

# SURFACE APPROXIMATION OF COASTAL REGIONS: LR B-SPLINE FOR DETECTION OF DEFORMATION PATTERN

Gaël Kermarrec<sup>1\*</sup>, Vibeke Skytt<sup>2</sup>, Tor Dokken<sup>2</sup>

<sup>1\*</sup> Geodetic Institute, Leibniz University Hannover, Nienburgerstr. 1, 30167 Hannover, Germany

[kermarrec@gih.uni-hannover.de](mailto:kermarrec@gih.uni-hannover.de) (corresponding author)

<sup>2</sup> SINTEF, P.O. Box 124 Blindern, 0314 Oslo, Norway

Commission II, WG II/3

**KEY WORDS:** multilevel B-spline approximation, GIS data set, LR B-spline surfaces

## ABSTRACT:

Geospatial data acquisition of terrains produces huge, noisy and scattered point clouds. An efficient use of the acquired data requires structured and compact data representations. Working directly in a point cloud is often not appealing. To face this challenge, approximation with tensor product B-spline surfaces is attractive. It reduces the point cloud description to relatively few coefficients compared to the volume of the original point cloud. However, this representation lacks the ability to adapt the resolution of the shape to local variations in the point cloud. The result is frequently that noise is approximated and that surfaces have unwanted oscillations. Locally Refined (LR) B-spline surfaces were introduced to face this challenge and provide a tool for approximating Geographic Information System point clouds. In our LR B-spline based approximation algorithm, iterative least-squares approximation is combined with a Multilevel B-spline Approximation to reduce memory consumption. We apply the approach to data sets from coastal regions in Norway and the Netherlands, and compare the obtained approximation with a raster method. We further highlight the potential of LR B-spline volumes for spatio-temporal visualisation of deformation patterns.

## 1. INTRODUCTION

Parametric spline surface approximation is an important step in reverse engineering to convert real object data to a computer-aided design model (Raja and Fernandes, 2008), or to perform statistical testing for deformation based on the estimated parameters (Kermarrec et al., 2020). The surface fitting begins by recording a point cloud of an object or scene with, e.g., a contactless sensor such as the Terrestrial Laser Scanner (TLS). Surface approximation techniques of point clouds from Geographic Information System (GIS) data sets can be divided into (i) non-adaptive methods, for which the optimal surface is globally adapted in the approximation process, and (ii) locally adaptive methods. For the latter, each new iteration depends on the result of the previous approximation iteration (Wang 2009). Such procedures save computation time as the number of parameters to estimate is less high than for (i). The widely used adaptive surface fitting strategies using B-splines belongs to (ii), but is limited by the tensor product formulation of the non-uniform rational B-splines (NURBS): it does not allow for local refinement and leads potentially to overfitting where no further refinement would have been necessary. There are three main approaches for extending spline surfaces to support local refinement, see Dokken et al. (2019):

- **Hierarchical B-splines** (HB-splines) were introduced in Forsey and Bartels (1988). Here tensor product B-splines on the coarser level are removed and B-splines at the finer level added in such a way that linear independence is guaranteed.

- **T-splines** are described, e.g., in Sederberg et al. (2003). They denote a class of locally refined splines. The starting point for T-spline refinement is a tensor product B-spline surface with control points and meshlines (initial T-mesh) with assigned knot values. The refinement is performed by successively adding new control points in-between two adjacent control points in the T-mesh. They were used in a geodetic context by Kermarrec et al. (2020) to approximate a bridge.

- **LR B-splines** were developed by Dokken et al. (2009). The refinement is performed successively by inserting axis parallel meshlines in the mesh of knotlines. Each meshline inserted has to split the support of at least one tensor product B-spline. In this contribution, we will focus on this approximation strategy which was shown to be flexible for approximating various point clouds (Skytt and Dokken, 2021). A toolbox called Gotools (Dokken et al., 2013) is freely available under <https://github.com/SINTEF-Geometry/GoTools/wiki/Module-LRSplines2D>. Interoperability with existing GIS-technology is essential for the deployment of LR B-spline within GIS: LR B-spline surfaces can be converted to standard formats used in GIS-systems. They are, thus, particularly attractive within a GIS context where huge, noisy and scattered point clouds are common. Furthermore, LR B-spline volume is a promising tool for visualising and analysing spatio-temporal deformations, without having to manipulate large point clouds.

The starting point of iterative surface approximation with LR B-spline is a tensor product B-spline surface. In the LR B-spline surface, extra degrees of freedom are inserted locally, where needed. The desired smoothness is maintained, and the growth in data volume is kept under control. The least square (LS) norm between the parametrised point cloud and the parametric surface is to be minimised. For gridded data, this approach is called surface skinning (Woodward, 1988). The LS method may reach its limits when the number of refinement iterations increases. In that case, it is convenient to use the Multilevel B-spline Approximation (MBA) developed by Lee et al. (1997) after a few initial step with the LS approximation. The MBA is based on a local approximation for which no equation system needs to be solved. The residuals of the data points obtained from the last fitted surface are recursively approximated using finer meshes. This procedure saves storage and was shown to be advantageous in case of scattered and/or non-gridded containing data gaps or outliers occur (Zhang et al. 1998; Skytt et al. 2015).

In this contribution, we propose to introduce the LR B-spline and the MBA. To relate LR B-spline to surface representations

already in use in GIS, we will compare our approximation results with raster representations and show the potential of LR B-spline volume for spatio-temporal deformation analysis of GIS dataset. We focus on point clouds from a bathymetry data set in Norway and a coastal region in the Netherlands, which are rather smooth and without edges.

## 2. MATHEMATICAL BACKGROUND

In this section, we propose to explain shortly what LR B-spline surfaces are starting from the concept of tensor product B-spline surfaces. The iterative approximation and principle of local refinement are shortly adressed. We finally focus on describing the combination between LS and MBA for iterative surface approximation.

### 2.1 Surface approximation using LR B-spline

#### 2.1.1 LR B-spline surfaces

Given a non-decreasing sequence  $\mathbf{u} = (u_0, u_1, \dots, u_{p+1})$  we define a B-spline  $B[\mathbf{u}]: \mathbb{R} \rightarrow \mathbb{R}$  of degree  $p \geq 0$  recursively as follows:

$$B[\mathbf{u}](u) := \frac{u - u_0}{u_p - u_0} B[u_0, \dots, u_p](u) + \dots$$

$$\frac{u_{p+1} - u}{u_{p+1} - u_1} B[u_1, \dots, u_{p+1}](u),$$

(1)

starting with

$$B[u_i, u_{i+1}](u) := \begin{cases} 1; & \text{if } u_i \leq u < u_{i+1}; \\ 0; & \text{otherwise,} \end{cases} \quad i = 0, \dots, p.$$

We have  $B[\mathbf{u}] \equiv 0$  if  $u_{p+1} = u_0$ ; terms with zero denominator are defined to be zero.

A univariate spline space can be defined by a polynomial degree  $p$  and a knot vector, where the knots satisfy:  $u_{i+1} \geq u_i$ ,  $i = 0, \dots, N + p - 1$ , and  $u_{i+p+1} > u_i$ ,  $i = 0, \dots, N - 1$ , i.e. a knot value can be repeated  $p + 1$  times. The number of times a knot value is repeated is called the multiplicity  $m$  of the knot value. The continuity across a knot value of multiplicity  $m$  is  $C^{p-m}$ .

A basis for the univariate spline space above can be defined in many ways. However, the approach most often used is the univariate minimal support B-spline basis. In that case, the B-splines are defined by selecting  $p + 2$  consecutive knots from  $\mathbf{u}$ , starting from the first knot. So

$$B_{i,p}(u) := B[u_i, \dots, u_{i+p+1}](u)$$

is defined by the knots  $u_i, \dots, u_{i+p+1}$ ,  $i = 0, \dots, N - 1$ .

Given two non-decreasing knot sequences  $\mathbf{u} = \{u_0, u_1, \dots, u_{N_1+p_1}\}$  and  $\mathbf{v} = \{v_0, v_1, \dots, v_{N_2+p_2}\}$  where  $p_1 \geq 0$  and  $p_2 \geq 0$ , we define a bivariate tensor product B-spline

$B_{i,j,p_1,p_2}: \mathbb{R}^2 \rightarrow \mathbb{R}$  from the two univariate B-splines

$$B_{i,p_1}(u) \text{ and } B_{j,p_2}(v) \text{ by:}$$

$$B_{i,j,p_1,p_2}(u, v) := B_{i,p_1}(u) B_{j,p_2}(v). \quad (2)$$

The support of  $B$  is given by the cartesian product

$$\text{supp}(B_{i,j,p_1,p_2}) := [u_i, u_{i+p_1+1}] \times [v_j, v_{j+p_2+1}].$$

A bivariate tensor product spline space is made by the tensor product of two univariate spline spaces. Assuming that both univariate spline spaces have a minimal support B-spline basis, the minimal support basis for the tensor product B-spline space is constructed by making all tensor product combinations of the B-splines of the two bases. The minimal support basis for this spline space contains the tensor product B-splines

$$B_{i,j,p_1,p_2}(u, v), \quad i = 0, \dots, N_1 - 1, \quad j = 0, \dots, N_2 - 1.$$

As in the univariate case, the basis has useful properties such as non-negativity and partition of unity. Figure 1 illustrates how a tensor product B-spline with the knot vector  $\{0, 1, 2, 3\}$  in both parameter directions looks like.

Spline surfaces are frequently represented using a bivariate minimal support tensor product B-spline basis.

$$F(u, v) = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} c_{i,j} B_{i,j,p_1,p_2}(u, v) \dots \quad (3)$$

$$(u, v) \in [u_{p_1}, u_{N_1}] \times [v_{p_2}, v_{N_2}].$$

Here  $c_{i,j} \in \mathbb{R}^d$ ,  $i = 0, \dots, N_1 - 1$ ,  $j = 0, \dots, N_2 - 1$  are the surface coefficients and  $d$  is the dimension of the geometry space.

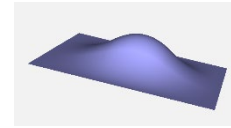


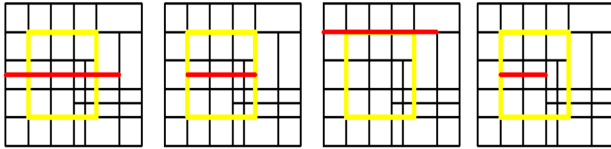
Figure 1: Example of a bi-quadratic tensor product B-spline basis functions.

#### 2.1.2 Local refinement

NURBS surfaces can be refined by an axis parallel meshline that crosses the complete parameter domain. LR B-spline can be considered as extension of NURBS, through insertion of shorter meshlines to refine the mesh locally. LR B-spline provides a richer set of refinements than HB-splines and T-splines according to Dokken et al. (2015). When we insert a meshline, at least one local B-spline should be split into two parts. The meshline must start and stop on meshlines in the other direction.

In a first step, all tensor product B-splines completely traversed by that meshline are identified. Then these B-splines are split into two B-splines. Duplicates are merged to clean the data structure. In the volumetric case, meshlines are replaced by mesh rectangles.

Figure 2 shows the mesh of an LR B-spline surface. The support of one of the B-splines is illustrated by a yellow box. In a) the meshline goes from the boundary and across the yellow support, and continues on the other side. The support of more than one B-spline is traversed, implying that several B-splines will be split. In b) the support of exactly one B-spline is split. In c) the meshline transverses the edge of the yellow support and will not result in refinement of this tensor product B-spline. However, other tensor product B-splines are affected since the meshline is on top of an existing meshline: the continuity of the surface crossing the line is decreased. The meshline in d) is too short to trigger refinements.



If after this insertion, some tensor product B-spline still do not have minimal support, the refinement process will continue until all the tensor product B-splines have minimal support.

Line traversing the interior of B (a) Line traversing the interior of B (b) Line traversing the edge of B (c) Meshline too short (d)

**Figure 2:** Refinement process by inserting a new meshline.

### 2.1.3 Iterative surface approximation using LS and MBA

The LR B-spline surface is iteratively refined using the principle of 2.1.2.

*Starting point:* The starting point of the iterative surface approximation is a tensor product B-spline surface defined over a coarse mesh. A refinement is performed iteratively in the mesh cells, where one observation is associated with an error term higher than a given threshold  $TH$ . With a large threshold, few cells are refined whereas with a low threshold  $TH$ , the result will be a global refinement corresponding to the NURBS. The choice of  $TH$  depends on the level of accuracy needed, balanced by the computation time and number of surface coefficients. A low threshold may lead to a noise fitting and should be avoided. It could result in ripples on the surface that are not the consequence of real pattern.

Let  $F$  be the LR B-spline surface  $F(x, y) = \sum_{i=1}^n P_i B_i(x, y)$

Here  $P_i$  are the surface coefficients and  $B_i$  are tensor product B-spline scaled to form a partition of unity.

*LS approximation:* LS approximation is a global method where the following expression is minimized with respect to the surface coefficients  $P_i$  over the entire surface domain:

$$\min_P \left[ \alpha_1 J(F(x, y)) + \alpha_2 \sum_{k=1}^K (F(x_k, y_k) - z_k)^2 \right].$$

Here  $\mathbf{x} = (x_k, y_k, z_k), k = 1, \dots, K$  are the data points. The expression is differentiated and turned into a linear, sparse equation system in the number of surface coefficients to estimate. A LS approximation method will result in a singular equation system if there exists B-splines with no data points in its support. As an LR B-spline surface is defined on a rectangular domain and a typical point cloud has a non-rectangular outline and may contain holes, parts of the surface domain will frequently lie outside the domain of the point cloud. To face that challenge, we add  $J(F(x, y))$ , which is a smoothness term that enables a non-singular equation system even if this situation occurs. In our case, the smoothness term is an approximation to the minimization of curvature and variation of curvature in the surface. These originally intrinsic measures are made parameter dependent to give rise to a linear equation system after differentiation. A more detailed description can be found in Mehlum and Skytt (1997). The weight on the smoothness term is kept low to emphasize the approximation accuracy. In the examples of the next sections we took  $\alpha_1 = 1.10^{-9}$  and  $\alpha_2 = 1 - \alpha_1$ .

*MBA:* The multilevel B-spline approximation was introduced in Lee et al. (1997) for data interpolation and extended by Zhang et al. (1998) for approximating scattered data with HB splines. We assume that all observations are of the form  $(x, y, z(x, y))$ , and that we have an initial surface approximation via the LS method. A mesh refinement is performed, and the coefficients of the new surface approximation are calculated individually using the residuals of the preceding approximation  $F_{j-1}$  (see Fig.3).

For each B-spline  $B_i$ , we call  $r_c = F_{j-1}(x_c, y_c) - z_c$  the residuals at points  $(x_c, y_c), c = 1, \dots, C$ , situated in the support of this B-spline. The corresponding coefficient  $Q_i$  of the approximated residual surface is determined as

$$Q_i = \frac{\sum_{c=1}^C (B_i(x_c, y_c))^2 \varphi_{i,c}}{\sum_{c=1}^C (B_i(x_c, y_c))^2} \quad (4)$$

The factor  $\varphi_{i,c}$  is computed individually for each point  $(x_c, y_c)$ . The corresponding value  $r_c$  is described as a linear combination of B-splines and a value  $\varphi_{h,c}$  for each B-spline,  $B_h, h = 1, \dots, H$  having  $(x_c, y_c)$  in its support.  $\varphi_{h,c}$  can be computed as the solution of an undetermined equation system.

$$\varphi_{h,c} = \frac{B_h(x_c, y_c) r_c}{\sum_{k=1}^H (B_k(x_c, y_c))^2}.$$

This gives different factors for each data point and each B-spline having these points in their support, and the final value  $\varphi_{i,c}$  to be used in (4) is found by minimizing the error

$$e(Q_i) = \sum_{c=1}^C (Q_i B_i(x_c, y_c) - \varphi_{i,c} B_i(x_c, y_c))^2 \text{ with respect to } Q_i.$$

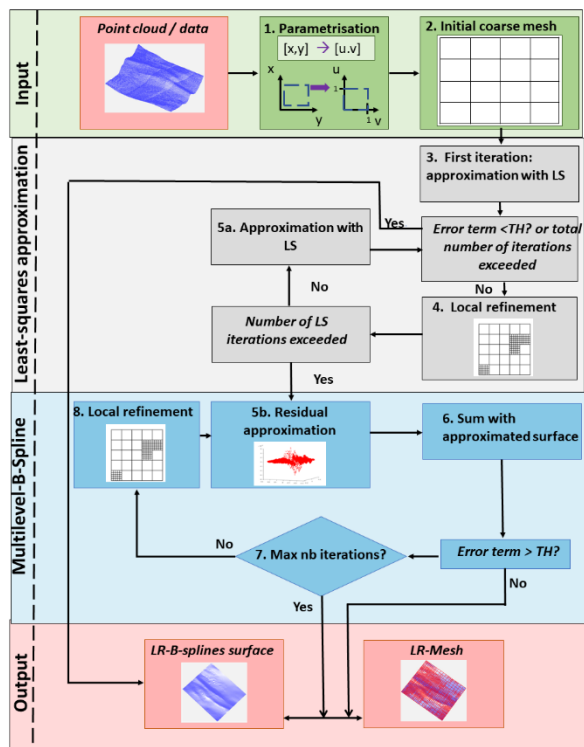
If all the data points in the support of  $B_i$ , are closer to the surface than a given threshold  $TH$ ,  $Q_i$  is set to zero.

The new LR B-spline surface is computed as the sum of the initial surface and the surface obtained from the residuals' approximation,  $F_j(x, y) = F_{j-1}(x, y) + \sum_{i=1}^n Q_i B_i(x, y)$ .

This step is performed until the maximum number of iterations or the threshold is reached. The MBA algorithm can also be applied for volume approximation. The points  $(x, y, z(x, y))$

can be replaced by  $(x, y, z, h(x, y, z))$ , and the method is directly applicable in the case of LR B-spline volumes.

The procedure is summarised in a flow chart in Figure 3. The stop criterion is that the maximum error (locally) is within the threshold. A geospatial point cloud may represent a very rough terrain and contain noise and possible outliers. Thus, it is not necessarily beneficial to continue the iterations until all points are closer to the surface than  $TH$ . Typically, the process is stopped by the number of iterations. The accuracy improvement tends to drastically slow down when most points have a distance to the surface less than the threshold (Skytt and Dokken, 2021).



**Figure 3:** Combination of MBA and LS for adaptive surface approximation with LR B-spline

## 2.2 Goodness of fit

The definition of an optimal surface is a controversial topic and should contain both heuristical (e.g. without “ripples”, “smooth”) and statistical components. Average and maximum distances are the standard way of measuring the error of a model. They are defined in the z-direction as  $Avd = \sum_{i=1}^n |z_i - \hat{z}_i|/n$ , and  $Maxd = \max |z - \hat{z}|$ , respectively where  $n$  is the number of data values,  $z_i$  are the given values and  $\hat{z}_i$  are the corresponding estimated values. The number of points outside the threshold is defined as the number of points for which the error term exceeds  $TH$  for a given iteration level. The degree of freedom or number of coefficients to be estimated for a given iteration  $k$  of the refinement is also an important quantity. For the computation time, we used a 64-bit operating system with 8 GB RAM and an Intel(R) Core(TM) i5-6300U CPU @ 2.40 Ghz, 5.8 GHz.

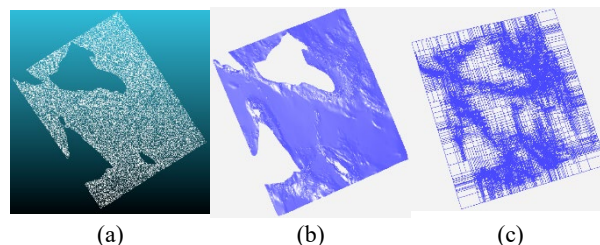
## 2.3 Raster representation

The raster representation is the most frequently used data format in GIS (Li et al., 2005). To make a raster representation of a point cloud, spatial interpolation is used to define values, for instance elevation or precipitation, in a regular grid. The Digital Elevation Model (DEM) is a raster representation and the most commonly used format for processed terrain data. It is a compact, highly structured and efficient representation. The accuracy depends on the resolution of the raster and the selected interpolation method. If the resolution is low compared to the variation of the data then the result will be inaccurate. If it is high, the data volume grows more than necessary. If there are large differences in the local variation of the data in different areas, then a trade-off between accuracy and data volume must be made. A raster represented surface is not completely defined by the sample points. Given a raster, values that are between the sample values have to be

estimated. Several approaches are available. A simple method is bivariate evaluation where the estimated value is computed from a bivariate surface interpolating the four surrounding sample values. Inverse distance weighting (Maleika, 2020) can be used also in this context.

## 3. REAL DATA APPROXIMATION

### 3.1 Bathymetry data set Norway



**Figure 4:** (a) Subsampled point cloud, shown in CloudCompare (b) LR B-spline surface approximating the point cloud, (c) the element structure of the surface (LR-mesh)

We will illustrate the LR B-spline approximation by applying the algorithm to a selected data set. The bathymetry data set Figure 4(a) covers an area of slightly less than one square kilometer close to the Norwegian coast. It consists of about 11 millions points with a quite uniform density. The area is shallow, the depth varies from -27.94 m to -0.55 m. The point cloud contains outliers, the most significant one is two points with the same x- and y-values and 2.38 m. difference in z. In Figure 4(b) the LR B-spline surface approximating the point cloud is trimmed to fit the point cloud domain.

The surface has a rectangular domain in  $x$  and  $y$ . Since the point cloud has a boundary with large bends and contains holes, the surface will extrapolate the known information in some areas. To face that challenge, a 2-step procedure is applied:

- Restricting the values of the coefficients to an interval
- In general, both the LS and MBA methods make smooth transitions between existing point groups. Unfortunately, a combination of holes and data with a steep trend can lead to exaggerated heights, in particular with LS. The positivity of the basis functions combined with the partition of unity property implies that B-spline curves are bounded by their coefficients. In non-degenerate cases, this applies also to LR B-spline surfaces. Thus, we ensure well behaved surfaces in areas with missing points by restricting the values of the coefficients to an interval depending on the height range of the data points.

- Trimming

We do not know the shape of the terrain in areas without points and use trimming to restrict the surface domain to the point cloud domain. To that aim, we bound the points by curves in the  $xy$ -plane. The curves are arranged in one loop for the outer boundary and one curve for each hole and associated to the parameter domain ( $xy$ -plane for points parameterized by their  $x$ - and  $y$ -values) of the surface. The outer loop is counter clockwise oriented, while eventual inner loops are clockwise oriented. By convention only the areas of the surface situated to the left of such trimming loops are considered valid.

The sea floor covered by the data set contains both smooth areas and areas with significant variation. This can be recognized by the element structure of the surface shown in Figure 4(c). We can see that the surface is less refined in areas where there are no points and in areas with little variation in the sea bed.

We approximate the point cloud with a LR B-spline surface using an increasing number of iterations. The threshold is set to 0.5 m

and an initial tensor-product surface of ten by ten coefficients approximating the points is given. In this contribution, we select an approximation approach that has stable good results with respect to execution time and number of coefficients (Skytt & Dokken, 2021). The results of the approximation are presented in Table 1 and more details are given in Table 2.

We compare the approximation accuracy of the LR B-spline surfaces with rasters of varying resolution. The raster surfaces are interpolated using inverse distance weighing with a valid distance of 10, i.e., each grid point is determined on the basis of approximately 8300 data points. The rasters are evaluated by linear interpolation between grid points. The results are presented in Table 1 and 2. The computation time excludes time for reading and writing.

**Table 1.** Results of the approximation with LR B-spline and raster representation. R, is for Raster of a given accuracy in m, LRB is for LR B-spline surface, the number is the iteration step.

Type	Max d (m)	Av d (m)	No. of coefs	No. out	Storage (MB)	CT (m,s)
LRB 7	2.82	0.065	47 687	32 365	3.3 MB	1m29s
LRB 9	2.49	0.06	124 498	23 215	9.4 MB	2m10s
LRB 16	1.19	0.055	467 811	840	38 MB	15m10s
LRB 20	1.19	0.055	481 411	493	39 MB	17m50s
R 2 m	4	0.12	443 x 492	320 762	852 KB	
R 1.5 m	3.66	0.1	590 x 656	223 527	1.5 MB	
R 1 m	3.23	0.084	885 x 984	136 250	3.4 MB	
R 0.5 m	2.98	0.066	1769 x 1967	65 952	14 MB	

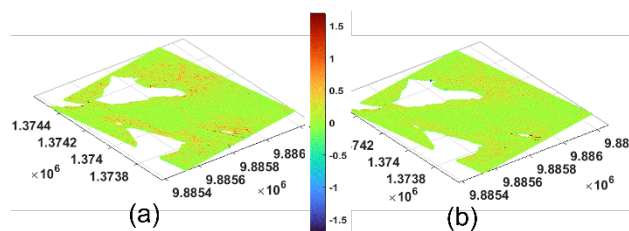
The maximum and average distances between the surface and the points are reported in Table 1 along with the number of LR B-spline coefficients or raster size. The number of points with a distance to the surface larger than the threshold and the file sizes are given. Execution time is reported for LR B-spline approximation, but not for computation of the raster surface. This implementation is less refined and the numbers would be misleading.

**Table 2.** Point distribution with respect to the distance to the approximated surface (number of points for each range of distance in m).

	$-\infty,-2.5$	$-2.5,-1.5$	$-1.5,-0.5$	$-0.5,0.5$	$0.5,1.5$	$1.5, \infty$
LRB 7	5	299	20 656	11 717 745	11 384	21
LRB 9	0	153	16 271	11 126 895	6 786	5
LRB 16	0	0	479	11 149 270	361	0
<b>LRB 20</b>	<b>0</b>	<b>0</b>	<b>258</b>	<b>11 149 617</b>	<b>235</b>	<b>0</b>
R 2 m	1 500	14 507	153 068	10 829 348	151 051	636
R 1.5 m	538	8 785	114 103	10 926 583	99 828	273
R 1 m	106	3 557	77 767	11 013 860	54 641	179
<b>R 0.5 m</b>	<b>21</b>	<b>951</b>	<b>42 971</b>	<b>11 084 158</b>	<b>21 948</b>	<b>61</b>

The accuracy of the LR B-spline surface is increased with an increasing number of iterations and surface coefficients, but the gain is drastically decreased from the 16<sup>th</sup> to the 20<sup>th</sup> iteration. The raster surfaces have systematically poorer accuracy than the LR B-spline surfaces, see Table 2. A raster is a very uniform representation while the LR B-spline surfaces are able to adapt to the variation in the data sets.

We further note that the LR B-spline surface after 7 iterations and the raster with resolution 1 m have similar file sizes. The raster surface is stored as GeoTiff, a binary format while the LR B-spline is stored in an ascii format. A compressed version of the LR B-spline file after 20<sup>th</sup> iteration has size 744KB.



**Figure 5.** (a) The distance field corresponding to the raster representation (1 m). (b) The distance field corresponding to the LR B-spline surface after 7 iterations. The same scaling was used for both representation

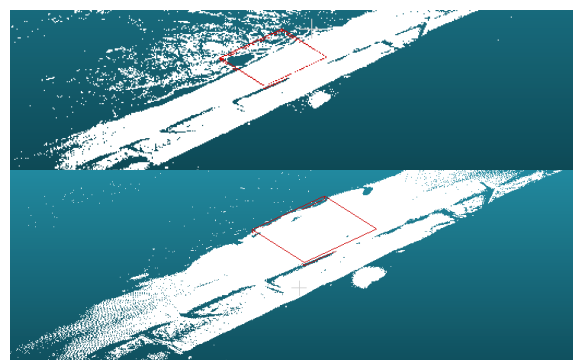
Figure 5 shows the distance fields between the point cloud and the raster surface with resolution 1 m (a) and the LR B-spline surface after 7 iterations (b). In both cases the majority of the points lies within a short distance to the surface. The largest distances can be found in steep and ragged areas. Much less points have a large distance to the approximated surface even with few iterations to compute the LR B-spline surface, compared to the raster representation. This finding confirms the results presented in Table 1 and highlights the goodness of fit that can be reached with the LR B-spline surfaces.

## 3.2 Coastal data set Netherlands

### 3.2.1 LR B-spline volume for spatio-temporal analysis

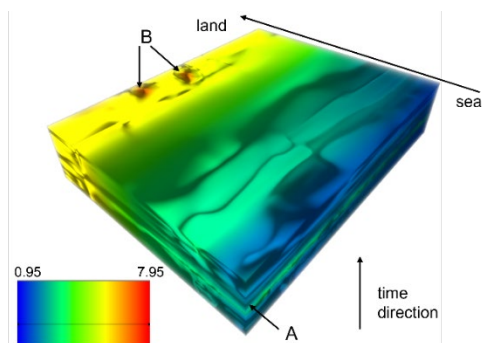
In order to further demonstrate the potential of the LR B-spline to approximate GIS dataset from coastal region method, we made use of a dataset freely available at <https://doi.pangaea.de/10.1594/PANGAEA.934058>.

The dataset was acquired for the CoastScan project (Vos and Kuschnerus, 2017) along the coast of Kijkduin in the



**Figure 6.** Selected point clouds for January 1 (top) and January 11 (bottom) visualized in CloudCompare. The domains have the same coordinates and are highlighted by the red box.

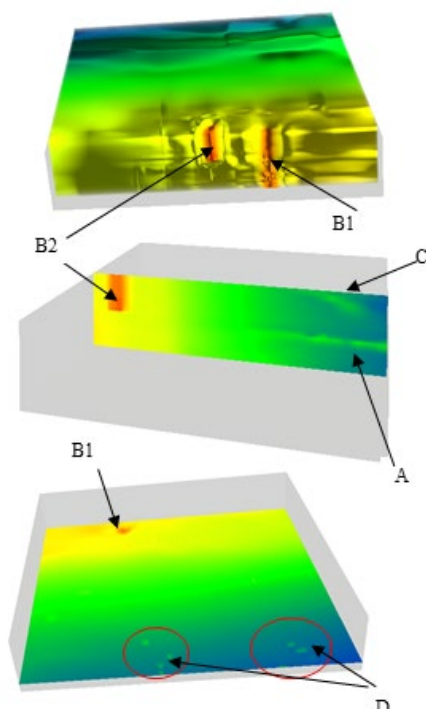
Netherlands. The Terrestrial Laser Scanner Riegl VZ2000 laser scanner was used mounted on the roof of a hotel and programmed to perform a scan of the nearby dune and beach area every hour. More information about the pre-processing of the point clouds can be found in the dedicated publication. We selected the same rectangular area of 110 by 140 m from the point clouds for daily acquisitions and removed obvious outliers. The elevation range is [1.56, 7.73]. Within this area the pattern of the point clouds differs between the days as illustrated in Figure 6.



**Figure 7.** Development of elevation from January the 1<sup>st</sup> to 31<sup>th</sup>

To get an overview of the elevation throughout the period, we add the time as a coordinate in addition to the x- and y-values of the points and interpret the elevation as an associated scalar. A point cloud assembled from all epochs is approximated by a LR B-spline volume. We apply a threshold of 0.2 m and 8 iteration steps. This is sufficient to get a very good approximation of the smooth component (see also Tab.3). The maximum distance between the points and the volume is 2.05 m and the average distance is 0.02 m. 1583 points out of 2.4 million have a larger distance than the threshold (0.066 %) and the average distance in these points is 0.41 m.

Figure 7 shows the corresponding result. The elevation is visualized as colours: blue close to the seaside represents the minimum height. The elevation is increased as we move further from the sea and is relatively constant during time with the exception of the area marked with A in the figure. Here the elevation is increased for a period of a couple of days. At the position marked with B, there are two spots where the elevation is significantly higher than in the surroundings.



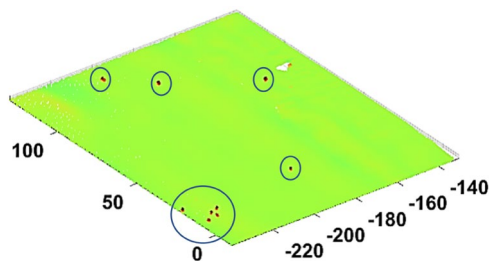
**Figure 8.** Details of the sand dune changes. Top: a closer view of the sand piles. Middle: a vertical cut of the volume in Figure 7 through one of the sand piles. Bottom: a horizontal cut in the period 5th to 6th of January.

We made some cuts in the volumetric approximation of the elevation during time to reveal information in the inner, see Figure 8. The direction from sea to land is now left to right while the time direction is vertical.

- We can recognize sand mounds at B. From Figure 8 (top), one is present during the entire period (B1) while the other appears mid-way (B2).
- A height difference at A is identified together with a second less pronounced difference towards the end of the month at C.
- Some objects seem to have appeared on the surface in a shorter period for later to be removed, e.g., at D, not visible in Figure 8 (top) but in Figure 8 (bottom).

### 3.2.2 Surface approximation to study deformation pattern

To reveal even more detail, we turn to the individual point clouds and approximate them with LR B-spline. In 16 cases the approximation converges totally with all points within the threshold. The approximation accuracy is reported in Table 3 for some data sets. In addition, the range of the reported numbers is presented. Distances are given in m. Their small values highlight the high confidence that can be put on the approximation. The number of coefficients varies with the data density as illustrated in Figure 6. The average distance is below 0.025 m. We note that the maximum distance is slightly higher for days 170105 and 170106. Fortunately, the average distance is similar to the other days: we link this discrepancy with a worthier approximation in a very specific domain. Indeed, we were able to identify local height variations related to some items on the surface. Figure 9 shows the positions of these items at January 6<sup>th</sup>. The point cloud of this day is compared to the surface approximating the points of January 8<sup>th</sup> when the items have been removed. Figure 9 shows



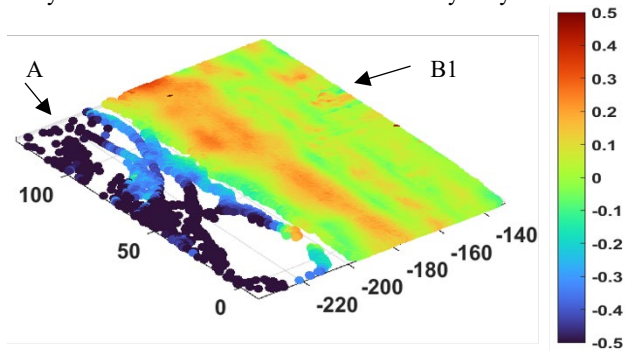
**Figure 9.** Prominent differences when point cloud 170106 is compared to the surface corresponding to 170108.

the corresponding residuals.

**Table 3.** Results of the surface approximation. The number of points, the maximum and average distance in m to the surface, the number of unresolved points and the number of surface coefficients are given.

	No pts	Max d (m)	Av d (m)	No out tolerance	No coeffs
Highest	84 041	0.983	0.028	122	3000
Lowest	57 041	0.179	0.014	0	186
170105	83 850	0.796	0.014	57	2605
170106	84 041	0.983	0.016	122	3000
170107	81 787	0.784	0.016	10	686
170108	83 838	0.196	0.02	0	267
170111	66 010	0.2	0.022	2	336
170113	63 180	0.339	0.023	5	903
170116	83 504	0.198	0.017	0	299
170117	83 698	0.198	0.017	0	517
170123	83 006	0.275	0.022	1	414
170130	82 365	0.198	0.025	0	337

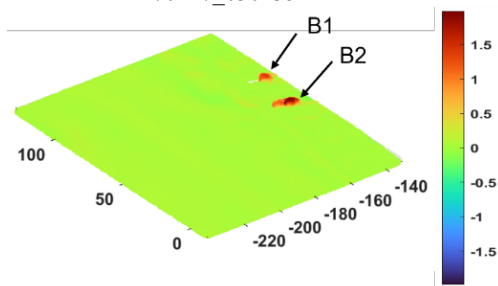
To go further into details, we investigate specifically the aforementioned domains A, B, and C (Fig.8) with the aim to analyse the deformation and find out the “story” beyond.



**Figure 10.** Difference in [m] between point cloud 170111 and the surface corresponding to 170113.

From Figure 7 and 8 we know that there was a temporary increase in elevation in a part of the domain (indicated with the flag A). This happens between point cloud 170110\_210054 and 170113\_230038 as illustrated in Figure 10, which shows the distance fields between the point cloud from 170111 and the surface approximating day 170113. A surface to surface approximation comparison is shown in Figure 12 but not for other examples: it could indicate misleading differences in areas with missing information, i.e., with sparse and varying coverage.

Kuschnerus et al. (2021) stated that on a small area of the beach at the bottom of a footpath, sand accumulated after a storm. Afterwards, the beach was cleared, and the superfluous sand forms a pile. This domain is indicated by B2 in Figure 8 and further analysed in Figure 11 where the points from 170117\_030135 are compared with the surface approximating the cloud from 170116\_010114. The pile marked B1 has been there during the entire period but has increased (see Fig.12). The pile B2 is new from 170117\_030135.

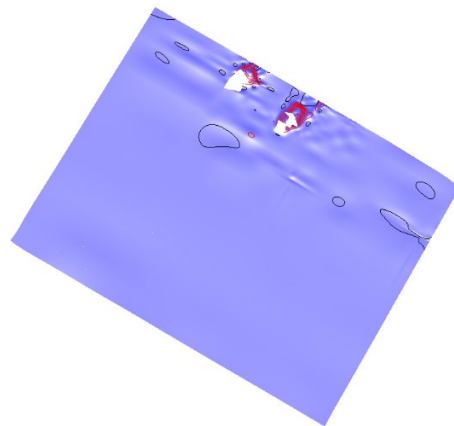


**Figure 11.** Comparison between point cloud 170117 and the surface corresponding to 170116

A surface-to-surface comparison for the same two point clouds is shown in Figure 12 for the sake of completeness. The point clouds are approximated by LR B-spline functional surfaces,  $F_{17016}$  and  $F_{17017}$ . The difference between them can be represented as a new surface,  $F_{diff}(x,y) = F_{170117}(x,y) - F_{170116}(x,y)$ . In Figure 12, this surface is shown together with contour lines at intervals of 0.2 m, reaching from 0.0 m. to 1.6 m. In most of the area, the height of  $F_{diff}$  is less than 0.2 m. The zero level is shown in black, the remaining contours are red. The elevation is higher at the 17<sup>th</sup>, with some exceptions. The major differences are at the sand piles: B2 is new while B1 is extended. *Dealing with holes and ragged boundaries*

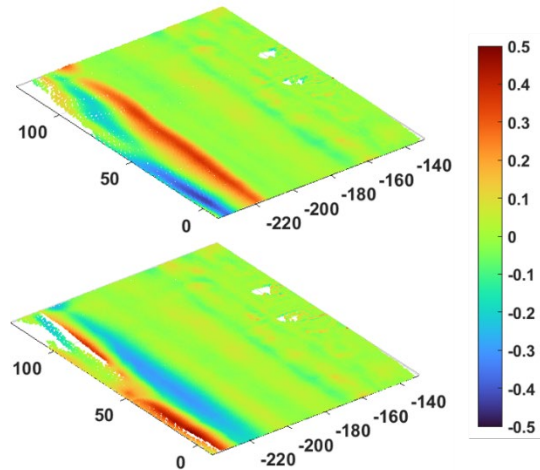
The coverage of the selected point clouds varies, and several of the clouds have holes and/or ragged boundaries. The structures of the clouds related to the two surfaces  $F_{17016}$  and  $F_{17017}$  are

relatively similar although 170117\_030135 has one more hole than 170116\_010114, and the size of the other hole is larger. To avoid comparing surface information without corresponding point cloud information,  $F_{diff}$  is trimmed with respect to 170117\_030135.



**Figure 12.** The distance between the surfaces approximating the data at 170116 and 170117 with contour lines.

Finally, we investigate the feature marked with C in Figure 8. Figure 13 highlights that the difference is linked with a local alteration in sand height.



**Figure 13.** Difference between the point cloud and surface. At the top points from 170130 are compared to the surface representing 170123 and at the bottom points from 170123 are compared to the surface corresponding to 170130.

#### 4. CONCLUSION

The LR B-spline surface allows adaptive surface refinement, and is a promising alternative to more standard NURBS representation for approximating GIS point clouds. In this contribution, we have set up the mathematical background to understand how local refinement is performed by means of B-splines splitting and insertion of new meshlines in the LR-mesh. We have presented an effective method called MBA that allows to face the drawbacks associated with the LS method for approximating large point clouds. The LR B-spline representation has some advantages over the effective regularity of the raster format. We have highlighted how an effective surface approximation after only seven iterations could be performed with a much lower storage size. With respect to the raster approximation, the maximum distance was improved by 1 dm compared with the raster representation of resolution 0.5 m

for our case study. Particularly in steep and ragged areas, the LR B-spline approximation performed better than the raster one. We have shown that increasing the number of iterations was not linked with an improved surface fitting.

We have further shown that LR B-spline surfaces can be used to identify patterns of change in point clouds from TLS recorded at different days. A sand dune was taken as an example. We could reconstruct the spatio-temporal story of the dune before and after a storm and highlighted specific height variations due to objects that were removed. We have shown that the proposed approximation method combining LS and MBA with LR B-spline could be used to approximate point clouds even in challenging domains with a variable data density and holes. LR B-spline surfaces are smooth and can, due to adaptivity, represent local details without a drastic increase in data size. Being a novel format, it is not supported in GIS systems. Fortunately, the surfaces can be exported as rasters in various resolutions as well as collections of tensor product spline surfaces.

Outliers may obstruct the surface computation. Single points do not influence the surface significantly, but collection of outliers points and outliers in areas with sparse data points will drag the surface in their direction. The outlier problem is an important topic for further work using robust estimation method.

Ongoing research focuses on the approximation threshold and a stop criterion for the number of iterations in the approximation algorithm using statistical quantity such as information criterion. The validation of the obtained surface is also an important topic with the intent to find a balance between the accuracy of the approximation and the number of coefficients representing the surface.

#### ACKNOWLEDGEMENTS

This study is supported by the Deutsche Forschungsgemeinschaft under the project KE2453/2-1 and the Norwegian research council under grant number 270922. The first data set is provided by the Norwegian map authorities, division Sjøkartverket. The Nederlandse Organisatie voor Wetenschappelijk Onderzoek is thanked for having founded the project which led to the freely available point clouds of the sand-dune.

#### REFERENCES

- CloudCompare (version 2.12.alpha) [GPL software], 2021. <http://www.cloudcompare.org/>
- Dokken T., Lyche T., Pettersen K.F., 2013: Polynomial splines over locally refined box-partitions, *CADG*, 30(3):331-356. <https://doi.org/10.1016/j.cagd.2012.12.005>.
- Dokken T., Skytt V., Barrowclough O., 2019: Trivariate Spline Representations for Computer Aided Design and Additive Manufacturing. *Computers & Mathematics with Application*, 78(7):2168–2182. <https://doi.org/10.1016/j.camwa.2018.08.017>.
- Kermarrec G., Kargoll B., Alkhatib, H., 2020: On the impact of correlations on the congruence test: a bootstrap approach. *Acta Geod Geophys* 55:495–513. <https://doi.org/10.1007/s40328-020-00302-8>.
- Kuschnerus M., Lindenbergh R., Vos S., 2021: Coastal change patterns from time series clustering of permanent laser scan data. *Earth Surf. Dynam.*, 9, 89–103, 2021 <https://doi.org/10.5194/esurf-9-89-2021>.
- Lee S., Wolberg G., Shin S.Y., 1997: Scattered data interpolation with multilevel B-splines. *IEEE Trans Vis Comput Graph*, 3(3):228–244. <https://doi.org/10.1109/2945.620490>.
- Li, Z., Zhu, Q. and Gold, C., 2005: Digital terrain modeling: principles and methodology, CRC Press, Boca Raton, FL.
- Maleika, W., 2020: Inverse distance weighting method optimization in the process of digital terrain model creation based

on data collected from a multibeam echosounder. *Appl Geomat* 12, 397–407. <https://doi.org/10.1007/s12518-020-00307-6>.

Mehlum E., Skytt V., 1997: Surface Editing, In: Dæ len M., Tveito A. (eds) *Numerical Methods and Software Tools in Industrial Mathematics*, Birkhäuser, Boston, 1997, 381–396. [https://doi.org/10.1007/978-1-4612-1984-2\\_19](https://doi.org/10.1007/978-1-4612-1984-2_19).

Raja V., Fernandes K.J., 2008: Reverse engineering: an industrial perspective. Springer Series in Advanced Manufacturing, Vol. 18. Godalming, UK: Springer Verlag London Ltd.; 2008. ISBN 978-1-84628-855-5

Sederberg T. W., Zheng J., Bakenov A., Nasri A., 2003: T-splines and T-NURCCs, *ACM Trans. Graph.*, 22(3):477–484. <https://doi.org/10.1145/882262.882295>.

Stangeby, I., Dokken, T., 2021, Properties of Spline Spaces Over Structured Hierarchical Box Partitions, in *Isogeometric Analysis and Applications 2018*, Harald van Brummelen, Cornelis Vuik, Matthias Moller Clemens Verhoosel, Bernd Simeon, Bert Juttler (ed), Springer, 2021, pp.177-208

Skytt V., Barrowclough O., Dokken T., 2015: Locally refined spline surfaces for representation of terrain data. *Comput Graph* 49:48–58. <https://doi.org/10.1016/j.cag.2015.03.006>.

Skytt V., Dokken T., 2021: Scattered data approximation by LR B-spline surfaces. A study on refinement strategies for efficient approximation. ArXiv:2012.08130. *To appear in the the Springer INdAM volume entitled "Geometric Challenges in Isogeometric Analysis"*, 2021.

Vos S, Kuschnerus M. CoastScan, 2017: data of daily scans at low tide Kijkduin January 2017. 4TU. ResearchData. Dataset. <https://doi.org/10.4121/uuid:409d3634-0f52-49ea-8047-aeb0fefe78af>

Wang Y., 2009: Free-form surface representation and approximation using T-splines. PhD, Nanyang Technological University, 2009. doi: 10.32657/10356/19090.

Woodward CD., 1988: Skinning techniques for interactive B-spline surface interpolation. *Comput Aided Des*, 20(8):441–451. doi: 10.1016/0010-4485(88)90002-4.

Zhang W., Tang Z., Li J., 1998: Adaptive hierarchical B-spline surface approximation of large-scale scattered data. *Computer graphics and applications*. Pacific Graphics 98. Sixth Pacific Conference; <https://doi.org/10.1109/PCCGA.1998.731993>.