Received: 10 May 2024 | Accepted: 16 October 2024

DOI: 10.1002/pamm.202400046

RESEARCH ARTICLE

Remeshing in the finite cell method for different types of geometry descriptions

Roman Sartorti | **Alexander Düster**

Numerical Structural Analysis with Application in Ship Technology, Institute for Ship Structural Design and Analysis (M-10), Hamburg University of Technology, Hamburg, Germany

Correspondence
Roman Sartorti, Numerical Structural Analysis with Application in Ship Technology, Institute for Ship Structural Design and Analysis (M-10), Hamburg University of Technology, Am Schwarzenberg Campus 4 (C), 21073 Hamburg, Germany.
Email: roman.sartorti@tuhh.de

Funding information
Deutsche Forschungsgemeinschaft, Grant/Award Numbers: DU 405/21-1, 505137962

Abstract

The numerical structural analysis of problems with complex geometries can be challenging, especially if standard finite elements are used. In contrast, immersed methods, such as the finite cell method, relieve the mesh generation such that simply shaped elements/cells can be used. Then, the domain boundary is considered during the numerical integration. In finite strain analysis, the elements/cells face large distortions, especially the cells that are intersected by the boundary. When the solution fails, remeshing can be applied to continue the simulation. This process is currently limited to geometries described by a triangulated surface. Therefore, the present work shows an alternative way of describing the deformed geometry by interpolating the displacement field. In this work, the inverse distance approach, and radial basis functions (RBF) with and without a constant extension are applied. It turns out that RBF with constant extension leads to the most robust results compared to the other methods. Moreover, different geometry description types are tested, and the present approach leads to promising results.

1 | INTRODUCTION

For the numerical solution of partial differential equations, the finite element method (FEM) [1, 2] is in most cases the method of choice. However, FEM-based simulations require a mesh that is conforming to the boundary of the physical domain. In contrast the finite cell method (FCM) [3] and other immersed-boundary methods like CutFEM [4, 5] or CutIGA [6] have the appealing benefit that they can handle complex structures without creating boundary conforming meshes. Thereby, the solution can be obtained using a simple shaped mesh, and the domain boundary is considered during the numerical integration of system matrices and vectors.

In the FCM, the elements are also referred to as cells. In both discretization techniques, the FCM and the FEM, the solution of large deformation problems can be challenging because the heavily distorted cells/elements can lead to failure of the incremental/iterative solution procedure. To resume the simulation, then a remeshing step can be performed such that the process can be continued with cells/elements of better quality. In the FCM context this is shown in ref. [7] for hyperelastic material models.

The immersed-boundary methods require a representation of the physical domain to capture and handle the domain boundary accurately. The physical domain can be described using constructive solid geometries (CSG), triangulated sur-

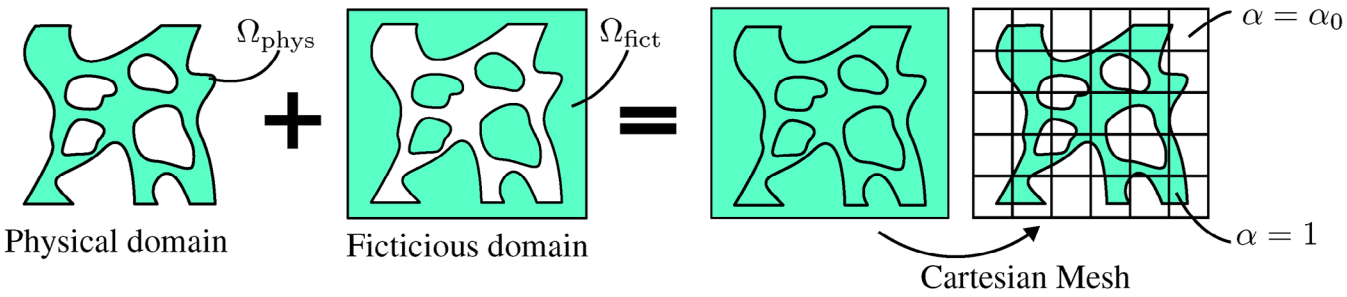


FIGURE 1 Sketch of the idea behind the FCM. The physical domain is embedded into a fictitious domain to such that the meshing is done on a simple shaped domain. The indicator function $\alpha(\boldsymbol{x}_0)$ distinguishes between physical and fictitious domain.

faces, level-set functions, or voxel files, for instance. To this end, remeshing in combination with FCM was only applied to geometries based on triangulated surfaces because then the update of the boundary is simply obtained by updating the triangles describing the boundary. Therefore, the question arises how to handle structures that are not given as a triangulated surface. This will be addressed in the present study.

The rest of the paper is organized as follows: in Section 2 the theoretical background is briefly explained and the alternative domain representation is introduced. Then, in Section 3 different numerical examples of varying complexity are presented. Lastly, in Section 4 the findings of the present study are summarized and a short outlook for future investigations is given.

2 | THEORY

2.1 | Finite Cell Method (FCM)

The FCM is a immersed-boundary method. Thereby, typically hierarchic ansatz functions based on integrated Legendre polynomials are used for the finite cells. The idea behind the FCM and other immersed-boundary methods is shown in Figure 1. The physical domain of interest is embedded into a fictitious domain such that the union of both domains has a simple shape. This simply shaped domain can easily be meshed using Cartesian grids, for example. To distinguish between physical and fictitious domains the indicator function

$$\alpha(\boldsymbol{x}_0) = \begin{cases} 1 & \forall\, \boldsymbol{x}_0 \in \Omega_{\text{phys}} \\ \alpha_0 & \text{else,} \end{cases} \tag{1}$$

is introduced where the stabilization parameter α_0 is a small positive number. This is required to avoid ill-conditioning of the tangent stiffness matrix when a cell is almost empty.

In order to solve non-linear problems, the linearized weak form is discretized and solved with a Newton-Raphson scheme. The procedure is thereby similar to the standard FEM and the derivation is for example given in refs. [7, 8]

Since the domain boundary introduces a discontinuity in the integrand, it is not possible to use standard Gaussian quadrature. Different methods exist to face this challenge. The most straight-forward and robust methods are space-tree methods [3, 9] which come at the cost of a large amount of integration points in the broken cells. Moment-Fitting techniques [10–12] reduce this overhead by creating new quadrature rules for each cell. Due to its robustness, in the present work we apply the space-tree method.

2.2 | Remeshing

In finite strain analysis, the structure and the underlying mesh face large deformations. Especially broken cells tend to distort heavily in the fictitious domain. These distortions can cause failure of the incremental/iterative solver. Therefore, remeshing is applied when the solution does not converge anymore or if a certain remeshing criterion is fulfilled, see for

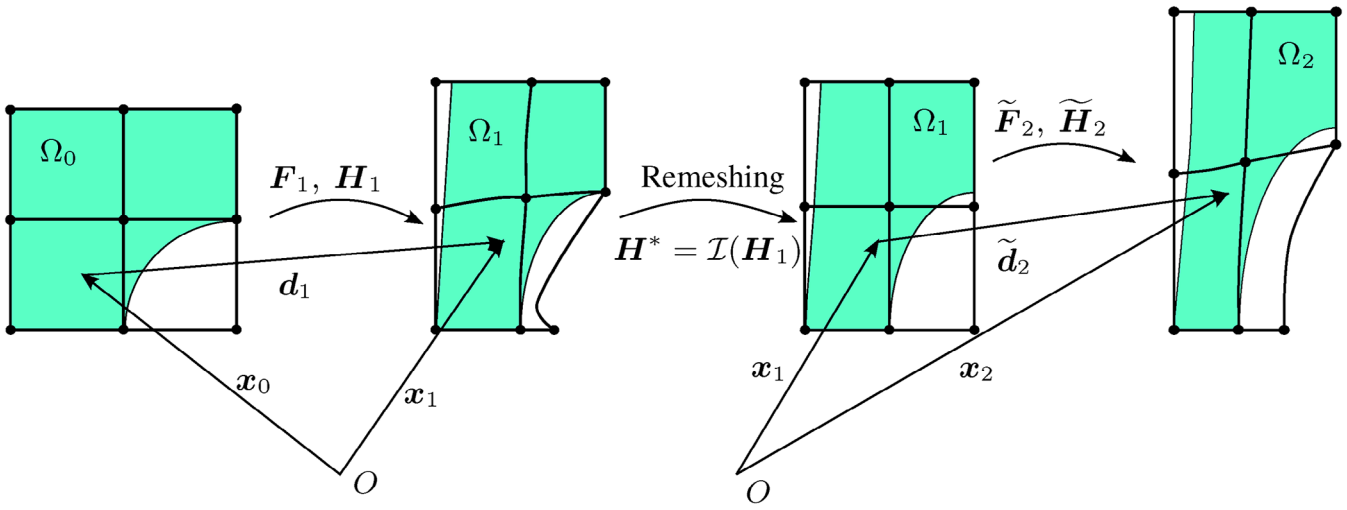


FIGURE 2 Visualization of the remeshing process along with the involved kinematic quantities.

example, [7, 13]. In the present study, remeshing is triggered if either the solver does not converge or if the determinant of the deformation gradient is less than or equal to zero.

The remeshing process itself is depicted in Figure 2. Starting from an initial geometry Ω_0 the boundary conditions are applied such that the deformation state Ω_1 is reached. Then the deformation gradient $\boldsymbol{F}_1$ and the displacement gradient $\boldsymbol{H}_1$ can be computed. The remeshing is initiated, and the displacement gradient is transferred to the new integration points by means of radial basis functions [7]. Then the simulation is continued starting from the new mesh. Note that the tilde indicates that the quantities are defined w.r.t. the intermediate configuration and not w.r.t. the initial configuration. By exploiting the chain rule, the total deformation and displacement gradient, respectively, are computed by

$$\boldsymbol{F}_2 = \frac{\partial \boldsymbol{x}_2}{\partial \boldsymbol{x}_1}\frac{\partial \boldsymbol{x}_1}{\partial \boldsymbol{x}_0} = \widetilde{\boldsymbol{F}}_2\boldsymbol{F}_1 \tag{2}$$

$$\boldsymbol{H}_2 = \boldsymbol{F}_2 - \boldsymbol{I} \tag{3}$$

Note that the $\boldsymbol{F}_1$ is computed for the new mesh using the interpolated displacement gradient, that is,

$$\boldsymbol{F}_1 \approx \boldsymbol{I} + \mathcal{I}(\boldsymbol{H}_1) \tag{4}$$

To this end the displacement gradient is transferred to the new mesh using a radial basis function interpolation, see also Section 2.4.2.

2.3 | Geometry description

Numerous ways exist to describe a geometry that is required for the FCM-based numerical analysis. The important feature of such a geometry is the ability to check if a point in space is inside or outside of the physical domain. In order to apply the remeshing procedure, the inside-outside test has to be performed on the deformed state during the remeshing process. Therefore, the original geometry description has to be updated. To this end, only triangulated surfaces have been considered in this context. The clear benefit of such geometry descriptions is that they can be updated by moving the vertices of all triangles to the deformed configuration. Then, the deformed geometry is given and can be used to proceed with the remeshing.

In the present study, the idea is to get rid of the geometry update itself. Instead, the argument passed into the geometry description to do the inside outside test is modified. This can be accomplished as follows: First, we assume that we have a geometry description $\mathcal{G}_0(\boldsymbol{x}_0)$ in the reference configuration. Once the structure is deformed, the geometry is described

by $\mathcal{G}_1(\boldsymbol{x}_1)$. The key idea is that the following equivalency must hold:

$$\mathcal{G}_1(\boldsymbol{x}_1) = \mathcal{G}_0(\boldsymbol{x}_1 - \boldsymbol{d}_1(\boldsymbol{x}_1)). \tag{5}$$

The only issue at this point is that the displacement field $\boldsymbol{d}_1(\boldsymbol{x}_1)$ is unknown at the deformed state. To tackle this, we use point based interpolation methods. Therefore, we end up with

$$\mathcal{G}_0(\boldsymbol{x}_1 - \boldsymbol{d}_1(\boldsymbol{x}_1)) \approx \mathcal{G}_0(\boldsymbol{x}_1 - \mathcal{I}(\boldsymbol{d}_1)(\boldsymbol{x}_1)) \tag{6}$$

where $\mathcal{I}$ represents an interpolation of the displacement field. The same holds true for multiple remeshing steps and since the geometry description $\mathcal{G}_0$ remains untouched, it is no longer important how the geometry is initially described.

2.4 | Interpolation methods

In the present study, we examine different point based interpolation schemes for the displacement field. In any case, we use the local interpolation approach where only the N^{s} nearest neighbors are considered. This is especially beneficial for the radial basis function based interpolations. Then, the system of equations that has to be solved is way smaller as compared to the global interpolation approach.

2.4.1 | Inverse distance weighting (IDW)

One simple approach to interpolate data based on a point cloud is the IDW method [14]. Thereby, the target point value v^{t} at target point location $\boldsymbol{x}^{\mathrm{t}}$ is computed by

$$v^{\mathrm{t}} = \frac{\sum_{i=1}^{N^{\mathrm{s}}} w_i v^{\mathrm{s}}}{\sum_{j=1}^{N^{\mathrm{s}}} w_j}, \quad \text{with} \quad w_i = \frac{1}{\|\boldsymbol{x}_i^{\mathrm{s}} - \boldsymbol{x}^{\mathrm{t}}\|^2}, \tag{7}$$

where the weights w_i are the inverse of the squared radial distances to the target point. Please note that this method is not able to extrapolate values beyond the absolute maximum value of the source points.

2.4.2 | Radial basis functions (RBF)

An alternative approach is the interpolation based on RBF $\phi(r)$. Then, the target point value can be expressed by

$$v^{\mathrm{t}}(\boldsymbol{x}^{\mathrm{t}}) = \sum_{i=1}^{N^{\mathrm{s}}} w_i \phi(\|\boldsymbol{x}^{\mathrm{t}} - \boldsymbol{x}_i^{\mathrm{s}}\|). \tag{8}$$

Thereby, the weights $\boldsymbol{w}$ are computed by solving

$$\boldsymbol{A}\boldsymbol{w} = \boldsymbol{b}, \tag{9}$$

where $A_{ij} = \phi(\|\boldsymbol{x}_i^{\mathrm{s}} - \boldsymbol{x}_j^{\mathrm{s}}\|)$, and $b_i = v_i^{\mathrm{s}}$. To this end we use the inverse multiquadric RBF, defined by

$$\phi(r) = \frac{1}{\sqrt{1 + r^2}} \tag{10}$$

In contrast to the previous method, the target point values can even exceed the minimum or maximum value of the source points. Note that the radius is normalized w.r.t. to the average distance to the three nearest neighbours. The quality of the interpolation heavily depends on the choice of the RBF. The inverse multiquadric fades out values with increasing distance to the source points.

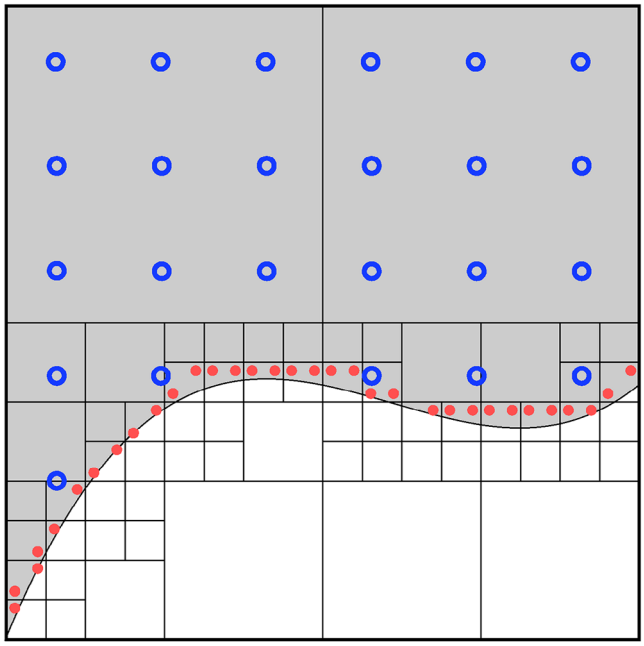


FIGURE 3 Distribution of the source points for the displacement field interpolation. To be accurate at the boundary the integration points of the initial mesh next to the boundary are chosen (red). Additionally, in the domain equally spaced points are distributed (blue).

2.4.3 | RBF with polynomial extension

To improve accuracy of the RBF for data that follows a polynomial distribution it is common practice to add polynomials to the interpolation equation. Then the target point value is computed by

$$\tilde{v}^{\mathrm{t}}(\boldsymbol{x}^{\mathrm{t}}) = v^{\mathrm{t}}(\boldsymbol{x}^{\mathrm{t}}) + \sum_{k=1}^{N^{\mathrm{p}}} \ell_k P_k(\boldsymbol{x}^{\mathrm{t}}) \tag{11}$$

This can be solved similarly as above but with additional constraint $\sum_{i=1}^{N^{\mathrm{s}}} \ell_k P_k(\boldsymbol{x}_i^{\mathrm{s}}) = 0$ for all N^{p} polynomial extensions. In the following we stick to a constant extension and neglect any higher order polynomials:

$$\tilde{v}^{\mathrm{t}}(\boldsymbol{x}^{\mathrm{t}}) = v^{\mathrm{t}}(\boldsymbol{x}^{\mathrm{t}}) + \ell_1 \tag{12}$$

Then the system to be solved is defined by

$$\begin{bmatrix} \boldsymbol{A} & \mathbf{1} \\ \mathbf{1}^{\top} & 0 \end{bmatrix} \tilde{\boldsymbol{w}} = \begin{bmatrix} \boldsymbol{b} \\ 0 \end{bmatrix} \tag{13}$$

where $\mathbf{1}$ is a column vector of ones.

2.5 | Source point distribution

One important aspect of point based interpolation is the positioning of source points. In the present study, we need to have accurate interpolations close to the surface to recover the boundaries properly. In contrast, inside the structure, small errors are not a problem. Therefore, in the present study first, the integration points of the initial mesh in the vicinity of the boundary are chosen as source points. Additionally, in each cell N^{cp} equidistant points per dimension are placed in each cell, inside the domain, see also Figure 3. The source points are generated once and updated in each remeshing step according to the current displacement field.

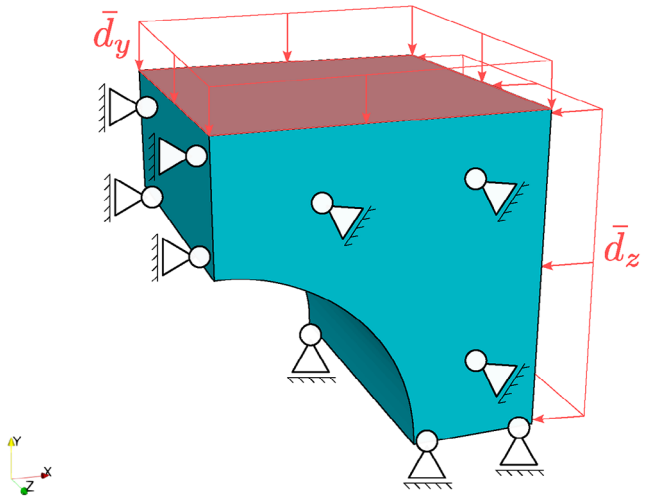


FIGURE 4 Cube with cylindrical hole along with the boundary conditions. The faces have prescribed displacements.

3 | NUMERICAL EXAMPLES

In the following, different numerical examples are presented with varying complexity. In any case a hyperelastic material of Neo-Hookean type is used with the first Lamé parameter $\lambda = 28.846\ N/mm^2$ and shear modulus $\mu = 19.231\ N/mm^2$, see also [13]. For stabilization purposes $\alpha_0 = 10^{-5}$ and a basis function removal [6, 15] with a removal factor of 0.1 is applied to increase the stability of the solution procedure.

3.1 | Cube with hole

This example is constructed by constructive solid geometry operations. The geometry as well as the boundary conditions are shown in Figure 4. Thereby, a cylinder of 0.6 mm is subtracted from a unit cube. The number of source points is varied between $N^{\text{s}} = 5, 10, \dots, 50$ as well as the number of additional points per cell $N^{\text{cp}} = 1, 2, \dots, 10$. Remeshing is initiated after $\bar{d}_z = \bar{d}_z = 0.1\ mm$.

The results of this study are shown in Table 1. For all interpolation methods, the average error μ_ε is below 1% so that the volume is almost preserved no matter which interpolation method was applied. However, the robustness clearly varies between the different methods. The RBF interpolation shows high variance in the error depending on the choice of parameters. In contrast, the RBF+const is insensitive to the parameter choice indicated by the small standard deviation σ_ε and shows the lowest average error μ_ε. The IDW shows varying error depending on the parameter choice especially for the coarse mesh, while it works robust for the fine mesh. In general the IDW tends to work more accurate when only a few source points are considered.

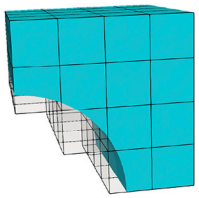
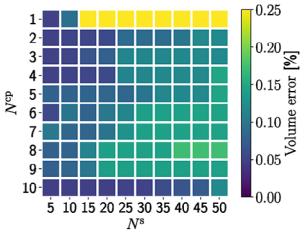
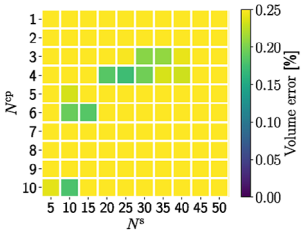
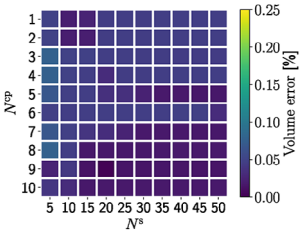
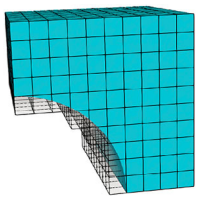
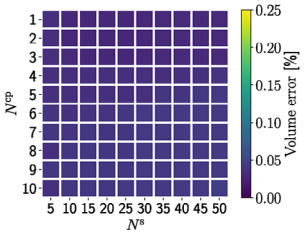
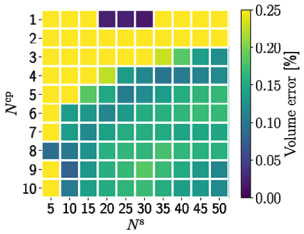
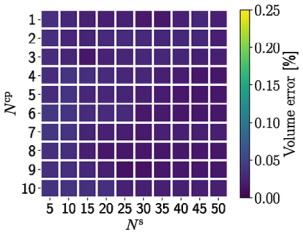
3.2 | Implicit geometry description of a pore of a foam

In the subsequent section, we apply the presented method to a geometry described by a level-set function in combination with two plates at the top and at the bottom. The level-set function without the plates is thereby defined by

$$\begin{aligned}\phi(x,y,z) = (2.5(x-1)x^2(x+1)+2y^2)^2 + (z^2-0.85)^2 \cdot (5(y-1)y^2(y+1)+2z^2)^2 \\ + (x^2-0.85)^2 \cdot (2.5(z-1)z^2(z+1)+2x^2)^2 + (y^2-0.85)^2 \cdot (-0.1) \leq 0.0.\end{aligned} \tag{14}$$

The structure and the mesh for the numerical analysis are shown in Figure 5. The displacement is incrementally applied and remeshing is performed after $\bar{d}_x = 0.5\ mm$. The shape function ansatz order is set to $p = 2$. As shown in the previous example, IDW works best if only few source points are considered. Therefore, we use $N^{\text{s}}_{\text{IDW}} = 5$ source point when IDW is

TABLE 1 Volume error induced during remeshing for different interpolation methods.

Cell size	Model	IDW	RBF	RBF+const
0.25		 $\mu_\epsilon = 0.52\ \sigma_\epsilon = 3.7$ $\mu_\epsilon = 0.52\ \ \sigma_\epsilon = 3.7$	 $\mu_\epsilon = 0.48\ \sigma_\epsilon = 0.70$ $\mu_\epsilon = 0.48\ \ \sigma_\epsilon = 0.70$	 $\mu_\epsilon = 0.035\ \sigma_\epsilon = 0.017$ $\mu_\epsilon = 0.035\ \ \sigma_\epsilon = 0.017$
0.1		 $\mu_\epsilon = 0.036\ \sigma_\epsilon = 0.0050$ $\mu_\epsilon = 0.036\ \ \sigma_\epsilon = 0.0050$	 $\mu_\epsilon = 0.64\ \sigma_\epsilon = 1.1$ $\mu_\epsilon = 0.64\ \ \sigma_\epsilon = 1.1$	 $\mu_\epsilon = 0.025\ \sigma_\epsilon = 0.0072$ $\mu_\epsilon = 0.025\ \ \sigma_\epsilon = 0.0072$

Abbreviations: IDW, inverse distance weighting; RBF, radial basis function.

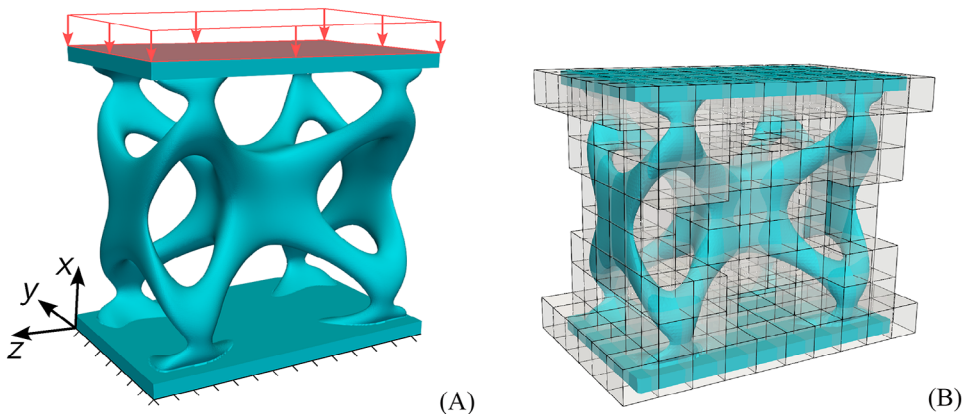


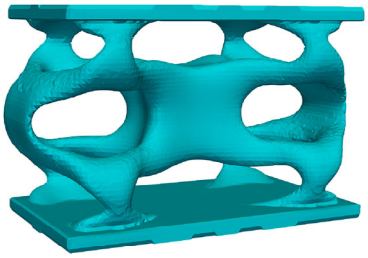
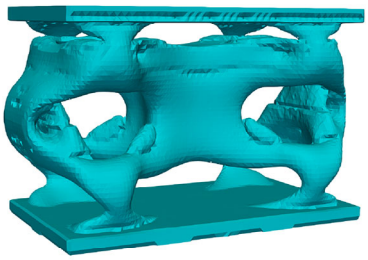
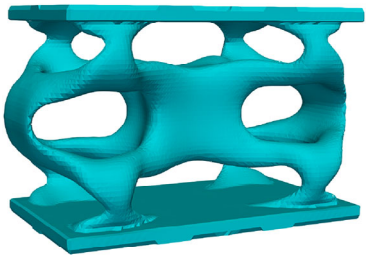
FIGURE 5 In (A) the model along with the boundary conditions are shown and (B) shows the FCM mesh that is used for the analysis.

applied for the interpolation. The RBF based interpolations tend to work better with more considered source points. Thus, we apply $N^{\mathrm{s}}_{\mathrm{RBF}} = 40$ and $N^{\mathrm{s}}_{\mathrm{RBF+const}} = 40$. After the remeshing, the deformed geometry is triangulated using a marching cube algorithm [16].

The triangulated surfaces after the remeshing are shown in Table 2. Thereby, the IDW and the RBF+const displacement interpolation lead to satisfactory results, while the pure RBF approach introduces artificial material during the remeshing. This indicates that the displacement interpolation does not work properly, leading to misinterpretation of the inside-outside test. This may be caused by the fact that the chosen RBF fades out values if the target point is an extrapolation of the source points. This, however, can lead to small displacements, and, hence, the displacement field is wrongly assumed to be zero in the void domain.

One minor difference is visible in the surface smoothness between IDW and RBF+const. The latter provides a rather smooth surface, while the IDW tends to roughen the surface. The error in the volume is in both cases small indicating that the volume is almost preserved. All methods smooth out the sharp edges of the end plates. This happens when the nearest neighbours stem from different faces of the plate, for example, when the interpolation is evaluated close to the corner.

TABLE 2 Triangulated representation of the domain boundaries and the relative volume error ϵ_V for the different displacement interpolation methods.

	IDW	RBF	RBF+const
Surface			
	0.125%	15.3%	0.17%
ϵ_V	0.125%	15.3%	0.17%

Abbreviations: IDW, inverse distance weighting; RBF, radial basis function.

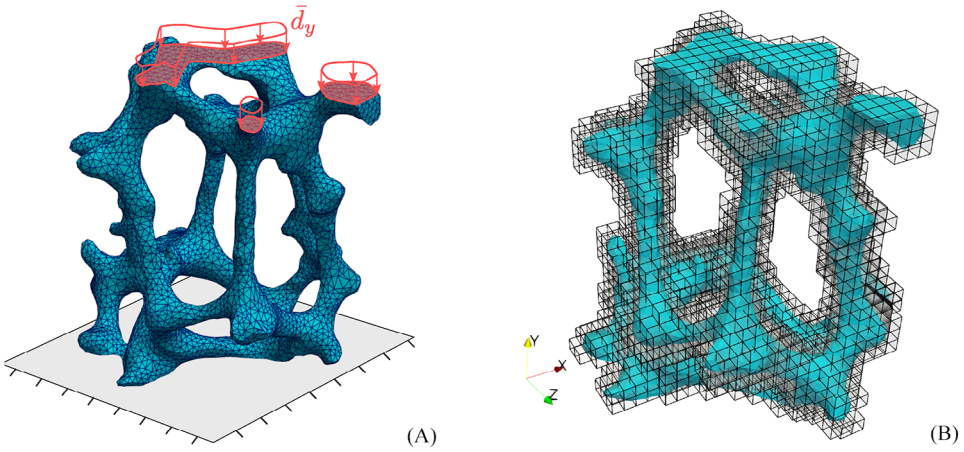


FIGURE 6 The triangulated surface of the single foam pore along with the boundary conditions are shown in A. The mesh that is used for the simulation is shown in B.

3.3 | Single pore of a foam

Lastly, the presented approach is compared to the direct update of the triangulate surface. Thereby, the model of a foam pore is given and shown along with the boundary conditions and the mesh in Figure 6. The displacement is applied incrementally at the highlighted surface of the foam and the base of the foam is clamped. To interpolate the displacement field, the RBF+const is used with $N^{\text{cp}} = 5$ and $N^{\text{s}} = 40$.

To begin with, remeshing is only applied once, and the surface is again visualized by triangulation. The results of the first simulation are shown in Figure 7. The color indicates the difference between the surfaces obtained by the different domain update methods after the remeshing. The contour plot shows only small differences in some parts of the structure. The average distance of the vertices of the triangulated surfaces is 2×10^{-6} mm and the maximum distance is 0.05 mm. Thereby, we conclude that the proposed method works similarly well as compared to the direct domain update approach when moderately large deformations occur.

Next, we further increase the displacement to investigate if the method works also accurately if the cell is collapsing. The triangulated surface, after the second remeshing step is shown in Figure 8. The artificial material that is introduced by displacement interpolation issues is highlighted in red. These tend to appear between two struts, caused by the displacement fields that are acting against each other. If a target point is evaluated, between the struts, then the interpolated displacement is of course interpolated between the values of the concurring struts. Therefore, it is not guaranteed that this correctly maps in to the void domain and, hence, fails to correctly detect the void. This may indicate that it can be beneficial to have also source points in the fictitious domain to be able to better interpolate the displacement in the void domain.

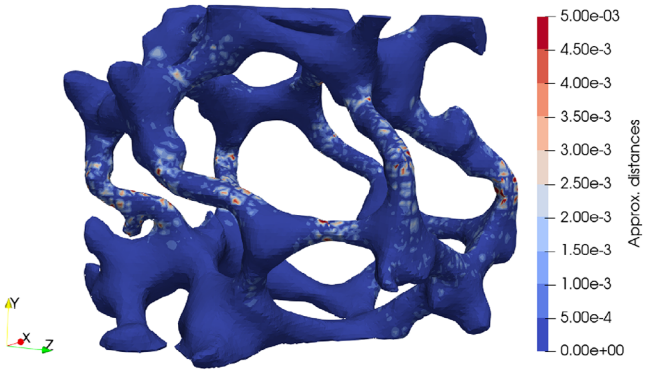


FIGURE 7 Error between the triangulated surfaces based on the present approach and the direct surface update approach. In large areas the error is close to zero.

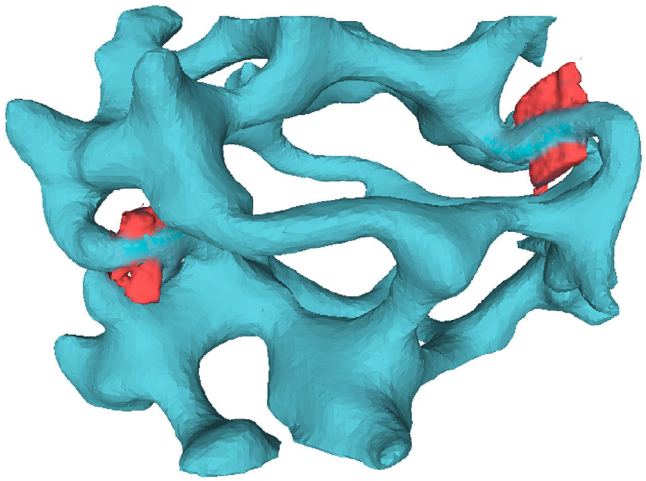


FIGURE 8 The triangulated surface after the second remeshing step. The artificial domain caused by the interpolation error is highlighted in red.

4 | CONCLUSION

In the present study, we have introduced an alternative method to combine the FCM with remeshing. Previously, remeshing was only possible when a triangulated surface was given. The present approach works as well for other types of geometry descriptions. Thereby, the displacement field is interpolated using different point based interpolation techniques. By doing so, we reformulated the inside-outside test such that the geometry does not have to be updated. It turned out that the RBF with constant extension provides the most robust results and works similarly well as the direct domain update used in previous publications [7, 13], at least for moderately large deformations. However, in the current implementation, the method does not work well for extremely large deformations. As shown in the last example, artificial material is added during remeshing in the void domain caused by poor interpolation in the regimes where the structure collapses and gets close to self-contact.

For future work, it is of interest if the interpolation can be improved in the fictitious domain to avoid the artificial material from growing in the fictitious domain. Then, further investigations could go in the direction of employing self-contact into the simulation procedure.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support provided by the Deutsche Forschungsgemeinschaft (DFG) under the grant number DU 405/21-1 and the project number 505137962.

Open access funding enabled and organized by Projekt DEAL.

ORCID

Roman Sartorti https://orcid.org/0000-0002-3410-8584

REFERENCES

1. Szabó, B., & Babuška, I. (1991). *Finite element analysis*. John Wiley & Sons.
2. Zienkiewicz, O., & Taylor, R. (2000). *The finite element method – solid mechanics*. Butterworth-Heinemann.
3. Parvizian, J., Düster, A., & Rank, E. (2007). Finite cell method – h- and p-extension for embedded domain problems in solid mechanics. *Computational Mechanics*, *41*, 121–133.
4. Burman, E., & Hansbo, P. (2010). Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method. *Computer Methods in Applied Mechanics and Engineering*, *199*(41-44), 2680–2686.
5. Burman, E., Claus, S., Hansbo, P., Larson, M., & Massing, A. (2015). CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, *104*, 472–501.
6. Elfverson, D., Larson, M. G., & Larsson, K. (2018). CutIGA with basis function removal. *Advanced Modeling and Simulation in Engineering Sciences*, *5*(1), 6.
7. Garhuom, W., Hubrich, S., Radtke, L., & Düster, A. (2020). A remeshing strategy for large deformations in the finite cell method. *Computers & Mathematics with Applications*, *80*, 2379–2398.
8. Wriggers, P. (2009). *Nonlinear finite-element-methods*. Springer-Verlag.
9. Petö, M., Garhuom, W., Duvigneau, F., Eisenträger, S., Düster, A., & Juhre, D. (2022). Octree-based integration scheme with merged sub-cells for the finite cell method: Application to non-linear problems in 3D. *Computer Methods in Applied Mechanics and Engineering*, *401*, 115565.
10. Joulaian, M., Hubrich, S., & Düster, A. (2016). Numerical integration of discontinuities on arbitrary domains based on moment fitting. *Computational Mechanics*, *57*, 979–999.
11. Legrain, G. (2021). Non-negative moment fitting quadrature rules for fictitious domain methods. *Computers & Mathematics with Applications*, *99*, 270–291.
12. Garhuom, W., & Düster, A. (2022). Non-negative moment fitting quadrature for cut finite elements and cells undergoing large deformations. *Computational Mechanics*, *70*, 1059–1081.
13. Sartorti, R., & Düster, A. (2023). Remeshing and data transfer in the finite cell method for problems with large deformations. *Proceedings in Applied Mathematics and Mechanics*, *23*(3), e202300088.
14. Shephard, M., Dey, S., & Flaherty, J. (1997). A straightforward structure to construct shape functions for variable p-order meshes. *Computer Methods in Applied Mechanics and Engineering*, *147*, 209–233.
15. Garhuom, W., Hubrich, S., Radtke, L., & Düster, A. (2022). Adaptive quadrature and remeshing strategies for the finite cell method at large deformations. In J. Schröder, & P. Wriggers (Eds.). *Non-standard discretisation methods in solid mechanics*, Lecture notes in applied and computational mechanics (pp. 327–353). Springer.
16. Lorensen, W. E., & Cline, H. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, *21*(4), 163–169.

How to cite this article: Sartorti, R., & Düster, A. (2024). Remeshing in the finite cell method for different types of geometry descriptions. *Proceedings in Applied Mathematics and Mechanics*, *24*, e202400046. https://doi.org/10.1002/pamm.202400046