



HAL
open science

An Orthogonal Collocation Method for Static and Dynamic Cosserat Rods

Radhouane Jilani, Pierre-Frédéric Villard, Erwan Kerrien

► **To cite this version:**

Radhouane Jilani, Pierre-Frédéric Villard, Erwan Kerrien. An Orthogonal Collocation Method for Static and Dynamic Cosserat Rods. International Conference on Intelligent Robots and Systems (IROS), IEEE, Oct 2023, Detroit, United States. 10.1109/IROS55552.2023.10341631 . hal-04246775

HAL Id: hal-04246775

<https://hal.science/hal-04246775v1>

Submitted on 17 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Orthogonal Collocation Method for Static and Dynamic Cosserat Rods

Radhouane Jilani*, Pierre-Frédéric Villard*, Erwan Kerrien*

*Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

Abstract—We propose an orthogonal collocation method (CM) for solving Cosserat rod Dirichlet-Neumann boundary value problems in static and dynamic modes. We interpolate the internal loading and collocate the strong form of the differential equations. The method uses Chebyshev polynomials in order to minimize Runge’s phenomenon. The time derivatives are implicitly discretized using the backward differentiation formula BDF- α . We compare our method with the shooting method (SM), multiple shooting method (MSM) and two isogeometric CM against three static and one dynamic applications. The results show that our CM is more stable than SM and faster than MSM.

I. INTRODUCTION

Elastic rods have attracted interest in many fields such as robotics [1], computer graphics [2], [3] and medicine [4]. An application of interest to the current work is the simulation of soft catheter navigation in blood vessels. This requires an accurate, fast, and numerically stable method capable of solving large deflections, contacts, and friction. Cosserat rod theory is widely used to formulate this problem. 3D finite element methods have been developed for its resolution, but they struggle to achieve both real-time performance and accuracy for the full range of states that a catheter passes through during navigation [5]. Therefore, this paper focuses on numerical methods that represent the state of the rod as a continuous function of its arclength that can be recovered by solving a boundary value problem (BVP).

In [6], [7] forward dynamics is solved by discretizing Kirchhoff/Cosserat partial differential equations (PDEs) in time using the backward differentiation formula BDF- α . The resulting BVPs in arclength dimension are solved in every time step using the shooting method (SM). This technique achieved real-time performance in many soft and continuum robots applications. Indeed, SM unknowns are the initial conditions of the ordinary differential equations (ODEs) which provides a relatively small Jacobian matrix to non-linear solvers. However, SM can be unstable in ill-posed shooting problems and may be fundamentally unsuitable for handling contacts at arbitrary locations [8]. The multiple shooting method (MSM) can be used to improve the stability of SM. It was used in [9] for multibody analysis. The method subdivides the BVP interval into multiple subintervals and solves an initial value problem for each one. MSM can be computationally expensive because all initial conditions are unknown at the subintervals and must be computed. Thus, it is impractical for solving dynamics. More stable techniques for solving Cosserat BVPs include collocation methods (CMs). Isogeometric CM was used in [10], [11] to

solve static problems by using NURBS to approximate centerline position and orientation. Due to NURBS smoothness, first and second derivatives of the approximated variables are computed which allows to collocate the strong form of the governing equations. This work was extended in [12] to tackle contacts and friction in dynamic mode by additionally solving for velocity and angular velocity using an implicit time integration. In [13] the linear and angular accelerations are approximated and a Newmark time integration is used. A significant advantage of isogeometric methods lies in their inherent compatibility with CAD systems, enabling a direct and efficient integration. However, in order to minimize Runge’s phenomenon, orthogonal CM which uses Chebyshev polynomials and their roots as collocation points, can be used. The Kirchhoff model was solved in [14] with collocation on curvature and internal forces. This work shows that for beam under vertical tip force experiment, the error decays exponentially with increasing the number of collocation points. In [15] the internal force and moment are approximated and the strong form of the equilibrium equations is collocated. The Kirchhoff model was solved in [16] with collocation on curvature. This work uses the Magnus expansion for arclength integration which allows to get the Jacobian in closed form. This method was compared to SM but only for applications that require low order polynomials (less than 11). All these methods were presented in the static case.

To the best of our knowledge, Chebyshev CM for solving Cosserat forward dynamics collocated on the strong form of PDEs have not been explored yet. Motivated by this and the limitations of SM, we propose a Chebyshev CM for solving Dirichlet-Neumann BVPs in static and dynamic modes. We interpolate the internal forces and moments and collocate the strong form of the differential equations. The time derivatives are computed using the implicit BDF- α as described in [7]. We compare time computation, precision and stability of our method against SM in three static and one dynamic applications. We include experiments that require high polynomial order. In addition, the dynamic numerical application is compared to the results of [13].

In section II Cosserat static ODEs and dynamic PDEs are introduced. Chebyshev approximation is briefly described in section III. Section IV shows how to solve static and dynamic Cosserat BVPs using CM. Numerical applications are presented in section V.

II. COSSERAT MODEL

Cosserat rod is described as a one-dimensional framed curve defined by the parameter $s \in [0, L]$, where L represents the length before strain. Unlike the Kirchhoff rod theory, this model allows for shear and extension. The rod is characterized by its centerline position, $p(s) \in \mathbb{R}^3$ and its orientation, $R(s) \in SO(3)$. The cross-sections are assumed to remain undeformed under loading [10].

Following the notation in [7], a partial derivative of a vector y w.r.t s is denoted y_s and w.r.t to time t is denoted y_t . The ODEs can be written as follows [7]:

$$\begin{aligned} n &= RK_{se}(v - v^*) \\ m &= RK_{bt}(u - u^*) \\ p_s &= Rv \\ R_s &= R\hat{u} \\ n_s &= -f \\ m_s &= -p_s \times n - l \end{aligned} \quad (1)$$

Linear elastic constitutive law is used. n and m are respectively the internal forces and moments in the inertial frame. The first derivative of p w.r.t s and the curvature vector, both in the local frame, are v and u . $(\hat{\cdot})$ is the skew-symmetric matrix operator. The initial configuration of v and u are v^* and u^* . The stiffness matrices for shear and extension and for bending and torsion are $K_{se} = \text{diag}(GA, GA, EA)$ and $K_{bt} = \text{diag}(EI_1, EI_2, GI_3)$, where E is Young's modulus, G the shear modulus, A the cross-sectional area, I_1 and I_2 are the second moments of area, and I_3 is the polar second moment of area. The external distributed forces are f and l .

Following Antman's [17] derivation, the PDEs can be written as follows [7]:

$$\begin{aligned} n &= K_{se}(v - v^*) + B_{se}v_t \\ m &= K_{bt}(u - u^*) + B_{bt}u_t \\ p_s &= Rv \\ R_s &= R\hat{u} \\ n_s &= \rho AR(\hat{\omega}q + q_t) - f \\ m_s &= \rho R(\hat{\omega}J\omega + J\omega_t) - \hat{p}_s n - l \\ q_s &= v_t - \hat{u}q + \hat{\omega}v \\ \omega_s &= u_t - \hat{u}\omega \end{aligned} \quad (2)$$

Kelvin-Voigt type viscous damping is used [18]. q and ω are respectively the velocity and the angular velocity in the local frame. The damping matrices for shear and extension and for bending and torsion are B_{se} and B_{bt} . ρ is the material density and $J = \text{diag}(I_1, I_2, I_3)$ is the second mass moment of inertia tensor. The Cosserat equations are reduced to the Kirchhoff model when $v = [0, 0, 1]^T$ and $v_t = 0$. It can be used in applications where the effects of shear and extension are negligible [6].

For simplicity, we parametrize the rotations $R(s)$ by rotation matrices. They may not remain orthonormal during the spatial numerical integration. However, we use a high-order integration scheme to ensure that the matrices remain sufficiently close to $SO(3)$. To avoid this issue, alternative methods including the modified Rodrigues parameters (MRP) with its shadow set (SMRP) [9] and the Magnus expansion [16] could be used.

III. CHEBYSHEV APPROXIMATION

In this section, the Chebyshev approximation used in our CM is briefly presented. The Chebyshev polynomials of the first kind of order N can be computed from this recurrence relation:

$$\begin{aligned} T_0(x) &= 1, \quad T_1(x) = x \\ T_N(x) &= 2xT_{N-1}(x) - T_{N-2}(x), N > 1 \end{aligned} \quad (3)$$

$x \in [-1, 1]$ is used, thus $T_N(x) \in [-1, 1]$. A mapping is employed in order to use the polynomials in $[0, L]$. Chebyshev roots $c_i \in (-1, 1)$ are chosen to be the interpolation nodes in order to minimize Runge's phenomenon [16]. A Chebyshev polynomial of order N has N roots:

$$c_{N-i+1} = \cos\left(\frac{\pi(2i-1)}{2N}\right), \quad i = 1, \dots, N \quad (4)$$

c_1 and c_N get closer to -1 and 1 respectively as the order N increases, but never reach them exactly. To avoid interpolation at the boundaries, $[0, L]$ is mapped to $[c_1, c_N]$ using $\Phi(s)$:

$$\Phi(s) = c_1 + \frac{s(c_N - c_1)}{L} \quad (5)$$

Let the mapped interpolation nodes $\tau_i = \Phi^{-1}(c_i) \in [0, L]$. A continuous function $\xi(s) \in \mathbb{R}^M$ can be approximated using its values at τ_i . Let $\Xi = [\xi^j(\tau_i)] \in \mathbb{R}^{N \times M}$, $j = 1, \dots, M$ and $H = \frac{2}{N} [T_k'(c_i)] \in \mathbb{R}^{(N+1) \times N}$, $k = 0, \dots, N$.

$$\xi(s)^T = [(1/2)T_0(\Phi(s)), \dots, T_N(\Phi(s))]H\Xi \quad (6)$$

A different number of interpolation nodes can be assigned for each component of ξ . The choice may depend on their non-linearity. For simplicity, the same number N is used in section III and IV. $\Xi_s = [\xi_s^j(\tau_i)]$ can be computed using the Chebyshev differentiation matrix $D_N \in \mathbb{R}^{N \times N}$ [16]:

$$\Xi_s = D_N \Xi \quad (7)$$

$$D_N(i, j) = \begin{cases} \frac{1}{2} \frac{T_N''(c_i)}{T_N'(c_i)} & \text{if } i = j \\ \frac{T_N'(c_i)}{(\tau_i - \tau_j)T_N'(c_j)} & \text{if } i \neq j \end{cases} \quad (8)$$

T_N' and T_N'' denote the first and second derivatives.

IV. COLLOCATION METHOD

In order to solve BVPs, CM approximate the unknown functions. In order to satisfy the differential equations and the boundary conditions (BCs), a specific set of collocation points are selected and their values at the unknown functions are iteratively computed using a non-linear solver. The choice of collocation points is important as it determines the stability of the method [10]. The following subsections describe how the orthogonal CM is used to solve static and dynamic Cosserat problems.

1) *Static*: We solve Cosserat Dirichlet-Neumann BVPs by approximating the unknowns n and m . Thus, $\xi = [n^T, m^T]^T \in \mathbb{R}^6$. The collocation points are chosen to be the same as the interpolation nodes τ_i . The method begins by choosing N . Then τ_i (4)(5), H and D_N (8) are computed. Next, the objective is to compute $n(\tau_i)$ and $m(\tau_i)$ (Ξ) that satisfy ODEs and BCs. We create two residuals $r_n \in \mathbb{R}^{N \times 3}$ and $r_m \in \mathbb{R}^{N \times 3}$. In $\tau_i, i = 1 \dots N - 1$, the strong form of ODEs is evaluated (1):

$$\begin{aligned} r_{n,i} &= n_s(\tau_i) + f(\tau_i) = 0 \\ r_{m,i} &= m_s(\tau_i) + p_s(\tau_i) \times n(\tau_i) + l(\tau_i) = 0 \end{aligned} \quad (9)$$

$n_s(\tau_i)$ and $m_s(\tau_i)$ are computed using D_N (7). $p_s(\tau_i)$ are computed by integrating the rotations $R(s)$ with the initial condition $\psi = [\bar{p}(0)^T, \bar{R}(0)^T, n(\tau_1)^T, m(\tau_1)^T]^T$, where $\bar{p}(0)$ and $\bar{R}(0)$ are the base BCs. An explicit Runge-Kutta of order 4(5) (RK45) is used for integration. $n(s)$ and $m(s)$ are needed for integrating $R(s)$. They are computed via interpolation (6). Finally, $n(\tau_N)$ and $m(\tau_N)$ are evaluated in order to satisfy the tip BCs:

$$\begin{aligned} r_{n,N} &= n(\tau_N) - \bar{n}(\tau_N) = 0 \\ r_{m,N} &= m(\tau_N) - \bar{m}(\tau_N) = 0 \end{aligned} \quad (10)$$

$\bar{n}(\tau_N)$ and $\bar{m}(\tau_N)$ are the tip BCs. The solution of r_n and r_m can be computed using a nonlinear root-finding solver such as Levenberg-Marquardt.

2) *Dynamic*: Similar to the static case, we solve Dirichlet-Neumann BVPs in forward dynamic by approximating the unknowns n and m . In $\tau_i, i = 1 \dots N - 1$, the strong form of PDEs is evaluated (2) (τ_i are omitted for simplicity):

$$\begin{aligned} r_{n,i} &= n_s - \rho A R(\dot{\omega} q + q_t) + f = 0 \\ r_{m,i} &= m_s - \rho R(\dot{\omega} J \omega + J \omega_t) + \hat{p}_s n + l = 0 \end{aligned} \quad (11)$$

$R(s), q(s)$ and $\omega(s)$ are computed by integration, using the initial condition $\psi = [\bar{p}(0)^T, \bar{R}(0)^T, n(\tau_1)^T, m(\tau_1)^T, \bar{q}(0)^T, \bar{\omega}(0)^T]^T$, where $\bar{q}(0)$ and $\bar{\omega}(0)$ are the known initial condition for velocity and angular velocity. We use RK45 for spatial integration. In the dynamic problem q_t, ω_t, v_t and u_t appear. They are discretized using the implicit BDF- α as described in [7]. In a time step j , $Z_t = [q_t^T, \omega_t^T, v_t^T, u_t^T]^T$ can be approximated as follows:

$$\begin{aligned} {}^{(j)}Z_h &= \beta_1 {}^{(j-1)}Z + \beta_2 {}^{(j-2)}Z + \gamma_1 {}^{(j-1)}Z_t \\ {}^{(j)}Z_t &= \beta_0 {}^{(j)}Z + {}^{(j)}Z_h \end{aligned} \quad (12)$$

Where Z_h is the history part of Z . Therefore, the derivatives depend on the solution of the last two time steps. On the first time step ${}^{(j-1)}Z$ is set to a known initial configuration which is usually static and ${}^{(j-1)}Z_t, {}^{(j-2)}Z$ can be set to 0. $\beta_0, \beta_1, \beta_2$ and γ_1 depend on the size of the time step δt and $\alpha \in [-0.5, 0]$:

$$\begin{aligned} \beta_0 &= (1.5 + \alpha) / [\delta t(1 + \alpha)] \\ \beta_1 &= -2 / \delta t \\ \beta_2 &= (0.5 + \alpha) / [\delta t(1 + \alpha)] \\ \gamma_1 &= \alpha / (1 + \alpha) \end{aligned} \quad (13)$$

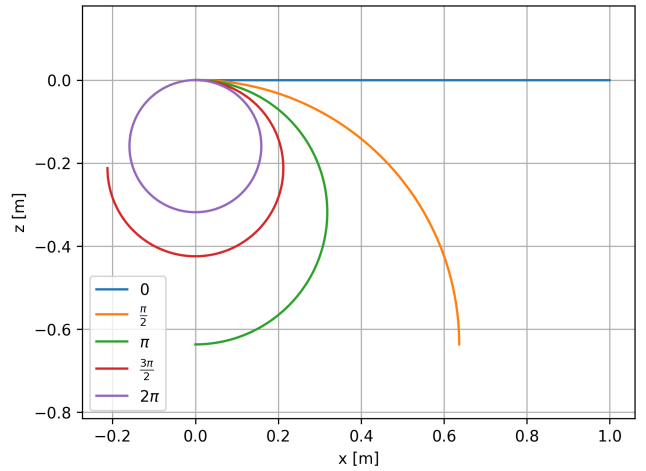


Fig. 1. Example of the circular bending application with 5 tip moments parametrized using $\lambda \in [0, 2\pi]$.

α allows to interpolate between the trapezoidal method at $\alpha = -0.5$ that exhibits very little damping and BDF2 (the second order formula) at $\alpha = 0$ [6]. The BDF- α is $O(\delta t^2)$ accurate. However, δt cannot be too small due to the implicit nature of the discretization that can cancel out two nearly equal values because of limited floating points precision [6]. Finally, (10) are included to evaluate the tip BCs. Because of the varying size of the spatial integration steps used by RK45, we should be able to compute $Z_h(s)$ at arbitrary locations. We use Chebyshev interpolation (6) ($\xi = Z_h$).

V. NUMERICAL APPLICATIONS

In this section, we compare the accuracy, stability, and computation time of CM and SM in static and dynamic applications. All methods were implemented in Python 3.8.10. We use Scipy's root-finding *hybr*, which is a modification of Powell's hybrid method as implemented in MINPACK. We have set the termination tolerance to 10^{-8} for all applications. The initial guesses were set to 0 for the first step and to the solution of the last iteration for the following steps, unless specified otherwise. We use Scipy's RK45 for integration. For orientations, SM uses rotation matrices, while MSM uses MRP and SMRP [9]. Furthermore, SM uses BDF- α for time derivatives discretization [7]. The code was executed on an i9-10900K 3.70Ghz CPU. In this section, the first three applications are static, while the fourth is dynamic.

A. Circular bending

In the first numerical example, we compare the different methods to an analytical solution. An initially straight beam directed along the x axis undergoes incremental bending, using a tip moment l_2 on the y axis, until it deforms into a full circle [9] (Fig. 1). The mechanical properties are: $L = 1$ m, $A = 0.25$ m², $I_1 = I_2 = 2I_3 = 2$ m⁴, $E = 1$ N/m² and Poisson's ratio $\nu = 0.3$. $l_2 = \lambda \frac{EI_1}{L}$, where λ takes 100 equally spaced values from 0 to 2π . The final shape of the beam corresponds to $\lambda = 2\pi$. The error is computed between

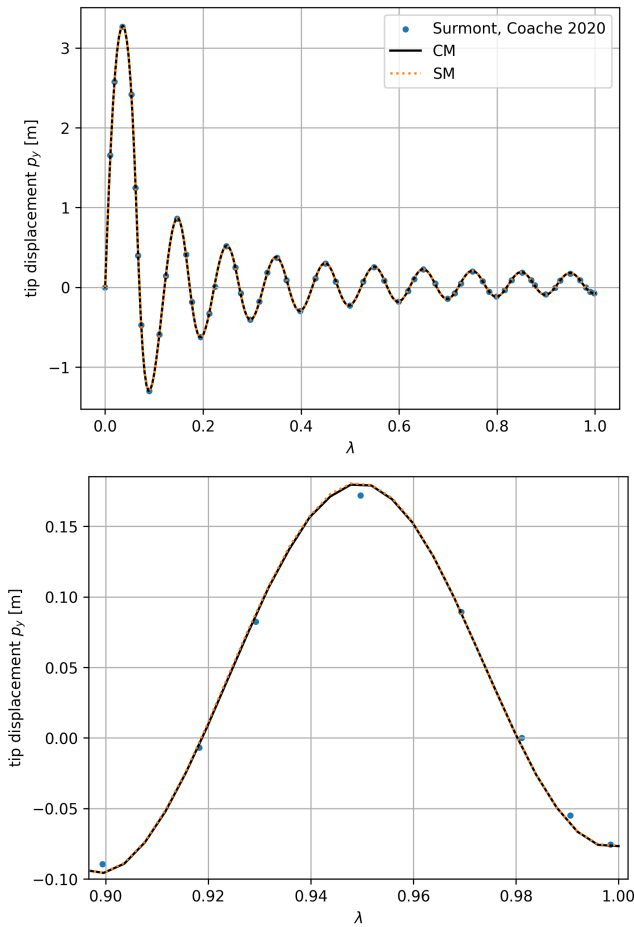


Fig. 2. Helical bending application: external loading parametrized using λ are applied to the tip incrementally. Bottom: a close-up of the last cycle.

the tip displacement of Cosserat $p(L)$ and the analytical solution ${}^{(a)}p(L)$ provided in [19] as a percentage of the rod length:

$$e_p = \frac{\|p(L) - {}^{(a)}p(L)\|}{L} \times 100 \quad (14)$$

We use $N = N_n = N_m = 2$ for n and m , which is sufficient since they are linear in this problem. The maximum e_p is 0.0412 for CM and 0.0514 for SM. The average e_p is 0.0232 for CM and 0.0281 for SM. The errors are very low and CM is more accurate than SM. Table I displays the execution time for all static applications. SM is twice as fast as CM.

B. Helical bending

The objective of the second application is to validate the use of RK45 for integrating rotation matrices. An initially straight beam along the x axis is subjected to an incremental out-of-plane tip force f_2 and tip moment l_2 along the y axis, causing it to deform into a helical shape [9]. $f_2 = 50\lambda$ and $l_2 = 200\pi\lambda$ are parameterized with $\lambda \in [0, 1]$, which is incremented using 250 equally spaced steps. $L = 10$ m, $K_{bt} = \text{diag}(10^2)$, $K_{sc} = \text{diag}(10^4)$ and $\nu = 0$. We use

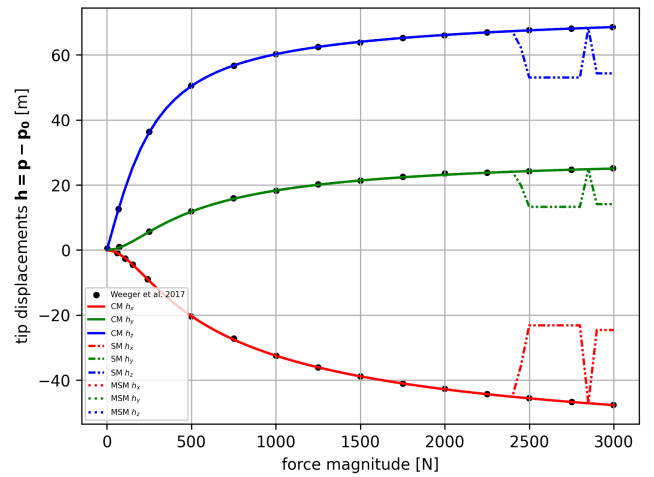


Fig. 3. Tip displacements of 45° bending application. Initial guesses for all iterations were set to 0. SM fails to converge before the end of the experiment.

TABLE I
TIME EXECUTION OF STATIC NUMERICAL APPLICATIONS

Method	Max time (s)	Avg. time (s)
Circular bending		
CM	0.104	0.0766
SM	0.0485	0.0348
Helical bending		
CM	22.3	6.80
SM	1.98	0.327
45° bending: initial guesses = last solution		
CM	0.974	0.821
SM	0.116	0.0784
45° bending: initial guesses = 0		
CM	1.49	1.12
MSM	5.63	1.93

$N_n = 2$ and $N_m = 43$ because the moments are highly non-linear in this problem.

Fig. 2 shows the tip displacement $p_y(L)$ compared to [9]. The results obtained using our approach are in excellent agreement with [9]. Therefore, the use of rotation matrices in combination with a high-order integration scheme can model large rotations while effectively limiting the $SO(3)$ drift off. SM is, on average, 20 times faster than CM. Mainly because CM requires a high polynomial order to approximate the highly non-linear moments.

C. 45° bending

The last static application involves large displacements and multiple coupled deformations. This particular test has been extensively studied in the literature [19], [10], [9] and is regarded as a benchmark case. The beam is curved as one-eighth of a circle with a radius of 100 m along the xy plane. Then, an incremental out-of-plane force f_3 along the z axis is applied on the tip in 60 steps, starting from 0 N and increased by 50 N until it reaches 3000 N [10]. The beam has a square cross-section with unit length for each side. $E = 1 \times 10^7$ N/m², $G = E/2$ and $\nu = 0$. We use $N_m = 18$ and $N_n = 2$. Two scenarios were considered. In the first one, we have

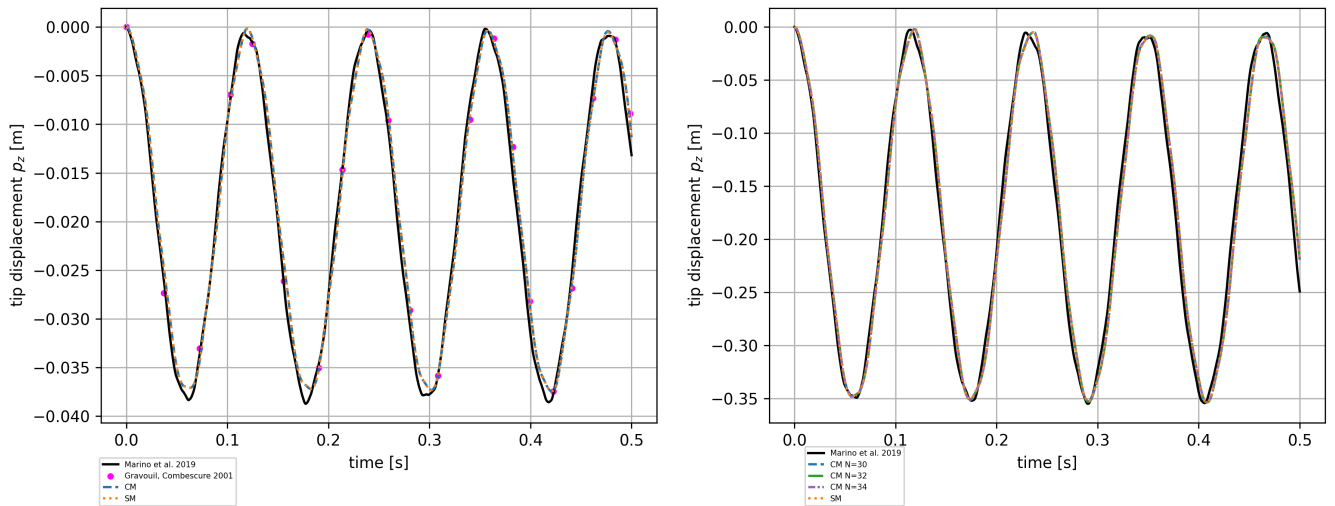


Fig. 4. Dynamic cantilever: a constant force f_3 is applied on the tip. Left: $f_3 = -10$ N. Right: $f_3 = -100$ N.

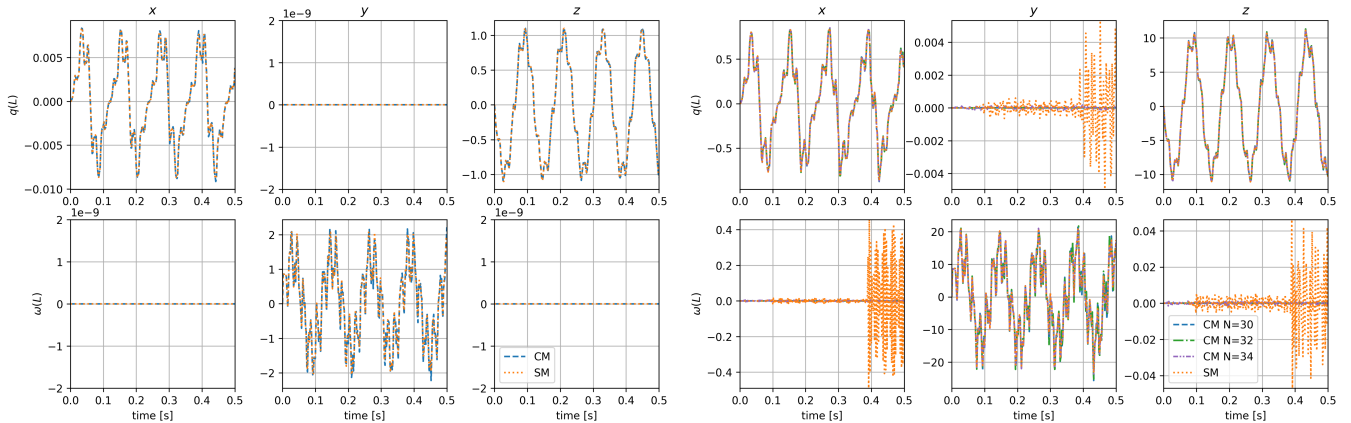


Fig. 5. Dynamic cantilever: tip velocity and angular velocity over time. Left: $f_3 = -10$ N. Right: $f_3 = -100$ N.

set the initial guesses to the solution of the last iteration. Both CM and SM are in excellent agreement with [10]. In the second one, we have set all the initial guesses to 0. We aim to study the impact of the initial guess on the different methods. This test can be considered as a big variation of external loading during a simulation. We compare CM with both SM and MSM with 3 equally spaced sub-intervals. Tip displacements are presented in Fig. 3. Both CM and MSM converged to a correct solution while SM failed to converge little before applying $f_3 = 2500$ N. Furthermore, CM was faster than MSM.

D. Dynamic cantilever

For the dynamic problem, we replicate the experiment conducted in [13], [20], [21]. The initial configuration, at $t = 0$, is a straight beam along the x axis. Then, at $t > 0$, a constant force f_3 is applied at the tip along the z -axis. The simulation lasts for a total of 0.5 s. Following [13], [20], two scenarios are adopted. In the first one, f_3 is set to -10 N, while in the second one f_3 is set to -100 N. The beam has a unit length and a square cross-section with a thickness of

0.01 m. $E = 210 \times 10^9$ N/m², $\nu = 0.2$, $\rho = 7800$ kg/m³. We use $B_{se} = B_{bt} = 0$, $\delta t = 2 \times 10^{-3}$ s and $\alpha = -0.5$ which corresponds to the trapezoidal method. The smallest feasible time step was employed. Utilization of any smaller value resulted in instabilities within the numerical methods. It is worth noting that the minimal time step is identical for both CM and SM.

In the first scenario, we use $N = 16$ and $N_{Z_h} = 10$ for the

TABLE II
TIME EXECUTION OF DYNAMIC NUMERICAL APPLICATIONS

Method	Max time (s)	Avg. time (s)
$f_3 = -10$ N		
SM	0.881	0.304
CM	6.49	3.18
$f_3 = -100$ N		
SM	3.62	1.19
CM N=30	36.2	9.9
CM N=32	66.6	11.7
CM N=34	105.4	14.9

Chebyshev order of Z_h . Fig. 4 (left) shows the tip displacements p_z over time compared to [13], [21]. CM and SM are in excellent agreement with [13], [20], [21]. In the second scenario, we try three orders $N = 30, 32, 34$ and $N_{Z_h} = 16$. Fig. 4 (right) shows the tip displacement p_z over time compared to [13], [20]. We can conclude that the trapezoidal method has very low numerical damping and error. We observed that $N > 34$ had a small effect on the results. However, when N fell below 30, the CM demonstrated instability. This is attributed to the highly nonlinear nature of both force and moment functions, which necessitate an approximation via high-order polynomial. Fig. 5 shows the velocity $q(L)$ and angular velocity $\omega(L)$ at the tip over time for the two scenarios. We observe that CM kept its stability. However, SM was less stable when applied a force $f_3 = -100$ N. The instability can be seen particularly at $t > 0.4$ where it produces inaccurate values of q_y, w_x, w_z that should be close to zero. Table II shows the time execution. SM is on average 10 times faster than CM.

VI. CONCLUSIONS

We have proposed a Chebyshev collocation method on internal loading to solve Cosserat rod Dirichlet-Neumann BVPs in static and dynamic modes. The time derivatives are discretized implicitly using BDF- α . The choice to parameterize the orientations with rotation matrices in conjunction with RK45 has been validated using the helical bending experiment. Compared to the analytical solution in the circular bending experiment, CM was more accurate than SM. When the initial guesses were set far from the correct solution, CM demonstrated better convergence than SM. In the dynamic experiment, CM kept its stability even when a large external tip force was applied, but SM deviated from the correct solution. Furthermore, SM was faster than CM, especially when a high number of collocation points was required. However, CM was faster than MSM even when only 3 subdivisions were used. We have used a constant Chebyshev order that gives correct convergence throughout each application. In the span of some applications, the non-linearity of the internal loading may change. To reduce the computational time, a possible future work is to adapt the Chebyshev order during the simulation. In further work, we are planning to include contacts and friction using external forces exerted on the rod.

ACKNOWLEDGMENT

This work was supported by the French Agence Nationale de la Recherche under Grant ANR-20-CE45-0011. The authors thank Enzo Marino for sharing his data.

REFERENCES

[1] John Till, Vincent Aloï, Katherine E Riojas, Patrick L Anderson, Robert James Webster III, and Caleb Rucker. A dynamic model for concentric tube robots. *IEEE Transactions on Robotics*, 36(6):1704–1718, 2020.

[2] Victor Romero, Mickaël Ly, Abdullah Haroon Rasheed, Raphaël Charronière, Arnaud Lazarus, Sébastien Neukirch, and Florence Bertails-Descoubes. Physical validation of simulators in computer graphics: A new framework dedicated to slender elastic structures and frictional contact. *ACM Transactions on Graphics (TOG)*, 40(4):1–19, 2021.

[3] Jonas Spillmann and Matthias Teschner. Corde: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 63–72, 2007.

[4] Raffaella Trivisonne, Erwan Kerrien, and Stéphane Cotin. Constrained Stochastic State Estimation of Deformable 1D Objects: Application to Single-view 3D Reconstruction of Catheters with Radio-opaque Markers. *Computerized Medical Imaging and Graphics*, 81, February 2020.

[5] Costanza Armanini, Frédéric Boyer, Anup Teejo Mathew, Christian Duriez, and Federico Renda. Soft robots modeling: A structured overview. *IEEE Transactions on Robotics*, 2023.

[6] John Till and D Caleb Rucker. Elastic rod dynamics: Validation of a real-time implicit approach. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3013–3019. IEEE, 2017.

[7] John Till, Vincent Aloï, and Caleb Rucker. Real-time dynamics of soft and continuum robots based on cosserat rod models. *The International Journal of Robotics Research*, 38(6):723–746, 2019.

[8] John Daniel Till. *On the Statics, Dynamics, and Stability of Continuum Robots: Model Formulations and Efficient Computational Schemes*. PhD thesis, University of Tennessee, 2019.

[9] Surmont Florian and Coache Damien. Geometrically exact static 3d cosserat rods problem solved using a shooting method. *International Journal of Non-Linear Mechanics*, 119:103330, 2020.

[10] Oliver Weeger, Sai-Kit Yeung, and Martin L Dunn. Isogeometric collocation methods for cosserat rods and rod structures. *Computer Methods in Applied Mechanics and Engineering*, 316:100–122, 2017.

[11] Oliver Weeger, Bharath Narayanan, Laura De Lorenzis, Josef Kiendl, and Martin L Dunn. An isogeometric collocation method for frictionless contact of cosserat rods. *Computer Methods in Applied Mechanics and Engineering*, 321:361–382, 2017.

[12] Oliver Weeger, Bharath Narayanan, and Martin L Dunn. Isogeometric collocation for nonlinear dynamic analysis of cosserat rods with frictional contact. *Nonlinear Dynamics*, 91:1213–1227, 2018.

[13] Enzo Marino, Josef Kiendl, and Laura De Lorenzis. Explicit isogeometric collocation for the dynamics of three-dimensional beams undergoing finite motions. *Computer Methods in Applied Mechanics and Engineering*, 343:530–549, 2019.

[14] Yan Liu, Mingbin Wang, and Lei Meng. Spectral collocation method in the large deformation analysis of flexible beam. *IAENG International Journal of Applied Mathematics*, 48(4):1–8, 2018.

[15] P Khaneh Masjedi and HR Ovesy. Chebyshev collocation method for static intrinsic equations of geometrically exact beams. *International Journal of Solids and Structures*, 54:183–191, 2015.

[16] Andrew L Orekhov and Nabil Simaan. Solving cosserat rod models via collocation and the magnus expansion. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8653–8660. IEEE, 2020.

[17] SS Antman. *Nonlinear problems of elasticity*, volume 107 of applied mathematical sciences, 2nd edn springer. *New York*, 1, 2005.

[18] Joachim Linn, Holger Lang, and Andrey Tuganov. Geometrically exact cosserat rods with kelvin-voigt type viscous damping. *Mechanical Sciences*, 4(1):79–96, 2013.

[19] Juan C Simo and Loc Vu-Quoc. A three-dimensional finite-strain rod model. part ii: Computational aspects. *Computer methods in applied mechanics and engineering*, 58(1):79–116, 1986.

[20] Enzo Marino, Josef Kiendl, and Laura De Lorenzis. Isogeometric collocation for implicit dynamics of three-dimensional beams undergoing finite motions. *Computer Methods in Applied Mechanics and Engineering*, 356:548–570, 2019.

[21] Anthony Gravouil and Alain Combescure. Multi-time-step explicit-implicit method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 50(1):199–225, 2001.