Least squares B-spline approximation with applications to geospatial point clouds

Amiri-Simkooei, Alireza; Esmaeili, Fatemeh; Lindenbergh, Roderik

DOI
10.1016/j.measurement.2025.116887

Publication date
2025

Document Version
Final published version

Published in
Measurement: Journal of the International Measurement Confederation

Measurement 248 (2025) 116887

Contents lists available at ScienceDirect

Measurement

journal homepage: www.elsevier.com/locate/measurement

Least squares B-spline approximation with applications to geospatial point clouds

Alireza Amiri-Simkooei [a,*], Fatemeh Esmaeili [b], Roderik Lindenbergh [c]

[a] Department of Control and Operations, Faculty of Aerospace Engineering, Delft University of Technology, Delft, the Netherlands
[b] Department of Geomatics Engineering, Faculty of Civil Engineering and Transportation, University of Isfahan, Isfahan 81746-73441 Iran
[c] Department of Geoscience and Remote Sensing, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, the Netherlands

ARTICLE INFO

Keywords:
Least squares B-spline approximation (LSBSA)
Quality control measures
4D point cloud data
Curve and surface fitting
Manifold fitting

ABSTRACT

Fitting a smooth curve to 2D, a surface to 3D, and a manifold to 4D irregular point cloud data is becoming a common practice in many engineering and science applications. Piecewise-polynomial spline functions provide a powerful tool applicable to interpolation and approximation problems. This study presents the least squares B-spline approximation (LSBSA) theory, which is a generalized version of the spline interpolation and can be applied to any irregularly scattered point cloud data at knots specified by the user. The formulation allows to apply the well-established body of knowledge of least squares theory to the B-spline approximation. This for example has the benefit of embedding quality control measures such as hypothesis testing and proper error propagation to assess the quality of the approximation problem. The method is applicable to many 1D curve, 2D surface and 3D manifold fitting problems of which both simulated and real data are used to illustrate the efficacy of the proposed theory. In particular, its real-world applications to multi-beam echo-sounder bathymetric data, digital terrain modeling and Greenland ice sheet deformation monitoring will be highlighted. The performance of the method for linear, quadratic, cubic and quartic spline functions will be investigated. The primary application of LSBSA lies in its ability to perform 3D manifold fitting for deformation monitoring. This capability provides the possibility of monitoring changes in continuous spatial and temporal domains. The Python and Matlab source codes of LSBSA are freely accessible at https://github.com/tudelft4d/lsbsa.

1. Introduction

Curve and surface fitting problems by means of interpolation and approximation of point cloud data have applications in geometric modeling, reverse engineering and computer vision [1]. One such application is to model point clouds of objects, buildings, infrastructures, land surface and seabed. Examples of collected geospatial point clouds are laser scanning data, multi-beam echo-sounder data and InSAR data. Other applications include computer aided geometrical design and 3D-printing.

Appropriate approximation and modelling of point cloud data has advantages above using the raw observations due to the following reasons: i) The original data usually contain noise and outliers, and if not accounted for, it is rather difficult to extract patterns of interest from such noisy observations [2,3]. ii) Huge amount of data in point clouds makes the processing and interpretation of results difficult. A point cloud approximation by piecewise polynomials provides a surface description consisting of, in general, a much smaller number of coefficients than the number of raw observations. This is in conjunction with manifold learning, as a dimension reduction algorithm for high-dimensional data splines [4]. iii) Time series analysis of point clouds (e.g. for deformation monitoring of objects) requires structured data sets because the irregularly scattered data points cannot be directly compared between successive epochs. iv) Deterministic modelling of data allows to obtain analytical expressions and hence to extract function values and their derivatives, e.g. its gradients, as a tool for deformation monitoring. v) Appropriate approximation is a way to generate a watertight surface implicitly defined at any desired location. Approximation and modelling of point cloud data can be performed by the least squares method [5], which will be elaborated upon in the present contribution.

One of the widely used tools for continuous surface representation is the parametric spline surface, having an explicit mathematical expression. They are among the simplest functions frequently used in many

* Corresponding author at: Department of Control and Operations, Faculty of Aerospace Engineering, Delft University of Technology, Delft, The Netherlands.
E-mail address: a.amirisimkooei@tudelft.nl (A. Amiri-Simkooei).

https://doi.org/10.1016/j.measurement.2025.116887
Received 29 July 2024; Received in revised form 2 January 2025; Accepted 28 January 2025
Available online 28 January 2025

approximation problems [6]. In the context of raster interpolation, the term 'spline' refers to piecewise-polynomial splines. 1D and 2D spline polynomials can be defined at any polynomial order. Examples include first-order polynomials, called linear splines, which have to be continuous at their knots. Second-order polynomials are quadratic splines, which are continuous and have continuous slopes at their knots. Cubic splines impose their continuity, and have continuous slopes and curvatures at the knots [7].

Different types of spline functions have been introduced in literature. They include B-splines [8–10], thin plate splines [4], T-splines [11], triangular B-splines [12], and locally refined B-splines [13,14]. The B-spline, imposing the above continuity constraints in an implicit form, is a function that has non-zero values only within a certain interval of its definition. The theory of B-splines was first introduced in [15], and it has gained attention because they are well-conditioned and can easily be implemented [16]. In recent years, with the development of data acquisition techniques and the production of point clouds, the use of B-splines in interpolation, approximation and modeling of 3D surfaces has received serious attention [17]. The ability of B-splines to reconstruct complex shapes has made these functions highly effective in point cloud approximation [18]. Applications of spline functions have been demonstrated in geoscience problems like cycle slip detection in GPS carrier phase observations [19], estimating calibration curve in sonar systems [20], and combining regional and global geoid models [21].

The B-spline polynomial approximation has also been used for wavefront estimation [22]. Unlike the conventional Shack-Hartmann sensors, which primarily use Zernike polynomials effective for circular apertures, B-Spline approximations are excellent at dealing with rectangular or ring-shaped light beams and surfaces with local irregularities. Constructed from the tensor product of basis functions defined over knot vectors in two directions and associated coefficients, B-spline surfaces have advantageous properties such as non-negativity and linear independence from B-spline curves. Studies have shown that B-spline approximations provide superior estimates of optical surface flatness and curvature compared to Zernike polynomials, particularly under challenging conditions involving noise and missing data [23,24].

This work employs B-splines and least squares approximation, a well-known methodology for surface fitting. Our method is framed within the context of Locally Refined (LR) B-splines, which provide a robust alternative for datasets with significant local variability. Dokken, et al. [25] first introduced LR B-splines, showcasing their capability for progressive local refinement in axes-parallel partitions. Skytt, et al. [26] demonstrated LR B-splines' ability to represent terrain data compactly by adapting the resolution of the spline space to local data features, combining least squares and multilevel B-spline approximation to handle large, complex datasets effectively. Their work highlights the potential of LR B-splines for applications requiring both global smoothness and localized detail. Recent advances, such as [27], have extended these methods to incorporate fault jump estimates for adaptive spline constructions, while [28] highlighted LR B-splines' practicality for GIS applications and their statistical optimization. In our study, tensor product B-spline surfaces will be used due to the relatively smooth nature of the data, which does not exhibit the extreme local variability where LR B-splines excel. While LR B-splines would provide additional flexibility for datasets with heterogeneous features, their computational overhead may not be justified for uniformly distributed data. Notably, for the data analysed in this article, which includes multi-beam echo-sounder data, the laser scanner data and the Greenland ice sheet mass data, the local variability is well-distributed, indicating that local refinements are unnecessary.

In this paper, we introduce the theory of least squares B-spline approximation (LSBSA), which offers innovation in two key aspects. i) The formulation of LSBSA in terms of least squares theory enables the application of well-established principles from this theory to B-spline approximation. This approach provides several advantages, including the integration of quality control measures such as hypothesis testing and accurate error propagation for evaluating result reliability. For example, hypothesis testing can be employed to determine optimal knot placement and the degree of the B-spline polynomial, ensuring a balance between model complexity and data fit. In addition, it can be applied for outlier detection, which allows the identification and exclusion of erroneous observations. ii) Deformation analysis in many engineering applications is often conducted using high order numerical manifold methods (HONMMs) [29]. In HONMMs, stress and strain parameters, the two key parameters in deformation analysis, commonly exhibit discontinuous at the boundary between elements, and these methods are restricted to solving solely 2D problems [30]. Therefore, it is essential to develop an effective method capable of handling 3D problems in a continuous form [31]. LSBSA offers a solution for deformation analysis through 3D manifold fitting, which also guarantees the continuity of functions at cell boundaries across spatial and temporal domains.

This paper is organized as follows. Section 2 presents the least squares B-spline approximation (LSBSA) method, as a generalized form of LS-BICSA proposed in [20]. The topics to be addressed are B-spline polynomials, the 1D curve fitting problem, the 2D surface fitting problem, and the 3D manifold fitting problem, all formulated within the least squares framework. Section 3 applies the existing body of knowledge of the least squares theory to LSBSA. In particular the quality control measures, hypothesis testing to identify outlying observations and proper error propagation laws can directly be extended to LSBSA. Numerical results are presented in Section 4. Both simulated and real-world applications are presented to illustrate the theory. For the real-world applications, we will investigate the performance of LSBSA method in earth surface modelling using multi-beam echo-sounder bathymetric data, terrestrial laser scanner data, and GRACE liquid water equivalent (LWE) thickness time series over the Greenland region. Section 5 presents the conclusions.

2. Least squares B-spline approximation (LSBSA)

In this section, we present the least squares B-spline approximation (LSBSA). LSBSA can be applied to 1D curve, 2D surface, and 3D manifold fitting problems. The method uses the basis spline (B-spline), which is a piecewise polynomial function of a given degree. The details are provided in the following five subsections.

2.1. B-spline function

A spline function of order $p+1$ is a piecewise polynomial function of degree p in a variable u, which is referred to as p^{th}-degree piecewise polynomial spline function. The places where the pieces meet are known as knots $\lambda_0, \lambda_1, \cdots, \lambda_h$ ($h \geq p+1$). The key property of the spline function is that it is continuous and has continuous derivatives up to and including degree $p-1$ at the intermediate knots $\lambda_1, \cdots, \lambda_{h-1}$.

There are different methods to make spline functions. B-splines are among the methods frequently used in many applications. B-splines of a given order are required as basis functions to construct a spline function of the same order defined over the same knots. The B-spline satisfies [16]

$$B_{i,p+1}(u) \equiv \begin{cases} \neq 0 & \text{if} \quad \lambda_i \leq u < \lambda_{i+p+1} \\ = 0 & \text{else} \end{cases} \tag{1}$$

If the B-splines are constrained to satisfy $\sum_i B_{i,p+1}(u) = 1$, for all u between the first and last knot, their scaling factors becomes fixed and they can uniquely be determined. An arbitrary spline function $S_p(u)$ of order p is expressed as a linear combination of B-splines

$$S_p(u) = \sum_i \alpha_i B_{i,p+1}(u) \tag{2}$$

indicating that all possible spline functions can be built as an appropriate linear combination of B-splines, and there is only one unique

combination for each spline function. B-splines can be constructed by means of the Cox–de Boor recursion formula. This recursive relation between B-spline functions of order $k+1$ and those of order k is as follows [16]:

$$B_{i,k+1}(u) = \frac{u-\lambda_i}{\lambda_{i+k}-\lambda_i} B_{i,k}(u) + \frac{\lambda_{i+k+1}-u}{\lambda_{i+k+1}-\lambda_{i+1}} B_{i+1,k}(u) \tag{3}$$

Let us assume the knots λ_i, λ_{i+1}, λ_{i+2},λ_{i+3} are given. As a starting point, the three B-splines $B_{k,1}(u), k = i, \cdots, i+2$ of order 1 are defined as

$$B_{k,1}(u) = \begin{cases} 1 & \text{if} \quad \lambda_k \le u < \lambda_{k+1} \\ 0 & \text{else} \end{cases} \tag{4}$$

Applying the above recursive formula gives the two B-splines $B_{k,2}(u)$, $k = i, i+1$ of order 2 as

$$B_{i,2}(u) = \begin{cases} \dfrac{u-\lambda_i}{\lambda_{i+1}-\lambda_i} & \text{if} \quad \lambda_i \le u < \lambda_{i+1} \\ \dfrac{\lambda_{i+2}-u}{\lambda_{i+2}-\lambda_{i+1}} & \text{if} \quad \lambda_{i+1} \le u < \lambda_{i+2} \end{cases} \tag{5}$$

and

$$B_{i+1,2}(u) = \begin{cases} \dfrac{u-\lambda_{i+1}}{\lambda_{i+2}-\lambda_{i+1}} & \text{if} \quad \lambda_{i+1} \le u < \lambda_{i+2} \\ \dfrac{\lambda_{i+3}-u}{\lambda_{i+3}-\lambda_{i+2}} & \text{if} \quad \lambda_{i+2} \le u < \lambda_{i+3} \end{cases} \tag{6}$$

which are piecewise polynomial functions of degree $p = 1$ (linear splines). $B_{i,2}(u)$ is continuous at λ_{i+1}, whereas $B_{i+1,2}(u)$ is continuous at λ_{i+2}. Based on these two functions we may apply the recursive formula (3) to obtain the only B-spline $B_{i,3}(u)$ of order 3 as

$$B_{i,3}(u) = \begin{cases} \dfrac{(u-\lambda_i)^2}{(\lambda_{i+2}-\lambda_i)(\lambda_{i+2}-\lambda_{i+1})} & \text{if} \quad \lambda_i \le u < \lambda_{i+1} \\ \dfrac{(u-\lambda_i)(\lambda_{i+2}-u)}{(\lambda_{i+2}-\lambda_i)(\lambda_{i+2}-\lambda_{i+1})} + \dfrac{(u-\lambda_{i+1})(\lambda_{i+3}-u)}{(\lambda_{i+3}-\lambda_{i+1})(\lambda_{i+2}-\lambda_{i+1})} & \text{if} \quad \lambda_{i+1} \le u < \lambda_{i+2} \\ \dfrac{(\lambda_{i+3}-u)^2}{(\lambda_{i+3}-\lambda_{i+1})(\lambda_{i+3}-\lambda_{i+2})} & \text{if} \quad \lambda_{i+2} \le u < \lambda_{i+3} \end{cases} \tag{7}$$

which is a piecewise polynomial function of degree $p = 2$ (quadratic spline), that is continuous and differentiable at λ_{i+1} and λ_{i+2}. In case of four knots λ_i, λ_{i+1},λ_{i+2} and λ_{i+3} this is the maximum-degree B-spline. Fig. 1 shows two typical examples of quadratic splines defined on the knots $\lambda_i = 1$, $\lambda_{i+1} = 2.5$, $\lambda_{i+2} = 4$, $\lambda_{i+3} = 5$ and knots $\lambda_{i+1} = 2.5$, $\lambda_{i+2} = 4$, $\lambda_{i+3} = 5$, $\lambda_{i+4} = 7$. Adding the fifth knot λ_{i+4} allows to make the only piecewise polynomial function of degree 3 (cubic spline). For the sake of brevity, a B-spline is denoted as $B_{i,p+1} = B_i$, where p denotes the degree of the piecewise polynomial and i denotes λ_i, the starting knot of the B-spline.

2.2. Curve fitting problem

A spline curve is a piecewise polynomial curve where the polynomial pieces are joined together at specified knots. This subsection explains how to make a spline curve using B-spline functions. Let $y = [y_1, y_2, \cdots, y_m]^T$ be a set of data points (e.g. heights) measured at fixed but randomly scattered positions $u = [u_1, u_2, \cdots, u_m]^T$, where m is the number of observations. In general, we assume the measurements contain random errors, known as noise. We look then for a best approximation rather than interpolation, indicating that the approximating function closely follows the data points. That is, the approximation is not necessarily passing through the observations, and is

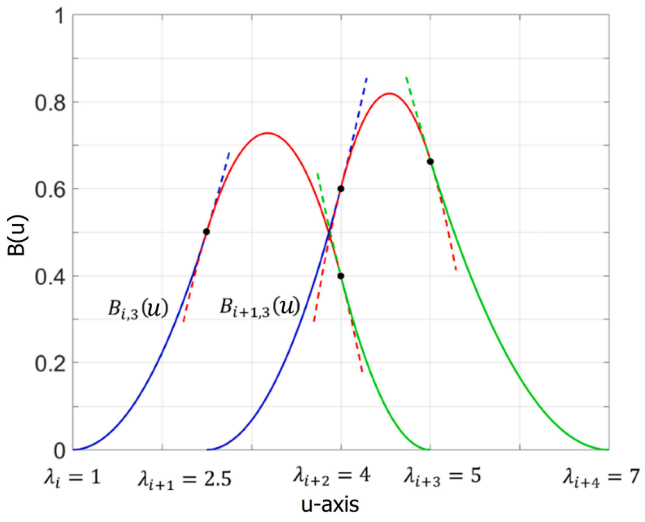

Fig. 1. Two typical B-spline functions $B_{i,3}(u)$ and $B_{i+1,3}(u)$ defined on knots $\lambda_i = 1$, $\lambda_{i+1} = 2.5$, $\lambda_{i+2} = 4$, $\lambda_{i+3} = 5$ and $\lambda_{i+1} = 2.5$, $\lambda_{i+2} = 4$, $\lambda_{i+3} = 5$, $\lambda_{i+4} = 7$, respectively.

therefore not exact. We aim to fit a spline function of order $p+1$, i.e. a piecewise polynomial function of degree p, which best approximates the data points in the least squares sense. This is achieved by the B-splines given in Section 2.1. The places for the knots at which the pieces meet are assumed to be $a = \lambda_0 \le \lambda_1 \le \cdots \le \lambda_h = b$, which are generally different from the entries u_i, $i = 1, \cdots, m$; we assume $u_i \in [a, b]$ and usually $h < m$.

To define the spline function on the entire range $[a, b]$, a full set of B-splines is required to be determined. We need to introduce $2p$ additional knots as $\lambda_{-p} < \cdots < \lambda_{-1} < a$ and $b < \lambda_{h+1} < \cdots < \lambda_{h+p}$. For linear, quadratic and cubic splines, p will take values 1, 2 and 3, respectively, corresponding to the so-called spline functions of order $p+1 = 2, 3, 4$. The entire spline function over $[a, b]$ consists of the sum of $n = h+p$ B-splines starting from M_{-p}, defined on the interval $[\lambda_{-p}, \lambda_0]$ and ending at M_{h-1} defined on the interval $[\lambda_{h-1}, \lambda_{h+p}]$. The spline function $y = f(u)$ is an unknown linear combination of the B-spline functions, denoted as

$$f(u) = \sum_{i=-p}^{h-1} \alpha_i B_i(u) \tag{8}$$

where the coefficients $\alpha_{i-p-1}, i = 1, \cdots, n$ are the unknown parameters to be estimated. The above equation should be written for each data point $y_k, k = 1, \cdots, m$ as $y_k = f(u_k) = \sum_{i=-p}^{h-1} \alpha_i B_i(u_k)$. The unknown parameters can be estimated by solving the following linear system of equations:

$$y = Ax + e \tag{9}$$

where y is a vector of m observations, e is a vector of m residuals, A is the $m \times n$ design matrix and $x = [\alpha_1, \cdots, \alpha_n]^T$ is the vector of n unknown coefficients. There is usually redundancy in the model ($m > n$), and therefore the least squares method is used to estimate the unknown parameters.

Example: Fig. 2 (top) illustrates a function $f(u)$ over the domain $a = 0$ to $b = 5$. To express this function as a linear combination of a series of B-spline functions, we first define the knots λ_i's, defined as $a = \lambda_0 = 0$, $\lambda_1 = 1$, …, $b = \lambda_5 = 5$ ($h = 5$). To represent this function using quadratic splines, $2 \times 2 = 4$ additional knots are introduced: $\lambda_{-2} = -2$ and $\lambda_{-1} = -1$ (before λ_0) and $\lambda_6 = 6$ and $\lambda_7 = 7$ (after λ_5), see Fig. 2 (bottom). The entire spline function over $[a, b]$ consists of the combination of $n = h+p = 5+2 = 7$ B-splines starting from B_{-2}, defined on the interval $[\lambda_{-2}, \lambda_1]$ and ending at B_4, defined on the interval $[\lambda_4, \lambda_7]$. For this example, the spline function $f(u)$ can be expressed as $f(u) =$

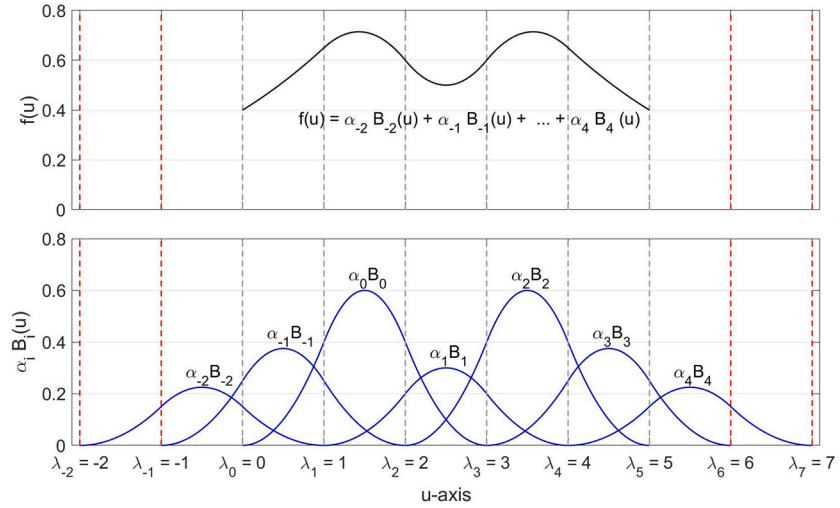

Fig. 2. A typical example of curve fitting using 7 quadratic B-spline polynomials. Vertical dashed lines indicate the position of knots. The function (top) is expressed as a combination of these B-splines (bottom): $f(u) = \alpha_{-2}B_{-2}(u) + \cdots + \alpha_4 B_4(u)$, where the coefficients are $\alpha_{-2} = 0.3$, $\alpha_{-1} = 0.5$, $\alpha_0 = 0.8$, $\alpha_1 = 0.4$, $\alpha_2 = 0.8$, $\alpha_3 = 0.5$, and $\alpha_4 = 0.3$.

$\sum_{i=-2}^{4}\alpha_i B_i(u)$, where the known coefficients are given as: $\alpha_{-2} = 0.3$, $\alpha_{-1} = 0.5$, $\alpha_0 = 0.8$, $\alpha_1 = 0.4$, $\alpha_2 = 0.8$, $\alpha_3 = 0.5$, and $\alpha_4 = 0.3$. However, in practical applications these coefficients are typically unknown and must to be estimated using the least squares method (see Section 2.5).

The observed value y_k is only linked to $p+1$ B-splines: indeed, for a given interval of adjacent knots $[\lambda_{i-1}, \lambda_i]$ that contains a data point (u_k, y_k), only $p+1$ B-splines have nonzero values, and therefore the remaining B-spline are zero. To further clarify this, let us assume that there exist m_i observations in $[\lambda_{i-1}, \lambda_i], i = 1, \cdots, h$ (note $m = \sum_{i=1}^{h} m_i$). The design matrix is then of the form

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,p+1} & 0 & 0 & \cdots \\ 0 & a_{2,1} & a_{2,2} & \cdots & a_{2,p} & a_{2,p+1} & 0 & \cdots \\ 0 & 0 & a_{3,1} & \cdots & a_{3,p-1} & a_{3,p} & a_{3,p+1} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{10}$$

where $a_{i,j}, i = 1, \cdots, h, j = 1, \cdots, p+1$ are column vectors of size m_i related to observations in $(\lambda_{i-1}, \lambda_i)$ and the j^{th} B-spline. This will accordingly make a sparse structure for the design matrix A. Subsection 2.5 explains how to obtain the least squares solution of the unknown coefficients.

2.3. Surface fitting problem

A spline surface is a piecewise 2D polynomial surface where the pieces are joined together at specified boundaries [7]. This subsection explains how to make the spline surface using 2D B-spline functions. Let $y = [y_1, y_2, \cdots, y_m]^T$ be a set of data points measured at fixed but randomly scattered 2D positions $(u_1, v_1), \cdots, (u_m, v_m)$, where m is the number of observations. Here we also assume the measurements contain random errors. We look for the best approximating surface that closely follows the data points. We aim to fit two spline functions of order $p+1$ (along u) and $q+1$ (along v), representing 2D piecewise polynomial functions of degrees p and q, respectively, which best approximates the data points in the least squares sense. The places for the knots at which the pieces meet are assumed to be $a_u = \lambda_0 \le \lambda_1 \le \cdots \le \lambda_h = b_u$ (along u) and $a_v = \gamma_0 \le \gamma_1 \le \cdots \le \gamma_g = b_v$ (along v). We assume $u_i \in [a_u, b_u]$ and $v_i \in [a_v, b_v]$, $i = 1, \cdots, m$.

A piecewise-polynomial spline surface in 2D consists of a set of polynomial pieces (surface patches) that are smoothly connected by continuity conditions at their boundaries [32]. These pieces represented by $\sum_{i=0}^{p}\sum_{j=0}^{q} c_{ij} u^i v^j$ are sections defined by each squared (rectangular) grid cell, which is a 2D polynomial of degree $p+q$. If $p = q$, it is called bi-polynomic of degree p. Examples of $p = q = 1, 2, 3$ refer to bilinear, biquadratic and bicubic polynomials, respectively. The least squares bicubic spline approximation (LS-BICSA) has imposed the continuity constraints of zero, first and second-orders as a few hard linear constraints in the linear model of observation equations [7]. The advantage of this method is that the user can explicitly impose the appropriate continuity and differentiability conditions at the boundary. For cubic splines, when the continuity constraints are up to and including the second order, the LS-BICSA can be solved by using the B-spline theory in which the continuity constraints are imposed in an implicit form to the estimation problem.

In general, when all continuity and differentiability constraints are imposed up to and including degree $p-1$ and $q-1$, along the u and v axes, respectively, it is easier to employ the B-spline theory of Section 2.1. This can be achieved by using a set of basis functions, along the u and v axes. Such a set can be constructed from the tensor product of two 1D sets, see for example [16]. We denote the set of basis functions in u with respect to the knots $\lambda_0 \le \lambda_1 \le \cdots \le \lambda_h$ as $B_u(u)$ and those in v with respect to the knots $\gamma_0 \le \gamma_1 \le \cdots \le \gamma_g$ as $B_v(v)$. The set of cross-products

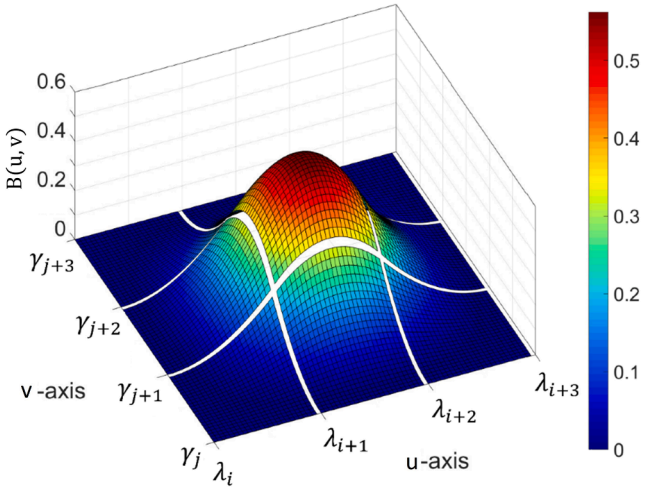

Fig. 3. A typical quadratic B-spline function $B_{uv}(u, v) = B_{i,u}(u)B_{j,v}(v)$ consisting of nine cells of piecewise second-degree polynomials defined on knots λ_i, λ_{i+1},λ_{i+2} and λ_{i+3} (along u) and γ_j, γ_{j+1},γ_{j+2} and γ_{j+3} (along v).

$B_{uv}(u,v) = B_u(u)B_v(v)$ provides a basis for the 2D polynomial of degree $p+q$. Fig. 3 illustrates a typical quadratic 2D B-spline function defined on knots λ_i, λ_{i+1},λ_{i+2} and λ_{i+3} (along u) and γ_j, γ_{j+1},γ_{j+2} and γ_{j+3} (along v).

To define the spline function on the entire plane $[a_u, b_u]$ and $[a_v, b_v]$, a full set of cross-product B-splines is required to be determined. We need to introduce $2p$ additional knots as $\lambda_{-p} < \cdots < \lambda_{-1} < a_u$ and $b_u < \lambda_{h+1} < \cdots < \lambda_{h+p}$ and $2q$ additional knots as $\gamma_{-q} < \cdots < \gamma_{-1} < a_v$ and $b_v < \gamma_{g+1} < \cdots < \gamma_{g+q}$. The entire spline function over the plane is a linear combination of $n = (h+p)(g+q)$ number of cross-product B-splines as follows

$$f(u,v) = \sum_{i=-p}^{h-1} \sum_{j=-q}^{g-1} \alpha_{ij} B_{i,u}(u) B_{j,v}(v) \tag{11}$$

where the coefficients $\alpha_{ij}, i = -p, \cdots, h-1, j = -q, \cdots, g-1$ are the n number of unknown parameters to be estimated. The above equation should be written for each data point $(u_k, v_k), k = 1, \cdots, m$ as $y_k = f(u_k, v_k) = \sum_{i=-p}^{h-1} \sum_{j=-q}^{g-1} \alpha_{ij} B_{i,u}(u_k) B_{j,v}(v_k)$. The unknown parameters should be estimated through the linear system of equations $y = Ax + e$, where y is a vector of m observations, e is a vector of m residuals, A is the $m \times n$ design matrix and $x = \alpha_{ij}$ is vector of n unknown coefficients. Fig. 4 presents a schematic algorithm illustrating the implementation of LSBSA for a surface fitting problem.

The observed value y_k is related to $(p+1)(q+1)$ number of B-splines because for a given cell of adjacent knots $[\lambda_{i-1}, \lambda_i]$ and $[\gamma_{j-1}, \gamma_j]$, which contain (u_k, v_k) and y_k, only $(p+1)(q+1)$ B-splines have nonzero values. For the bi-quadratic B-spline in Fig. 5, this corresponds to $3 \times 3 = 9$ B-

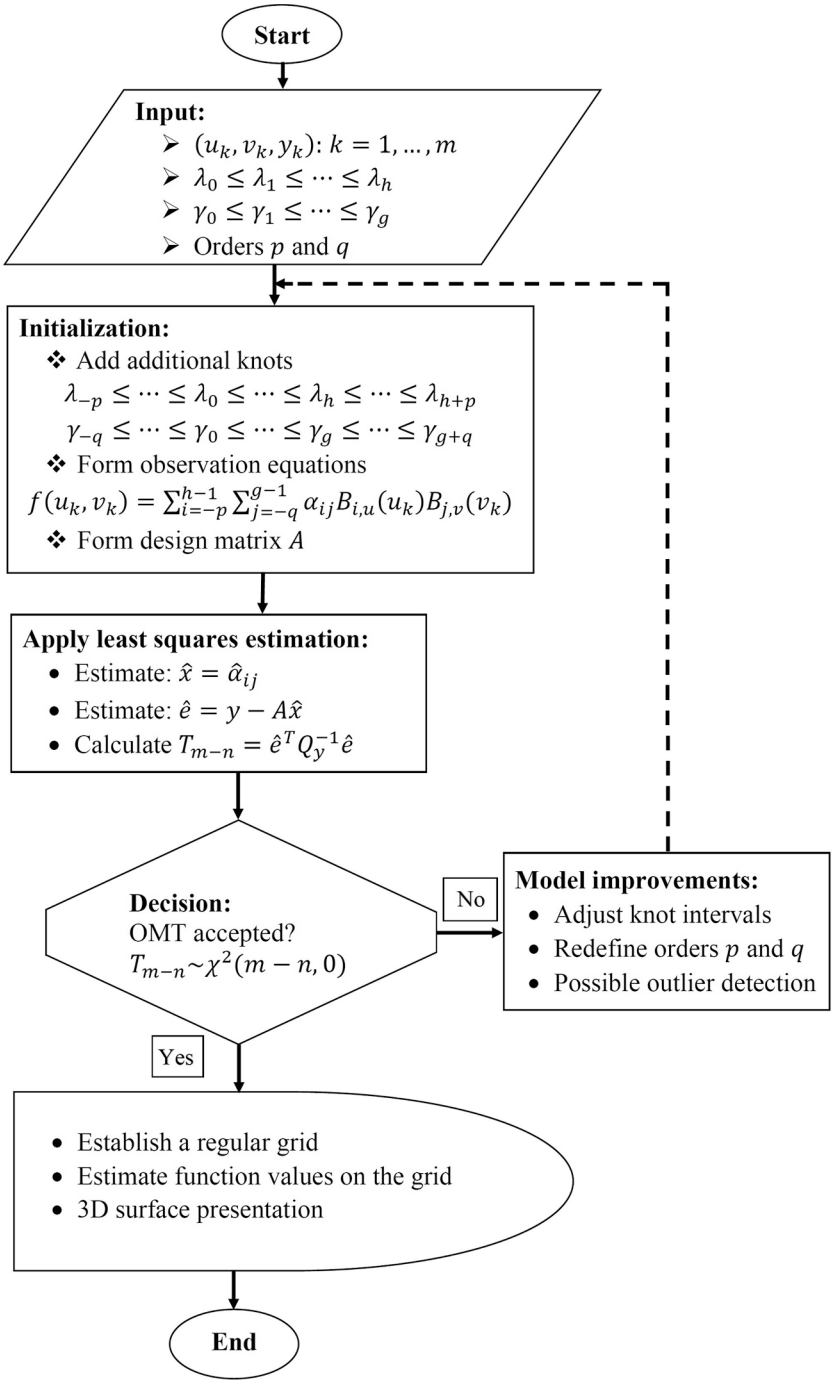

Fig. 4. Schematic representation of the algorithm for implementing LSBSA in the 2D surface fitting problem.

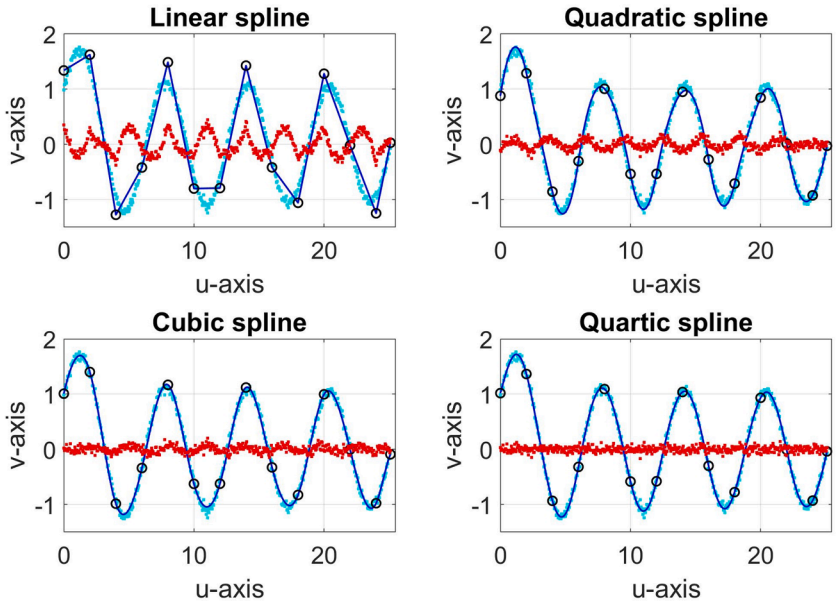

Fig. 5. Least squares approximation of $f(u) = \sin u + \operatorname{sinc} u$ contaminated by random normal noise with a standard deviation of 0.05 using linear, quadratic, cubic and quartic spline functions. Cyan dots are data, red dots are residuals, black circles are spline knots, and the blue curve is the spline function. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

splines. Therefore, data points given in a specific cell are related to only $(p+1)(q+1)$ columns in the design matrix, referring to the same number of coefficients in A. A structure for the design matrix similar as presented in Eq. (10) can also be formulated for the 2D case.

2.4. 3D manifold fitting problem

The theory of B-spline curve and surface approximation can be extended to any dimension. We briefly explain how the method can be extended to 3D manifold approximation. A spline manifold is a piecewise 3D polynomial where the pieces can be glued together at specified boundaries (planes). Let $y = [y_1, y_2, \cdots, y_m]^T$ be a set of data points measured at the fixed but randomly scattered 3D positions $(u_1, v_1, t_1), \cdots, (u_m, v_m, t_m)$, where m is the number of observations. For example, the coordinates (u_i, v_i) can be Cartesian or geographical coordinates while t_i is the time instant (epoch). We look for the best approximating manifold that closely follows the data points. We aim to fit three spline functions of order $p+1$ (along u), $q+1$ (along v), and $r+1$ (along t), representing a 3D piecewise polynomial function of degrees p, q, and r which best approximates the data points in the least squares sense. The knot places at which the pieces meet are assumed to be identical to those specified in the 2D surface approximation along the u and v axes. Further, along t, we have $a_t = \eta_0 \le \eta_1 \le \cdots \le \eta_d = b_t$ (along t). We therefore assume $u_i \in [a_u, b_u]$, $v_i \in [a_v, b_v]$, and $t_i \in [a_t, b_t]$, $i = 1, \cdots, m$.

We will impose all the continuity and differentiability constraints up and including to degree $p-1$, $q-1$ and $r-1$, along the u, v and t axes, respectively. This is achieved by using a set of basis functions, along the u, v and t axes. Similar to the previous subsection, such set can be constructed from the tensor product of three 1D sets of basis functions as $B_{uvt}(u,v,t) = B_u(u)B_v(v)B_t(t)$, which provides a basis for the 3D polynomial of degree $p+q+r$.

Similar to the 1D and 2D cases, to define the spline function on the entire rectangular cuboid $[a_u, b_u]$, $[a_v, b_v]$ and $[a_t, b_t]$, a full set of cross-product B-splines is required to be determined. In addition to the $2p$ and $2q$ additional knots (along u and v), we need to introduce $2r$ additional knots as $\eta_{-r} < \cdots < \eta_{-1} < a_t$ and $b_t < \eta_{d+1} < \cdots < \eta_{d+r}$ along t. The entire spline function over the rectangular cuboid is a linear combination of $n = (h+p)(g+q)(d+r)$ number of cross-product B-splines as follows:

$$f(u,v,t) = \sum_{i=-p}^{h-1} \sum_{j=-q}^{g-1} \sum_{k=-r}^{d-1} \alpha_{ijk} B_{i,u}(u) B_{j,v}(v) B_{k,t}(t) \tag{12}$$

where the coefficients α_{ijk}, $i = -p, \cdots, h-1$, $j = -q, \cdots, g-1$ and $k = -r, \cdots, d-1$ are n number of unknown parameters to be estimated. The above equation should be written for each data point (u_ℓ, v_ℓ, t_ℓ), $\ell = 1, \cdots, m$ as $y_\ell = f(u_\ell, v_\ell, t_\ell) = \sum_{i=-p}^{h-1} \sum_{j=-q}^{g-1} \sum_{k=-r}^{d-1} \alpha_{ijk} B_{i,u}(u_\ell) B_{j,v}(v_\ell) B_{k,t}(t_\ell)$. The unknown parameters should be estimated through the linear system of equations $y = Ax + e$, where y is a vector of m observations, e is a vector of m residuals, A is the $m \times n$ design matrix and $x = \alpha_{ijk}$ is vector of n unknown coefficients.

2.5. Least squares estimation of B-spline coefficients

Having the linear system of equations $y = Ax + e$ available, for either the curve, surface or manifold fitting problem, we may use the least squares method to estimate the coefficients x. The spline function can then be determined. The least squares estimate of x is, [5]

$$\hat{x} = \left(A^T Q_y^{-1} A\right)^{-1} A^T Q_y^{-1} y \tag{13}$$

where Q_y is the given $m \times m$ covariance matrix of the observations. Because this problem is formulated in the least squares framework, the existing body of the least squares theory can directly be applied to the least squares B-spline problem. A few issues are highlighted as follows. The least squares estimates of the observations and residuals are

$$\hat{y} = A\hat{x} = P_A y, \hat{e} = y - A\hat{x} = P_A^\perp y \tag{14}$$

where $P_A = A\left(A^T Q_y^{-1} A\right)^{-1} A^T Q_y^{-1}$ and $P_A^\perp = I_m - P_A$ are two orthogonal projectors [5]. The covariance matrices of the above estimates are

$$Q_{\hat{y}} = A Q_{\hat{x}} A^T, Q_{\hat{e}} = Q_y - A Q_{\hat{x}} A^T \tag{15}$$

where $Q_{\hat{x}}$ is the covariance matrix of the estimated parameters $\hat{x}$, given as

$$Q_{\hat{x}} = \left(A^T Q_y^{-1} A\right)^{-1} \tag{16}$$

describing the variances of the estimated coefficients and covariances among different coefficients.

The execution speed of an algorithm is influenced by its computational complexity, which is determined by the mathematical calculations required for the task. Simplifying these calculations is an effective way to evaluate its algorithm efficiency. To analyse computational complexity, we assume $m > n$, a typical scenario in least-squares-based methods. A critical step in B-spline approximation is the inversion of the $m \times m$ covariance matrix Q_y. For a fully populated Q_y, the inversion has a computational complexity of $\mathscr{O}(m^3)$, which becomes inefficient when $m \gg n$, as is often the case with point clouds. We however consider the case when Q_y is a diagonal matrix in general, or an identity matrix in particular. In this case, the primary operations are: i) computing $A^T Q_y^{-1} A$, with a complexity of $\mathscr{O}(mn)$, and ii) calculating the inverse of $A^T Q_y^{-1} A$, i.e. $\left(A^T Q_y^{-1} A\right)^{-1}$, with a complexity of $\mathscr{O}(n^3)$. Therefore under the diagonal Q_y assumption, the overall computational complexity is $\mathscr{O} = \max\left(\mathscr{O}(n^3), \mathscr{O}(mn)\right)$. This is also investigated and validated using numerical results presented in Section 4.6.

3. Embedding of LSBSA into least squares theory

The above least squares formulation of LSBSA allows to apply the well-established body of knowledge of least squares theory to the B-spline approximation. For example it has the benefit of embedding quality control measures such as hypothesis testing and proper error propagation to assess the quality of results. This section briefly discusses some of such possibilities.

3.1. Detection, identification and adaptation (DIA)

As a validation step, we have now the possibility to apply the detection, identification and adaptation (DIA) procedure [33], to test the quality of results. Having the least squares residuals, the measure of discrepancies between observations and spline approximation functions (overall model test) is obtained as

$$T_{m-n} = \hat{e}^T Q_y^{-1} \hat{e} \; \chi^2(m-n, 0) \tag{17}$$

which has a chi-squared distribution χ^2 with $m-n$ degrees of freedom under the null hypothesis, i.e. $T_{m-n} \; \chi^2(m-n, 0)$. The above test statistics can be tested for a given confidence level. This is referred to as the overall model test (OMT). If the test is rejected it indicates that there are unresolved issues on the consistency between the model and observations. This is referred to as the 'detection' step in DIA [33]. The use of OMT in this contribution addresses two distinct problems.

a) A common issue in approximation using B-splines is determining the appropriate placement of the knots, which, as mentioned earlier, can be specified by the user. Another interrelated factor is selecting the degree of the polynomials in the B-splines. These two factors are not independent; there is a trade-off between the intervals of the knots and the degree of the polynomials. Basically, lower-degree polynomials require smaller knot intervals to achieve sufficient accuracy, while higher-degree polynomials can accommodate larger intervals. OMT provides an effective framework to manage this trade-off. After specifying the knots and polynomial degree, the least squares fit can be performed, and the test statistic OMT is calculated as: $T_{m-n} = \hat{e}^T Q_y^{-1} \hat{e}$. To ensure that the approximation error is not statistically significant given the measurement error, T_{m-n} must be smaller than the chi-squared critical value $\chi^2_\alpha(m-n, 0)$ for a given confidence level, such as $\alpha = 0.01$. If this condition is met, it indicates that the fitted data accurately represents the measurements within the allowed error bounds. This methodology will be demonstrated in Section 4.1, which focuses on a curve-fitting problem. The example will illustrate the interplay between knot placement, polynomial degree, and approximation accuracy, showcasing how OMT is used to achieve optimal results.

b) Another possible reason for the rejection of the OMT is due to the presence of outlying observations, which can be identified in the 'identification' step. In the data snooping method [34], each individual observation is screened for the presence of an outlier. The w-test statistic is then [33]:

$$w = \frac{c_y^T Q_y^{-1} \hat{e}}{\sqrt{c_y^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} c_y}} \sim \mathbf{N}(0,1) \tag{18}$$

which has a standard normal distribution. In this equation c_y is a given vector on m entries. An important application of the w-test is blunder detection. A blunder, or outlier affects just a single observation. To screen the observations, in order to identify those that are grossly falsified by outliers, m alternative hypotheses are formulated, one for each observation. They are all tested against the default or null hypothesis H_0. Vector c_y is a canonical unit vector, i.e. a vector with all zeros except for a one at the i-th position $c_{y_i} = [0, \cdots, 0, 1, 0, \cdots, 0]^T$, where i ranges from 1 to m. This screening of the observations with the above equation is referred to as data snooping. If the above w-test statistic exceeds the critical value ($Z_{\alpha/2}$) at a given confidence level ($1-\alpha$) the test is rejected. When the test for observation i is rejected, it is concluded that it is affected by some extraordinary large errors, and hence an outlier.

When outlying observations or any other misspecifications were confidently identified, they can then be compensated for in the functional model or removed from the list of observations. The final $\hat{x}$ and its statistical inference can then be presented, with all outlying samples removed. This is referred to as the 'Adaptation' step of the DIA.

So far the estimation and testing were performed individually. A new DIA estimator can combine parameter estimation and statistical testing in an integrated manner [35]. The theory of this DIA estimator can also directly be applied to the LSBSA problem, which is a direction for further research.

3.2. Variance component estimation

So far, we assumed that the covariance matrix Q_y of observations is known. This however is not always the case for many practical applications including point cloud approximation. The simplest case that violates the above assumption is that the covariance matrix is reformulated as

$$Q_y = \sigma^2 Q \tag{19}$$

where Q is a known $m \times m$ matrix and σ^2 is an unknown variance component, to be estimated. The least squares estimate of this variance component is [36]

$$\hat{\sigma}^2 = \frac{\hat{e}^T Q^{-1} \hat{e}}{m-n} \tag{20}$$

where $b = m-n$ is the redundancy of the linear model. When σ^2 is unknown, the test statistic in Eq. (17) cannot be determined, but its estimates $\hat{\sigma}^2$ provides an alternative measure for the precision of the observations and their fits to the linear model. In addition, using $\hat{\sigma}^2$ provides the possibility to apply the identification and adaptation steps of DIA [33]. It is important to note that the estimated $\hat{\sigma}^2$ is not only affected by the measurement noise but also by the possible mismodelling

due to the lack of sufficient smoothness (e.g. roughness) of the approximated surface. The curve, surface and manifold patches should be sufficiently small to avoid the mismodelling problem.

3.3. Data interpolation and interpolation quality on a regular grid

Having the coefficients $\widehat{x}$ available, we can estimate/predict values for a given horizontal position (u, v) because the coefficients $\widehat{x}$ fully determine the B-spline function $f(u,v)$. In fact, the B-spline function gives a watertight surface explicitly defined at any desired location. This is practically of importance because we can simply estimate the function values of the unseen/unobserved points at any required location. Assume we are given horizontal positions (u_1,v_1), …, (u_k,v_k), and we want to estimate the $z=f(u,v)$ B-spline values for these points, for example on a regular grid of fixed grid size to make a mesh of the predicted surface. For the given positions (u_i,v_i), $i=1,\cdots,k$, we then may construct the design matrix A_p based on the same settings as the LSBSA fit. Note that the design matrix in this case is made from the required positions (u_i, v_i) and in general not from the original observations. We can then predict (approximate) the function values at the grid as

$$\widehat{z}=\widehat{y}_p=A_p\widehat{x} \tag{21}$$

Applying the error propagation law to Eq. (21) gives the covariance matrix of the estimated values as

$$Q_{\widehat{z}}=A_pQ_{\widehat{x}}A_p^T \tag{22}$$

The square root of the diagonal entries of the above matrix provides the standard deviations of the predicted values. It is therefore possible to make confidence intervals for the estimated values. For example, assuming that the original observations are normally distributed, the maximum error expressed in the 99 % confidence interval is computed as

$$\sigma_{99}=2.58\sqrt{\mathrm{diag}(Q_{\widehat{z}})} \tag{23}$$

The above confidence internal can similarly be defined for other distributions, such as the Student's t-distribution [28]. The above measure of the maximum error is expressing the uncertainty of the model caused by observation noise. However, this measure does not take into account the modelling error, caused by the fact that, often, the object, for example road surface, seafloor, or bridge ceiling, cannot be considered as a perfectly smooth surface. Irregularities in the object under study along with its roughness can degrade the reliability of the above measure. One way to have a more realistic precision description for the estimated surface is to take into account a realistic covariance matrix Q_y of point cloud observations, as described in Section 3.2. An alternative could be to use the estimated residuals of the least squares fit to present a so-called empirical precision description for the estimated surface. These topics can also be the subject of further research.

A final remark on the approximated surface is in order. The present contribution provided the 'estimation' of the unobserved function values. With this problem, we guess the value of an unknown parameter $z=f(u,v)$ from an observation of a random vector y. So far, the randomness of the random variable z has been ignored. A more sophisticated problem deals with 'prediction' rather than the 'estimation'. This indicates that the estimation problem should become a prediction one because the unobserved parameter of interest is a random variable, and not just a deterministic parameter. The outcome prediction requires not only the functional deterministic model but also incorporates the stochastic nature of the data. This is outside the scope of this contribution and can be a topic for further research in which the stochastic characteristics of point clouds contribute to the prediction.

3.4. Regularized least squares solution of LSBSA

There are sometimes complications solving the above least squares problem. When there are too many curves or patches to be connected, or when there are gaps in the point cloud data, the system of equations $y=Ax+e$ becomes ill-posed. The design matrix A is then nearly rank deficient and therefore the normal matrix $N=A^TQ_y^{-1}A$ cannot be regularly inverted. To obtain a regular solution for the ill-posed problem, a regularization method needs to be used. The regularized least squares (RLS) theory can directly be used in the LSBSA.

One way to obtain a regular solution is to take possible prior information on the unknown parameters into account. To obtain such prior information we may use simple interpolation methods to create a larger amount of data in data sparse intervals/areas. Such information can be added to the system of equations, with much lower weights (i.e. large variances) compared to the original observations. This can simply be implemented in the weighted least squares formulation of Eq. (13), and can indeed have a regularization effect. This is achieved by introducing pseudo observations y_0 into the linear model in Eq. (9), which provides an extended linear model of observation equations as

$$\begin{bmatrix} y \\ y_0 \end{bmatrix}=\begin{bmatrix} A \\ A_0 \end{bmatrix}x+\begin{bmatrix} e \\ e_0 \end{bmatrix}, \tag{24}$$

with the joint covariance matrix

$$\mathrm{D}\begin{bmatrix} y \\ y_0 \end{bmatrix}=\begin{bmatrix} Q_y & 0 \\ 0 & Q_{y_0} \end{bmatrix} \tag{25}$$

where Q_y $(m\times m)$ and Q_{y_0} $(m_0\times m_0)$ are the covariance matrices of the original and pseudo observations, respectively. Usually, the variances of Q_{y_0} are much larger than those given in Q_y (i.e. $Q_{y_0}\gg Q_y$) to express that the prior information will not much affect the final least squares solution, but rather has a regularization effect. The regularized solution is then

$$\widehat{x}=\left(A^TQ_y^{-1}A+A_0^TQ_{y_0}^{-1}A_0\right)^{-1}(A^TQ_y^{-1}y+A_0^TQ_{y_0}^{-1}y_0) \tag{26}$$

The Tikhonov regularization method is a special case of the above equation [37]. In the particular case when $A_0=I_n$, $y_0=0$ and $Q_{y_0}=s^2I_n$, Eq. (26) simplifies to

$$\widehat{x}=\left(A^TQ_y^{-1}A+\kappa I_n\right)^{-1}A^TQ_y^{-1}y=(N+\kappa I_n)^{-1}u \tag{27}$$

where $\kappa=s^{-2}$ is the regularization parameter, to be determined. One way to determine κ is to use the L-curve method, as explained in [38].

3.5. Computational burden reduction

There are occasions that the linear system of equations is very large. This can happen when too many data points are available (m is too large), which is often the case when dealing with point cloud data such as multi-beam echo-sounder or laser scanning data. In such situation, block-structured (partitioned) least-squares models can directly be used. For this purpose, the design and normal matrices are calculated for individual (smaller) regions. Each region is a particular interval (of two consecutive knots for curve fitting) or a surface patch (of four borders of a rectangle for surface fitting). In this case, the linear system of equations can be decomposed as follows:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}=\begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix}x+\begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{bmatrix},\ \mathrm{D}\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}=\begin{bmatrix} Q_1 & 0 & \cdots & 0 \\ 0 & Q_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & Q_k \end{bmatrix} \tag{28}$$

with $y_i\in\mathbb{R}^{m_i\times 1}$ and $e_i\in\mathbb{R}^{m_i\times 1}$ respectively the vectors of m_i observations and residuals in the region $i\in(1,\cdots,k)$, $A_i\in R^{m_i\times n}$ its corresponding

design matrix, and Q_i its corresponding covariance matrix, the total number of observations is then $m = \sum_{i=1}^{k} m_i$. The unknown coefficients in Eq. (27) can then be estimated as

$$\begin{aligned}\hat{x} &= \left(\sum_{i=1}^{k} A_i^T Q_i^{-1} A_i + \kappa I_n\right)^{-1} \sum_{i=1}^{k} A_i^T Q_i^{-1} y_i \\ &= \left(\sum_{i=1}^{k} N_i + \kappa I_n\right)^{-1} \sum_{i=1}^{k} u_i\end{aligned} \tag{29}$$

where $N_i = A_i^T Q_i^{-1} A_i$'s are the individual normal matrices and $u_i = A_i^T Q_i^{-1} y_i$. The recursive least squares method is also an alternative procedure to obtain the final estimate $\hat{x}$.

In general, other existing theories of linear models, can also directly be generalized to LSBSA. For example we can apply the robust estimation methods [39] using the L1-norm minimization techniques [40,41].

4. Applications to simulated and real data sets

The least squares B-spline approximation (LSBSA) can be applied to 1D curve fitting, 2D surface fitting and 3D manifold fitting problems. The method is demonstrated for to both simulated and real data sets. The simulated data, generated on known mathematical functions, are mainly used to test the performance of the proposed method. The detailed applications are described in the following six subsections.

4.1. Approximation of a known curve

This subsection is devoted to the curve fitting problem using the 1D spline functions. For this purpose, the known function $f(u) = \sin u + \mathrm{sinc} u$, with $\mathrm{sinc} u = \frac{\sin u}{u}$, is discretely resampled on the interval $u \in [0, 8\pi]$, with a sampling interval of $\mathrm{d}u = 0.05$, providing a total of $m = 503$ observations. Independent and normally distributed noise with a standard deviation of $\sigma = 0.05$ was added to the generated data to make the final observations (cyan dots in Fig. 5).

The generated data are then approximated using the least squares B-spline function of Sections 2.2 and 2.5. Four spline functions made of linear, quadratic, cubic and quartic B-splines are used to approximate the data. The knots were identical for all cases and the knot intervals were set to a constant value of $\Delta u = 2$, making in total 14 knots. The covariance matrix is considered to be $Q_y = \sigma^2 I_m$, where I_m is an identity matrix of size m and $\sigma = 0.05$. The approximation results are presented in Fig. 5. The figure shows, that given the above settings, the linear spline does not provide appropriate approximation results. In general, the higher the spline order is, the better the approximation results will be. To further verify this, the variance components of the fitting problem were estimated as $\hat{\sigma}^2 = \frac{\hat{e}^T \hat{e}}{m-n}$, where $\hat{e} = y - A\hat{x}$ is the least squares residual vector. The approximation error is $\hat{\sigma} = 0.180, 0.087, 0.061,$ and 0.050 for the linear, quadratic, cubic and quartic spline functions, respectively. The linear spline with an approximation error of $\hat{\sigma} = 0.180$ provides the worst results. The quartic spline provides the best fit as the approximation error of $\hat{\sigma} = 0.050$ follows closely the standard deviation of the measurements (i.e. $\sigma = 0.05$).

As an additional analysis, to determine the optimal knot placement and the degree of the B-spline polynomial, we employed the overall model test (OMT) as a criterion (Eq. (17). While the knot intervals were initially set to a constant value of $\Delta u = 2$, additional tests were conducted using $\Delta u = 1$, $\Delta u = 2$ and $\Delta u = 3$. The OMT results for these three cases, indicating whether the test was accepted or rejected, are summarized in Table 1. These evaluations can be used to analyse the trade-off between model complexity and fit quality, ultimately aiding in the selection of the most appropriate configuration for the spline model. For example, an "accept" outcome indicates that the approximation error is not statistically significant, given the standard deviation $\sigma = 0.05$, whereas a "reject" outcome highlights that the approximation error is statistically significant.

Table 1
Overall model test (OMT) results for three cases of $\Delta u = 1$, $\Delta u = 2$ and $\Delta u = 3$ across linear, quadratic, cubic and quartic models.

Case	Knot int. (Δu)	Test statistic	Linear	Quadratic	Cubic	Quartic
1	3	Normalized T_{m-n}	109.62	31.34	48.44	11.88
		Result	reject	reject	reject	reject
2	2	Normalized T_{m-n}	11.56	2.96	1.25	**0.96**
		Result	reject	reject	reject	**accept**
3	1	Normalized T_{m-n}	1.26	**0.89**	**0.87**	**0.87**
		Result	reject	**accept**	**accept**	**accept**

4.2. Approximation of a mathematical surface

This subsection demonstrates the approximation of a known surface using the 2D spline functions. Consider the following 2D mathematical function

$$f(u,v) = u\exp\left(-u^2 - v^2\right) \tag{30}$$

Fig. 6a shows the above function on a regular grid on $u = -2 : 0.1 : 2$ and $v = -2 : 0.1 : 2$. A data set of $m = 20000$ points, generated at random positions using a uniform distribution over the square $-2 \le u, v \le 2$ is used to illustrate the efficacy of the proposed method (Fig. 6b). The generated data is not contaminated by random noise to merely obtain the approximation errors of different scenarios.

The goal is to approximate this data set using the least squares 2D spline theory as proposed in Sections 2.3 and 2.5. The covariance matrix is considered to be $Q_y = I_m$, where I_m is an identity matrix of size m. The root mean squares error (RMSE) is used as a measure to investigate the approximation error of the fitting problem. Two cases under two scenarios are investigated, making in total four cases. Under scenario I, the surface cells are considered to be 0.8×0.8, making in total 25 cells, whereas under scenario II they are 0.4×0.4, making in total 100 cells. Cases 1 and 2 are the linear and cubic splines, respectively. The RMSE of the fitting problem (approximation error) are presented for comparison of the above cases.

The proposed method is applied to the irregular data set in Fig. 6b to approximate the true mathematical function in Fig. 6a using the 2D B-splines. The estimated coefficients of the approximated surface are then used to predict the function values $f(u,v)$ on a regular grid $u = -2 : 0.1 : 2$ and $v = -2 : 0.1 : 2$, for comparison with their true values. The results are presented in Fig. 7 under Scenarios I and II. Table 2 provides statistics on the above four cases. Given the above settings, it is observed that a linear spline does not provide appropriate approximation results. The higher the spline order is, the better the approximation results will be. To quantify this, the root mean squares error (RMSE) is also presented in Table 2. The RMSE, for the linear and cubic splines, is 0.022 and 0.004 under scenario I, and 0.005 and 0.0002 under scenario II. In theory, because the data do not contain random errors, it is expected that the RMSE approximates zero. The best result is obtained under scenario II for the cubic splines. This is expected as the cubic spline, having first and second continuous derivatives, introduces more parameters to approximate the function $f(u,v)$.

4.3. Approximation of multi-beam echo-sounder data

Seafloor bathymetric data is usually collected by single- or multi-beam echo-sounders (SBES or MBES). The use of MBES is a common practice in many maritime applications. MBES is a type of sonar, typically used by hydrographic surveyors, to determine the depth of water

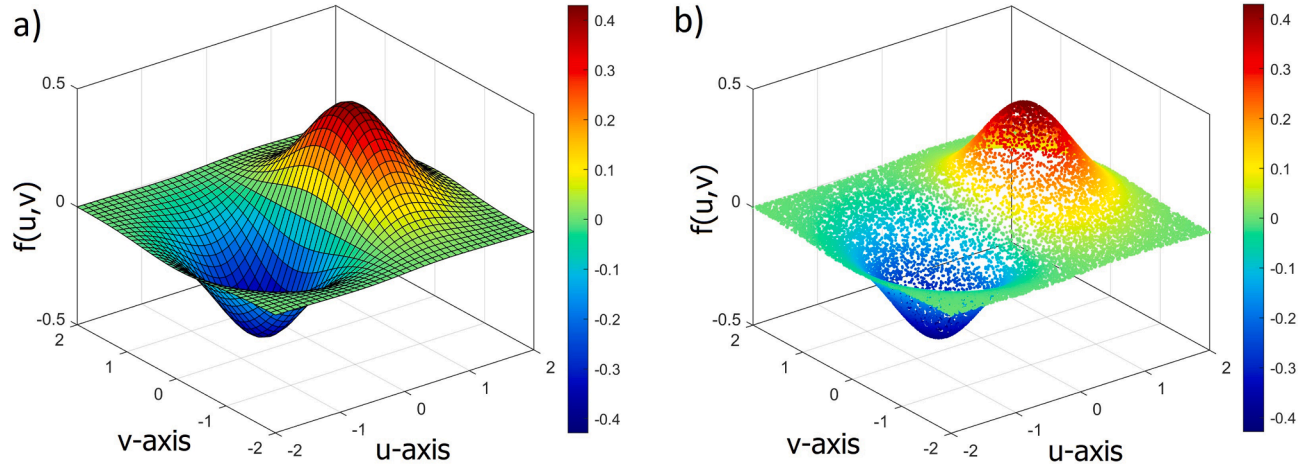

Fig. 6. Known mathematical function $f(u,v) = uexp(-u^2 - v^2)$ illustrated on a regular grid; $u = -2 : 0.1 : 2$ and $v = -2 : 0.1 : 2$ (a), on 20,000 irregular sample points (b).

and the composition of the seafloor [42]. This system emits acoustic waves in a fan shape beneath its transceiver. MBES uses beamforming to extract directional information from the returning acoustic signals, producing a swath of depth readings from a single ping. The time it takes for the soundwaves to reflect off the seafloor and return to the receiver is used to calculate the water depth. If the sound speed in the full water column profile is known, the depth and position of the returned signals can be determined from the receive angle and the two-way travel time.

We employ the LSBSA method to approximate the seafloor bathymetric data using an analytical spline function. We consider the input to be one point cloud where the measured bathymetric data is translated to points represented by their u, v (horizontal coordinates), and z (vertical coordinate). Bathymetry data is normally projected onto their horizontal u and v-coordinates, and therefore we can parameterize the points by u and v coordinates and approximate the depth values (z) by a smooth function.

A data set was obtained from [20] in the Brown Bank, located in the North Sea between the UK and the Netherlands. The dominant bathymetric features of this area are sand banks (Brown Bank), sand waves, and megaripples. Megaripples, having a wavelength of ~ 15 m and ~ 1 high, are of interest in this study. Seafloor backscatter classification on megaripples has already been implemented in [43]. We focus here on the bathymetric modeling of the megaripples. The Brown Bank was surveyed by the Royal Netherlands Institute for Sea Research (NIOZ) vessel, the Pelagia, from 27 October to 03 November 2017. The data were collected by a Kongsberg EM 302 MBES system. Fig. 8 (a) shows a typical MBES bathymetric example of a megaripple in the North Sea. The area consists of $m = 1727$ depth measurements. The goal is to approximate this data set with the LSBSA explained in the previous sections.

This area was divided into 27×27 surface patches of size 3m $\times$ 3m, starting from 10 and ending to 91 in both u and v directions. This will then introduce $27 \times 27 = 900$ cubic B-spline coefficients to be estimated using the least squares method. The estimated standard deviation of the least squares fit is $\sigma = 5$cm. This standard deviation incorporates the noise of the measurements, the roughness of the seabed, and the presence of outliers. Outlier detection can be performed using the 'identification' step in DIA (Section 3.1). The w-test values of Eq. (18) is used to identify outliers.. The critical value at the 99.9 % confidence level is $Z_{0.0005} = 3.29$. Four residuals, having the largest w-test values of $w_{854} = 4.40$, $w_{1150} = -3.84$, $w_{811} = 3.48$ and $w_{859} = 3.37$ exceed the critical value and thus can be identified as outliers (Fig. 9). This also highlights the capability of the LSBSA method to perform statistical testing in point cloud approximation using the B-spline theory.

The estimated coefficients of the approximated surface are now used to predict the depth values on a regular grid, with a grid size of 75 cm in both u and v directions. The results are presented in Fig. 8 (b). Different features of megaripples including their orientation and direction, crest and trough, and other morphologic characteristics can simply be observed in the approximated surface. Such megaripples are the most dominant bathymetric feature in the North Sea. It is important to note that this surface has now a mathematical expression and therefore can be used for further analysis, interpretation, migration and monitoring of megaripples.

4.4. Approximation of laser scanner data

Permanent laser scanning (PLS), also called continuous laser scanning, is a new technique based on LIDAR (light detection and ranging) that is used to monitor beaches and their changing processes. Data sets of this technique, consisting of a series of point clouds with the spatial resolution of centimeters, make the detection of the beach deformation processes possible. Although coastal deformation can be caused by extreme weather events such as storms, the changes result from a storm on an urban beach cannot be analyzed without subsequent human activities. This is for example due to working with the bulldozers immediately after or during the storms [44]. Laser scanning point clouds data can be efficient in modeling the impact of weather events and their following anthropogenic activities on beaches.

A permanent laser scanning data set is selected within the Coast Scan project. The data set was collected by a Riegl VZ2000 laser scanner mounted on the roof of a hotel, at an urban beach in Kijkduin, the Netherlands, in January 2017 [44,45]. The files contain a three-dimensional point cloud in the u, v, and z coordinate system (in meters) for each day. The origin of the coordinate system is at the location of the laser scanner (u, v coordinates) and the height (z coordinate) corresponds to height above the sea level. We will look for the relative changes between scans, and therefore using the local coordinate system of the data set is sufficient for this application.

We selected an area of 40m $\times$ 30m from the coordinates of -155 to -115 in the u and 45 to 75 in the v direction. The chosen area is located at the bottom of the footpath. Between January 8 and 16, rough weather conditions caused the entrance to the path to become buried under sand. A bulldozer subsequently cleared the path, leaving a pile of accumulated sand [46]. A scan file on 16 January (end of stormy weather) and a scan file on 17 January (after working with a bulldozer) were selected. The file of 16 January contains 14,464 points scanned in the study area and that of the 17 January contained 15,233 points. The standard deviation of the measured elevations is reported to be below 2 cm (1 σ) for each of the individual data files. To fit a B-spline surface to the spatial variations

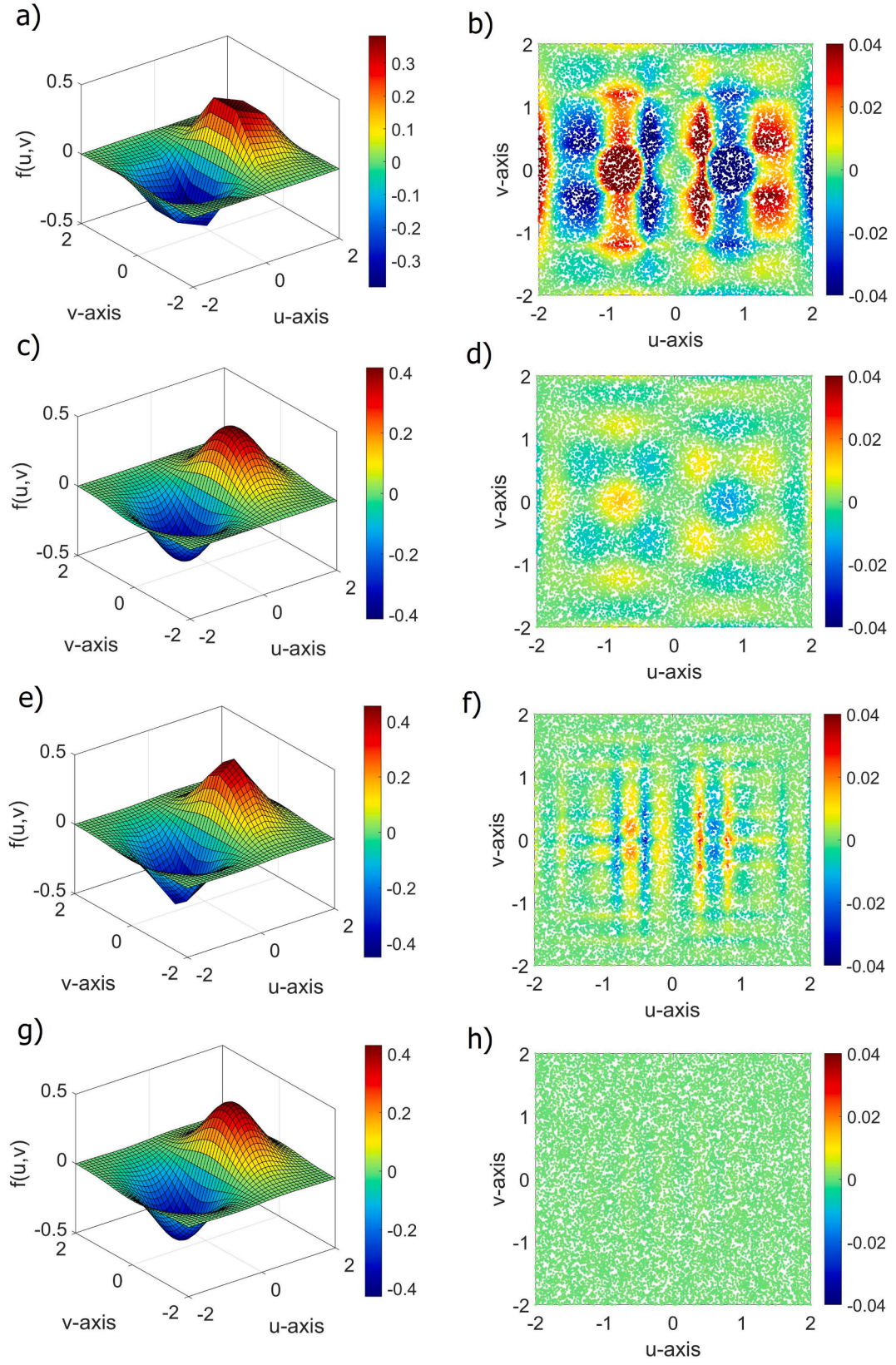

Fig. 7. Least squares approximation of function $f(u,v) = uexp(-u^2 - v^2)$ using linear (a) and cubic (c) B-splines on 25 surface cells of size 0.8×0.8 (scenario I), and linear (e) and cubic (g) B-splines on 100 surface cells of size 0.4×0.4 (scenario II). The rights plots (b, d, f and h) show the least squares residuals of the left frames.

Table 2
Statistics on approximation of the point cloud in Fig. 6b in two scenarios and two cases. Statistics include number of observations m, number of unknown parameters n, number of cells, and root mean squares error (RMSE).

Scenario/Case	m	No. unknown:n	No. cells	RMSE
I/1	20,000	36	25	0.022
I/2	20,000	64	25	0.004
II/1	20,000	121	100	0.005
II/2	20,000	169	100	0.0002

of the beach elevations, $40 \times 30 = 1,200$ surface patches of size 1m × 1m were formed. This introduces $43 \times 33 = 1419$ cubic B-spline coefficients, which were estimated for both the 16th and 17th of January. The estimated standard deviation of the least squares fit is $\sigma = 18$ mm for January 16th and $\sigma = 16$ mm for January 17th, which closely follow the nominal value of $\sigma = 2$ cm.

Having the estimated coefficients available, the elevations of knots on a regular grid size of 50 cm, in both u and v directions, were then estimated (Eq. (21). Fig. 10 (a) shows the beach elevation on the regular grid for January 16th and the beach elevation of January 17th is indicated in Fig. 10 (b). Comparing the two frames, the location of the sand accumulation after the work of the bulldozer in the study area is recognizable. The difference between the two fitted surfaces can mathematically model the effect of the high energy storm event and the subsequent human activities on the coast. This provides only preliminary results and can be the subject of further coastal research.

4.5. Approximation of a 3D mathematical manifold

This subsection demonstrates the application of LSBSA to deformation analysis using a known 3D mathematical function. Deformation analysis has applications in many fields including geoscience [47] and computer vision [48]. Consider the following 3D mathematical function:

$$f(u,v,t) = (1+0.05t)u\exp\left(-u^2-v^2\right) + 0.02uvt \tag{31}$$

which is a generalized form of Eq. (30) in which variable t can play the role of time. Equation (31) can express gradual temporal evolution (deformation) of Eq. (30); if $t = 0$ Eq. (30) will follow. It is rather difficult to visualize function $f(u,v,t)$ for three independent variables in 4D space. We just then need to present some frames on specific time instances t or we will use the colors as the fourth dimension.

A data set of $m = 10000$ points, generated at random 3D space–time locations is obtained using a uniform distribution over the cube $-2 \le u,v \le 2$ and $0 \le t \le 4$. This 4D data set is used to illustrate the efficacy of the proposed method (Fig. 11). The generated data is not contaminated with random noise. The goal is to approximate this data set using the least squares 3D spline theory proposed in Sections 2.4 and 2.5 (manifold approximation). The covariance matrix is considered as $Q_y = I_m$, where I_m is an identity matrix of size m. The standard deviation $\widehat{\sigma}$ of the fit is used as a measure to investigate the approximation error of the fitting problem. The cubic cells of the 3D manifold fit are considered to be of size 0.5 × 0.5 × 0.5, making in total 512 cells. The cubic B-splines are used to approximate the manifold. This will subsequently introduce $11\times11\times11 = 1331$unknown coefficients.

LSBSA is applied to the irregular data set in Fig. 11 to approximate the true mathematical function using the 3D B-splines. The estimated coefficients of the approximated surface are then used to predict the function values $f(u,v,t)$ on a regular grid $u = -2:0.1:2$ and $v = -2:0.1:2$, at four time instances $t = 0,1,2$and3. The results are presented in Fig. 12. The temporal evolution of the manifold can be observed in these frames. Given the above settings, with the fact that the standard deviation of the fit is $\widehat{\sigma} = 8\times10^{-4}$, it is indicated that the cubic spline does indeed provide appropriate approximation results. These results show that the proposed manifold approximation method can be used to study deformation analysis of many geoscience data series.

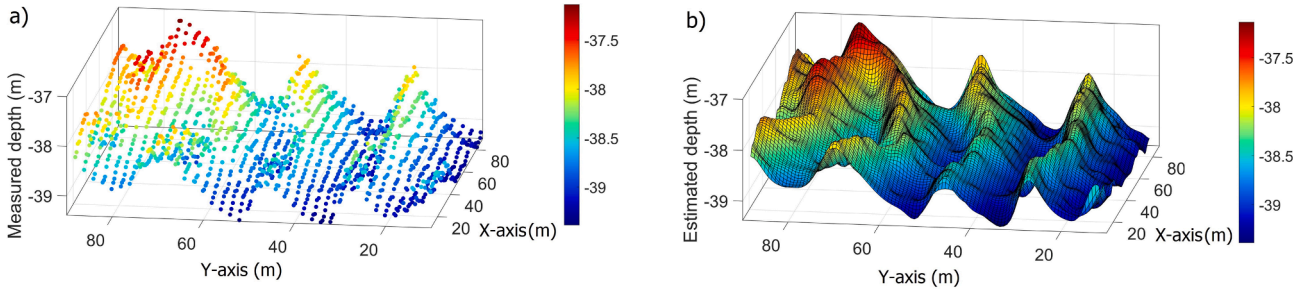

Fig. 8. MBES bathymetric data set of a megaripple in the North Sea (a), and its least squares approximation using cubic LSBSA (b); colorbar is in [m].

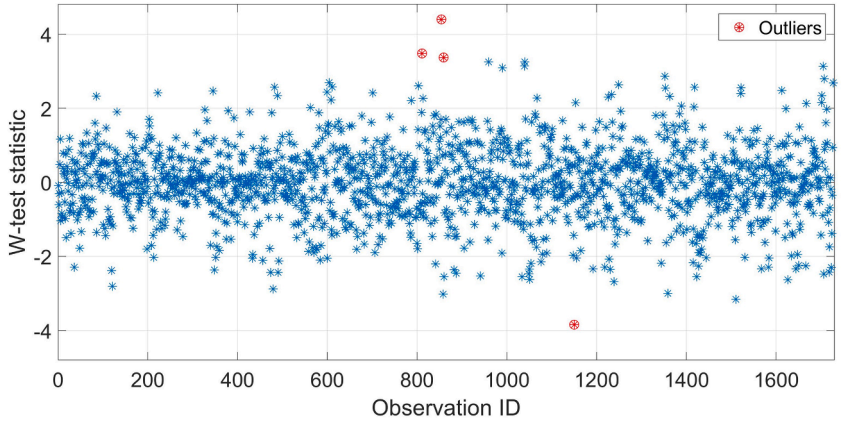

Fig. 9. W-test statistic of multi-beam echo-sounder observations using cubic LSBSA, red dots are outlying observations identified using the w-test statistic in the DIA process. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

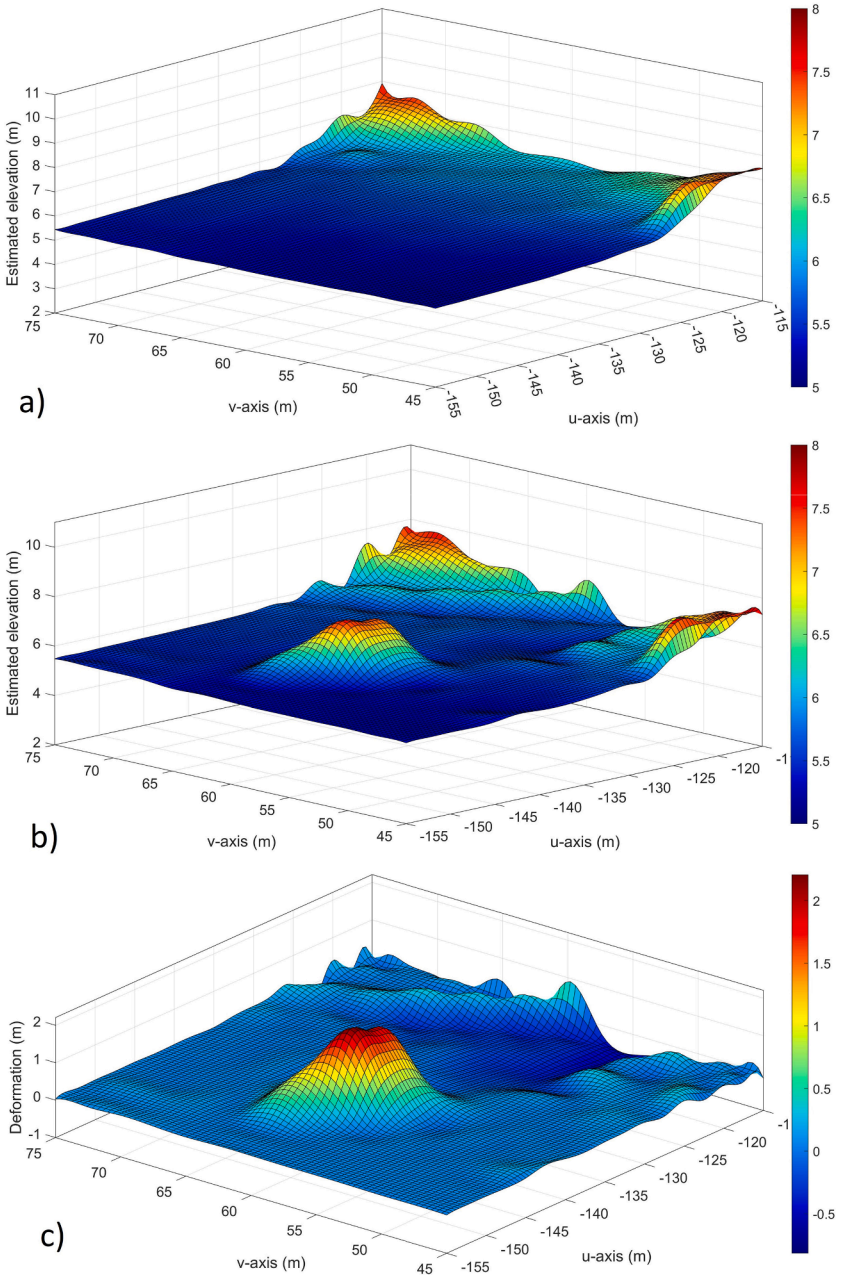

Fig. 10. Approximated permanent laser scanning point cloud data of Kijkduin beach elevation for January 16th 2017 (a) and January 17th 2017 (b), after bulldozer work, and their difference (c); colorbar is in [m].

4.6. Approximation of LWE-thickness of Greenland ice sheet

A novel application of 3D LSBSA is the ice sheet monitoring using satellite data. The Greenland ice sheet (GrIS) and its mass change patterns, which significantly contribute to the global sea level rise, have gained the attention of researchers in recent years [49]. The gravitational recovery and climate experiment (GRACE) and its follow-on mission, GRACE-FO, provide essential data in this field, offering time series of Greenland Ice Sheet (GrIS) mass changes. These data are presented globally through mass concentration (mascons) blocks on monthly basis. This indicates that when working with GRACE mission data, we analyse *mass change* parameters with specific spatial and temporal resolutions in 3D.

To explore the application of LSBSA to 4D point cloud data, we apply the method to JPL-RL06M-v03 data [50]. This dataset provides numerical values of Equivalent Water Thickness (EWT) parameters at specific latitudes and longitudes on monthly basis. EWT is a way to represent mass anomalies on the earth surface. An increase of $+1$ cm of this parameter for a mascon of size (0.5×0.5) is like adding mass equal to water with a thickness of one centimetre all over the mascon. To demonstrate the effectiveness of LSBSA in capturing seasonal changes in GrIS mass, we analysed monthly data from 2021 to 2022 (24 months) within the longitudinal and latitudinal range of $12-74$ [deg] west and $59-85$ [deg], respectively.

The EWT parameters were extracted for a total of $m = 138240$ points across the studied area over this time span. The number of $31 \times 13 \times 9 = 3584$ 3D manifold cubic cells of $2(\text{deg}) \times 2(\text{deg}) \times 3(\text{moths})$ size were formed, resulting in $n = 6528$ coefficients. The design matrix A, and hence the coefficients of the B-splines were then estimated for the ice sheet manifold using LSBSA. This generates a 3D manifold implicitly defined at any desired 2D geographical location and time instant. This allows to predict EWT thickness parameter values at regular meshes of

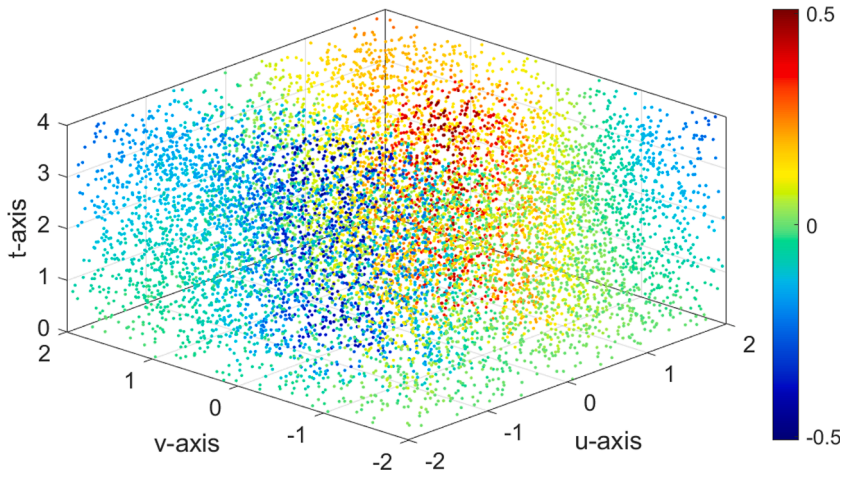

Fig. 11. A data set of $m = 10000$ 3D space–time points generated randomly from Eq. (31) using a uniform distribution over the cube $-2 \le u, v \le 2$ and $0 \le t \le 4$. The colors show function values $y = f(u, v, t)$.

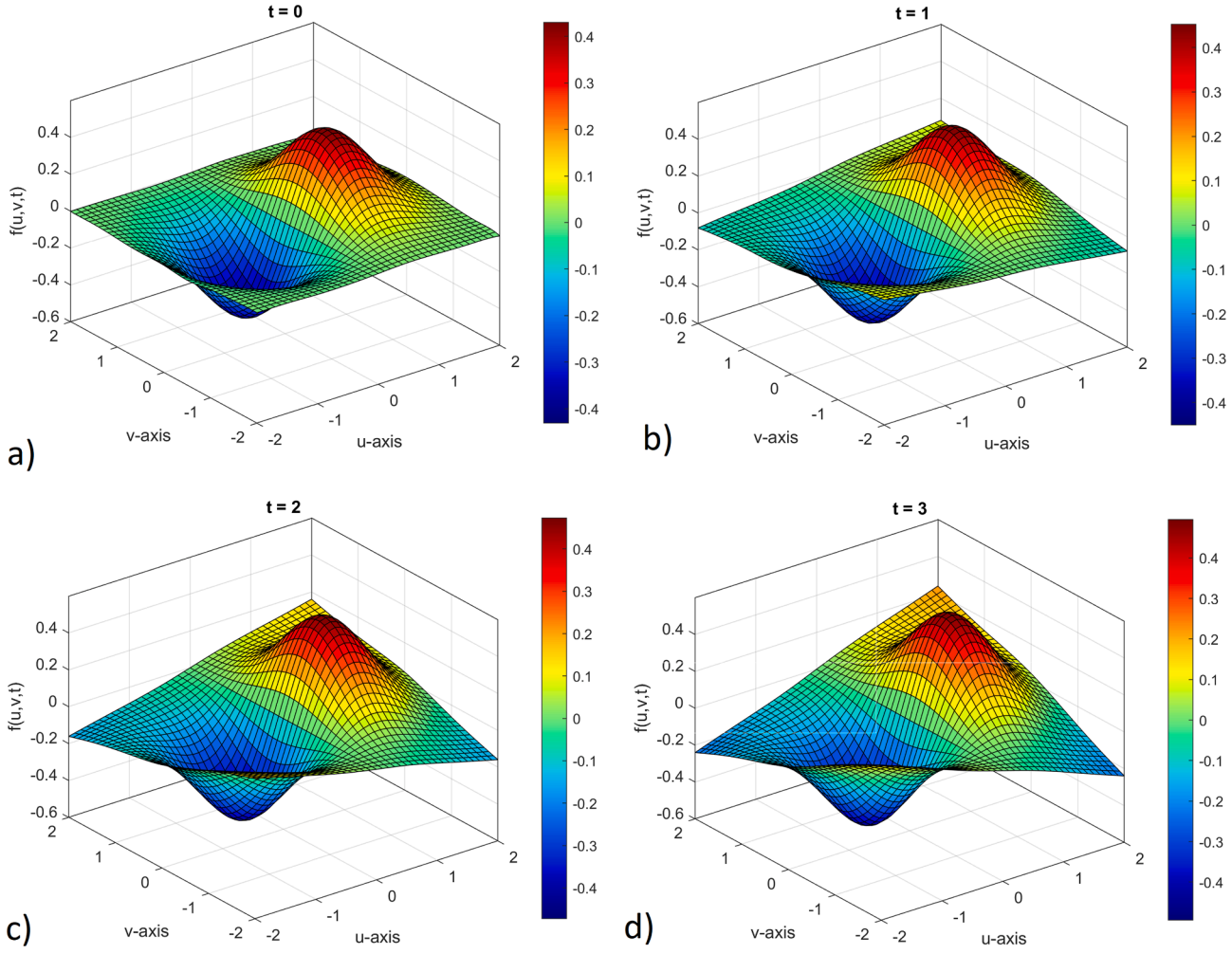

Fig. 12. Least squares approximation of the function $f(u, v, t) = (1 + 0.05t)uexp(-u^2 - v^2) + 0.02uvt$ using cubic B-splines on 512 cubic patches of size 0.5×0.5×0.5. Results are presented at four time instances $t = 0$ *(a)*, $t = 1$ *(b)*,$t = 2$ *(c)*, and $t = 3$ *(d)*.

any location and time, monitoring spatial and temporal changes in the GrIS effectively. The EWT values were determined at a regular grid of 0.5[deg] × 0.5[deg]. The Greenland grids were then cut from the analysed rectangular area by an irregular mask.

The frames in Fig. 13 illustrate changes in this parameter for the months January (winter), April (spring), July (summer), and October (autumn) in both 2021 and 2022. Based on Fig. 13, during warmer months (April and July), there is a decrease in mass change rate compared to colder months (January and October) in the Greenland Ice Sheet (GrIS). The mass change patterns differ between the southern/

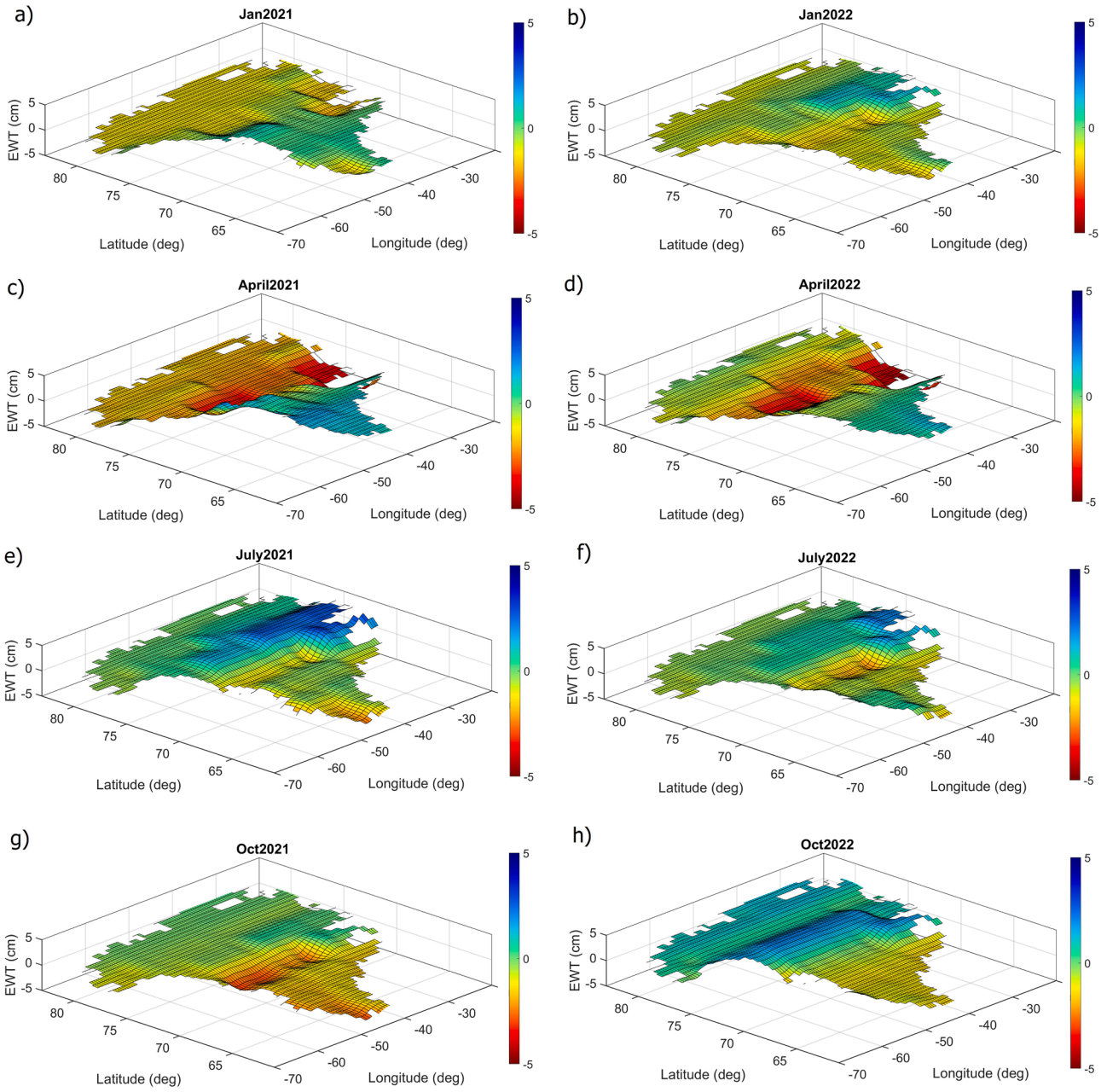

Fig. 13. Estimated seasonal changes of EWT related to the Greenland ice sheet using GRACE missions ice mass data. From top to bottom, each frame shows variations across January (winter, a and b), April (spring, c and d), July (summer, e and f), and October (autumn, g and h) for both 2021 (left) and 2022 (right).

central regions and the northern regions of the ice sheet. Previous research has noted these geographical variations, leading to the subdivision of the GrIS into smaller subregions [51–53]. One of the main advantages of using LSBSA in this application is its high spatial and temporal resolution in the study of ice mass changes pattern. The 2022 Arctic report indicates a decreasing trend in GrIS ice mass changes from 1 September 2021, to mid-August 2022, which is supported by data from October 2021, January 2022, and April 2022 shown in Fig. 13. The consistent reflection of ice melting changes validates the effectiveness of LSBSA for analyzing temporal and spatial variations using the manifold fitting. Geophysical researchers may consider utilizing this method to monitor (ice) mass changes using GRACE missions for many geoscience applications.

As a final remark, we evaluate the computational burden of estimating the B-spline coefficients on a computer equipped with an Intel Core i7-10610U CPU @ 1.80 GHz (2.3 GHz, 4 cores) and 16 GB RAM. Under the described settings, the computation involves $31 \times 13 \times 9 = 3584$ 3D manifold cubic cells, each with dimensions of 2(deg) $\times$ 2(deg) $\times$ 3(month), resulting in estimating $n = 6528$ coefficients. The computational time required to estimate $\widehat{x}$ using Eq. (13) is 351 s. In an alternative setup, the computation involves $16 \times 7 \times 5 = 560$ 3D manifold cubic cells, each with dimensions of 4(deg) $\times$ 4(deg) $\times$ 6 (month), resulting in $n = 1520$ coefficients. The computational time required to estimate $\widehat{x}$ reduces to approximately 8 s. These results align well with the overall computational complexity of $\mathcal{O}(n^3)$, as explained earlier.

5. Concluding remarks

We presented a simple, flexible and attractive least squares B-spline approximation (LSBSA) method, which has its own strengths and weaknesses. LSBSA is simple because it uses the well-known linear least

squares theory to approximate point cloud data. The method can be applied to estimate/interpolate data points on a regular grid. The method is flexible because it can approximate 2D, 3D and 4D point clouds, with the possibility of extension to higher dimensions when more independent variables are involved. LSBSA is not restricted to cubic splines, but is defined for any degree. The formulation of LSBSA in the standard least squares framework is attractive, because it enables to directly employ the extensive existing least squares theory to B-spline approximation. Notably, our formulation provides the best linear unbiased estimation (BLUE) for linear models, and directly incorporates several quality control measures. Hypothesis testing can be used to identify outlying observations while the error propagation law can be applied to obtain the covariance matrix of the interpolated variables. These important cases were presented, more cases need to be developed. In particular, robust estimation methods, variance component estimation and inclusion of the approximation error in error analysis are directions for further research and implementation in related fields.

There are also weaknesses and challenges to be addressed. For LSBSA, the continuity constraints should be specified on forehand and hence cannot be defined by the user. Particularly, LSBSA provides B-splines that are continuous and have continuous derivatives up to and including $p-1$, as is common for B-splines. This can be too constrained in some application fields. An alternative way is to use for example a cubic Hermite spline where each third-order polynomial piece has continuous values and first derivatives at the knots (so the second derivative is not continuous). Also, the randomness of interpolated variables were ignored in this contribution. An enhanced formulation can deal with 'prediction' rather than 'estimation', which includes the stochastic characteristics of point clouds to improve prediction. This will also affect the quality control measures such as the precision of the interpolated variables.

LSBSA offers new research topics in geoscience point cloud approximation. The method is a generalized version of the spline interpolation method and can be applied to irregularly scattered point cloud data at knots specified by the user. The performance of the method in linear, quadratic, cubic or quartic spline function was investigated. A few simulated and real data sets were used to illustrate the efficacy of the proposed theory. We briefly demonstrated its applications to multi-beam echo-sounder data, laser scanning data, and digital terrain modeling using LIDAR data. Future research will show how this technique can become fully operational for these applications.

We also demonstrated the application of LSBSA to 3D manifold approximation problems (4D irregular point clouds). This application has significant impact to study temporal variations of a 2D surface. It is particularly applicable when the function variables encompass geographical positions or image pixels plus time. Most deformation analysis methods in the manifold fitting problem typically examine surface changes from a 2D perspective. Additionally, the fitted polynomials often exhibit discontinuities at the boundaries of surface elements, i.e. triangles in traditional methods and meshes in recently developed methods. LSBSA offers two key advantages in deformation analysis and manifold fitting. Firstly, this method considers surface parameters and their changes in the 3D space, avoiding the simplification of complex deformation problems to two dimensions. Secondly, the use of B-spline functions in LSBSA maintains the continuity of functions and their derivatives at the boundaries of surface cells, preserving continuity of changes in spatial and temporal dimensions. The LSBSA method was utilized to estimate changes in EWT in the Greenland ice sheet. The results were consistent with the geophysical information of the region in 2022, which shows the capability of LSBSA in analysing temporal and spatial variations.

CRediT authorship contribution statement

Alireza Amiri-Simkooei: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Formal analysis, Conceptualization. **Fatemeh Esmaeili:** Writing – review & editing, Visualization, Software, Formal analysis. **Roderik Lindenbergh:** Writing – review & editing, Validation, Investigation, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] F. Remondino, From point cloud to surface: the modeling and visualization problem, International Archives of the Photogrammetry, Rem. Sens. Spatial Informat. Sci. 34 (2003).
[2] A.M. Araújo, M.M. Oliveira, A robust statistics approach for plane detection in unorganized point clouds, Pattern Recogn. 100 (2020) 107115.
[3] Y. Zhou, R. Chen, Y. Zhao, X. Ai, G. Zhou, Point cloud denoising using non-local collaborative projections, Pattern Recogn. 120 (2021) 108128.
[4] P. He, X. Xu, X. Chang, J. Ding, S. Chen, Multi-manifold discriminant local spline embedding, Pattern Recogn. 129 (2022) 108714.
[5] P.J. Teunissen, Adjustment theory: an introduction, Delft University of Technology, VSSD, 2003.
[6] C. Conti, L. Romani, D. Schenone, Semi-automatic spline fitting of planar curvilinear profiles in digital images using the Hough transform, Pattern Recogn. 74 (2018) 64–76.
[7] A. Amiri-Simkooei, M. Hosseini-Asl, A. Safari, Least squares 2D bi-cubic spline approximation: theory and applications, Measurement (2018).
[8] C. Harmening, C. Hobmaier, H. Neuner, Laser scanner–based deformation analysis using approximating B-spline surfaces, Remote Sens. (Basel) 13 (2021) 3551.
[9] C. Harmening, H. Neuner, A constraint-based parameterization technique for B-spline surfaces, J. Appl. Geodesy 9 (2015) 143–161.
[10] W. Yu, M. Tannast, G. Zheng, Non-rigid free-form 2D–3D registration using a B-spline-based statistical deformation model, Pattern Recogn. 63 (2017) 689–699.
[11] X. Li, M.A. Scott, Analysis-suitable T-splines: characterization, refineability, and approximation, Math. Models Methods Appl. Sci. 24 (2014) 1141–1164.
[12] Y. He, H. Qin, Surface reconstruction with triangular B-splines, Geometric Modeling and Processing, 2004. Proceedings, IEEE, 2004, pp. 279-287.
[13] J. Gu, T. Yu, T.-T. Nguyen, Y. Yang, T.Q. Bui, Fracture modeling with the adaptive XIGA based on locally refined B-splines, Comput. Methods Appl. Mech. Eng. 354 (2019) 527–567.
[14] F. Patrizi, T. Dokken, Linear dependence of bivariate Minimal Support and Locally Refined B-splines over LR-meshes, Comput. Aided Geom. Des. 77 (2020) 101803.
[15] I.J. Schoenberg, Contributions to the problem of approximation of equidistant data by analytic functions. Part B. On the problem of osculatory interpolation. A second class of analytic approximation formulae, Q. Appl. Math. 4 (1946) 112–141.
[16] C. De Boor, On calculating with B-splines, J. Approx. Theory 6 (1972) 50–62.
[17] L. Pagani, J. Wang, B.M. Colosimo, X. Jiang, P.J. Scott, Fast hierarchical fusion model based on least squares B-splines approximation, Precis. Eng. 60 (2019) 570–586.
[18] C. Harmening, Spatio-temporal deformation analysis using enhanced B-spline models of laser scanning point clouds, Wien, 2020.
[19] F. Zangeneh-Nejad, A. Amiri-Simkooei, M. Sharifi, J. Asgari, Cycle slip detection and repair of undifferenced single-frequency GPS carrier phase observations, GPS Solutions 21 (2017) 1593–1603.
[20] A.R. Amiri-Simkooei, L. Koop, K.J. van der Reijden, M. Snellen, D.G. Simons, Seafloor characterization using multibeam echosounder backscatter data: methodology and results in the North Sea, Geosciences 9 (2019) 292.
[21] M. Hosseini-Asl, A. Amiri-Simkooei, A. Safari, Establishment of a corrective geoid surface by spline approximation of Iranian GNSS/levelling network, Measurement 111341 (2022).
[22] I. Galaktionov, A. Nikitin, J. Sheldakova, A. Kudryashov, B-spline approximation of a wavefront measured by Shack-Hartmann sensor, Laser Beam Shaping XXI, SPIE, 2021, pp. 143-146.
[23] L. Seifert, H.J. Tiziani, W. Osten, Wavefront reconstruction algorithms for the adaptive Shack-Hartmann sensor, Optical Measurement Systems for Industrial Inspection IV, SPIE, 2005, pp. 544-553.
[24] I. Sergievskaya, S. Royo, M. Ares, B-spline basis for adaptive piezoelectric mirror shape reconstruction, Proc. of 7th AOIM Workshop (2009).
[25] T. Dokken, T. Lyche, K.F. Pettersen, Polynomial splines over locally refined box-partitions, Comput. Aided Geom. Des. 30 (2013) 331–356.
[26] V. Skytt, O. Barrowclough, T. Dokken, Locally refined spline surfaces for representation of terrain data, Comput. Graph. 49 (2015) 58–68.
[27] C. Bracco, C. Giannelli, F. Patrizi, A. Sestini, Local spline refinement driven by fault jump estimates for scattered data approximation, Math. Comput. Simul 228 (2025) 103–123.

[28] G. Kermarrec, V. Skytt, T. Dokken, Optimal surface fitting of point clouds using local refinement: Application to GIS data, Springer Nature, 2023.
[29] G. Zhang, Y. Yang, H. Wang, Improved numerical manifold method (iNMM)—An extra-DOF free and interpolating NMM with continuous nodal stress, Eng. Anal. Bound. Elem. 84 (2017) 117–128.
[30] G. Sun, Y. Yang, S. Cheng, H. Zheng, Phreatic line calculation and stability analysis of slopes under the combined effect of reservoir water level fluctuations and rainfall, Can. Geotech. J. 54 (2017) 631–645.
[31] Y. Yang, W. Wu, H. Zheng, X. Liu, A high-order three dimensional numerical manifold method with continuous stress/strain field, Eng. Anal. Bound. Elem. 117 (2020) 309–320.
[32] R.H. Bartels, J.C. Beatty, B.A. Barsky, An introduction to splines for use in computer graphics and geometric modeling, Morgan Kaufmann, 1995.
[33] P.J. Teunissen, Testing theory, VSSD Delft (2006).
[34] W. Baarda, A testing procedure for use in geodetic networks, Tech. rep. Pub. on Geod., New Series 977 (1968) 2.
[35] P.J. Teunissen, Distributional theory for the DIA method, J. Geod. 92 (2018) 59–80.
[36] A. Amiri-Simkooei, Least-squares variance component estimation: theory and GPS applications, Mathematical Geodesy and Positioning, Delft University of Technology, Delft, the Netherlands, 2007.
[37] A.N. Tihonov, Solution of incorrectly formulated problems and the regularization method, Soviet Math. 4 (1963) 1035–1038.
[38] P.C. Hansen, The L-curve and its use in the numerical treatment of inverse problems, (1999).
[39] A. Nurunnabi, F. Teferle, R. Lindenbergh, J. Li, S. Zlatanova, Robust approach for urban road surface extraction using mobile laser scanning 3D point clouds, Int. Archiv. Photogramm., Rem. Sens. Spatial Inf. Sci. 43 (2022) 59–66.
[40] A. Amiri-Simkooei, Formulation of L1 norm minimization in Gauss-Markov models, J. Surv. Eng. 129 (2003) 37–43.
[41] A. Khodabandeh, A. Amiri-Simkooei, M. Sharifi, GPS position time-series analysis based on asymptotic normality of M-estimation, J. Geod. 86 (2012) 15–33.
[42] X. Lurton, An introduction to underwater acoustics: principles and applications, Springer, 2002.
[43] L. Koop, A. Amiri-Simkooei, K.J. van der Reijden, S. O'Flynn, M. Snellen, D. G. Simons, Seafloor classification in a sand wave environment on the Dutch continental shelf using multibeam echosounder backscatter data, Geosciences 9 (2019) 142.
[44] E.D. Lazarus, E.B. Goldstein, Is there a bulldozer in your model? J. Geophys. Res. Earth 124 (2019) 696–699.
[45] S. Vos, K. Anders, M. Kuschnerus, R. Lindenbergh, B. Höfle, S. Aarninkhof, S. de Vries, A high-resolution 4D terrestrial laser scan dataset of the Kijkduin beach-dune system, The Netherlands, Scientific Data 9 (2022) 1–11.
[46] K. Anders, R. Lindenbergh, S. Vos, H. Mara, S. De Vries, B. Höfle, High-frequency 3D geomorphic observation using hourly terrestrial laser scanning data of a sandy beach, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. 4 (2019) 317–324.
[47] D. Jia, W. Zhang, Y. Liu, Systematic approach for tunnel deformation monitoring with terrestrial laser scanning, Remote Sens. (Basel) 13 (2021) 3519.
[48] H. Zhou, Z. Xu, Y. Tian, Z. Yu, Y. Zhang, J. Ma, Interpolation-based nonrigid deformation estimation under manifold regularization constraint, Pattern Recogn. 128 (2022) 108695.
[49] S. Coulson, S. Dangendorf, J.X. Mitrovica, M.E. Tamisiea, L. Pan, D.T. Sandwell, A detection of the sea level fingerprint of Greenland Ice Sheet melt, Science 377 (2022) 1550–1554.
[50] J. RL06, Monthly Mass Grids - Global mascons (JPL RL06), 2024,.
[51] W. Li, C. Shum, F. Li, S. Zhang, F. Ming, W. Chen, B. Zhang, J. Lei, Q. Zhang, Contributions of greenland GPS observed deformation from multisource mass loading induced seasonal and transient signals, Geophys. Res. Lett. 47 (2020) e2020GL088627.
[52] K.D.M. Moon T.A., R.S. Fausto, X. Fettweis, B.D. Loomis, T.L. Mote, K. Poinar, M. Tedesco, A. Wehrle, C.D. Jensen, Greenland Ice Sheet, Arctic Report Card 2022, Series: NOAA technical report OAR ARC, 22-05, 2022 (2022) 32.
[53] E. Rignot, J. Mouginot, Ice flow in Greenland for the international polar year 2008–2009, Geophys. Res. Lett. 39 (2012).