Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 00 (2017) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

26th International Meshing Roundtable, IMR26, 18-21 September 2017, Barcelona, Spain

Comparison of 2D boundary curving methods with modal shape functions and a piecewise linear target mesh

V. S. Ziel[a,*], H. Bériot[b], O. Atak[b], G. Gabard[c]

[a] *Institute of Sound and Vibration Research, University of Southampton, SO17 1BJ Southampton, United Kindom*
[b] *Siemens Industry Software NV, Simulation and Test Solutions, Interleuvenlaan 68, B-3001, Leuven, Belgium*
[c] *Laboratoire d'Acoustique de l'Université du Maine, CNRS, LAUM UMR 6613, Av. O. Messiaen, 72085 Le Mans, France*

Abstract

It is well known that high-order simulation techniques demand an accurate geometric representation and a coarse mesh. To fulfill both requirements, curved meshes are generated. In most cases, curving methods assume that the exact geometry is known. But it can be useful to develop curving methods with only a limited knowledge of the target geometry. In this paper, three curving methods are described that take a piecewise fine linear mesh as input: a least squares approach, a direct optimisation in the H^1-seminorm, and a H^1-seminorm optimisation in a reference space. Hierarchic, modal shape functions are used as basis for the geometric approximation. The methods are compared on two test geometries, a unit circle and a distorted ellipse. Considering both test cases, the direct optimisation approach shows the most promising results. Finally, the main steps for the extension to 3D are outlined.

Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.

Keywords: Mesh Curving; High-Order Methods; Integrated Legendre Polynomials; Lobatto Polynomials; Linear Target Mesh

1. Introduction

In the literature, it is well acknowledged that high-order simulation techniques require an accurate description of the simulation geometry [1,2]. Typically, high-order polynomial interpolation is used to obtain curved mesh boundaries that allow for better geometrical discretisation than linear boundaries. The generation and optimisation of high-order meshes is an active field of research. Many curving techniques follow an a-posteriori scheme (e.g. [3,4]): First, a linear mesh is generated, using an established linear meshing algorithm. The boundary of this mesh is then curved towards the target geometry. Finally, optimisation and untangling procedures can be applied to obtain a valid mesh for numerical simulations or to increase the element quality.

In general, the target geometry is assumed to be known exactly. However, in many practical situations, for instance considering simulation companies acting as subcontractors, the original CAD-geometry is not accessible. Instead,

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: vss1g14@soton.ac.uk

Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.

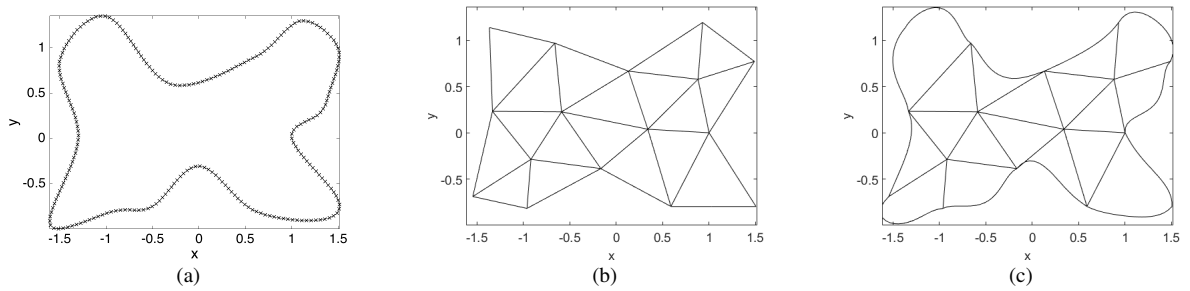

Fig. 1. Schematic of (a) the fine linear Mmesh of the boundary, (b) the coarse linear Cmesh of the domain, and (c) the curved Cmesh.

a fine linear description of the geometry is often provided, or could be still available from previous simulations. In biomedical applications, geometries are often described by scanned data only that form a cloud of points [5].

This paper compares a-posteriori curving methods that take a fine linear mesh as target curve. This target curve will be called the *model mesh* or *Mmesh*, and will be denoted as $\mathcal{M}$. As its only function is to provide an improved description of the geometric boundary, it is only a mesh of the domain boundary. The boundary of the coarse mesh that is curved during the procedure will be called *computational mesh* or *Cmesh*, and will be denoted C. In a preprocessing step, C could be obtained by applying a coarsening algorithm to $\mathcal{M}$. The algorithm should ensure that areas of high curvature are less coarsened, whereas areas with low curvature allow for larger Cmesh elements. In this paper, we only look at the curving step. The input C is assumed to be of appropriate mesh quality. The principle of the Cmesh and Mmesh is illustrated in Figure 1.

Each curved C element is described as a linear combination $\hat{\boldsymbol{f}} = \sum_{i=1}^{n_q} \boldsymbol{\alpha_i} b_i$ of curving order q, with a polynomial basis $\{b_i\}_{i=1}^{n_q}$ and coefficients $\boldsymbol{\alpha_i}$. These coefficients will be determined by the curving processes described in this paper. As bases $\{b_i\}_{i=1}^{n_q}$, integrated Legendre polynomials are used

$$
\begin{aligned}
l_0(\xi) &= \tfrac{1-\xi}{2}, \\
l_1(\xi) &= \tfrac{\xi+1}{2}, \quad \text{and} \\
l_k(\xi) &= \tfrac{1}{\|L_{k-1}\|_2} \int_{-1}^{\xi} L_{k-1}(x)\,\mathrm{d}x, \quad \text{for } 2 \le k,\ \xi \in [-1,1]\,,
\end{aligned}
$$

where $L_i(\xi)$ is the i^{th} Legendre polynomial. These polynomials are also called *Lobatto polynomials* and have several advantageous properties which justify their consideration [6]:

- In 2D, the basis functions fulfill the equation

$$
\int_{-1}^{1} l_i'(\xi) l_j'(\xi)\,\mathrm{d}\xi = \delta_{ij}, \ \text{with } \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{else} \end{cases}, \tag{1}
$$

  which will be called the *δ-property*. It is based on the orthogonality of the Legendre shape functions in the L^2-norm. This leads to explicit formulas for the $\boldsymbol{\alpha_i}$ if they are obtained by optimisation in the H^1-seminorm. In 3D, this property is not conserved for triangular face shape functions, but the construction of orthonormal bubble shape functions is described in [7]. This paper only considers 2D cases.
- The shape functions form hierarchical bases. Together with the δ-property, this results in hierarchical coefficients α_i, meaning that for an increase of the q by one, only one additional $\boldsymbol{\alpha_i}$ needs to be calculated.
- Furthermore, the Lobatto bases are modal bases that do not rely on node positions. This simplifies the embedding of the curved boundary for full simulation meshes, as no element internal nodes have to be placed. In combination with the hierarchical property, the modal structure also allows for simpler local adaption of the curving order.

Three a-posteriori modal curving methods in 2D are further examined hereafter. The first is a least squares approach, where only Mmesh vertices that lie on the exact geometry are used to describe the target curve. For the other two methods, an optimisation with respect to the H^1-seminorm is calculated in two different function spaces. These

optimisations take full advantage of the δ-property of the Lobatto shape functions and do not include any matrix inversion. But they rely on the linearly interpolated Mmesh elements that only form an approximation of the exact target geometry. For all three methods, the curving is done one Cmesh element at a time. It is assumed that all Cmesh vertices coincide with Mmesh vertices. This leads to a straightforward assignment of Mmesh vertices and edges to the Cmesh edges.

The three methods are compared with two test cases: A unit circle described by 12 Cmesh elements, and a distorted ellipse, described by 20 Cmesh elements. For both cases, the geometric discretisation error (GDE) and its convergence for an increasing geometric order q are reported.

The structure of the paper is as follows: In Section 2, the three curving methods are explained. The test cases are described and evaluated in Section 3.

2. Curving methods

In this section, the three mesh curving algorithms are presented. As described in the introduction, they are all based on a modal approximation. The exact target geometry is known only pointwise. For the two curving methods based on a H^1-seminorm optimisation, these points are interpolated linearly.

The polynomial approximation $\hat{\boldsymbol{f}}(\boldsymbol{\xi})$ can be split up into a known linear part and a non-linear part. The linear part is fully described by the Cmesh vertex positions $\boldsymbol{p_1} = (p_{1,x}, p_{1,y})$ and $\boldsymbol{p_2} = (p_{2,x}, p_{2,y})$ that lie on the exact geometry. Therefore, the $\boldsymbol{\alpha_0}$ and $\boldsymbol{\alpha_1}$ are known and only $\boldsymbol{\alpha_j}$ for $j = 2, \dots, q$ for the non-linear part have to be determined. The non-linear part will be called the *polynomial deflection*.

$$\hat{\boldsymbol{f}}(\xi) = \underbrace{\frac{1}{2}(-\xi+1)\boldsymbol{p_1} + \frac{1}{2}(\xi+1)\boldsymbol{p_2}}_{\text{linear part}} + \underbrace{\sum_{i=2}^{q} \boldsymbol{\alpha_i} \boldsymbol{l_i}(\xi)}_{\text{polynomial deflection}} .$$

2.1. Least squares modal curving

Least squares is a standard method that minimises the squared distance between a target and its approximation. Here, the pointwise squared distance between the Mmesh vertices and corresponding points on the Cmesh is evaluated. This way, only the exact geometry is taken as the target curve, as the Mmesh vertices lie on the exact geometry. No interpolation between the exact points is needed. A similar approach using Bézier curves is described in [5].

For the curving of a single Cmesh element, the n corresponding Mmesh vertices (x_i, y_i) are approximated by the polynomial $\tilde{\boldsymbol{f}} = \sum_{j=0}^{q} \boldsymbol{\alpha_j} \boldsymbol{l_j}$. The coefficients $\boldsymbol{\alpha_j}$ are determined such that they fulfill the minimisation

$$\sum_{i=1}^{n} \|(x_i, y_i) - \tilde{\boldsymbol{f}}(\xi_i)\|_2^2 \rightarrow min_{\boldsymbol{\alpha}},$$

with $\xi_i \in [-1, 1]$ is the parameter corresponding to (x_i, y_i) for a element length based parametrisation of the Mmesh arc.

It is well known that the least squares approach results in a linear system of equations $\boldsymbol{A\alpha} = \boldsymbol{b}$. Both the x- and y coordinate are described independently, which leads to two independent linear systems, one for each coordinate, with $\boldsymbol{\alpha_i} = \left(\alpha_{i,x}, \alpha_{i,y}\right)$. Here the system for the x-coordinate is shown. The subindex x is dropped for readability.

$$\boldsymbol{A} = \begin{pmatrix} a_{2,2} & \cdots & a_{2,q} \\ \vdots & \ddots & \vdots \\ a_{q,2} & \cdots & a_{q,q} \end{pmatrix}, \quad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_2 \\ \vdots \\ \alpha_q \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} b_2 \\ \vdots \\ b_q \end{pmatrix}, \text{ with}$$

$$a_{k,j} = \sum_{i=1}^{n} l_k(\xi_i) l_j(\xi_i),$$

$$b_k = \sum_{i=1}^{n} x_i l_k(\xi_i) - \sum_{j=0}^{1} \alpha_j \sum_{i=1}^{n} l_k(\xi_i) l_j(\xi_i).$$

The system for the y-coordinate follows analogously.

2.2. Direct modal curving

A second modal based curving method is presented, where a projection based interpolation [6] is applied to obtain the $\boldsymbol{\alpha_i}$ coefficients in the polynomial description of the curved edges. Thereby, the δ-property (1) of the Lobatto shape functions is used, by considering an optimisation in the H^1-seminorm. In this optimisation, the difference between the target curve and the approximating curve is minimised in each coordinate independently. As in the least squares approach, only the formula for the x-coordinate will be given, showing the approximation of the target f_x by $\hat{f}_x$. The polynomial approximation of the y-coordinate follows analogously and will result in an additional set of α_i-values.

The target curve is the linear interpolation of the Mmesh vertices. Similarly to $\hat{f}_x(\xi)$, the parametrisation of the target curve can be split up in a linear part, and a deflection d. Here, d is generally non-polynomial and depends on the Mmesh vertices on the target segment. The polynomial approximation is done in the H^1-seminorm by finding α_i such that

$$|f_x - \hat{f}_x|_{H^1} = \int_{-1}^{1} (f_x' - \hat{f}_x{}')^2 \mathrm{d}\xi \rightarrow min_{\boldsymbol{\alpha}}.$$

The α_k that solve the minimisation problem have to satisfy

$$\frac{\partial}{\partial \alpha_k} \int_{-1}^{1} (d' - \sum_{i=2}^{q} \alpha_i l_i'(\xi))^2 \mathrm{d}\xi = 0.$$

This can be reformulated as an explicit formula for the coefficients

$$\alpha_k = \textstyle\int_{-1}^{1} d' l_k' \mathrm{d}\xi.$$

The calculation of α_k depends on the parametrisation of the Mmesh arc that forms the target curve. For this, a physical length based parametrisation will be considered. The lengths of the subintervals in the reference space are adapted to the lengths of the Mmesh elements. The resulting formula for the α_k is

$$\alpha_k = \textstyle\sum_{i=1}^{m-1} \left(\int_{-1+2\frac{\sum_{j=1}^{i-1} h_j}{\sum_{j=1}^{m-1} h_j}}^{-1+2\frac{\sum_{j=1}^{i} h_j}{\sum_{j=1}^{m-1} h_j}} \frac{\sum_{j=1}^{m-1} h_j}{2h_i} (x_{i+1} - x_i) l_k'(\xi) \mathrm{d}\xi \right) + \left(\tfrac{1}{2} p_{1,x} - \tfrac{1}{2} p_{2,x}, \right) \int_{-1}^{1} l_k'(\xi) \mathrm{d}\xi,$$

with m the number of Mmesh vertices on the Mmesh arc and h_j the lengths of the j^{th} Mmesh element on the Mmesh arc. As this algorithm directly approximates the target curve in the physical space, unlike the one that will be presented in the next Section, it will be referred to as *direct modal curving*.

2.3. Reference-based modal curving algorithm

The third considered method will be called *reference based modal curving*. This method was previously described in [8]. As in the direct approach, a projection based interpolation is applied to obtain the polynomial interpolation coefficients. But for this method, the interpolation takes place in a (ξ, η)-reference space. Thanks to this transformation, only one set of α_i for the η-coordinate approximation has to be determined. The mapping to the reference space is designed such that the linear Cmesh element corresponds to the $[-1, 1]$ interval on the ξ-axis. The Mmesh vertices are mapped onto this reference space and are then linearly interpolated to obtain a target curve. After the curving, the resulting curve is mapped back into the physical space. In Figure 2, the method is illustrated.

The mapping $\boldsymbol{F}_Q(\xi, \eta)$ linking the reference and the physical space is defined by the mapping of the standard $[(-1, -1), (1, 1)]$ reference quadrangle to a straightsided quadrangle in the physical space. The quadrangle in the physical space is formed by an extrusion of the Cmesh vertices in positive and negative vertex normal direction by a predefined extrusion length. The extruded points form a quadrangle element $\boldsymbol{Q} = (\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3, \boldsymbol{q}_4)$ around the straight sided Cmesh element. It is assumed that the quadrangles of different elements are distinct, that their union contains the whole Mmesh, and that they are untangled.

All Mmesh vertices $\boldsymbol{v_i}$ enclosed by the bounding interval $I = \left[\min_j \left(\boldsymbol{q}_j\right)_x, \max_j \left(\boldsymbol{q}_j\right)_x\right] \times \left[\min_j \left(\boldsymbol{q}_j\right)_y, \max_j \left(\boldsymbol{q}_j\right)_y\right]$ are mapped onto the reference space as $(\xi_i, \eta_i) = \boldsymbol{F}_Q^{-1}(\boldsymbol{v_i})$. Generally, no closed-form expression is available for the inverse mapping $\boldsymbol{F}_Q^{-1}(x, y)$. Instead, the (ξ_i, η_i) are obtained by optimisation.

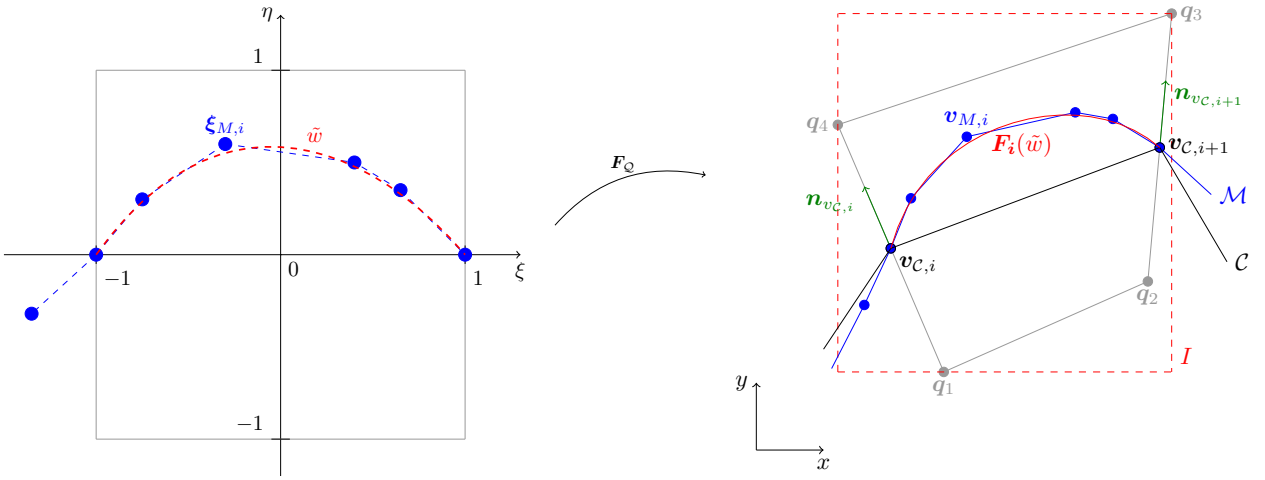

Fig. 2. Schematic of the reference based modal curving algorithm

The target curve w in the reference space is formed by linear interpolation of the mapped Mmesh vertices. As $\boldsymbol{F}_Q$ is generally not linear, the image $\boldsymbol{F}_Q(\xi, w(\xi))$ of w does not necessarily correspond to $\mathcal{M}$. The linear interpolation only approximates the inverse image of the $\mathcal{M}$.

The target curve is then approximated by a polynomial interpolation $\tilde{w} = \tilde{w}(\boldsymbol{\alpha}) = \sum_{i=0}^{q} \alpha_i l_i$. The coefficients $\boldsymbol{\alpha} := \left[\alpha_0, \ldots, \alpha_q\right]$ are obtained with an optimisation in the H^1-seminorm

$$\boldsymbol{\alpha} := \operatorname{argmin}_{\boldsymbol{\alpha}} \left| w - \tilde{w}^b \right|_{H^1} = \operatorname{argmin}_{\boldsymbol{\alpha}} \int_{-1}^{1} \left(w - \tilde{w}^b\right)' \left(w - \tilde{w}^b\right)' \, \mathrm{d}\xi,$$

where function $\tilde{w}$ is splitted into the linear term $\tilde{w}^v = \sum_{i=0}^{1} \alpha_i l_i$ and the high-order term $\tilde{w}^b = \sum_{i=2}^{q} \alpha_i l_i$. The linear term $\tilde{w}^v$ vanishes because $w(-1) = w(1) = 0$, which uniquely determine $\alpha_0 = \alpha_1 = 0$. Analogously to the direct modal method, the projection based interpolation results in an explicit formula

$$\alpha_i = \int_{-1}^{1} w'(\xi) l_i(\xi) \, \mathrm{d}\xi.$$

3. Comparison of the methods

The described curving methods are tested and compared on two test cases, the unit circle and a distorted ellipse.

3.1. Circle test case

3.1.1. Geometry and meshes

First, the methods are tested on the unit circle. As a reference, the same meshes are curved with the nodal approach implemented in *Gmsh* [9]. Therefore, the target curve is defined by four quadrants, using the *Gmsh*-internal command *Circle*. A mesh with evenly sized elements is created and then curved. The curving algorithm is described in [10]. The Cmesh and Mmesh for the tested curving methods are generated the same way. In order to fulfill the assumption that the Cmesh vertices are a subset of the Mmesh vertices, the Cmesh vertices are snapped onto the closest Mmesh vertices.

3.1.2. GDE measure for the comparison

The accuracy of the curving results is evaluated as the area A between the approximating curve and the exact circle. For this test case, the polar to cartesian coordinates transformation provides an inverse of the exact boundary's

parametrisation. This can be used to calculate the area directly. An approximation of both curves by a polygon as done in [4] is not necessary. Instead, parametrisations $c_i(s)$ and $\hat{c}_i(s)$ of the two curves are constructed elementwise over the same reference interval $s \in [s_0, s_1]$. Using the formula $A_{\text{quad}} = \frac{1}{2}|h_1 \times h_2|$ of a quadrilateral with diagonals h_1 and h_2, the area is increased by

$$A_i(s + \mathrm{d}s) - A_i(s) = \frac{1}{2}\left|(c_i(s + \mathrm{d}s) - \hat{c}_i(s)) \times (c_i(s) - \hat{c}_i(s + \mathrm{d}s))\right|$$

when traveling along the parametrisation of the curves from s to $s + \mathrm{d}s$. Applying linear Taylor series expansions of c_i and $\hat{c}_i$, and rearranging the formula, the ordinary differential equation

$$\frac{\mathrm{d}A_i}{\mathrm{d}s}(s) = \frac{1}{2}\left|(\hat{c}_i(s) - c_i(s)) \times (\hat{c}_i'(s) + c_i'(s))\right|$$

is obtained. To get the area between the two curves over one element, (3.1.2) is solved for the initial value $A_e(s_{0,i}) = 0$. Finally, the elementwise areas A_i are summed up for the total area $A_{\text{abs}} := \sum_i A_i$. A relative area measure of the geometric discretisation error is obtained by dividing A_{abs} by the circumference U_c of the circle $\text{GDE}_A := A_{\text{rel}} := \frac{A_{\text{abs}}}{U_c}$, similar to the measure suggested in [4].

3.1.3. Results

Figure 3 shows q-convergence results of GDE_A for the nodally curved elements and for modally curved elements with 12 Cmesh elements and varying numbers of Mmesh elements. Due to the construction of the two meshes, the number of Mmesh elements per Cmesh element is constant and can be calculated as $\text{MpC} = \frac{n_M^e}{n_C^e}$. The results for the Lagrange elements are included as reference.

The q-convergence plot of the reference-based modal curving approach (see Fig. 3(a)) shows a strong stepwise behaviour. For $q = 2k$ and $q = 2k + 1$, nearly the same GDE_A values are observed. Lobatto basis polynomials of even order are axially symmetric in $\xi = 0$ ($l_{2k}(-\xi) = l_{2k}(\xi)$), whereas those of odd orders are anti-symmetric with respect to the origin ($l_{2k+1}(-\xi) = -l_{2k+1}(\xi)$). Therefore, Lobatto basis polynomials of odd order do not significantly contribute to the arc approximation. A less distinct stepwise behaviour is also observed for the approximations in the physical space, both with Lobatto shape functions and for the reference solution with Lagrange shape functions. This is expected, as both shape function bases span the same polynomial space. For more general, unsymmetric geometries, this behaviour is not expected. The data points for $q = 9$ and $q = 10$ for $\text{MpC} = 5000$ are missing as their computation would demand a long run time. Anyhow, note that the implementation was not optimised for computational costs. Therefore, we will not draw any conclusions or comparisons in terms of run time.

Stagnation is observed for the q-convergence curves for the reference based modal curving approach. With increasing MpC, the stagnation onset shifts to higher q and the stagnation level decreases. From this behaviour it is deduced that the stagnation is introduced and controlled by the geometric accuracy of $\mathcal{M}$. With increasing q, the approximation of $\mathcal{M}$ becomes more accurate and the difference between $\mathcal{M}$ and the exact sphere becomes the dominating error. As this error is fixed with the Mmesh size, the overall error stagnates.

Comparing the results for the cases of the reference-based modal curving with those of the reference Lagrange curving, the latter indicate stronger convergence, leading to a gap in the geometric accuracy of the two methods. The reason for this convergence gap lies in the restricted function space in which the reference-based modal approach approximates the circle. The restriction of the function space comes from the fixed mapping from the reference space to the physical space.

The results for the direct approach are shown in Figure 3(b). As for the reference based approach, the stagnation due to the Mmesh accuracy can be observed. Besides this, the direct approach performs significantly better than the reference based approaches. It does not show the convergence gap observed with the reference based methods, but closely follows the results of the *Gmsh* reference. This is because the polynomial function space within which the H^1-optimisation is run is not restricted as it was described for the reference-based modal approach.

Finally, the results with the least squares approach are shown in Figure 3(c). Since the curving is also run in the physical space without restriction of the polynomial space, the error curve follows the reference curve. As the method considers only the geometrically exact Mmesh vertices, the linear interpolation in between vertices is not taken into account. Therefore, the stagnation induced by the inaccuracy of the Mmesh is overcome. However, it should be

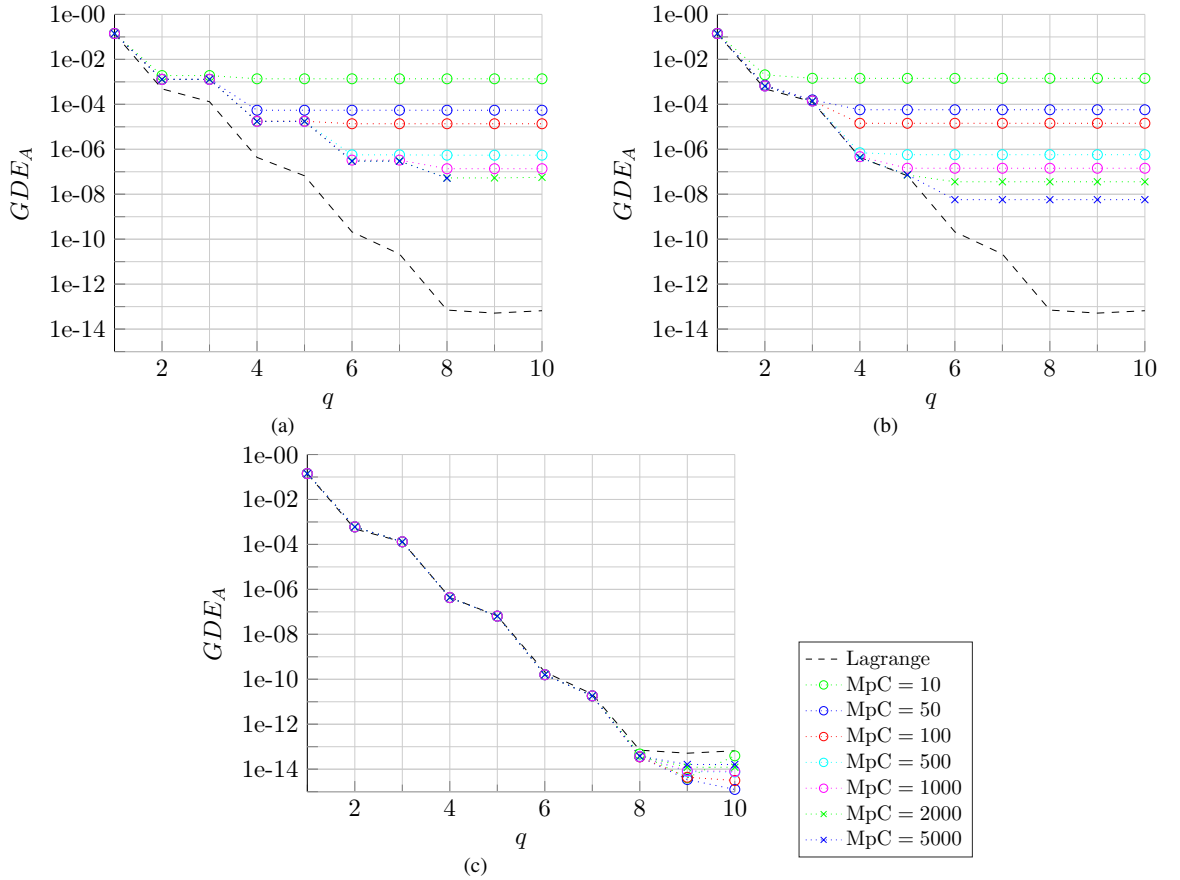

Fig. 3. Plot of the relative area between the exact circle and the approximation curved with (a) the reference based method, (b) the direct modal method, or (c) the least squares approach for different ratios MpC.

emphasized that the method demands Mmesh refinement of q elements per Cmesh element in order to be valid. For high q orders, the curves stagnate at error levels close to machine precision. Thereby, high Mmesh refinements lead to higher errors. It is assumed that this behaviour can be explained by accumulated numerical errors.

3.2. Distorted ellipse

3.2.1. Geometry and meshes

As a second, more complicated test case, a distorted ellipse is considered. The same geometry was also tested in [8]. It is obtained by adding cosine bumps along the curve of a high aspect ratio ellipse

$$\begin{pmatrix} x(\theta) \\ y(\theta) \end{pmatrix} = \begin{pmatrix} (r_x + h_b \cos(\theta n_b)) \cos(\theta) \\ \left(r_y + h_b \cos(\theta n_b)\right) \sin(\theta) \end{pmatrix} \quad \text{for } \theta \in [0, 2\pi] \,, \tag{2}$$

with radii $r_x = 0.5$, $r_y = 5$, $n_b = 10$ the number of bumps and $h_b = 0.1$ their height. Figure 4(a) visualises the geometry.

In general applications, the meshes are not expected to be as regularly sized as those tested on the circle test case. Therefore, the linear meshes on the distorted ellipse are randomly perturbed. Using the *rand* command and seed 28 in *Matlab*, n_v uniformly distributed pseudo random values γ_i in the interval $[-\frac{1}{3n_v}, \frac{1}{3n_v}]$ are created, where n_v is the number of equidistant nodes $\tilde{\theta}_i \in [0, \pi]$ and the first and last value are defined to be zero. The final reference nodes are obtained as $\theta_i = \tilde{\theta}_i + \gamma_i$ and the mesh vertices are then calculated with the parametrisation (2). This constructs the

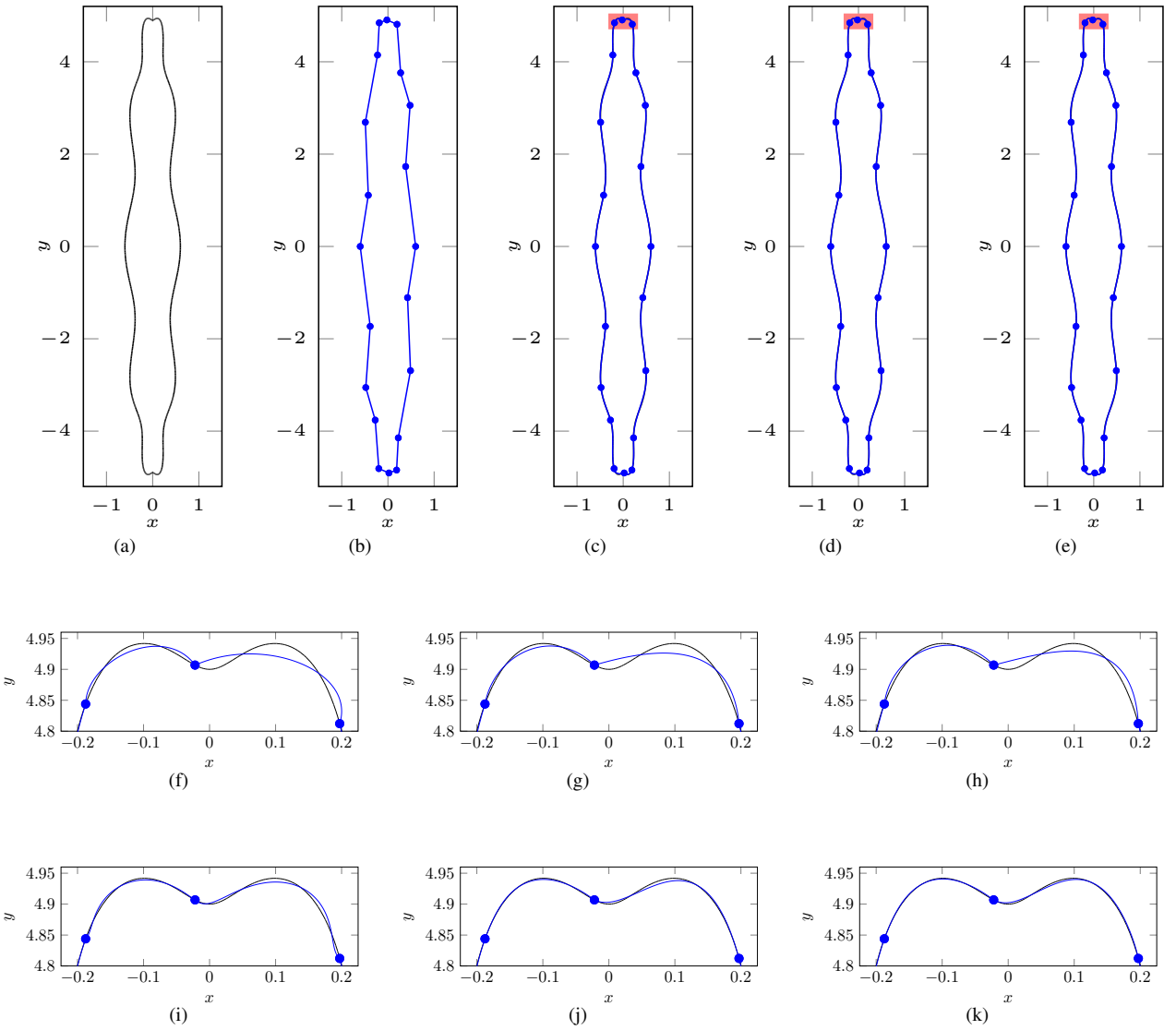

Fig. 4. The distorted ellipse test case. (a) shows the target geometry, (b) the linear Cmesh with 20 elements. Third order curves after the curving are shown for (c) the reference based approach, (d) the direct modal approach and (e) the least squares approach. The red rectangle indicates the zoom-in area which is shown in figures (f-h) for $q = 3$ and (i-k) for $q = 4$. Figures (f) and (i) correspond to the reference based method, (g) and (j) to the direct modal approach and (h) and (k) to the least squares.

upper half of the geometry. For the other half, $\theta_i + \pi$ are mapped onto the physical space. The resulting curve is closed due to the choice of $\gamma_0 = \gamma_{n_v} = 0$. As in the circle test case, the Cmesh vertices are finally snapped onto the closest Mmesh vertices before the actual curving.

3.2.2. GDE measure for the comparison

As for the previous test case, an area based GDE is evaluated. But for this test case, the three Cmesh-Mmesh approaches are compared among each other. Besides, there is no explicit inverse of the parametrisation. Therefore, the area between the high-order curve and the target curve $\mathcal{M}$ is calculated. It is divided into small quadrangles $\mathrm{Q}_i = \left(\mathbf{q}_{i,1}, \mathbf{q}_{i,2}, \mathbf{q}_{i,3}, \mathbf{q}_{i,4}\right)$ with the area

$$A_{\mathrm{Q}_i} = \frac{1}{2}\left|\left(\mathbf{q}_{i,2} - \mathbf{q}_{i,4}\right) \times \left(\mathbf{q}_{i,1} - \mathbf{q}_{i,3}\right)\right|.$$

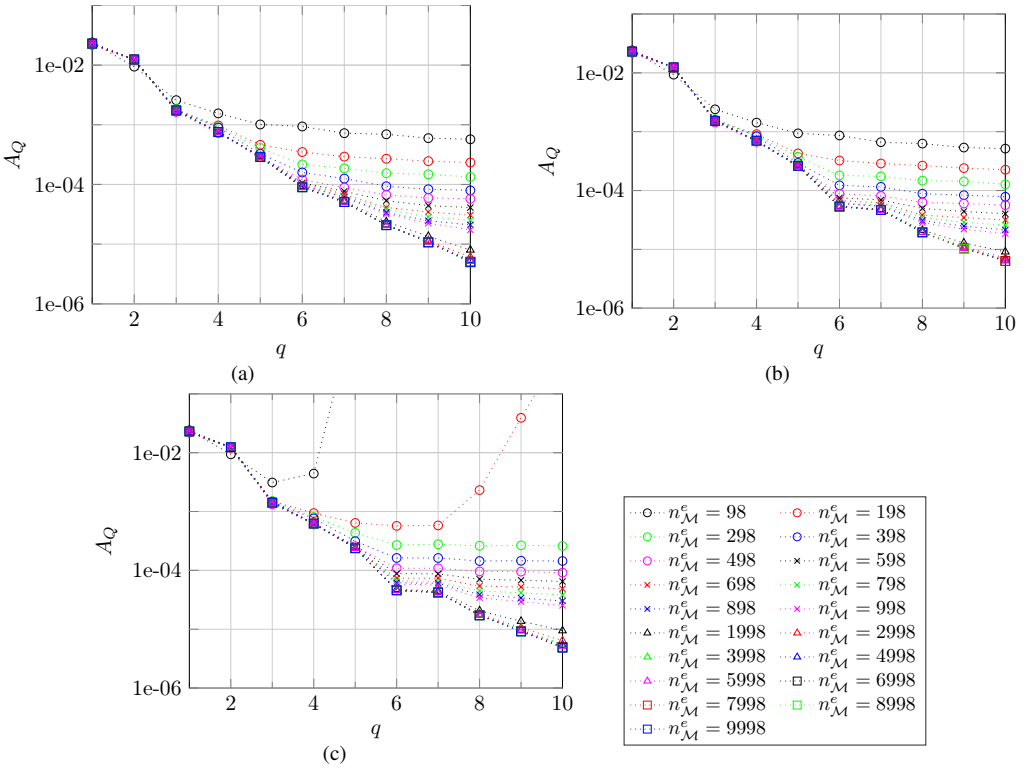

Fig. 5. q-convergence curves for (a) the ref. based modal and (b) the direct modal method, (c) the direct modal least squares, all with fixed $n^e_C = 20$.

Note that the formula only holds for convex quadrangles. Several restrictions on the node positions reduce the non-convex quadrangles, and the quadrangles size is adapted to fulfill a predefined accuracy in the area calculation. To obtain the considered GDE measure, the sum over all quadrangles' areas is divided by the length $l_{\mathcal{M}}$ of $\mathcal{M}$:

$$\mathrm{GDE}_{A_Q} := \frac{\sum_i A_{Q_i}}{l_{\mathcal{M}}}.$$

3.2.3. Results

For the tests, the Cmesh size is fixed to 20 elements. The number of Mmesh elements $n^e_{\mathcal{M}}$ is varied and takes the numbers $98, 198, 298, \ldots, 998, 1998, 2998, \ldots, 9998$. As the meshes have a varying meshsize, there is no fixed MpC that holds for each of the Cmesh elements. The q-convergence curves of the three methods with these parameters are shown in Figure 5.

As it was seen in the previous test case, the curves for different Mmesh sizes cases coincide for small q. Corresponding to the stagnating behaviour on the circle, the curves stagnate one after each other, as the curving order q increases. As expected for this more general geometry, odd and even order Lobatto shape functions contribute to a similar extent.

The convergence rate is about the same with all three methods. The previously seen disadvantage of the reference based method does not arise. For some q and Mmesh combinations it even leads to the best results. A relatively big difference between the reference based modal approach and the other two approaches is for fine Mmeshes and order $q = 6$. It seems that the geometry of the distorted ellipse in the physical space is particularly well approximated by 6th order polynomials. An increase of the order to $q = 7$ does not lead to much improvement.

For the two coarsest tested Mmeshes, the order restriction of the least squares model can be observed. In these cases, the number of Mmesh vertices is not high enough to robustly determine the α_i. Note that for the coarser Mmeshes the least squares seems to obtain less accurate results. No conclusions can be drawn from this, as the GDE

is measured with respect to the Mmesh, whereas the least squares might actually provide a slightly more accurate approximation of the exact geometry.

4. Extension to 3D

In this paper, the algorithms were described and tested only for 2D geometries. The main steps for the extension of the methods to 3D are explained in this section. For a 3D geometry, the boundary forms a surface which we assume to be triangulated. As in the 2D case we consider a coarse linear Cmesh and a fine linear Mmesh as curving target, where the Cmesh vertices form a subset of the Mmesh vertices.

To ensure continuity in between the boundary triangle elements, the edges are curved first. In second curving step, the triangle faces are curved, taking into account the contribution of the curved edges. Here, another advantageous property of the Lobatto shape functions is exploited: On the triangle, the shape functions can be classified in vertex, edge and face shape functions, each only contributing on the geometric entity it corresponds to and higher entities. Edge shape functions are zero on all vertices and on the two edges they do not belong to. All face shape functions contribute only on the face interior and vanish on the element vertices and edges. Thanks to this property, the face curving step can be applied without any effects on the previously curved edges.

In 2D, the Mmesh arc over a single Cmesh element is bounded by the $\mathcal{M}$ vertices coinciding with the C vertices. In 3D, the edge target curves form the boundary of the Mmesh surface corresponding to a single Cmesh element. They are defined as the intersection between the plane in edge normal direction and the Mmesh. The edge curving itself follows the 2D algorithms. In this step, the treatment of feature lines can be introduced. Assuming that the feature line is marked on both, $\mathcal{M}$ and C, the edge target is simply defined by the Mmesh vertices on the feature line. For the face curving, all Mmesh vertices between the edge target curves are considered. Mean value coordinates (see [11]) are used to find a parametrisation of the triangle that corresponds to the idea of length-based parametrisation in 2D.

As it is mentioned in the Introduction, the δ-property does not hold for the standard triangle face shape functions as they are introduced in [6]. This means that no explicit formula can be found for the corresponding face α values in the H^1 optimisation-based algorithms. Instead, a linear system has to be solved for the face curving. Alternatively, orthogonalised shape functions based on [7] could be used.

Currently, the implementation and testing of the direct modal approach in 3D is under way.

5. Summary, conclusions, and future work

Three 2D curving methods are compared: a least squares approach in the physical space, a H^1-seminorm optimisation in the physical space (called direct modal method), and a H^1-seminorm optimisation in a reference space (called reference based method). All three methods are based on the approximation of the curve using hierarchical, modal shape functions. The geometry is described by a fine linear target mesh. Two test cases are considered, a circle test case and a more complicated distorted ellipse.

The circle test case highlights the limitations of the reference based method, which results in a lower convergence order than the other two methods. This effect vanishes for the distorted ellipse, but overall the results are not significantly better than for the direct modal approach. Furthermore, the quadrangles in the physical space that define the mapping from the reference space can not always be constructed in such a way that they fulfill all requirements, see [8].

The main drawback of the direct modal approach relates to the stagnation due to the Mmesh accuracy. A preprocessing step running a mesh reconstruction method for the Mmesh before the curving could improve this, similar to what is proposed in [5].

The advantage of least squares, using only exact points, is clearly observed for the circle test case. It becomes less distinct though on the more complicated geometry. There, the less refined Mmeshes reveal a disadvantage of the least squares. If the Mmesh becomes too coarse, the least squares approach becomes unstable or even uncalculable.

All together, the direct modal method is considered the most promising approach of the three compared methods. It shows the expected stagnation due to the inaccuracy in the target curve, but otherwise converges with the best obtained order and does not show instabilities for coarse Mmeshes.

More test cases in 2D should be assessed to strengthen the conclusions drawn in this paper. Further, we plan to compile guidelines on the choice of the Cmesh size for a desired target accuracy in the geometric approximation and specific simulation results. As mentioned above, our current and future work includes the extension of the direct modal method to 3D.

Acknowledgements

This work was performed as part of the CRANE project (Community and Ramp Aircraft NoisE, www.crane-eid.eu, GA: 606844) funded by the European Union under the Framework Programme 7. The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of this work.

References

[1] S.Dey, M.A. Shephard, J.E. Flaherty, Geometry representation issues associated with p-version finite element computations, Comput. Method. Appl. M. 150(1) (1997) 39–55.

[2] X.J. Luo, M.A. Shephard, J.-F. Remacle, The influence of geometric approximation on the accuracy of high order methods, Rensselaer SCOREC report 1 (2001).

[3] Z. Xie, R. Sevilla, O. Hassan, K. Morgen, The generation of arbitrary order curved meshes for 3D finite element analysis, Comput. Mech. 51(3) (2013) 361–374.

[4] J.-F. Remacle, J. Lambrechts, C. Geuzaine, T. Toulorge, Optimizing the geometrical accuracy of 2D curvilinear meshes, Procedia Engineering 82 (2014) 228–239.

[5] K. Bock, J. Stiller, Generation of high-order polynomial patches from scattered data, in Spectral and High Order Methods for Partial Differential Equations-ICOSAHOM 2012 (2014) 157–167.

[6] P. Šolín, K. Segeth, I. Doležel, Higher order finite element methods, Chapmann & Hall CRC, 2004.

[7] P. Šolín, T. Vejchodský, M. Zítka, F. Ávila, Imposing orthogonality to hierarchic higher-order finite elements, Math. Comput. Simulat. 76(1) (2007) 211–217.

[8] V.S. Schmid, H. Bériot, O. Atak, G. Gabard, High-order curved mesh generation by using a fine linear target mesh, ECCOMAS Congress 2016 Proceedings, Vol. 1 (2016) 493–503.

[9] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, Int. J. Numer. Meth. Eng. 79(11) (2009) 1309–1331.

[10] C. Geuzaine, A. Johnen, J. Lambrechts, J.-F. Remacle, T. Toulorge, The generation of valid curvilinear meshes, IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach (2015) 15–39.

[11] M.S. Floater, Mean value coordinates, Comput. Aided Geom. D., 20(1) (2003) 19–27.