Curve Fitting by Interpolation, Approximation, and Fourier Reconstruction

Jinghao Chen[a]

[a]University of Science and Technology of China, No. 96 Jinzhai Road, Baohe District, Hefei, 230026, Anhui, China

ARTICLE INFO

Keywords:
Curve Fitting
Interpolation
Approximation
B-spline
Cubic Spline
Fourier Series
Epicycle Reconstruction

ABSTRACT

This report studies planar curve fitting from sampled points under two related viewpoints. In the first part, open and closed curves are represented by parametric functions and approximated with cubic spline interpolation, B-spline interpolation, polynomial least-squares fitting, and B-spline least-squares fitting. The comparison is carried out under different parameterization rules, different numbers of sample nodes, and several noise levels. In the second part, closed curves are modeled as periodic signals in the complex plane and reconstructed by truncated Fourier series. This formulation explains why a closed contour can be viewed as a superposition of circular motions and also provides a natural way to analyze the effect of the harmonic order. Quantitative evaluation is based mainly on Chamfer distance, together with visual inspection of local details and reconstruction stages. The experiments show that spline-based methods are more reliable on curves with local bends, that parameterization strongly affects interpolation quality on non-uniformly curved shapes, and that Fourier reconstruction achieves a clear accuracy gain as more harmonics are introduced, especially for contours with repeated oscillatory structure.

1. Introduction

The goal of this assignment is to recover smooth planar curves from discrete sample points and to understand how different modeling choices affect the final fit. The implementation is organized around two complementary questions. The first is how to build interpolation and approximation functions that pass through or stay close to the sampled data. The second is how to exploit periodic structure when the target is a closed contour.

For the first question, the project treats a planar curve as a parametric mapping

$$\boldsymbol{r}(t) = \big(x(t), y(t)\big), \qquad t \in [0, 1]. \tag{1}$$

Once parameter values are assigned to the sampled points, the fitting problem becomes the recovery of two one-dimensional functions, one for each coordinate. This makes it possible to compare global polynomial approximation, piecewise cubic interpolation, and B-spline models within a unified framework. Because the same geometric curve can behave very differently under different parameterizations, the study also compares uniform, chord-length, centripetal, and Foley–Nielsen rules.

Here we should emphasize that the implementation treats the sampled points as an ordered point sequence. The order does not need to be uniformly spaced in arclength, because the parameterization step can redistribute the parameter values according to chord length or local geometry. If only an unordered point set were given, an additional ordering stage, such as a traveling-salesman-type path construction, would be needed before interpolation or approximation. This ordering problem is not implemented in the project; instead, the experiments assume that the input points are already ordered in a way that is consistent with the underlying curve.

For the second question, a closed curve is rewritten as a periodic complex-valued signal,

$$z(t) = x(t) + i\,y(t), \qquad z(t+1) = z(t). \tag{2}$$

This periodic viewpoint leads directly to truncated Fourier reconstruction, where each harmonic contributes a rotating vector. The resulting representation is both computationally convenient and geometrically interpretable.

The experiments cover representative open and closed curves, including an S-curve, a sinusoidally modulated open curve, a circle, an ellipse, a cardioid, and a wavy circle. The code also includes an interactive GUI so that the influence of method choice and Fourier order can be inspected directly.

To make the organization of the project clearer, Figure 1 summarizes the full workflow. Starting from ordered planar sample points, the pipeline separates into two branches. For general curve fitting, the samples are first parameterized and

then processed by interpolation or least-squares approximation models, including cubic spline, B-spline interpolation, polynomial least squares, and B-spline least squares. For closed contours, the data can also be closed, periodically resampled over an ordered spline parameter, and rewritten as a periodic complex signal, which leads naturally to truncated Fourier reconstruction and epicycle-style interpretation. The outputs are then compared through geometric distances, robustness experiments, and visual inspection.

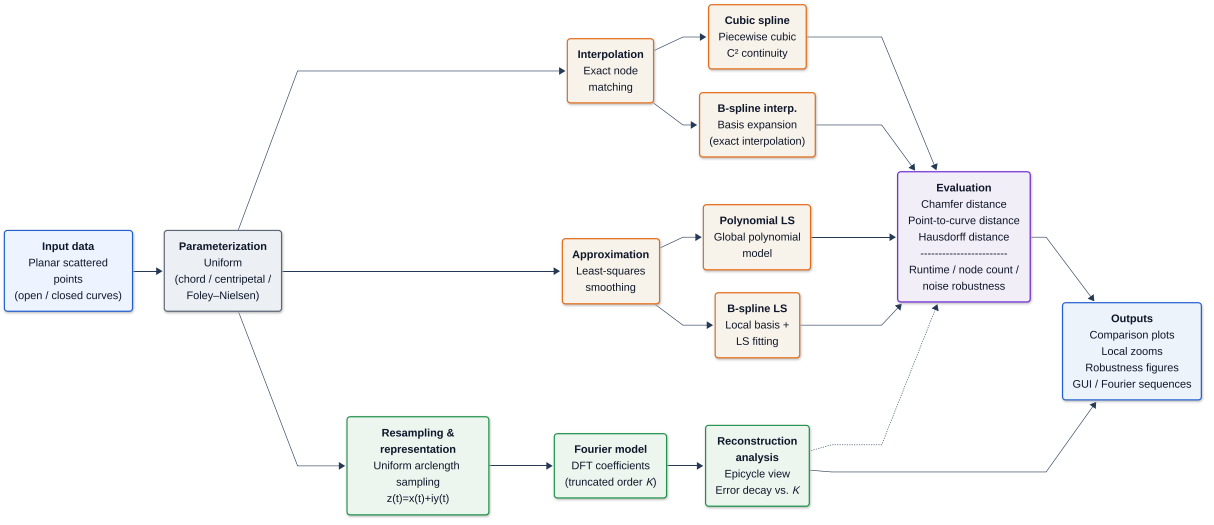

Figure 1: Overall workflow of the curve-fitting and periodic reconstruction pipeline. The upper branch handles parameterized interpolation and approximation for planar sample points, while the lower branch reformulates closed contours as periodic complex signals for Fourier reconstruction and staged epicycle analysis.

2. Mathematical formulation and fitting models

2.1. Parameterization of sample points

Suppose the sampled points are

$$\boldsymbol{p}_i = (x_i, y_i), \qquad i = 0, 1, \dots, n-1. \tag{3}$$

Before fitting, one must assign a parameter value t_i to each point. The implementation contains four rules.

Uniform parameterization uses equally spaced nodes,

$$t_{i+1} - t_i = \text{const} \tag{4}$$

Chord-length parameterization uses cumulative Euclidean distance,

$$t_{i+1} - t_i \propto \left\| \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} - \begin{pmatrix} x_i \\ y_i \end{pmatrix} \right\| \tag{5}$$

Centripetal parameterization weakens the influence of long segments by replacing the increment with its square root,

$$t_{i+1} - t_i \propto \sqrt{\left\| \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} - \begin{pmatrix} x_i \\ y_i \end{pmatrix} \right\|} \tag{6}$$

The Foley–Nielsen option in the code is implemented as a practical angle-weighted variant. Let

$$d_i = \|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i\|_2,$$

and let α_i denote the local turning angle at $\boldsymbol{p}_i$, computed from the two adjacent segments. The implemented increment is

$$\Delta t_i \propto d_i \left(1 + \lambda \frac{\alpha_i + \alpha_{i+1}}{\pi}\right), \qquad \lambda = 0.5. \qquad \alpha_i = \arccos\left(\frac{(\boldsymbol{p}_i - \boldsymbol{p}_{i-1}) \cdot (\boldsymbol{p}_{i+1} - \boldsymbol{p}_i)}{\|\boldsymbol{p}_i - \boldsymbol{p}_{i-1}\|_2 \|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i\|_2}\right). \tag{7}$$

For open curves, endpoint angles are set to zero; for closed curves, adjacent indices are taken cyclically. Thus, compared with pure chord-length parameterization, the parameter spacing is enlarged near locally turning regions. This is useful when the same Euclidean distance corresponds to very different local geometry. [1]

2.2. Interpolation models

The cubic spline interpolant fits $x(t)$ and $y(t)$ separately with piecewise cubic polynomials:

$$S_x(t_i) = x_i, \qquad S_y(t_i) = y_i, \qquad i = 0, \dots, n-1, \tag{8}$$

while preserving C^2 continuity across knots. The final curve is

$$\boldsymbol{S}(t) = \big(S_x(t), S_y(t)\big). \tag{9}$$

Because the interpolation conditions are enforced pointwise, the fitted curve passes through every sample.

The B-spline interpolant uses the basis expansion

$$\boldsymbol{C}(t) = \sum_{j=0}^{m-1} \boldsymbol{c}_j N_{j,p}(t), \tag{10}$$

where $N_{j,p}$ is the B-spline basis function of degree $p = 3$. The control points $\boldsymbol{c}_j$ are obtained by solving

$$\boldsymbol{C}(t_i) = \boldsymbol{p}_i, \qquad i = 0, \dots, n-1. \tag{11}$$

Hence the interpolation condition is again satisfied exactly, although the curve is described through basis functions and control points instead of piecewise Hermite data.

2.3. Approximation models

The polynomial least-squares model uses a degree-d basis:

$$x(t) \approx \sum_{k=0}^{d} a_k t^k, \qquad y(t) \approx \sum_{k=0}^{d} b_k t^k. \tag{12}$$

The coefficients are determined by minimizing

$$\min_{\boldsymbol{a},\boldsymbol{b}} \sum_{i=0}^{n-1} \left(x_i - \sum_{k=0}^{d} a_k t_i^k\right)^2 + \sum_{i=0}^{n-1} \left(y_i - \sum_{k=0}^{d} b_k t_i^k\right)^2. \tag{13}$$

This model does not enforce $\boldsymbol{r}(t_i) = \boldsymbol{p}_i$; instead, it trades interpolation accuracy for global smoothing.

The B-spline least-squares model keeps the B-spline basis but replaces exact interpolation by

$$\min_{\{\boldsymbol{c}_j\}} \sum_{i=0}^{n-1} \left\| \boldsymbol{p}_i - \sum_{j=0}^{m-1} \boldsymbol{c}_j N_{j,p}(t_i) \right\|_2^2. \tag{14}$$

Compared with global polynomial fitting, this retains locality; compared with interpolation, it is less sensitive to perturbations in the data.

[1] https://zhuanlan.zhihu.com/p/416902631

2.4. Evaluation metric

The main quantitative metric is the symmetric Chamfer distance between a dense set of target points A and sampled points on the fitted curve B:

$$d_{\mathrm{ch}}(A,B)=\frac{1}{|A|}\sum_{a\in A}\min_{b\in B}\|a-b\|_2+\frac{1}{|B|}\sum_{b\in B}\min_{a\in A}\|b-a\|_2. \tag{15}$$

For selected experiments, we also report the average point-to-curve distance from the input samples P to the fitted curve Γ,

$$d_{\mathrm{pc}}(P,\Gamma)=\frac{1}{|P|}\sum_{p\in P}\min_{x\in\Gamma}\|p-x\|_2, \tag{16}$$

and the Hausdorff distance

$$d_{\mathrm{H}}(A,B)=\max\left\{\sup_{a\in A}\inf_{b\in B}\|a-b\|_2,\ \sup_{b\in B}\inf_{a\in A}\|b-a\|_2\right\}. \tag{17}$$

Runtime is recorded as a measure of computational cost. Smaller distance values indicate better geometric agreement, while lower runtime indicates higher efficiency. [2]

3. Experimental setting

The synthetic curves are generated analytically so that the fitting error can be judged against a known target. For closed curves, the code uses $\theta=2\pi t$ and includes

$$\text{circle:}\quad (x,y)=(\cos\theta,\sin\theta), \tag{18}$$

$$\text{ellipse:}\quad (x,y)=(1.4\cos\theta,0.8\sin\theta), \tag{19}$$

$$\text{cardioid:}\quad (x,y)=\big((1-\cos\theta)\cos\theta,\ (1-\cos\theta)\sin\theta\big), \tag{20}$$

$$\text{wavy circle:}\quad (x,y)=\big(r(\theta)\cos\theta,\ r(\theta)\sin\theta\big),\quad r(\theta)=1+0.2\cos(6\theta)+0.1\sin(3\theta). \tag{21}$$

For open curves, the code first maps $t\in[0,1]$ to $x=2t-1$ and uses examples such as

$$\text{S-curve:}\quad (x,y)=\big(x,\sin(\pi x)\big), \tag{22}$$

$$\text{cubic polynomial curve:}\quad (x,y)=\big(x,0.8x^3-0.4x\big), \tag{23}$$

$$\text{sinusoidally modulated curve:}\quad (x,y)=\big(x,0.5\sin(4\pi t)+0.25\sin(9\pi t)\big). \tag{24}$$

Unless stated otherwise, the main fitting experiments sample each fitted curve at 500 points and evaluate it against a 2000-point reference curve. The basic reconstruction and noise experiments use 40 sample nodes, the interpolation-versus-approximation comparison uses 48 nonuniform nodes, and the parameterization comparison uses 42 nonuniform nodes. The polynomial least-squares model uses degree $\min(7,\max(3,\lfloor n/4\rfloor))$, and the B-spline models use cubic basis functions, with smoothing $s=0.001n$ for B-spline least squares. The node-count study tests

$$n\in\{12,20,32,48,72,100\},$$

and the noise study adds Gaussian perturbations with

$$\sigma\in\{0,0.01,0.02,0.04,0.06\}.$$

Each noise level is repeated over five random trials. The parameterization comparison is carried out with cubic spline interpolation so that the effect of the parameter rule is isolated from the effect of the fitting family. The Fourier experiments use 80 nonuniform input points, resample the closed contour to 512 points before the Fourier transform, and reconstruct 600 curve samples for evaluation.

[2] https://zhuanlan.zhihu.com/p/651463316

4. Interpolation and approximation results

4.1. Method comparison on clean and mildly noisy data

Table 1 summarizes the average Chamfer distance of the four fitting methods in the interpolation-versus-approximation benchmark, averaged over the clean case and a mildly noisy case with $\sigma = 0.02$. Two patterns are immediately clear. First, curves with gentle global structure, such as the circle and the cubic polynomial curve, can be represented very well by the polynomial least-squares model. Second, once the target contains stronger local bending or repeated oscillation, spline-based models become more reliable. On the S-curve, cubic spline interpolation gives the smallest error, while on the wavy circle the global polynomial fit deteriorates sharply because the oscillatory boundary is difficult to capture with a low-degree polynomial.

Table 1
Average Chamfer distance on the interpolation-versus-approximation benchmark, averaged over $\sigma = 0$ and $\sigma = 0.02$.

Shape	B-spline interp.	B-spline LS	Cubic spline	Polynomial LS
circle	0.0200	0.0378	0.0208	0.0094
cubic_poly	0.0181	0.0249	0.0174	0.0116
s_curve	0.0309	0.0357	0.0236	0.0507
wavy_circle	0.0272	0.0594	0.0292	0.2194

These numerical differences are consistent with the plotted fits. The S-curve already reveals that a local representation is preferable near its turning region, where the cubic spline tracks the bend cleanly and the global polynomial begins to flatten the shape (Figure 2). The sinusoidally modulated curve tells the same story more strongly: repeated local oscillations are preserved by the spline families, whereas the polynomial approximation remains smoother and cannot recover every wave crest with the same fidelity.

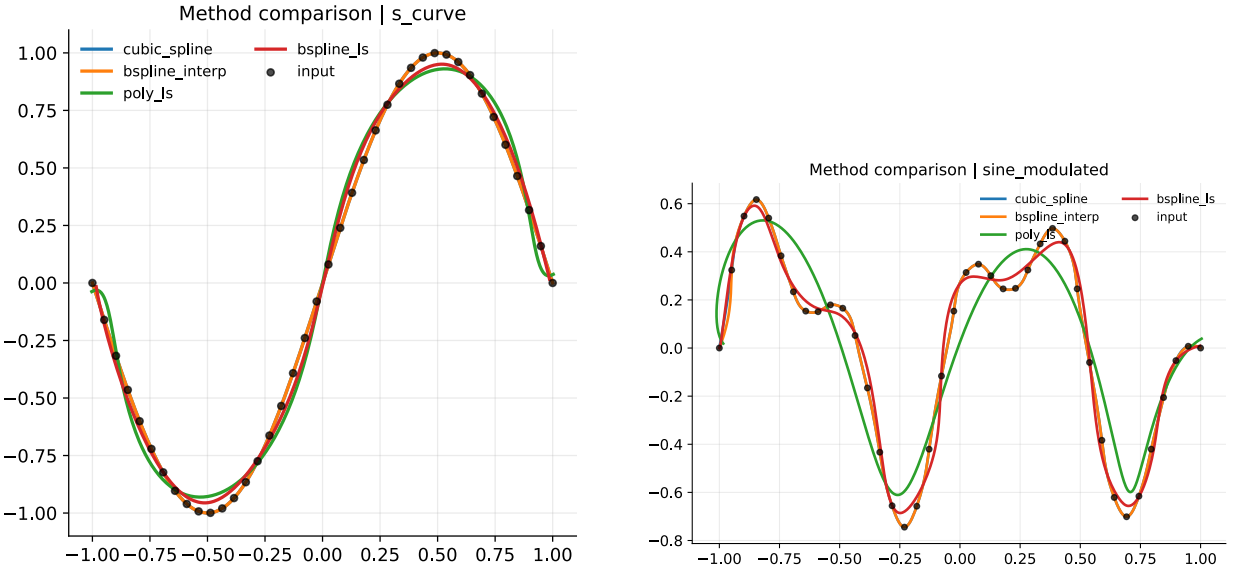

Figure 2: Representative comparisons between interpolation and approximation methods on noiseless data.

The local magnification is particularly useful because the global plots can hide where the geometric error actually comes from. Near the highlighted segment of the S-curve, the fitted curves remain visually close overall, yet the polynomial and least-squares versions already show small but systematic displacement relative to the target. On the modulated open curve, the local zoom makes the phase mismatch much easier to see, especially around a short high-curvature interval where underfitting accumulates rapidly (Figure 3).

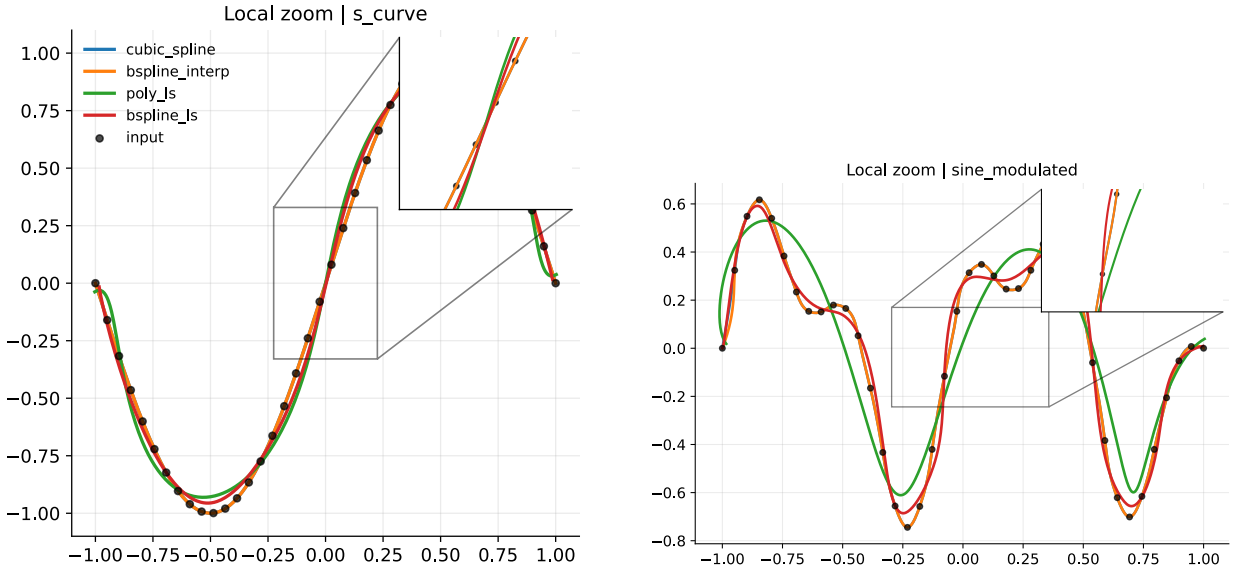

Figure 3: Local zooms used to inspect where geometric discrepancies concentrate.

4.2. Why parameterization matters

When the fit is built from parametric basis functions, the parameter values are not a minor implementation detail; they determine where the model spends its degrees of freedom. If the nodes are distributed uniformly in parameter space while the geometry changes non-uniformly in physical space, regions with sharp turning are effectively undersampled.

Table 2 reports the Chamfer distance of cubic spline interpolation under different parameterization rules. Uniform spacing is consistently the weakest choice. Chord-length and Foley–Nielsen parameterization are clearly better on the cardioid and ellipse, while centripetal parameterization is best on the sinusoidally modulated open curve. This makes sense: the centripetal rule suppresses the dominance of long segments but still allocates more resolution to locally varying regions.

Table 2
Chamfer distance under different parameterization rules for cubic spline interpolation.

Shape	Centripetal	Chord	Foley–Nielsen	Uniform
cardioid	0.0130	0.0055	0.0055	0.0198
ellipse	0.0073	0.0050	0.0051	0.0117
sine_modulated	0.0104	0.0166	0.0152	0.0152

The geometry behind these numbers is visible in the parameterization comparison plots. On the sinusoidally modulated curve, centripetal spacing better balances the dense oscillatory intervals against the smoother stretches, so the reconstructed curve remains more uniform along the full domain. The cardioid provides an even clearer contrast because the cusp-like region is very sensitive to how the nodes are distributed; once the parameter spacing follows the geometry rather than the sample index, the sharp feature is recovered far more cleanly (Figure 4).

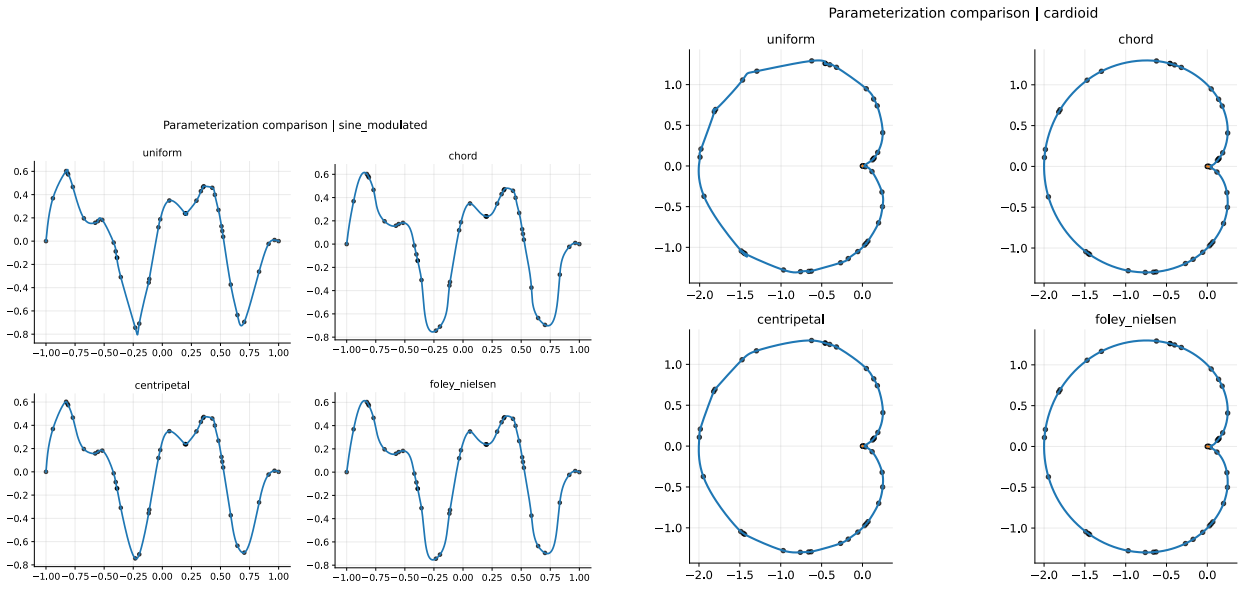

Figure 4: Influence of parameterization on cubic spline interpolation for an open curve and a closed curve.

4.3. Effect of the number of sample nodes

Increasing the number of nodes reduces the geometric mismatch for every method, but the rate of improvement depends strongly on the complexity of the target. For a relatively smooth open curve such as the S-curve, all methods improve steadily, and the gap between spline models and polynomial least squares becomes smaller once n is large enough. The wavy circle is more demanding: at low node counts, the oscillatory boundary is only coarsely represented, so the error remains high; once the sampling density reaches about 48 or 72 points, the reduction becomes much more pronounced.

This trend is apparent in the error curves in Figure 5. The main message is not simply that "more points are better"; rather, the number of nodes must match the geometric frequency content of the target. A globally smooth model may already be sufficient for a mild shape, but a contour with repeated fine structure needs both enough data points and a fitting family with enough local flexibility.

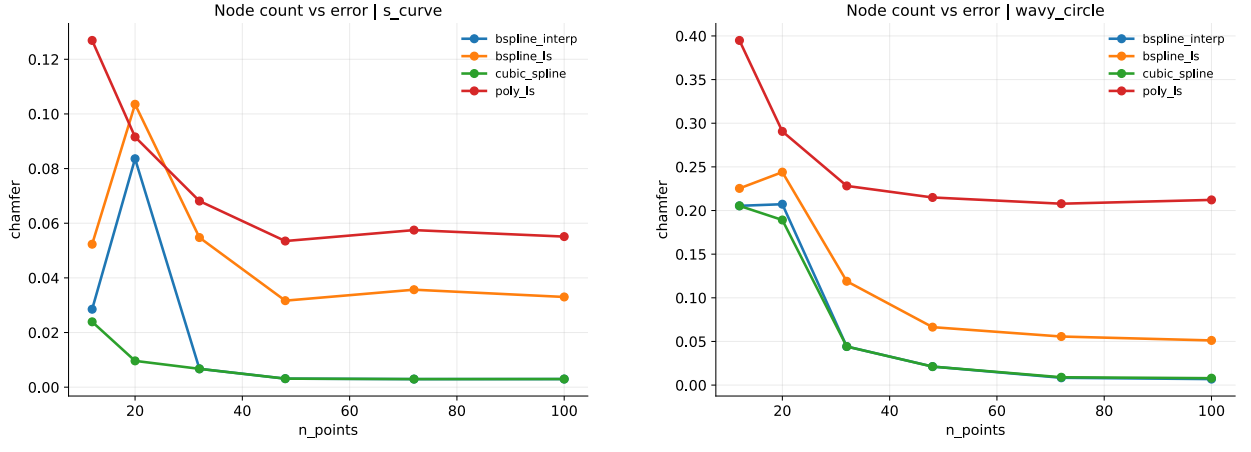

Figure 5: Error decay as the number of sampled nodes increases.

4.4. Noise robustness

Interpolation has a clear advantage when the samples are clean, but that advantage weakens once the data are perturbed. On the S-curve, cubic spline interpolation is very accurate at zero noise, yet its error grows rapidly as the perturbation level increases. Polynomial least squares starts with a larger bias, but the growth is much slower, which is exactly what one would expect from a smoothing model. B-spline least squares lies between these two extremes and acts as a compromise between local flexibility and robustness (Figure 6).

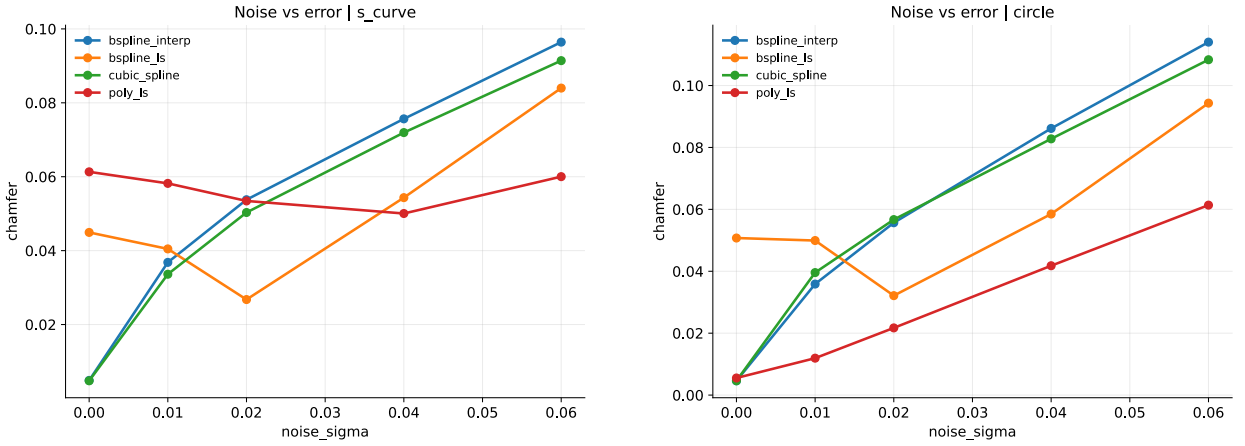

Figure 6: Average Chamfer distance under additive Gaussian noise. Each point averages five random trials.

The distribution plots make the robustness pattern much easier to read because they separate the results by method and by nonzero noise level instead of collapsing everything into a single pooled summary. On the S-curve, the boxplots show a clear upward drift for the interpolation-based reconstructions as the perturbation increases: both the cubic spline and the interpolating B-spline move from relatively small errors at low noise to substantially higher medians and wider interquartile ranges at larger noise levels. Their upper whiskers also become longer, which means that once the samples are contaminated, the fitted curve is not only less accurate on average but also more likely to produce visibly worse runs. This is consistent with the mechanism of interpolation itself. Since every sampled point must be matched exactly, local perturbations are transferred directly into local bending of the reconstructed curve, and the effect becomes more pronounced near the turning regions of the S-shape. The least-squares models behave differently. The B-spline approximation remains noticeably more compact across noise levels, indicating that the smoothing term filters out part of the high-frequency disturbance instead of reproducing it geometrically. The polynomial least-squares fit shows a more mixed pattern: its center does not rise as sharply as that of the interpolation methods, but its spread still enlarges at higher noise. In other words, the polynomial model is less sensitive to pointwise perturbation than strict interpolation, yet it can still produce occasional poorer fits because its global basis must absorb the distortion through a limited number of coefficients.

The violin plot for the circle reveals the same contrast from a more explicitly distributional viewpoint. The polynomial least-squares fit stays concentrated in the lower-error region even when noise is present, while the interpolation-based methods develop broader violins whose mass shifts upward as the noise level increases. This means that the degradation is not merely a matter of mean error; the entire error distribution moves to a less favorable range. The reason is geometric. A circle is globally smooth, closed, and strongly low-frequency in structure, so a global approximation can absorb random radial perturbations without having to preserve every local fluctuation. Interpolation, however, enforces those fluctuations point by point, effectively turning noisy samples on an ideal boundary into small oscillations along the recovered contour. The least-squares B-spline again sits between these two extremes: it is more stable than interpolation because of smoothing, but it still does not compress the noisy closed shape as effectively as the polynomial fit in this particular experiment. Taken together, these distribution plots show that the choice between interpolation and approximation under noise is not only a question of which method gives the smallest average error. It is also a question of dispersion, tail growth, and failure mode: whether the method degrades gradually, or whether a few noisy samples can trigger a visibly unstable reconstruction (Figure 7).

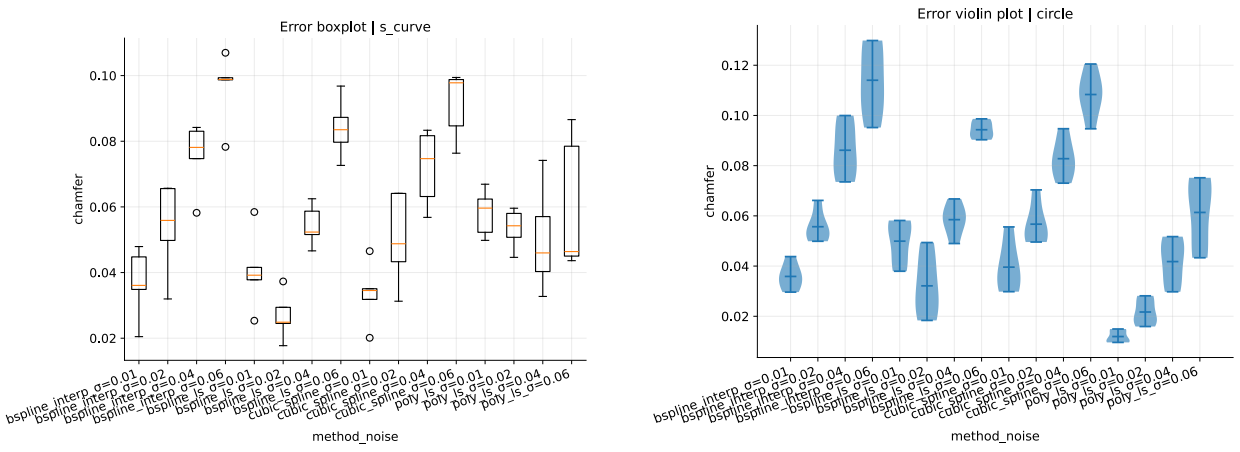

Figure 7: Distributional summaries of robustness across nonzero noise levels and random trials.

5. Periodic Fourier reconstruction of closed curves

For a closed contour, the most natural model is periodic rather than polynomial. After closing the ordered contour and resampling it uniformly over a periodic spline parameter, the code forms the complex signal

$$z_m = x_m + iy_m, \qquad m = 0, 1, \ldots, M-1. \tag{25}$$

The discrete Fourier coefficients are then

$$c_k = \frac{1}{M} \sum_{m=0}^{M-1} z_m e^{-i2\pi km/M}, \qquad k \in \mathbb{Z}, \tag{26}$$

and the truncated reconstruction is

$$z_K(t) = \sum_{k=-K}^{K} c_k e^{i2\pi kt}. \tag{27}$$

This formulation explains why the problem should be treated as periodic. A closed contour has no privileged start or end in geometry; once parameterized around the loop, both coordinates repeat after one cycle. The Fourier basis therefore matches the topology of the object directly. Equation (27) also explains the well-known epicycle interpretation. Each term

$$c_k e^{i2\pi kt}$$

is a complex vector rotating with angular speed $2\pi k$, so the full contour is generated by summing circular motions with different radii, phases, and frequencies. Low-order terms recover the global body of the curve, while high-order terms restore sharp detail and fine oscillation.

The error curves confirm this mechanism. On the cardioid, the Chamfer distance drops from 0.6588 at $K = 1$ to 0.0322 at $K = 5$, and then decreases more gradually to about 0.0207 at $K = 30$. The wavy circle improves more slowly because its repeated ripples require higher frequencies, but the same pattern is visible there as well, with the Chamfer distance decreasing from 0.2406 at $K = 1$ to 0.0267 at $K = 30$ (Figure 8). The small non-monotone fluctuations at intermediate K come from the interaction between truncation, discrete sampling, and the evaluation metric, but the overall decay remains clear.

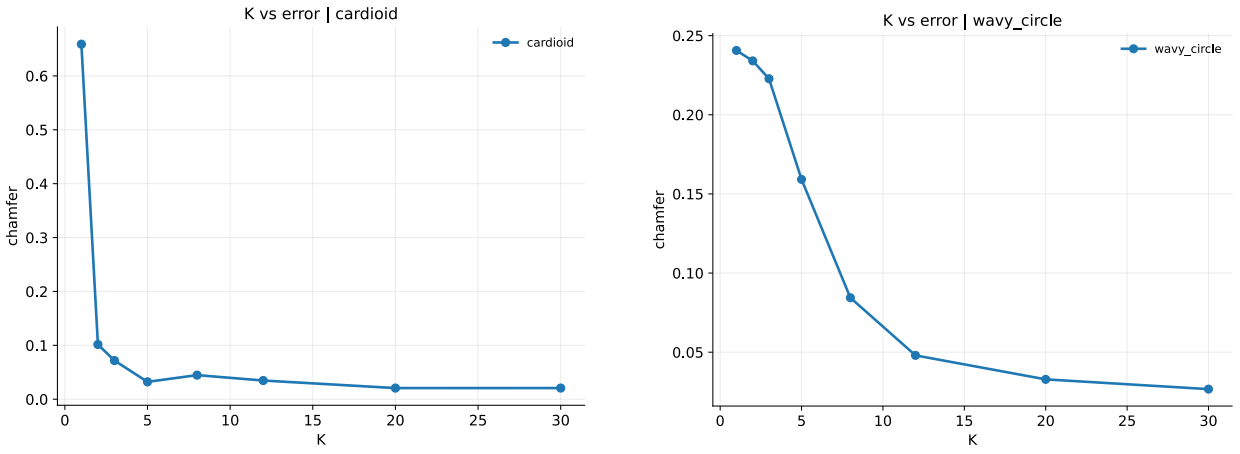

Figure 8: Reconstruction error as a function of the Fourier truncation order K.

The staged reconstructions make the frequency effect more concrete. For the cardioid, the first harmonic already captures the global loop, but the cusp is still badly rounded. By $K = 5$, the main shape is present; by $K = 20$ and $K = 30$, the cusp and boundary alignment are largely restored. The wavy circle is different: the coarse circular outline appears very early, yet the repeated ripple structure only becomes accurate once enough harmonics are admitted. This is exactly what Fourier theory predicts, since repeated boundary undulations are encoded in higher-frequency coefficients (Figures 9 and 10).

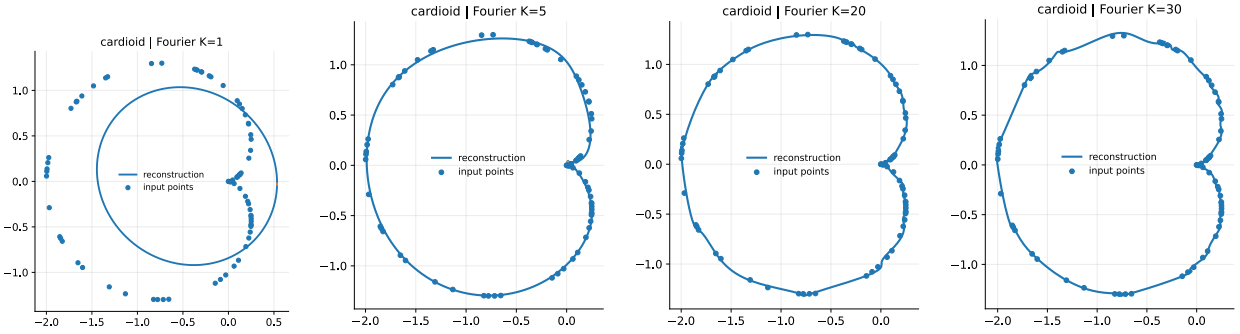

Figure 9: Cardioid reconstruction at increasing Fourier orders.

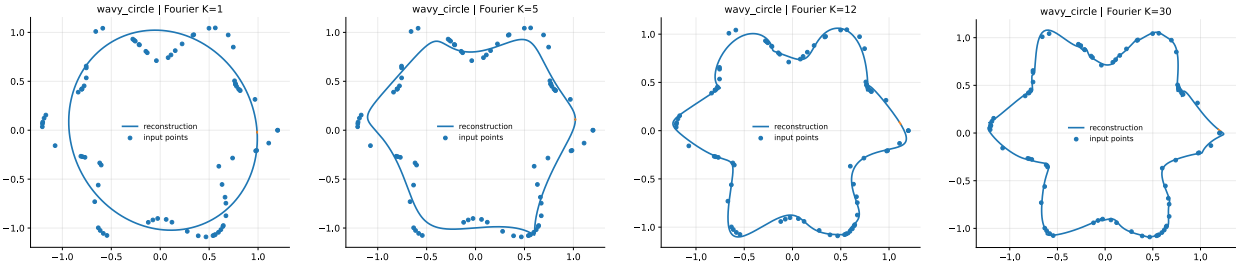

Figure 10: Wavy-circle reconstruction at increasing Fourier orders.

6. Interactive interface

Beyond the batch experiments, the project also provides a Streamlit interface for interactive inspection. The left control panel allows the user to choose the curve family, the fitting method, the number of nodes, the parameterization rule, and the Fourier order for periodic reconstruction. The main panel then renders the fitted curve together with the sampled points and, when appropriate, the corresponding quantitative summary. This interface is especially useful when comparing small parameter changes that are difficult to appreciate from CSV tables alone, because the geometric effect can be checked immediately in the plot (Figure 11).

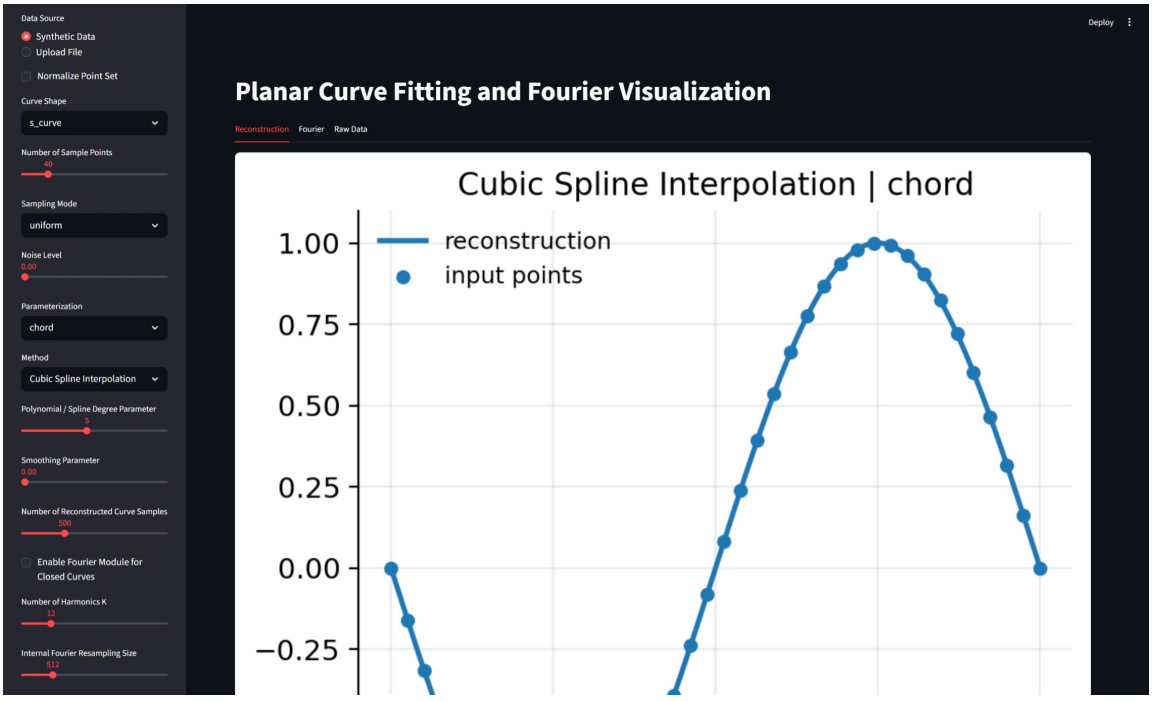

Figure 11: Interactive GUI for switching among fitting methods, parameterization rules, and Fourier settings.

7. Conclusion

This project implemented both interpolation-based and approximation-based curve fitting and then extended the study to periodic Fourier reconstruction for closed contours. The experiments support four main conclusions.

First, the fitting family must match the geometry. Global polynomial least squares is efficient and accurate on smooth low-complexity shapes, but it is less reliable on curves with repeated oscillation or strong local turning. Spline-based methods are more robust in those cases because they allocate flexibility locally.

Second, interpolation conditions matter. Cubic spline interpolation and B-spline interpolation satisfy the node constraints exactly, which is beneficial on clean data, whereas least-squares models intentionally relax those constraints and become more stable under noise.

Third, parameterization is a genuine modeling component rather than a preprocessing detail. Chord-length, centripetal, and Foley–Nielsen rules consistently outperform uniform spacing when the target geometry is non-uniform.

Finally, closed curves are naturally periodic objects. Once rewritten in the Fourier basis, the reconstruction process becomes both analyzable and visually interpretable: low harmonics recover the global outline, while higher harmonics restore local detail. The staged cardioid and wavy-circle experiments make this progression especially clear.