Lagrange Piecewise-Quadratic Interpolation Based on Planar Unordered Reduced Data

Ryszard Kozera and Piotr Szmielew(✉)

Faculty of Applied Informatics and Mathematics, Warsaw University of Life Sciences - SGGW, Nowoursynowska Str. 159, 02-776 Warsaw, Poland
{ryszard_kozera,piotr_szmielew}@sggw.pl

Abstract. This paper discusses the problem of fitting non-parametric unordered reduced data (i.e. a collection of interpolation points) with piecewise-quadratic interpolation to estimate an unknown curve γ in Euclidean space E^2. The term reduced data stands for the situation in which the corresponding interpolation knots are unavailable. The construction of ordering algorithm based on *e-graph of points* (i.e. a complete weighted graph using euclidean distances between points as respective weights) is introduced and tested here. The unordered set of input points is transformed into an ordered one upon using a minimal spanning tree (applicable for open curves). Once the order on points is imposed a piecewise-quadratic interpolation $\hat{\gamma}_2$ combined with the so-called *cumulative chords* is used to fit unordered reduced data. The entire scheme is tested initially on sparse data. The experiments carried out for dense set of interpolation points and designed to test the asymptotics in γ approximation by $\hat{\gamma}_2$ result in numerically computed cubic convergence order. The latter coincides with already established asymptotics derived for γ estimation via piecewise-quadratic interpolation based on ordered reduced data and cumulative chords.

1 Introduction

In a classical interpolation setting, a sampled ordered data points $Q_m = \{q_i\}_{i=0}^m$ with $\gamma(t_i) = q_i \in E^n$ define the so-called parametric data $(\{t_i\}_{i=0}^m, Q_m)$. Here we also assume to deal with a parametric curve $\gamma : [0,T] \to E^n$ with $t_0 = 0$ and $t_m = T < \infty$. Once the corresponding interpolation knots $\{t_i\}_{i=0}^m$ are missing the set Q_m represents the so-called non-parametric data (or reduced data). Under such circumstances the unknown knots $\{t_i\}_{i=0}^m$ must be first somehow estimated by properly guessed $\{\hat{t}_i\}_{i=0}^m \approx \{t_i\}_{i=0}^m$. The latter combined with Q_m permits to apply a given interpolation scheme $\hat{\gamma} : [0,\hat{T}] \to E^n$, with $\hat{t}_0 = 0$ and $\hat{t}_m = \hat{T}$. It is also required here that $t_i < t_{i+1}$ and $q_i \neq q_{i+1}$. In addition, the curve γ is assumed to be regular (i.e. $\gamma' \neq \mathbf{0}$) and of class C^3. From now on we consider and discuss the situation when $n = 2$, i.e. the case of interpolating planar curves.

We introduce now an interpolation scheme based on non-parametric unordered data $\bar{Q}_m = \{\bar{q}_i\}_{i=0}^m$ (with permuted points from Q_m) for which natural order of points in Q_m (dictated by the condition $q_i = \gamma(t_i)$) is somehow lost.

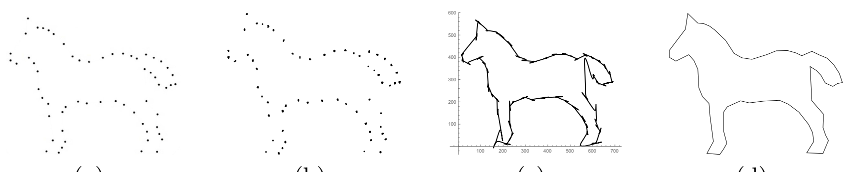
K. Saeed and W. Homenda (Eds.): CISIM 2015, LNCS 9339, pp. 423–434, 2015.
DOI: 10.1007/978-3-319-24369-6_35

Clearly, in order to estimate the unknown curve γ with an arbitrary interpolant $\hat{\gamma} : [0, T] \to E^2$ based on $\bar{Q}_m$ it is necessary to propose a proper ordering of points in $\bar{Q}_m$ (accounting for the geometry of $\bar{Q}_m$) converting $\bar{Q}_m$ into an ordered set $\hat{Q}_m = \{\hat{q}_i\}_{i=0}^m$. This paper discusses a method designed to find such ordering for open curves serving subsequently as an input to piecewise-quadratic interpolation $\hat{\gamma} = \hat{\gamma}_2$ used with the guessed knots $\{\hat{t}_i\}_{i=0}^m$ according to the so-called cumulative chords [1]. Initially, we present the examples for different planar curves illustrating the proposed data fitting scheme implemented on sparse data $\bar{Q}_m$. Finally, we test numerically the convergence order α in approximating γ by $\hat{\gamma}_2$ based on dense $\bar{Q}_m$ converted into $\hat{Q}_m$ as prescribed in this paper. The conducted tests suggest cubic order of convergence.

Specific examples for interpolating real life ordered (or unordered) reduced data Q_m (or $\bar{Q}_m$) in *computer graphics* (light-source motion estimation or image rendering), *computer vision* (image segmentation or video compression), *geometry* (trajectory, curvature of area estimation) or in *engineering and physics* (fast particles' motion estimation) can be found among all in [1], [2] and [3].

2 Problem Formulation and Motivation

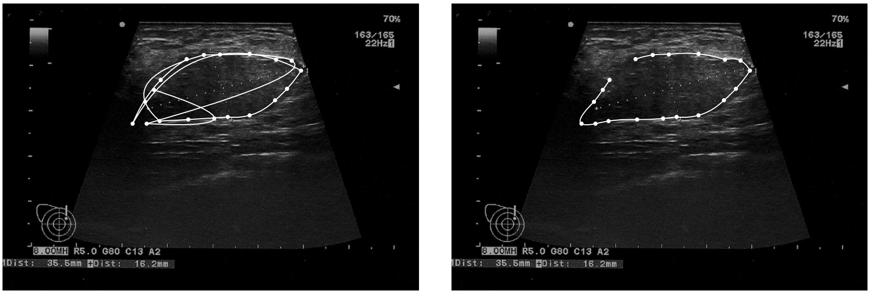
Frequently, once dealing with real life reduced data (i.e. the collection of multi-dimensional points) the exact ordering of the points (whatever they represent) remains unknown. Indeed, consider e.g. recognizing a shape in the picture as shown in Fig. 1. A desired output forms an image of a dog built from various curves (or lines). However, upon detecting data points $\bar{Q}_m$ it is almost impossible to be certain of the proper intrinsic ordering. Nevertheless, still the latter is a prerequisite to any interpolation scheme subsequently chosen to fit $\bar{Q}_m$. Thus a strong need for a scheme to fit such unordered data arises naturally here.



(a) (b) (c) (d)

Fig. 1. a) Original image, b) points deducted from image using Mathematica tools, c) points connected using curves, d) points connected using lines.

The second example refers to a medical image processing application. Consider e.g. a problem of encircling a tumor in the USG image (a classical segmentation problem). A physician marks a border of a tumor and the system automatically passes an interpolating curve through it as seen in Fig. 2. This curve is meant to mimic the tumor boundary and serves as an automatic tumor segmentation tool. As it stands the data forms an ordered set Q_m provided the physician stores the order of the marked points. This may not be the case. Additionally, another specialist may add later a point (or points) in the middle (judged by him/her as vital for further medical examination). This would result in unordered points $\bar{Q}_m$ for which a need of imposing an appropriate ordering reappear again.



(a) (b)

Fig. 2. *Hamartoma mammae* a) encircled without and b) with points ordering.

Before passing to the detailed description of the proposed algorithm designed to order an unordered set of points $\bar{Q}_m$ we recall a definition of the *e-graph of points* (see [4]) which in turn is used as an auxiliary tool in the subsequent interpolation scheme.

Definition 1. *We call an* e-graph of points *based on the set of planar points* $P = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ *as the graph* (V, E, f) *with* $V = \{1, 2, \ldots, n\}$, $E = V^2$, $f : \{1, \ldots, n\} \times \{1, \ldots, n\} \to \mathbb{R}$, *where* $f(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

Obviously e-graph of points P has the corresponding adjacency matrix $A \in M_{n,n}(\mathbb{R})$, $A = [a_{i,j}]_{i,j=1}^n$, where $a_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

Lemma 1. *E-graph of points and the corresponding adjacency matrix can be easily constructed from each other.*

Proof. Step 1: Constructing adjacency matrix from given graph:
Let (V, E, f) be e-graph of points, $n = \text{card}(V)$. Define $A \in M_{n,n}(\mathbb{R})$, $A = [a_{i,j}]$, where $a_{i,j} = f(i, j)$. A is a needed matrix.
Step 2: Constructing graph from given adjacency matrix
Let $A \in M_{n,n}(\mathbb{R})$, $A = [a_{i,j}]_{i,j=1}^n$. Define $f : \{1, \ldots, n\} \times \{1, \ldots, n\} \to \mathbb{R}$, $f(i, j) = a_{i,j}$. $(\{1, 2, \ldots, n\}, \{1, 2, \ldots, n\}^2, f)$ is a needed graph.

□

Remark 1. Note that due to Lemma 1 an e-graph and adjacency matrix can be used interchangeably.

Once the order of interpolation points is fixed with $\hat{Q}_m = \{\hat{q}_i\}_{i=0}^m$ (appropriately permuted ordered points from $\bar{Q}_m$) the corresponding approximating knots $\{\hat{t}_i\}_{i=0}^m$ can be chosen in accordance with *cumulative chords* (see [5]), for $i = 1, 2, \ldots, m$:

$$\hat{t}_0 = 0, \qquad \hat{t}_i = \hat{t}_{i-1} + \|\hat{q}_i - \hat{q}_{i-1}\|. \tag{1}$$

Based on $(\{\hat{t}_i\}_{i=0}^m, \hat{Q}_m)$ an interpolation scheme $\hat{\gamma}$ (in this paper chosen as a piecewise-quadratic curve $\hat{\gamma}_2$) can be applied. A detailed description of piecewise-quadratic Lagrange interpolation can be found e.g. in [6] or [7].

3 Description of the Algorithm

3.1 Example of Curves and Samplings

We introduce now the curves and samplings used later for testing both on sparse and dense data $\bar{Q}_m$ (and thus on $\hat{Q}_m$). In particular the case when $m \to \infty$ is needed to perform numerical experiments to estimate the convergence order in γ approximation by $\hat{\gamma}_2$.

a) Curves

Example 1. (i) Define first a simple planar spiral γ_{sp} - see Fig. 3a:

$$\gamma_{sp}(t) = ((t + 0.2)\cos(\pi(1 - t)), (t + 0.2)\sin(\pi(1 - t))) \in E^2, \text{ for } t \in [0, 1], \tag{2}$$

and also another planar spiral γ_{spl} - see Fig. 3b:

$$\gamma_{spl}(t) = ((6\pi - t)\cos(t), (6\pi - t)\sin(t)) \in E^2, \text{ for } t \in [0, 5\pi].$$

As easily verifiable both γ_{sp} and γ_{spl} are regular curves, i.e. curves for which $\dot{\gamma} \neq \mathbf{0}$. □

b) Samplings

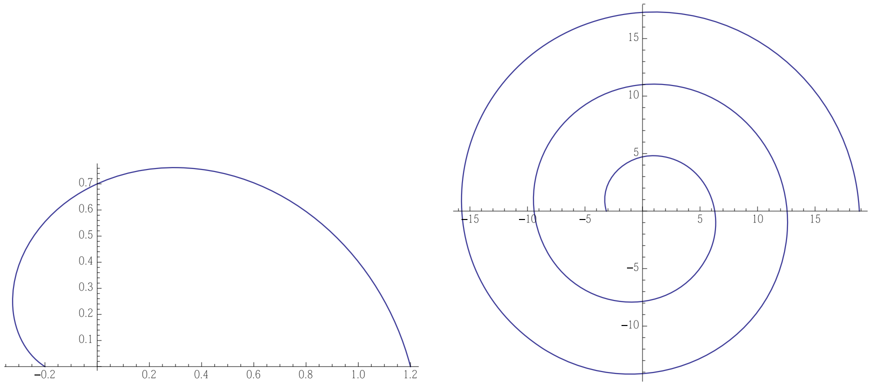
Three types of testing samplings (other random samplings can be found in [5]) are considered in this paper. They are needed to simulate the points $q_i = \gamma(t_i)$ (here $\{Q_m\}_{i=0}^m$) on the trajectory of γ. Once the points are generated their implicit order marked by the ascending order of the interpolation knots $\{t_i\}_{i=0}^m$ is erased which renders the unordered set of reduced data $\bar{Q}_m$. This set in turn will be latter converted to the ordered set $\hat{Q}_m$.

The first selected sampling for our tests is a uniform sampling (for $0 \leq i \leq m$):

$$t_i = \frac{i}{m}. \tag{3}$$

The second sampling applied reads as:

$$t_i = \begin{cases} \frac{i}{m}, & \text{if } i \text{ is even,} \\ \frac{i}{m} + \frac{1}{2m} & \text{if } i = 4k + 1, \\ \frac{i}{m} - \frac{1}{2m} & \text{if } i = 4k + 3. \end{cases} \tag{4}$$



(a) (b)

Fig. 3. The trajectories of the testing curves: a) γ_{sp}, b) γ_{spl}.

The last sampling considered here is defined as follows:

$$t_i = \frac{i}{m} + \frac{(-1)^{i+1}}{3m}. \tag{5}$$

Note that all samplings (3), (4) and (5) meet a requirement on general samplings permitting legal distribution of $\{t_i\}_{i=0}^{m}$. Namely, the following necessary *admissibility condition* must hold (if asymptotical analysis is to be later performed):

$$\lim_{m\to\infty} \delta_m = 0, \quad \text{where} \quad \delta_m = \max_{0\le i\le m-1}(t_{i+1} - t_i). \tag{6}$$

□

3.2 Open Curves

In this subsection we assume to deal with the open curves. First let us define an open curve.

Definition 2. *The curve* $\gamma : [a, b] \to \mathbb{E}^n$ *is called* open *if* $\gamma(a) \neq \gamma(b)$ *(see [8]).*

If curve γ is closed (i.e. $\gamma(a) = \gamma(b)$), the experiments showed that our algorithm converting $\bar{Q}_m$ into $\hat{Q}_m$ does not perform so well and this remains an open problem.

The algorithm enforcing an order in a given set of points $\bar{Q}_m$ (rendering ordered set $\hat{Q}_m$) is formulated below. Subsequently this procedure is used later for interpolating $\hat{Q}_m$ (and thus also $\bar{Q}_m$).

In Algorithm 1, ε is to be chosen arbitrarily small (to prevent loops to appear).

One can easily see that Algorithm 1 is also well defined for an arbitrary Euclidean E^n space. However, the effectiveness in E^n spaces other than E^2 is not a subject of this paper.

Algorithm 1. Finding order of interpolation points for open curves

```
function FIND ORDERING OF POINTS(point_set : set of points )
    graphMatrix ← {}
    for i = 1; i ≤ Length(point_set); i++ do
        Append {} to end of graphMatrix
        for k = 1; i ≤ Length(point_set); k++ do
            e ← euclidean distance between i-th and k-th points in point_set
            if e < ε then
                e ← ∞
            if i == k then
                e ← ∞
            Append e to end of graphMatrix[i]
    g ← graph based on graphMatrix as its weighted adjacency matrix
    tree ← minimal spanning tree of graph g
    root ← first node of tree
    ordering ← order of vertices in g based on depth first scan using root as starting point
return point_set ordered by ordering
```

At this point we illustrate the execution of the above Algorithm 1. In doing so, consider γ_{sp} defined as in (2) to be sampled uniformly (3) with $m = 10$ (the number of interpolation points is 11). The interpolation knots $\{t_i\}_{i=0}^{10}$ are as follows:

$$A = \left(0, \frac{1}{10}, \frac{1}{5}, \frac{3}{10}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{7}{10}, \frac{4}{5}, \frac{9}{10}, 1\right),$$

which ultimately yields the points $\{(t_i, \gamma_{sp}(t_i))\}_{i=0}^{10}$ positioned on the trajectory of the curve γ_{sp}:

$$\begin{aligned} Q_{10} = \gamma_{sp}(A) = &\{(-0.2000, 0.0000), (-0.2853, 0.09271), (-0.3236, 0.2351), \\ &(-0.2939, 0.4045), (-0.1854, 0.5706), (0.0000, 0.7000), \\ &(0.2472, 0.7608), (0.5290, 0.7281), (0.8090, 0.5877) \\ &(1.0461, 0.3399), (1.2000, 0.0000)\} \subset \mathbb{E}^2. \end{aligned}$$

To simulate now the unordered set $\bar{Q}_{10}$ we apply an arbitrary permutation of the set of $\{0, 1, 2, \ldots, 10\}$:

$$P = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 5 & 6 & 4 & 10 & 0 & 3 & 9 & 2 & 1 & 8 & 7 \end{pmatrix}.$$

This permutation results in the following random ordering of points:

$$\begin{aligned} \bar{Q}_{10} = \gamma_{sp}(A)^P = &((0.0000, 0.7000), (0.2472, 0.7608), (-0.1854, 0.5706), \\ &(1.2000, 0.0000), (-0.2000, 0.0000), (-0.2939, 0.4045), \\ &(1.0462, 0.3399), (-0.3236, 0.2351), (-0.2853, 0.0927), \\ &(0.8090, 0.5877), (0.5290, 0.7281)). \end{aligned}$$

In the next step, based on $\bar{Q}_m$ (see also Definition 1), an e-graph $G_{\bar{Q}_m}$ from these points can be created with the corresponding adjacency matrix (see Remark 1):

$$A_{\bar{Q}_m} = \begin{pmatrix} \infty & 0.255 & 0.226 & 1.39 & 0.728 & 0.417 & 1.11 & 0.566 & 0.671 & 0.817 & 0.53 \\ 0.255 & \infty & 0.473 & 1.22 & 0.883 & 0.648 & 0.903 & 0.776 & 0.854 & 0.588 & 0.284 \\ 0.226 & 0.473 & \infty & 1.5 & 0.571 & 0.198 & 1.25 & 0.363 & 0.488 & 0.995 & 0.732 \\ 1.39 & 1.22 & 1.5 & \infty & 1.4 & 1.55 & 0.373 & 1.54 & 1.49 & 0.706 & 0.99 \\ 0.728 & 0.883 & 0.571 & 1.4 & \infty & 0.415 & 1.29 & 0.266 & 0.126 & 1.17 & 1.03 \\ 0.417 & 0.648 & 0.198 & 1.55 & 0.415 & \infty & 1.34 & 0.172 & 0.312 & 1.12 & 0.884 \\ 1.11 & 0.903 & 1.25 & 0.373 & 1.29 & 1.34 & \infty & 1.37 & 1.35 & 0.343 & 0.647 \\ 0.566 & 0.776 & 0.363 & 1.54 & 0.266 & 0.172 & 1.37 & \infty & 0.147 & 1.19 & 0.985 \\ 0.671 & 0.854 & 0.488 & 1.49 & 0.126 & 0.312 & 1.35 & 0.147 & \infty & 1.2 & 1.03 \\ 0.817 & 0.588 & 0.995 & 0.706 & 1.17 & 1.12 & 0.343 & 1.19 & 1.2 & \infty & 0.313 \\ 0.53 & 0.284 & 0.732 & 0.99 & 1.03 & 0.884 & 0.647 & 0.985 & 1.03 & 0.313 & \infty \end{pmatrix}.$$

The respective e-graph $G_{\bar{Q}_m}$ together with the associated weights (taken accordingly from $A_{\bar{Q}_m}$) can be seen in Fig. 4.

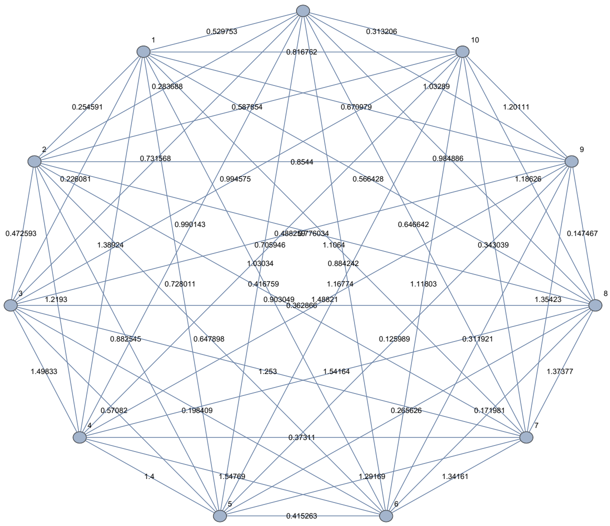


Fig. 4. E-graph created from points $\gamma_{sp}(A)^P$.

The minimal spanning tree (see [4]) generated from this graph can be seen on Fig. 5.

Fig. 5. Minimal spanning tree of e-graph created from points $\gamma_{sp}(A)^P$.

The minimal tree from Fig. 5 is highlighted on graph in Fig. 6.

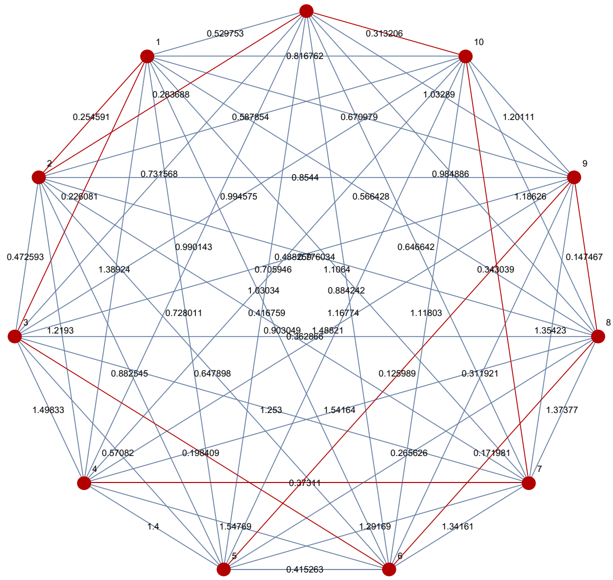


Fig. 6. E-graph created from points $\gamma_{sp}(A)^P$ with highlighted minimal spanning tree.

Having found the minimal tree, we can naturally enforce the ordering in $\bar{Q}_m$ since the depth first scan through graph yields the desired points in order (see [4]). Consequently our scheme results in the explicit ordering of the points (rendering therefore the set $\hat{Q}_m$):

$$
\begin{aligned}
&((1.2000, 0.000), (1.0461, 0.3399), (0.8090, 0.5878),\\
&(0.5290, 0.7281), (0.2472, 0.7608), (0.0000, 0.7000),\\
&(-0.1854, 0.5706), (-0.2939, 0.4045), (-0.32360, 0.2351),\\
&(-0.2853, 0.09271), (-0.2000, 0.0000)).
\end{aligned}
$$

In the last step a piecewise-quadratic Lagrange interpolation $\hat{\gamma}_2$ is invoked (with $\{\hat{t}_i\}_{i=0}^m$ defined according to (1)) and based on either ordered or unordered set of reduced data i.e. either on $\hat{Q}_m$ or on $\bar{Q}_m$, respectively. The interpolating curves are presented in Fig. 7.

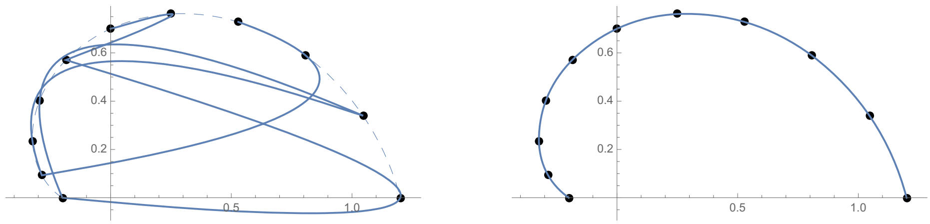


Fig. 7. a) Interpolation based on $\gamma_{sp}(A)^P$ without and b) with points ordering

One can easily see that ordering of points is the reverse original order. Which does not matter for the interpolation - trajectory is the same, however the curve $\hat{\gamma}_2$ passes in the opposite direction. Therefore the generated interpolant is optimal for these points, i.e. the same as generated from the points with original ordering. □

We exploit here one particular phenomenon - when given points are forming open curve, it is safe to assume that minimal spanning tree of e-graph is actually a linear graph. Therefore from our observation and asymptotic testing, minimal spanning tree of e-graph generates an optimal ordering of points. This result combined with proven theorem from [7] lay foundations to the formulation of the following conjectured result (to hold asymptotically):

Theorem 1. *Let a regular $\gamma \in C^3$ be sampled according to the admissibility condition (6). Assume Algorithm 1 for unordered reduced data $\bar{Q}_m$ yields $\hat{Q}_m$ with ordering coinciding with the unknown interpolation knots $\{t_i\}_{i=0}^m$. Then a piecewise-quadratic interpolant $\hat{\gamma}_2 : [0, \hat{T}] \to E^2$ fitting $\hat{Q}_m$ with $\{\hat{t}_i\}_{i=0}^m$ (see (1) built on $\hat{Q}_m$) renders:*

$$\hat{\gamma}_2 \circ \psi = \gamma + O(\delta_m^3), \tag{7}$$

where $\psi : [0, T] \to [0, \hat{T}]$ is a piecewise-quadratic Lagrange interpolant defined as in [9].

It should be underlined here that ψ is not needed for the construction of the Algorithm 1 and the interpolant $\hat{\gamma}_2$ based on unordered data $\bar{Q}_m$. However, to test numerically (or for proving analytically Th. 1) it is necessary to introduce such reparameterization as both curves γ and $\hat{\gamma}_2$ need to be compared only over the same domain $[0, T]$. The interpolant in question $\hat{\gamma}_2$ is evidently defined over external domain $[0, \hat{T}]$, which generically does not coincide with the internal one i.e. with $[0, T]$. The function ψ is defined here a sum-track of the $\psi_i : [t_i, t_{i+2}] \to [\hat{t}_i, \hat{t}_{i+3}]$ (with $i = 2k$), where each ψ_i is a quadratic satisfying $\psi_i(t_{i+j}) = \hat{t}_{i+j}$, for $j = 0, 1, 2$. Note also that if the order of $\hat{Q}_m$ is the same as $\{t_i\}_{i=0}^m$ then ψ_i preserves the γ motion along t and (7) examines a real difference between the interpolant $\hat{\gamma}_2$ and the curve γ. Our conjecture is that for reasonable curves, admissible samplings $\{t_i\}_{i=0}^m$ and dense data $\bar{Q}_m$ the Algorithm 1 determines the set $\bar{Q}_m$ having the same order as original interpolation knots $\{t_i\}_{i=0}^m$ - this remains still an open problem to be resolved analytically. However, the numerical tests performed in the next section confirm the above fact, at least for all samplings and curves considered in this paper. Once the latter is achieved the asymptotics in (7) follows from [9]. Finally, note that ψ should be a re-parameterization, which also is proved not always to hold in [9]. A recent result [10] formulates sufficient conditions imposed on $\{t_i\}_{i=0}^m$ to guarantee that ψ is a genuine reparameterization. Note that the corresponding result established for ordered data points Q_m is also discussed in [7]. In the next section the numerical verification of Theorem 1 is experimentally accomplished.

4 Experiments

Our tests are performed in *Mathematica* 10.0.0 using Intel Core i7 3.5 GHz processor with 32 GiB of RAM.

We introduce now a formal definition of convergence orders.

Definition 3. *Consider the family $F_{\delta_m} : [0, T] \to E^n$ (in our case $F_{\delta_m} = (\hat{\gamma} \circ \psi - \gamma)(t)$). We say that $F_{\delta_m} = O(\delta_m^\alpha)$ if $\|F_\alpha\| = O(\delta_m^\alpha)$ (where $\|\cdot\|$ denotes the Euclidean norm). The latter can be reformulated to: $\exists_{K>0}\exists_{\bar{\delta}}\|F_{\delta_m}\| \leq K\delta_m^\alpha$, for all $\delta_m \in (0, \bar{\delta})$ and $t \in [0, T]$.*

Since $T = \Sigma_{i=1}^m(t_{i+1} - t_i) \leq m\delta_m$ the following holds $m^{-\alpha} = O(\delta_m^\alpha)$, for arbitrary $\alpha > 0$ mentioned in Definition 3 (see also [7]). Therefore, for the verification of any asymptotics expressed in terms of $O(\delta_m^\alpha)$ it is sufficient to examine the claims of Th. 1 in terms of $O(1/m^\alpha)$ asymptotics.

Recall that for a parametric smooth planar curve $\gamma : [0, T] \to E^2$ (with $[0, T]$ compact) and m varying between $m_{min} \leq m \leq m_{max}$ the i-th component of the error for γ estimation by $\hat{\gamma}^i$ is defined as follows:

$$E_m^i = \sup_{t\in[t_i,t_{i+2}]} \|(\hat{\gamma}^i \circ \psi_i)(t) - \gamma(t)\| = \max_{t\in[t_i,t_{i+2}]} \|(\hat{\gamma}^i \circ \psi_i)(t) - \gamma(t)\|. \quad (8)$$

The maximal value E_m for each $m = 2k$ is found by using *Mathematica* numerical optimization function: *NMaximize* [11]. From the set of *absolute errors* $\{E_m\}_{m=m_{min}}^{m=m_{max}}$ the numerical estimate of α is calculated using a linear regression applied to the collection of points $(\log(m), -\log(E_m))$ (where $m_{min} \leq m \leq m_{max}$). The *Mathematica's* built-in function *LinearModelFit* renders the estimated coefficient α from the computed regression line $y(x) = \hat{\alpha}x + b$. The results estimating α from (7) are presented in Table 1 for three types of samplings: (3), (5), (4) and two testing curves γ_{sp} and γ_{spl} using $\varepsilon = 0.001$ introduced in Algorithm 1.

Tests (performed with $m \in \{151, \ldots, 181\}$) shown in Table 1 visibly confirm Theorem 1.

Table 1. Result of numerical experiments

Curve	Sampling	$\hat{\alpha} \approx \alpha$
γ_{sp}	(3)	3.017
γ_{sp}	(5)	3.021
γ_{sp}	(4)	3.015
γ_{spl}	(3)	3.015
γ_{spl}	(5)	3.017
γ_{spl}	(4)	3.02

5 Conclusion

In this paper we verify numerically the asymptotics (7) from Theorem 1. In doing so, first an Algorithm 1 for generating an interpolation scheme $\hat{\gamma}_2$ based on planar unordered reduced data set $\bar{Q}_m = \{\bar{q}_i\}_{i=0}^m$ is introduced. More specifically, Lagrange piecewise-quadratic interpolation $\hat{\gamma}_2$ combined with cumulative chords and a procedure of converting unordered reduced data $\bar{Q}_m$ into ordered reduced data $\hat{Q}_m$ are introduced and tested on both sparse and dense data. Additionally, our experiments confirm cubic order of convergence in trajectory approximation by the proposed data fitting scheme. This asymptotic results coincides with already proved one for the ordered data Q_m - see [9]. Naturally, the data fitting scheme in question is extendable to other interpolation schemes and to multidimensional unordered reduced data.

The open problems include interpolating the ordered reduced data positioned on the trajectory of closed curve γ. The theoretical proof of asymptotics observed and verified here forms another possible research task. This would immediately follow once we find sufficient conditions guaranteeing that asymptotically the ordering in $\hat{Q}_m$ coincides with the order of interpolation knots. This is our conjecture and it remains as another open problem. One can also analyze the asymptotics in approximating other geometrical features including length (see e.g. [12] or [13]) or curvature of γ. Finally, the next open problem may include testing effectiveness of our data fitting algorithm in E^3 (or in E^n).

More discussion on applications (including real data examples - see [2] or [14]) and theory of non-reduced data interpolation can be found in [3], [5], [15], [16], [17], [18], [19], [20] or [21]. In particular different parameterizations $\{\hat{t}\}_{i=0}^m$ of the unknown interpolation knots $\{t\}_{i=0}^m$ are discussed e.g. in [1], [22], [23] or [24].

References

1. Kvasov, B.I.: Methods of Shape-Preserving Spline Approximation. World Scientific Publishing Company, Singapore (2000)
2. Janik, M., Kozera, R., Kozioł, P.: Reduced data for curve modeling - applications in graphics, computer vision and physics. Advances in Science and Technology **7**(18), 28–35 (2013)
3. Piegl, L., Tiller, W.: The NURBS Book. Springer, Heidelberg (1997)
4. Ross, K.A., Wright, C.R.B.: Discrete Mathematics. Pearson (2002)

5. Kozera, R.: Curve modeling via interpolation based on multidimensional reduced data. Studia Informatica **25**(4B–61), 1–140 (2004)
6. De Boor, C.: A Practical Guide to Splines. Springer, Heidelberg (2001)
7. Kozera, R., Noakes, L., Szmielew, P.: Trajectory estimation for exponential parameterization and different samplings. In: Saeed, K., Chaki, R., Cortesi, A., Wierzchoń, S. (eds.) CISIM 2013. LNCS, vol. 8104, pp. 430–441. Springer, Heidelberg (2013)
8. Krantz, S.G.: Handbook of Complex Variables. Springer Science+Business Media, New York (1999)
9. Noakes, L., Kozera, R.: Cumulative chord piecewise quadratics and piecewise cubics. In: Klette, R., Kozera, R., Noakes, L., Weickert, J. (eds.) Geometric Properties for Incomplete Data Computational Imaging and Vision, vol. 31, pp. 59–76. Springer (2006)
10. Kozera, R., Noakes, L.: Piecewise-quadratics and reparameterizations for interpolating reduced data. In: Gerdt, V., Koepf, W., Seiler, W., Vorozhtsov, E. (eds.) CASC 2015. LNCS, vol. 9301, pp. 260–274. Springer Int. Pub. Switzerland (2015)
11. Wolfram Mathematica 9, Documentation Center. http://reference.wolfram.com/mathematica/guide/Mathematica.html
12. Kozera, R., Noakes, L., Klette, R.: External versus internal parameterizations for lengths of curves with nonuniform samplings. In: Asano, T., Klette, R., Ronse, C. (eds.) Geometry, Morphology, and Computational Imaging. LNCS, vol. 2616, pp. 403–418. Springer, Heidelberg (2003)
13. Noakes, L., Kozera, R.: More-or-less uniform sampling and lengths of curves. Quarterly of Applied Mathematics **61**(3), 475–484 (2003)
14. Bator, M., Chmielewski, L.: Finding regions of interest for cancerous masses enhanced by elimination of linear structures and considerations on detection correctness measures in mammography. Pattern Analysis and Applications **12**(4), 377–390 (2009)
15. Kozera, R., Noakes, L.: Piecewise-quadratics and exponential parameterization for reduced data. Applied Mathematics and Computation **221**, 620–638 (2013)
16. Kozera, R., Noakes, L., Szmielew, P.: Length estimation for exponential parameterization and ϵ-uniform samplings. In: Huang, F., Sugimoto, A. (eds.) PSIVT 2013. LNCS, vol. 8334, pp. 33–46. Springer, Heidelberg (2014)
17. Farin, G.: Curves and Surfaces for Computer Aided Geometric Design, 3rd edn. Academic Press, San Diego (1993)
18. Epstein, M.P.: On the influence of parameterization in parametric interpolation. SIAM Journal of Numerical Analysis. **13**, 261–268 (1976)
19. Kozera, R., Noakes, L.: C^1 interpolation with cumulative chord cubics. Fundamenta Informaticae **61**(3–4), 285–301 (2004)
20. Homenda, W., Pedrycz, W.: Processing uncertain information in the linear space of fuzzy sets. Fuzzy Sets and Systems **44**(2), 187–198 (1991)
21. Budzko, D.A., Prokopenya, A.N.: Symbolic numerical methods for searching equilibrium states in a restricted four-body problem. Programming and Computer Software **39**(2), 74–80 (2013)
22. Mørken, K., Scherer, K.: A general framework for high-accuracy parametric interpolation. Mathematics of Computation **66**(217), 237–260 (1997)
23. Kocić, L.M., Simoncelli, A.C., Della Vecchia, B.: Blending parameterization of polynomial and spline interpolants, Facta Universitatis (NIŠ). Series Mathematics and Informatics **5**, 95–107 (1990)
24. Lee, E.T.Y.: Choosing nodes in parametric curve interpolation. Computer-Aided Design **21**(6), 363–370 (1987)