Skeleton-Guided Rolling-Contact Kinematics for Arbitrary Point Clouds via Locally Controllable Parameterized Curve Fitting

Qingmeng Wen[1], Ze Ji[2], Yu-Kun Lai[3], Mikhail Svinin[4] and Seyed Amir Tafrishi[1]

***Abstract*— Rolling contact kinematics plays a vital role in dexterous manipulation and rolling-based locomotion. Yet, in practical applications, the environments and objects involved are often captured as discrete point clouds, creating substantial difficulties for traditional motion control and planning frameworks that rely on continuous surface representations. In this work, we propose a differential geometry-based framework that models point cloud data for continuous rolling contact using locally parameterized representations. Our approach leverages skeletonization to define a rotational reference structure for rolling interactions and applies a Fourier-based curve fitting technique to extract and represent meaningful controllable local geometric structure. We further introduce a novel 2D manifold coordinate system tailored to arbitrary surface curves, enabling local parameterization of complex shapes. The governing kinematic equations for rolling contact are then derived, and we demonstrate the effectiveness of our method through simulations on various object examples.**

I. Introduction

Rolling is a motion type common in numerous robotic applications [1], [2], spanning from spherical rolling robots [3], [4] to fine manipulation tasks [5], [6]. The foundational relationship between rolling motion and surface curvature was well-formulated by Montana [7] for robotics applications. Nevertheless, the majority of prior research emphasizes continuous-time motion planning and control strategies [8]–[10], which often fail to align with practical implementation challenges, especially since sensor data is inherently discrete, commonly obtained as point clouds via LiDAR or RGB-D sensors. Moreover, analyzing rolling dynamics over unknown and arbitrary geometries demands identifying a reference contact point, a task complicated when the contact location is not directly observable [11]. Despite these complexities, current literature lacks a unified framework for deriving continuous rolling kinematics from discrete, real-world point cloud inputs.

Rolling contact kinematics defines the relationship between two rolling bodies based on their curvature properties [12]. In an underactuated model with five states—two for each body in local coordinates and a spin angle between them—the kinematics is governed by velocity dependencies, including angular orientation and slippage velocity [13]. Initially, this model attracted significant attention from mathematicians and roboticists, primarily for studying motion planning problems in the continuous domain [2], [8], [10]. Rolling systems have been widely explored in applications such as disks, wheels, and omnidirectional rolling robots [2]. This concept has also been extended to more complex problems, such as dexterous manipulation, where a grasping mechanism's fingertips roll against an arbitrary object to enable unique motion capabilities [6]. However, despite the promising applications, sensing and modeling rolling objects in real-world scenarios remain challenging [11]. LiDAR and RGB-D sensors capture surfaces as point clouds, but linking these discrete models to continuous rolling kinematics remains challenging, especially for deformable or curved objects. Deriving a mesh representation from a point cloud or using a signed distance field to represent the object shape might be a solution. However, deriving those representations is typically computationally heavy, and requires high-quality and complete data input [14], [15].

In addition to the lack of integration between discrete point clouds and continuous rolling contact kinematics, a key challenge lies in identifying the center of rotation or a reliable moving frame–a quantity that is often unknown or difficult to define in real-world scenarios. While assuming a fixed center simplifies analysis, such an assumption restricts applicability to highly idealized or symmetric geometries, making it unsuitable for most freeform surfaces encountered in practice. Curve skeletonization offers a promising alternative by extracting medial curves that capture the intrinsic structure of 3D shapes and can serve as potential motion references [16], [17]. Although skeletons have been widely employed in graphics and robotics for tasks such as shape abstraction and motion planning [18], [19], their use as a structural basis for rolling contact modeling remains largely unexplored. Leveraging skeletons as dynamic reference frames not only mitigates the difficulty of defining rotation centers but also provides a geometric foundation to bridge discrete shape data and continuous kinematic models. This motivates the development of a principled framework that connects skeleton-guided slicing with differentiable surface modeling and rolling motion analysis.

In this work, we propose a novel approach for formulating skeleton-based rolling contact kinematics on arbitrary object surfaces, using Fourier curve fitting in local coordinate form to achieve continuous and differentiable reconstructions from raw point cloud inputs. Our contributions are summarized as follows:

- We propose a new point cloud slicing method based on curve skeletonization and geometric curvature, enabling structured cross-sectional analysis.
- We develop a local surface reconstruction model from discrete point clouds using Fourier-based curve fitting, providing a smooth and differentiable sectional surface representation.

[1]Qingmeng Wen and Seyed Amir Tafrishi are with Geometric Mechanics and Mechatronics in Robotics (gm^{2}R) Lab, School of Engineering, Cardiff University, Queen's Buildings, The Parade, Cardiff, CF24 3AA {wen1, tafrishisa}@cardiff.ac.uk

[2]Ze Ji is with the School of Engineering, Cardiff University, CF24 3AA, UK iz1@cardiff.ac.uk

[3]Yu-Kun Lai is with School of Computer Science and Informatics, Cardiff University, CF24 4AG, UK. laiy4@cardiff.ac.uk

[4]Mikhail Svinin is with Information Science and Engineering Department, Ritsumeikan University, Shiga, Japan svinin@fc.ritsumei.ac.jp

- We introduce a differential geometry-based rolling contact model formulated in locally parametrized coordinates, suitable for general semi-convex surfaces and check the controllability.
- We validate the proposed framework through extensive simulations on real-world object data.

The paper is structured as follows: Section II presents the preliminaries on point cloud representations, skeleton extraction, and the proposed slicing strategy for curve fitting. Section III details the generalized rolling contact model tailored to locally parametrized surfaces and its integration with the sliced and fitted surface representations. Section IV discusses simulation results, demonstrating the effectiveness of our approach across a variety of object geometries.

II. Point Cloud Skeletonization, Slicing and Curve Fitting

A. Skeletonization of Point Cloud Surfaces

An object's point cloud is typically represented as a discrete set of 3D points, $\mathbf{P} = \mathbf{p}_i \mid \mathbf{p}_i \in \mathbb{R}^3, i = 1, \ldots, n$. Its curve skeleton is modeled as a graph $G_s = (V, E)$, where $V \subseteq \mathbb{R}^3$ is a set of vertices connected by edges $e_i \in E$. The process of generating such a skeleton is referred to as *curve skeletonization*. In this work, we adopt a Laplacian-based approach [20]. The method begins by identifying local neighborhoods using the k-nearest neighbors (KNN) algorithm, which are then refined via Delaunay triangulation. Principal Component Analysis (PCA) is subsequently applied to extract ring structures from the triangulated neighborhoods.

Using the neighbor ring information, the Laplacian matrix $\mathbf{L} \in \mathbb{R}^{n \times n}$ is defined as:

$$L_{ij} = \begin{cases} \omega_{ij} = \cot \alpha_{ij} + \cot \beta_{ij}, & \text{if } \mathbf{p}_j \in \boldsymbol{\mu}_i; \\ -\sum_{k \in \boldsymbol{\mu}_i} \omega_{ik}, & \text{if } i = j; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

where $\boldsymbol{\mu}_i$ is the ring point set of $\mathbf{p}_i$, and α_{ij}, β_{ij} are the opposite angles of edge (i, j) in the point ring triangles. The contracted cloud points at iteration $k+1$ are obtained by solving:

$$\begin{bmatrix} \mathbf{W}_L \mathbf{L} \\ \mathbf{W}_H \end{bmatrix} \mathbf{P}_{k+1} = \begin{bmatrix} \mathbf{0} \\ \mathbf{W}_H \mathbf{P}_k \end{bmatrix}, \tag{2}$$

where $\mathbf{W}_L, \mathbf{W}_H \in \mathbb{R}^{n \times n}$ are contraction and attraction weights, jointly balancing the movement of points toward the medial axis. These weights are updated iteratively to ensure continuous contraction [20]. The process runs until a termination criterion—based on global or local point cloud features—is met [17], [20].

The resultant contracted points form an approximation of the medial surface. This is further reduced to skeleton vertices using farthest point sampling. Connections among vertices are inferred from neighbor ring information, followed by refinement to produce the final *curve skeleton*.

B. Slicing Algorithm and Fourier Curve Fitting

We begin by introducing the general slicing algorithm that leverages curve skeleton information to reconstruct local geometric surfaces of an object point cloud. This algorithm enables a structured approach to local surface modeling by defining cross-sectional slices guided by the curve skeleton and fitting them with a continuous representation.

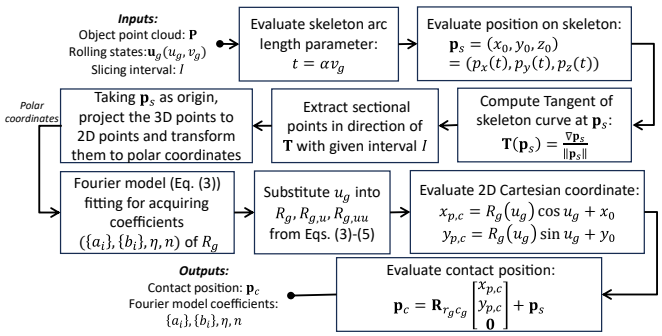


Fig. 1: Slicing algorithm and Fourier curve fitting process

As illustrated in Fig. 1, the slicing and modeling pipeline starts with the extraction of cross-sectional shapes based on a point v_g along the object's skeleton. This point determines the slicing plane, guiding the selection of a local cross-sectional region from the input point cloud $\mathbf{P}$. Each extracted section is then fitted using a Fourier series model, which offers an explicit, differentiable representation of the surface geometry. This representation allows us to define a surface parameterization $\mathbf{u}_g(u_g, v_g)$, where v_g indicates the slicing position along the skeleton and u_g denotes the polar angle in the fitted cross section. This parameterization enables the computation of a contact point $\mathbf{p}_c$ on the object surface and yields a smooth surface approximation from raw point cloud data.

To support kinematic modeling in rolling contact systems, it is essential to define a differentiable manifold over the object surface. While curve skeletonization provides a compact, topology-preserving representation of a shape in 3D space [19], it typically yields a graph-like structure composed of edge-connected vertices. As seen in Fig. 2(a), this discrete representation is not differentiable at vertex junctions, posing challenges for continuous geometric modeling.

To address this, we further smooth the curve skeleton by parameterizing it with respect to arc length. Given a point v_g along the skeleton, we fit a branch of the skeleton using a polynomial curve, adaptively selecting the polynomial order to minimize the mean squared error (MSE) between the fit and the discrete skeleton points. This results in a continuous, differentiable curve model $\theta(t) = \theta(\alpha v_g) = (p_x(t), p_y(t), p_z(t))$, where $t = \alpha v_g$ is the arc length parameter and α is a scalar. Under the assumption that the curve skeleton has either a single path or can be semantically decomposed into separate branches, this modeling approach enables efficient and differentiable slicing of the shape. As shown in Fig. 2(a), a carrot-shaped point cloud can thus be sectioned along the tangent direction of the curve skeleton at predefined intervals.

After modeling an individual curve skeleton branch, a point cloud shape branch can be continuously sectioned, allowing the extraction of corresponding points for sectional shape modeling. As shown in Fig. 2(a), the extracted points from the point cloud shape are marked in red. Assuming the section thickness is sufficiently small, the sectional shape surface can be approximated as a 2D closed shape. Consequently, these extracted points can be projected onto a 2D plane aligned with the tangent at the associated skeleton

point. However, directly modeling a closed shape is challenging. Inspired by [21], we approximate the closed curve of a 2D shape using a Fourier series, as outlined in Fig. 1, where the extracted points are represented in the polar coordinate system. The fitting equations are formulated as a function of the slice radius R_g at the v_g-th slice angle of the considered object by

$$R_g(u_g) = a_0 + \sum_{i=1}^{n} \left[a_i \sin(i\eta u_g) + b_i \cos(i\eta u_g)\right], \quad (3)$$

where u_g is the angle in the polar coordinate system relative to the section center point. The coefficients $\{a_0, a_i, b_i\}$ are Fourier coefficients obtained through curve fitting using a Fourier series model, and n denotes the number of harmonic terms. Additionally, η represents the fundamental frequency. Note that n is chosen adaptively to obtain the best fitting performance by minimizing a loss given by

$$L(R_g(u_g)) = w_1 \text{RMSE} + w_2 \|\mathbf{m}_s - \mathbf{m}_e\|, \quad (4)$$

where RMSE is the root mean squared error of the fitted model $R_g(u_g)$, and $\mathbf{m}_s$, $\mathbf{m}_e$ are the start and end polar coordinates of the final points in the closed shape of the section (see Fig. 2). The two terms, RMSE and point distance, represent fitting loss and disclosure loss, respectively, with w_1 and w_2 as their corresponding weights. The fitting process is considered converged if the loss $L(R_g(u_g)) < \epsilon$, where ϵ is a constant. In practice, the original polar coordinate data of the section points only covers one period, but we perform periodic extension to generate multiple periods of data for curve fitting. Using data from several periods enhances the periodic information and improves accuracy.

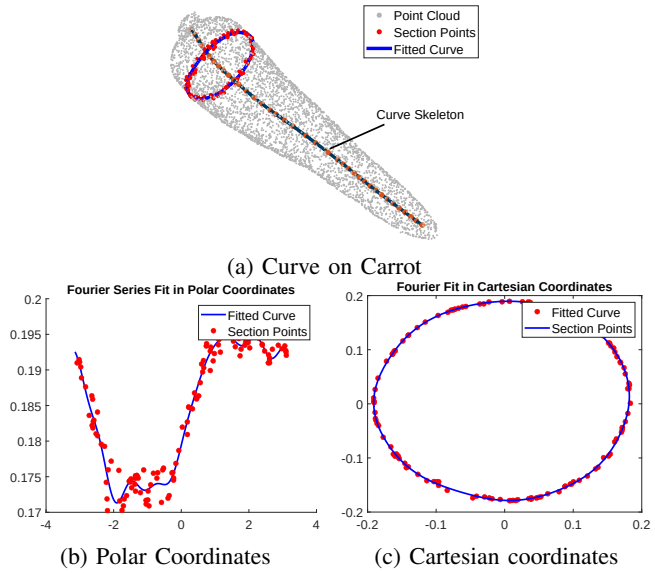
As shown in Fig. 2(b)-(c), fitting with Fourier models recovers detailed curvature variations of the point cloud surface, filtering out noise and providing a smooth curve fitting result for the projected section points in Fig. 2(a). Also, note that the first and second derivatives of the fitted curve model are given by

$$\begin{aligned} R_{g,u} &= \sum_{i=1}^{n} i\eta \left[b_i \cos(i\eta u_g) - a_i \sin(i\eta u_g)\right], \\ R_{g,uu} &= -\sum_{i=1}^{n} i^2\eta^2 \left[b_i \sin(i\eta u_g) + a_i \cos(i\eta u_g)\right]. \end{aligned} \quad (5)$$

III. The Locally Generalized Semi-Convex Surface

This section introduces a continuous 2D manifold coordinate formulated locally for allowing us to determine the kinematics of rolling contact through a curve-fitted cloud point model. To do this, a semi-convex 2D manifold is designed and a rolling object for simplification is considered a sphere.

Firstly, we consider a spin-rolling sphere Σ_S on a geometrically fitted surface Σ_G, as shown in Fig. 3, describing the rolling contact between the sphere's velocity $\dot{\mathbf{u}}_s$ and the geometric surface's velocity $\dot{\mathbf{u}}_g$. Additionally, there exists a relative spin angle ψ between the contact coordinate frames. Note that the contact points of both surfaces, Σ_{CS} and Σ_{CG}, coincide. In this formulation, by considering a sphere with a variable radius as a generalized coordinate for a symmetric surface, we can define as follows



(a) Curve on Carrot

(b) Polar Coordinates (c) Cartesian coordinates

Fig. 2: Point cloud section and fitting

$$\begin{aligned} &f_g : U_G \to \mathbb{R}^3 : \mathbf{u}_g(u_g, v_g) \mapsto \big[R_g(u_g)\cos u_g, \\ &R_g(u_g)\sin u_g,\ \alpha v_g\big], \\ &f_s : U_S \to \mathbb{R}^3 : \mathbf{u}_s(u_s, v_s) \mapsto \big[R_s \cos u_s \cos v_s, \\ &-R_s \cos u_s \sin v_s, R_s \sin u_s\big], \end{aligned} \quad (6)$$

where $U_G = \{(u_g, v_g) | -\pi < u_g < \pi, 0 < v_g < \pi\}, U_s = \{(u_s, v_s) | -\frac{\pi}{2} < u_s < \frac{\pi}{2}, -\pi < v_s < \pi\}$, $\alpha = L_s/\pi$ (L_s is the total skeleton length), R_g is the fitted curve model function as expressed by Eq. (3). For mapping f_g, it is formulated with the concept of local slices, where v_g indicated the selected slice.

We utilized *Gauss frame* to explain the rolling contact kinematics. As explained by [7], the coordinates of Gauss frame is $\mathbf{x}(\mathbf{u}) = f_u(\mathbf{u})/\|f_u(\mathbf{u})\|$, $\mathbf{y}(\mathbf{u}) = f_v(\mathbf{u})/\|f_v(\mathbf{u})\|$, $\mathbf{z}(\mathbf{u}) = g(f(\mathbf{u}))$, where g is an outward normal map of surface points. As there is no explicit outward normal map function for an irregular object surface Σ_G, we derive $\mathbf{z}_g(\mathbf{u_g})$ by $\mathbf{x}_g(\mathbf{u_g}) \times \mathbf{y_g}(\mathbf{u_g})$. Thus, we can derive an orthogonal Gauss frame for object surface Σ_G as:

$$\begin{aligned} &\mathbf{x}_g(\mathbf{u}_g) = (R_g^2 + R_{g,u}^2)^{-1/2}\big[R_{g,u}\cos u_g - R_g \sin u_g, \\ &R_{g,u}\sin u_g + R_g \cos u_g,\ 0\big]^T, \quad \mathbf{y}_g(\mathbf{u_g}) = [0, 0, 1]^T, \\ &\mathbf{z}_g(\mathbf{u_g}) = (R_g^2 + R_{g,u}^2)^{-1/2}\big[R_{g,u}\sin u_g + R_g \cos u_g, \\ &R_g \sin u_g - R_{g,u}\cos u_g,\ 0\big]^T. \end{aligned} \quad (7)$$

By substituting metric, curvature form, torsion form respectively explained as $\mathbf{M}_g = \text{diag}(\|f_{g,u}(\mathbf{u}_g)\|, f_{g,v}(\mathbf{u}_g)\|)$, $\mathbf{K}_g = \big[\mathbf{x}_g, \mathbf{y}_g,\big]^T[\mathbf{z}_{g,u}/\|f_{g,u}(\mathbf{u}_g)\|, \mathbf{z}_{g,v}/\|f_{g,v}(\mathbf{u}_g)\|\big]$, $\mathbf{T}_g = \mathbf{y}_g^T[\mathbf{z}_{g,u}/\|f_{g,u}(\mathbf{u}_g)\|, \mathbf{z}_{g,v}/\|f_{g,v}(\mathbf{u}_g)\|]$, we derive corresponding metric and form matrices for the rolling

sphere surface as

$$\mathbf{M}_g = \begin{bmatrix} \left(R_g^2 + R_{g,u}^2\right)^{\frac{1}{2}} & 0 \\ 0 & \alpha \end{bmatrix}, \mathbf{K}_g = \begin{bmatrix} k_{nu}^g & \tau_{nu}^g \\ \tau_{nv}^g & k_{nv}^g \end{bmatrix}$$
$$\mathbf{T}_g = \begin{bmatrix} k_{gu}^g \ , \ k_{gv}^g \end{bmatrix}, \tag{8}$$

where

$$k_{nu}^g = (R_g^2 + R_{g,u}^2)^{-3/2}(R_g^2 - R_g R_{g,uu} + 2R_{g,u}^2),$$
$$\tau_{nu}^g = \tau_{nv}^g = k_{nv}^g = 0,\ k_{gu}^g = k_{gv}^g = 0.$$

Similarly, we can derive these matrices for sphere surface as

$$\mathbf{M}_s = \begin{bmatrix} R_s & 0 \\ 0 & R_s \cos u_s \end{bmatrix}, \mathbf{K}_s = \begin{bmatrix} \frac{1}{R_s} & 0 \\ 0 & \frac{1}{R_s} \end{bmatrix}$$
$$\mathbf{T}_s = [0 \ , \ -\tan u_s / R_s]. \tag{9}$$

The rolling contact kinematics equation is explained as [7]

$$\dot{\mathbf{u}}_g = \mathbf{M}_g^{-1}\left(\mathbf{K}_g + \tilde{\mathbf{K}}_s\right)^{-1}\left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} - \tilde{\mathbf{K}}_s \begin{bmatrix} v_x \\ v_y \end{bmatrix}\right)$$
$$\dot{\mathbf{u}}_s = \mathbf{M}_s^{-1}\mathbf{R}_\psi\left(\mathbf{K}_g + \tilde{\mathbf{K}}_s\right)^{-1}\left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} + \mathbf{K}_g \begin{bmatrix} v_x \\ v_y \end{bmatrix}\right)$$
$$\dot{\psi}_i = \omega_z + \mathbf{T}_g\mathbf{M}_g\dot{\mathbf{u}}_g + \mathbf{T}_s\mathbf{M}_s\dot{\mathbf{u}}_s,\ v_z = 0, \tag{10}$$

where $\boldsymbol{\omega}_{\text{rel}} = [\omega_x,\ \omega_y,\ \omega_z]^T$ and $\mathbf{V}_{\text{rel}} = [v_x,\ v_y,\ v_z]^T$ are the relative angular and linear velocities between the rolling objects with respect to the orthogonal Gauss frame at the contact point. By substituting Eqs. (8)-(9) into Eq. (10), we finalize the kinematic equations of a sphere rolling with a rigid object maintaining contact as

$$\dot{\mathbf{u}}_g = \begin{bmatrix} \frac{R_s S_1}{S_1^{3/2} + R_s S_2} & 0 \\ 0 & \frac{R_s}{\alpha} \end{bmatrix}\left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} - \begin{bmatrix} \frac{1}{R_s} & 0 \\ 0 & \frac{1}{R_s} \end{bmatrix}\begin{bmatrix} v_x \\ v_y \end{bmatrix}\right)$$
$$\dot{\mathbf{u}}_s = \begin{bmatrix} \frac{\cos\psi S_1^{3/2}}{S_1^{3/2} + R_s S_2} & -\sin\psi \\ \frac{-\sin\psi S_1^{3/2}}{\cos u_s (S_1^{3/2} + R_s S_2)} & -\frac{\cos\psi}{\cos u_s} \end{bmatrix} \cdot \left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} + \begin{bmatrix} S_1^{-3/2} S_2 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} v_x \\ v_y \end{bmatrix}\right)$$
$$\dot{\psi} = \omega_z + \begin{bmatrix} 0 & 0 \end{bmatrix}\dot{\mathbf{u}}_g + \begin{bmatrix} 0 & -\sin u_s \end{bmatrix}\dot{\mathbf{u}}_s \tag{11}$$

where $S_1 = R_g^2 + R_{g,u}^2,\ S_2 = R_g^2 - R_g R_{g,uu} + 2R_{g,u}^2$

For spin-rolling, let $\mathbf{x} = [u_g, v_g, u_s, v_s, \psi]^T$ denote the state vector, the re-ordered kinematic equation under the no-sliding constraint ($\mathbf{V}_{\text{rel}} = 0$) becomes

$$\dot{\mathbf{x}} = \mathbf{g}_1\omega_x + \mathbf{g}_2\omega_y + \mathbf{g}_3\omega_z, \tag{12}$$

where

$$g_1 = [0, R_s/\alpha, -\sin\psi, -\cos\psi/\cos u_s, \tan u_s \cos\psi]^T,$$
$$g_2 = \frac{1}{R_s S_2 + S_1^{3/2}}[-R_s S_1, 0, -S_1^{3/2}\cos\psi,$$
$$S_1^{3/2}\sin\psi/\cos u_s, S_1^{3/2}\sin\psi/\cos u_s]^T,$$
$$g_3 = [0, 0, 0, 0, 1]^T,$$

where singularities occur only if $R_s S_2 + S_1^{3/2} = 0$. We define the distribution $Q(\mathbf{x}) = \{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3, [\mathbf{g}_1, \mathbf{g}_3], [\mathbf{g}_2, \mathbf{g}_3]\}$, and the system is controllable if $\dim(Q) = 5$ [22]. By evaluating the determinant of the matrix formed by these vector fields, we obtain

$$\det(Q) = -\frac{R_s S_1^2}{\alpha \cos u_s (S_1^{3/2} + R_s S_2)^2}. \tag{13}$$

Since $\det(Q) \neq 0$, the distribution Q is full rank, implying that the system is controllable under the proposed spin-rolling kinematics.

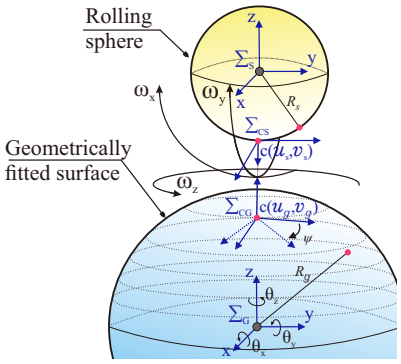


Fig. 3: The parametrization and frame definitions of rolling contact kinematics between rolling sphere and geometrically fitted surface.

IV. Results and Discussion

In this section, we present and analyze our proposed method for geometrically fitting the point cloud objects, using the example of a rolling sphere on multiple surfaces. Additionally, we discuss the performance and capabilities of the resulting geometric kinematic model.

The simulation is performed in MATLAB using the ODE45 solver on a 13th Gen Intel(R) Core(TM) i5-13500H @ 2.60 GHz platform. The absolute error tolerance for the ODE45 solver is set to 1×10^{-5}, and the total simulation time is 10 seconds. The number of harmonic terms in the Fourier fitting (Eq. (3)) is chosen from the range $n \in [3, 8]$, with a default convergence threshold of $\epsilon = 1 \times 10^{-3}$ for the fitting loss. The initial states are set as follows: $\mathbf{u}_g = [0.1, \frac{\pi}{2}]$, $\mathbf{u}_s = [0.1, 0.1]$, and $\psi = \frac{\pi}{2}$. And the rolling sphere radius is chosen as $R_s = 0.1$. In this simulation study, we consider three distinct objects as examples, for which real-world experimental point cloud data models were obtained from real-world-captured data [23].

The first object considered is a carrot, where the sphere starts at approximately $v_g = \frac{\pi}{2}$, corresponding to its center, and begins moving around it, as illustrated by Fig. 4. The simulation results show that for both pure rolling and rolling with slippage, the trajectories of the sphere are smooth and realizable, without any unexpected deviations. Furthermore, the state responses demonstrate that the performance is stable, exhibiting smooth trajectories across all states, including the sphere contact point state $\mathbf{u}_g$, the sphere contact point state $\mathbf{u}_s$, and the orientation angle ψ, as illustrated in Fig. 7(a). To examine the effects of sliding, we induce small sliding in the simulation to determine if any singularities or unexpected behavior emerge. The results clearly show that sliding motion leads to longer trajectory in the direction of sliding velocity than the one without sliding component,

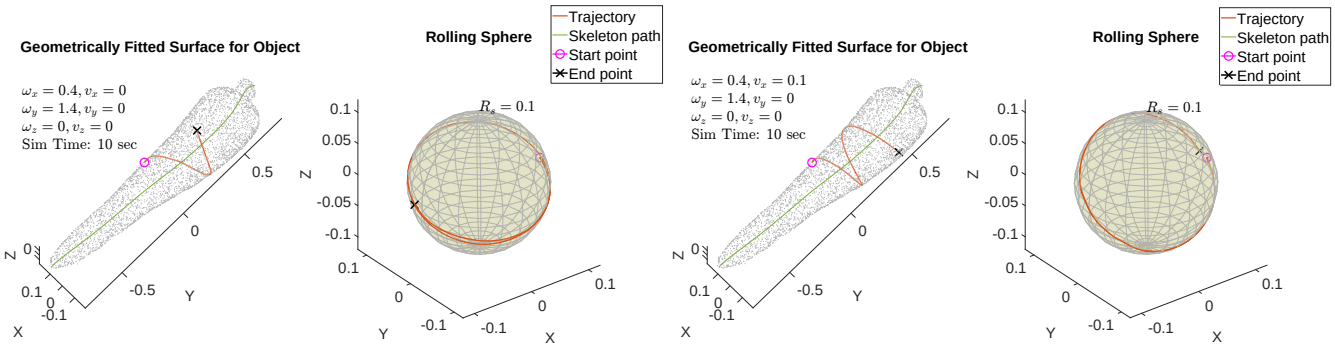


Fig. 4: Simulation results for carrot and rolling sphere with and without slippage.

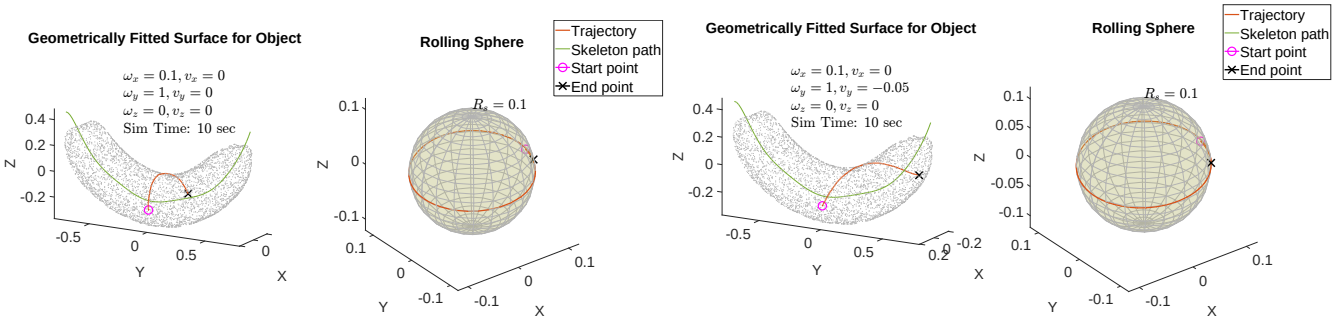


Fig. 5: Simulation results for banana and rolling sphere with and without slippage.

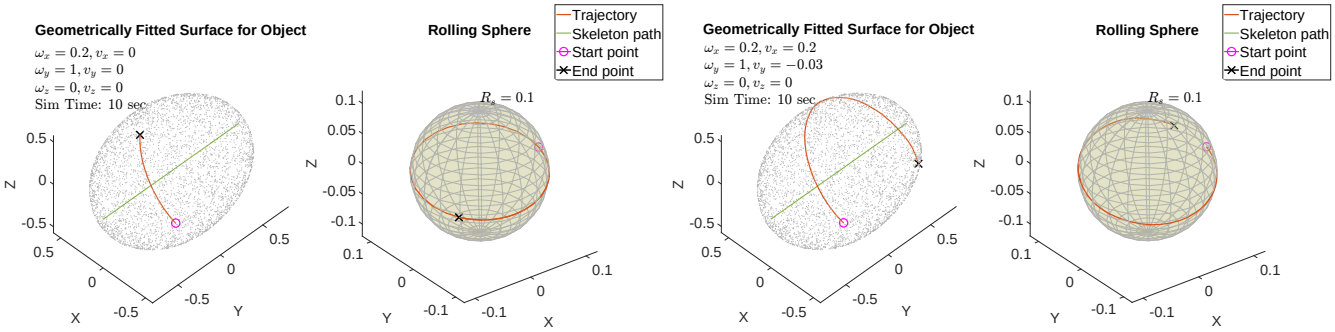


Fig. 6: Simulation results for egg and rolling sphere with and without slippage.

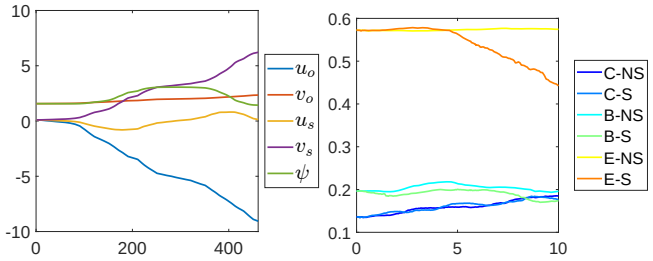


Fig. 7: (a) Time responses of states $\mathbf{u}_g$, $\mathbf{u}_s$, and ψ for the example simulation corresponding to right figure of Fig. 4. (b) R_g Responses corresponding to the example simulations in Figs. 4 - 6 respectively.

demonstrating the effectiveness of our kinematics model. Besides,we can notice that there is no significant issue in the simulation, with no singularities detected.

The next object considered is a banana, which presents a more complex geometry compared to the carrot. The skeleton curve of the banana is much more curved than that of the carrot, as observed in Fig. 5. To comprehensively test our kinematics performance, we change the slippage direction in the simulation of banana rolling contact motion. We can notice that the direction change are correctly affecting the expected rolling motion along another axis, showing longer trajectory along the skeleton path. Similar results can also be found in Fig 6, where the rolling contact motion of an egg object is also simulated. The egg is an object with distinct structure in comparison with the other two. In the simulation with slippage, we can notice that the slippage in both directions leads to longer displacement of the contact point both in the direction of skeleton path and the direction orthogonal to the skeleton path.

Finally, we demonstrated the variation in R_g, as shown in Fig. 7(b). The radius change is consistent with how the sphere rolls along the objects, demonstrating the effective-

ness of our closed curve fitting model via Fourier series model. For example, in the banana case (B-NS), the sphere travels a short distance along a path with nearly constant radius, so R_g remains nearly unchanged. In contrast, in the carrot case (C-NS), the sphere travels a longer distance toward a region with larger radius, resulting in an increase in R_g. Besides, the radius continuous decrease in the R_g response of egg rolling motion simulation is align with the behavior that the contact point on egg surface is moving from middle toward a side point. However, some fluctuations occur in the simulation as shown in the R_g response, especially in the results of egg rolling with slippage (E-S) and carrot rolling simulation (C-NS and C-S). This is because the fitted Fourier series model varies along the skeleton path due to data change and the fitted model is sensitive to the data quality, including density and the noise level. It is evident that for more consistent or denser point clouds, such as the banana, the fluctuations are significantly reduced. Thus, a more detailed study of the sensitivity to noise or sparsity in the data is essential in future.

The simulation, based on real-world object shape data, demonstrates our method's effectiveness and potential for rolling motion control and planning in scenarios such as in-hand manipulation [6], [24].

V. Conclusions

In this study, we introduced a novel method for locally fitting arbitrary point cloud data using a skeleton-guided reference frame, integrated with a semi-convex spin-rolling-sliding kinematic model grounded in differential geometry. Our framework was validated through simulations on representative objects, including a carrot, banana, and egg, using real-world point cloud data acquired from physical objects. The results indicate that both pure rolling and rolling with slippage yield smooth and physically plausible trajectories, with consistent state evolution along the skeleton-guided kinematic paths.

Furthermore, our experiments reveal a strong correlation between point cloud density and model accuracy. Higher-density point clouds enable more reliable slicing and yield more precise surface and motion reconstructions. In contrast, sparsity in the input data, such as that observed in the banana model, can lead to minor fluctuations in the resulting kinematics.

Looking ahead, we plan to extend our slicing strategy to handle more complex, multi-branch geometries by incorporating finer curvature analysis, thereby enabling applications to structures such as trees, tunnels, or branching anatomical shapes. We also aim to implement a motion planning framework and evaluate our kinematic model in real-world fingertip rolling experiments, further validating its practical applicability in dexterous manipulation scenarios.

VI. Acknowledgments

This work was partially supported by the Royal Society research grant under Grant RGS\R2\242234. Also, this work was partially supported by China Scholarship Council via a stipend (No. 202006760092).

References

[1] K. M. Lynch and T. D. Murphey, "Control of nonprehensile manipulation," in *Control problems in robotics*. Springer, 2003, pp. 39–57.

[2] S. A. Tafrishi, M. Svinin, and K. Tahara, "A survey on path planning problem of rolling contacts: Approaches, applications and future challenges," *arXiv preprint arXiv:2501.04442*, 2025.

[3] R. H. Armour and J. F. Vincent, "Rolling in nature and robotics: A review," *J. Bionic Eng.*, vol. 3, no. 4, pp. 195–208, 2006.

[4] S. A. Tafrishi, M. Svinin, E. Esmaeilzadeh, and M. Yamamoto, "Design, modeling, and motion analysis of a novel fluid actuated spherical rolling robot," *ASME J. Mech. Robot.*, vol. 11, no. 4, p. 041010, 2019.

[5] K. Tahara, K. Maruta, and M. Yamamoto, "External sensorless dynamic object manipulation by a dual soft-fingered robotic hand with torsional fingertip motion," in *Proc. IEEE In. Conf. Robot. Autom.*, 2010, pp. 4309–4314.

[6] S. Yuan, L. Shao, C. L. Yako, A. Gruebele, and J. K. Salisbury, "Design and control of roller grasper v2 for in-hand manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2020, pp. 9151–9158.

[7] D. J. Montana, "The kinematics of contact and grasp," *Int. J. Robot. Res.*, vol. 7, no. 3, pp. 17–32, 1988.

[8] M. Svinin and S. Hosoe, "Motion planning algorithms for a rolling sphere with limited contact area," *IEEE Trans. Robot.*, vol. 24, no. 3, pp. 612–625, 2008.

[9] K. Harada, T. Kawashima, and M. Kaneko, "Rolling based manipulation under neighborhood equilibrium," *Int. J. Robot. Res.*, vol. 21, no. 5-6, pp. 463–474, 2002.

[10] S. A. Tafrishi, M. Svinin, M. Yamamoto, and Y. Hirata, "A geometric motion planning for a spin-rolling sphere on a plane," *Appl. Math. Modell.*, vol. 121, pp. 542–561, 2023.

[11] S. E. Navarro, S. Mühlbacher-Karrer, H. Alagi, H. Zangl, K. Koyama, B. Hein, C. Duriez, and J. R. Smith, "Proximity perception in human-centered robotics: A survey on sensing systems and applications," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1599–1620, 2021.

[12] D. J. Montana, "Contact stability for two-fingered grasps," *IEEE Trans. Robot. Autom.*, vol. 8, no. 4, pp. 421–430, 1992.

[13] L. Cui and J. S. Dai, "A darboux-frame-based formulation of spin-rolling motion of rigid objects with point contact," *IEEE Trans. Robot.*, vol. 26, no. 2, pp. 383–388, 2010.

[14] S. Song, Z. Cui, and R. Qin, "Vis2mesh: Efficient mesh reconstruction from unstructured point clouds of large scenes with learned virtual view visibility," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6514–6524.

[15] J. Zhou, B. Ma, Y.-S. Liu, and Z. Han, "Fast learning of signed distance functions from noisy point clouds via noise to noise mapping," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 8936–8953, 2024.

[16] N. Cornea, M. Demirci, D. Silver, Shokoufandeh, S. Dickinson, and P. Kantor, "3D object retrieval using many-to-many matching of curve skeletons," in *International Conference on Shape Modeling and Applications 2005 (SMI' 05)*, 2005, pp. 366–371.

[17] Q. Wen, S. A. Tafrishi, Z. Ji, and Y.-K. Lai, "Glskeleton: A geometric laplacian-based skeletonisation framework for object point clouds," *IEEE Robot. Autom. Lett.*, pp. 1–7, 2024.

[18] A. Tagliasacchi, T. Delame, M. Spagnuolo, N. Amenta, and A. Telea, "3D skeletons: A state-of-the-art report," in *Comput. Graphics Forum*, vol. 35, no. 2. Wiley Online Library, 2016, pp. 573–597.

[19] N. Vahrenkamp, E. Koch, M. Waechter, and T. Asfour, "Planning high-quality grasps using mean curvature object skeletons," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 911–918, 2018.

[20] J. Cao, A. Tagliasacchi, M. Olson, H. Zhang, and Z. Su, "Point Cloud Skeletons via Laplacian Based Contraction," in *2010 Shape Modeling International Conference*, June 2010, pp. 187–197.

[21] E. Persoon and K.-S. Fu, "Shape discrimination using fourier descriptors," *IEEE Trans. Syst. Man Cybern.*, vol. 7, no. 3, pp. 170–179, 1977.

[22] Z. Li and J. Canny, "Motion of two rigid bodies with rolling constraint," *IEEE Trans. Robot. Autom.*, vol. 6, no. 1, pp. 62–72, 1990.

[23] T. Wu, J. Zhang, X. Fu, Y. Wang, J. Ren, L. Pan, W. Wu, L. Yang, J. Wang, C. Qian, D. Lin, and Z. Liu, "OmniObject3D: Large-vocabulary 3D object dataset for realistic perception, reconstruction and generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognit. (CVPR)*, June 2023, pp. 803–814.

[24] J. Chapman, G. Gorjup, A. Dwivedi, S. Matsunaga, T. Mariyama, B. MacDonald, and M. Liarokapis, "A locally-adaptive, parallel-jaw gripper with clamping and rolling capable, soft fingertips for fine manipulation of flexible flat cables," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6941–6947.