Convergence and Remeshing Criteria for Fitting Method Based on Iterative Reparameterization via Plane-Stress Model

Ivo Marinić-Kragić[1] and Damir Vučina[2]

[1]University of Split, FESB, imarinic@fesb.hr
[2]University of Split, FESB, vucina@fesb.hr

Corresponding author: Ivo Marinić-Kragić, imarinic@fesb.hr

Abstract. In the process of reverse engineering a physical part is used to construct a CAD model. The physical part is usually represented as a triangulated point cloud (i.e., mesh). Triangulated mesh parameterization can be achieved by many different methods. When the parameterization is presented as mapping the 3D surface to planar domain, it can be illustrated that different methods result in different distortions in angles and areas. These distortions are important in reverse engineering, as they lead to a distorted B-spline control-point grid after fitting. Ideally, the fitting method based on parameterization should lead to a smooth B-spline control-point grid while keeping fitting error as small as possible. This is usually a problem in which compromises must be made where one objective is sacrificed for the other. The method for parameterization-based B-spline fitting presented in this paper can be regarded as a compromise between the harmonic method (very smooth control-point grid, but possible large fitting error) and the feature sensitive parameterization (low error but distorted control-point grid). In our method, initial parametrization is constructed using the harmonic mapping method. The parameterization is subsequently revised using feature-related and smoothness-related metrics. Different convergence criteria and remapping metrics were investigated on 4 different geometries obtained by 3D scanning. It was shown that the method can be slightly tuned depending on the desired output (small fitting error vs. smooth surface).

Keywords: Reverse Engineering, Parametric Surface Fitting, Shape Parameterization, Shape Optimization, B-spline
DOI: https://doi.org/10.14733/cadaps.2021.1000-1017

1 INTRODUCTION

In the process of reverse engineering [1] a physical part is used to construct a CAD model. Two main steps of the reverse engineering process are: first part is the digitalization of physical object which commonly performed by the object 3D surface scanning and the second step is constructing a compact surface CAD model. In one of the reverse engineering paths, the object can be divided

in a user-selected patches which can be represented by a NURBS surface. For example, in boat hull representation a user might select each side of the hull to be a separate surface patch. In this case, a NURBS surface fitting algorithm is required for each patch separately. As surface fitting is common topic in computer-aided design there already exists a large amount of literature. Good summary on this topic can be found in [3,2]. The techniques for surface fitting can be divided to parameterization-based and the ones that do not use parameterization. In the parameterization-based methods, the crucial part of the fitting method is the point-cloud parameterization in which parameters *u,v* are allocated to all points of the 3D point-cloud. These methods can be divided to parameterization of organized [4] and unorganized points [5]. This paper deals with reverse engineering methods in which we assume that 3D model exists in a form of triangulated point cloud (surface mesh), and the objective is to construct a B-spline surface using a user-defined part of the surface.

Triangulated mesh parameterization can be achieved by many different methods. When the parameterization is presented as mapping the 3D surface to planar domain, it can be illustrated that different methods result in different distortions in angles and areas. These distortions are important in reverse engineering, as they lead to a distorted B-spline control-point grid after fitting. Ideally, the fitting method based on parameterization should lead to a smooth B-spline control-point grid but keeping fitting error as small as possible. This is usually a problem in which compromises must be made where one objective is sacrificed for the other. One of the basic parameterization methods is the harmonic mapping [6]. Harmonic map can be computed efficiently (using finite element method (FEM) for example), and the results are good in most simple cases, as the method minimizes deformation of the map (Dirichlet energy). However, for complex examples this method can allocate too few control points to areas which are of interest in engineering cases. Many other different methods exist. A very different method is the one in [8,7], which performs parameterization based on elastic springs. This method normally leads to more distortions but is easy to implement and can be even more computationally efficient. To improve the fitting regarding the capturing of geometric features, method was developed in [9] called the feature sensitive parameterization. This method performs point cloud projections to parametric *u*-*v* space such that the areas which contain geometric features are allocated more space in the parameter domain. A drawback of this method is that the smoothness of the resulting B-spline control-point is low as many points are sharply concentrated at the geometric features ignoring possible distortions. The method used for parameterization-based B-spline fitting in this paper can be regarded as a compromise between the harmonic method (very smooth control-point grid, but possible large fitting error) and the feature sensitive parameterization (low error but distorted control-point grid). The method is based on the fitting method used in [10,11]. In this method, initial parametrization is constructed using the harmonic mapping method. The parameterization is subsequently revised using feature-related and smoothness-related metrics. To perform the re-parameterization, plane-stress problem is iteratively solved where the local modulus of elasticity is reduced (equivalent to heating the elastic plate) according to selected set of values.

2 SURFACE REPRESENTATION AND FITTING

Here, point-cloud is defined as a set of 3D points $\boldsymbol{PC}_i$, where *i*=1 to *n* Figure 1 shows an example of a point-cloud for a boat hull geometry. The set of points is connected with edges as shown in the figure (triangulation). Here, the geometry represents complete boat-hull. This is a user-defined segment (hull outer surface) that is extracted from full 3D scanned boat-hull. The segment has a boundary *B* and the boundary has 4 corner points which are also assumed to be user-defined, and the objective of this paper is to investigate methods for B-spline fitting which would afterwards require no additional (of very little) user input.

Other important terms regarding the point cloud in this paper are the point-cloud parameterization and its inverse. The parameterization $\boldsymbol{f}(\boldsymbol{PC}_i)$ is a mapping operator $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ while the inverse operation of the parameterization is called the functionally determined point-cloud

$\mathbf{F}(u,v)$ ($\mathbb{R}^2 \rightarrow \mathbb{R}^3$). The objective of the parameterization is to assigns parameter (u_i,v_i) to each point *i* of the point-cloud. Thus, the inverse operator should result in the original geometry. Since the original surface is assumed to be composed of triangles, the functionally determined point-cloud $\mathbf{F}(u,v)$ uses the trilinear interpolation.

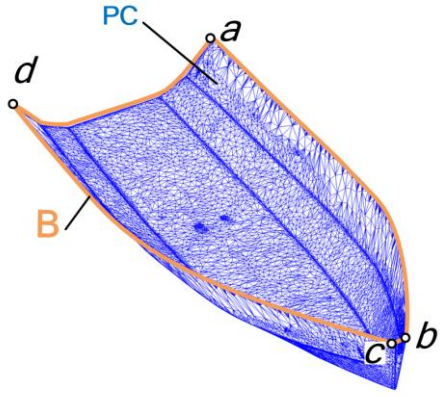


Figure 1: Point-cloud triangulation for boat hull (*B* –boundary, with *a,b,c* and *d* as corner points; *PC* – point cloud points).

The parametric surface described by B-splines [12] can be defined as:

$$\mathbf{S}(u,v) = \sum_{i0=0}^{n0} \sum_{i1=0}^{n1} N_{i0,d0}(u) \cdot N_{i1,d1}(v) \cdot \mathbf{Q}_{i0i1}, u, v \in \left[0,1\right] \tag{2.1}$$

where $N_{i0,d0}(u)$ and $N_{i1,d1}(v)$ are the B-spline basis functions with degrees $d0, d1 \in \mathbb{N}$ and $(\mathrm{n0}+1)\times(\mathrm{n1}+1)$ are the respective numbers of the control points (CPs). The dimensionality of the control points $\boldsymbol{Q}_{\mathrm{ij}}$, can be arbitrary. Here, we will use only clamped B-spline, but this is not necessary in a general case. Furthermore, the fitting method explained in the next section can be used with any parametric surface such as B-spline, NURBS, T-spline, RBF and others if the surface can be represented as $\mathbf{S}\ u,v\ = sum(\boldsymbol{B}_i(u,v) \cdot \boldsymbol{Q}_i)$, where *i*=1 to *n*, (*n* is the number of basis functions).

Based on point cloud, one can easily construct the functionally determined point-cloud $\mathbf{F}(u,v)$ and then, surface ($\boldsymbol{S}(u,v)$) fitting is performed by minimizing the function (finding control points $\boldsymbol{Q}_{\mathrm{i}}$):

$$A = \sum_{j=1}^{m} \left\| \mathbf{S}\ u_j, v_j\ - \mathbf{F}(u_j, v_j) \right\|^2 \tag{2.2}$$

where the parametric coordinates (u_j,v_j) can be located on an equidistant grid in the *u*-*v* space. In this paper we used 100x100 points grid. This problem can be solved efficiently by converting it in a linear system of equations. A smoothing term can easily be added to equation 2.2. However, this was omitted so that the results of the proposed fitting method can be easily evaluated and eventually compared with other methods.

3 RE-PARAMETERIZATION BASED ON PLANE-STRESS MODEL

This section presents novel additions to the previously devised re-parameterization algorithm variations in [10,11]. The overview of the proposed fitting method algorithm is given in the pseudocode:

Modified adaptive fitting method

Input: triangulated point cloud **PC**, B-spline control point grid dimensions and degree
Output: B-spline control points $\mathbf{Q}_i$

1: **Begin**
2: Harmonic mapping: $(u_i,v_i)=\mathbf{f}(\mathbf{PC}_i)$
3: Construct functionally determined point-cloud $\mathbf{F}(u,v)$ using trilinear interpolation
4: Fitting: find $\mathbf{Q}_i$ a by minimizing function *A*, *Eqn.* (2.2)
5: $k=0$
6: **do**
7: $k=k+1$
8: Calculate *relaxation field* $R(u,v)$
9: Re-parameterization based on scalar field *R:* $(u_i,v_i)=\mathbf{f_R}(R, (u_i,v_i))$ using plane-stress model
10: Construct $\mathbf{F}(u,v)$ by trilinear interpolation
11: Fitting: find $\mathbf{Q}_i$ by minimizing function *A*, *Eq.* (2.2)
12: Calculate *ConvergenceCriterion*
13: **while** (*ConvergenceCriterion* > *tolerance*)
14: **End**

The first steps of the method (lines 1-6) are the same as in the previous research. The point cloud is parameterized using the harmonic mapping $(u_i,v_i)=\mathbf{f}(\mathbf{PC}_i)$. This allows us to construct a functionally determined point-cloud $\mathbf{F}(u,v)$ using trilinear interpolation which is used for the initial fitting by Equation (2.2).

The crucial part of the algorithm is the re-parameterization (line 9), which is now different than in previous papers and it is based on solving plane-stress problem [13]. Once the initial parameterization is performed using harmonic mapping, each 3D point is projected to 2D space $(u_i,v_i)=\mathbf{f}(\mathbf{PC}_i)$. In the reparameterization, plane-stress problem is solved for displacement, and this displacement is used to modify the (u_i,v_i) points. The plane-stress problem is defined as follows.

The initial modulus of elasticity was $E=1$. To fully define the problem, the Poisson's ratio ν has to be selected. In this paper, several tests were performed, and only small differences were noticed among which selecting the negative value $\nu=-0.5$ has shown slightly better smoothness, so this value was selected. The finite element mesh used 99x99 grid of four-node rectangular elements (which corresponds to 100x100 nodes, the same as fitting point-grid). For more complex problems, 200x200 nodes mesh was used (in case of damaged clutch housing and part of statue as shown in the results section). The problem is solved using boundary conditions as shown in Figure 2. Once the scalar field for relaxation ($R(u,v)$) is selected, the relaxation process is relatively simple. The selected scalar field modifies the local plate modulus of elasticity by $E_{new}=E/(1+R(u,v))$. Solving the plane-stress problem provides the displacement fields $\Delta u(u,v)$ and $\Delta v(u,v)$, and the reparameterization $\mathbf{f_R}$ is obtained as $(u_i,v_i)=(u_i+\Delta u(u_i,v_i), v_i+\Delta v(u_i,v_i))$.

There are two additional novelties in this paper. First, convergence criterion in previous papers was only the mean square, but here we have shown that different criteria might be slightly better suited for engineering applications. Similarly, different variants of the relaxation field $R(u,v)$ were tested here.

3.1 Test Geometries

Selected objects for most test cases are shown in the Figure 3. Although additional different objects were subjected to tests, here we show mainly the results related to these. The first two examples (boat hull and propeller) represent very different engineering objects which contain features such as flat faces with sharp edges and blade shape with complex curved surfaces. A non-engineering test was also used (Figure 3d) but the main focus of the paper was on the engineering examples Figure 3 a-c.

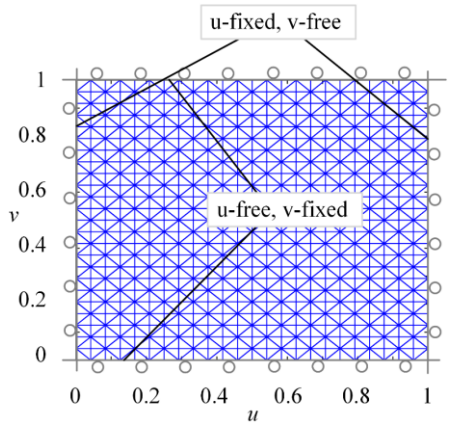


Figure 2: Boundary conditions for the selected plane-stress problem used in re-parameterization.

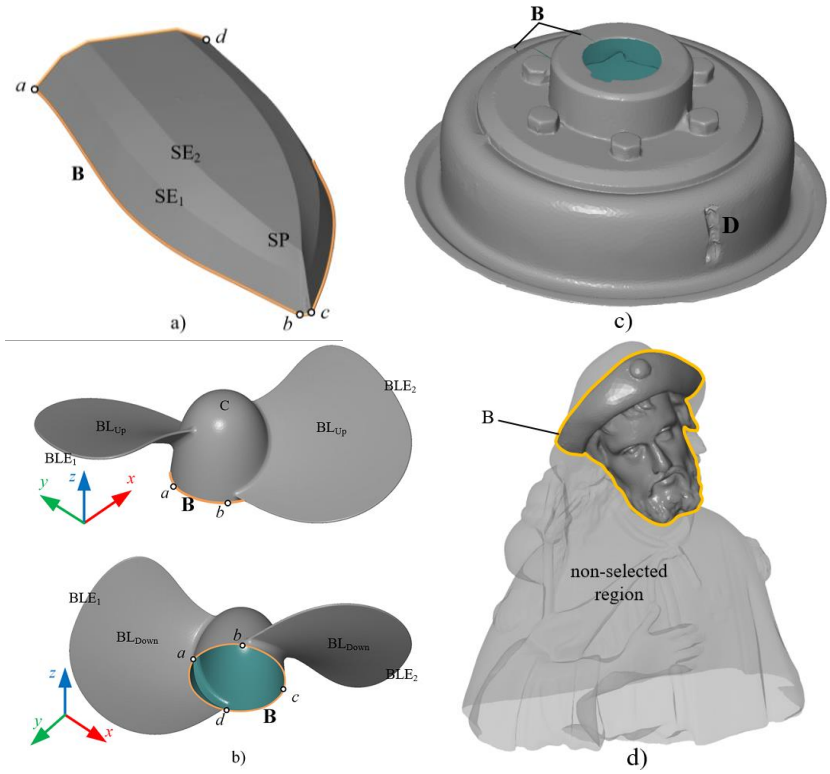
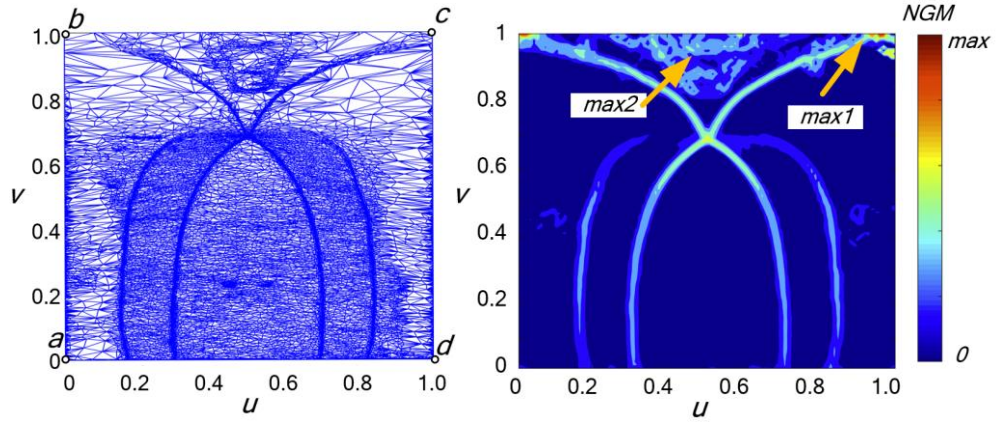


Figure 3: Selected test objects: a) Boat hull and b) Boat propeller c) damaged (**D**) clutch housing and d) part of a statue.

3.2 Considered Relaxation Fields

In this section, we present the novel variations of possible relaxation fields which can yield better results in comparison to previously used methods. The general idea of the relaxation field is to use higher relaxation values at locations with geometric features which require larger number of control points to describe. The application of the relaxation field to the plane-stress problem causes the re-parameterized mesh to enlarge at areas with larger values, and contract at areas with smaller values. For example, in paper [9], changes in local surface normal value was used to

enlarge the mesh parameterization. Although this alone can be useful, the problem is that this parameterization would lead to a non-smooth B-spline control grid as the changes in the parameterization are rapid. The following figure shows the harmonic mapping of the boat hull and the amount of changes in local normal values. To evaluate change in local normal, we use gradient of normal vector (NGM), defined as $NGM = \sqrt{(\partial_u n_i \partial_u n_i + \partial_v n_i \partial_v n_i)}$ (in index notation). This value was calculated based on numerical partial derivatives of normal vector $\vec{n}$ in *u* and *v* directions. The surface normal itself ($\vec{n}$) was calculated numerically using bicubic fit on functionally determined point cloud **F**(*u*,*v*). As shown in figure 4b, the *NGM* value is the largest at the areas at which no important geometric features exist. These areas are the result of non-perfect 3D mesh, and they are likely to exist in most 3D meshes obtained by 3D scanning.



a) b)

Figure 4: Boat hull: a) Harmonic mapping to *u*-*v* domain and b) Changes in normal values

Relaxation field based on *NGM* was modified so that it is multiplied by the local surface face area ($A_f(u,v)$). The result of this value is shown in Figure 5. This produces a better result regarding highlighting of the important features. However, this method leads to other smaller errors (marked *err1-3*). These errors are small in the initial parameterization. However, when used in combination with the adaptive fitting method based on reparameterization, this small error become very significant at later fitting iterations as shown in Figure 5b. The small errors become gradually larger since they are expanded in the parametric domain. Their expansion in the parametric domain means that they will be a larger part of the functionally determined point cloud **F**(*u*,*v*) which is used in the adaptive reparameterization fitting method. Enlargement of these errors leads to a gradual divergence of the adaptive fitting method.

The fitting error (equal-parameter error) is defined as $E(u,v) = \left\| \mathbf{S}(u,v) - \mathbf{F}(u,v) \right\|$, where **S**(*u*,*v*) is the fitted parametric surface. Error can also be defined as signed error $E_{sign}(u,v)$, which can be both positive and negative depending on the local normal vector on the surface. We considered the application of magnitude of the signed error gradient (*MEG*), defined as $MEG = \left\| \nabla E_{sign}(u,v) \right\|$.

The idea of this operation is to re-parameterize only the parts of the geometry in which the fitting error is changing. If the fitting error is constant and large in some area, the control points of the fitted B-spline surface should be moved from the neighborhood in which the error is changing. Also, if the fitting error is small in some area, the gradient *MEG* will also be small. As opposed to the previous re-meshing criteria in which this is not used, now the errors in the relaxation field appear at larger iteration number (see Figure 6c for 20th iteration). This means that better fitting would be achieved before the method starts to diverge.

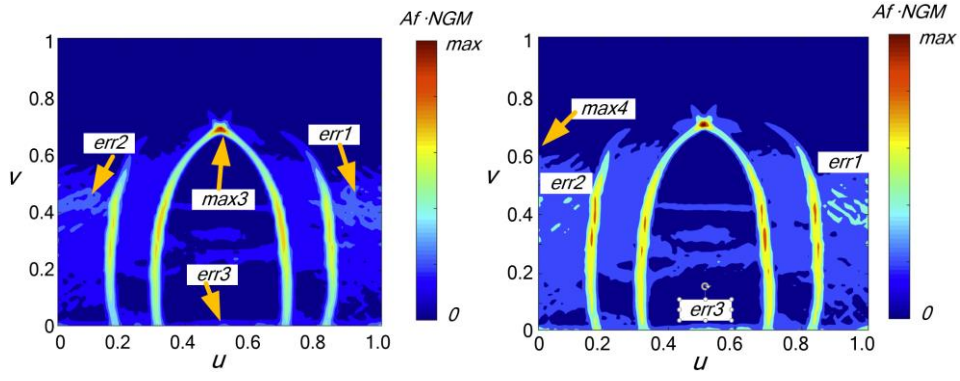


Figure 5: Boat hull changes in normal values multiplied with local face area: a) Harmonic mapping and b) 5th iteration of adaptive fitting method.

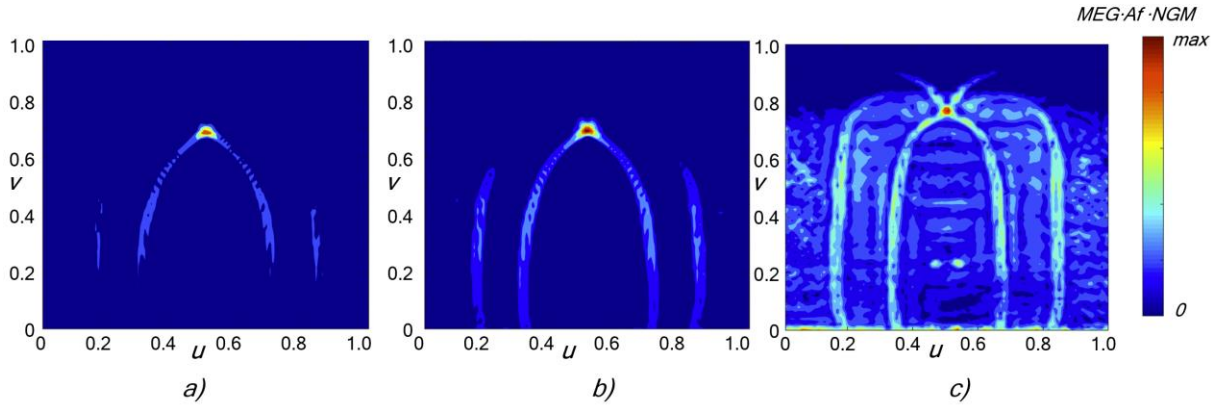


Figure 6: Boat hull changes in custom relaxation field (MEG·A_f·NGM) during fitting: a) Harmonic mapping, b) 5th iteration and c) 20th iteration of the adaptive fitting method.

Further variations can be considered if these fields are combined with the error field *E(u,v).* This increases the reparameterization intensity in the areas with higher error, which is usually desirable. In some cases, the surface locally becomes highly skewed (distorted). To prevent this, one can attempt to modify the relaxation field by local surface skewness field *skew(u,v)*. To prevent reparameterization in the highly skewed areas, the relaxation field was multiplied by (1-*skew(u,v)*), since *skew*=1 is the highest value. Overall, 5 different considered fields can be combined in 15 different permutations. The results section will discuss several different combinations.

Previous relaxation fields were based on values calculated by equal-parameter error. A possible improvement was considered by using the actual distance (minimum distance) instead of parametric fitting error to calculate *MEG(u,v)*. Figure 7 shows that significant difference exists between these two approaches. Note that the actual distance is smaller than the equal-parameter error as it represents the minimal distance between the fitted surface and the corresponding point cloud.

3.3 Considered Convergence Criteria

First criteria that comes to mind is the root mean square error. This can be defined as:

$$RMSE_P = \sqrt{\frac{1}{m}\sum_{j=1}^{m}\left\|\mathbf{S}\left(u_j, v_j\right) - \mathbf{F}(u_j, v_j)\right\|^2} \tag{2.3}$$

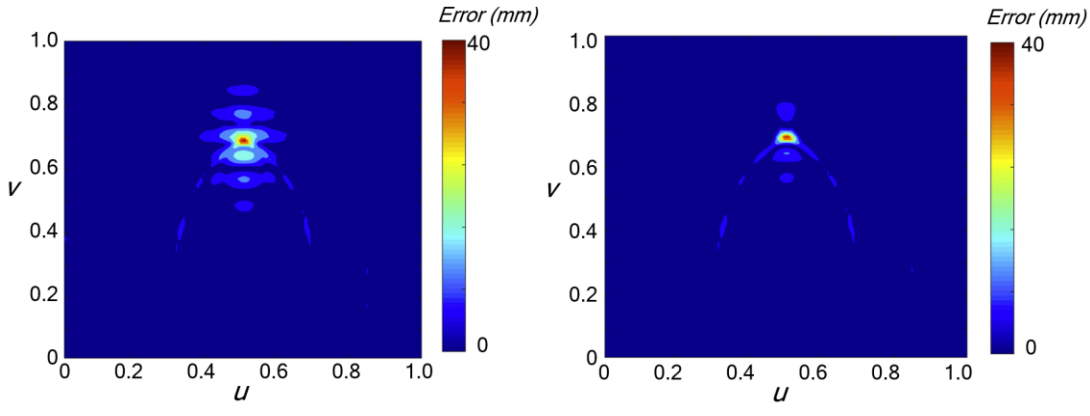


Figure 7: Fitting error based on harmonic mapping fitting, calculated using: a) equal-parameter error and b) actual distance.

where the parametric coordinates (u_j,v_j) were located on an equidistant grid in the *u*-*v* space (100x100 points). The index *P*, designates that this is equal-parameter based distance. Related to this error, one can simply observe the maximum error and use this as a convergence criteria. This value is labeled as $MaxE_P$. These two criteria can be defined by using actual distance instead of the previous definition. For actual distance, each distance is calculated by keeping the point on the $\mathbf{F}(u_j, v_j)$ fixed and minimizing the distance $\left\|\mathbf{S}\ u^*, v^* - \mathbf{F}(u_j, v_j)\right\|$ where *u** and *v** are the variables over which the distance is minimized. This finds the closest point on B-spline surface for each point-cloud point. This definition of error leads to root mean square error $RMSE_A$ and and maximum error $MaxE_A$, where the index *A* designates that the error is based on the actual distance.

Second criteria is related to the area surface *Af(u,v)*. As mentioned in discussion related to Figure 5, the local area of individual triangles changes as a function of the parameters *u* and *v*. The locations where the local sudden peaks in the function were discussed as a possible source of errors. This is also shown for the boat hull fitting example in Figure 8. As it can be seen, the area surface is very smooth in the initial iterations. In later iterations, multiple peaks appear at locations where no geometric features are expected. So, a criterion can be devised to measure the amount of this "peaks". This criterion was defined as an average of the functions gradient magnitude squared. That is the average of $\left\|\nabla A_j(u,v)\right\|$ was calculated over the mesh. This sum was divided by total surface area in order to "normalize" the results, and this value was named *AGR*.

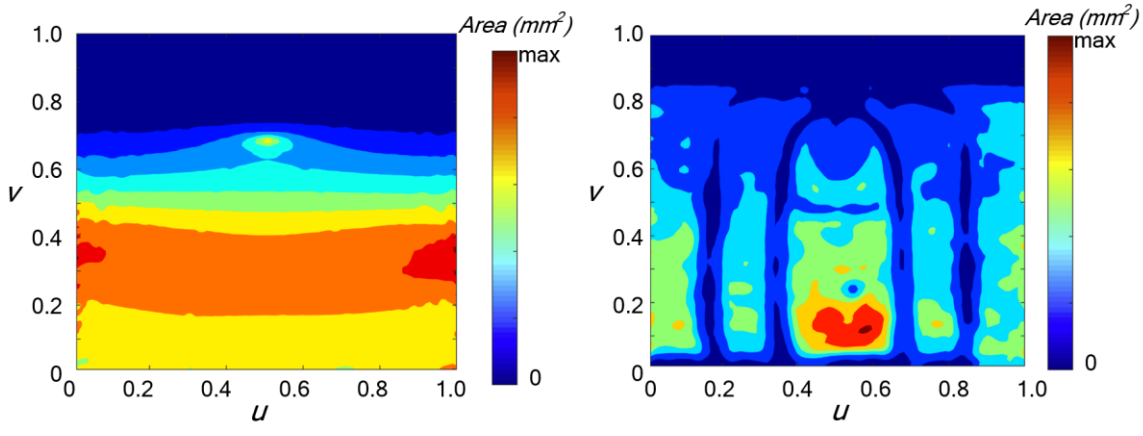
The last considered error is calculated using noise estimation. A noise-related criterion is devised based on the adaptive low-pass Wiener filter. The was calculated on the relaxation surface (MEG·A_f·NGM). The ratio of the standard deviation of the noise surface to the standard deviation of the relaxation surface was used as the convergence criterion. This value was designated as *NR*. Skewness of the surface was initially also considered as a convergence criterion. It was difficult to generalize on this basis, nevertheless the average skewness was monitored as it is still useful indicator of the B-spline surface smoothness. Root mean square skewness of the B-spline surface was monitored and the value was designated as RMSS.

4 RESULTS

4.1 Results for Different Convergence Criteria Tests

In first test cases, relaxation field is kept fixed, so that the effect of different convergence criteria can be tested. Reparameterization was initially performed based on all considered metrics

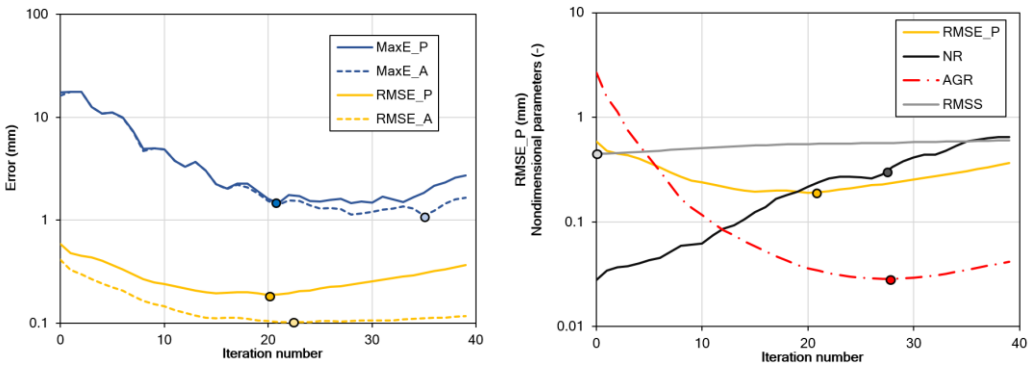
(*MEG·A*$_f$*·NGM·E·*(1-*skew*), see Section 3.2), and further sections will consider other combination. The selected convergence criteria were monitored during the iterative fitting and the results for the propeller case are shown in Figure 9.



a) b)

Figure 8: Area surface of the boat hull $A_f(u,v)$ during fitting: a) Harmonic mapping, b) 20th iteration of the adaptive fitting method.

When observing the error-related values, the minimal value can be considered as the converged result. Different points are obtained if the maximum error or the RMSE is used. Also, these points are different when using equal-parameter distance (_P) or actual distance (_A). The most representative error might be the RMSE based on the actual distance, which is obtained at 23rd iteration. However, the maximum error is still reduced up to 35th iteration while the RSME remains practically the same up to 34th iteration. So, 34th iteration might be considered as the best solution here. However, it is computationally expensive to calculate the actual error during fitting. The remaining considered criteria are shown in Figure 9b.



a) b)

Figure 9: Iterative fitting of propeller: a) Fitting error based on P – equal-parameter distance and A – actual distance, b) Different considered convergence criteria (NR – noise-based, AGR – area-based and skewness-based).

As the main criteria the selected was the noise-based criterion. When the NR value obtained 0.3, we consider the solution as converged. The fitting result is shown in Figure 10. For the boat hull case, the selected convergence criteria during the iterative fitting are shown in Figure 11. Considering the actual error, it can be seen (marked by small circle) that the solution is converged at 25th iteration. If for example, the minimum of RMSE calculated by equal parameter distance was

used as an indicator of convergence, the maximum actual error would be 38mm (5th iteration) while the overall minimum is 7.9mm.

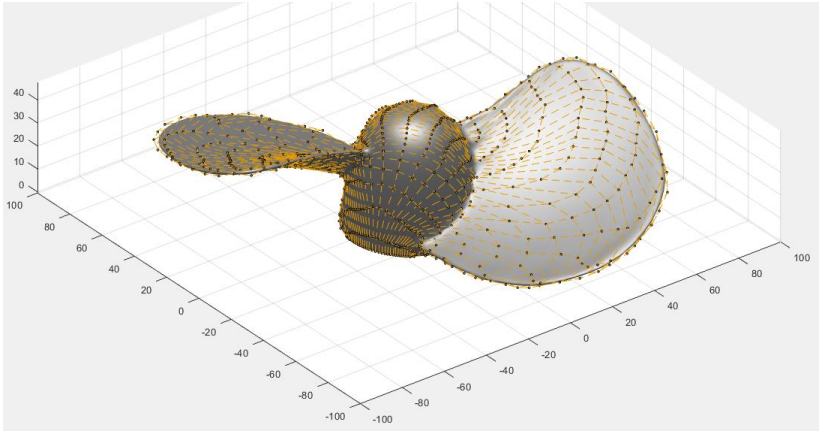
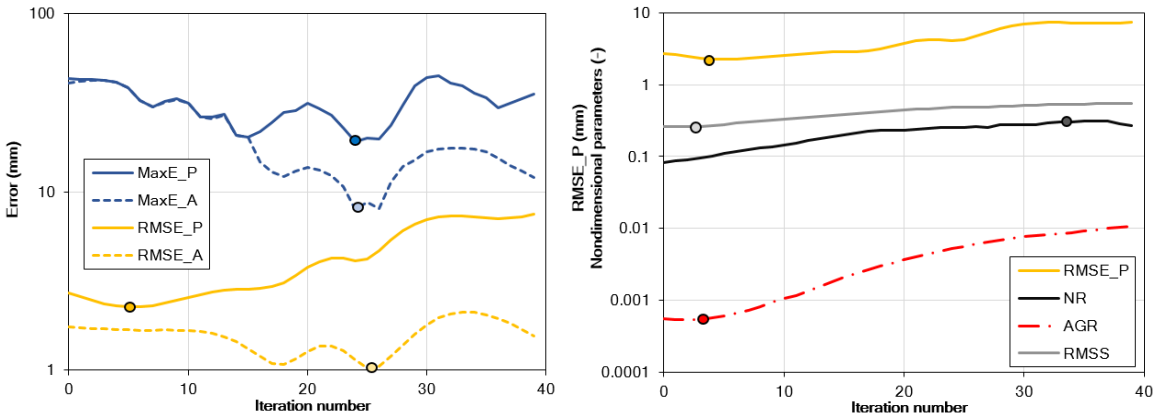


Figure 10: Boat propeller fitting with noise-based convergence criterion on the (iteration = 28).

This example clearly shows why the equal parameter-based error is not a reliable criterion for convergence. Among the remaining selected convergence criteria, the closest to this value is minimum of the maximum parameter-based error, 24th iteration (shown in Figure 12). Also, important to consider in all examples is the root mean square of the surface skewness (RMSS). If the fitting error is reduced at the expense of large increase of RMSS, the result would not be good regarding reverse engineering aspects. The RMSS is practically monotonically increasing in all cases, meaning that lower iteration numbers will give a smoother B-spline grid. This fact could also be used for selecting the appropriate criteria. If some criteria require too much iterations until it is reached, the result would probably be a non-smooth surface.



a) b)

Figure 11: Iterative fitting of boat hull: a) Fitting error based on P – equal-parameter distance and A – actual distance, b) Different considered convergence criteria (NR – noise-based, AGR – area-based and skewness-based).

The fitting result for the damaged clutch is shown in Figure 13. B-spline with 50x20 control point grid was used. The control point mesh is redistributed along the geometric features, which is

significant for the damaged part (D) and the hexagonal bolts (HB). It can also be noted that the control point grid is relatively smooth on the flat (featureless) parts of the geometry.

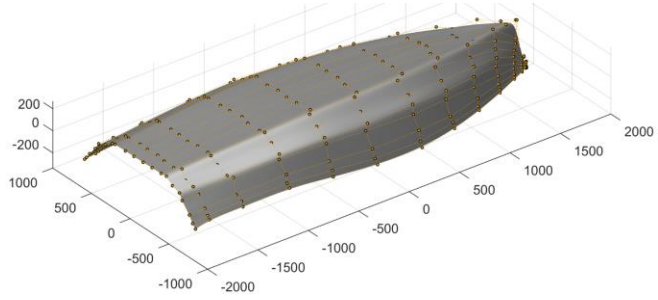


Figure 12: Boat hull fitting with noise-based convergence criterion on the (iteration = 18).

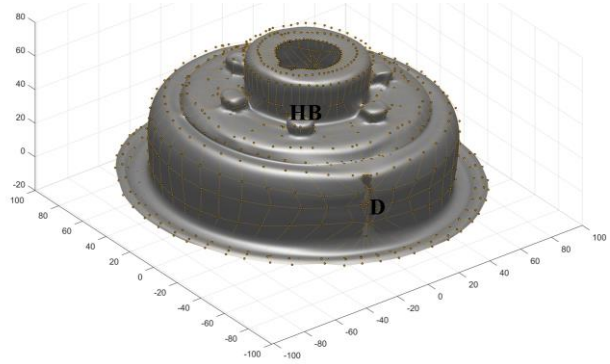
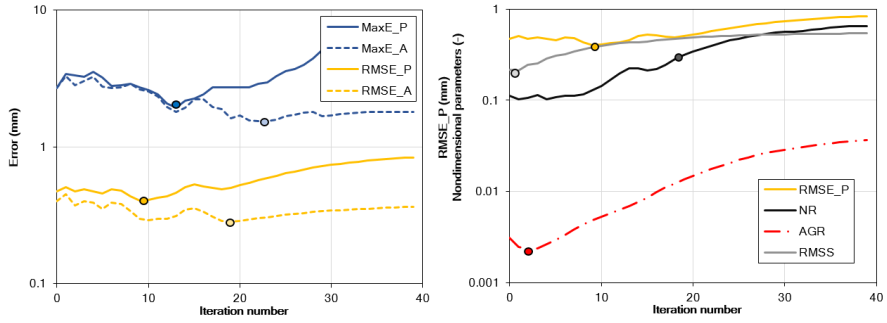


Figure 13: Damaged clutch housing on 22 iteration (50x20 B-spline surface).

The important results regarding the convergence criteria are shown in Figure 14. The convergence according to the actual distance-based error is at iteration 19 and 22 for RMSE and maximum error respectively. The closest to this value is the noise-related criterion which was achieved at iteration 19. The remaining criteria are reached at significantly lower iterations



a) b)

Figure 14: Iterative fitting of damaged clutch housing: a) Fitting error based on P – equal-parameter distance and A – actual distance, b) Different considered convergence criteria (NR – noise-based, AGR – area-based and skewness-based).

The final example is the selected non-engineering object – a part of a small statue. The objective of this paper is to obtain a method for engineering applications. But still, this example is good to show the robustness of the selected convergence criterion. Using the selected initial relaxation field and the convergence criterion, the resulting control point mesh is visibly concentrated at areas which contain geometric feature (see Figure 15). It is also important to note that the mesh is not highly distorted in the areas with higher concentration of control points.

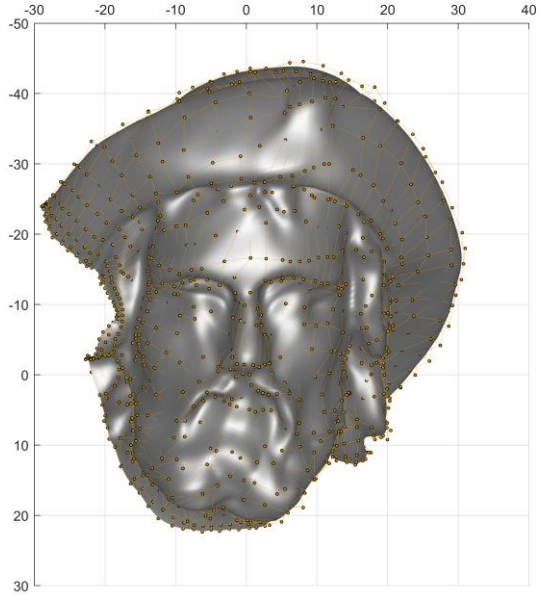
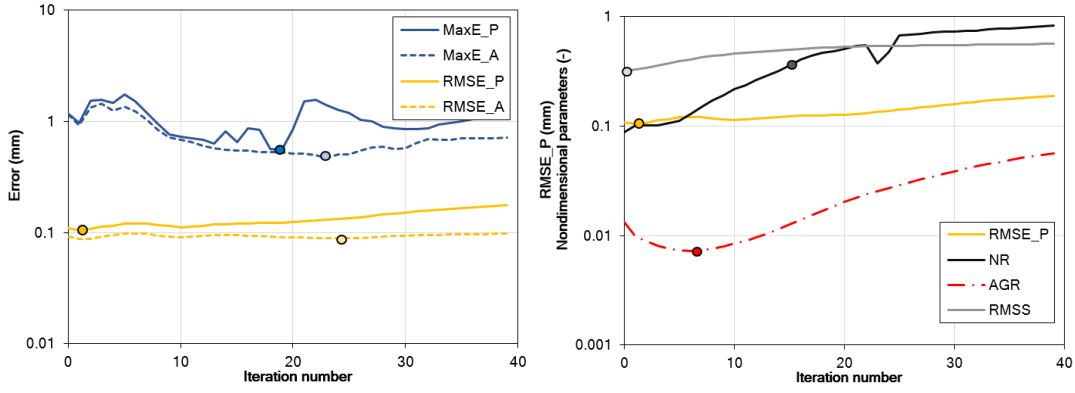


Figure 15: Part of statue at 13th iteration (40x40 B-spline surface).

The overall error results of the iterative method are shown in Figure 16. The best solution regarding actual maximum error and RMSE are at iteration 23. In this case, the equal parameter-based maximum error is at 2nd iteration. The noise related criterion is achieved at iteration 17 which is not far from the actual minimum. But the best criteria here is the maximum parameter-based error, which is achieved at iteration 19.



a) b)

Figure 16: Iterative fitting of the part of statue: a) Fitting error based on P – equal-parameter distance and A – actual distance, b) Different considered convergence criteria (NR – noise-based, AGR – area-based and skewness-based).

4.2 Fitting by Equal-parameter and Actual Distance

For comparison, the fitting was performed by constructing the relaxation surface based on actual error and parametric error respectively. When observing the resulting geometry, the fitting results were nearly the same. Also, when the results were quantified regarding the actual distance between the surface, the results were practically the same. The equal-parameter case yielded surface with mean square error RMSE=1.08 mm and maximum error E_{max}=12.64 mm while for the case of using the actual distance the obtained values were RMSE=1.17 mm and E_{max}=11.33 mm. The comparison of error is shown in Figure 17. It can also be noted that RMSE of around 1 mm and maximum error ~10mm (depending on the relaxation field) is very good in both cases considering that the overall size of the geometry was about 4000 mm.

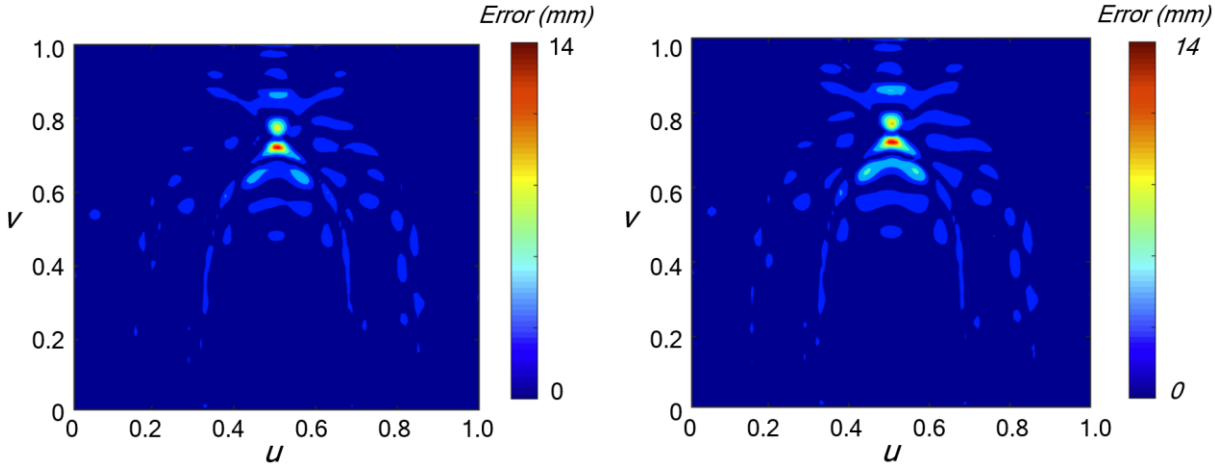


Figure 17: Boat hull fitting error using adaptive fitting iteration with relaxation field based on: a) Using equal-parameter error and b) Using actual distance error.

Similar was valid for the boat propeller geometry as shown in Figure 18 (compare with Figure 3 for full geometry). Here, the fitting results were also practically the same. The RMSE=1.5 mm for equal-parameter based relaxation surface and RMSE=1.1 mm for actual distance-based relaxation surface. This is an important result, as it shows that the method which is computationally more intensive does not obtain significantly better results. A more important advantage of the actual distance-based relaxation surface might be in the smoother result of the relaxation field and less noise (see Figure 14). This would lead to a more robust fitting.

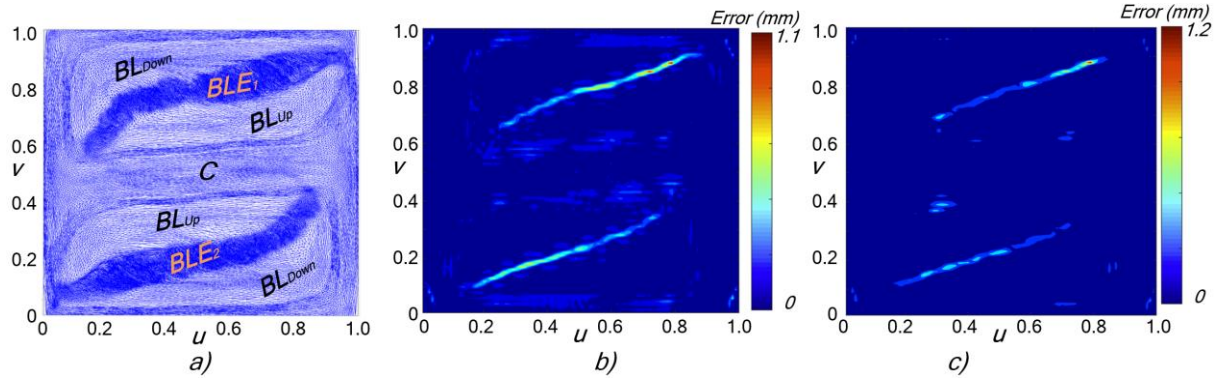


Figure 18: Boat propeller: a) mesh parameterization and with fitting error using relaxation field based on b) Using equal-parameter error and c) Using actual distance error.

4.3 Computational Time

The computational time is one of the most important considerations. The results are now presented for the boat fitting by 35x35 B-spline and using the 100x100 fitting grid (and FEM grid). The calculations were performed on a PC with 3.00 GHz Intel Pentium G2030 CPU.

Most of the computational time is spent solving the linear fitting equations (solving the minimum of equation 2.2), which amount to 2.5 s/iteration. The remaining steps of the algorithm are relatively fast, the solution to the FEM problem requires about 0.4 s/iteration (step 9 of the modified adaptive fitting algorithm presented in Section 3). Next is the trilinear interpolation required in the step 3 and step 10 of the modified adaptive fitting algorithm, which requires about 0.3 s/iteration. The time required for calculation of the relaxation surface (and the remaining intermediate steps) have negligible computational requirements.

When using the fitting based on relaxation surface based on the actual distance, the computational time required for calculation of the actual distance is the most demanding part of the algorithm. The implemented method was a robust minimization method (pattern search) which efficiently avoids local minima, but at a price of being slow. Gradient-based methods were attempted but they lack the required robustness (do not provide good result in most cases). The required time just for calculation of the actual distance was 22.2 s/iteration. If this process can be significantly improved by a robust method, it would become feasible to use this as a convergence criterion. Currently, this was not achieved which was the main reason for investigating different convergence criteria in this paper.

4.4 Summary of Different Relaxation Fields and Convergence Criteria

The summary of the results which includes all the test geometries, different relaxation surfaces and different convergence criteria is given in Table 1. Each row represents a "case", i.e., single geometry with selected relaxation surface (see Section 3.2 and 3.3). For each case, the best convergence criterion is shown in bold. As it can be seen, in most cases the noise-based criterion or the maximum parameter-based error were the best. But, the area-based criterion was better in some cases. Here, it is important to note that in the cases where the area-based criterion was not better regarding the maximum error, the remaining criteria provided substantially better results. So, the noise-based criterion or the maximum parameter-based criterion seams as a more robust option. The criteria based on equal parameter-based RMSE provided larger error in all cases. In a general case, the maximum parametric error might be the best suited criterion, as it is easy to define and calculate. The errors based on actual distance (A) are probably the best indicators of the fitting performance, but they are computationally difficult to calculate.

Regarding the relaxation field, several different variations are shown. For each geometry, the results with the minimal error is shown in italic. Also, the minimum B-spline skewness (RMSS) is shown in italic. In all cases, the minimal error and minimal skewness were obtained by different relaxation fields. In most cases, the best result regarding the minimal error were based on calculation of actual distance. If this is not practical to use, the best alternative in average was: MEG·Af·NGM·E. In all the cases it provided almost the best results, regarding both the fitting error and the surface quality (RMSS). It can also be seen that a slight improvement of RMSS with a slight loss in fitting quality can be achieved by adding the skewness related term to the relaxation field.

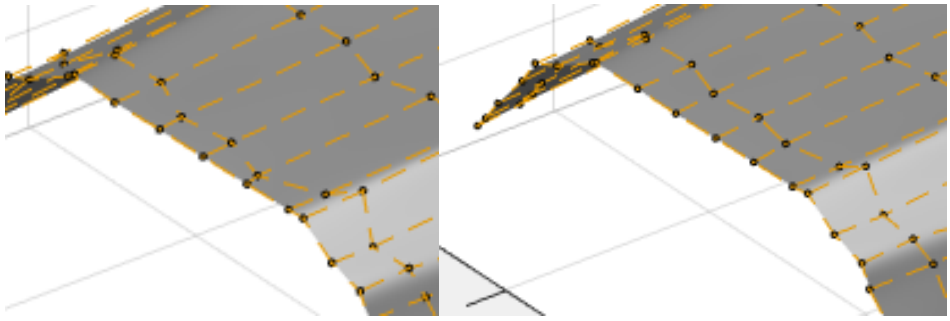
4.5 Removing Noise from the Relaxation Field

Several variations of the relaxation field were tested in section 3.2 and the resulting geometry of the finally selected variation of the relaxation field is shown in Figure 19a. Here, we performed final considered variation. Here, we consider removing the noise from the relaxation field using the same filter as earlier. The fitting results are shown in Figure 19. If closely observed, it can be seen that the B-spline control point grid has a more straight-lined segments when the noise is removed using the previous filter. These differences might be important in some cases with large noise in

the geometry obtained by 3d scanning. In the cases that we considered, the differences were small.

Geometry		Actual fitting error (mm)		Surface quality	MaxE_A obtained by different convergence criteria (mm)				RMSE_A obtained by different convergence criteria (mm)			
	Relaxation field	MaxE_A	RMSE_A	RMSS	RMSE_P	MaxE_P	NR	AGR	RMSE_P	MaxE_P	NR	AGR
Boat hull	NGM	9.28	1.01	0.77	27.00	**12.34**	17.46	40.53	1.41	**1.01**	1.19	1.75
	NMG·Af	6.47	1.04	0.54	40.53	**16.25**	16.34	40.53	1.75	1.30	**1.10**	1.75
	MEG·Af	11.65	1.40	0.52	24.38	**13.57**	20.55	54.77	1.51	**1.40**	1.44	2.62
	c=MEG·Af·NGM	7.94	0.92	0.60	42.15	14.98	**9.15**	40.53	1.72	**0.92**	1.22	1.75
	c·E	6.67	0.82	0.54	36.83	10.01	**7.32**	41.80	1.65	0.97	**0.82**	1.71
	c·E·(1-skew)	7.90	1.03	*0.49*	38.13	**7.90**	17.52	42.02	1.68	**1.14**	2.11	1.71
	c (A)	8.52	0.92	0.60	42.06	15.58	**9.43**	40.53	1.72	**0.92**	1.22	1.75
	c·E (A)	*6.01*	*0.81*	0.59	37.93	6.98	**6.62**	42.99	1.71	0.99	**0.89**	1.73
	c·E·(1-skew) (A)	6.35	1.02	0.50	39.09	21.45	**9.66**	43.08	1.73	1.58	**1.10**	1.73
Boat propeller	NGM	3.64	0.19	0.65	14.64	6.72	**3.64**	12.39	0.31	**0.20**	0.21	0.28
	NMG·Af	8.42	0.39	*0.58*	16.30	16.30	**14.86**	24.58	0.42	**0.42**	1.03	1.55
	MEG·Af	1.02	0.13	0.60	1.36	1.13	**1.13**	1.39	0.14	0.17	**0.17**	0.21
	c=MEG·Af·NGM	4.88	0.26	0.83	16.30	16.30	7.38	**5.84**	0.42	**0.42**	0.47	0.51
	c·E	0.61	0.08	0.63	0.71	0.64	**0.61**	0.63	0.08	0.09	**0.09**	0.09
	c·E·(1-skew)	1.11	0.10	0.59	1.54	1.41	1.28	**1.13**	0.11	0.10	**0.10**	0.10
	c (A)	4.42	0.23	0.72	16.30	11.29	**6.57**	11.29	0.42	0.47	**0.42**	0.47
	c·E (A)	*0.42*	*0.07*	0.64	1.36	**0.76**	0.85	0.82	0.08	0.09	**0.08**	0.08
	c·E·(1-skew) (A)	0.80	0.09	0.61	2.42	1.59	**0.80**	1.31	0.10	**0.09**	0.10	0.09
Clutch housing	NGM	2.23	0.33	0.61	3.48	2.68	3.21	**2.68**	0.33	0.40	0.51	**0.40**
	NMG·Af	2.54	0.36	0.43	3.12	2.68	**2.54**	3.12	0.36	0.40	0.42	**0.36**
	MEG·Af	1.90	0.30	*0.39*	2.04	**1.90**	2.20	2.48	0.30	**0.31**	0.35	0.39
	c=MEG·Af·NGM	1.72	0.32	0.52	3.15	2.34	**2.23**	2.68	0.32	0.35	**0.35**	0.40
	c·E	*1.48*	0.27	0.47	2.56	**1.48**	2.00	2.74	0.29	**0.27**	0.29	0.37
	c·E·(1-skew)	1.53	0.28	0.51	2.61	**1.81**	1.88	2.81	0.30	0.31	**0.29**	0.37
	c (A)	1.83	0.32	0.52	3.14	2.34	**2.20**	2.68	0.32	0.35	**0.35**	0.40
	c·E (A)	1.53	*0.26*	0.49	2.47	**1.66**	1.99	3.21	0.30	**0.26**	0.31	0.38
	c·E·(1-skew) (A)	1.62	0.28	0.54	2.59	1.87	**1.64**	3.19	0.28	**0.29**	0.34	0.38
Part of statue	NGM	0.56	*0.07*	0.61	0.83	0.66	**0.63**	0.88	0.08	**0.07**	0.10	0.08
	NMG·Af	0.78	0.09	*0.48*	0.94	**0.79**	0.83	0.94	0.09	0.09	0.10	**0.09**
	MEG·Af	0.73	0.09	0.49	1.17	**0.73**	0.77	0.76	0.09	0.11	0.11	**0.10**
	c=MEG·Af·NGM	0.62	0.09	0.57	1.15	**0.66**	0.74	1.01	0.09	0.10	**0.10**	0.11
	c·E	*0.40*	0.09	0.59	0.95	**0.48**	0.53	1.18	0.09	0.09	**0.09**	0.10
	c·E·(1-skew)	0.47	0.09	0.54	0.93	**0.52**	0.58	1.23	0.09	**0.09**	0.09	0.10
	c (A)	0.64	0.09	0.57	1.15	**0.65**	0.74	1.03	0.09	0.10	**0.10**	0.11
	c·E (A)	0.40	0.09	0.60	0.97	**0.48**	0.55	1.10	0.09	0.10	**0.09**	0.09
	c·E·(1-skew) (A)	0.45	0.09	0.55	0.99	**0.49**	0.62	1.18	0.09	**0.09**	0.10	0.09

Table 1: Summary of fitting errors for different relaxation fields and convergence criteria (MaxE – maximum error, RMSE – root mean square error, P – equal-parameter distance, A – actual distance, NR – noise ratio-based, AGR – area-based, E-error, skew – skewness, RMSS -root mean skewness squared).



a) n)

Figure 19: Part of the boat hull fitting for relaxation field: a) Without removing noise and b) with removed noise.

4.6 Comparison to Feature-based Fitting

When comparing the proposed variation of the adaptive parameterization method to the feature-sensitive fitting method [9], first it can be noted that the feature-sensitive method is about 20 times faster. In the feature-sensitive method, the result depends on a single parameter which is selected by user. Here, several different values were tried and the best obtained results are presented (several iterations of the method are required to obtain the appropriate parameter values). The results are mean square error RMSE=1.94 mm^2 and maximum error Emax=15.23 mm. Compare this to the results of the adaptive fitting method where mean square error RMSE=1.03 mm and maximum error E_{max}=7.90 mm. Also, when the fitted surface (Figure 20) is compared to previous results (Figure 11), it can be noted that the control points are highly concentrated at the sharp edges. The smoothness of the control-point grid greatly reduced, and any eventual user modifications are thereby difficult.

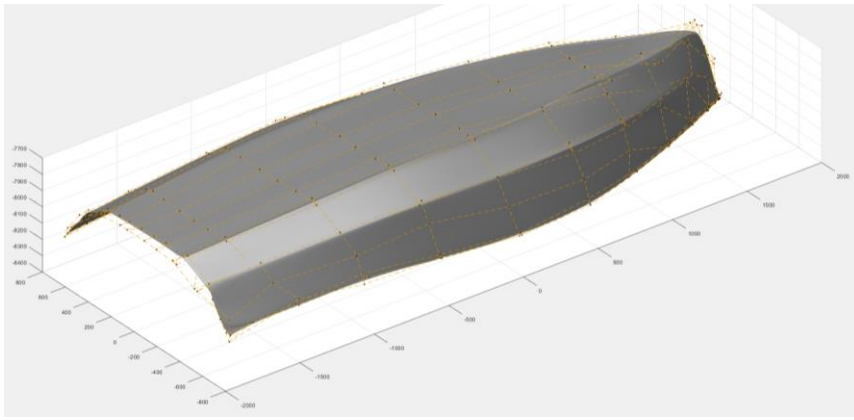


Figure 20: Boat hull fitted using feature-sensitive fitting method.

5 CONCLUSIONS

The paper investigated novel variations of adaptive re-parameterization method for fitting B-spline surface to four different point-clouds obtained by 3D scanning. Good fitting of a smooth B-spline surface was shown possible on complex single-patch surfaces. The method has several factors which can be selected differently. Here, we considered different relaxation fields and different convergence criterion. Among different relaxation fields, the one which uses a product of values

related to surface normal, surface area and fitting error has shown the best results. Even when considering actual distance instead of equal-parameter error, the previous method obtained better results. This is a desirable result since the calculation of actual distance (minimum distance) between the point cloud and the fitted surface is computationally expensive. As the method is iterative, one of the main problems was to select an appropriate criterion for convergence of the method. It was shown that equal parameter-based mean square error is not suited for convergence criterion. A method based on noise-related criterion was proposed, and it performed much better in the performed tests. A simpler criterion based only on maximum parameter-based error performed almost equally well. As the future work, this convergence methods would have to be tested on larger variety of different geometry to prove the robustness. The same is valid for the selected relaxation field. As there exist large flexibility in the selection of appropriate relaxation field, future work might consider further variations. Also, the method can be used as a part of an engineering shape-optimization procedure which requires local B-spline mesh refinement and a constant set of shape variables.

ACKNOWLEDGEMENTS

This work was supported by the Croatian Science Foundation [HRZZ-IP-2018-01-6774].

Ivo Marinić-Kragić, https://orcid.org/0000-0002-9680-6193
Damir Vučina, https://orcid.org/0000-0003-2003-5481

REFERENCES

[1] Várady, T.; Martin, R. R.; Cox, J.: Reverse engineering of geometric models—an introduction. Comput-Aided Design, 29 1997, 255–68. https://doi.org/10.1016/S0010-4485(96)00054-1.
[2] Weiss, V.; Andor, L.; Renner, G.; Várady, T.: Advanced surface fitting techniques, Comput Aided Geom Des, 19, 2002, 19–42. https://doi.org/10.1016/S0167-8396(01)00086-3.
[3] Vaitkus, M.; Várady, T.: A Labeling Algorithm for Trimmed Surface Fitting, Comput Aided Des Appl, 16, 2018, 720–32. https://doi.org/10.14733/cadaps.2019.720-732.
[4] Floater, M. S.: Parametrization and smooth approximation of surface triangulations, Comput Aided Geom Des, 14, 1997, 231–50. https://doi.org/10.1016/S0167-8396(96)00031-3.
[5] Floater, M. S.; Reimers, M.: Meshless parameterization and surface reconstruction, Comput Aided Geom Des, 18, 2001, 77–92. https://doi.org/10.1016/S0167-8396(01)00013-9.
[6] Remacle, J-F.; Geuzaine, C.; Compère, G.; Marchandise, E.: High-quality surface remeshing using harmonic maps, Int J Numer Methods Eng, 2010:n/a-n/a. https://doi.org/10.1002/nme.2824.
[7] Greiner, G.; Hormann, K.: Interpolating and Approximating Scattered 3D-data with Hierarchical Tensor Product B-Splines, Surf. Fitting Multiresolution Methods, Vanderbilt University Press, 1997, 163–72.
[8] Becker, G.; Schäfer, M.; Jameson, A.; Schafer, M.; Jameson, A.; Schäfer, M.; et al.: An advanced NURBS fitting procedure for post-processing of grid-based shape optimizations, 49th AIAA Aerosp Sci Meet Incl New Horizons Forum Aerosp Expo 2011:1–19. https://doi.org/10.2514/6.2011-891.
[9] Lai, Y-K.; Hu, S-M.; Pottmann, H.: Surface fitting based on a feature sensitive parametrization, Comput-Aided Design, 38, 2006, 800–807. https://doi.org/10.1016/j.cad.2006.04.007.
[10] Marinić-Kragić, I.; Ćurković, M.; Vučina, D.: Adaptive re-parameterization based on arbitrary scalar fields for shape optimization and surface fitting, Eng Appl Artif Intell, 67, 2018, 39–51. https://doi.org/10.1016/j.engappai.2017.09.004.
[11] Marinić-Kragić, I.; Perišić, S.; Vučina, D.; Ćurković, M.: Superimposed RBF and B-spline parametric surface for reverse engineering applications, Integr Comput Aided Eng, 27, 2019, 17–35. https://doi.org/10.3233/ICA-190611.
[12] Piegl, L.; Tiller, W.: The NURBS Book, Berlin, Heidelberg, Heidelberg: Springer Berlin

Heidelberg; 1995. https://doi.org/10.1007/978-3-642-97385-7.
[13] Krishnamoorthy, G. S.; Krishnamoorthy, C. S.: Finite Element Analysis: Theory and Programming, McGraw-Hill Companies; 1995.