traversal of these kd-trees using first-order information about the surfaces. Similarly, Snyder et al. [1993] describe an algorithm to find multiple collision points between a pair of deforming surfaces by refining parameter-space intervals using the jacobian of the gap function. Regarding curves, Lakshmanan et al. [2019] propose a routine to compute curve-curve distances by subdividing parametric curves and bounding sub-curves with ellipsoids using first-order information, and suggest using this routine for collision detection.

Low-order contact detection between curves. Efficient geometric representations are all the more crucial in the case of thin rod structures which easily deform when subjected to contact, and for which robustness with respect to non-penetration is crucial to prevent instabilities and tunnelling. In Computer Graphics, most numerical models for thin elastic rods are built using a segment-based discretisation of the curve geometry [Bergou et al. 2008; Hadap 2006; Selle et al. 2008; Spillmann and Teschner 2007]. As such they naturally resort to segment-based proximity queries, which benefit from simple closed forms for the distance computations. To ensure that collision detection does not introduce convergence issues or incorrect results, it is noteworthy that "contact smoothing" has started being investigated in Mechanics for low-order primitives (see e.g. section 8.5 of [Wriggers and Laursen 2006]), a research direction that is orthogonal to ours.

Still, a few higher-order simulation primitives have been proposed for elastic rods, such as spline-based models [Nocent and Remion 2001] or curvature-based models [Bertails et al. 2006; Casati and Bertails-Descoubes 2013]. However, those usually employ an additional low-order, segment-based representation to perform collision detection. In this paper we show that resorting to a low-order detection scheme substantially deteriorates the quality of the contact force signal, even in the presence of a high-order simulation primitive.

High-order contact detection between curves. In Mechanics, Isogeometric analysis (IGA), as introduced by [Hughes et al. 2005], builds the simulated primitives directly on top of CAD data and provides stronger convergence guarantees than disjointed representations for simulation and detection. De Lorenzis et al. [2014] review the use of IGA in contact simulation, and notably mention that resorting to higher-order primitives can help remove some spurious

oscillations. However, their focus is put on the continuity of the normal and tangent fields, which is not the cause of the artifacts we identify and remove, as detailed in section 4. In the particular case of IGA applied to rods, Meier et al. [2017] propose a local Newton algorithm coupled to specific segment-based initial guesses, which allows for high-order closest point queries between cubic curves. While our method also falls into the category of – isogeometric – contact solvers, it is more generic than Newton-based approaches, and naturally handles configurations where the collision detection amounts to a non-convex optimisation problem, as discussed below.

In Graphics, Marschner et al. [2021] have recently introduced a new framework for high-order geometry processing, which is expanded upon in a follow-up paper ([Zhang et al. 2023]). Among other problems, this *sum-of-squares geometry processing* approach solves continuous collision detection between high-order (polynomial) patches. Although Zhang et al. [2023] greatly improve the performance of the approach, it remains computationally expensive and relies on existing general-purpose optimisation packages to find solutions. Besides, the sum-of-squares programming framework is restricted to polynomial approximation, and as such only applies to polynomial patches in the context of geometry processing, while the algorithm proposed in our work addresses the case of arbitrary curves with a bounded second derivative.

One should also mention the complementary Eulerian on Lagrangian (EoL) approach, previously devised in Mechanics as an Arbitrary Lagrangian Eulerian (ALE) framework [Pechstein and Gerstmayr 2013; Vu-Quoc and Li 1995], and extended in the Graphics community to handle contacting rods [Cirio et al. 2014; Sánchez-Banderas et al. 2020; Sueda et al. 2011] or solids [Fan et al. 2013]: in order to bypass the difficulties pertaining to contact detection, these approaches directly represent contact points with additional Eulerian degrees of freedom, and naturally incorporate the contact response into the overall dynamics of the system. While this naturally ensures that the motion at contact points is compatible with the underlying geometry, it is however restricted in essence to persistent contacts, and changes in the contacting set must resort to standard detection algorithm in order to update the set of EoL points.

The impact of spurious noise in contact forces was also analysed in the work of Zheng and James [2011], where the contact *response* is adjusted to remove an important source of noise in sound synthesis, related to the non-uniqueness of contact force solutions. However, their approach does not consider the detailed role of contact *detection* and its effect on the resulting contact force signal, even in the case when contact forces are spatially and temporally well-defined. Our work is complementary to this study, and aims at addressing *geometrical* discontinuities at the detection level.

To our knowledge, the effect of using low-order collision primitives on the resulting force signal has not been examined in the Computer Graphics community, which generally focuses its analysis on geometry rather than forces. In this work, we identify and analyse force discontinuities caused by low-order collision detection, and fix the problem by computing the distance between parametric C^2-smooth curves, devising an efficient branch-and-bound like iterative procedure and leveraging second-order information about the curves to accelerate inner iterations.

3 PROBLEM AND CONTRIBUTIONS

As a first step to motivate the forthcoming developments, we focus on an elementary hair combing scenario featuring a single hair fibre (see fig. 2, top). Our goal here is not only to look at the resulting geometry of the fibre, but also and especially at the *force* applied by the comb onto the fibres. Predicting accurately contact forces emerging from fibres would be crucial for many applications ranging from reliable physical measurements in cosmetology to immersive haptic feedback within a virtual hair salon, and even realistic sound generation from bowed strings. Besides, inaccurate forces may lead to unphysical oscillations or convergence issues, and may end up influencing the geometry itself.

To maximise our chances to achieve a realistic prediction, we perform the virtual comb experiment using graphics tools known for their robustness and physical reliability, as they have been extensively validated in the past on challenging scenarios [Romero et al. 2021]. On the one hand, we rely on the *Discrete Elastic Rod* model [Bergou et al. 2010] coupled with the `so-bogus` frictional contact solver [Daviet et al. 2011] through an adaptive nonlinear scheme [Kaufman et al. 2014]. We directly use the free `creamystrand` library released by Fei et al. [2019]. On the other hand, we use the *Super-Helix* model [Bertails et al. 2006], which code has been provided by the authors, again coupled with `so-bogus`.

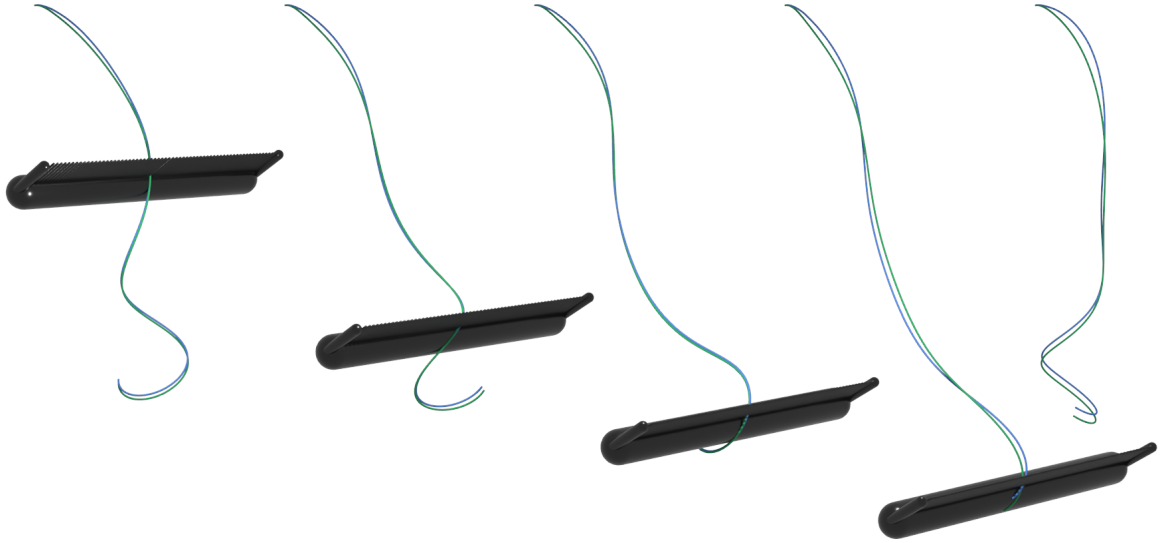
The results of our virtual experiment are shown in fig. 2 (bottom). Unfortunately, it is clear that they look flawed, for both simulators. Indeed we observe spurious jumps in the central force signal[1], even in the *Super-Helix* case, where the centreline is smooth. Worse, the noise increases with time, i.e. with the indentation of the comb. Since the fibre portion which is in contact with the comb gets more and more curved as the comb travels through the wisp, one can reasonably suspect that the noise is amplified by the *curvature* of the fibre at the contact point.

What is the origin of these jumps? Could it be that they come from physical events, e.g. some kind of stick-slip behaviour? A lack of fibre resolution? The frictional contact solver? The algorithm for collision detection?

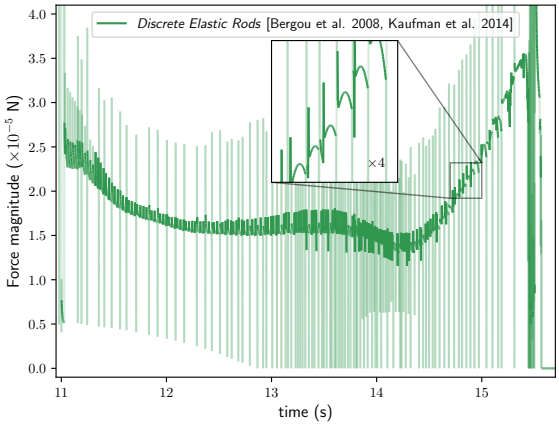
In this paper, we demonstrate that *the choice of low-order proxys for collision detection* is the *very cause* of these jumps, and we fix this issue in the case of smooth rod models such as *Super-Helices*. More precisely, our contributions are three-fold:

- In section 4 we carefully analyse the origin of the jumps in the force signal during sliding motion on a model case, namely the curved three-point bending test, for which we derive an analytic solution. In particular we find out that the jumps stem directly from the low-order proxy elements used for collision detection, and that they worsen as the curvature of the fibre at contact points increases. Our study is carefully led on two popular rod models, *Discrete Elastic Rods* and *Super-Helices*.
- In section 5 we devise a new detection scheme for rod models described by a smooth centreline, with no need for low-order

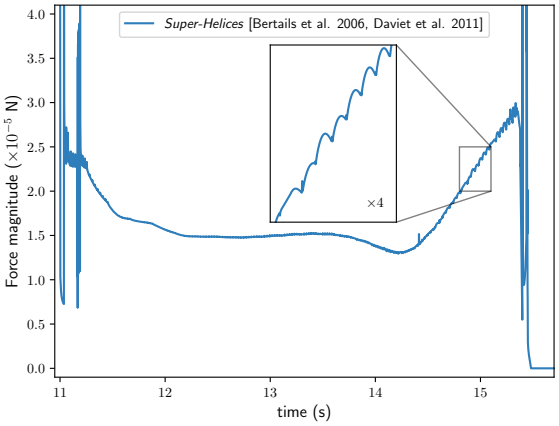
[1]Here, we do not focus on another source of noise, which yields the very high amplitude discontinuities well above and below the main, central force signal. This additional noise, especially visible on *DER*, is due to isolated (i.e. for one time-step) losses of contact which appear as unfortunate consequences of the constraint-based contact response which is not activated immediately as the gap function becomes positive, possibly as a result of numerical approximations.



(a) Combing a wavy hair fibre, either with a *Discrete Elastic Rod* (in green) or a *Super-Helix* (in blue)



(b) Comb force onto the *Discrete Elastic Rod*



(c) Comb force onto the *Super-Helix*

Fig. 2. Noisy contact forces: Simulating the combing of a wavy hair fibre using graphics tools dedicated to physical realism... and yet, discovering the noisy, unphysical aspect of the resulting forces. In green we use the *Discrete Elastic Rod* model [Bergou et al. 2010] coupled with the so-bogus frictional contact solver [Daviet et al. 2011] through an adaptive nonlinear scheme [Kaufman et al. 2014]. In blue we use the *Super-Helix* model [Bertails et al. 2006] coupled with so-bogus [Daviet et al. 2011]. Albeit yielding apparent consistent geometries (top), both simulations hide in reality fairly noisy contact forces (bottom), whose noise increases with the curvature of the fibre at contact point. The origin of this noise, as we demonstrate in this paper, is *directly due* to the low-order proxy geometry (segments) used for collision detection.

proxys such as segments. Our method works for any pair of smooth curves whose second derivative is bounded, including helices and splines. Furthermore, compared to the standard approach of using a fixed-resolution segment discretisation it scales favourably in terms of arc-length resolution vs. runtime.

- In section 6 we apply our new detection algorithm to the *Super-Helix* model and show that it removes the force artifacts identified previously. We demonstrate the relevance of our method on complex and large-scale hair combing simulations featuring 2025 hairs with various natural curliness, and tens of thousands of frictional contact points on average. On such a large-scale scenario, the classical segment-based detection method causes significant drift in the resulting contact force due to an accumulation of noise on every fibre, which is removed by our method. Finally (section 7), our algorithm is shown to apply onto polynomial curves as well and to

compare favourably to recent spline-based detection methods of the literature.

4 ANALYSIS OF THE CONTACT FORCE JUMPS ON THE CURVED THREE-POINT BENDING TEST

Our goal in this section is to *explain* and *quantify* the jumps observed previously in the contact force signal from a hair strand during sliding motion. To this aim we consider the well-known three-point rod bending test [Frisch-Fay 1962] for which an analytic force solution can be computed. We furthermore extend this test to the *naturally curved* case in order to analyse the influence of the curvature of the rod at contact points on the force profile.

4.1 Protocols

Classical three-point bending test. The three-point rod bending test is a simple and canonical 2D mechanical protocol used to probe the bending stiffness of a (naturally straight) rod. It is usually performed experimentally by quasi-statically indenting a rod placed on two horizontal supports spaced by some distance Δ, while measuring the resulting vertical force F applied onto the indenting body as a function of the indentation (c.f. fig. 3). To further simplify the analysis, we do not simulate the indenting body, but equivalently enforce horizontal clamping of the rod in the middle, and account for indentation by lifting the supports. The supports are represented as disks of radius R, with $R \ll \Delta$; each of them is in contact with the rod at a single contact point. In practice, as the problem is symmetric, we consider only one half of the system (in red in fig. 3), and measure the contact force f on the support to compute the equivalent load as $F = -2f_y$, where f_y is the vertical component of f. Let Δ the distance between the two supports, and $s \in [0, \frac{L}{2}]$ the arclength along the half rod, where L is the total length of the rod. We denote by $s_c \in [\frac{\Delta}{2}, \frac{L}{2}]$ the arclength of the contact point with the support, which varies with the indentation δ.

Curved three-point bending test. We augment the test by adding a uniform *natural curvature* κ_0 to the rod. Our resulting *curved* three-point bending test is illustrated in fig. 4. Note that due to the absence of gravity and to the particular boundary conditions of the three-point bending test (no force nor torque applied beyond the contact point), the curvature $\kappa(s)$ of the rod at the contact point s_c and beyond is theoretically equal to the natural curvature κ_0 of the rod. In other words, the rod describes a circular arc of radius $\frac{1}{\kappa_0}$ for $s \geq s_c$. Varying the natural curvature κ_0 of the rod hence directly controls the value of the curvature $\kappa(s_c)$ at contact, and allows to study the influence of this curvature on the force response. Note that numerically, this equivalence only holds at the limit $n_{\text{elts}} \to \infty$, where n_{elts} is the number of elements of the rod model. In practice we checked that our choice of n_{elts} (see table 2) ensures a good approximation of the limit case, for both rod models.

The whole set of parameters used in our protocol is summarised in table 1 (note that for visualisation purposes, the supports have been magnified in fig. 4 and fig. 3 compared to their size in the simulations, taken identical to the radius of the rod). To allow for proper indentation of the rod (i.e. the contact force f between the rod and the support should be activated), κ_0 has to be chosen below the tangent limit $\frac{2}{\Delta - R - r}$, where R is the radius of the support and

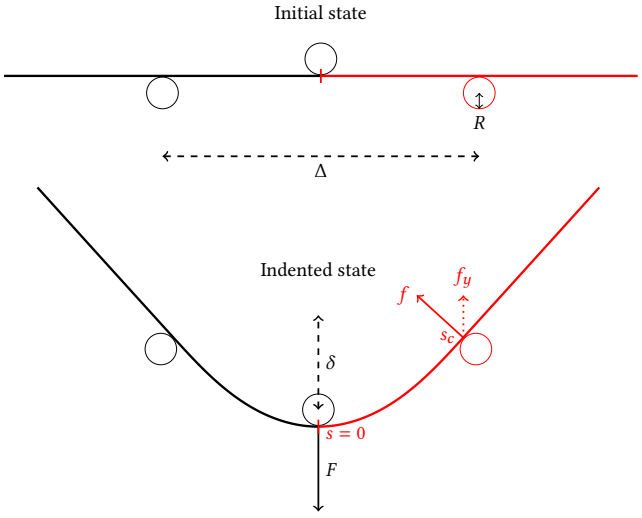


Fig. 3. Classical three-point bending test setup. In our simulations, we consider for simplicity the equivalent "half-system" with an horizontal clamp in the middle, as shown in red. Indentation is obtained by lifting the right-most support.

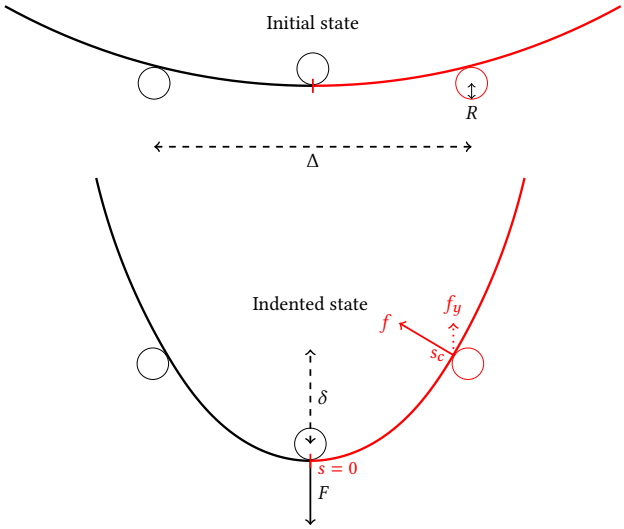


Fig. 4. Our curved three-point bending test setup, where the rod may have some non-vanishing natural curvature κ_0, chosen between 0 and $\frac{2}{\Delta - R - r}$, where R is the radius of the support.

r is the radius of the rod . In practice we vary κ_0 between 0 and $0.3\,\text{cm}^{-1} < \frac{2}{\Delta - R - r} \approx 0.4\,\text{cm}^{-1}$.

Analytic model. We first focus on the (classical) case where the rod is naturally straight ($\kappa_0 = 0$). In the – linear – small deflection regime, the flexural test provides direct measurement of the bending modulus $B \equiv EI$ of the rod through the linear relation between the load F and the indentation δ,

$$\bar{\delta} \equiv \frac{\delta}{\Delta} = \frac{F\Delta^2}{48B} \equiv \bar{F}.$$

Rod radius r	1.85×10^{-4} m
Distance between supports Δ	5×10^{-2} m
Support radius R	1.85×10^{-4} m
Natural curvature κ_0	$[0, 0.1, 0.2, 0.3]\,\mathrm{cm}^{-1}$
Density ρ	$6.45 \times 10^{3}\,\mathrm{kg\,m}^{-3}$
Young modulus E	83 GPa
Poisson ratio ν	0.33
Indentation velocity v	$5 \times 10^{-3}\,\mathrm{m\,s}^{-1}$
Maximal indentation $\delta_{\max}$	2×10^{-2} m

Table 1. Parameters used in our (curved) three-point bending protocol. Contact is assumed to be frictionless. Note that the lifting velocity v of the support is chosen so as to be much smaller than the characteristic flexural wave velocity $\frac{r}{\Delta}\sqrt{\frac{E}{\rho}}$ to ensure quasi-static indentation.

The barred variables are dimensionless quantities which allow for equations that are independent of the problem's specific physical parameters.

When the rod is thin enough to satisfy the Kirchhoff assumptions (i.e. no stretching, nor cross-section shearing), this relation can be analytically extended to the large deflection regime [Batista 2015; Frisch-Fay 1962]. In the case of supports without friction, this yields the following implicit relation:

$$\forall \alpha \in \left[0; \frac{\pi}{2}\right], \quad \begin{cases} \bar{\delta} = \bar{D}\,(1 - \cos(\alpha)) + \left(\dfrac{1}{2} - \bar{D}\sin(\alpha)\right) \dfrac{I_2(\alpha)}{I_1(\alpha)} \\ \bar{F} = \dfrac{\cos(\alpha)}{\left(1 - 2\bar{D}\sin(\alpha)\right)^2}\, I_1(\alpha)^2 \end{cases} \tag{1}$$

with $\bar{D} = \frac{r+R}{\Delta}$ the normalised distance between the center of the obstacle and the centerline of the rod, and

$$I_1(\alpha) \equiv \int_0^\alpha \frac{\cos(\alpha - \theta)}{\sqrt{\sin(\theta)}}\, d\theta$$
$$I_2(\alpha) \equiv \int_0^\alpha \frac{\sin(\alpha - \theta)}{\sqrt{\sin(\theta)}}\, d\theta.$$

In practice, both integrals are evaluated numerically using the *scipy* library and a pair $(\bar{\delta}, \bar{F})$ is deduced for a range of known values of α.

When the rod is naturally curved, the formulas for computing the exact contact force become more intricate. We provide them in appendix A.

4.2 Numerical setting

To demonstrate the impact of collision detection across different rod models, we consider the three model variants described hereafter.

Numerical models of rods and contact. We study the contact force response from two popular and very different 3D discrete rod models from the Computer Graphics literature, possibly coupled with different proxy elements for detection:

- the low-order *Discrete Elastic Rod* model (*DER-segments*) from Bergou et al. [2010, 2008] coupled with the segment-based collision detection method from Kaufman et al. [2014] implemented in the open-source `creamystrand` software [Fei et al. 2019], that we have improved to also handle contacts between rods and rigid cylinders;
- the curvature-based *SuperHelix* model from Bertails et al. [2006], coupled either with a standard segment-based proxy for collision detection [Daviet et al. 2011] (*SH-segments*, original code provided by the authors), or with the high-order helix-based algorithm that we introduce in this paper (*SH-helices*, **ours**). We recall that *Super-Helices* discretise 3D Kirchhoff rods using N elements $i \in \{1, ..., N\}$ with uniform material curvatures and twist $\boldsymbol{\kappa}_i$ along their constant length l_i, resulting in a G^1-smooth, piecewise helical centreline. Given appropriate boundary conditions (e.g. setting the position and material frame at one end of the rod), and given the continuity of the curve and the attached material frame at element junctions, the $\boldsymbol{\kappa}_i$ vectors define the centreline and material frame of the rod unequivocally. From this geometric description, a reduced-coordinate model is built. The equations of motion take the form of a linear system $M\ddot{\boldsymbol{q}} = \boldsymbol{f}$, where M is a dense $3N \times 3N$ mass matrix, $\boldsymbol{q} = (\boldsymbol{\kappa}_1, ..., \boldsymbol{\kappa}_N)$ is the state vector, and $\boldsymbol{f}$ is a generic reduced force term which includes inertial forces, internal elastic forces, as well as external forces such as the weight of the rod. Time-stepping the simulation involves solving this linear problem, possibly under non-linear frictional contact constraints. See [Bertails et al. 2006] and [Daviet et al. 2011] for more details.

-	n_{elts}	n_{prox}	dt (s)
DER-segments (100)	100	100	1×10^{-4}
SH-segments	20	100	1×10^{-4}
SH-helices **(ours)**	20	20	1×10^{-4}

Table 2. Three-point bending numerical parameters. Number of elements n_{elts} (segments for *DER*, helices for *Super-Helices*), number of collision proxy elements n_{prox} segments for *DER* and *SH-helices*, helices for *SH-segments*) and time-step dt used for the three-point bending simulations.

Note that in all three cases, contact response is handled with the same non-smooth constraint-based frictional contact solver from [Daviet et al. 2011] implemented in the `so-bogus` library. This allows us to focus our study on the exact influence of the order of the rod model and detection scheme used. Note that the general conclusions drawn below also hold in the case of smoother contact responses, as shown in appendix C, where the three-point bending test is simulated with the Incremental Potential Contact method from Li et al. [2020a]. The parameters used for each simulation are summarised in table 2. They are non-dimensionalised in the simulations in order to prevent round-off errors and ensure numerical stability.

Computing the normals at contact. We always consider that a contact involves no more than two bodies. In the ideal continuous setting, two smooth contacting objects define a unique tangent plane, hence a unique contact normal. In our case however, due to time-stepping, contact may be detected once objects have slightly

interpenetrated, which makes the contact normal more delicate to estimate and leaves us with several options. By testing a few of them, we have noticed that such a choice is actually critical in the way the contact force response is computed. Too approximate a normal estimation may artificially amplify the jumps we observe. Worse, it may cause the loss of contact and further interpenetration. To avoid such issues while remaining consistent over the two numerical rod models that we use, we choose to systematically compute the contact normal as the cross product between tangents of the two objects at the contact point, should the objects in contact be rods or obstacles. With this choice, the normal is guaranteed to be orthogonal to both objects' surfaces, even though objects may not be touching at a single point.

4.3 Force results

Discovering jump points. For each value κ_0 ranging in $[0, 0.1, 0.2, 0.3]$ we perform the three-point bending test and report the (dimensionless) contact force $\bar{F}$ as a function of the (dimensionless) indentation $\bar{\delta}$. Results are shown in fig. 5 for both *DER-segments* (left, green) and *SH-segments* (right, blue). Both models use the same number of proxy segments ($n_{\text{prox}} = 100$) for collision detection. The absolute force curves (top) match analytic curves (dotted black curves) very well for *SH-segments*, and fairly well for *DER-segments*. However, as κ_0 increases (darker curves) we note the presence of jumps in the force signal. These are particularly visible in the *DER-segments* plots, but they are actually also present in the *SH-segments* plots albeit less directly visible. To better visualise these jumps, we normalise each force curve by its analytic counterpart (bottom). In both cases one now clearly sees jumps, which occur with a larger amplitude in *DER-segments* than in *SH-segments*, but at the same constant frequency in both models. As shown in appendix B, the jumps appear exactly when a node of the segment-based detection proxy comes in contact with the obstacle.

One can also note by comparing the *DER-segments (100)* and the twice as refined *DER-segments (200)* curves (respectively, the *SH-segments (100)* and the refined *SH-segments (200)* curves) shown in appendix B that the force jumps induced at the junctions between segments do not disappear as the number of segments is increased, but occur twice as often, while producing smaller force oscillations. Despite the decrease in discontinuities achieved by increasing the number of segments, we observe that the convergence rate is very low (see fig. 20), with jumps amplitudes still much larger than the numerical noise even with 200 or 400 elements in the *DER-segments* case. Segment-based refinement is particularly detrimental for fully low-order models such as *DER-segments*, for which increasing the number of segments not only affects collision detection but also the mechanical model itself, thereby strongly raising the overall computational cost.

Last but not least, the natural curvature κ_0 (equivalently the curvature $\kappa(s_c)$ at the contact point) turns out to play a major role in the jump amplification.

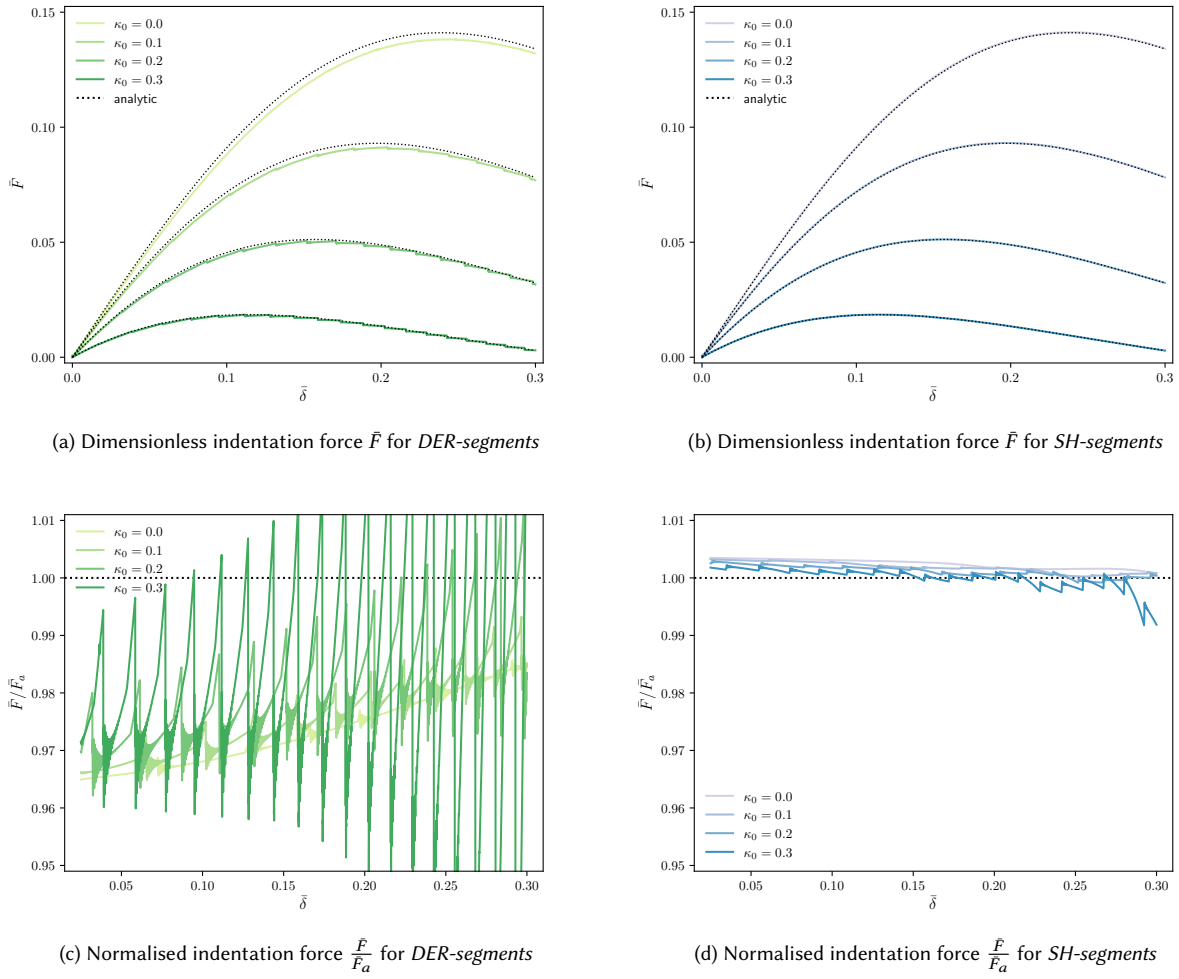
Quantifying the jumps w.r.t. curvature. To quantify more precisely the role of the curvature $\kappa(s_c)$ at the contact point, in fig. 6 we plot the L_1 norm (bottom) of the finite difference derivative of the force-indentation curves (top). Remarkably, and for both models *DER-segments* and *SH-segments*, this analysis reveals a *linear dependence* of the jump size with respect to the curvature at the contact point.

Rationale behind the force jumps. The observed force jumps can be related to the low-order approximate geometry used to detect contacts, as illustrated in fig. 7 and explained in detail in our accompanying video. Imagine a fibre sliding upon an external body (which can be another fibre or a rigid body) at some contact point (denoted in orange in fig. 7. If this exact theoretical contact point lies anywhere in the normal cone at the intersection of the two segments used as detection proxy elements for the fibre, it gets projected onto the same curve abscissa (in green), thereby leading to incorrect contact force locations, and possible impacts when the contact leaves the cone. These impacts, regularly spaced according to the length of the segment proxy, match the "jumps" that appear with constant frequency in the force signals of the three-point bending test. Furthermore, note that even when the contact point lies outside the junction normal cone, the error between the exact location of the contact point and its projection onto the segment proxy gets amplified all the more so as the rod centreline is curved near the contact location, hence leading to larger jumps.

On the contrary, exact contact detection with high-order simulated rods would correctly project any contact point onto the point closest to it *on the centreline* (in pink), leading to smooth contact evolution. Although similar artefacts have been detected in the numerical mechanics literature (see e.g. [De Lorenzis et al. 2014]), the proposed solution is to ensure that the normal and tangent fields vary smoothly over the contacting bodies, especially at element boundaries. This is not sufficient however, as demonstrated by our *SH-segments* scenario. In that setting, normals are smooth along the entire fiber (i.e. in space), but force discontinuities remain because the contact detection scheme ends up sampling this smooth normal field at a sequence of abscissae that is itself not smooth with respect to the actual sliding motion. In other words, our contribution is to identify and fix an artificial lack of smoothness *with respect to the sliding motion* (i.e. in time) rather than only with respect to the surface.

What we propose to fix. In this paper we focus on high-order fibre models, whose centreline is a smooth curve, and propose an efficient high-order detection algorithm between two such smooth curves. Specifically, our method avoids having to resort to low-order proxy elements with nonsmooth junctions (e.g. segments) and directly builds upon the smooth geometry of the fibre centreline. We apply our new algorithm to the *Super-Helix* model, which features a C^1-smooth centreline made of helical arcs that are tangentially connected. Our new algorithm is presented in the next section.

As a conclusion to this analysis, in fig. 8 we provide a synthetic view of our findings on the curved three-point bending test, for a rod featuring a large natural curvature $\kappa_0 = 0.3$. The plot shows the force-indentation curves obtained for all three simulators, including our new *SH-helices* method (described in the following section), compared with the smooth analytic solution. While the forces computed by the three simulators are for the most part in very good agreement with the analytic model, we retrieve the localised force



(a) Dimensionless indentation force $\bar{F}$ for *DER-segments*

(b) Dimensionless indentation force $\bar{F}$ for *SH-segments*

(c) Normalised indentation force $\frac{\bar{F}}{\bar{F}_a}$ for *DER-segments*

(d) Normalised indentation force $\frac{\bar{F}}{\bar{F}_a}$ for *SH-segments*

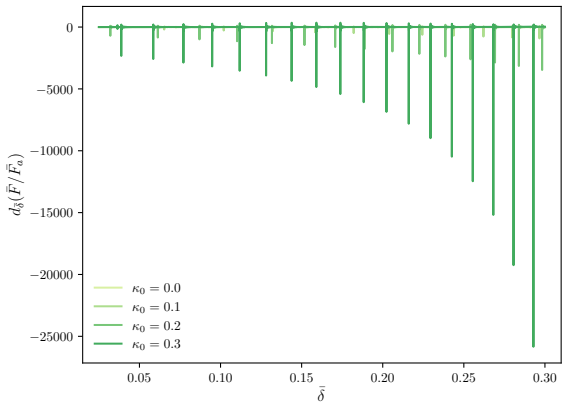
Fig. 5. Curved three-point bending test: noisy force results. Top: Force-indentation curves for *DER* (left, green) and *SH-segments* (right, blue), as the natural curvature κ_0 is increased (light to dark). Bottom: Normalised curves to better highlight the presence of force jumps in both models, due to segment-based detection.

jumps already observed earlier for the two segment-based detection scenarios (severe jumps for *DER-segments* (in green), visible at full scale, and moderate jumps for *SH-segments* (in blue), revealed by zooming in), which precisely match the geometrical discontinuities at the junctions between the segments used for collision detection. On the contrary, in anticipation to the next section, we can see that our new high-order detection scheme coupled with the smooth *Super-Helix* rod model (in red) completely fixes the jump issue and yields a nice, smooth force-indentation curve even when largely zoomed in.

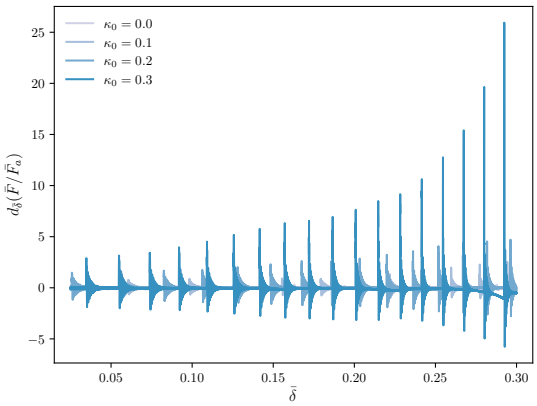
5 HIGH-ORDER CONTACT DETECTION

Detecting contact points between two shapes $A, B \subset \mathbb{R}^d$ amounts to building a numerical representation of the set $A \cap B$ from numerical representations of A and B. In the following, we focus on the case of fibres with constant thickness, so that A and B can be represented as the ϵ-neighbourhoods of the image of finite intervals by some parametric curves: $A = \{\boldsymbol{p}\, s.t.\, \exists s \in \mathcal{I}_A,\ \|\boldsymbol{a}(s) - \boldsymbol{p}\| \leq \epsilon\}$, where $\boldsymbol{a}$ defines the parametric curve in question, and similarly for B.

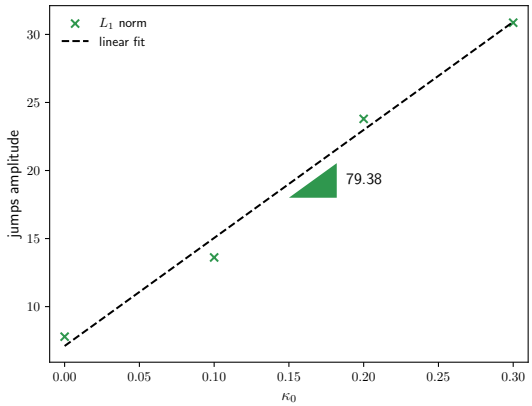
In this case, computing $A \cap B$ may be framed as computing a number of contact regions S in the parametric space $\mathcal{I}_A \times \mathcal{I}_B$ such that $\forall (s_A, s_B) \in S,\ \|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\| \leq r_A + r_B$. Although these regions may be sampled more or less densely, as is the case in [Snyder et al. 1993], we instead yield a single contact "point" (s_A, s_B) per curve pair, which corresponds to the pair (s_A, s_B) at which the distance $\|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\|$ is minimal. That is, our collision problem amounts to a *closest point query* between two parametric curves. More contact points may be added by subdividing the curves prior to running the collision algorithm. All contact points are then used to build the



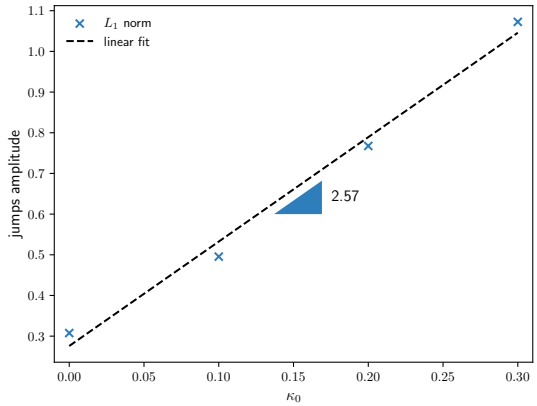
(a) Force derivative for *DER-segments*



(b) Force derivative for *SH-segments*



(c) L_1 norm of derivative force for *DER-segments*



(d) L_1 norm of derivative force for *SH-segments*

Fig. 6. Curved three-point bending test: analysis of the force discontinuities. Top: Derivatives of the force-indentation curves for *DER* (left, green) and *SH-segments* (right, blue), as the natural curvature κ_0 is increased (light to dark). Bottom: L_1 norm of the above curves depict the linear dependency of jump size with respect to the curvature at contact.

system of equations to be solved (as described in [Daviet et al. 2011] for our applications).

Producing a single contact point reduces the collision detection problem to a generally non-convex global optimisation problem of two variables (the two parameters along the curves), whose objective function is

$$F : (s_A, s_B) \mapsto \|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\|^2,$$

the pointwise distance between the two curves at the given parameters.

As such, our algorithm takes inspiration from the global optimisation literature, and especially from so-called *branch and bound* methods (see e.g. the first chapters of [Scholz 2011] for a review). The starting intervals $(\mathcal{I}_A, \mathcal{I}_B)$ are recursively subdivided until either

(1) the desired precision δ is reached, i.e.

$$\max(|\mathcal{I}_A|, |\mathcal{I}_B|) \leq \delta,$$

(2) or the shapes A and B can be proven not to intersect, i.e.

$$\forall s_A \in \mathcal{I}_A, \forall s_B \in \mathcal{I}_B, \|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\| > r_A + r_B.$$

In order to fulfil the second criterion, more information needs to be gathered regarding the two curves. In particular, given two bounding volumes V_A and V_B such that $\boldsymbol{a}(\mathcal{I}_A) \subseteq V_A$ and $\boldsymbol{b}(\mathcal{I}_B) \subseteq V_B$, we have $d(\boldsymbol{a}(\mathcal{I}_A), \boldsymbol{b}(\mathcal{I}_B)) \geq d(V_A, V_B)$ where $d(X, Y) = \min_{x \in X, y \in Y} \|x - y\|$ denotes the minimum distance between two point sets. Criterion 2 may therefore be checked by building a bounding volume for each piece of curve, and checking for

$$d(V_A, V_B) > r_A + r_B. \tag{2}$$

Hence, bounding volumes for which this distance may be easily computed must be constructed.

At its simplest, such a volume may for instance be a sphere centred at the midpoint of the curve, with a radius equals to half its length,

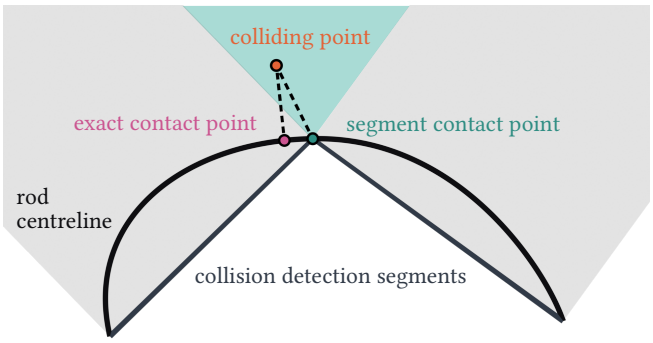


Fig. 7. Segment-based detection generates spurious contact discontinuities: as a potential point contacting the rod (in orange) enters the normal cone (highlighted in green) at the junction between collision detection segments, it gets constantly projected onto the same rod curvilinear abscissa (in green) rather than onto the geometrically exact closest curve point (in pink), thereby causing incorrect contact position and force response despite the smoothness of the original centreline.

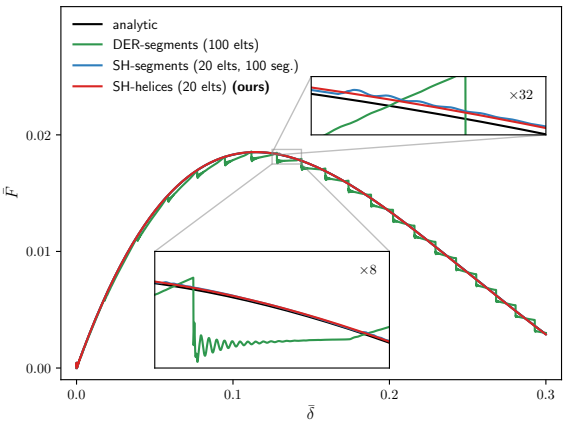


Fig. 8. Three-point bending force-indentation results. The insets show a zoom of the correspondingly framed zone of the curve to illustrate that, even though segment-based detection can provide a seemingly accurate signal, their geometric discontinuities produce non-physical force jumps. In contrast, our method (in red) yields a perfectly smooth signal.

or the axis-aligned bounding box of the curve. In [Lakshmanan et al. 2019], a bounding ellipsoid is derived by bounding the arc-length of the curve from above.

In the following, we derive a simpler volume by taking advantage of Taylor's inequality for C^2 regular curves. The obtained volume corresponds to the fixed distance neighbourhood to a line segment, which is often called a *capsule* (or *swept sphere volume* [Larsen et al. 1999]) in the real-time physics literature [Ericson 2004; Van Den Bergen 2003]. In contrast with the ellipsoids, which require numerical solves to evaluate the distance between each other (c.f. [Choi 2020]), the minimum distance between two capsules derives from the distance between two line segments, which can be computed in constant time (see [Ericson 2004], pp. 148-151). Figure 9 illustrates this bounding volume over a short curve.

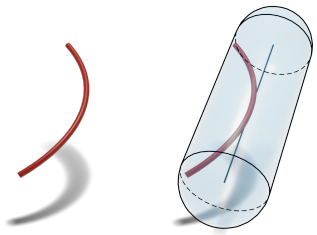


Fig. 9. A short curve (left), and the bounding capsule computed from it with Taylor's inequality (right).

Curve bounding volume. Suppose $\boldsymbol{a}$ is C^2, and let

$$K(\mathcal{I}_A) \equiv \max_{s \in \mathcal{I}_A} \|\boldsymbol{a}''(s)\|.$$

Then, according to Taylor's inequality [Apostol 1991; Taylor 1717], for any s and s_0 in $\mathcal{I}_A$ such that $|s - s_0| \leq \Delta$:

$$\|\boldsymbol{a}(s) - (\boldsymbol{a}(s_0) + (s - s_0)\boldsymbol{a}'(s_0))\| \leq \frac{1}{2} K(\mathcal{I}_A)\Delta^2 \tag{3}$$

Having fixed s_0, the above equation bounds the distance between a curve patch and its first order approximation – that is, a line segment – by a constant. Note that $\Delta \leq |\mathcal{I}_A|$, which means that as our algorithm progresses and $\mathcal{I}_A$ shrinks to a singleton, the bound given by (3) improves quadratically.

Branch and bound algorithm. As mentioned above, the second-order bound obtained in (3) can be exploited within a recursive subdivision algorithm to compute the minimum distance or reject collision, algorithm 1. Note that by letting the user input a distance upper bound b, the algorithm may opt-out early when the curves under consideration are distant enough: if the curves are more than b units apart, the algorithm only does enough work to prove that this is the case and exits without providing the curve parameters where the minimum distance is reached. Setting $b = r_A + r_B$ allows for exiting early when detecting collisions, whereas setting b to $+\infty$ (or to any known upper bound to the minimum distance between the two curves) provides the minimum distance and the matching parameters unconditionally.

We illustrate this algorithm in two ways. First, fig. 10 shows the geometric nature of the procedure, whereby capsules are built from the tangents at the curves' centres (in red), and the curves get subdivided when these capsules intersect pairwise. Second, fig. 11 and the accompanying video depict it as a generic optimisation process, over a 1D domain for clarity. The same ideas generalise to 2D or higher-dimensional domains. Depending on the value of d^* when a specific region is considered, it will either be ruled out because the function is proven to stay above d^* in this region (figure 11a), or subdivided into two regions that will be re-examined later (figure 11b).

Algorithm 1: Curve-curve distance computation

Data:

- precision $\delta > 0$,
- curves $\boldsymbol{a}, \boldsymbol{b} : \mathbb{R} \to \mathbb{R}^d$,
- intervals $\mathcal{I}_A, \mathcal{I}_B$,
- distance upper bound b

Result:

- Distance d^* between the two curves and parameters s_A^* and s_B^* where this distance is reached
- OR $d^* = b$ and null values for s_A^* and s_B^* if the distance between the two curves is larger than b.

1 $Q \leftarrow$ empty priority queue of interval pairs;
2 enqueue(Q, $(\mathcal{I}_A, \mathcal{I}_B)$);
3 $k \leftarrow 0$;
4 $d^* \leftarrow b$;
5 $s_A^*, s_B^* \leftarrow$ null;
6 **while** Q *is not empty* **do**
7 $\quad (\mathcal{I}_A^k, \mathcal{I}_B^k) \leftarrow$ pop(Q);
8 $\quad s_A \leftarrow$ sample from $\mathcal{I}_A^k$;
9 $\quad s_B \leftarrow$ sample from $\mathcal{I}_B^k$;
10 $\quad$ **if** $|\mathcal{I}_A^k| \geq \delta$ *and* $|\mathcal{I}_B^k| \geq \delta$ **then**
11 $\quad\quad l \leftarrow$ `distanceLowerBound`$(\boldsymbol{a}, \boldsymbol{b}, \mathcal{I}_A^k, \mathcal{I}_B^k)$;
12 $\quad\quad$ **if** $l < d^*$ **then**
13 $\quad\quad\quad$ **if** $|\mathcal{I}_A^k| > |\mathcal{I}_B^k|$ **then**
$\quad\quad\quad\quad$ `/* Split interval A */`
14 $\quad\quad\quad\quad$ enqueue(Q, $(\mathcal{I}_A^k \cap]-\infty, s_A], \mathcal{I}_B^k)$);
15 $\quad\quad\quad\quad$ enqueue(Q, $(\mathcal{I}_A^k \cap [s_A, +\infty[, \mathcal{I}_B^k)$);
16 $\quad\quad\quad$ **else**
$\quad\quad\quad\quad$ `/* Split interval B */`
17 $\quad\quad\quad\quad$ enqueue(Q, $(\mathcal{I}_A^k, \mathcal{I}_B^k \cap]-\infty, s_B])$);
18 $\quad\quad\quad\quad$ enqueue(Q, $(\mathcal{I}_A^k, \mathcal{I}_B^k \cap [s_B, +\infty[)$);
19 $\quad d \leftarrow \|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\|$;
20 $\quad$ **if** $d < d^*$ **then**
21 $\quad\quad s_A^* \leftarrow s_A$;
22 $\quad\quad s_B^* \leftarrow s_B$;
23 $\quad\quad d^* \leftarrow d$;
24 $\quad k \leftarrow k + 1$;

Our method may be seen as part of the framework introduced by Johnson and Cohen [1998], and our main algorithmic contribution is the quickly converging yet easy to compute lower bound provided in equation (3). Still, we have kept algorithm 1 as generic as possible, and the following points can be tuned depending on the application:

- **The `distanceLowerBound` function**: as discussed above, we use the Taylor bound from equation (3) around the centre of the fibre element, but any other valid lower bound (including the constant 0) would work as well, although with varying levels of performance. The tighter this bound, the more the algorithm can prune interval pairs from the search tree [2] (line 12 of algorithm 1). Tighter bounds may however be more expensive to compute, yielding an algorithm that is overall slower despite examining fewer steps.
- **The policy to select the sample points** s_A and s_B at lines 8 and 9. In practice, we compute the pair of closest points between the two line segments that are used to approximate both curves, and then reproject these two points onto the curves. A more sophisticated approach may however lead to a lower value of d, in turn decreasing d^* and accelerating the algorithm. Again, a compromise must be made between tightness and performance.
- **The splitting method.** When splitting one of the intervals under examination (lines 12 to 18), another criterion than the interval width may be taken into account. For instance, [Von Herzen et al. 1990] use a metric based on the Jacobian matrix of the parametric function which defines the shapes under consideration, making it more likely that the enqueued interval pair will decrease d^*. A curvature-based metric might similarly be derived to match our lower bound function. Note that the stopping criterion (line 10) must also be updated to match the splitting policy, in order to ensure that the algorithm terminates.
- **An ordering for the priority queue** Q**.** Although it does not affect the result of the algorithm, the order in which interval pairs are handled drastically changes its running time. Indeed, each pair has the opportunity to improve d^*, which in turn helps prune later branches of the search tree. In our experiments, ordering Q by decreasing interval width – that is, handling interval pairs that are widest first – resulted in significantly faster running times than handling the narrowest pairs first. More sophisticated metrics may again lead to faster results.

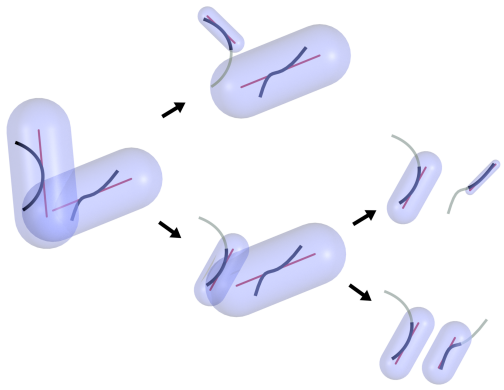
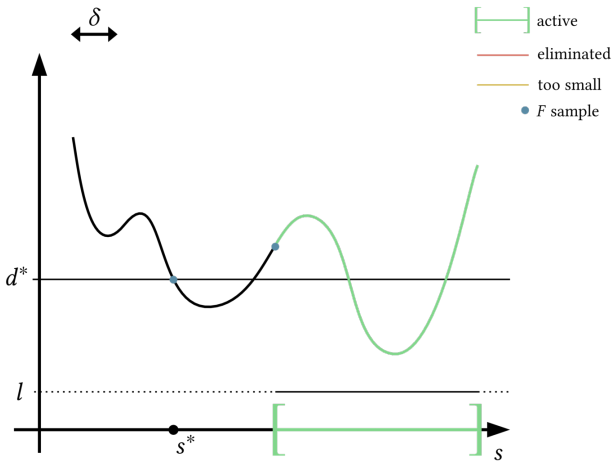
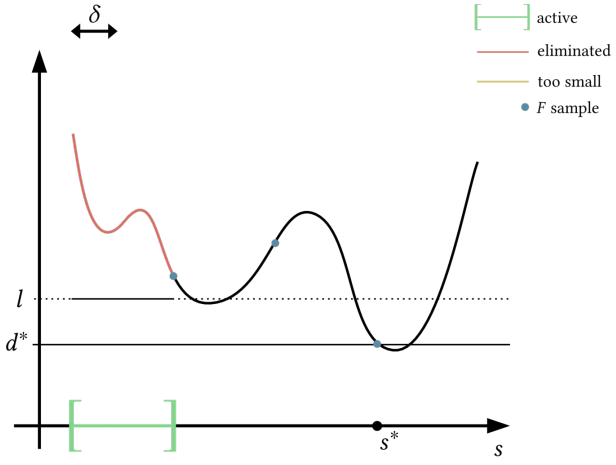


Fig. 10. Geometric view of algorithm 1. Subdivision happens adaptively, only when the current capsule pair intersects. The curve portions under consideration at each step are in black, the tangents used to build the capsules in red.

[2] N.B.: This so-called "search tree" is purely conceptual and is never fully stored in memory.



(a) The active (green) interval cannot be ruled out because $l < d^*$. It must be further subdivided.



(b) The active (green) interval can now be ruled out because $l > d^*$.

Fig. 11. Several steps of our minimisation algorithm, in 1D for clarity.

We have only examined these points to the degree that they make the algorithm usable in practice. Perfecting each of them may yield better performance, but in our simulation scenarios and with a requested precision $\delta = 10^{-8}$ on the result of our algorithm, the computational cost of collision detection remained comparable to that of time-stepping the physical simulation itself, allowing us to run complex large-scale scenarios. However, the points above are interesting directions for improving the performance of the algorithm with no impact on its precision, potentially opening the way for its use in real time simulation or other interactive applications.

Application to the Super-Helix *model.* In the following section, we apply this collision algorithm to the *Super-Helix* model. The curves $\boldsymbol{a}$ and $\boldsymbol{b}$ are therefore helices, and despite the simplicity of the mathematical expression of a single helix, combining two of them into the distance functional $F(s_A, s_B) = \|\boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)\|^2$ results in a non-convex function, whose global minima may not be computed in closed form. Numerical minimisation is non-trivial as well: although in simpler cases, such as helix-point distance computation, only a finite number of convex regions must be examined (see [Nievergelt 2009]), the function F for helix-helix distance is non-convex and does not present as simple a structure. Hence, this case is well suited to the use of our algorithm.

Up to a rigid transformation, any helix can be parametrised as

$$\boldsymbol{a}(s) = (\alpha \cos s, \alpha \sin s, \beta s),$$

for some parameters $\alpha, \beta \in \mathbb{R}$. The norm of the second-derivative is thus $\|\boldsymbol{a}''(s)\| = \sqrt{\alpha^2 + \beta^2}$ for all s, which gives

$$K(\mathcal{I}_A) = \sqrt{\alpha^2 + \beta^2}.$$

The value of $K(\mathcal{I}_A)$ therefore does not depend on $\mathcal{I}_A$ in this case, and may be computed ahead of time.

Contact basis. To conclude the contact detection, we need to construct a local basis at the contact point. A first tangent is constructed as the average of both tangent of the fibres

$$\boldsymbol{t}_1 = \frac{\boldsymbol{a}'(s_A) + \boldsymbol{b}'(s_B)}{\|\boldsymbol{a}'(s_A) + \boldsymbol{b}'(s_B)\|}.$$

The general formula for the normal is given by

$$\boldsymbol{n} = \boldsymbol{a}'(s_A) \times \boldsymbol{b}'(s_B).$$

When both tangents are nearly parallel or anti-parallel, the formula above becomes ill-defined; instead, we then construct the tangent plane orthogonal to the gap function $\boldsymbol{g} = \boldsymbol{a}(s_A) - \boldsymbol{b}(s_B)$ and use the normal

$$\boldsymbol{n} = \frac{(\boldsymbol{t}_1 \times \boldsymbol{g}) \times \boldsymbol{t}_1}{\|(\boldsymbol{t}_1 \times \boldsymbol{g}) \times \boldsymbol{t}_1\|}.$$

When exactly one of the contact is an endpoint of a fibre, the parallel case is used to compute the normal. In the special case where two endpoints are in contact, the normal is given by $\boldsymbol{n} = \boldsymbol{g}/\|\boldsymbol{g}\|$ and an arbitrary tangent $\boldsymbol{t}_1$ is constructed. We also make sure the normal is correctly oriented from A to B and normalised.

The last vector is computed to ensure the basis is orthonormal and direct, that is

$$\boldsymbol{t}_2 = \boldsymbol{t}_1 \times \boldsymbol{n}.$$

Note that at the junction of elements, a contact point could be duplicated in case of small penetration. To handle this specific case, when two contact are closer than a given threshold (set in practice to the fibre width), only the most indented one is kept.

6 RESULTS AND EVALUATION

We apply our detection algorithm, presented in section 5, to the particular case of piecewise helical curves, which allows us to simulate contacting super-helices [Bertails et al. 2006] with high-order contact detection. Contact and dry friction are then resolved precisely using the so-bogus solver [Daviet et al. 2011] with a low tolerance ($\times 10^7$ lower than in [Daviet et al. 2011]), a calibration setup which has already been validated experimentally [Romero et al. 2021].

Please watch our accompanying video for the corresponding animations.

6.1 Benchmark setup

Simulation scenarios. We come back to the **Hair combing** scenario introduced in section 3, and propose to evaluate our algorithm on three main variants:

- **Hair combing (1)**: As in section 3, a single fibre is gently combed from root to tip. This scenario is intended to evaluate precisely the benefit of our method compared to previous ones and to connect our analysis performed in section 4 to a more sophisticated, 3D uncontrolled scenario.
- **Hair combing (2025)**: A wisp of 2025 tightly packed hair – naturally straight, wavy, or curly– is gently combed from root to tip. We furthermore vary its natural curliness, from straight to wavy and curly. Such a challenging scenario is meant, on the one hand, to demonstrate the scalability and robustness of our new detection algorithm. On the other hand, it allows us to show the significant gain in accuracy obtained by our method compared to others, as spurious noise can accumulate over the large number of contacts at play.
- **Tightly coiled hair combing (100)**: A wisp of 100 tightly packed and extremely curled hair, is gently combed from root to tip. To our knowledge, this is the first time that such an extreme case can be simulated effectively. Besides, we report some interesting physical observations which give a glimpse of the potential of our tool for exploring complex entanglement phenomena that still remain poorly understood.

Physical hair parameters. We set the geometrical and physical parameters of our fibres using realistic hair parameters [Robbins 2012]. To study different physical settings, we vary their natural curliness, from **Straight** to **Wavy**, **Curly** and **Tightly Coiled**. When multiple fibres are simulated within a wisp, we may in addition vary the hair-hair friction coefficient, from $\mu = 0.1$ ("**clean**") to $\mu = 0.3$ ("**dirty**"). Note that the friction coefficient between the fibres and the comb is set to the same value for the sake of simplicity. See table 3 for a complete description of the chosen physical parameters.

Combing protocol. Each fibre is clamped at one end, free at the other end. It is first left to fall under gravity until it stabilises. In the case of a wisp, the roots of the fibres are set on a grid, with small random offsets to break regularity. To generate pernetration-free initial configurations, we start with straight – aligned – hair fibres, and then wait until the wisp relaxes under its own weight: it then slowly adopts a static equilibrium involving a competition between the natural curliness of the fibres, gravity, and frictional contact between the fibres (see fig. 14, top panel, and the accompanying video).

Following the fibre relaxation process, we start moving a virtual comb into the fibre or the wisp. The comb is designed with realistic dimensions (teeth with diameter 1.1 mm, spaced by 1.2 mm), and is in practice modelled as a rigid kinematic body composed of smooth capsules (one thin capsule per tooth, and one larger capsule for the handle), which allows us to detect contacts between the comb and the fibres just by reusing our fibre-fibre detection algorithm. The comb is first translated so that its teeth catch on the fiber(s), before being pulled forwards and downwards (following a diagonal trajectory) with constant velocity.

Comparison to other fibre simulators. Our main baseline for comparison is *SH-segments*, which consists of the smooth *Super-Helix* fibre model coupled with segment-based collision detection [Daviet et al. 2011]. We do not keep the – low-order – *DER-segments* model in the comparisons as it has already been shown in fig. 2b to generate prohibitive noise even for moderate curvatures (**Wavy** fibre) and for a large number of segments (200): we can only expect noise to worsen as the curvature is increased.

Fibre radius r	50 μm
Fibre length L	30.5 cm
Natural curvature κ_0 **Straight**	0.1 cm^{-1}
Natural curvature κ_0 **Wavy**	0.6 cm^{-1}
Natural curvature κ_0 **Curly**	1.0 cm^{-1}
Natural curvature κ_0 **Tightly Coiled**	2.0 cm^{-1}
Density ρ	1 g cm^{-3}
Young modulus E	1 GPa
Poisson ratio ν	0.48
Friction coefficient μ	0.1 ("**clean**") or 0.3 ("**dirty**")

-	n_{elts}	n_{coll}	dt (s)
SH-segments (12)	12	100	1×10^{-3}
SH-helices (12) **(ours)**	12	12	1×10^{-3}

Tolerance for the detection **Hair combing (1)**	1×10^{-8}
Tolerance for the detection **Hair combing (2025)**	1×10^{-6}
Tolerance for the constraint solver	1×10^{-12}
Max number of iterations for the constraint solver	500

Table 3. Hair combing setup and numerical parameters.

6.2 Hair combing (1)

Wavy fibre. We apply our new high-order *SH-helices* algorithm to our introductory example, **Hair combing (1)** with a single **Wavy** fibre. Results are presented in fig. 12, and compared against the previous *SH-segments* method. In contrast to *SH-segments* (in blue), which shows significant noise as soon as the curliness becomes moderate, i.e. close to the tip of the fibre, our method (in red) completely gets rid of this noise and allows one to retrieve a smooth force signal at any fibre curvature. Note that the impact at $t = 14.3$ s corresponds to a physical impact between the end of the fibre and a teeth of the comb.

Connection with our analysis from section 4. Our observations above are qualitatively in line with our analysis in section 4 regarding the influence of curvature at contact. To make a clearer and more quantitative connection, we measure, in the **Hair combing (1)** – **Wavy** scenario, the amplitude of the noise as a function of the actual curvature at the contact point (which differs from the natural curvature of the fibre) for *SH-segments*. The amplitude of the noise is computed in a similar way than in the three point bending experiment. However, because no analytical curve is available in the

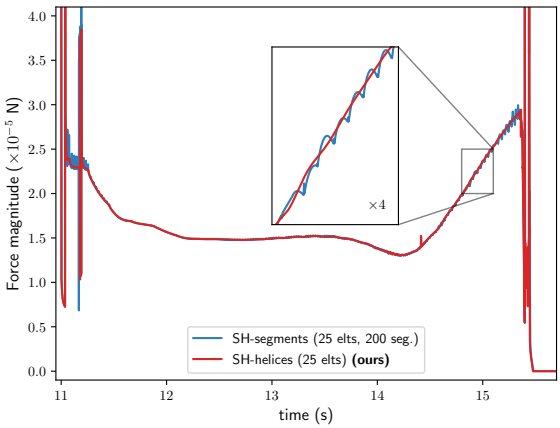


Fig. 12. **Hair combing (1)** result: comparison between our method and the previous *Super-Helices* method

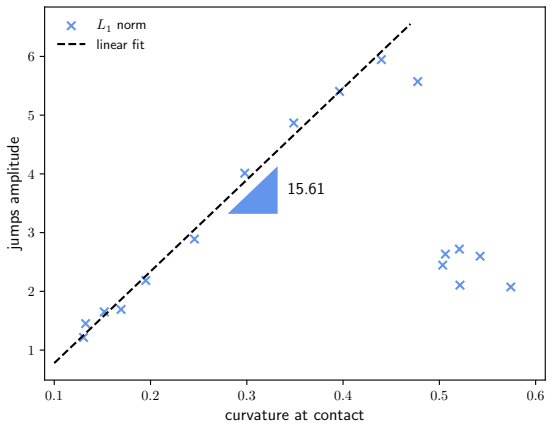


Fig. 13. Force jumps with the actual curvature at contact in the **Hair combing (1)** scenario. Normalisation of *SH-segments* by *SH-helices*.

combing test, the normalisation is done using the curve obtained with *SH-helices*, considered as the reference.

The actual curvature at contact is defined as the ℓ_2-norm of the vector of 3 material curvatures/twist (or equivalently, as the ℓ_2-norm of the Darboux vector). To prevent incorporating large force variations coming from real impacts at the beginning and end of the simulation, we restrict our analysis to the interval between 11.3 s and 15.3 s. Results are reported in fig. 13. Interestingly, although our scenario is much more complex than the three-point bending test (3D, multiple contact points, uncontrolled curvature at contact), we retrieve on a large range of curvatures[3] the linear dependency of the noise amplitude with respect to the curvature at the contact point in the *SH-segments* case – as predicted by our analysis in section 4. The linear factor appears to be 6× greater[4] than in the three-point bending protocol, which makes the **Hair combing (1)** scenario even more sensitive to detection artifacts. As for our method, it systematically removes all the jumps regardless of the curvature at the contact point.

6.3 Hair combing (2025)

We now turn to the challenging scenario of combing hair wisps made of thousands tightly contacting fibres.

Geometries. Our visual results are depicted in fig. 14, from **Straight** (top) to **Wavy** (middle) and **Curly** (bottom). For the **Wavy** and **Curly** types, we observe interesting rearrangements of individual hair fibres after combing. In particular, as in reality, the initially well-formed curls are broken by the comb. In the **Wavy** case, the fibres eventually reshape into two entwined main wisps. In the **Curly** case, the wisp gets divided into more smaller wisps, surrounded by many individual hairs that give a fuzzy appearance to the whole set.

Forces. To analyse the combing force applied onto the hair wisp, we focus on the **Wavy** wisp, as it already allows us to address a challenging case while keeping timings reasonable.

During the combing process, the total force applied by the comb is measured by summing up all the forces at each contact point, and plotted as a function of its indentation. We then compare the force profiles between our method and *SH-segments*. Results are shown in fig. 1, right. A first observation is that, unlike in **Hair combing (1)**, *SH-segments* now yields a reasonably regular force signal, suggesting that averaging over a large number of contacts tends to decrease the punctual impact of individual jumps. However, a second observation is that this signal significantly deviates from our solution, especially at large indentations where the curvature at contact points becomes large. This shows that the noise due to segment-based contact detection does not cancel out as multiple hairs are involved. Quite the opposite, the noise eventually accumulates, due to both comb-hair and hair-hair interactions, giving rise in the end to a flawed force signal.

In contrast, our approach eliminates artifacts due to collision detection in curved fibres. As such, it is expected to increase significantly the reliability of the force predictions. In the future, this newly reached accuracy paves the way to further investigate and better understand tightly interacting fibre systems, for instance to disentangle the respective roles of elasticity and friction in complex fibrous media.

As a first illustrative study towards this goal, we vary the friction coefficient on the **Hair combing (2025) Wavy** between $\mu = 0.1$ ("**clean** ") and $\mu = 0.3$ ("**dirty** ") and observe in fig. 15 the resulting forces during combing and the resulting geometries after combing. As expected, the forces in the **dirty** case incorporates much more dissipation than in the **clean** case. Interestingly, these changes in

[3]In this more complex scenario compared to the curved three-point bending test, we observe at the end of the comb indentation a shift between the reference curve and the average of the noisy curve (see inset of fig. 12). This shift at large curvatures actually degrades the method we set in section 4.3 to compute the amplitude of the jumps, which explains the outliers in the lower part of fig. 13.

[4]Note that it cannot be expected to retrieve the same linear factor in the two sliding protocols, as the magnitude of the force jumps depends on the (perturbed) dynamic system to be integrated.

forces incur different trajectories of the individual fibres. In particular the initial wisp breaks into more and thinner wisps in the **dirty** case, which rearrange themselves into different patterns compared to the **clean** wisp.

6.4 Tightly coiled hair combing (100)

An extreme case of curliness is *tightly coiled* hair, where fibres may feature a radius of curvature of a few millimeters only (down to 1.5 mm), i.e. a natural curvature well beyond $1\,\mathrm{cm}^{-1}$ (up to around $7\,\mathrm{cm}^{-1}$). We set curl radii to 5 mm, corresponding to $\kappa_0 = 2\,\mathrm{cm}^{-1}$. These parameters, though in the lower bound for "kinky" or "Afro" hair, already yield the tight spring-like shape represented in fig. 16, categorised as Type V-VI hair [De La Mettrie et al. 2007]. Note also the extreme shortening of the apparent length of the wisp at equilibrium (only 4 cm), about 87% compared to a straight wisp (30 cm). Very few approaches in the past have attempted to simulate such highly curly hair. Bertails et al. [2005] simulate a static head made of a few hundreds Type V-VI hair wisps, where each wisp is guided by a single super-helix at equilibrium under gravity and contacts. With a similar wisp-based strategy, Shi et al. [2023] has recently achieved the fast simulation of several thousands tightly coiled hair wisps. Their guide hair is animated using an ad-hoc position-based coiled spring model with volumetric twist energy, which turns out to be orders of magnitude faster than *Discrete Elastic Rods*. However, none of these former methods tackle the accurate simulation of each individual hair that composes a coiled wisp, let alone the coupling to strong contact constraints imposed inside the wisp.

To capture such a high curl density correctly, we had to raise the number of elements of the *Super-Helices* from 12 to 32. Otherwise, the discretisation creates an artificial stiffness, making the simulation harder to converge and causing penetrations. To keep timings reasonable (10 ms per time step), we reduce the size of the wisp to 100 hair fibres (up to 10000 contacts, 6946 on average). As illustrated in fig. 16 and in our accompanying video, we manage to simulate an impressive scenario where the tightly coiled wisp is entirely combed from root to tip. To the best of our knowledge this is the first time that one can access such a level of precision in the simulation of tightly coiled hair assemblies. In particular, we notice the appearance of a helical perversion after combing, as shown in fig. 16, bottom left and middle. This physical phenomenon is typical of scenarios where a fibre with strong natural curvature is extended and then relaxed [McMillen and Goriely 2002],[Bergou et al. 2008, fig.2].

6.5 Performance

Because it subdivides the curves of interest adaptively, the algorithm presented in section 5 allows us to reach arc-length tolerances on the order of 10^{-8} in reasonable computational time. Hence, using it in lieu of the segment-based collision schemes used in fibre simulations not only adresses the force artifacts described in section 3, but also keeps running times reasonable.

In our simulations, we prune out non-colliding curve pairs using a sweep-and-prune algorithm ([Ericson 2004], pp. 329-338) on the axis-aligned bounding boxes of the curves. This broad phase is followed by a narrow phase, in which we may use either algorithm 1 or segment-based detection. For segment-based detection, we use the following routine:

(1) Set a required arc-length tolerance δ.
(2) Split each curve into segments by sampling it evenly such that the endpoints of each segment are no more than δ arc-length units apart.
(3) Gather the segments for each curve into two bounding volume hierarchies BVH_A and BVH_B consisting of axis-aligned bounding boxes.
(4) Find colliding segments by going through BVH_A and BVH_B simultaneously, pruning out subtrees when their bounding boxes are further than $r_A + r_B$ apart.

This fairly simple algorithm may be tweaked in a number of ways in order to improve its performance, but using BVHs already constitutes a drastic improvement over naïvely checking each pair of segments. As such, we use it as a baseline against which to compare our algorithm.

To evaluate both algorithms, we use a trace of all calls to the narrow phase collision routine during a frame of the comb scenario as a benchmark, which amounts to about 18 000 pairs of curves. This data set allows us to assess "real life" performances more clearly than using a set of random curves would.

Figure 17 displays the convergence rate of algorithm 1 – that is, the computation time as a function of the reached arc-length error. For fairness, we use the mean reached error (computed as the parameter-space L^∞ distance to a "ground truth" result computed by our algorithm with $\delta = 10^{-12}$) rather than the input tolerance. This matters for example when approximating nearly-straight curves with segments, where even low segment counts may yield very low errors.

As we can see, algorithm 1 outperforms segment-based detection at all but the highest errors. However, because our implementation efforts were focused on algorithm 1 rather than on segment-based detection, comparing raw CPU timings is not entirely fair. Still, our algorithm is able to reach much lower errors because its complexity scales up more slowly than that of segment-based detection. Indeed, dividing the arc-length error by 10 results multiplies the computation time by more than 15 when using segments, but only by about 1.4 when using our algorithm.

Besides, although it is a narrow phase collision detector, our algorithm exhibits a behaviour similar to broad phase collision detection. Indeed, it wastes little time on non-colliding curve pairs: as figure shown in fig. 17, rejecting a collision is at least two orders of magnitude faster than accepting it, and the time taken to reject a collision does not increase with the precision. This makes sense considering that when the curves of interest do not collide, the algorithm never explores a higher precision than that required to prove separation.

Figure 18 shows this in more detail: as soon as the gap between fibres is positive, the computation time drops by two orders of magnitude, as described above, and continues dropping for further separated curves.

Finally, our algorithm is lighter on memory than one using a BVH to store segments. Indeed, while the segment-based algorithm must

Fig. 14. Combing the three hair wisps **Straight** (brown), **Wavy** (blond), and **Curly** (red) (2025 fibres) with $\mu = 0.1$.

scenario	# contacts	time per step	total time	detection	construction	solve
Hair combing (2025), **Straight**, $\mu = 0.1$	21191	12.9 s	90 h	1.7 s (13.3 %)	7.1 s (55.2 %)	3.8 s (29.8 %)
Hair combing (2025), **Wavy**, $\mu = 0.1$	28709	24.4 s	170 h	10.3 s (42.1 %)	7.2 s (29.4 %)	6.7 s (27.6 %)
Hair combing (2025), **Curly**, $\mu = 0.1$	84806	62.5 s	435 h	22.5 s (36.0 %)	7.6 s (12.1 %)	32.2 s (51.4 %)
Hair combing (2025), **Wavy**, $\mu = 0.3$	29282	23.6 s	165 h	9.3 s (39.3 %)	7.2 s (30.5 %)	6.9 s (29.3 %)
Tightly coiled hair combing (100), $\mu = 0.1$	6946	9.5 s	66 h	0.5 s (5.2 %)	6.3 s (66.5 %)	2.6 s (27.9 %)

Table 4. Average number of contacts per step over the whole simulation, average time per time step ($\delta t = 1$ ms for all cases), total time for the whole simulation, and time per timestep and percentage of time spent in the three main phases of the simulation. Note that only the solve and the detection of contacts are parallelised. The construction of the system is monocore. The simulations were run on Intel Xeon CPU with 8 cores running at 3.5 GHz. Each **Hair combing (2025)** represents 25 s of simulated time. The **Tightly coiled hair combing (100)** has a duration of 30 s.

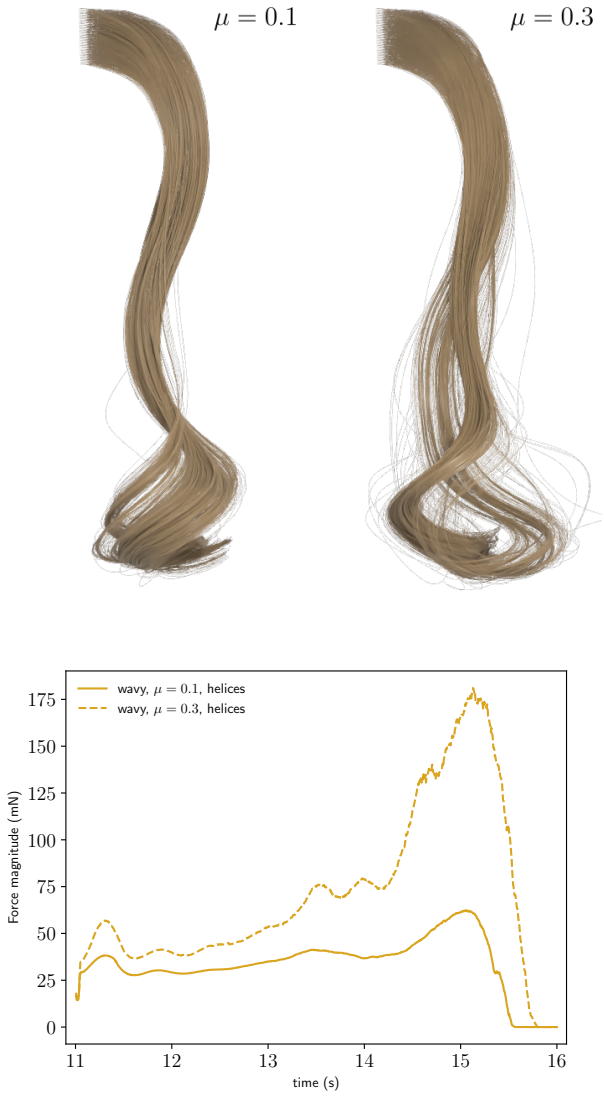


Fig. 15. Varying the friction coefficient gives different result both geometrically (top) and force-wise (bottom). The "dirty" ($\mu = 0.3$) wisp realistically splits into multiple smaller wisps.

Fig. 16. Simulation of the combing of a tightly coiled wisp consisting of 100 super-helices made of 32 elements each. Combing not only breaks the initially regular curls into small fuzzy strands, but it also generates an helical perversion (i.e. a change of handedness) in the middle of the main strand, as in reality (bottom left).

set up and hold the BVH structure in memory before traversing it, our algorithm builds the search tree dynamically as it traverses it. Hence only the priority queue Q must be stored in memory, and our algorithm is unlikely to become memory bound at higher precisions.

7 DISCUSSION AND PERSPECTIVES

Our analysis and results show that the segment-based *Discrete Elastic Rods* model, combined with a low-order detection scheme, systematically produces spurious high-frequency jumps in the force signal, which do not disappear when increasing the number of elements. For *Super-Helices* coupled with low-order detection, similar artifacts arise, but only appear to be critical in curved fibre configurations.

Our high-order detection algorithm was proved to eliminate such artifacts efficiently and robustly in the case of *Super-Helices*, yielding smoother and much cleaner force profiles. Our method turns out

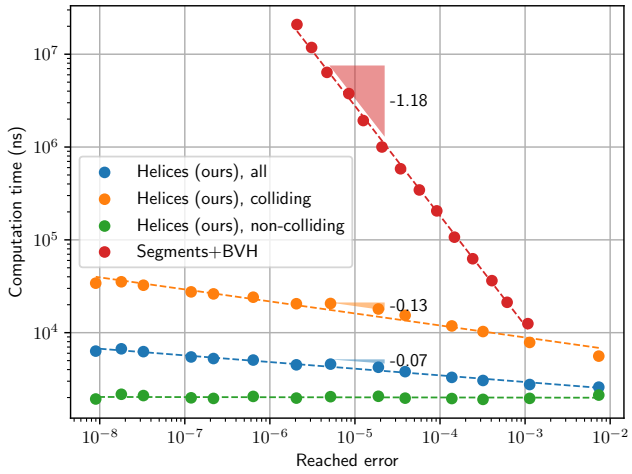


Fig. 17. Computation time as a function of the reached arc-length error. Note the log-log scale. The green curve's slope is not drawn because it is approximately zero.

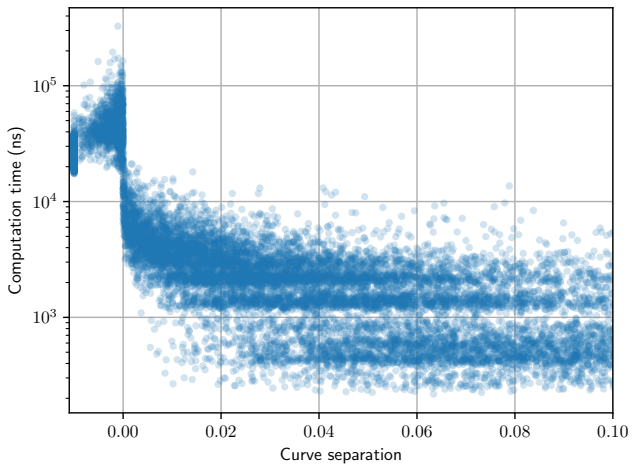


Fig. 18. Our algorithm's computation time as a function of the gap between the two curves. Here, the benchmark was run with $\delta = 10^{-8}$.

to be particularly effective in eliminating the substantial noise that can be accumulated in large curved fibre systems due to low-order detection, hence suggesting that contact forces in wavy and curly fibre assemblies are predicted with significantly better accuracy using our approach.

Overall, our method represents a solid starting point to further explore the potential of high-order collision detection schemes, and could still be improved in a number of different ways. Below we comment on current limitations of our approach and discuss exciting potential venues for improvement and extension in the future.

7.1 Contact point pruning and time consistency

While contact points can be restricted to the mesh vertices in the case of polyhedral bodies [Baraff 1989; Palmer 1989], the discretisation of smooth (and thus potentially non-convex) continuous contact zones has not been thoroughly studied, and is often handled with exhaustive contact detection and pruning, which lead to poorly controlled contact positions and correspondingly ill-conditioned systems to solve for the constraint contact forces. Despite its ability to naturally generate a discretisation of a helix-helix contact zone with a resolution prescribed by the arc-length tolerance (δ), our algorithm does not handle the case of contact manifolds across several pieces of curves, which are actually handled pairwise and separately; as such, it offers no guarantee to consistently discretise the overall contact area.

Time-aware collision detection is also important in many dynamical simulations, as it ensures robustness of the non-penetration constraints at the discrete time level and allows for accurate dynamics simulation with time-stepping (non event-driven) methods. Introducing time dependency within collision detection is often performed by resorting to *Continuous Collision Detection* approaches, which try to predict collisions in the next time-step *a priori* by extrapolating the motion of bodies at play. This paradigm has been successfully used for simplicial rigid [Redon et al. 2002], deformable shapes [Brochu et al. 2012], or high-order parametric surfaces [Marschner et al. 2021; Snyder et al. 1993; Von Herzen et al. 1990], and extending our algorithm to perform ahead-of-time detection would not only improve its robustness when using large time steps, but also enable consistent evolution of contact points in time. Directly handling persistent contact points as an additional constraint to the system, as performed in Kry and Pai [2003], could also provide valuable extension of our approach to ensure that the penetration does not increase at contact points as the system is evolved in time. Their approach however uses surface patches rather than curves, and is limited to one contact point, but nevertheless provides an interesting direction for future work.

7.2 Performance

As outlined at the end of section 5, we have done little work to push the performance of our algorithm to the best possible level. To sum up, the lowest-hanging fruits are:

(1) Using geometric information to subdivide curves based on how accurate Taylor's approximation is, rather than always splitting the longest curves. This would focus computational effort on the regions that most need it.
(2) Further investigating how the order of the priority queue affects the computation time, and using geometric information to accelerate the traversal of the search tree.

7.3 Generalizability

Note that algorithm 1 is not limited to helical curves, and is in fact quite general. In order to run it on a pair of curves, all that is needed of each curve is the ability to :

(1) Sample the curve at any point
(2) Sample the curve's derivative at any point

(3) Compute an upper bound on the norm of the curve's second derivative on a given closed interval.

Of these three, requirement 3 is of course the most restrictive, but it is by no means prohibitive in practice, because the second derivative of any C^2 curve is bounded on any closed interval. Moreover, the curvature upper bound needs not be tight, as the Δ^2 term in equation (3) ensures that in the limit, the error made by approximating the curve with a line segment goes to zero. Looser bounds are easier to compute in many cases, and they incur little additional computational cost, because the Δ^2 term quickly compensates the poor initial fit as Δ gets lower with subdivisions.

For instance, consider polynomial splines, which are often used in Graphics. The second derivative of a degree d polynomial is a degree $(d-2)$ polynomial, for which a loose upper bound may easily be computed by e.g. bounding each monomial individually :

$$\forall t \in [a,b], \quad \left\| \sum_{k=0}^{d-2} \boldsymbol{p_k} t^k \right\| \leq \sum_{k=0}^{d-2} \|\boldsymbol{p_k}\| \cdot |t|^k \leq \sum_{k=0}^{d-2} \|\boldsymbol{p_k}\| (\max(|a|,|b|))^k$$

In practice it is beneficial to compute tighter bounds, especially for lower values of d where analytical bounds may be readily computed.

In the more common case of quadratic (respectively cubic) splines, the second derivative of the curve is a constant (resp. affine) function, hence a closed-form tight upper bound can be used. Our algorithm is fast in that use case as well, taking on the order of 10 to 100 microseconds to compute the minimum distance between two cubic curves with 10^{-8} on a single core of a modern processor. This makes it competitive with recent techniques such as that presented in [Zhang et al. 2023]: their approach takes about 3 milliseconds on a 4-core CPU to detect a static collision between two so-called "tapered cubic cylinders", which are cubic curves swept by a sphere whose radius varies linearly with the curve parameter. A direct comparison would not be quite fair, however. First, the primitives are not perfectly identical (ours lack the varying radius). Second, and perhaps more importantly, our approaches to handling smooth contact generalize in orthogonal directions : theirs applies to polynomial patches of any dimension and allows for continuous collision detection, whereas ours applies to arbitrary C^2 curves, including those with transcendental expressions (as illustrated in our application to helices), and those without a closed-form expression. Still, considering these rough timing estimates allows us to envision how such approaches might be used in practice. In a real-time simulation context for instance, spending 3ms of the 16 or 30 ms time budget for a frame on a single pair of primitives is prohibitive, but shrinking this down to a few tens of microseconds makes simulating many more curves attainable.

Closer to our approach is the one by Chang et al. [2011]. The execution times presented in the paper are similar to ours, on the order of 100 microseonds to compute the minimum distance between two randomly selected Bézier curves. However, their technique relies on the properties of Bézier curves to bound and subdivide curves, and as such does not generalize to other curve types.

8 CONCLUSION AND FUTURE WORK

Our contributions are three-fold.

First, we have outlined the issues that arise when using line segments for collision detection between fibres in physics-based animation. No matter the number of line segments used for collision detection, their geometric nature will introduce noise to the contact forces between fibres. As we have demonstrated with a simple curved three point bending experiment, these erroneous forces are not negligible and worsen as the curvature increases.

Second, we propose a novel branch-and-bound algorithm to compute the distance between parametric curves, which may be applied to collision detection when using high-order models such as the *Super-Helix* model. By using curvature information to bypass unnecessary computational effort, this algorithm can compute contact point positions up to a very high precision, thereby suppressing the force artifacts that segment-based detection leads to. Besides, we have compared the performance of our algorithm to that of its segment-based counterpart and found that it scales better to the high precision needed to handle the aforementioned force artifacts.

Finally, we have tested our end-to-end high-order contact simulation in a larger scale scenario with hair combing, and shown that it remains robust even with a high number of fibres and contact points. Our force measurements in these scenarios again highlight the need for accurate high-order collision detection.

This work has allowed us to consider in detail the impact of collision detection on fibre simulation, and to substantially improve the accuracy of the resulting force response. We are excited to further enhance our entire pipeline towards the high-fidelity of large scale simulation of fibre assemblies with extreme curvatures. Another important avenue we plan for future research is the fine capture of stick-slip events, which are critical to the subtle dynamics of knitted cloth or to the sound generation from bowed string instruments.

ACKNOWLEDGMENTS

We would like to thank Laurence Boissieux for rendering the 3D animations of the hair combing scenario with 3DSMax, and Mélina Skouras for providing us with the tightly coiled hair sample shown in fig. 16. We are also grateful to the reviewers for their helpful comments, and in particular for bringing to our attention the helical perversion in our simulation of tightly coiled hair. Figures 7, 9, 10, 11, as well as our accompanying video, were generated using the free software Blender.

REFERENCES

Steven S. An, Doug L. James, and Steve. Marschner. 2012. Motion-driven Concatenative Synthesis of Cloth Sounds. *ACM Trans. Graph.* 31, 4, Article 102 (July 2012), 10 pages. https://doi.org/10.1145/2185520.2185598

Tom M Apostol. 1991. *Calculus, volume 1.* John Wiley & Sons.

David Baraff. 1989. Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Proceedings of the 16th annual conference on Computer graphics and interactive techniques.* 223–232.

David Baraff. 1994. Fast contact force computation for nonpenetrating rigid bodies. In *Computer Graphics Proceedings.* ACM, New York, NY, USA, 23–34.

Aric Bartle, Alla Sheffer, Vladimir Kim, Danny M. Kaufman, Nicholas Vining, and Floraine Berthouzoz. 2016. Physics-driven Pattern Adjustment for Direct 3D Garment Editing. *ACM Trans. Graph.* 35, 4, Article 50 (July 2016), 11 pages.

Milan Batista. 2015. Large deflections of a beam subject to three-point bending. *International Journal of Non-Linear Mechanics* 69 (2015), 84–92.

Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. 2010. Discrete viscous threads. *ACM Transactions on graphics (TOG)* 29, 4 (2010), 1–10.

Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. 2008. Discrete elastic rods. In *ACM SIGGRAPH 2008 papers*. 1–12.

Florence Bertails, Basile Audoly, Marie-Paule Cani, Bernard Querleux, Frédéric Leroy, and Jean-Luc Lévêque. 2006. Super-helices for predicting the dynamics of natural hair. *ACM Transactions on Graphics (TOG)* 25, 3 (2006), 1180–1187.

Florence Bertails, Basile Audoly, Bernard Querleux, Frédéric Leroy, Jean-Luc Lévêque, and Marie-Paule Cani. 2005. Predicting Natural Hair Shapes by Solving the Statics of Flexible Rods. In *Eurographics'05 (short papers)*, J. Dingliana and F. Ganovelli (Eds.). http://www-evasion.imag.fr/Publications/2005/BAQLLC05

Robert Bridson, Ronald Fedkiw, and John Anderson. 2002. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph.* 21, 3 (2002), 594–603. http://www.cs.ubc.ca/~rbridson/docs/cloth2002.pdf

Tyson Brochu, Essex Edwards, and Robert Bridson. 2012. Efficient geometrically exact continuous collision detection. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–7.

Romain Casati and Florence Bertails-Descoubes. 2013. Super Space Clothoids. *ACM Transactions on Graphics (Proc. ACM SIGGRAPH'13)* 32, 4, Article 48 (July 2013), 12 pages. https://doi.org/10.1145/2461912.2461962

Jung-Woo Chang, Yi-King Choi, Myung-Soo Kim, and Wenping Wang. 2011. Computation of the minimum distance between two Bézier curves/surfaces. *Computers & Graphics* 35, 3 (2011), 677–684. https://doi.org/10.1016/j.cag.2011.03.025 Shape Modeling International (SMI) Conference 2011.

Min Gyu Choi. 2020. Computing the closest approach distance of two ellipsoids. *Symmetry* 12, 8 (2020), 1302.

Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A Otaduy. 2014. Yarn-level simulation of woven cloth. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–11.

Alexander Clegg, Jie Tan, Greg Turk, and C. Karen Liu. 2015. Animating Human Dressing. *ACM Trans. Graph.* 34, 4, Article 116 (July 2015), 9 pages. https://doi.org/10.1145/2766986

Gilles Daviet, Florence Bertails-Descoubes, and Laurence Boissieux. 2011. A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. In *Proceedings of the 2011 SIGGRAPH Asia Conference*. 1–12.

Roland De La Mettrie, Didier Saint-Léger, Genevièvève Loussouarn, Annelise Garcel, Crystal Porter, and André Langaney. 2007. Shape variability and classification of human hair: a worldwide approach. *Human biology* 79, 3 (2007), 265–281.

Laura De Lorenzis, Peter Wriggers, and Thomas J.R. Hughes. 2014. Isogeometric contact: a review. *GAMM-Mitteilungen* 37, 1 (2014), 85–123. https://doi.org/10.1002/gamm.201410005 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/gamm.201410005

Zackory Erickson, Alexander Clegg, Wenhao Yu, Greg Turk, C. Karen Liu, and Charles C. Kemp. 2017. What does the person feel? Learning to infer applied forces during robot-assisted dressing. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 6058–6065. https://doi.org/10.1109/ICRA.2017.7989718

Christer Ericson. 2004. *Real-time collision detection*. Crc Press.

Ye Fan, Joshua Litven, David I. W. Levin, and Dinesh K. Pai. 2013. Eulerian-on-Lagrangian Simulation. *ACM Trans. Graph.* 32, 3, Article 22 (jul 2013), 9 pages. https://doi.org/10.1145/2487228.2487230

Yun Fei, Christopher Batty, Eitan Grinspun, and Changxi Zheng. 2019. A multi-scale model for coupling strands with shear-dependent liquid. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–20.

Robert Frisch-Fay. 1962. *Flexible bars*. Butterworths.

Rony Goldenthal, David Harmon, Raanan Fattal, Michel Bercovier, and Eitan Grinspun. 2007. Efficient simulation of inextensible cloth. In *ACM Trans. Graph.* (San Diego, California) *(SIGGRAPH '07)*. ACM, New York, NY, USA, Article 49. https://doi.org/10.1145/1275808.1276438

Sunil Hadap. 2006. Oriented strands - dynamics of stiff multi-body system. In *ACM SIGGRAPH - EG Symposium on Computer Animation (SCA'06)*. ACM-EG SCA, 91–100.

Thomas J.R. Hughes, J. Austin Cottrell, and Yuri Bazilevs. 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 194, 39 (2005), 4135–4195. https://doi.org/10.1016/j.cma.2004.10.008

David E. Johnson and Elaine Cohen. 1998. A framework for efficient minimum distance computations. *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)* 4 (1998), 3678–3684 vol.4. https://api.semanticscholar.org/CorpusID:5297080

Danny M. Kaufman, Shinjiro Sueda, Doug L. James, and Dinesh Pai. 2008. Staggered Projections for Frictional Contact in Multibody Systems. *ACM Trans. Graph.* 27, 5, Article 164 (Dec. 2008), 11 pages.

Danny M. Kaufman, Rasmus Tamstorf, Breannan Smith, Jean-Marie Aubry, and Eitan Grinspun. 2014. Adaptive nonlinearity for collisions in complex rod assemblies. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–12.

Paul G. Kry and Dinesh K. Pai. 2003. Continuous Contact Simulation for Smooth Surfaces. *ACM Trans. Graph.* 22, 1 (jan 2003), 106–129. https://doi.org/10.1145/588272.588280

Arun Lakshmanan, Andrew Patterson, Venanzio Cichella, and Naira Hovakimyan. 2019. Proximity queries for absolutely continuous parametric curves. *arXiv preprint arXiv:1902.05027* (2019).

Eric Larsen, Stefan Gottschalk, Ming C Lin, and Dinesh Manocha. 1999. *Fast proximity queries with swept sphere volumes*. Technical Report TR99-018. Department of Computer Science, University of North Carolina.

Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy R. Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. 2020a. Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. *ACM Trans. Graph.* 39, 4 (2020), 49.

Minchen Li, Danny M. Kaufman, and Chenfanfu Jiang. 2020b. Codimensional incremental potential contact. *arXiv preprint arXiv:2012.04457* (2020).

Zoë Marschner, Paul Zhang, David Palmer, and Justin Solomon. 2021. Sum-of-squares geometry processing. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–13.

Jonàs Martínez, Mélina Skouras, Christian Schumacher, Samuel Hornus, Sylvain Lefebvre, and Bernhard Thomaszewski. 2019. Star-Shaped Metrics for Mechanical Metamaterial Design. *ACM Transactions on Graphics* 38, 4 (July 2019), Article No. 82 :1–13. https://doi.org/10.1145/3306346.3322989 Special issue, SIGGRAPH 2019.

Tyler McMillen and Alain Goriely. 2002. Tendril perversion in intrinsically curved rods. *Journal of Nonlinear Science* 12, 3 (2002), 241–281.

Christoph Meier, Wolfgang A Wall, and Alexander Popp. 2017. A unified approach for beam-to-beam contact. *Computer Methods in Applied Mechanics and Engineering* 315 (2017), 972–1010.

Ken Museth. 2020. Physics simulations: Is it Hollywood magic or rocket science. http://computeranimation.org/2020/program.html#keynote1

Yves Nievergelt. 2009. Computing the distance from a point to a helix and solving Kepler's equation. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 598, 3 (2009), 788–794.

Olivier Nocent and Yannick Remion. 2001. Continuous deformation energy for dynamic material splines subject to finite displacements. In *EG workshop on Computer Animation and Simulation (EG CAS'01)*. 88–97.

Richard S Palmer. 1989. *Computational complexity of motion and stability of polygons*. Technical Report. Cornell University.

Astrid Pechstein and Johannes Gerstmayr. 2013. A Lagrange–Eulerian formulation of an axially moving beam based on the absolute nodal coordinate formulation. *Multibody System Dynamics* 30 (2013), 343–358.

Stéphane Redon, Abderrahmane Kheddar, and Sabine Coquillart. 2002. Fast continuous collision detection between rigid bodies. In *Computer graphics forum*, Vol. 21. Wiley Online Library, 279–287.

Clarence Robbins. 2012. *Chemical and physical behavior of human hair*. Vol. 4. Springer.

Victor Romero, Mickaël Ly, Abdullah-Haroon Rasheed, Raphaël Charrondière, Arnaud Lazarus, Sébastien Neukirch, and Florence Bertails-Descoubes. 2021. Physical validation of simulators in Computer Graphics: A new framework dedicated to slender elastic structures and frictional contact. *ACM Transactions on Graphics* 40, 4 (Aug. 2021), Article 66: 1–19. https://doi.org/10.1145/3450626.3459931

Rosa M. Sánchez-Banderas, Alejandro Rodríguez, Héctor Barreiro, and Miguel A. Otaduy. 2020. Robust Eulerian-on-Lagrangian Rods. *ACM Trans. Graph.* 39, 4, Article 59 (aug 2020), 10 pages. https://doi.org/10.1145/3386569.3392489

Daniel Scholz. 2011. *Deterministic global optimization: geometric branch-and-bound methods and their applications*. Vol. 63. Springer Science & Business Media.

Eston Schweickart, Doug L. James, and Steve Marschner. 2017. Animating Elastic Rods with Sound. *ACM Transactions on Graphics* 36, 4 (July 2017). https://doi.org/10.1145/3072959.3073680

Andrew Selle, Michael Lentine, and Ronald Fedkiw. 2008. A mass spring model for hair simulation. *ACM Transactions on Graphics (Proc. ACM SIGGRAPH'08)* 27, 3 (2008), 1–11. https://doi.org/10.1145/1360612.1360663

Alvin Shi, Haomiao Wu, Jarred Parr, A. M. Darke, and Theodore Kim. 2023. Lifted Curls: A Model for Tightly Coiled Hair Simulation. *Proc. ACM Comput. Graph. Interact. Tech.* 6, 3, Article 42 (aug 2023), 19 pages. https://doi.org/10.1145/3606920

John M Snyder, Adam R Woodbury, Kurt Fleischer, Bena Currin, and Alan H Barr. 1993. Interval methods for multi-point collisions between time-dependent curved surfaces. In *Proceedings of the 20th annual conference on Computer Graphics and interactive techniques*. 321–334.

Jonas Spillmann and Matthias Teschner. 2007. CoRdE: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *ACM SIGGRAPH - EG Symposium on Computer Animation (SCA'07)*. ACM-EG SCA, 63–72.

Shinjiro Sueda, Garrett L Jones, David IW Levin, and Dinesh K Pai. 2011. Large-scale dynamic simulation of highly constrained strands. In *ACM SIGGRAPH 2011 papers*. 1–10.

Brook Taylor. 1717. *Methodus incrementorum directa & inversa*. Inny.

Matthias Teschner, Stefan Kimmerle, Bruno Heidelberger, Gabriel Zachmann, Laks Raghupathi, Arnulph Fuhrmann, M-P Cani, François Faure, Nadia Magnenat-Thalmann, Wolfgang Strasser, et al. 2005. Collision detection for deformable objects. In *Computer graphics forum*, Vol. 24. Wiley Online Library, 61–81.

Thibault Tricard, Vincent Tavernier, Cédric Zanni, Jonàs Martínez, Pierre-Alexandre Hugron, Fabrice Neyret, and Sylvain Lefebvre. 2020. Freely orientable microstructures for designing deformable 3D prints. *ACM Transactions on Graphics* (Dec. 2020). https://doi.org/10.1145/3414685.3417790

Gino Van Den Bergen. 2003. *Collision detection in interactive 3D environments*. CRC Press.

Brian Von Herzen, Alan H Barr, and Harold R Zatz. 1990. Geometric collisions for time-dependent parametric surfaces. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques*. 39–48.

Li Vu-Quoc and S Li. 1995. Dynamics of sliding geometrically-exact beams: large angle maneuver and parametric resonance. *Computer methods in applied mechanics and engineering* 120, 1-2 (1995), 65–118.

Peter Wriggers and Tod A Laursen. 2006. *Computational contact mechanics*. Vol. 2. Springer.

Paul Zhang, Zoë Marschner, Justin Solomon, and Rasmus Tamstorf. 2023. Sum-of-Squares Collision Detection for Curved Shapes and Paths. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) *(SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, Article 76, 11 pages. https://doi.org/10.1145/3588432.3591507

Changxi Zheng and Doug L. James. 2011. Toward High-Quality Modal Contact Sound. *ACM Trans. Graph.* 30, 4, Article 38 (jul 2011), 12 pages. https://doi.org/10.1145/2010324.1964933

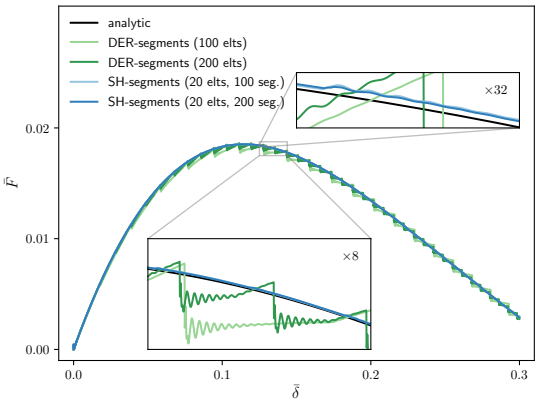


Fig. 19. Three-point bending force-indentation results: influence of segment discretisation for collision detection.

A CURVED THREE-POINT BENDING TEST: ANALYTIC CURVES

In the naturally curved scenario, the initial constraints are slightly modified to take into account the natural curvature κ_0 of the rod. The resulting equations no longer offer the possibility to eliminate a variable to get two independant equations. Instead, the following system of two equations and three unknowns must be solved as a whole:

$$\begin{cases} \int_0^\alpha \frac{\cos(\alpha-\theta)d\theta}{\sqrt{\bar{\kappa}_0^2 - \tilde{F}\frac{\sin\theta}{\cos\alpha}}} &= \frac{1}{2} - \bar{D}\sin\alpha \\ \int_0^\alpha \frac{\sin(\alpha-\theta)d\theta}{\sqrt{\bar{\kappa}_0^2 - \tilde{F}\frac{\sin\theta}{\cos\alpha}}} &= \bar{\delta} + \bar{D}\cos\alpha + \frac{1}{\bar{\kappa}_0} - \sqrt{\left(\frac{1}{\bar{\kappa}_0} + \bar{D}\right)^2 - \frac{1}{4}} \end{cases} \tag{A1}$$

where $\bar{\kappa}_0 = \kappa_0 \Delta$ and the three unknowns are α, $\bar{\delta}$ and $\tilde{F} \equiv F\Delta^2/B = 48\bar{F}$.

In the simulations, the controlled quantity is the displacement δ and the measured quantity is the force F. To get a similar parametrisation in the analytic case, we keep as unknowns α the angle at which the contact is made with the support and $\tilde{F} = 48\bar{F}$ the normalised force up to a numeric factor. We fix $\bar{\delta}$ and solve the resulting system of two equations and two unknowns. The process is repeated over the range of $\bar{\delta} \in [0.0, 0.3]$.

In practice the system is solved using the `root` function from the *scipy* library. The initial seed is obtained from geometric consideration, and the following steps use the previous result as seed for the new system. To avoid the singularities, this method is only used for $\kappa_0 \geq 0.1$ and $\alpha < \frac{\pi}{2}$.

B INFLUENCE OF THE DISCRETISATION ON THE FORCE DISCONTINUITIES

We further evaluate the role of a low-order discretisation in the collision detection and contact response by varying the number of segments used in the *DER-segments* and *SH-segments* models. Note that while the proxy discretisation only affects collision detection for the latter model, it also impacts the underlying mechanical model in the former case, as the "discrete elastic" segments are naturally used as both collision and rod primitives. Figure 19 shows the impact of increasing the number of segments (the length of the rod being fixed) on the indentation force for the three-point bending setup presented in section 4. As expected, we observe that as the number of segments doubles, so does the frequency of force jumps for both *DER-segments* and *SH-segments*, which confirms that the apparent discontinuities are precisely created by segment-segment junctions.

Increasing the number of segments also seems to decrease the amplitude of the jumps, albeit non linearly. The convergence of the amplitude of discontinuities under refinement is further studied in fig. 20, where we show the evolution of the L_1 norm of the force derivative, normalised by its value in the most refined case, as a function of the normalised segment length n_{seg}^{max}/n_{seg}. We vary the number of segments from $n_{seg}^{min} = 60$ to $n_{seg}^{max} = 400$, and plot the results in log-log scale, to exhibit the characteristic power-law convergence rates.

We observe sub-linear convergence rates for both segments-based models, though much smaller for the low-order *DER-segments* model. While this suggests that increasing the number of segments used to detect collisions could definitely increase the accuracy of the force response by ironing out spurious discontinuities, we should stress that such low-order detection approach would not recover C^1 smoothness at any finite segment size. Furthermore, note that the complexity of collision detection ranges from $O(n \log n)$ to $O(n^2)$, highlighting the inefficiency of low-order refinement as a strategy to control force responses.

C INFLUENCE OF THE CONTACT RESPONSE

For the sake of completeness, we investigate the role of the contact response method in the regularity of the forces, by considering the same low-order Discrete Elastic Rod model presented above, but now coupled to the penalty-based Incremental Potential Contact method [Li et al. 2020a], which enforces the non-penetration condition through an implicit energy barrier. We use in practice the DER+IPC

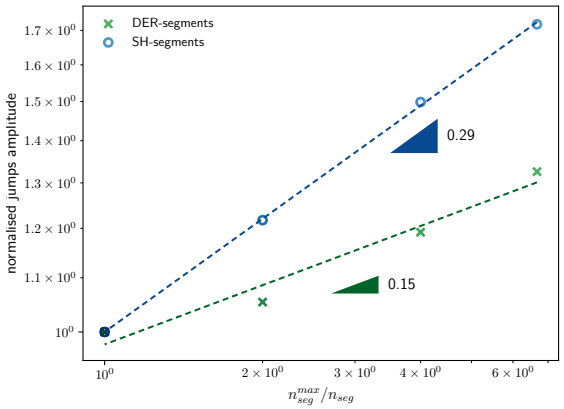


Fig. 20. Evolution of the amplitude of force jumps under refinement: log-log plot of the L_1 norm of the force derivative normalised by its value in the most refined case as a function of the normalised segment length. Segment lengths correspond to 0.1cm, 0.06cm, 0.03cm and 0.015cm for a rod of length 6cm.

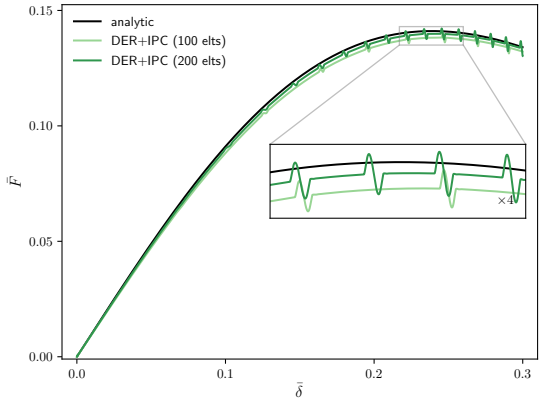


Fig. 21. Three-point bending scenario: force-indentation results obtained with the segment-based DER method, coupled to the Incremental Potential Contact (IPC) method to solve for the contact constraints.

coupling from Li et al. [2020b], implemented in the `codim-ipc` open-source software, and denoted *DER-IPC* in the following.

To evaluate potential force jumps in a controlled configuration, we focus on the frictionless classic ($\kappa_0 = 0$) three-point bending scenario presented in section 4, and again plot the non-dimensional indentation force force $\bar{F} = \frac{F\Delta^2}{48B}$ as a function of the normalised indentation $\bar{\delta} = \frac{\delta}{\Delta}$. The results obtained for *DER-IPC*, with respectively 100 and 200 elements, are shown in fig. 21, and demonstrate that the use of a "smoother" contact response such as IPC does actually not remove the jumps in the forces, as these are inherent to the underlying geometrical discontinuities.

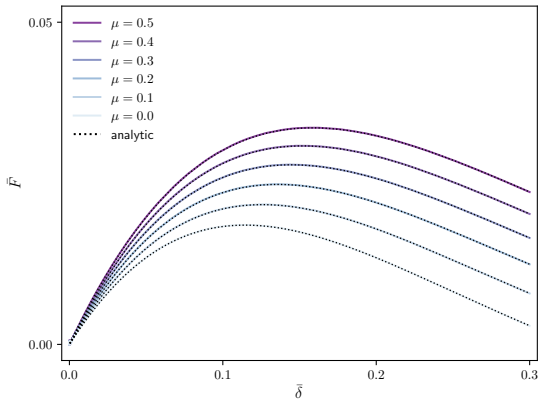


Fig. 22. Curved three-point bending test: dimensionless force-indentation curves for *SH-segments*, as the friction coefficient μ is increased (light to dark).

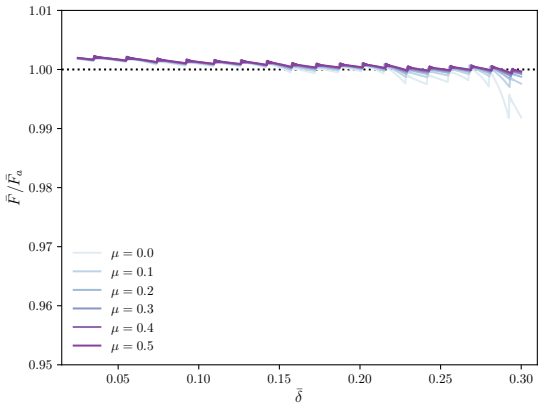


Fig. 23. Curved three-point bending test: noisy force results. Normalised indentation force $\frac{\bar{F}}{\bar{F}_a}$ for *SH-segments* to highlight the presence of force jumps, due to segment-based detection.

D FRICTIONAL THREE POINT BENDING TEST

We evaluate the role of friction in the collision detection and contact response by adding a friction coefficient between the fibre and the support in the three point bending experiment.

The analytic force needs some modification to take into account the effect of friction. The development in the curved case relies on geometric consideration, developed in Batista [2015].

In our case we fix the natural curvature to $\kappa_0 = 0.3$ and vary the friction coefficient μ between 0.0 and 0.5. A very good agreement is obtained between the theoretical and simulated curves, as shown in fig. 22.

As for the frictionless case, despite the seemingly smooth force-indentation curves, spurious discontinuities still occur at proxy-segment junctions in the frictional case, as highlighted by rescaling

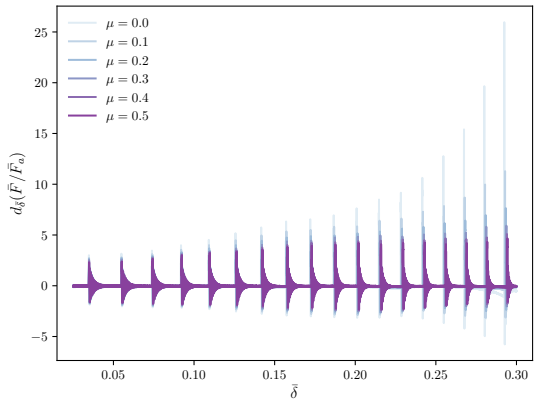


Fig. 24. Curved three-point bending test: analysis of the force discontinuities. Derivatives of the force-indentation curves for SH-segments, as the friction coefficient μ is increased (light to dark).

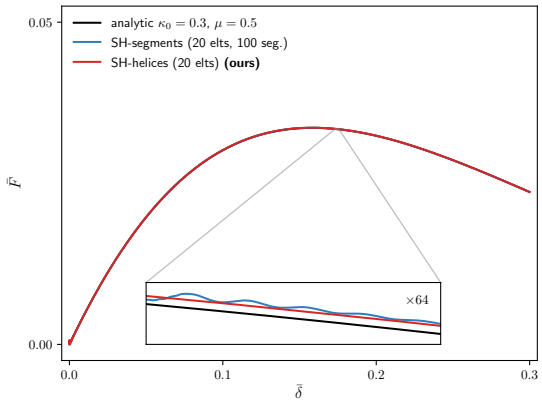


Fig. 26. Curved three-point bending test with our high-order detection scheme: high curvature ($\kappa_0 = 0.3$) and high friction ($\mu = 0.5$) case. Unlike the segment-based detection scheme (in blue), our method (in red) yields a perfectly smooth signal.

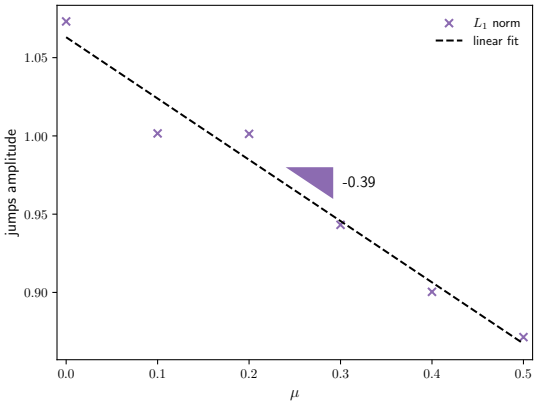


Fig. 25. Curved three-point bending test: analysis of the force discontinuities depending on the friction coefficient. The L_1 norm of the normalised force derivative (which quantifies the amplitude of the jumps) shows a decreasing linear dependency with respect to the friction coefficient at contact.

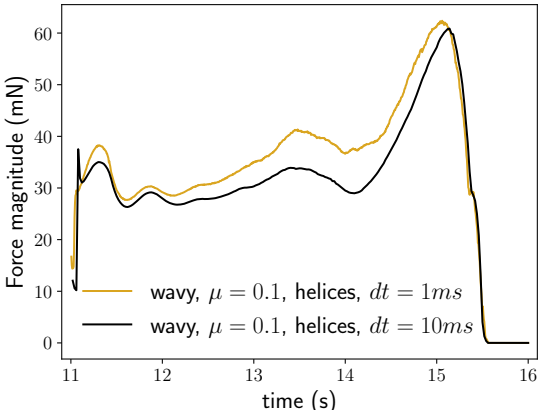


Fig. 27. Increasing the timestep from 1ms to 10ms gives lower forces, which may be explained by the punctal contact losses.

the force to the analytic solution (see fig. 23). Though the amplitude of the jumps appears to decrease with friction (c.f. figs. 24 and 25), the absolute value remains non-negligible even for values as high as $\mu = 0.5$.

In contrast, our new exact contact detection method again completely removes the jumps in the frictional case, as shown in fig. 26.

E DEGRADABILITY

We study the degradability of our method by increasing the timestep on different experiments.

In the three-point bending test, no change in the force is visible with $dt = 10ms$ (×100 compared to original) eventhough we do not perform CCD. With a single contact, so-bogus reaches good precision in a single step ($< 1 \times 10^{-12}$), no matter the required tolerance.

In **Hair combing (1)**, the force still matches pretty well at $dt = 10ms$, (×10 compared to original), though a punctual loss of contact can be observed, which can degrade the solver convergence.

In **Hair combing (2025)** by multiplying the timestep by 10 to get $dt = 10ms$, the **Wavy** wisps converges properly, with an approximate ×10 overall speedup of the simulation, going from $170h$ of simulation down to $18.5h$. However, a slight shift of the curve is observed, see figure 27.