Interpolating Strange Attractors via Fractional Brownian Bridges

Sebastian Raubitzek *, Thomas Neubauer, Jan Friedrich, Andreas Rauber

Posted Date: 18 April 2022

doi: 10.20944/preprints202204.0167.v1

Keywords: time series interpolation; phase space reconstruction; takens' theorem; interpolation; stochastic interpolation; genetic algorithm; time series data; preprocessing; strange attractor; attractor; attractor reconstruction

Article

Interpolating Strange Attractors via Fractional Brownian Bridges

Sebastian Raubitzek*[1], Thomas Neubauer[2], Jan Friedrich[3] and Andreas Rauber[4]

[1] TU Wien, Information and Software Engineering Group, Favoritenstrasse 9-11/194, 1040 Vienna, Austria; sebastian.raubitzek@gmail.com
[2] TU Wien, Information and Software Engineering Group, Favoritenstrasse 9-11/194, 1040 Vienna, Austria; thomas.neubauer@tuwien.ac.at
[3] ForWind, Institute of Physics, University of Oldenburg, Küpkersweg 70, 26129 Oldenburg, Germany; jan.friedrich@uni-oldenburg.de
[4] TU Wien, Information and Software Engineering Group, Favoritenstrasse 9-11/194, 1040 Vienna, Austria; rauber@ifs.tuwien.ac.at
* Correspondence: sebastian.raubitzek@gmail.com

Abstract: We present a novel method for interpolating univariate time series data. The proposed method combines multi-point fractional Brownian bridges, a genetic algorithm, and Takens' theorem for reconstructing a phase space from univariate time series data. The basic idea is to first generate a population of different stochastically interpolated time series data, and secondly, to use a genetic algorithm to find the pieces in the population which generate the smoothest reconstructed phase space trajectory. A smooth trajectory curve is hereby found to have a low variance of second derivatives along the curve. For simplicity, we refer to the developed method as *PhaSpaSto*-interpolation, which is an abbreviation for **pha**se-**spa**ce-trajectory-smoothing **sto**chastic interpolation. The proposed approach is tested and validated with a univariate time series of the Lorenz system, five non-model data sets and tested against a cubic spline interpolation and a linear linear interpolation. We find that the criterion for smoothness guarantees low errors on known model and non-model data. Finally, we interpolate the discussed non-model data sets, and show the corresponding improved phase space portraits. The proposed method is useful for interpolating low-sampled time series data sets for, e.g., machine learning, regression analysis, or time series prediction approaches.

Keywords: time series interpolation; phase space reconstruction; takens' theorem; interpolation; stochastic interpolation; genetic algorithm; time series data; preprocessing; strange attractor; attractor; attractor reconstruction;

1. Introduction

Many real-life time series data sets originate from complex real-life systems and/or non-linear phenomena. Often these data sets are sparsely sampled as, e.g., long-term temperature, yield, or environmental data sets. The non-linear and stochastic nature of these data sets, in addition to being sparsely sampled, makes predictions and analysis rather difficult. Thus, one tends to employ data augmentation, and, in the case of univariate time-series data, interpolation techniques. There are various interpolation techniques, such as linear, polynomial, fractal, and stochastic interpolation methods. Choosing a suitable interpolation method can be difficult, and one should take into account the characteristics of the data at hand. For fluctuating data with inherent randomness, for example one would choose a stochastic interpolation as discussed in [1]. But when it comes to deterministically chaotic systems where one can reconstruct a phase space based on Taken's theorem [2], the choice is not so obvious. Here, we present a method taking into account the reconstructed phase space properties of time series data. Thereby, we want our reconstructed phase space trajectories to be as smooth as possible. But how can one achieve a smooth phase space trajectory? To solve this

 doi:10.20944/preprints202204.0167.v1

issue, we developed a method combining multi-point Brownian bridges [1] and a genetic algorithm. For simplicity, we refer to the developed method as *PhaSpaSto*-interpolation, which is an abbreviation for **pha**se-**spa**ce-trajectory-smoothing **sto**chastic interpolation. We show, that the developed method performs well for the reconstructed phase space of the Lorenz system and several univariate, sparsely sampled time series data. The results show that the presented method effectively can interpolate the Lorenz system and the discussed non-model data sets with comparatively low errors on known data points and convincing phase space portraits.

Furthermore, many of today's most employed time series analysis and prediction techniques stem from the domain of machine and/or deep learning. These methods are data-based, i.e., they learn from data; thus, a sufficient amount of data and data of good quality is necessary to, e.g., train a neural network. It is shown that interpolating time series data using a fractal or linear interpolation can improve the accuracy of the algorithm drastically, [3]. We thus suggest the presented method be tested and used for data-based learning algorithms.

This article is structured as follows: Section 2 collects publications related to the developed method and discusses them briefly. Section 3 describes the multi-point Brownian bridges [1], the Lorenz system, and the employed genetic algorithm, and further sums up the developed scheme. All results with the corresponding error tables and figures are discussed in Section 4. Section 5 concludes the findings of this article.

2. Related Work

The presented research is motivated by findings of [3] and [4]. It is further based on the stochastic interpolation method presented in [1]. Thus we will briefly describe the mentioned publications and chronologically, i.e. by their date of publication, list related ideas below.

- Ref [5]: This publication presents a method to determine if images are blurry. For this purpose, the second derivatives of grey-scale images are taken, and the corresponding variance over all pixels is analyzed. If the variance is below a certain threshold, the image is a blurry image. This concept is used in the presented article. We adapted the idea of variances of second derivatives, which is discussed in Section 3.3.1.
- Ref [6]: In this research, a combination of inverse distance methods, fuzzy set theory, and a genetic algorithm are applied to interpolate rainfall data. The Genetic algorithm was used to determine the parameters of the corresponding fuzzy membership functions. Thus the idea of improving interpolation techniques is adapted from this publication.
- Ref [1]: This publication presents a novel interpolation technique where the idea of a Brownian bridge, i.e., a constrained fractional Brownian motion (fBm), is extended to more than two points, i.e., to *multi-point fractional Brownian bridges*. The authors present an explicit construction that operates linearly on the fBm and can thus be interpreted as a Gaussian random process constrained on multiple, prescribed points. Further applications of this method are presented, such as determining optimal Hurst exponents for sparsely sampled time series filled up by multi-point fractional Brownian bridges with varying Hurst exponents. This method is used in the presented research to fuel the genetic algorithm.
- Ref [3]: In this publication, a fractal interpolation to interpolate univariate time series data is presented. The presented method considers the Hurst exponent of the data under study. The authors show that fractal interpolation can increase the predictability of a given univariate time series data. This research suggests that different interpolation methods for univariate time series data may yield predictions of different qualities. Thus, as presented here, an attractor-based interpolation is an obvious next step in contrast to a fluctuation-based interpolation.

 doi:10.20944/preprints202204.0167.v1

- Ref [4]: This publication is a continuation of [3]. The fractal interpolation and LSTM neural network approach is continued as ensembles of predictions. Randomly parameterized LSTM neural networks are generated from non-, linear-, and fractal-interpolated data. Afterward, these predictions are filtered based on their signal complexities. Some of the mentioned complexity measures require a suitable phase space embedding of the data under study and are related to the presented research in this article. Further, some of the data sets used here are discussed and predicted. We expect LSTM neural network predictions of stochastically interpolated data to outperform other interpolated approaches when considering the reconstructed phase space.
- Ref [7] describes a multi-point reconstruction of a given time series. The method is based on the assumption of Markovianity of the time series and a refinement algorithm is presented which allows to systematically fill up data points based on the empirically determined transition probability from one level to the next.

3. Methodology

The developed method consists of two steps, first generating a population of stochastically interpolated time series data, each with a different Hurst exponent. Firstly, these stochastically interpolated time series data are generated using multi-point fractional Brownian bridges, see Section 3.1. Secondly, these multi-point fractional Brownian bridges are improved using a genetic algorithm, to minimize the variance of second-order derivatives along the reconstructed phase space trajectory (see Section 3.3). The whole scheme is depicted in Figure 1. Finally, we briefly discuss the Lorenz system and its implementation in Section 3.4.

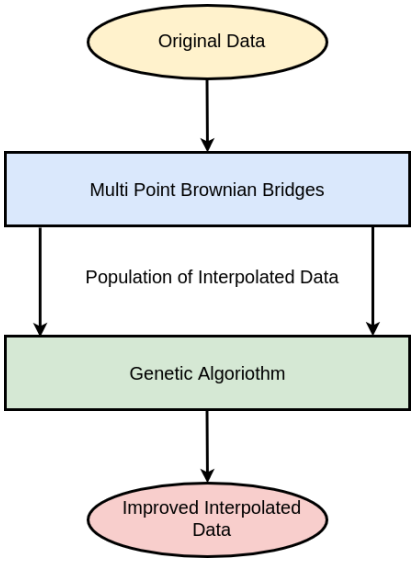


Figure 1. Depiction of the employed scheme.

3.1. Multi Point Fractional Brownian Bridges

As depicted in Figure 1, the employed genetic algorithm is fueled by a population of stochastically-interpolated time series data, in our case *multi-point fractional Brownian bridges*. To generate these stochastically interpolated time series data, multi-point fractional Brownian bridges [1] were used. Thus we briefly summarize this approach.

We consider a Gaussian random process $X(t)$ whose covariance is defined as $C(t,t') = \langle X(t)X(t')\rangle$. In the following, we focus on fractional Browian motion where the covariance is given according to $\langle X(t)X(t')\rangle = \frac{1}{2}\left(t^{2H} + t'^{2H} - |t-t'|^{2H}\right)$, where H is the Hurst exponent. To elucidate our interpolation scheme, we first define a so-called

 doi:10.20944/preprints202204.0167.v1

fractional Brownian bridge [8],[9], which is a construction of fBm starting from 0 at $t = 0$ and ending at X_1 at $t = t_1$, i.e.,

$$X^B(t) = X(t) - (X(t_1) - X_1)\frac{\langle X(t)X(t_1)\rangle}{\langle X(t_1)^2\rangle} \,. \tag{1}$$

This construction ensures that $X^B(t_1) = X_1$, which is also depicted in Figure 10. This single bridge can now be generalized to an arbitrary number of (non-equidistant) prescribed points X_i at t_i by virtue of a multi-point fractional Brownian bridge [1]

$$X^B(t) = X(t) - (X(t_i) - X_i)\sigma_{ij}^{-1}\langle X(t)X(t_j)\rangle \,, \tag{2}$$

where $\sigma_{ij} = \langle X(t_i)X(t_j)\rangle$ denotes the covariance matrix. Furthermore, we imply summation over identical indices. Latter linear operation on the Gaussian random process $X(t)$ ensures that the bridge takes on exactly the values X_k at t_k, which can be seen from $X^B(t_k) = X(t_k) - (X(t_i) - X_i)\sigma_{ij}^{-1}\sigma_{kj} = X(t_k) - (X(t_i) - X_i)\delta_{ik} = X_k$, where δ_{ik} denotes the Kronecker-delta. Hence, this method allows for the reconstruction of a sparse signal where small-scale correlations are determined by the choice of the Hurst exponent H.

3.2. Phase Space Reconstruction

Before describing the fitness of our phase space trajectories, we need to introduce the concept of reconstructed phase spaces, [10,11].

We estimate a phase space embedding for all data under study. To find a suitable phase space embedding, one has to determine two parameters, the embedding dimension and the time delay.

To estimate the time delay τ, i.e., the delay between two consecutive time steps, we use the method based on the average information between two signals [12].

To estimate the embedding dimension d_E, we use the algorithm of false nearest neighbors [13]. Also, because the evaluations performed in this paper aim to graphically depict the embedding space, but with no limitations to the general applicability of our approach, we only chose data sets with an embedding dimension of three, i.e. $d_E = 3$.

The phase space embedding for a given signal $[x_1, x_2, \ldots, x_n]$, thus is:

$$\vec{y}(i) \;=\; \Big[x_i, x_{i+\tau}, \ldots, x_{i+(d_E-1)*\tau}\Big] \quad , \tag{3}$$

and the corresponding three dimensional phase space embedding, thus is

$$\vec{y}(i) \;=\; [x_i, x_{i+\tau}, x_{i+2\tau}] \quad . \tag{4}$$

3.3. Genetic algorithm

We build a simple genetic algorithm to find the best possible interpolation given the data's phase space reconstruction using Taken's theorem. We want our reconstructed phase space curve to be as smooth as possible and thus define the trajectory's fitness as follows.

3.3.1. The Fitness of a Trajectory

The basic idea is to use a concept from image-processing, i.e., the blurriness of a picture, and apply it to phase space trajectories. This is because we want our trajectory as blurry, i.e., as smooth as possible. In image processing the blurriness is determined via second-order derivatives of grey-scale images at each pixel, [5]. We employ this concept, but instead of using it at each pixel, we calculate the variance of second-order derivatives along our phase space trajectories. Similar to the concept from image-processing, where the low variance of second-order derivatives implies more blurriness, curves with a low variance of second-order derivatives exhibit comparatively smooth trajectories. The reason here is intuitively clear, whereas curves with a high variance of second-order

 doi:10.20944/preprints202204.0167.v1

derivatives have a range of straight and pointy sections, curves with a low variance of second-order derivatives have a similar curvature along the trajectory and thus are smoother. Hence, in order to guarantee smoothness along the trajectory, we want this variance to be as low as possible, which thus is our loss L. Concluding, our fitness is maximal when our loss L is minimal.

Again we start with the phase space vector and the corresponding embedding dimension d_E and time delay τ (See Section 3.2) of each signal as

$$\vec{y}(i) = \left[x_i, x_{i+\tau}, \ldots, x_{i+(d_E-1)\cdot\tau}\right] \quad . \tag{5}$$

Thus we have one component for each dimension of the phase space. Consequently we can write the individual components as:

$$y_j(i) = x_{i+(j-1)*\tau} \quad , \tag{6}$$

where $j = 1, 2, \ldots, d_E$. We then take the second-order finite difference central derivative of a discrete function [14]:

$$u_j''(i) = x_{i+(j-1)*\tau+1} - 2x_{i+(j-1)*\tau} + x_{i+(j-1)*\tau-1} \quad , \tag{7}$$

at each point, and for each component. Next we add up all the components as:

$$u''(i) = \sqrt{\sum_{j=1}^{d_E} u_j''(i)^2} \quad . \tag{8}$$

And finally, we use the variance of the absolute values of second derivatives along the phase space curve as our loss L of a phase space trajectory:

$$L = \mathrm{Var}_i\left[u''(i)\right] \, . \tag{9}$$

3.3.2. Genetic Algorithm Architecture

The employed genetic algorithm consists of the following building blocks:

A candidate solution is an interpolated time series using a random Hurst exponent $H \in]0;1[$. The corresponding population of candidates is, e.g.m 1000 of these stochastically interpolated time series data with random Hurst exponents. A population of interpolated time series data is generated using the multi-point Brownian bridges such that, for each member of the population, a random Hurst exponent with $H \in]0;1[$ is chosen, which then defines the interpolation of the member of the population. After generating the population, all members are sorted with respect to their fitness, i.e., the lower the loss L, the better an interpolation is. The mating is implemented such that only the best 50%, with respect to the fitness, can mate to produce new offsprings. The mating is done such that, for every gene, i.e., each interpolation between two data points, there is a 50:50 chance to inherit it from either one of the parents. The mutation was implemented that, in each generation, there is a 20% chance that a randomly chosen interpolated time series is replaced with a new interpolated time series within a corresponding randomly chosen new Hurst exponent. Also, we implemented a criterion for aborting the program, which was fulfilled if the population fitness mean did not change for ten generations. This described procedure is then performed for 1000 generations. But, actually, the 1000 generations were never reached, as the criterion for abortion always triggered, and the program was ended, thus yielding the best interpolation with respect to the fitness of the phase space trajectories before reaching the 1000th generation.

3.4. The Lorenz System

For this research and to show the applicability of the proposed interpolation method, we chose the Lorenz system, [15] as a model to illustrate our ideas.

 doi:10.20944/preprints202204.0167.v1

The Lorenz system is a set of three nonlinear equations:

$$\begin{aligned} \frac{dx}{dt} &= 10(-x+y)\,, \\ \frac{dy}{dt} &= 28x - y - xz\,, \\ \frac{dz}{dt} &= xy - \frac{8}{3}z\,. \end{aligned} \tag{10}$$

We solved this system using a basic Runge-Kutta 4 approach, [16]. We chose the step size and length of the simulation with respect to the number of interpolation points to test the quality of our interpolation scheme,

$$dt = \frac{0.1}{n_I + 1}, \quad L = 200 \cdot (n_I + 1) \quad , \tag{11}$$

where dt is the step size and L is the length of the simulation. The initial conditions were chosen to be:

$$x = -8, \quad y = 8, \quad z = 27\,. \tag{12}$$

Finally, we need a univariate signal for the phase space reconstruction, and to test our method, thus we choose one of the three variables. Accordingly, here we chose $x(t)$.

4. Results

Here we present the results of the Genetic algorithm for all data sets; first for the Lorenz system, then for five non-model data sets. For both cases we validate the developed method such that we delete data points from the original time series and reconstruct the missing data points using the presented interpolation technique. Further, we tested the presented interpolation technique against the best random interpolation of the population, against a linear interpolation and against a cubic spline interpolation [17]. Both the linear and spline interpolation were performed using the python package `scipy`, **??**.For the validation we put emphasis on the Lorenz system, as the generated model data allows us to test arbitrary settings, i.e. using different numbers of missing data points. Contrary to that, for the non-model data sets we delete every second data point and reconstructed the missing data points using the presented method. For the non-model data sets we also present actual interpolation results, i.e. data sets with smoothed-out phase space trajectories.

4.1. Results for the Lorenz System

For the actual equations and further details on the Lorenz system, see Section 3.4. We perform our interpolation for a number of different interpolation points $N_I = \{1, 2, 3, 4, \ldots, 20\}$.

We develop the following experimental steps to assess the performance of our interpolation scheme:

1. Obtain a univariate time series from the Lorenz system.
2. Delete points from the data which will be reconstructed later on.
   Given some univariate time series data of the Lorenz system, $[x_1, x_2, \ldots, x_n]$, we extract certain data points with respect to the number of interpolation points $n_I \in N_I$ and the interpolated data set such that:

$$\begin{aligned} &[x_1, x_{n_I+2}, x_{2n_I+2}, \ldots, x_n] \quad \text{Original data points to be kept for interpolation,} \\ &[x_1, \hat{x}_1, \ldots, \hat{x}_{n_I}, x_{n_I+2}, \hat{x}_{n_I+1}, \ldots, \hat{x}_{2n_I+1}, x_{2n_I+2}, \hat{x}_{2n_I+2}, \ldots x_n] \quad \text{Interpolated data,} \end{aligned} \tag{13}$$

   where $\hat{x}_i$ are the new found interpolated data points.
3. Perform the interpolation according to Section 3.3.
4. Calculate RMSE for the interpolated data points with respect to the previously extracted original data points $[x_2, \ldots, x_{n_I+1}, x_{n_I+3}, \ldots, x_{2n_I+1}, \ldots]$. Do the same for the population mean and each time series of the initial population.

 doi:10.20944/preprints202204.0167.v1

Thus we obtained errors for the mean values of the initial population, for each time series *in* the initial population and the time series that was improved using the presented genetic algorithm. Also, from all randomly generated interpolations we picked the one with the lowest RMSE to test it against the gen. alg. improved ones. Here, the root mean squared error E_{RMSE}, which is applied throughout this article, is given as:

$$E_{RMSE} = \left(\frac{1}{n} \sum_{i=1}^{n} [\hat{x}_i - x_i]^2 \right)^{\frac{1}{2}} , \tag{14}$$

where x_i are the original data points, $\hat{x}_i$ are the predicted (in this case interpolated) values and n is the length of the signal.

Table 1: Errors for the interpolated data on the Lorenz system depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations and for the interpolation that was improved using the presented genetic algorithm. *Lowest RMSE in population* refers to the best randomly interpolated result, i.e. the one interpolation from the population that produced the lowest error by chance. We also featured the results for the linear and spline interpolation. We highlighted the interpolations where the genetic-algorithm-based interpolation outperformed the whole population of interpolations. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation. This table is depicted in Figure 3.

n_I	1	2	3	4	5	6	7	**8**	9	10
RMSE population mean	0.77419	0.88263	0.91026	0.89442	0.87013	0.86858	0.89120	**0.84220**	0.90323	0.88777
lowest RMSE in population	0.17939	0.18068	0.16757	0.19206	0.16126	0.18134	0.17782	**0.17211**	0.18216	0.19371
RMSE linear interpolated	0.42534	0.44752	0.44179	0.44185	0.41574	0.42406	0.42894	**0.40883**	0.43263	0.43353
RMSE spline interpolated	0.12263	0.11808	0.09968	0.12586	0.09678	0.12195	0.12280	**0.10862**	0.11554	0.13008
RMSE gen. alg. improved	1.03779	0.24381	0.17517	0.19488	0.16264	0.18182	0.17818	**0.17121**	0.18239	0.19374
below best %	74.4%	21.6%	4.3%	2.2%	1.4%	1.1%	0.7%	**0.1%**	0.8%	0.3%

n_I	11	**12**	**13**	**14**	**15**	16	17	18	19	20
RMSE population mean	0.90844	**0.92145**	**0.91509**	**0.90686**	**0.91750**	0.90326	0.90789	0.90080	0.88835	0.89651
lowest RMSE in population	0.18789	**0.19238**	**0.18693**	**0.18632**	**0.19943**	0.19640	0.19208	0.18449	0.18415	0.20291
RMSE linear interpolated	0.43649	**0.44211**	**0.43687**	**0.43423**	**0.44142**	0.43534	0.43720	0.43170	0.42685	0.43398
RMSE spline interpolated	0.12086	**0.12646**	**0.11530**	**0.11765**	**0.12532**	0.12873	0.13098	0.12581	0.11912	0.12581
RMSE gen. alg. improved	0.18816	**0.19141**	**0.18670**	**0.18626**	**0.19943**	0.19663	0.19215	0.18462	0.18441	0.20300
below best %	0.8%	**0.1%**	**0.1%**	**0.1%**	**0.1%**	0.8%	0.6%	0.6%	0.8%	0.6%

 doi:10.20944/preprints202204.0167.v1

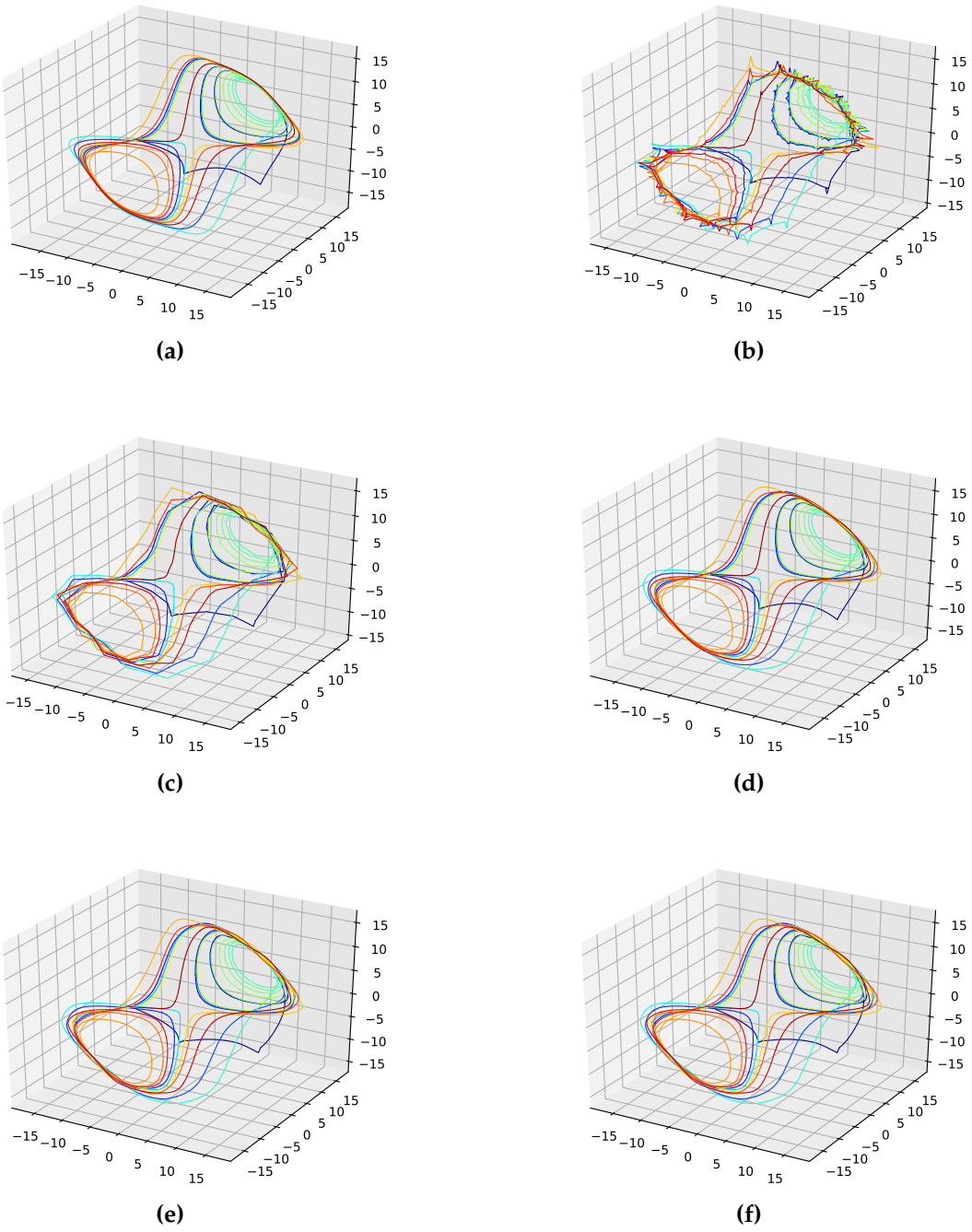


Figure 2. Reconstructed attractors for the interpolated Lorenz system.
(a): Non-interpolated original data (i.e. the one the errors are calculated with);
(b): The average interpolation of the whole population;
(c): Linear interpolated;
(d): Spline interpolated;
(e): The one interpolation of the population that has the lowest RMSE;
(f): Interpolation improved by the presented genetic algorithm approach.

 doi:10.20944/preprints202204.0167.v1

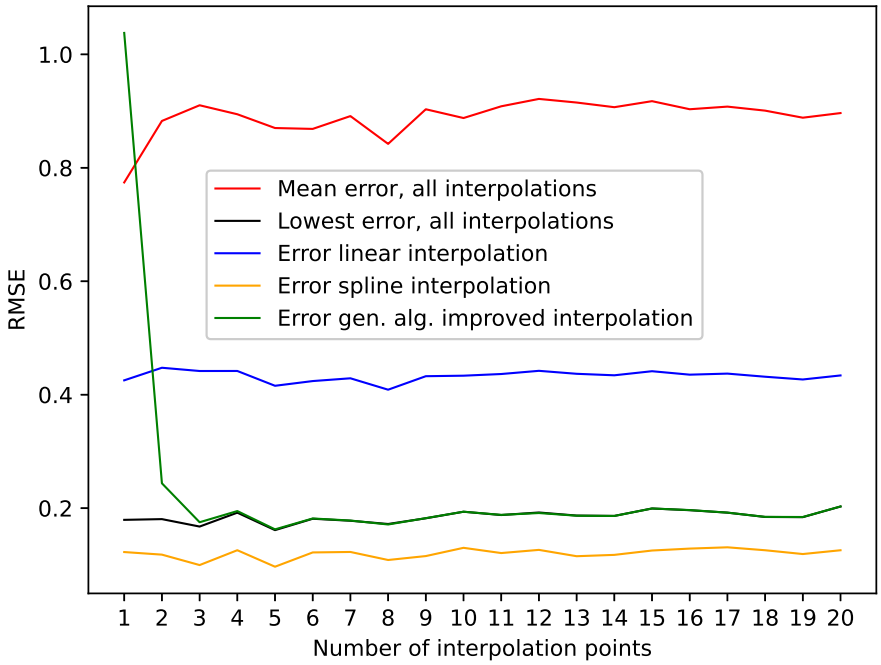


Figure 3. Shown in this Figure are the errors from Table 1 depending on the different numbers of interpolation points.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 April 2022 doi:10.20944/preprints202204.0167.v1

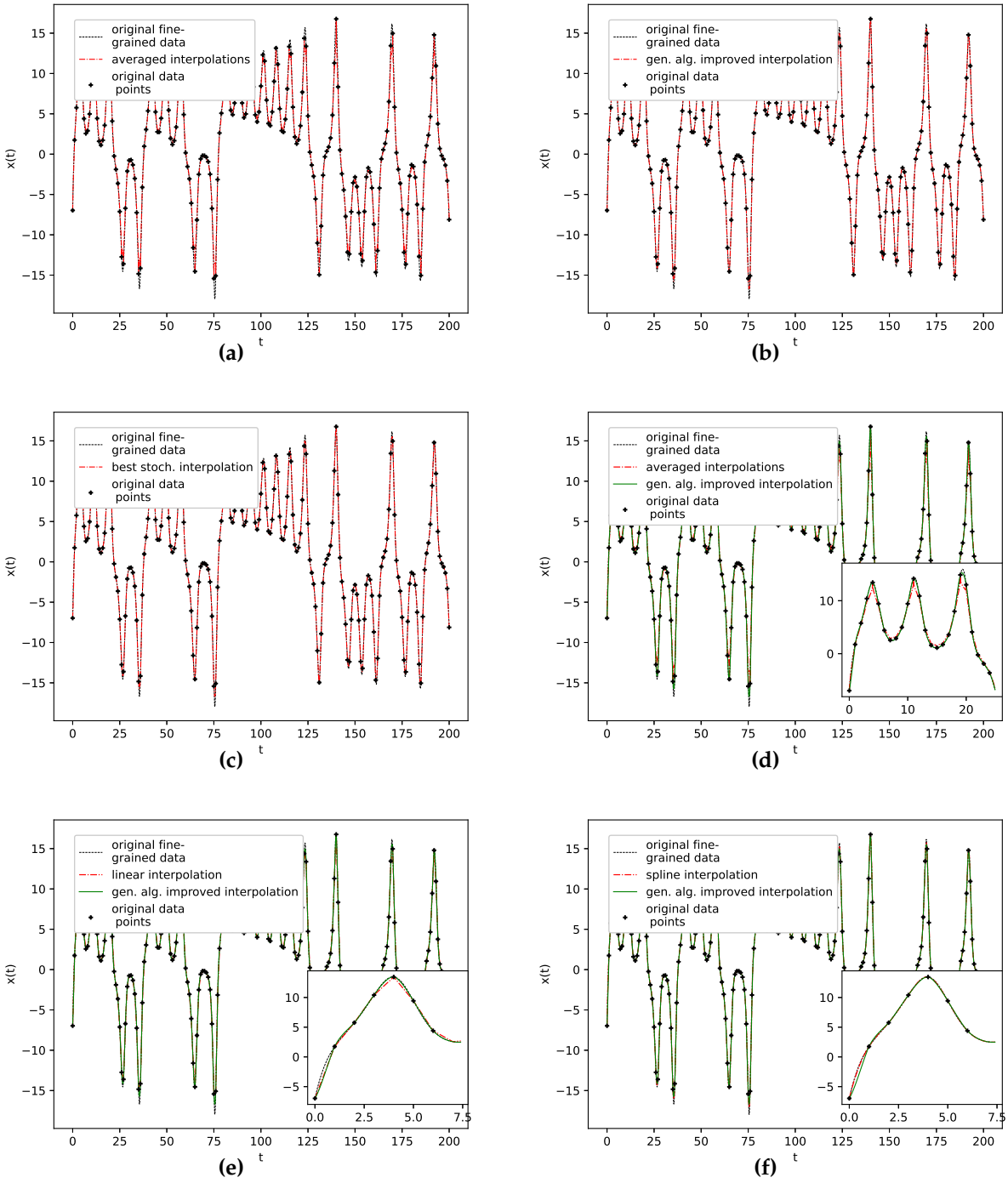


Figure 4. Original vs. interpolated time series data.
(a): Non-interpolated original data (i.e. the one the error's are calculated with) and population average;
(b): Genetic-algorithm-improved interpolation;
(c): The one interpolation of the population that has the lowest RMSE;
(d): Population average vs. genetic-algorithm-improved interpolation;
(e): Linear interpoaltion vs. genetic-algorithm-improved interpolation;
(f): Spline interpolation vs. genetic-algorithm-improved interpolation.

The presented results for the Lorenz system show that the algorithm can identify/generate the best interpolation in terms of a low RMSE on missing data points

 doi:10.20944/preprints202204.0167.v1

out of the given initial population. This can be seen in Table 1, where we highlighted the results where the genetic-algorithm-improved-interpolation outperformed every random interpolation of the population. Still the spline interpolation outperforms the presented approach. This is also depicted in Figure 3, where we plotted the RMSE on missing data points for varying numbers of interpolation points. This graphic shows that the presented approach requires a certain amount of interpolation points, in this case, three, to be close to the best random interpolation of the population. We assume that the reason for this is that the variance of second derivatives along a phase space trajectory requires a certain *density* of phase space points to be apple to differ between smooth and edgy phase space trajectories. The spline interpolation on the other hand performs well right from the start.

The corresponding reconstructed phase space plots (Figure 2) show that both the best random interpolation (e) and the genetic-algorithm-improved interpolation (f) provide convincing phase space portraits, as both are indeed close to the ground truth (a). On the other hand, the population mean (b) is far off and features many sharp edges and pointy sections. Also, the linear interpolation (c) provides a very edgy phase space portrait, just as one wouold expect froma linear interpolation. Contrary to the that, from all presented phase space portraits, the one for the spline interpolation (d) is most similar to the original one, i.e. even the initial abbreviations caused by the time delay are almost perfectly reconstructed.

We further plotted all obtained results for 13 interpolation points as time series in Figure 4. The results show that the population mean (a) is far off the ground truth, and differs drastically at the high and low peaks, as it does not reach the actual data points. Both the genetic-algorithm-improved (b) and the best random interpolation of the initial population (c) capture most of the high and low peaks compared to the population mean. Further, when comparing the genetic-algorithm-improved and the population mean (d), one can see that the improved interpolation provides a smoother curve when depicted as a time series. In contrast, the population mean tends to produce sharp peaks.Finally, we compare the linear interpolation (e) and the spline interpolation (f) to the genetic algorithm improved interpolation. The linear interpolation here is far off, but the spline interpolation reproduces the Lorenz system almost perfectly, and thus outperforms the genetic-algorithm-improved interpolation.

4.2. Results for Non-Model Data Sets

This section tests our interpolation scheme on real-life data sets that possess only a limited number of sampled data points. But these are actually the focus of the proposed method, ie., to increase the fine-grainedness of short, sparsely sampled time series data, e.g., environmental or agricultural data sets. We must stress that our method is not restricted to equidistant time series: Due to the general form of the bridge construction (2), non-equidistant time series excerpts can be interpolated as well.

For this reason, we chose 5 data sets to demonstrate our method further, i.e., first we validated the interpolation with missing data points and then we present an actual interpolation and the improved phase space trajectories for each time series data. We consider a phase space trajectory to be improved if we achieve smoother trajectories, which exhibit fewer edgy points in a phase space representation. For this reason, we chose only data sets that can be depicted in reconstructed three-dimensional phase space, as an, e.g., four-dimensional phase space makes an intuitive understanding impossible.

The validation on these non-model data sets is performed such that every second data point of the original time series is deleted, then all the gaps are interpolated to reconstruct the missing data points. The results are shown for the average prediction of the population, the random interpolation with the lowest RMSE, a linear interpolation, a cubic spline interpolation and the improved interpolation using the presented genetic algorithm. This section features only the validation errors, the corresponding plots are collected in Appendix A to keep the main text focused.

 doi:10.20944/preprints202204.0167.v1

4.2.1. NYC Measles outbreaks data Set

This is a data set that we obtained from [18], where it is discussed and shown to feature an attractor structure in the embedded phase space. The corresponding original source is [19]. It depicts measles outbreaks in New York City (NYC) from 1928 to 1964, binned every two weeks, with a total of 432 data points.

The corresponding phase space embedding, with a time delay $\tau = 1$, was normalized such that the range of all data is between $[0, 1]$.

The results on how well the presented interpolation can reproduce missing data points of this data set are collected in Table 2 and depicted in Figure 11 (a). These results show that, though the genetic-algorithm-improved interpolation drastically outperforms the average random interpolation, the algorithm did not once outperform the best interpolation of the population. Still, starting with seven interpolation points, the genetic-algorithm-improved interpolation performs well and is very close to the best of 1000 randomly interpolated results, i.e., always below or around the best 1% of the population. Further, *PhaSpaSto*-interpolation does outperform the cubic spline interpolation starting with five interpolation points. We thus conclude that the presented interpolation technique captures the phase-space properties of this data set and effectively can be used to interpolate this time series data. Also, compared to the cubic and linear interpolation, the proposed method takes at least seven interpolation points to reach peak performance for this data set. All validation plots, are collected in Appendix A.2.

An interpolation of the original data set is depicted in Figure 5. Comparing the reconstructed phase space of the original data set, the population mean (c), and the presented interpolation technique (d); we see that the phase space portrait of the latter features a smoothed-out phase space trajectory compared to the original time series (b) and the population mean (c), which are both pointy and have many sharp edges. Further, considering the graph of the actual time series (a), we see that the presented interpolation technique increases the major peaks, thus making extreme events more prominent.

 doi:10.20944/preprints202204.0167.v1

Table 2: Errors for the interpolated data on the NYC measles data set depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations, as well as for the lowest error in the population and for the interpolation that was improved using the presented genetic algorithm. We highlighted the interpolation where the genetic-algorithm-based interpolation performed best. The corresponding plots for the best interpolation are shown in Appendix A.2. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation.

n_I	1	3	5	7	9	11	13	15
RMSE Population Mean	860.56140	860.56165	860.56098	860.56235	860.56124	860.56210	860.56145	860.56220
Lowest RMSE in population	594.27833	594.27832	594.27832	594.27832	594.27748	594.27831	594.27832	594.27833
RMSE linear interpolated	713.61079	713.61089	713.61089	713.61089	713.61089	713.61089	713.61089	713.61089
RMSE spline interpolated	607.03778	607.03778	607.03778	607.03778	607.03778	607.03778	607.03778	607.03778
RMSE gen. alg. improved	1138.28460	621.70136	602.03361	594.36367	594.33054	594.34891	594.34819	594.34132
Below Best %	75.3%	25.8%	13.4%	0.8%	0.8%	0.8%	0.8%	0.8%

n_I	17	19	21	23	25	27	29	31
RMSE Population Mean	860.56196	860.56132	860.56168	860.56090	**860.56153**	860.56287	860.56138	860.56192
Lowest RMSE in population	594.27901	594.27750	594.27934	594.27834	**594.28039**	594.27831	594.28069	594.27833
RMSE linear interpolated	713.61089	713.61089	713.61089	713.61089	**713.61089**	713.61089	713.61089	713.61089
RMSE spline interpolated	607.03778	607.03778	607.03778	607.03778	**607.03778**	607.03778	607.03778	607.03778
RMSE gen. alg. improved	594.33772	594.33837	594.33400	594.35508	**594.31806**	594.36050	594.42183	594.39145
Below Best %	0.8%	0.8%	0.8%	0.8%	**0.6%**	0.8%	1.1%	1.1%

 doi:10.20944/preprints202204.0167.v1

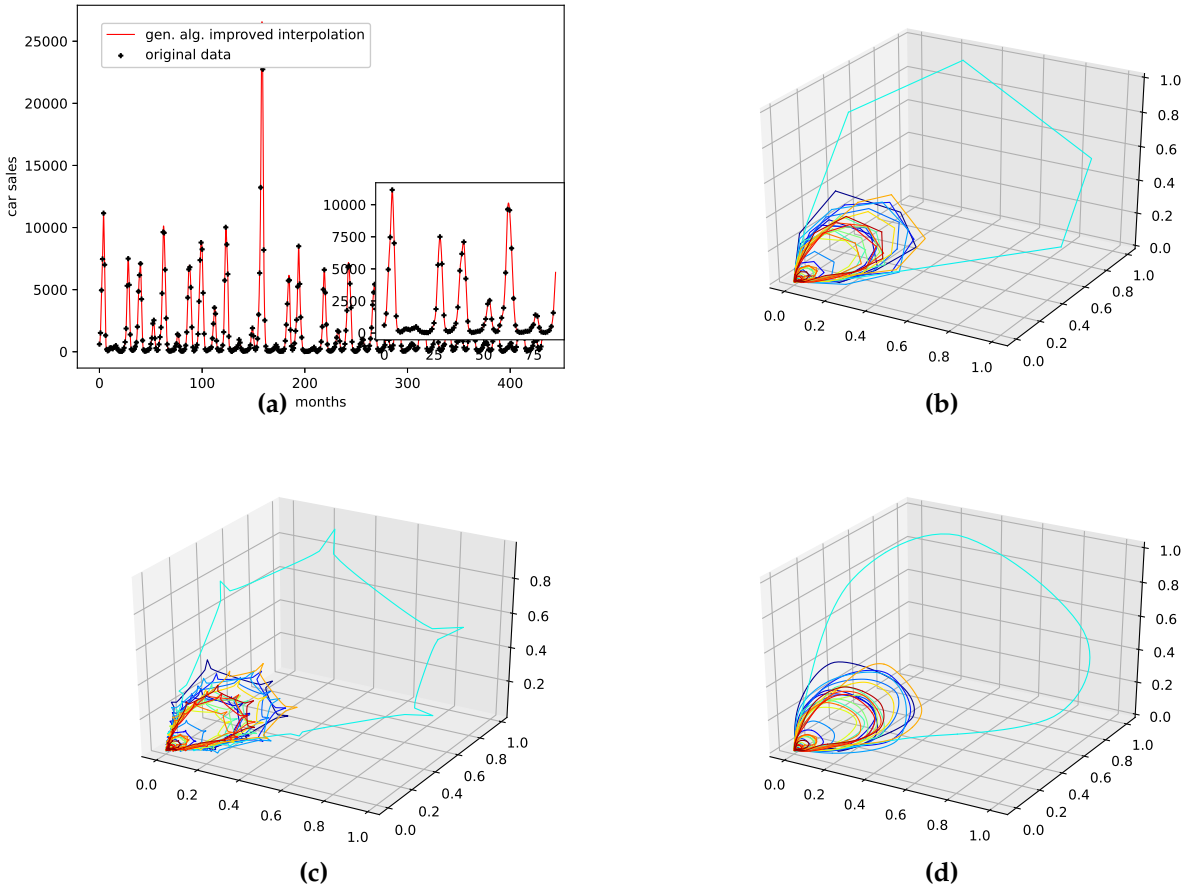


Figure 5. Interpolated data and reconstructed attractors for the NYC measles outbreaks data set.
(a): The original and interpolated time series data;
(b): Phase space reconstruction of the original data;
(c): Phase space reconstruction of the average population data;
(d): Phase space reconstruction of the genetic-algorithm-improved data.

 doi:10.20944/preprints202204.0167.v1

4.2.2. Car Sales in Quebec Data Set

This is a data set from the Time Series Data Library, [20]. It depicts monthly car sales in Quebec from January 1960 to December 1968, with a total of 108 data points.

The corresponding phase space embedding, with a time delay $\tau = 1$, was detrended by subtracting a linear fit from the data and normalized such that the range of all data is between $[0, 1]$.

The results on how well the presented interpolation can reproduce missing data points of this data set are collected in Table 3, and depicted in Figure 11 (b). The genetic-algorithm-improved interpolation drastically outperforms the average random interpolation. Further, the *PhaSpaSto*-interpolation always outperforms the cubic spline and linear interpolation. Still, the genetic-algorithm-improved interpolations did not always outperform the best random interpolations, but for one, three, and five interpolation points. Overall, the genetic-algorithm-improved interpolation performs well and is very close to the best of 1000 randomly interpolated results, i.e., for most cases below or around the best 1% of the population. Thus, we conclude that the presented interpolation technique effectively captures the phase-space properties of this data set and can be used to interpolate this time series data. All additional plots for the validation are collected in Appendix A.3, whereas one can find the reconstructed attractors for all interpolated validation sets and the corresponding time series plots.

An interpolation of the original data set is depicted in Figure 6. Here Figures 6 (c) and (d) present the population mean and the improved interpolation, respectively. When comparing them, one can see that the genetic algorithm improves the phase space portrait in terms of a smoothed-out phase space trajectory compared to the original time series (b) and the population mean (c), which are both pointy and have many sharp edges. When considering the actual time-series graph (a), the presented interpolation technique increases the major peaks, thus making extreme events more prominent. Further, it provides a rather smooth curve, i.e., no pointy edges, as depicted in the zoomed-in plot in (a).

 doi:10.20944/preprints202204.0167.v1

Table 3: Errors for the interpolated data on the car sales in Quebec data set depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations, the linear interpolation, the cubic spline interpolation, as well as for the lowest error in the population and for the interpolation that was improved using the presented genetic algorithm. We highlighted the interpolation where the genetic-algorithm-based interpolation performed best. The corresponding plots for the best interpolation are shown in Appendix A.3. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation.

n_I	1	3	5	7	9	11	13	15
RMSE Population Mean	**2030.11005**	2030.11166	2030.11148	2030.11230	2030.11030	2030.11138	2030.11106	2030.11110
Lowest RMSE in population	**1954.95010**	1954.95013	1954.95016	1954.95013	1954.95005	1954.95009	1954.95020	1954.95015
RMSE linear interpolated	**2017.79949**	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949
RMSE spline interpolated	**1971.23755**	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755
RMSE gen. alg. improved	**1907.40084**	1960.21475	1954.94790	1954.94792	1954.95375	1954.97452	1958.57232	1954.97468
Below Best %	**0.1%**	17.2%	0.1%	0.1%	0.6%	1.01%	14.6%	1.01%

n_I	17	19	21	23	25	27	29	31
RMSE Population Mean	2030.11260	2030.11057	2030.11226	2030.11047	2030.11078	2030.11105	2030.11171	2030.11013
Lowest RMSE in population	1954.95010	1954.95007	1954.95011	1954.95014	1954.95007	1954.95010	1954.95003	1954.95021
RMSE linear interpolated	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949	2017.79949
RMSE spline interpolated	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755	1971.23755
RMSE gen. alg. improved	1954.97730	1954.99153	1955.00052	1954.99273	1955.02450	1955.02418	1955.01367	1954.98108
Below Best %	1.3%	1.4%	1.4%	1.4%	1.6%	1.6%	1.4%	1.4%

 doi:10.20944/preprints202204.0167.v1

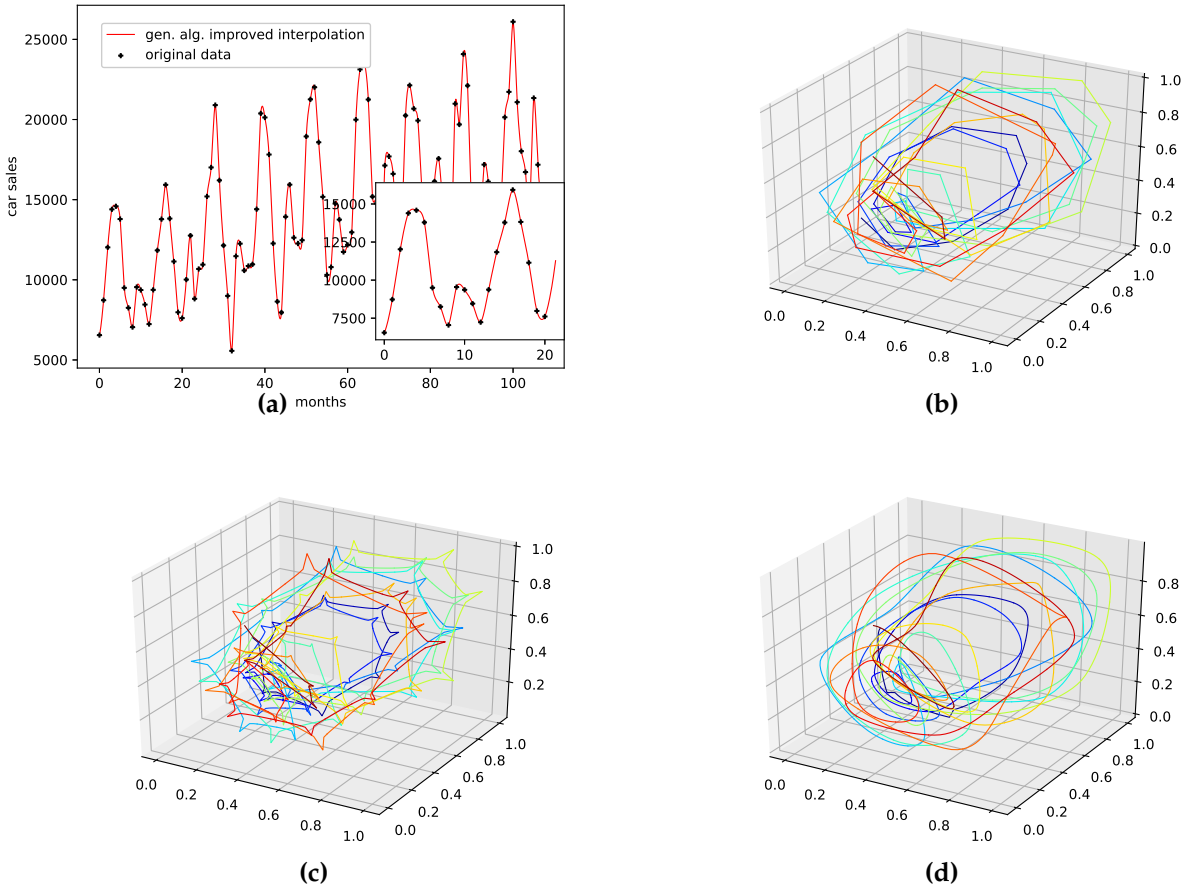


Figure 6. Interpolated data and reconstructed attractors for the car sales in Quebec data set.
(a): The original and interpolated time series data;
(b): Phase space reconstruction of the original data;
(c): Phase space reconstruction of the average population data;
(d): Phase space reconstruction of the genetic-algorithm-improved data.

 doi:10.20944/preprints202204.0167.v1

4.2.3. Perrin Freres Champagne Sales Data Set

This is a data set from the Time Series Data Library, [20]. It depicts Perrin Freres Champagne sales from January 1964 to September 1972, with a total of 105 data points.

The corresponding phase space embedding, with a time delay $\tau = 1$, was normalized such that the range of all data is between $[0, 1]$.

The results on how well the presented interpolation can reproduce missing data points of this data set are collected in Table 4 and Figure 11 (c).

Though the genetic-algorithm-improved interpolation drastically outperforms the average random interpolation, the algorithm did not once outperform the best interpolation of the population. Still, starting with five interpolation points, the genetic-algorithm-improved interpolation performs well and is very close to the best of 1000 randomly interpolated results, i.e., consistently below or around the best 1% of the population. Overall the cubic spline interpolation performed best on this data set. The linear interpolation, though outperforming the population mean, is still far off. We thus conclude that the presented interpolation technique does capture the phase-space properties of this data set from a given population and can be used to interpolate this time series data, but the cubic spline interpolation is the better choice.

An interpolation of the original data set is depicted in Figure 5. We again show the population mean (c) and the improved interpolation (d). The presented interpolation technique improves the phase space portrait in terms of a smoothed-out phase space trajectory (d) compared to the original time series (b), and the population mean (c), which are both pointy and have many sharp edges. Further, considering the graph of the actual time series (a), the presented interpolation technique increases the major peaks, thus making extreme events more prominent and provides a thoroughly smooth curve, as depicted in the zoom-in window in (a).

 doi:10.20944/preprints202204.0167.v1

Table 4: Errors for the interpolated data on the Perrin Freres champagne sales data set depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations, as well as for the lowest error in the population, the linear interpolation, the cubic spline interpolation and for the interpolation that was improved using the presented genetic algorithm. We highlighted the interpolation where the genetic-algorithm-based interpolation performed best. The corresponding plots for the best interpolation are shown in Appendix A.4. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation.

n_I	1	3	5	7	9	11	13	15
RMSE Population Mean	2320.03501	2320.03532	2320.03403	**2320.03366**	2320.03524	2320.03333	2320.03195	2320.03301
Lowest RMSE in population	2144.04985	2144.05002	2144.04987	**2144.05007**	2144.04991	2144.04986	2144.04981	2144.04995
RMSE linear interpolated	2264.24606	2264.24606	2264.24606	**2264.24606**	2264.24606	2264.24606	2264.24606	2264.24606
RMSE spline interpolated	2099.58713	2099.58713	2099.58713	**2099.58713**	2099.58713	2099.58713	2099.58713	2099.58713
RMSE gen. alg. improved	2540.17542	2153.68153	2144.43642	**2144.07624**	2144.16502	2144.15580	2144.14292	2144.16357
Below Best %	82.5%	6.9%	1.01%	**0.3%**	0.6%	0.6%	0.6%	0.6%

n_I	17	19	21	23	25	27	29	31
RMSE Population Mean	2320.03312	2320.03073	2320.03244	2320.03250	2320.03279	2320.03565	2320.03406	2320.03476
Lowest RMSE in population	2144.04987	2144.04986	2144.04967	2144.04982	2144.04997	2144.04985	2144.04976	2144.04999
RMSE linear interpolated	2264.24606	2264.24606	2264.24606	2264.24606	2264.24606	2264.24606	2264.24606	2264.24606
RMSE spline interpolated	2099.58713	2099.58713	2099.58713	2099.58713	2099.58713	2099.58713	2099.58713	2099.58713
RMSE gen. alg. improved	2144.09078	2144.13364	2144.17573	2144.13637	2144.16380	2144.10973	2144.10709	2144.13681
Below Best %	0.5%	0.6%	0.6%	0.6%	0.6%	0.6%	0.6%	0.6%

 doi:10.20944/preprints202204.0167.v1

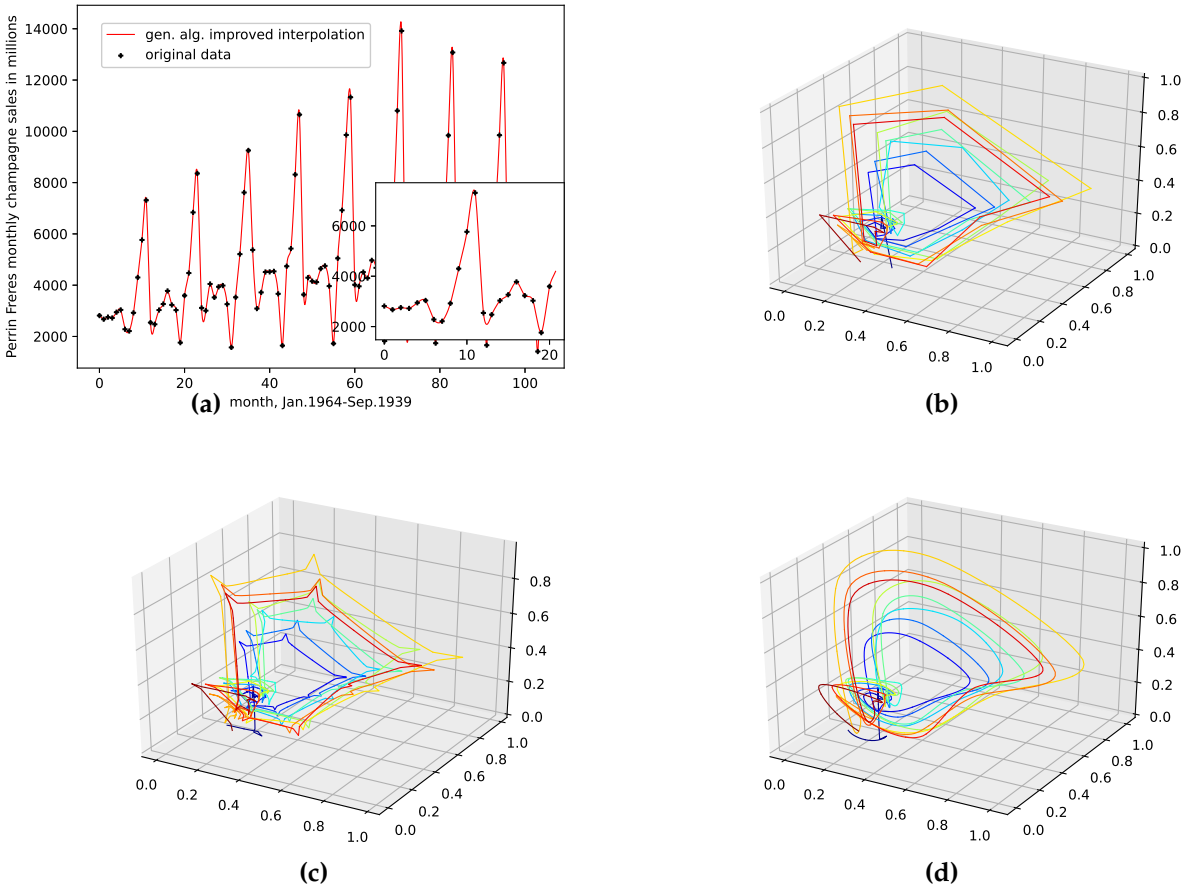


Figure 7. Interpolated data and reconstructed attractors for the Perrin Freres Champagne sales data set.
(a): The original and interpolated time series data;
(b): Phase space reconstruction of the original data;
(c): Phase space reconstruction of the average population data;
(d): Phase space reconstruction of the genetic-algorithm-improved data.

 doi:10.20944/preprints202204.0167.v1

4.2.4. Monthly Airline Passengers Data Set

This is a data set from the Time Series Data Library, [20]. It depicts monthly international airline passengers from January 1949 to December 1960, with a total of 144 data points, given in units of 1000.

The corresponding phase space embedding, with a time delay $\tau = 1$, was detrended by subtracting a linear fit from the data and normalized such that the range of all data is between $[0, 1]$.

The results on how well the presented interpolation can reproduce missing data points of this data set are collected in Table 5 and depicted in Figure 11 (d). The results show that, though the genetic-algorithm-improved interpolation drastically outperforms the average random interpolation, the algorithm did not once outperform the best interpolation of the population. Still, starting with three interpolation points, the algorithm did outperform both the linear and the cubic spline interpolation. What's curious, though, is that, for this data set, of all the non-model data sets, the linear interpolation outperforms the cubic spline interpolation.

The genetic-algorithm-improved interpolation does not perform that well for this data set compared to a random interpolation of the time series. The improved interpolation is only around the best $\approx 40\%$ of the initial population for this data set. We thus conclude that the presented interpolation technique does not capture the phase-space properties of this data set very well. Still, the genetic algorithm does improve the initial population such that the population mean, the linear interpolation, and the cubic spline interpolation are outperformed, starting with three interpolation points. All-time series and reconstructed attractor plots for this data set can be found in Appendix A.5.

An actual interpolation of the original data set is depicted in Figure 8. We again show the population mean (c) and the improved interpolation (d). The presented *PhaSpaSto*-interpolation (d) improves the phase space portrait in terms of a smoothed-out phase space trajectory, compared to the original time series (b) and the population mean (c), which are both pointy and have many sharp edges. Further, considering the actual time series (a) graph, we see that the presented interpolation technique slightly increases the major peaks. Also, compared to the other non-model data sets, the improved interpolation does provide a relatively smooth curve, but it looks way sharper than for, e.g., the car sales in Quebec (See Figure 6 (a))data set.

 doi:10.20944/preprints202204.0167.v1

Table 5: Errors for the interpolated data on the monthly airline passengers data set data set depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations, the linear interpolation, the cubic spline interpolation, as well as for the lowest error in the population and for the interpolation that was improved using the presented genetic algorithm. We highlighted the interpolation where the genetic-algorithm-based interpolation performed best. The corresponding plots for the best interpolation are shown in Appendix A.5. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation.

n_I	1	3	5	7	9	11	13	15
RMSE Population Mean	19.93996	**19.93999**	19.93841	19.94072	19.93976	19.93873	19.94070	19.93889
Lowest RMSE in population	16.55624	**16.55779**	16.55732	16.55753	16.55558	16.55836	16.55719	16.55776
RMSE linear interpolated	17.39496	**17.39496**	17.39496	17.39496	17.39496	17.39496	17.39496	17.39496
RMSE spline interpolated	18.33872	**18.33872**	18.33872	18.33872	18.33872	18.33872	18.33872	18.33872
RMSE gen. alg. improved	18.65257	**16.81653**	16.84539	17.02728	16.84536	16.84545	16.84540	16.84539
Below Best %	59.4%	**35.6%**	38.0%	42.20%	38.1%	38.0%	38.1%	38.0%

n_I	17	19	21	23	25	27	29	31
RMSE Population Mean	19.94029	19.94030	19.93985	19.93939	19.93659	19.94172	19.94023	19.93909
Lowest RMSE in population	16.55752	16.55730	16.55810	16.55715	16.55733	16.55603	16.55789	16.55741
RMSE linear interpolated	17.39496	17.39496	17.39496	17.39496	17.39496	17.39496	17.39496	17.39496
RMSE spline interpolated	18.33872	18.33872	18.33872	18.33872	18.33872	18.33872	18.33872	18.33872
RMSE gen. alg. improved	16.84546	16.84545	16.84548	16.84540	16.84535	16.84544	16.84544	16.84546
Below Best %	38.1%	38.1%	38.0%	38.1%	38.1%	38.1%	38.2%	38.1%

 doi:10.20944/preprints202204.0167.v1

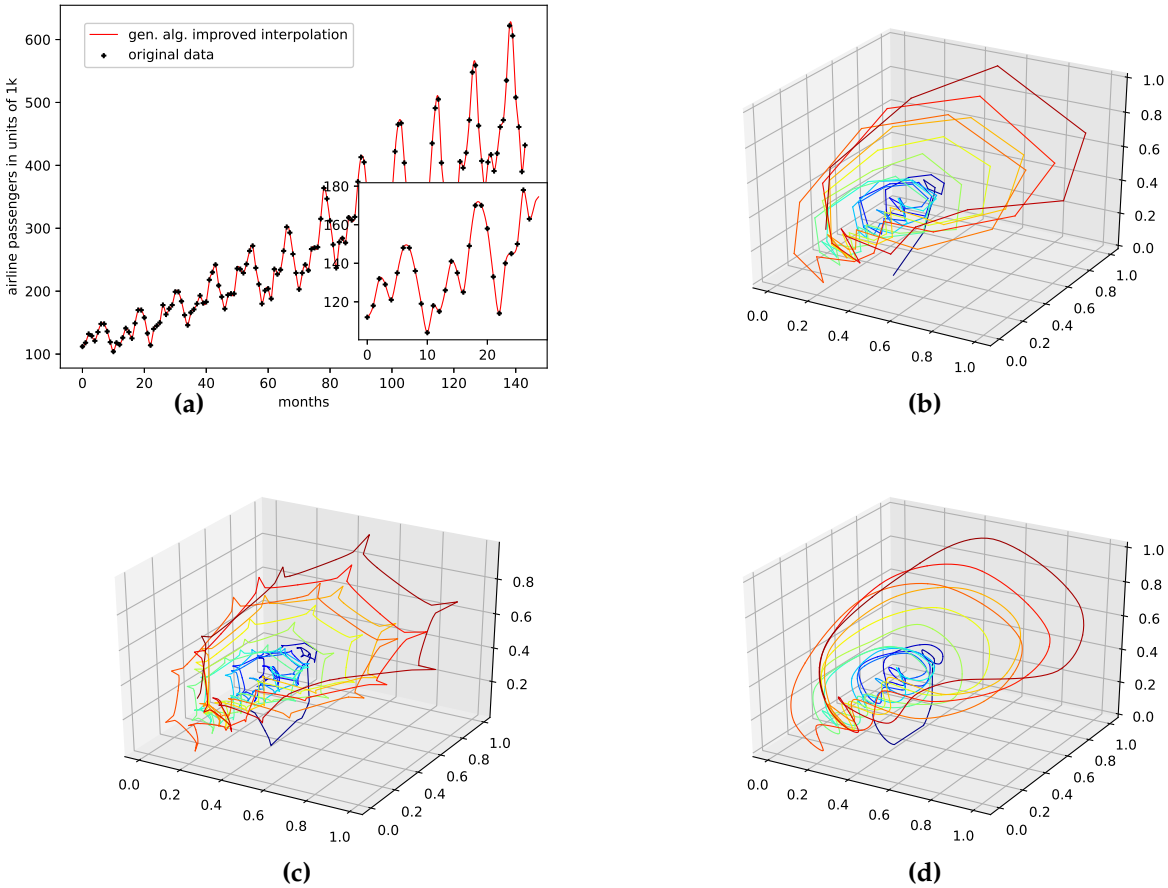


Figure 8. Interpolated data and reconstructed attractors for the monthly international airline passengers data set.
(a): The original and interpolated time series data;
(b): Phase space reconstruction of the original data;
(c): Phase space reconstruction of the average population data;
(d): Phase space reconstruction of the genetic-algorithm-improved data.

 doi:10.20944/preprints202204.0167.v1

4.2.5. Monthly Mean Temperature in Nottingham Castle

This is a data set from the Time Series Data Library, [20]. It depicts the mean monthly temperature in Nottingham castle from January 1920 to December 1939, given in degrees Fahrenheit, with a total of 240 data points.

The corresponding phase space embedding, with a time delay $\tau = 3$, was normalized such that the range of all data is between $[0, 1]$, with a time delay $\tau = 1$.

The corresponding phase space embedding, with a time delay $\tau = 1$, was detrended by subtracting a linear fit from the data and normalized such that the range of all data is between $[0, 1]$.

The results on how well the presented interpolation can reproduce missing data points of this data set are collected in Table 6 and depicted in Figure 11. The results show that, though the genetic-algorithm-improved interpolation drastically outperforms the average random interpolation, the algorithm did not once outperform the best interpolation of the population, although outperforming the linear and the cubic spline interpolation. The genetic-algorithm-improved interpolation does not perform that well for this data set compared to a random interpolation of the time series, as the improved interpolation is only around the best $\approx 34\%$ for this data set. We thus conclude that the presented interpolation technique does not capture the phase-space properties of this data set very well. The corresponding time-series and reconstructed phase space plots are collected in Appendix A.6.

An interpolation of the original data set is depicted in Figure 9. We again show the population mean (c) and the improved interpolation (d). The presented interpolation technique improves the phase space portrait (d) in terms of a smoothed-out phase space trajectory compared to the original time series (b) and the population mean (c), which are both pointy and have many sharp edges. Also, given the time-series depiction of the *PhaSpaSto*-interpolation (Figure 9 (a)), we see the same behavior as for all the other data sets; the major peaks are increased.

 doi:10.20944/preprints202204.0167.v1

Table 6: Errors for the interpolated data on the monthly mean temperature in Nottingham castle data set depending on the number of interpolation points. The errors are shown for the mean interpolation of all populations, as well as for the lowest error in the population and for the interpolation that was improved using the presented genetic algorithm. We highlighted the interpolation where the genetic-algorithm-based interpolation performed best. The corresponding plots for the best interpolation are shown in Appendix A.6. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation.

n_I	1	3	5	7	9	11	13	15
RMSE Population Mean	**3.09115**	3.09170	3.09167	3.09055	3.09088	3.09055	3.09166	3.09165
Lowest RMSE in population	**2.47879**	2.47858	2.47910	2.47890	2.47886	2.47901	2.47900	2.47875
RMSE linear interpolated	**2.61279**	2.61279	2.61279	2.61279	2.61279	2.61279	2.61279	2.61279
RMSE spline interpolated	**2.59028**	2.59028	2.59028	2.59028	2.59028	2.59028	2.59028	2.59028
RMSE gen. alg. improved	**2.48413**	2.50179	2.50279	2.50406	2.50420	2.50521	2.50512	2.505089
Below Best %	**12.6%**	31.3%	32.4%	33.5%	33.8%	34.4%	34.4%	34.1%

n_I	17	19	21	23	25	27	29	31
RMSE Population Mean	3.09095	3.09177	3.09115	3.09122	3.09146	3.09143	3.09179	3.09023
Lowest RMSE in population	2.47887	2.47920	2.47925	2.47899	2.47867	2.47941	2.47885	2.47892
RMSE linear interpolated	2.61279	2.61279	2.61279	2.61279	2.61279	2.61279	2.61279	2.61279
RMSE spline interpolated	2.59028	2.59028	2.59028	2.59028	2.59028	2.59028	2.59028	2.59028
RMSE gen. alg. improved	2.50494	2.50541	2.50529	2.50552	2.50505	2.50547	2.50550	2.50533
Below Best %	33.9%	34.6%	34.4%	35%	34.6%	34.6%	34.7%	34.7%

 doi:10.20944/preprints202204.0167.v1

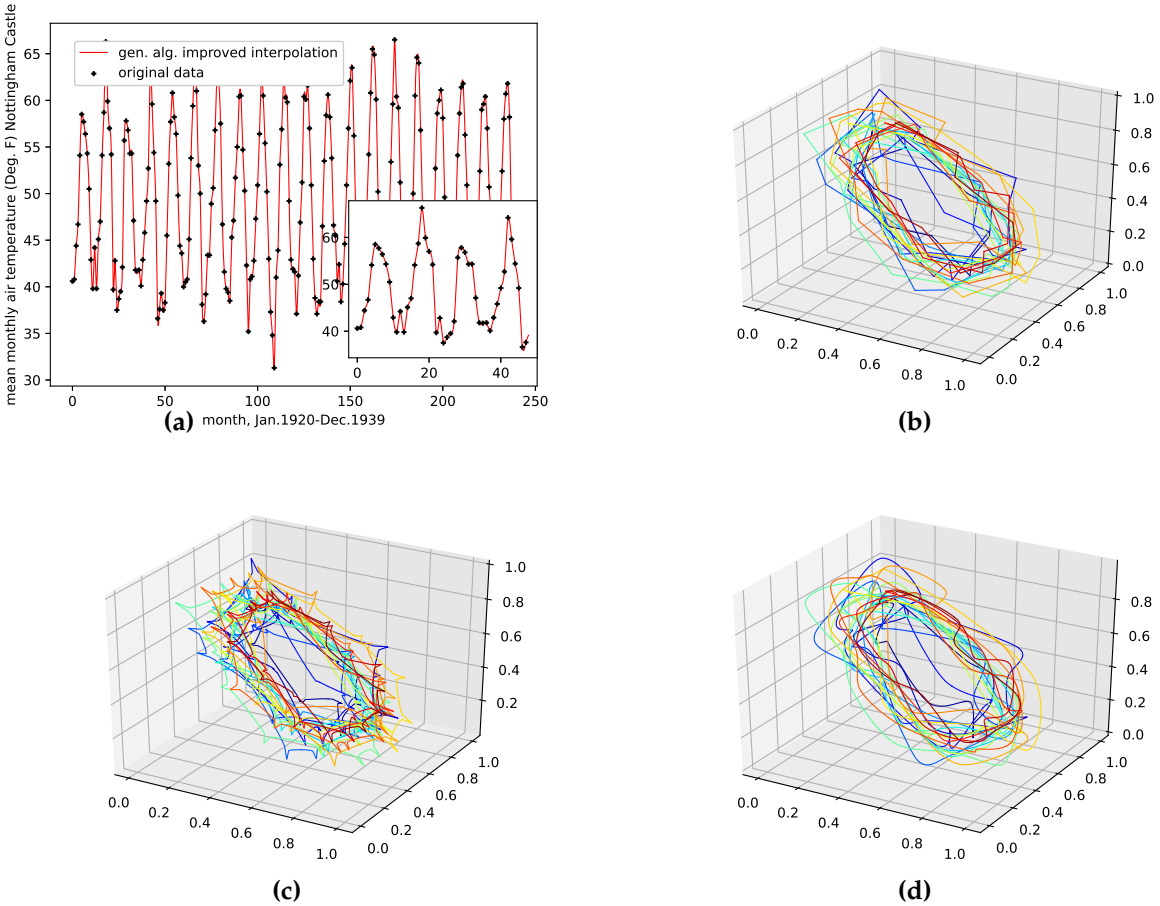


Figure 9. Interpolated data and reconstructed attractors for the monthly mean temperature in Nottingham castle data set.
(a): The original and interpolated time series data;
(b): Phase space reconstruction of the original data;
(c): Phase space reconstruction of the average population data;
(d): Phase space reconstruction of the genetic-algorithm-improved data.

 doi:10.20944/preprints202204.0167.v1

4.3. Summary

We briefly summarize this research and point out the main findings:

- We presented a genetic algorithm to improve a stochastic interpolation, i.e., multi-point fractional Brownian bridges, in order to enhance the reconstructed phase space of any given time series data. For simplicity, we named this method *PhaSpaSto*-interpolation.
- We presented a novel approach to measure the quality of a phase space reconstruction according to Taken's theorem. Here we used an idea from image processing, i.e., to identify blurry images via the variance of second derivatives. These second derivatives are calculated along the reconstructed phase space curve for any given reconstructed phase space. The variance of these second derivatives is used to measure the quality of our phase space reconstruction. Given two interpolated phase space curves of the same time series, the one with the lower variance of second derivatives along the curve is found to be the better phase space reconstruction.
- We showed that the developed technique performed well in the case of a model data set, i.e., one variable of the Lorenz system. Here we deleted data points from the original time series data and were able to outperform, in some cases, any best random interpolations of this time series data. Also, the presented method outperformed a linear interpolation when it comes to finding the missing data points. Still, the proposed method did not outperform the presented cubic spline interpolation on this task. This was done to validate our method and to show its applicability. Further, the presented reconstructed phase spaces plots show that the interpolated phase space reconstruction is very similar to the original reconstructed phase space. The results for the Lorenz system are collected in Section 4.1.
- We validated the presented method using five sparsely sampled non-model data sets. The validation was done such that we deleted every second data point from the original time series and reconstructed these missing data points using the developed technique. For three out of five data sets, the developed method effectively can identify the interpolations or parts of it with low errors, i.e., the result is around the best 1% of the population in terms of the RMSE for the reconstructed data points. Also, *PhaSpaSto*-interpolation outperformed the spline interpolation for four of five non-model data sets and the linear interpolation on all non-model data sets. Also, the best random interpolation outperformed the cubic spline interpolation on four of five non-model data sets. For two data sets, the *PhaSpaSto*-interpolation does not perform very well as it is only around the best $30-40\%$ of all RMSEs of the population. The interpolation performed well in case of the measles cases in NYC data set (Section 4.2.1), the car sales in Quebec data set (Section 4.2.2) and the Perrin Freres champagne sales data set (Section 4.2.3). The two cases where the presented method did not perform well are the monthly international airline passengers data set (Section 4.2.4) and the monthly mean temperature in Nottingham castle data set (Section 4.2.5).
- We also used the five non-model data sets to show the applicability of the developed technique as an actual interpolation technique, i.e., no deleted data points. The plots of the reconstructed phase spaces show that it softens the edges and provides are thoroughly smoother and cleaner reconstructed phase space trajectory. Therefore the authors conclude that this technique applies to arbitrary univariate data sets. All of these plots are collected in Section 4.2. We further recommend it when dealing with sparsely sampled time series, e.g., attractor reconstructions of real-life time series, e.g., to improve machine learning time-series predictions.

5. Conclusion

This article presents a novel approach to interpolate univariate time series data. For simplicity, we named this method *PhaSpaSto*-interpolation. The concept is first to generate a population of, e.g., 1000, different stochastically interpolated time series data.

 doi:10.20944/preprints202204.0167.v1

This is done using multi-point Brownian bridges, each assigned with a random Hurst exponent. Then, as a second step, a genetic algorithm generates one time series out of the population with a low variance of second-order derivatives along the corresponding reconstructed phase space trajectory. I.e., we want this curve to be as smooth as possible. Using the variance of second-order derivatives is adapted from image processing, where the variance of second-order derivatives is used to differentiate between blurry and sharp images. To illustrate this idea, we plotted the randomly generated stochastic interpolations and the improved interpolation in Figure 10 for the monthly international airline passengers data set. We also tested the discussed approach with different loss functions that, in the end, did not work. These failed attempts are collected in Appendix C.

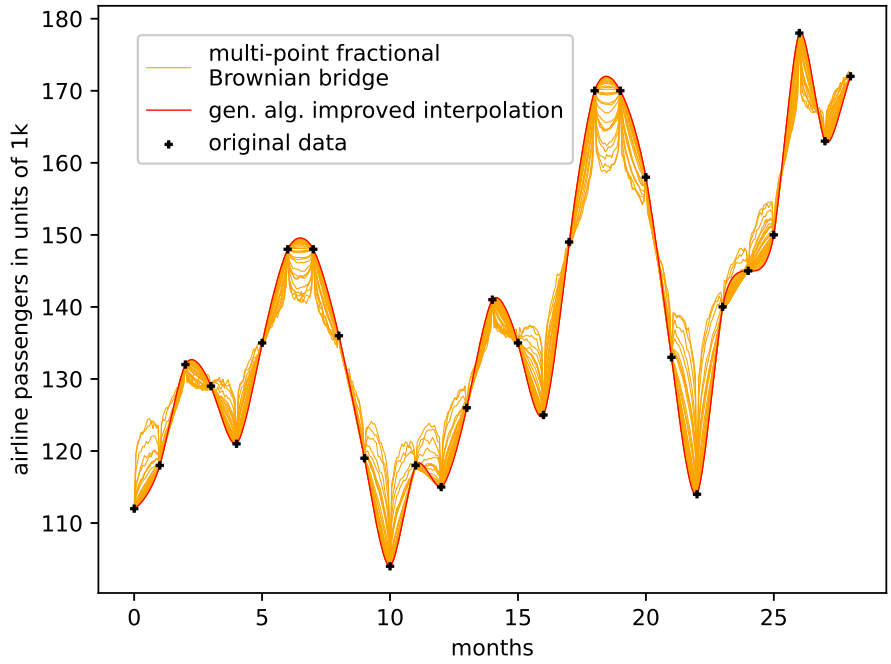


Figure 10. Random and gen. alg. improved interpolations for the monthly international airline passengers data set, see Section 4.2.4. The orange lines are the randomly generated stochastic interpolations, i.e., multi-point fractional Brownian bridges with different Hurst exponents.

We then applied the presented interpolation technique to the Lorenz system, or to be specific, to one of the variables of the Lorenz system, since we're dealing with univariate data only. We then deleted data points from this time series and interpolated the missing data points with the presented interpolation technique. We also tested the proposed approach against a linear and a cubic spline interpolation. The results show that the presented attractor-based stochastic interpolation can reproduce the Lorenz system, i.e., the genetic algorithm can find the best parts of the initial population to reconstruct the Lorenz system. Still, the spline interpolation outperformed the *PhaSpaSto*-interpolation for the Lorenz system. Finally, we applied the presented approach to various real-life and/or benchmark data sets. There's no fine-grained model data available for data sets like these. We cannot verify the interpolation as we did with the Lorenz system. Instead, we deleted every second data point of these data sets and reconstructed them using the developed method, i.e., generated interpolations using a range of interpolation points, picking the missing data points, and verifying them against the ground truth. *PhaSpaSto*-interpolation performed well on three of five data sets, as the genetic algorithm can, in

 doi:10.20944/preprints202204.0167.v1

fact, identify/build interpolations with low errors for the missing data points. Further, *PhaSpaSto*-interpolation outperformed the spline-interpolation on four of five data sets. Thus, we conclude that the presented method can also be applied to non-model data sets. Lastly, we show actual interpolations on these non-model data sets, i.e., no deleted data points. Given that the reader is familiar with how strange attractors of chaotic systems look, it should be clear from the presented reconstructed phase space portraits that our approach can interpolate real-life data as one would expect a phase space embedding of a strange attractor to look like, see Section 4.2. Future research will also be devoted to generalizations of the bridge process (2) to random processes which exhibit non-Gaussian features [21].

We expect the presented research to be useful for predicting and analyzing sparsely sampled time series data, e.g., in agriculture or other fields where fine-grained measurements are expensive. We further expect the presented research to be utilized for improving machine and deep learning approaches with insufficient data. We expect that an enhanced phase space structure improves forecasts' accuracy.

Future improvements and applications of this technique include the expansion to multi-variate data sets and using the presented loss function, i.e., the variance of second derivatives along phase space trajectories, to find better phase space embeddings. The interested reader is therefore referred to appendix B where we present the loss-surface of the Lorenz system with a varying time delay τ and a varying embedding dimension d_E.

And, as previously mentioned, we want to test to what degree improved phase-space embeddings can be beneficial for machine and/or deep learning approaches for learning and predicting time series data.

Lastly, the presented interpolation technique code will be available on GitHub from the corresponding author in the future.

Acknowledgments

The authors acknowledge the funding of the project "DiLaAg – Digitalization and Innovation Laboratory in Agricultural Sciences", by the private foundation "Forum Morgen", the Federal State of Lower Austria, by the FFG; Project AI4Cropr, No. 877158 and by TU Wien Bibliothek for financial support through its Open Access Funding Program. J.F. acknowledges funding from the Humboldt Foundation within a Feodor-Lynen fellowship.

Appendices

A. Additional Plots

This section provides additional plots for all data sets discussed in Section 4.2. As such we plotted the evolution of errors for the validation depending on the varying number of interpolation points, i.e. the errors from Tables 2, 3, 4, 5 and 6. Further, we added each time series and the corresponding best validation interpolation, and finally, the corresponding phase space plots.

 doi:10.20944/preprints202204.0167.v1

A.1. Evolution of Errors

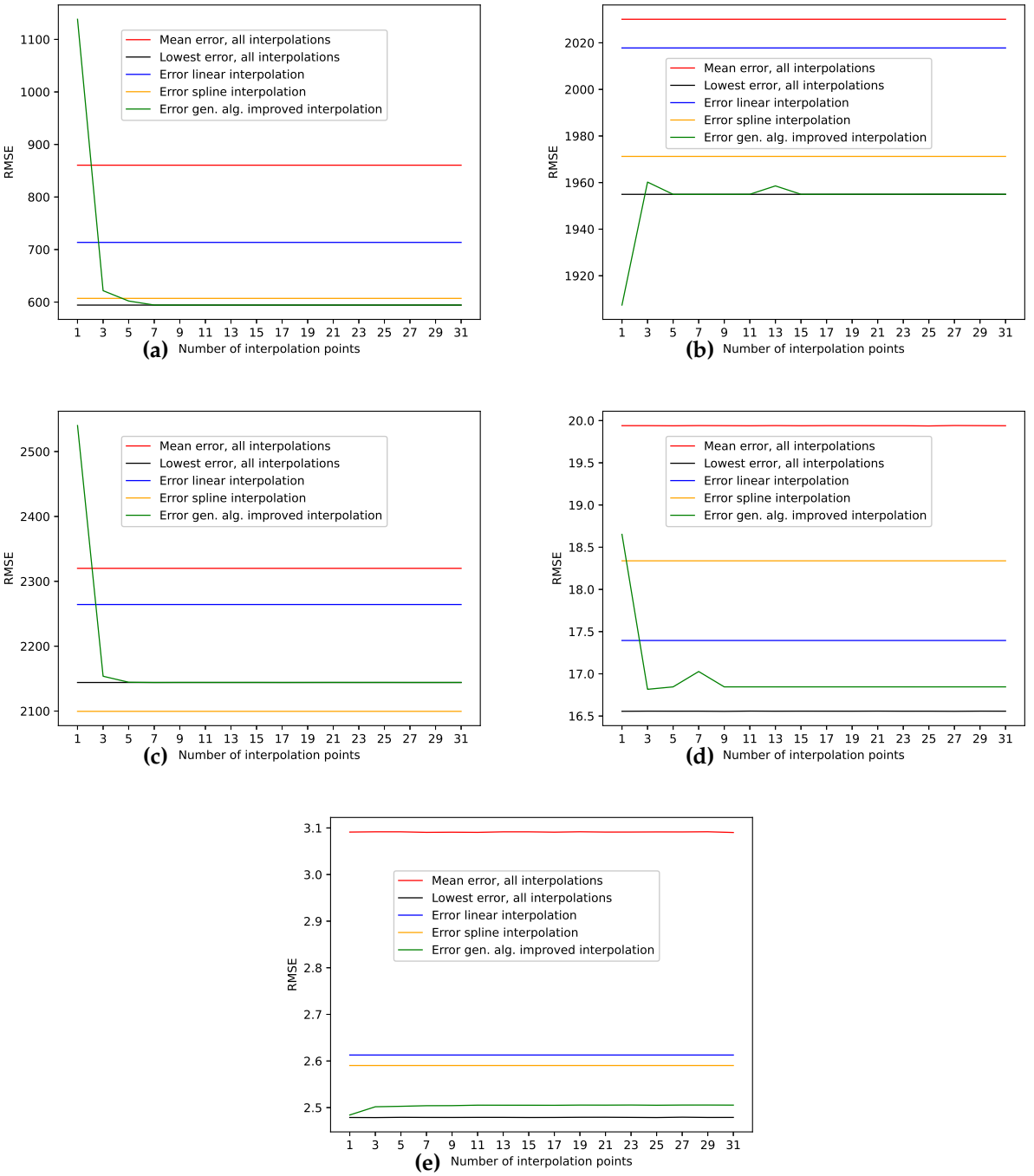


Figure 11. Evolution of errors depending on the number of interpolation points for the non-model data validation.
(a): Measles cases in NYC data set, results from Table 2;
(b): Car Sales in Quebec data set, results from Table 3;
(c): Perrin Freres champagne sales data set, results from Table 4;
(d): Monthly international airline passengers data set, results from Table 5;
(e): Monthly mean temperature in Nottingham castle data set, results from Table 6;

 doi:10.20944/preprints202204.0167.v1

525 *A.2. NYC Measles Outbreaks Data Set*

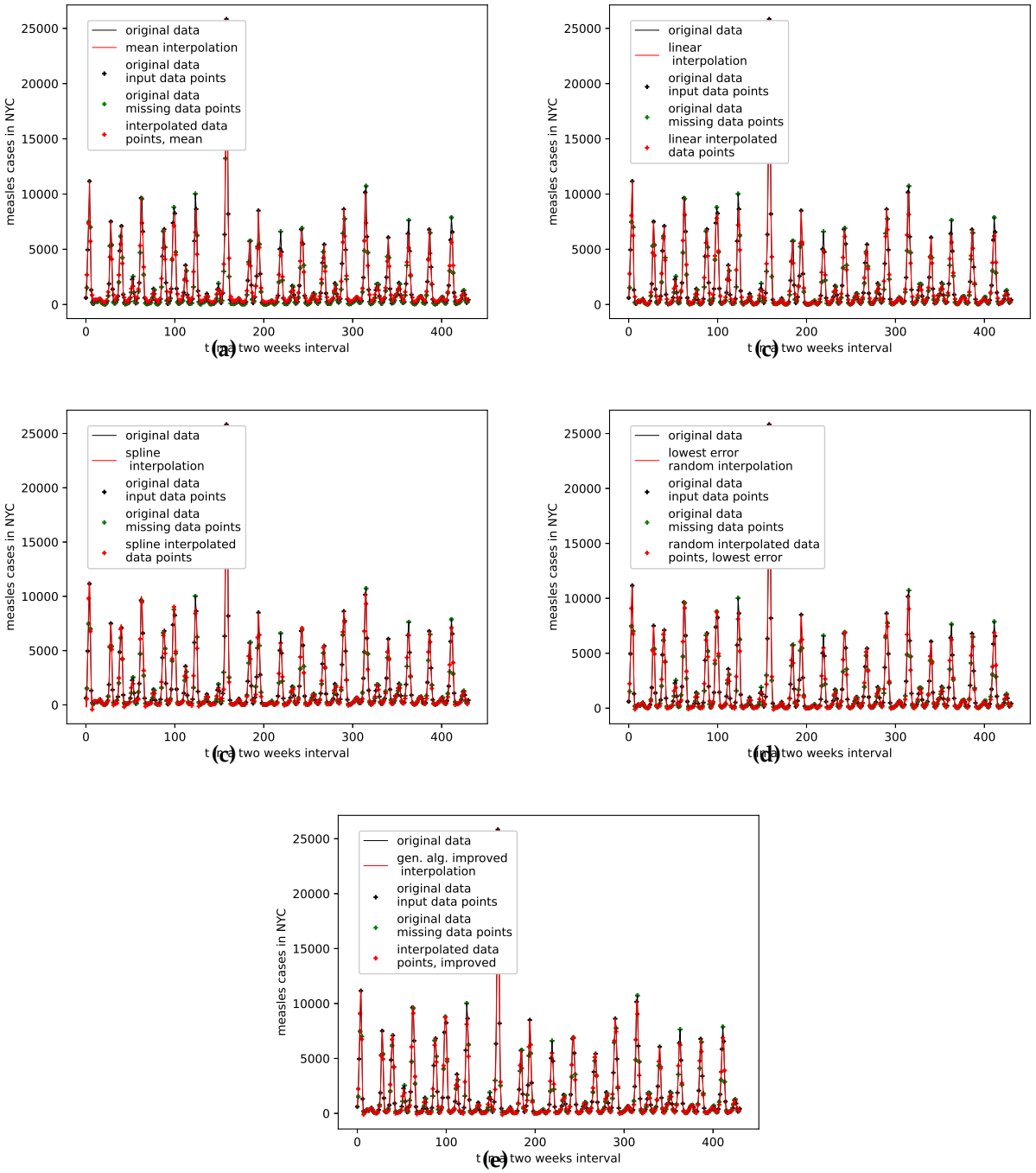


Figure 12. Interpolated validation data (25 interpolation points) for the measles cases in NYC data set.
(a): Average population validation;
(b): Validation, linear interpolation;
(c): Validation, spline interpolation;
(d): Validation, best random interpolation;
(e): Validation, gen. alg. improved interpolation;

 doi:10.20944/preprints202204.0167.v1

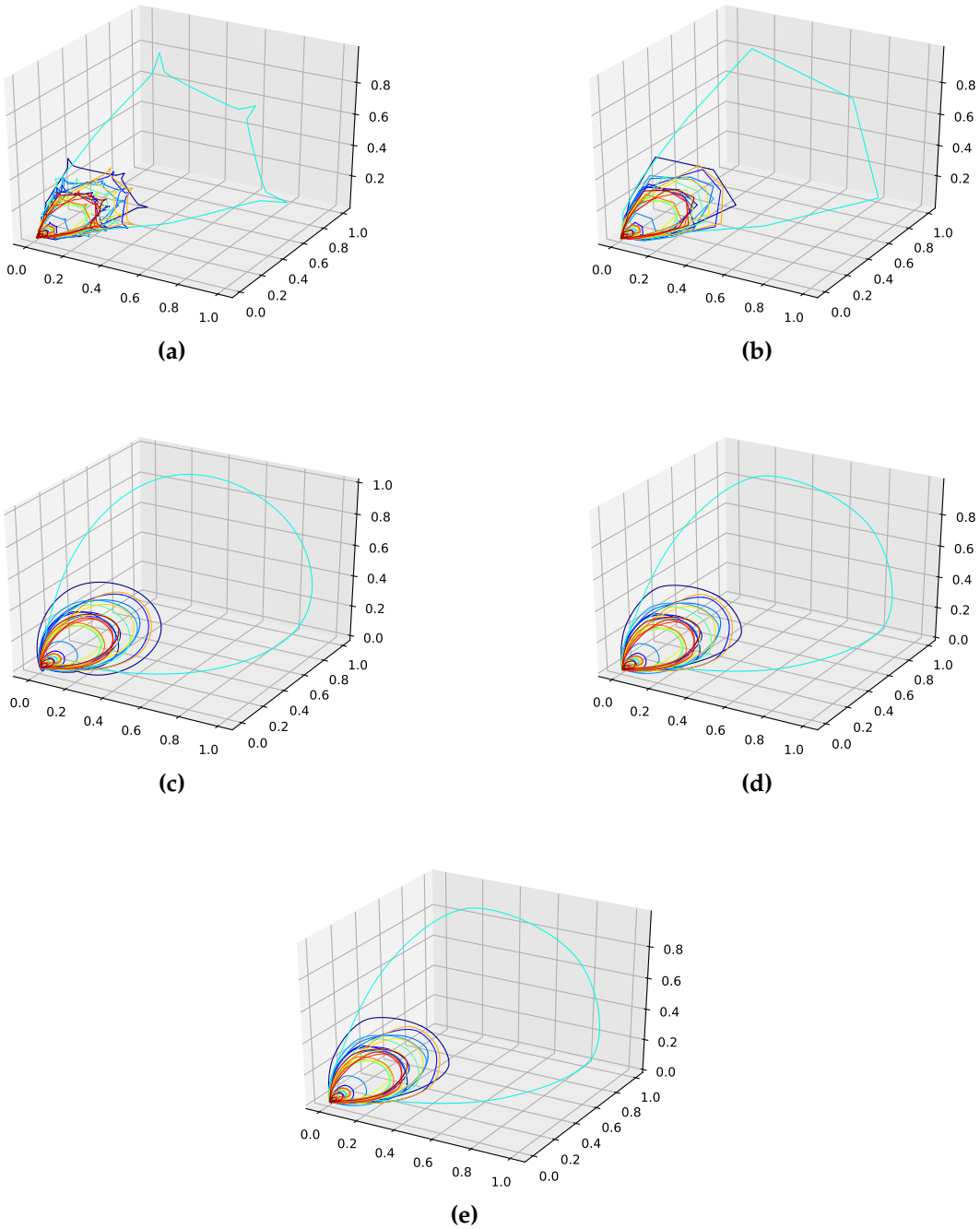


Figure 13. Reconstructed validation attractors (25 interpolation points) for the measles cases in NYC data set.
(a): Reconstructed attractor, average population validation interpolation;
(b): Reconstructed attractor, linear interpolation;
(c): Reconstructed attractor, spline interpolation;
(d): Reconstructed attractor, best random validation interpolation;
(e): Reconstructed attractor, gen. alg. improved validation interpolation;

 doi:10.20944/preprints202204.0167.v1

526 *A.3. Car Sales in Quebec Data Set*

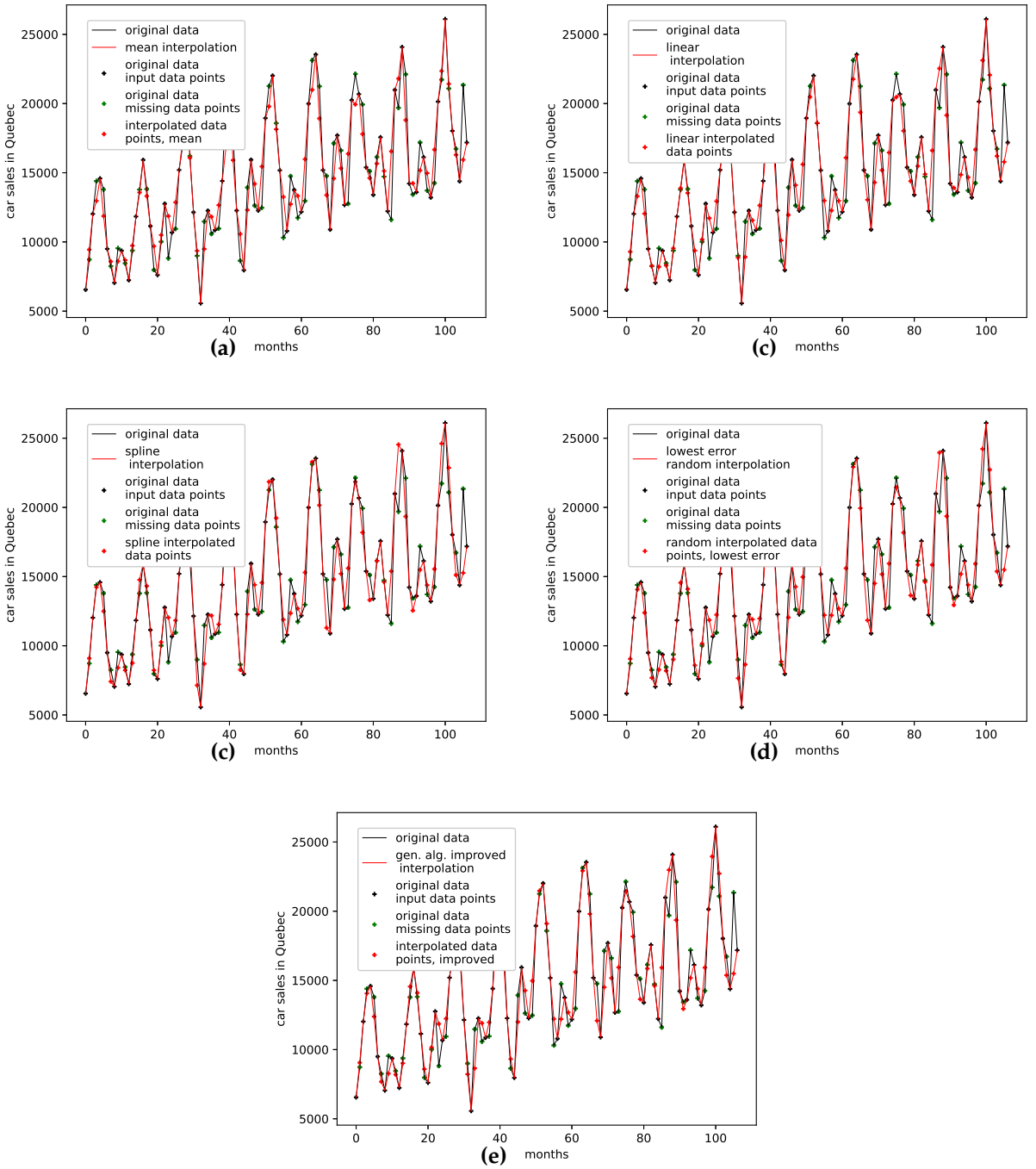


Figure 14. Interpolated validation data (one interpolation point) for the car sales in Quebec data set.
(a): Average population validation;
(b): Validation, linear interpolation;
(c): Validation, spline interpolation;
(d): Validation, best random interpolation;
(e): Validation, gen. alg. improved interpolation;

 doi:10.20944/preprints202204.0167.v1

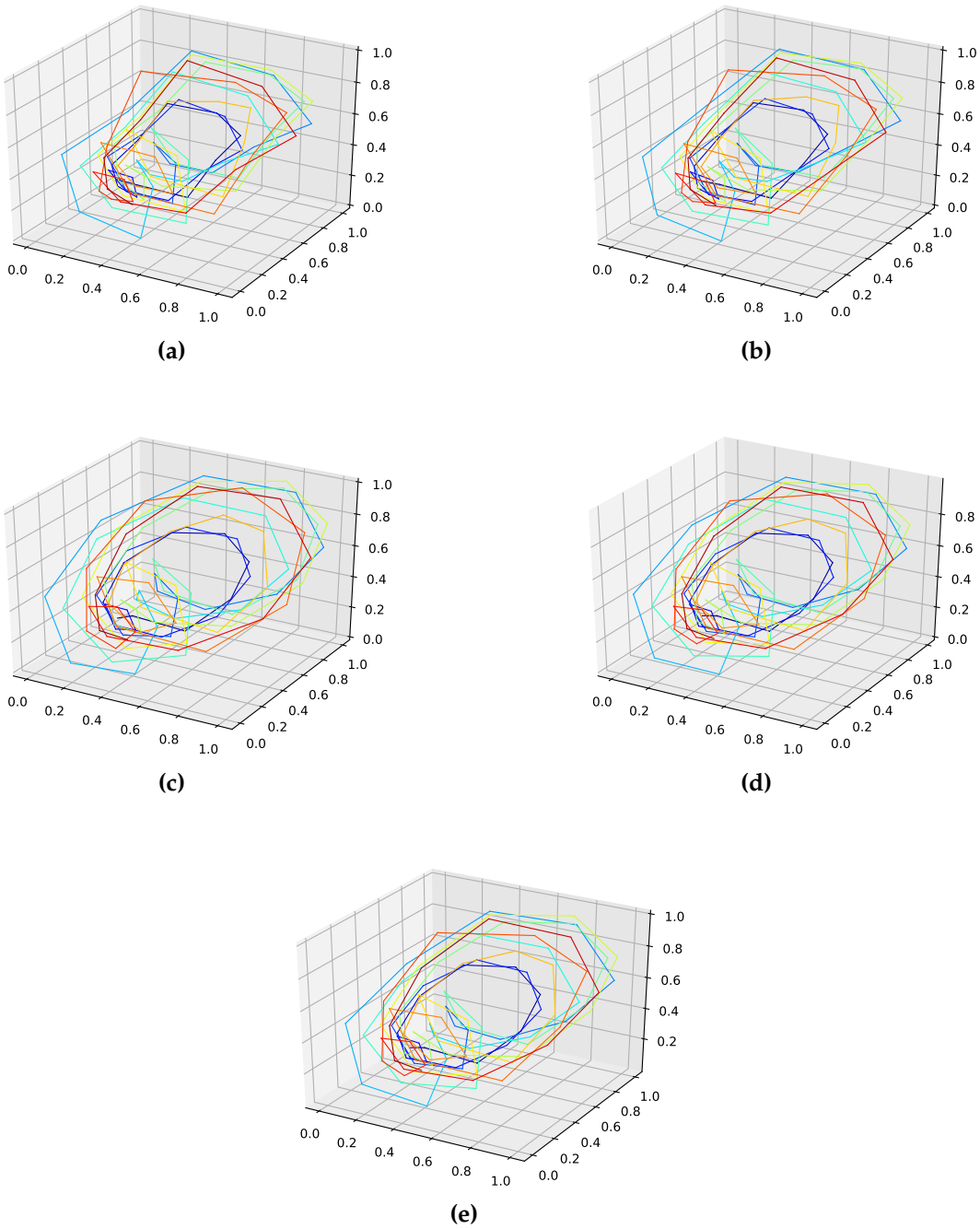


Figure 15. Reconstructed validation attractors (one interpolation point) for the car sales in Quebec data set.
(a): Reconstructed attractor, average population validation interpolation;
(b): Reconstructed attractor, linear interpolation;
(c): Reconstructed attractor, spline interpolation;
(d): Reconstructed attractor, best random validation interpolation;
(e): Reconstructed attractor, gen. alg. improved validation interpolation;

 doi:10.20944/preprints202204.0167.v1

A.4. Perrin Freres Champagne Sales Data Set

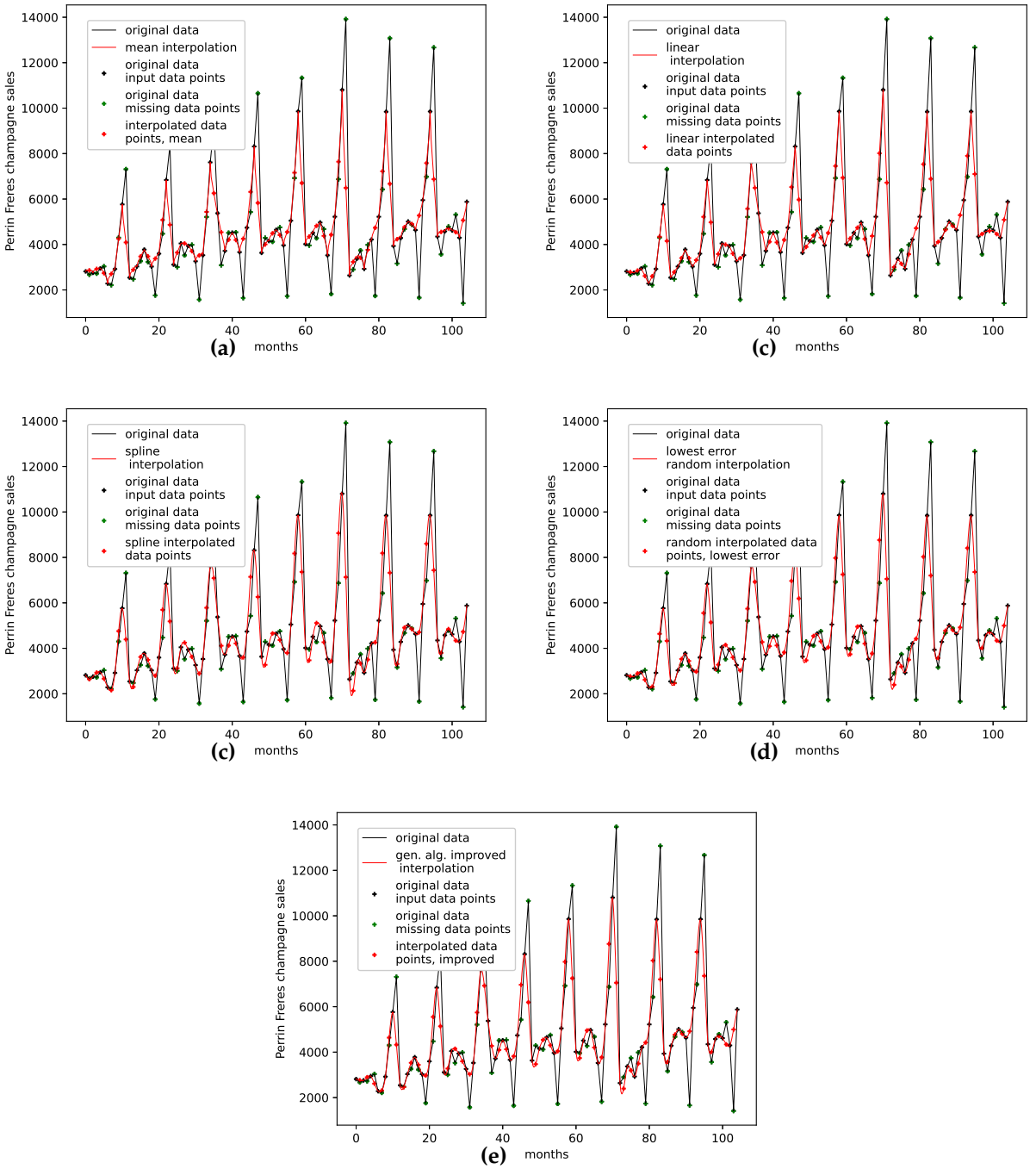


Figure 16. Interpolated validation data (seven interpolation points) for the Perrin Freres champagne sales data set.
(a): Average population validation;
(b): Validation, linear interpolation;
(c): Validation, spline interpolation;
(d): Validation, best random interpolation;
(e): Validation, gen. alg. improved interpolation;

 doi:10.20944/preprints202204.0167.v1

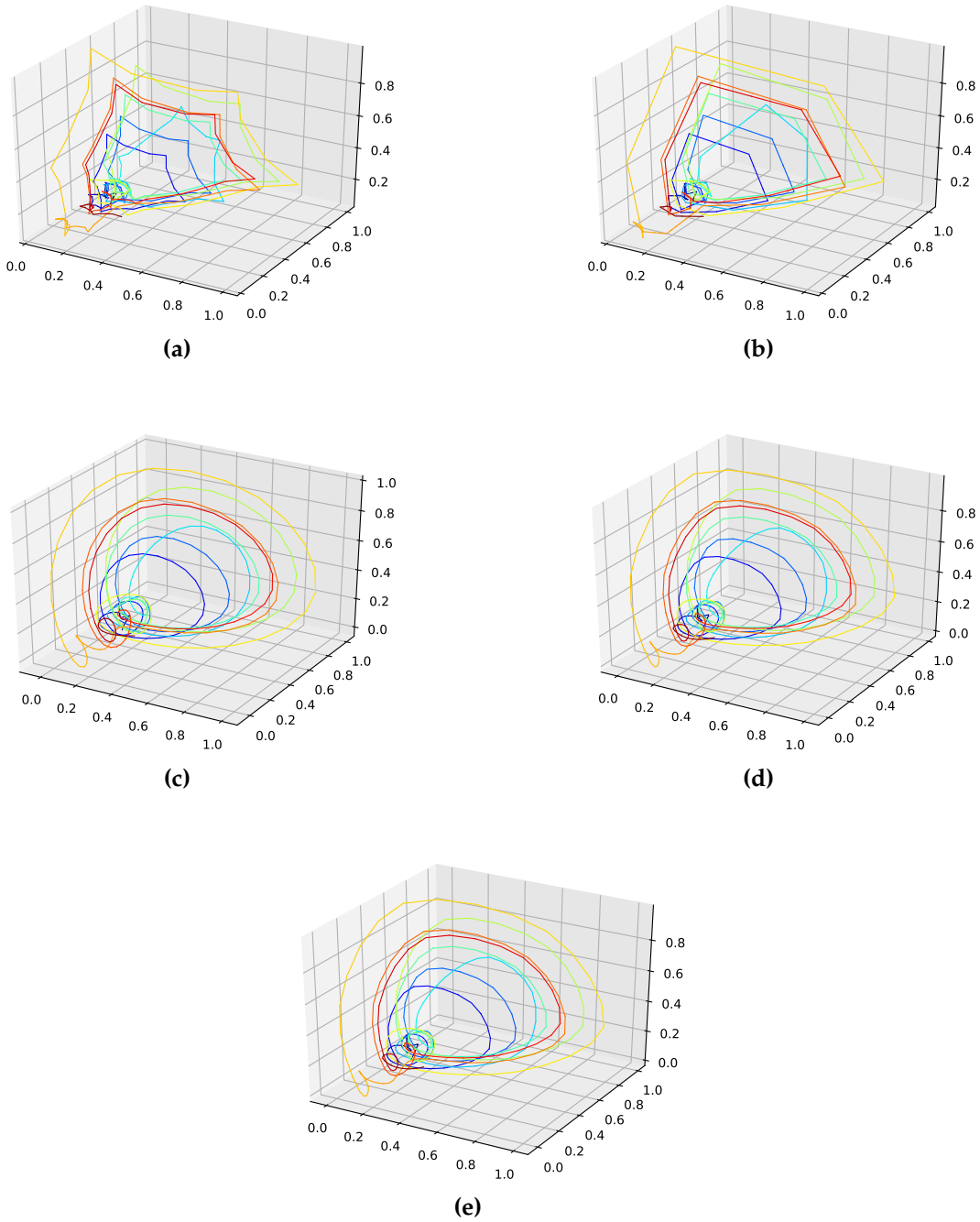


Figure 17. Reconstructed validation attractors (seven interpolation points) for the Perrin Freres champagne sales data set.
(a): Reconstructed attractor, average population validation interpolation;
(b): Reconstructed attractor, linear interpolation;
(c): Reconstructed attractor, spline interpolation;
(d): Reconstructed attractor, best random validation interpolation;
(e): Reconstructed attractor, gen. alg. improved validation interpolation;

 doi:10.20944/preprints202204.0167.v1

528 *A.5. Monthly Airline Passengers Data Set*

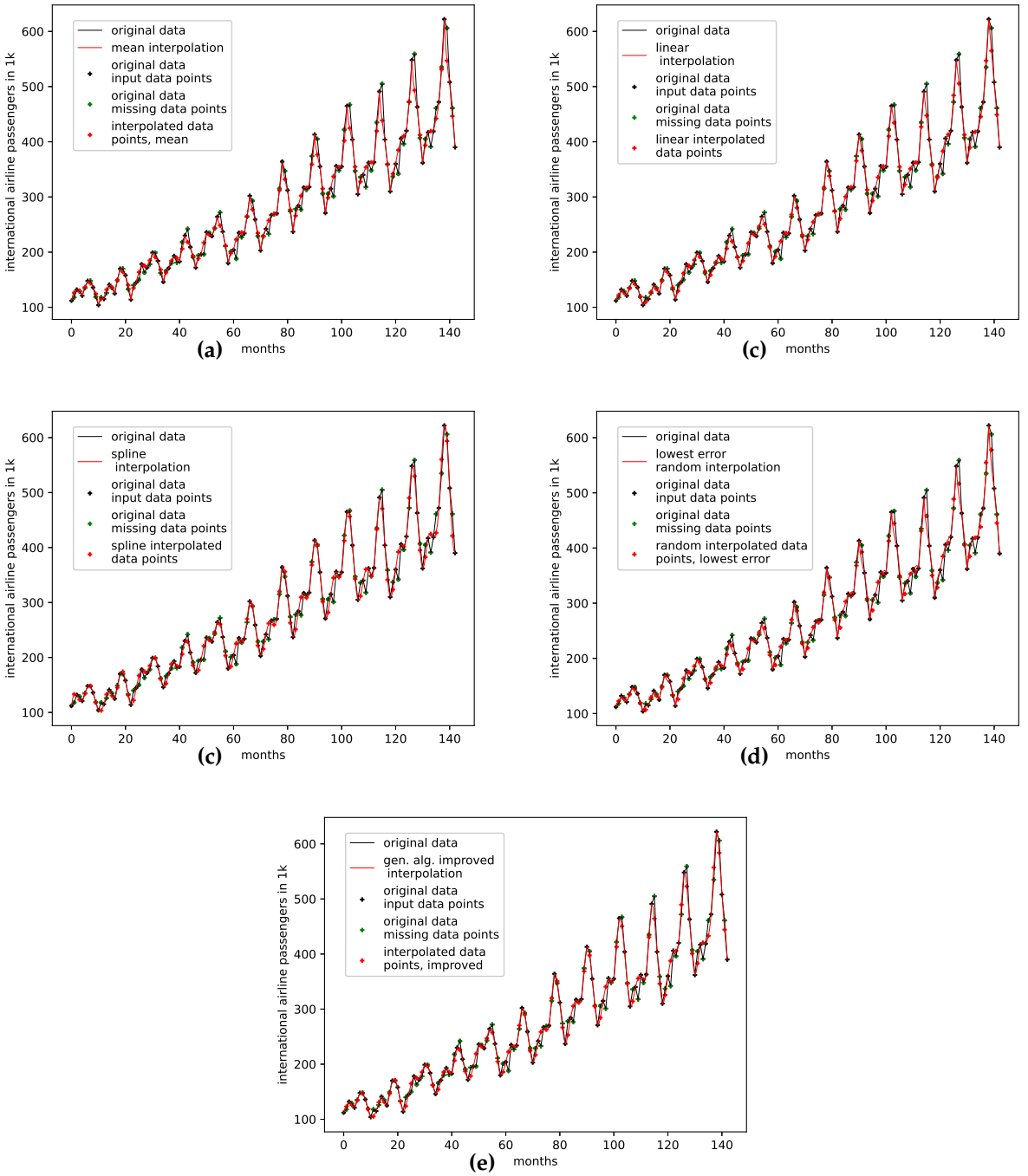


Figure 18. Interpolated validation data (three interpolation points) for the monthly international airline passengers data set.
(a): Average population validation;
(b): Validation, linear interpolation;
(c): Validation, spline interpolation;
(d): Validation, best random interpolation;
(e): Validation, gen. alg. improved interpolation;

 doi:10.20944/preprints202204.0167.v1

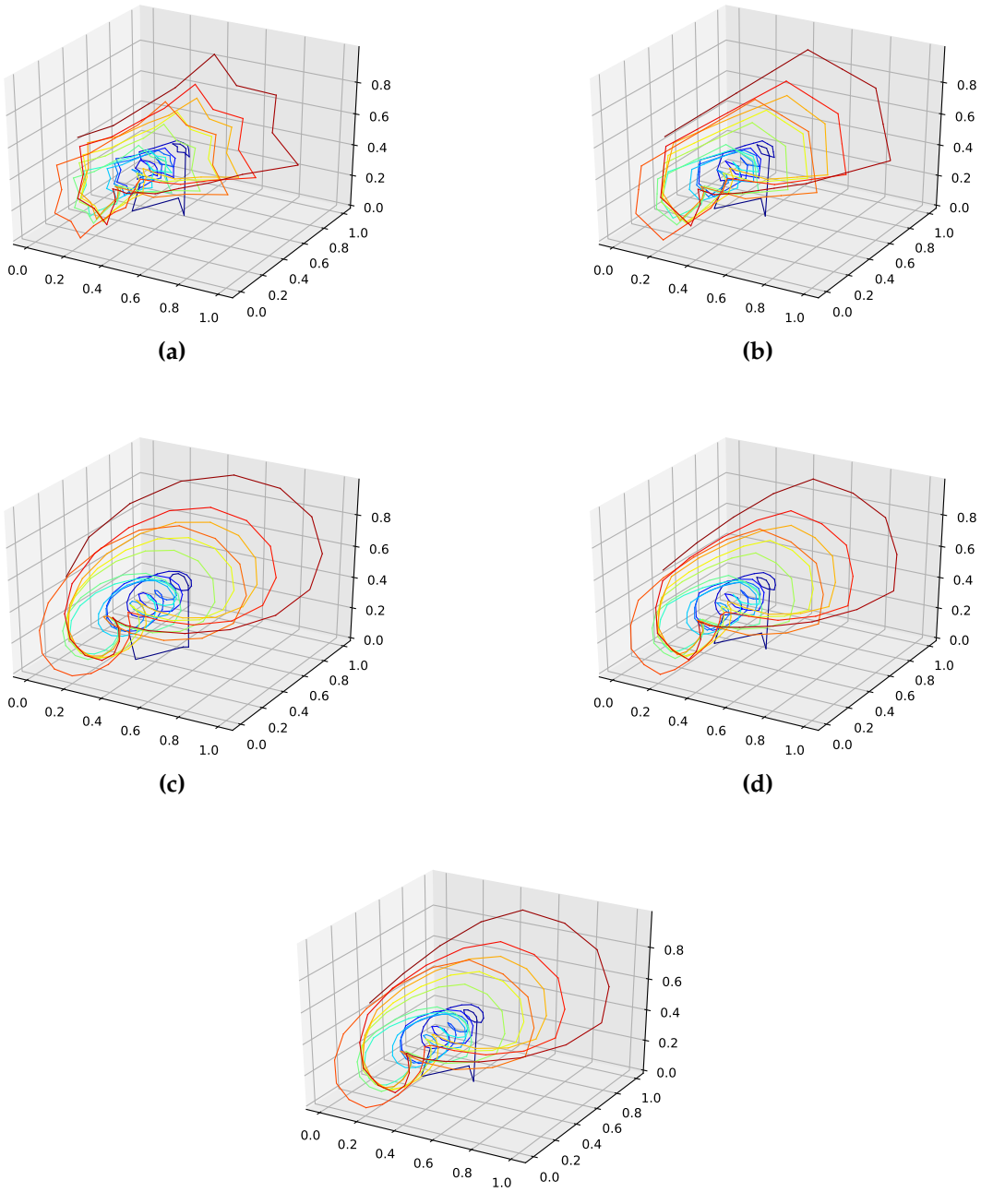


Figure 19. Reconstructed validation attractors (three interpolation points) for the monthly international airline passengers data set.
(a): Reconstructed attractor, average population validation interpolation;
(b): Reconstructed attractor, linear interpolation;
(c): Reconstructed attractor, spline interpolation;
(d): Reconstructed attractor, best random validation interpolation;
(e): Reconstructed attractor, gen. alg. improved validation interpolation;

 doi:10.20944/preprints202204.0167.v1

529 *A.6. Monthly Mean Temperature in Nottingham Castle Data Set*

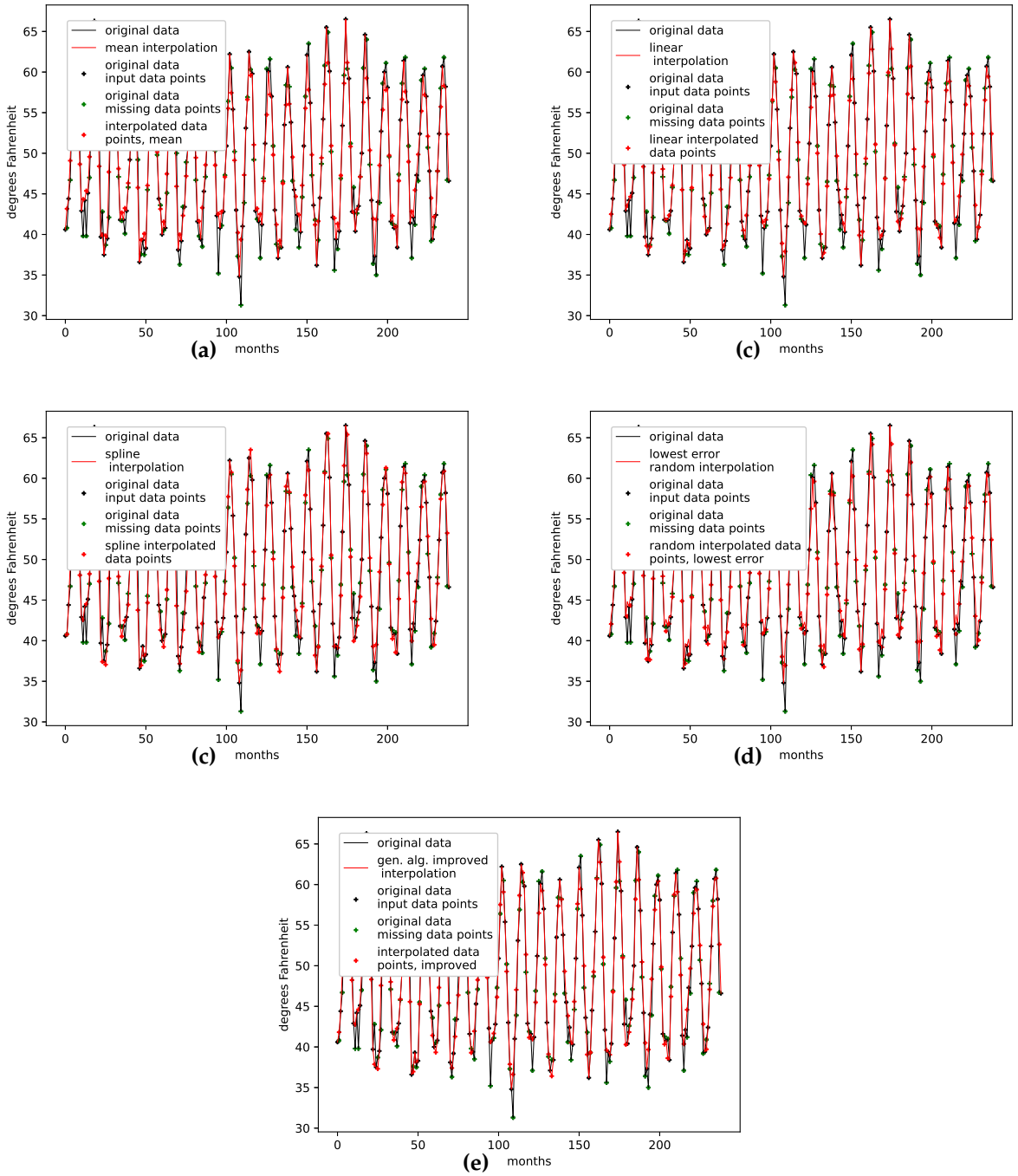


Figure 20. Interpolated validation data (one interpolation point) for the monthly mean temperature in Nottingham castle data set.
(a): Average population validation;
(b): Validation, linear interpolation;
(c): Validation, spline interpolation;
(d): Validation, best random interpolation;
(e): Validation, gen. alg. improved interpolation;

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 April 2022 doi:10.20944/preprints202204.0167.v1

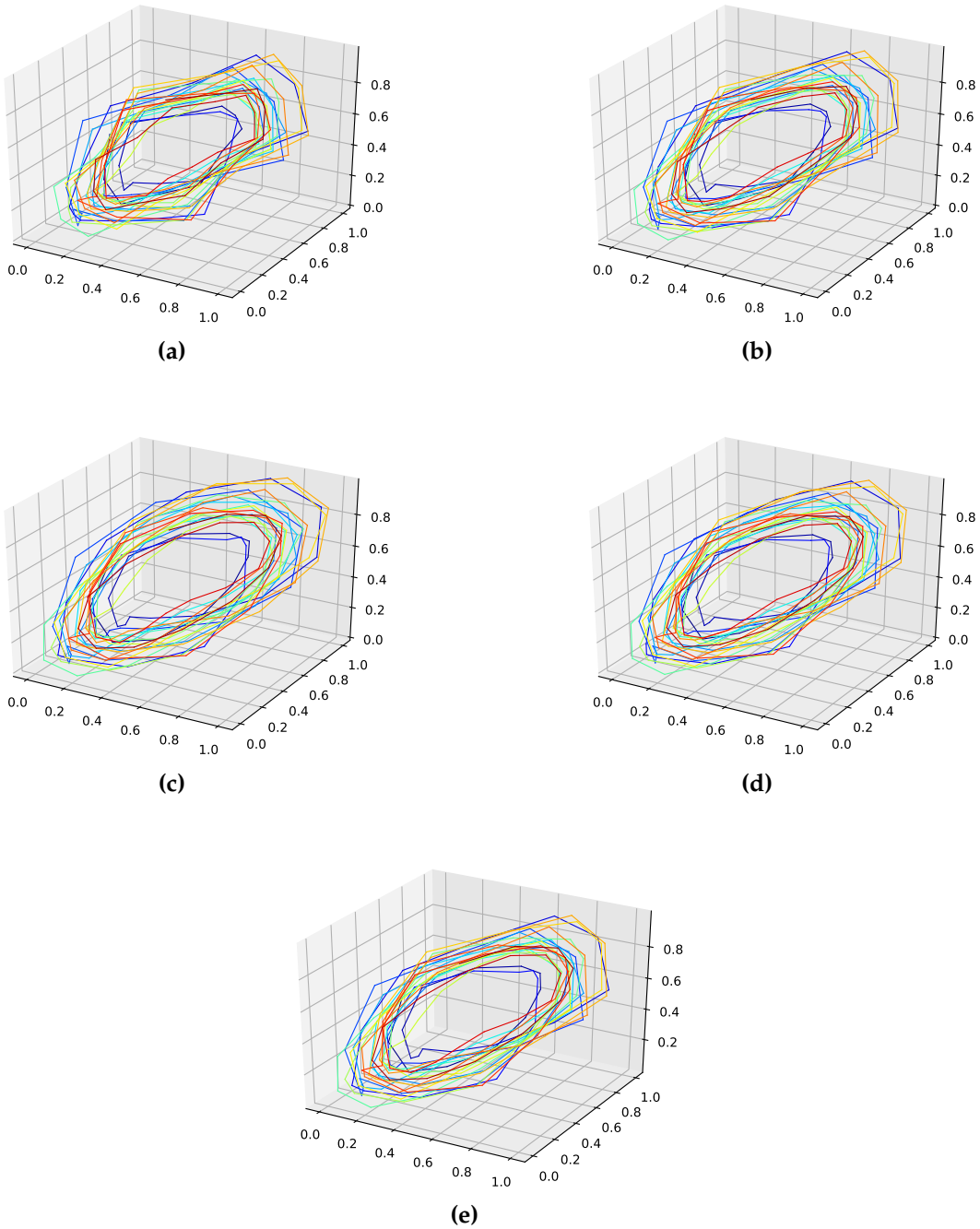


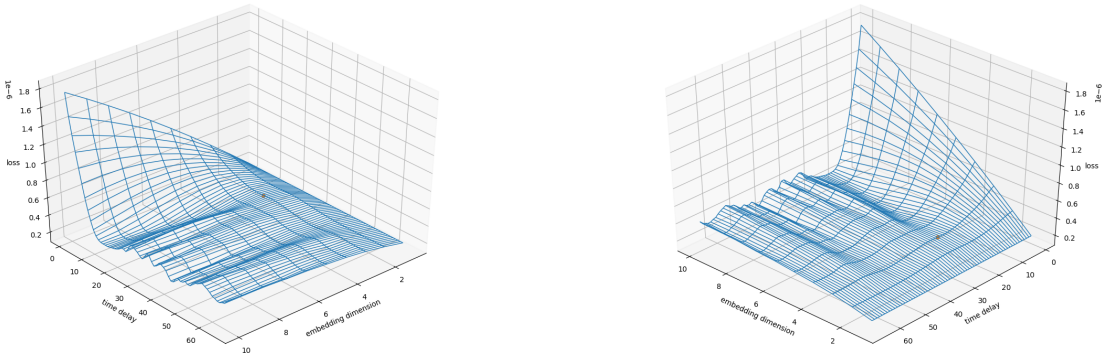
Figure 21. Reconstructed validation attractors (one interpolation point)for the monthly mean temperature in Nottingham castle data set.
(a): Reconstructed attractor, average population validation interpolation;
(b): Reconstructed attractor, linear interpolation;
(c): Reconstructed attractor, spline interpolation;
(d): Reconstructed attractor, best random validation interpolation;
(e): Reconstructed attractor, gen. alg. improved validation interpolation;

B. Loss Surface

We present the loss surface for the Lorenz attractor in Figure 22 from two perspectives. The orange dot marks the actual embedding of the Lorenz system. The plot suggests that the correct phase space embedding is located in an area where the loss

 doi:10.20944/preprints202204.0167.v1

surface flattens out. At this point, we did not check for possible ways to locate the correct phase space embedding in the loss surface. Future approaches might find ways to do so.



(a) **(b)**

Figure 22. Loss surface for the Lorenz attractor.
(a) and **(b)** both show the same surface from different angles. This is the employed loss function (Section 3.3.1) depending on a varying embedding dimension and time delay. The orange dot marks the correct embedding dimension and time delay.

C. Failed Attempts

This section provides additional material for failed attempts to find a smooth phase space trajectory. For this reason, we provide additional plots (Figure 23) and the corresponding errors for the Lorenz system in Table 7. These attempts for different loss functions include:

- Minimizing the nearest neighbour distance between phase space points.
- Minimizing the mean of first-order derivatives along the phase space trajectory.
- Minimizing the variance of first-order derivatives along the phase space trajectory.
- Minimizing the mean of second order derivatives along the phase space trajectory.

 doi:10.20944/preprints202204.0167.v1

Table 7: Errors for the interpolated data on the Lorenz system for 14 interpolation points and different loss functions. The Errors are shown for the mean interpolation of all populations, the lowest error in the population, and the interpolation that was improved using the presented genetic algorithm. Further, we give the percentage of how much of the population is outperformed by the genetic algorithm improved interpolation. Here, one can see that only methods including the second derivatives performed well. Further, the variance of second-order derivatives along the phase space trajectory performed best.

Loss Function $\rightarrow$	**Nearest Neighbour Distance**	**First Derivative Mean**	**First Derivative Variance**	**Second Derivative Mean**	**Second Derivative Variance**
RMSE Population Mean	0.90686				
Lowest RMSE in population	0.18632				
RMSE gen. alg. improved	1.13779	0.67649	0.54291	0.19274	**0.18626**
Below Best %	73.9%	62.9%	55.5	4.2%	**0.1%**

 doi:10.20944/preprints202204.0167.v1

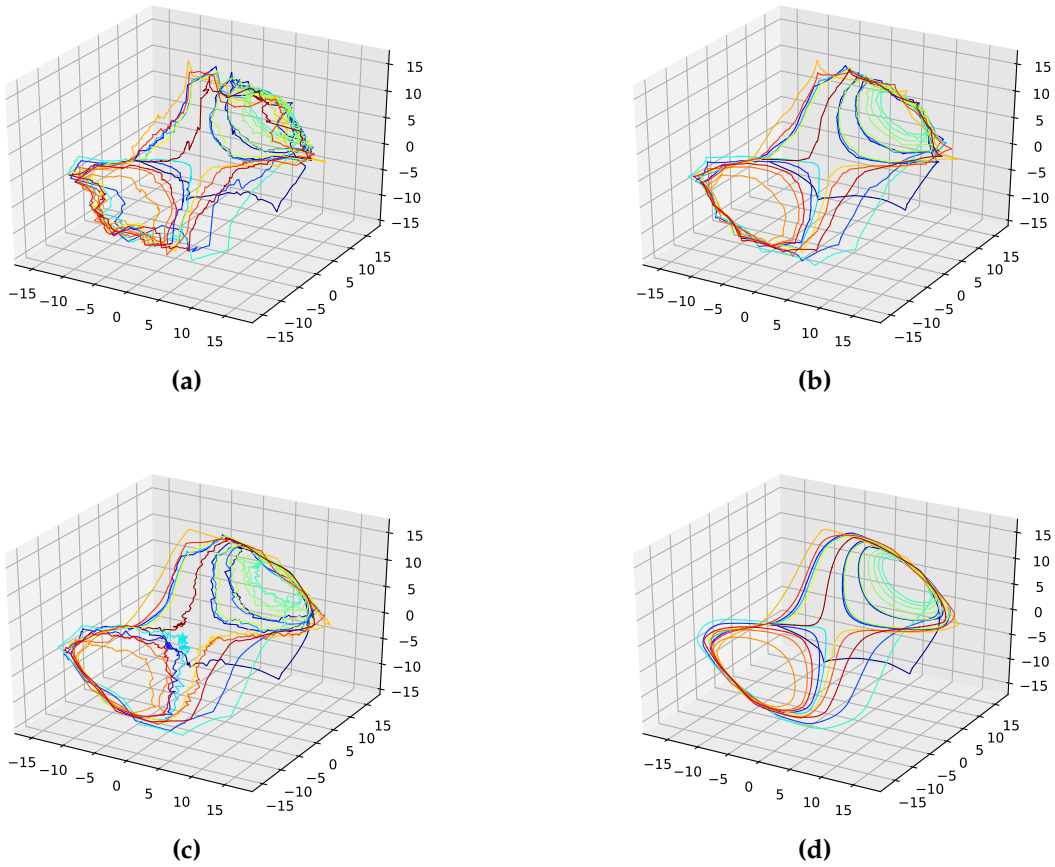


Figure 23. Reconstructed attractors for the interpolated Lorenz system for different loss functions.
(a): Nearest neighbour distance loss function;
(b): First derivative mean loss function;
(c): First derivative variance loss function;
(d): second derivative mean loss function;

References

1. Friedrich, J.; Gallon, S.; Pumir, A.; Grauer, R. Stochastic Interpolation of Sparsely Sampled Time Series via Multipoint Fractional Brownian Bridges. Phys. Rev. Lett. **2020**, 125, 170602. doi:10.1103/PhysRevLett.125.170602.
2. Stark, J.; Broomhead, D.S.; Davies, M.E.; Huke, J. Takens embedding theorems for forced and stochastic systems. Nonlinear Analysis: Theory, Methods & Applications **1997**, 30, 5303 – 5314. doi:https://doi.org/10.1016/S0362-546X(96)00149-6.
3. Raubitzek, S.; Neubauer, T. A fractal interpolation approach to improve neural network predictions for difficult time series data. Expert Systems with Applications **2021**, 169, 114474. doi:https://doi.org/10.1016/j.eswa.2020.114474.
4. Raubitzek, S.; Neubauer, T. Taming the Chaos in Neural Network Time Series Predictions. Entropy **2021**, 23. doi:10.3390/e23111424.
5. Pech-Pacheco, J.; Cristobal, G.; Chamorro-Martinez, J.; Fernandez-Valdivia, J. Diatom autofocusing in brightfield microscopy: a comparative study. Proceedings 15th International Conference on Pattern Recognition. ICPR-2000, 2000, Vol. 3, pp. 314–317 vol.3. doi:10.1109/ICPR.2000.903548.
6. Chang, C.; Lo, S.; Yu, S. Applying fuzzy theory and genetic algorithm to interpolate precipitation. Journal of Hydrology **2005**, 314, 92–104. doi:10.1016/j.jhydrol.2005.03.034.
7. Sinhuber, M.; Friedrich, J.; Grauer, R.; Wilczek, M. Multi-level stochastic refinement for complex time series and fields: A data-driven approach. New J. Phys. **2021**.
8. Delorme, M.; Wiese, K.J. Extreme-value statistics of fractional Brownian motion bridges. Phys. Rev. E **2016**, 94, 052105. doi:10.1103/PhysRevE.94.052105.
9. Sottinen, T.; Yazigi, A. Generalized Gaussian bridges. Stochastic Processes and their Applications **2014**, 124, 3084–3105. doi:https://doi.org/10.1016/j.spa.2014.04.002.
10. Takens, F. Detecting strange attractors in turbulence. In Dynamical Systems and Turbulence, Warwick 1980, Lecture Notes in Mathematics; Rand, D.; Young, L.S., Eds.; Springer-Verlag: Berlin Heidelberg, 1981; Vol. 898, pp. 366–381.

11. Packard, N.H.; Crutchfield, J.P.; Farmer, J.D.; Shaw, R.S. Geometry from a Time Series. Phys. Rev. Lett. **1980**, 45, 712–716. doi:10.1103/PhysRevLett.45.712.
12. Fraser, A.M.; Swinney, H.L. Independent coordinates for strange attractors from mutual information. Phys. Rev. A **1986**, 33, 1134–1140. Publisher: American Physical Society, doi:10.1103/PhysRevA.33.1134.
13. Rhodes, C.; Morari, M. The false nearest neighbors algorithm: An overview. Computers & Chemical Engineering **1997**, 21, S1149–S1154. Supplement to Computers and Chemical Engineering, doi:https://doi.org/10.1016/S0098-1354(97)87657-0.
14. Quarteroni, A.; Sacco, R.; Saleri, F. Numerical Mathematics; Vol. 37, 2007. doi:10.1007/b98885.
15. Lorenz, E.N. Deterministic nonperiodic flow. Journal of atmospheric sciences **1963**, 20, 130–141.
16. Albrecht, P. The Runge–Kutta theory in a nutshell. SIAM Journal on Numerical Analysis **1996**, 33, 1712–1735.
17. Hall, C.A.; Meyer, W. Optimal error bounds for cubic spline interpolation. Journal of Approximation Theory **1976**, 16, 105–122. doi:https://doi.org/10.1016/0021-9045(76)90040-X.
18. Brunton, S.; Brunton, B.; Proctor, J.; Kaiser, E.; Kutz, J. Chaos as an Intermittently Forced Linear System. Nature Communications **2016**, 8. doi:10.1038/s41467-017-00030-8.
19. LONDON, W.P.; YORKE, J.A. RECURRENT OUTBREAKS OF MEASLES, CHICKENPOX AND MUMPS: I. SEASONAL VARIATION IN CONTACT RATES1. American Journal of Epidemiology **1973**, 98, 453–468, [https://academic.oup.com/aje/article-pdf/98/6/453/269755/98-6-453.pdf]. doi:10.1093/oxfordjournals.aje.a121575.
20. Hyndman, R.; Yang, Y. Time Series Data Library v0.1.0. pkg.yangzhuoranyang.com/tsdl, 2018.
21. Friedrich, J.; Peinke, J.; Pumir, A.; Grauer, R. Explicit construction of joint multipoint statistics in complex systems. J. Phys. Complexity **2021**, 2, 045006.