



# An eigenvalue stabilization technique to increase the robustness of the finite cell method for finite strain problems

Wadhah Garhuom<sup>1</sup> · Khuldoon Usman<sup>1</sup> · Alexander Düster<sup>1</sup>

Received: 16 November 2021 / Accepted: 5 January 2022 / Published online: 30 January 2022  
© The Author(s) 2022

## Abstract

Broken cells in the finite cell method—especially those with a small volume fraction—lead to a high condition number of the global system of equations. To overcome this problem, in this paper, we apply and adapt an eigenvalue stabilization technique to improve the ill-conditioned matrices of the finite cells and to enhance the robustness for large deformation analysis. In this approach, the modes causing high condition numbers are identified for each cell, based on the eigenvalues of the cell stiffness matrix. Then, those modes are supported directly by adding extra stiffness to the cell stiffness matrix in order to improve the condition number. Furthermore, the same extra stiffness is considered on the right-hand side of the system—which leads to a stabilization scheme that does not modify the solution. The performance of the eigenvalue stabilization technique is demonstrated using different numerical examples.

**Keywords** Finite cell method · Condition number · Eigenvalues · Stabilization · Large deformations

## 1 Introduction

In fictitious domain or immersed boundary methods, such as the finite cell method (FCM) [13,32], CutFEM [6–8], or CutIGA [15], the cells/elements generally do not conform to the boundary of the geometry of interest. As a result, the geometry can be easily discretized using, for instance, Cartesian grids. Nevertheless, this simplification in the mesh generation leads to broken cells/elements that are cut by the boundary of the domain. This can result in an ill-conditioned global system matrix, especially if badly broken cells/elements are filled only with a small fraction of material. In the extended finite element method (XFEM), conditioning problems can also arise. An overview of different approaches to overcome the conditioning problem is given in [29]. In nonlinear finite strain analysis, badly broken

cells/elements tend to deform significantly compared to non-broken cells which leads to convergence problems during the iterative/incremental procedure.

To this end, different stabilization techniques have been developed for fictitious domain methods to overcome the problem of ill-conditioning. In the context of the CutFEM, Burman et al. [6–8] proposed a stabilization approach called the *ghost penalty* method, to obtain a better condition number for elements that are cut by the boundary. To this end, an extra term is added to the weak form so as to penalize jumps of the normal derivatives of the shape functions of the neighboring cut and non-cut elements. Consequently, the cut elements are supported by the interior non-cut elements. Here, the penalty term is referred to as *ghost* since it is related to the fictitious domain.

In the context of the FCM, the fictitious material or  $\alpha$ -method is suggested in [13,32], where a very soft material is introduced in the fictitious domain of every broken cell. This is done by scaling the material parameters with the indicator function  $\alpha$ , which has a value of one for points inside the physical domain and a value of  $\alpha = 10^{-q}$  for points outside the physical domain. For practical problems, we set  $q = 5, \dots, 9$  to achieve reasonable results. To improve the condition number without changing the problem under consideration substantially, it is important not to add too much stiffness in the fictitious domain [10]. This approach also

✉ Wadhah Garhuom  
wadhah.garhuom@tuhh.de  
Khuldoon Usman  
khuldoon.usman@tuhh.de  
Alexander Düster  
alexander.duester@tuhh.de

<sup>1</sup> Numerical Structural Analysis with Application in Ship Technology (M-10), Hamburg University of Technology, Am Schwarzenberg-Campus 4 (C), Hamburg 21073, Germany

helps to control the solution within the fictitious domain while at the same time improving the condition number. This is especially necessary when simulating nonlinear finite strain problems to prevent convergence issues. Generally, the approach based on the fictitious material works well for linear and nonlinear problems under small strains [10,13,22,40]. One drawback of this approach is that the same artificial stiffness is added to all points in the fictitious domain, which can result in a significant modification of the solution.

Additionally, for the analysis using high-order shape functions, degrees of freedom that have a small support in the physical domain tend to increase the condition number and decrease the robustness of the FCM. To this end, [15,17,42] proposed a *basis function removal* (BFR) strategy to improve the condition number by removing shape functions that do not contribute much to the global stiffness matrix. Thereby, a global criterion was developed based on the energy contribution of each degree of freedom – allowing to determine which shape function needs to be removed. In doing so, the high-order modes of the broken cells that cause the problems are removed and consequently the condition number is improved significantly.

Moreover, the severe distortion of the mesh in nonlinear finite strain problems can cause the analysis to terminate before reaching the desired deformation state. To this end, a *remeshing strategy* proposed in [17,18] can improve the robustness of the FCM by creating a new mesh whenever the old mesh cannot take any more deformations. Starting with an initial mesh, the structure is deformed until the remeshing criteria indicate that a remeshing step is required because the mesh is strongly distorted. Then, a new mesh is created that covers the deformed geometry. Thanks to the fictitious domain approach, creating the mesh is straightforward. Subsequently, important quantities between the old and new mesh are interpolated. Finally, the analysis is continued until the desired load step is reached. It was shown in [17] that a combination of the remeshing strategy, the basis function removal, and the  $\alpha$ -method can improve the robustness even for very large deformations.

In [34], a *preconditioning technique* was developed to prevent poor conditioning of the FCM caused by broken cells. This approach is referred to as the *Symmetric Incomplete Permuted Inverse Cholesky* preconditioner. To this end, the preconditioning involves a diagonal scaling of the shape functions as well as an orthonormalization process utilizing the Gram–Schmidt procedure.

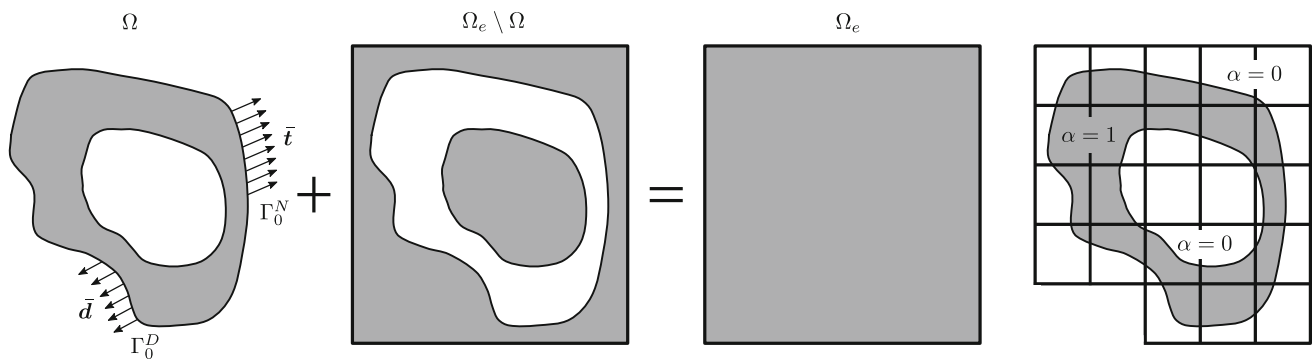
In this paper, we present an *eigenvalue stabilization* technique, based on the approach of Loehnert [29], to improve the robustness of the FCM for nonlinear problems by reducing the condition number of broken cells without affecting the solution significantly. To this end, an eigenvalue decomposition of the cell stiffness matrix is computed for every broken cell. Afterwards, the modes that have small eigenvalues or

even zero eigenvalues can be grouped together. Those modes cause the FCM to be less robust in nonlinear computations. They appear especially when badly broken cells are present together with high-order shape functions. The modes with zero eigenvalues that belong to rigid body motions are not considered for stabilization—in order to avoid non-physical behavior. Finally, the bad modes are supported by adding extra stiffness to the cell stiffness matrix of the corresponding mode and to the load vector, to ensure that the solution is not modified significantly. Furthermore, we introduce a criterion that allows to adaptively add stiffness to the modes depending on their corresponding eigenvalue.

This paper is organized as follows: Sect. 2 gives a brief summary of the basic formulation of the FCM. In Sect. 3, the stabilization of the FCM based on the  $\alpha$ -method is explained. In Sect. 4, an eigenvalue stabilization technique for the FCM is discussed in detail. In Sect. 5, we study the accuracy of the proposed method on a linear benchmark example and on more complex nonlinear finite strain problems. Finally, the paper is summarized in Sect. 6.

## 2 The finite cell method

This section offers a brief overview of the basics of the finite cell method (FCM)—for more details see [13,14,32,36]. The FCM is a combination of high-order finite elements with a fictitious domain approach. To this end, hierarchic shape functions based on integrated Legendre polynomials are utilized [38,39]. The FCM was successfully applied to a variety of problems in solid mechanics such as thermoelasticity [44], geometrical nonlinearities [18,37], explicit and implicit elastodynamics [16,24], biomechanics [35,42], elastoplasticity [3,26,40], and microstructured materials [20,21,27]. Figure 1 illustrates the basic concept of the method using a two-dimensional geometry of solid mechanics. Here, the physical domain  $\Omega$  is embedded into a fictitious domain  $\Omega_e \setminus \Omega$  which results in an extended domain  $\Omega_e$  of a simple shape. The extended domain  $\Omega_e$  can then be easily discretized using, for example, Cartesian grids. Dirichlet boundary conditions  $\vec{d}$  as well as Neumann boundary conditions  $\vec{t}$  are applied on the boundary of the physical domain  $\Omega$ . For the sake of simplicity, we neglect body forces in this context. As a result of the fictitious domain approach, the approximation of the displacements is decoupled from the approximation of the geometry—in contrast to the standard finite element method (FEM). Therefore, the term *cells* is used to differentiate them from the geometry conforming finite elements. Next, to account for the boundary of the geometry, the indi-



**Fig. 1** Basic concept of the FCM

cator function

$$\alpha = \begin{cases} 1, & \text{for } x \text{ in } \Omega \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

is introduced, which has a value of one inside the physical domain and a value of zero elsewhere. The nonlinear weak form  $G_e^\alpha$  reads [43]

$$G_e^\alpha = \int_{\Omega_e} \alpha \mathbf{P} \cdot \text{Grad } \boldsymbol{\eta} \, dV - \int_{\Gamma_0^N} \bar{\mathbf{t}} \cdot \boldsymbol{\eta} \, dA = 0 \quad (2)$$

where  $\mathbf{P}$  denotes the first Piola–Kirchhoff stress tensor and  $\boldsymbol{\eta}$  corresponds to the test function. Since our focus later on will be on large deformation analysis, Eq. (2) is highly nonlinear. Therefore, we utilize the Newton–Raphson method, which is based on the linearization of the weak form. Assuming that the external loads are independent of the deformation, the directional derivative of  $G_e^\alpha$  reads

$$DG_e^\alpha \cdot \Delta \mathbf{d} = \int_{\Omega_e} \alpha A \text{Grad } \Delta \mathbf{d} \cdot \text{Grad } \boldsymbol{\eta} \, dV, \quad (3)$$

where

$$A = \frac{\partial \mathbf{P}}{\partial \mathbf{F}} \quad (4)$$

and  $\Delta \mathbf{d}$  represents the displacement increment. Accordingly, the linearized weak form results in

$$\int_{\Omega_e} \alpha A \text{Grad } \Delta \mathbf{d} \cdot \text{Grad } \boldsymbol{\eta} \, dV = \int_{\Gamma_0^N} \bar{\mathbf{t}} \cdot \boldsymbol{\eta} \, dA - \int_{\Omega_e} \alpha \mathbf{P} \cdot \text{Grad } \boldsymbol{\eta} \, dV. \quad (5)$$

Next, the extended domain is discretized using finite cells based on hierarchic shape functions in order to compute an

approximate solution of the problem at hand, as illustrated in Fig. 1. Inserting the discretization of the displacement and the test function into the weak form results in the following linear equation system

$$\mathbf{K}_T^i(\mathbf{d}^i) \Delta \mathbf{d}^{i+1} = -\mathbf{G}^i(\mathbf{d}^i) \quad (6)$$

which needs to be solved within each Newton–Raphson iteration for the unknown displacement increment. Here,  $\mathbf{K}_T^i(\mathbf{d}^i)$  refers to the global tangent stiffness matrix. Furthermore,  $\mathbf{G}^i(\mathbf{d}^i)$  defines the global out of balance vector. During the assembly process of the local stiffness matrices and vectors of each finite cell, the global quantities are acquired as follows

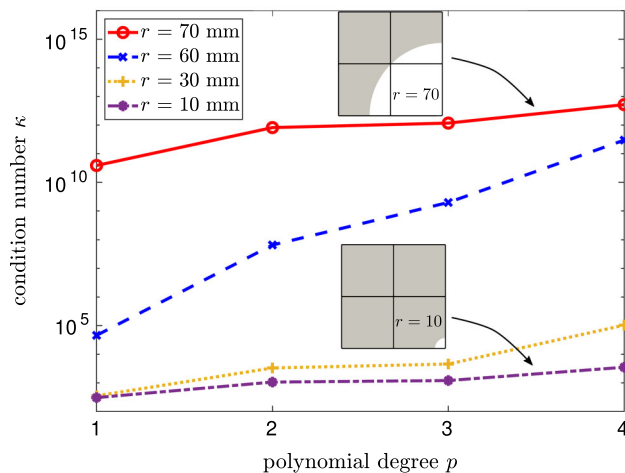
$$\mathbf{K}_T^i = \bigwedge_{c=1}^{n_c} \mathbf{k}^{c,i}, \quad \mathbf{G}^i = \bigwedge_{c=1}^{n_c} \mathbf{g}^{c,i}. \quad (7)$$

The simplification in the mesh generation using the FCM leads to broken cells which are intersected by the boundary of the domain. This results in discontinuous integrands, see Eq. (5). Consequently, standard Gauss quadratures do not perform well for those cells. Therefore, special integration schemes need to be used—such as, for instance, the adaptive integration scheme based on spacetree decomposition [2,13,28,33], the moment fitting quadrature [11,12,19,25], or the approach based on equivalent polynomials [1,41].

Furthermore, fictitious domain methods such as the FCM often suffer from ill-conditioning of the resulting system matrices caused by the broken cells. Figure 2 shows the condition number

$$\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}, \quad \text{with } \lambda_{\min} \neq 0 \quad (8)$$

for a plate with a hole, discretized using four finite cells applying different values of the radius  $r$  for the circle. The condition number is plotted against the polynomial degree  $p$  of the hierarchic shape functions applied in the finite cells. It can be seen that the condition number increases drastically



**Fig. 2** Condition number of the plate with a hole, discretized with four cells and using different radii  $r$  for the geometry

as the radius increases since the volume fraction of the cell at the bottom right decreases gradually. This causes difficulties when solving the linear system of equations.

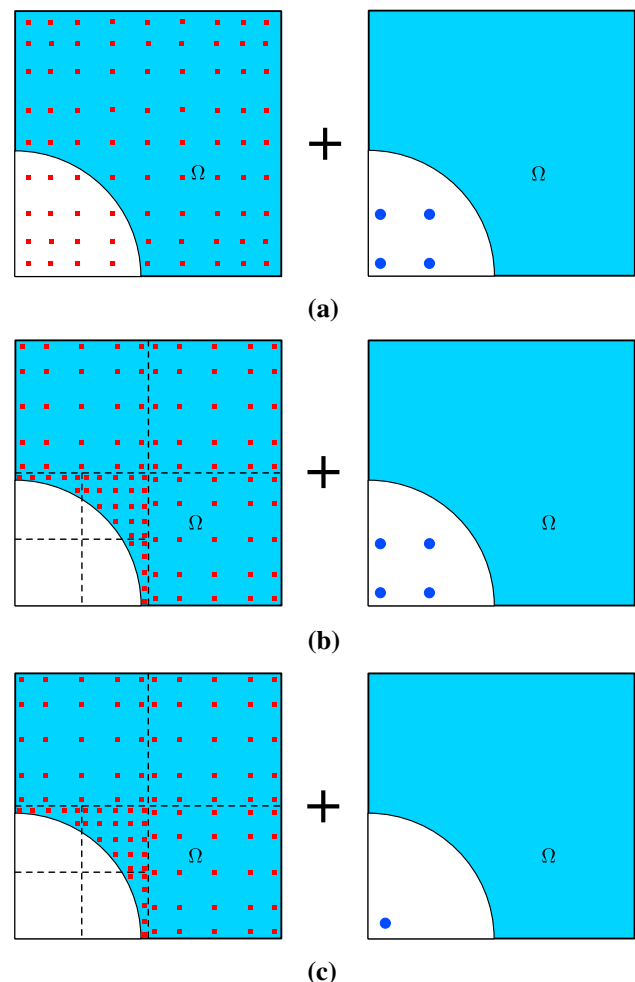
### 3 Material stabilization ( $\alpha$ -method)

The commonly used method to improve the condition number of the FCM is the so-called *material stabilization technique* or the  $\alpha$ -method [10,13]. The main idea is to introduce a soft material into the fictitious domain. To this end, the indicator function  $\alpha$  is set to a small positive value for integration points that are located in the fictitious domain

$$\alpha = \begin{cases} 1, & \text{for } x \text{ in } \Omega \\ 10^{-q}, & \text{otherwise} \end{cases} \quad (9)$$

In doing so, the condition number can be improved by introducing an artificial stiffness, as will be shown in Sect. 5. By setting the indicator function  $\alpha$  to zero in the fictitious domain, the geometry is exactly preserved. To achieve reasonable results without losing too much accuracy, the  $q$  values should be chosen in a range of  $q = 5, \dots, 9$ . One advantage of this approach is the simple implementation, since a new set of Gaussian points can be introduced in the fictitious domain to reduce the ill-conditioning of the system.

Figure 3 shows the number of integration points needed utilizing a shape function order of  $p = 4$ . Here, the red dots refer to the points required to integrate the physical domain and the blue dots represent the stabilization points. In Fig. 3a, the moment fitting scheme is applied with  $(2p+1)$  points in each direction to integrate the physical domain. Additionally, a set of  $(p+1)$  points in each direction is added to stabilize the fictitious domain, excluding the points that are located in



- Points for integrating the physical domain
- Points for stabilizing the fictitious domain

**Fig. 3** The integration points used for stabilization (blue dots) and the points used for integrating the physical domain (red dots) applied on one broken cell with an ansatz order of  $p = 4$ . **a** Moment fitting with  $(2p+1)^2$  physical points plus  $(p+1)^2$  stabilization points. **b** Quadtree with tree depth  $k = 2$  and  $(p+1)^2$  physical points in each sub-cell plus  $(p+1)^2$  stabilization points. **c** Quadtree with tree depth  $k = 2$  and  $(p+1)^2$  physical points in each sub-cell plus  $(p-1)^2$  stabilization points. (Color figure online)

the physical domain. This scheme will be used in Sect. 5.1 for the linear computations. In the nonlinear analysis (Sect. 5.2 and 5.3), the adaptive octree is applied to integrate the physical domain with  $(p+1)$  points in each direction of every sub-cell, excluding the points located in the fictitious domain. Additionally, a number of  $(p+1)$  points in each direction are applied to stabilize the fictitious domain, excluding the points located in the physical domain, as can be seen in Fig. 3b for a tree depth of  $k = 2$ .

One drawback of the material stabilization approach is the loss of accuracy especially, if large  $\alpha$  values are to be used. In the nonlinear analysis, the  $\alpha$ -method will be combined with an eigenvalue stabilization technique (see Sect. 4) to

increase the robustness of the FCM when undergoing large deformations. To this end, a set of  $(p - 1)$  points will be added in each direction of the fictitious domain, excluding the points located in the physical domain and setting a small value of  $\alpha = 1e-7$  for those points, as can be seen in Fig. 3c.

#### 4 Eigenvalue stabilization technique

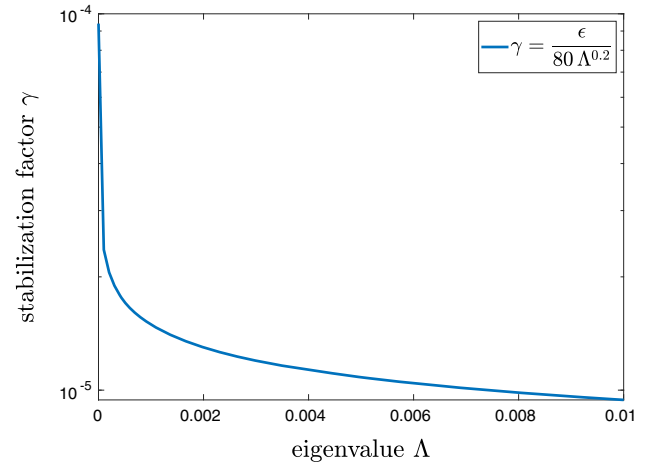
In this section, following Loehnert [29], an eigenvalue stabilization technique will be applied to improve the conditioning and robustness of the FCM. This approach is based on the eigenvalues of the cell's stiffness matrix. Usually, cells that have a small volume fraction and are intersected by the geometry will exhibit eigenvalues that are very close to zero—in addition to the zero eigenvalues that correspond to rigid body modes (RBM). This means that some of the modes will tend to have linear dependencies which result in an increasing condition number. To overcome this problem, following the ideas of Loehnert [29,30] and Loehnert and Beese [31], we can compute the eigenvalues and eigenmodes of each broken cell by factorizing the cell stiffness matrix  $\mathbf{k}^c$  using the eigenvalue decomposition

$$\mathbf{k}^c = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T = (\tilde{\mathbf{V}} \mathbf{V}_0) \begin{pmatrix} \tilde{\mathbf{\Lambda}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{V}}^T \\ \mathbf{V}_0^T \end{pmatrix}. \quad (10)$$

Here,  $\mathbf{k}^c$  is a symmetric positive semi-definite matrix which can be factorized into the diagonal matrix  $\mathbf{\Lambda}$  that contains the eigenvalues with the corresponding eigenmodes  $\mathbf{V}$ . The eigenvalues  $\mathbf{\Lambda}$  consist of non-zero eigenvalues  $\tilde{\mathbf{\Lambda}}$ , with the corresponding eigenvectors  $\tilde{\mathbf{V}}$ , and of eigenvalues that are equal to zero with the related eigenvectors  $\mathbf{V}_0$ . Next, the eigenmodes that correspond to a zero eigenvalue or to a small eigenvalue are grouped into the eigenspace  $\tilde{\mathbf{V}}$  with their corresponding eigenvalues  $\tilde{\mathbf{\Lambda}}$ . The modes that are related to rigid body translations and rotations are removed from the eigenspace  $\tilde{\mathbf{V}}$ . To this end, all modes remaining within the eigenspace  $\tilde{\mathbf{V}}$  need to be supported. To do so, we construct the stiffness matrix

$$\mathbf{k}^\epsilon = \sum_{j=1}^n \epsilon_j \tilde{\mathbf{v}}^j \otimes \tilde{\mathbf{v}}^j. \quad (11)$$

Here,  $\epsilon_j$  defines a fixed stabilization factor for all eigenvalues, set to a small positive value, while  $n$  refers to the total number of modes that need to be supported. The  $\tilde{\mathbf{v}}^j$  corresponds to the  $j$ th eigenmode that belongs to  $\tilde{\mathbf{V}}$ . Alternatively, one can define a nonlinear stabilization factor  $\gamma_j$  that depends on  $\epsilon_j$  as well as the corresponding eigenvalue  $\tilde{\Lambda}_j$ . To this end, the additional stiffness  $\gamma_j$  can be adaptively defined depending on the size of the eigenvalue, see Fig. 4. Thus, Eq. (11)



**Fig. 4** A nonlinear stabilization factor  $\gamma$  with respect to the eigenvalues  $\Lambda$  by setting, for example,  $\epsilon = 3e-4$

becomes

$$\mathbf{k}^\epsilon = \sum_{j=1}^n \gamma_j \tilde{\mathbf{v}}^j \otimes \tilde{\mathbf{v}}^j, \quad (12)$$

with

$$\gamma_j = \frac{\epsilon_j}{80 \tilde{\Lambda}_j^{0.2}}. \quad (13)$$

The condition of the system is improved by adding  $\mathbf{k}^\epsilon$  to the cell stiffness matrix  $\mathbf{k}^c$ , which results in a modified stiffness matrix

$$\tilde{\mathbf{k}}^c = \mathbf{k}^c + \mathbf{k}^\epsilon, \quad (14)$$

as well as to the out of balance vector  $\mathbf{g}^c$  (right-hand side—RHS) of the nonlinear computation—for the sake of consistency—as follows

$$\tilde{\mathbf{g}}^c = \mathbf{g}^c + \mathbf{k}^\epsilon \mathbf{d}^c. \quad (15)$$

Finally, the Newton-Raphson method is applied to solve the modified system

$$\tilde{\mathbf{K}}_T^i(\mathbf{d}^i) \Delta \mathbf{d}^{i+1} = -\tilde{\mathbf{G}}^i(\mathbf{d}^i) \quad (16)$$

where  $\tilde{\mathbf{K}}_T^i(\mathbf{d}^i)$  denotes the modified global tangent stiffness matrix and  $\tilde{\mathbf{G}}^i$  denotes the modified global residual vector, which are the result of the assembly process (7). Having solved the linear system (16), the solution is updated:

$$\mathbf{d}^{i+1} = \mathbf{d}^i + \Delta \mathbf{d}^{i+1}. \quad (17)$$

The procedure for the eigenvalue stabilization is summarized in Algorithm 1. In Sect. 5, we will investigate the influence



of the proposed method on the robustness and the accuracy of the FCM.

---

**Algorithm 1** Eigenvalue stabilization strategy
 

---

```

1: for  $k = 1$  to  $k = k_{\text{end}}$  do ▷ Load increment loop
2:   for  $i = 1$  to  $i = i_{\text{max}}$  do ▷ Newton iteration
3:     Compute  $\mathbf{k}^{c,i}$ , and  $\mathbf{g}^{c,i}$  for each cell  $c$ 
4:     if cell  $c$  is cut then
5:       if  $i = 1$  then
6:         Factorize  $\mathbf{k}^{c,i} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T$  (10)
7:         Identify the modes to be supported  $\tilde{\mathbf{V}}$ 
8:         Remove the modes related to RBMs
9:         Build the stabilization matrix  $\mathbf{k}^\epsilon$  (12)
10:      end if
11:      Modify the stiffness  $\tilde{\mathbf{k}}^{c,i} = \mathbf{k}^{c,i} + \mathbf{k}^\epsilon$  (14)
12:      Correct the RHS  $\tilde{\mathbf{g}}^{c,i} = \mathbf{g}^{c,i} + \mathbf{k}^\epsilon \mathbf{d}^{c,i}$  (15)
13:    end if
14:    Assemble  $\tilde{\mathbf{K}}_T^i, \tilde{\mathbf{G}}^i$  (7)
15:    Solve the system  $\tilde{\mathbf{K}}_T^i \Delta \mathbf{d}^{i+1} = -\tilde{\mathbf{G}}^i$  (16)
16:    Update the solution  $\mathbf{d}^{i+1} = \mathbf{d}^i + \Delta \mathbf{d}^{i+1}$  (17)
17:    Check convergence of Newton iteration
18:    if converged then
19:      break ▷ Proceed with next load step
20:    end if
21:  end for
22: end for
  
```

---

#### 4.1 Detection of the rigid body modes

The eigenvectors with zero eigenvalues that are related to the rigid body modes need to be extracted from the set of modes to be stabilized  $\tilde{\mathbf{V}}$ . Otherwise, stabilizing those modes would lead to a non-physical behavior of the system. To this end, one way to detect the rigid body modes of a cell is to first compute the RBMs analytically. Afterwards, the RBMs can be easily identified and removed from the space  $\tilde{\mathbf{V}}$  by means of a Gram–Schmidt orthogonalization procedure, as proposed by Loehnert [29]. The RBMs can be computed cheaply based on the nodal coordinates of the cells. Generally, for instance, a hexahedral element in a three-dimensional space has six RBMs: three translation modes—which can be calculated by applying a small displacement at every node in the corresponding direction of the mode—and three rotational modes that can be computed using a spherical coordinate system, as suggested by [4,5,23]. The high-order modes of the hierarchical basis can be ignored, since the rigid body motions are represented with the nodal modes.

## 5 Numerical investigations

Before applying the proposed eigenvalue stabilization technique for nonlinear applications, we first investigate it for a

linear problem. Thus, in the first example (Sect. 5.1), a linear elastic isotropic material model is applied with a Young’s modulus of  $E = 50$  MPa and a Poisson’s ratio of  $\nu = 0.3$  assuming small strains. In the second and third application, (Sects. 5.2 and 5.3), a hyperelastic material model is utilized in a finite strain analysis. The strain energy density function [9] is defined as follows

$$W = \frac{\mu}{2} (\text{tr}(\mathbf{C}) - 3) + \frac{\lambda}{4} (J^2 - 1) - \left( \frac{\lambda}{2} + \mu \right) \ln(J), \quad (18)$$

where  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  refers to the right Cauchy–Green tensor and  $J = \det(\mathbf{F})$ . The material parameters read

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} = 28.846 \text{ MPa}, \quad (19)$$

$$\text{and } \mu = \frac{E}{2(1+\nu)} = 19.231 \text{ MPa}, \quad (20)$$

where  $\lambda$  and  $\mu$  denote the Lamé first parameter and the shear modulus, respectively. The first Piola–Kirchhoff stress tensor is computed as

$$\mathbf{P} = 2 \mathbf{F} \frac{\partial W}{\partial \mathbf{C}} = \frac{\lambda}{2} (J^2 - 1) \mathbf{F}^{-T} + \mu (\mathbf{F} - \mathbf{F}^{-T}) \quad (21)$$

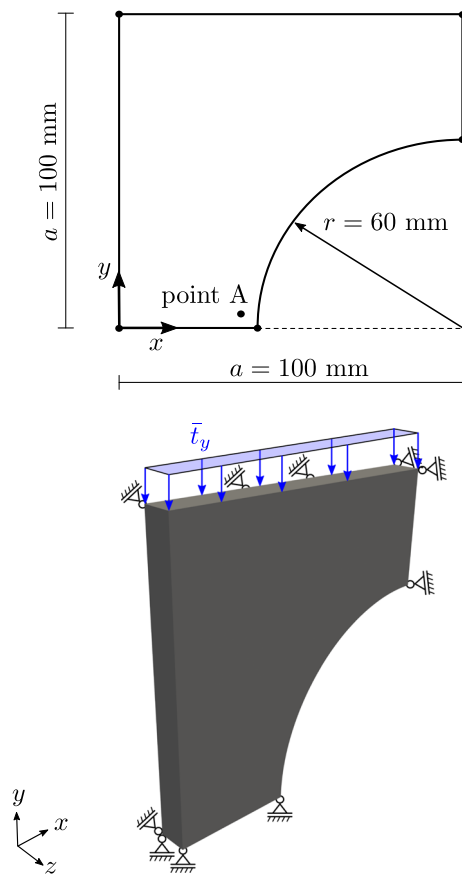
and the elasticity tensor reads

$$\mathbf{A} = \frac{\partial \mathbf{P}}{\partial \mathbf{F}} = \mu (\mathbf{I} - \mathbf{M}) + \frac{\lambda}{2} \left( (J^2 - 1) \mathbf{M} + 2 J^2 (\mathbf{F}^{-T} \otimes \mathbf{F}^{-T}) \right), \quad (22)$$

where  $\mathbf{I}$  refers to the fourth order identity tensor ( $I_{ijkl} = \delta_{ik} \delta_{jl}$ ), and  $\mathbf{M}$  is computed in index notation as  $M_{ijkl} = -F_{jk} F_{li}$ . For all numerical examples, Eq. (12) is used to compute the additional stiffness matrix of the eigenvalue stabilization.

### 5.1 Plate with a cylindrical hole

In the first example, we consider a plate with a cylindrical hole. The goal is to compare the performance of the *eigenvalue stabilization* technique with the  $\alpha$ -method in terms of accuracy and improvement of the condition number. The results will be compared to a reference solution generated from the standard  $p$ -FEM. The geometry of the plate has a side length of  $a = 100$  mm and a radius of  $r = 60$  mm. Symmetry boundary conditions are applied together with a prescribed pressure on the top surface of  $\bar{t}_y = 0.1$  MPa, as can be seen in Fig. 5. The reference solution is obtained with 3200 curved  $p$ -version hexahedral elements utilizing a polynomial degree of  $p = 5$ . The FCM computations are

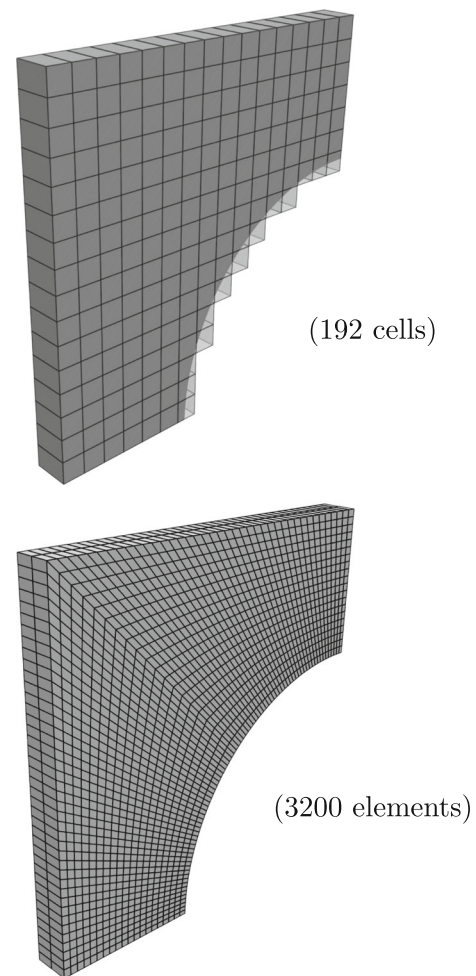


**Fig. 5** Plate with a cylindrical hole: geometry and boundary conditions

performed using 192 hexahedral cells with a subdivision of  $(16 \times 16 \times 1)$  and by increasing the polynomial degree of the hierarchic shape functions,  $p = 1, 2, 3, \dots$ , see Fig. 6. The moment fitting quadrature is used to integrate the broken cells of the FCM with  $(2p + 1)^3$  integration points and an octree depth of  $k = 8$  is applied to compute the moments. In order to stabilize the cells (partially) located in fictitious domain using the  $\alpha$ -method, the standard Gauss-quadrature is utilized with  $(p + 1)^3$  integration points, and the points located in the physical domain are removed, as illustrated in Fig. 3a. Applying the eigenvalue stabilization, no points are added in the fictitious domain.

### 5.1.1 Eigenvalue stabilization without correcting the RHS

First, we investigate the eigenvalue stabilization technique without modifying the load vector as is the case in Eq. (15). This is because, in a linear analysis, it is not possible to compute the correction term  $(k^\epsilon d^c)$  directly without knowing the solution vector  $d^c$ . We study the condition number of the global stiffness matrix, plotted against the number of degrees of freedom, see Fig. 7. Here, different factors are used for  $\epsilon$  (top of Fig. 7) and for  $\alpha$  (bottom of Fig. 7). It can be seen

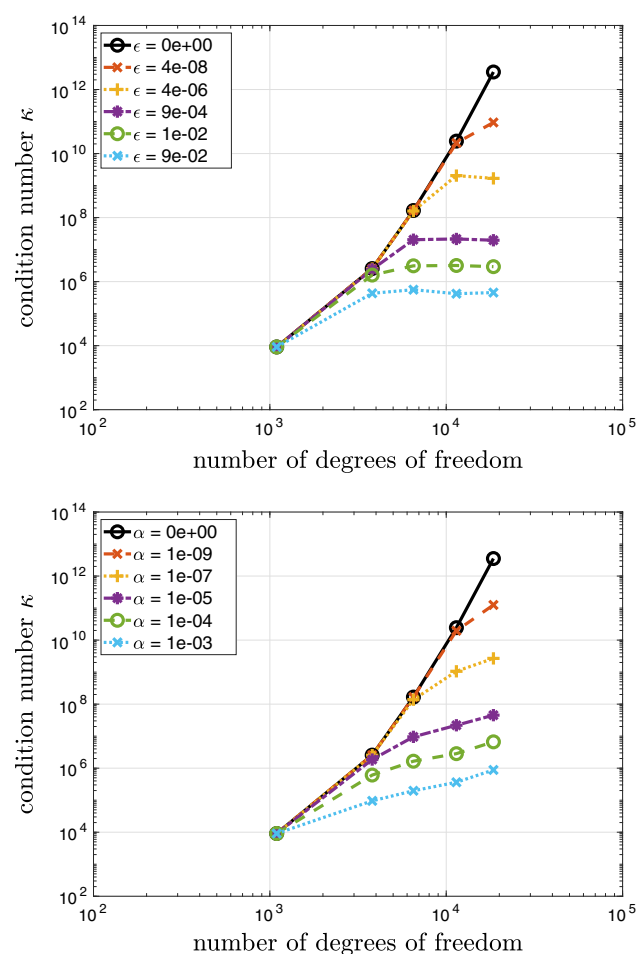


**Fig. 6** Plate with a cylindrical hole. (Top) FCM mesh. (Bottom) FEM mesh

that both methods yield a drastic improvement of the condition number as the stabilization factors increase. Here, we tried to choose the  $\epsilon$  and  $\alpha$  values such that a comparable improvement in the condition number from both methods is obtained.

Furthermore, we examine the accuracy of the von Mises stress  $\sigma_{vM}$  at point A ( $x = 34.0$  mm,  $y = 4.0$  mm,  $z = 1.0$  mm). In Fig. 8, the relative error of the von Mises stress is plotted versus the number of degrees of freedom for an increase of the polynomial degree of the shape functions. The results for the eigenvalue stabilization are depicted at the top, the results for the  $\alpha$ -method at the bottom. Using the  $\alpha$ -method, we do not observe any change in the accuracy for  $\alpha = 0$  to  $\alpha = 1e-7$ . However, by increasing the values from  $\alpha = 1e-5$  to  $\alpha = 1e-3$ , we start to lose accuracy quite significantly as compared to the results of the eigenvalue stabilization. There, we still have a good accuracy until  $\epsilon = 1e-2$  before we start losing accuracy using  $\epsilon = 9e-2$ .

For better visualization, in Fig. 9, the relative error of the von Mises stress is plotted versus the condition number

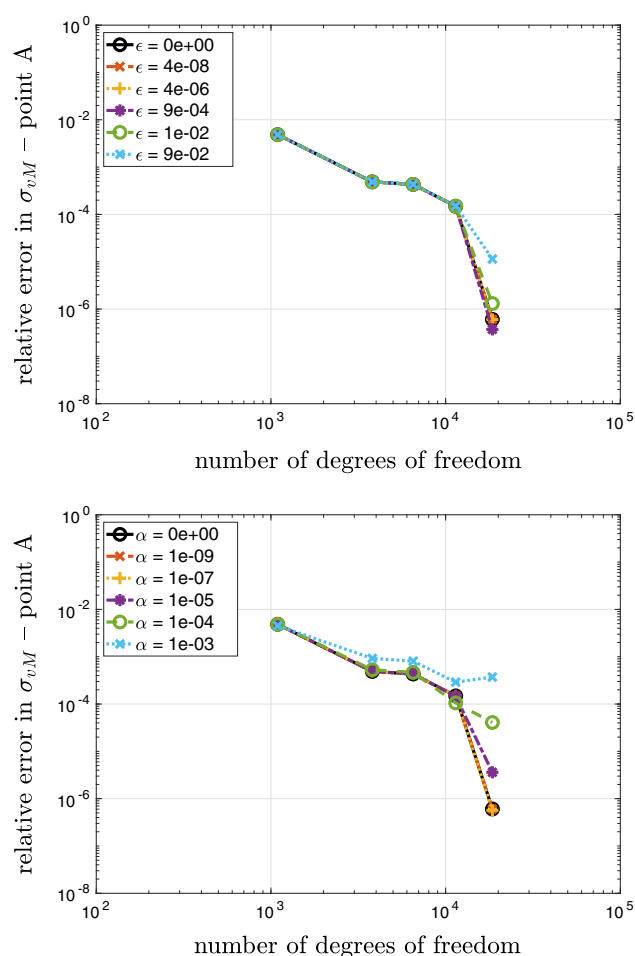


**Fig. 7** Condition number. (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

for an increase of the polynomial degree of the shape functions. It can be seen that the condition number is reduced significantly when using the eigenvalue stabilization (top), without affecting the solution to a great extent as compared to the  $\alpha$ -method (bottom). This is because, in the eigenvalue stabilization, only modes that have a small support (small eigenvalue) are stabilized—whereas a larger loss of accuracy is observed for the  $\alpha$ -method, where all points in the fictitious domain are stabilized with the same amount.

Furthermore, we also investigate the accuracy of the strain energy. In Fig. 10, the relative error in energy norm is plotted versus the number of degrees of freedom for an increase of the polynomial degree of the shape functions. To this end, using the  $\alpha$ -method (bottom of Fig. 10), we do not see any change in the accuracy for  $\alpha = 0$  to  $\alpha = 1e-7$ . However, increasing the values from  $\alpha = 1e-5$  to  $\alpha = 1e-3$ , we start to lose accuracy quite significantly as compared to the results of the eigenvalue stabilization at the top of Fig. 10.

For a better visualization, in Fig. 11, the relative error in energy norm is plotted versus the condition number. It can be seen that the condition number is reduced significantly when



**Fig. 8** Relative error in von Mises stress at point A. (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

using the eigenvalue stabilization (top), without affecting the solution too much as compared to the  $\alpha$ -method (bottom).

### 5.1.2 Eigenvalue stabilization with correcting the RHS

In this section, we apply the eigenvalue stabilization by also modifying the load vector on the RHS. This is done in two steps. First, we compute the solution  $d_1$  by modifying only the stiffness matrix

$$\tilde{K} = K + K^\epsilon \quad (23)$$

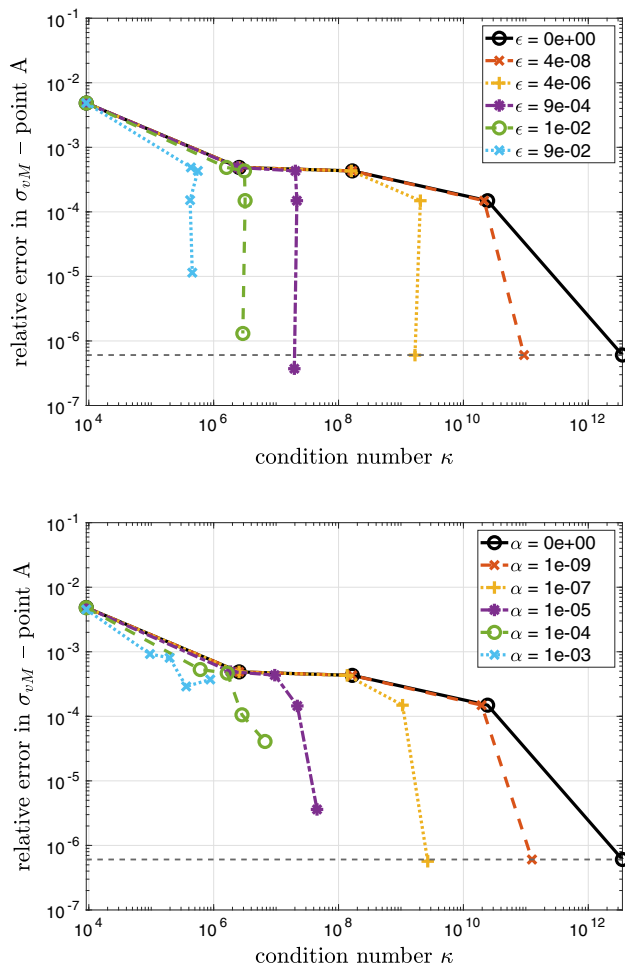
and solving the system

$$\tilde{K} d_1 = F_{\text{ext}}, \quad (24)$$

where  $F_{\text{ext}}$  defines the global external load vector. In the next step, we solve the system again by correcting the load vector iteratively as follows

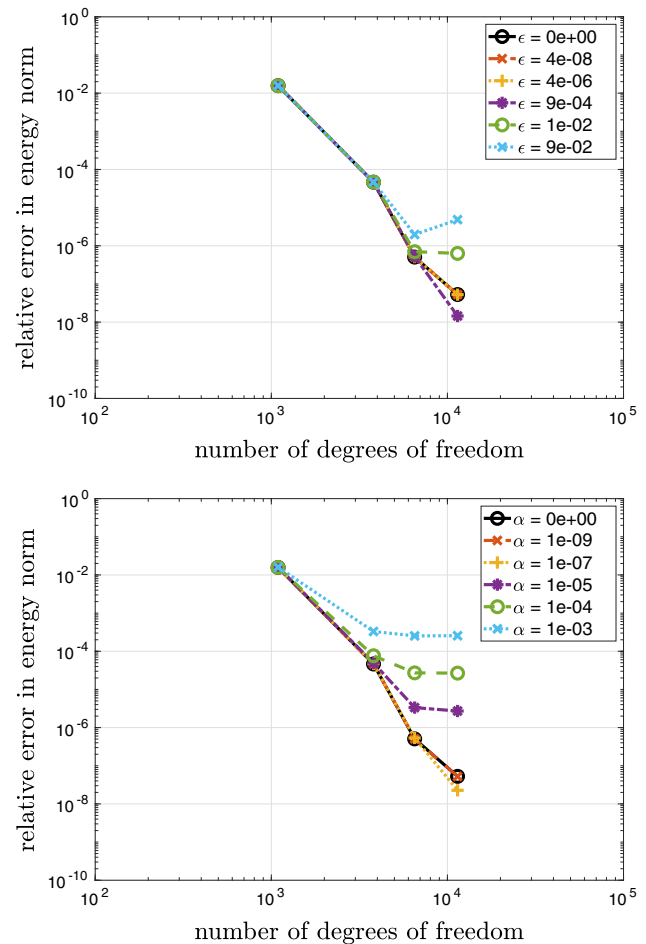
$$\tilde{K} d_{i+1} = F_{\text{ext}} + K^\epsilon d_i. \quad (25)$$





**Fig. 9** Relative error in von Mises stress at point A versus the condition number. (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

In the first iteration ( $i = 1$ ), we apply the solution  $\mathbf{d}_1$ , which is obtained from Eq. (24). Please note that an exact correction of the RHS would call for a repeated solution of the linear system (25) until the displacement does not change anymore. However, we found that three correction steps are already able to provide a good accuracy. By using direct solvers, it is furthermore possible to store the factorization of the global stiffness matrix  $\tilde{\mathbf{K}}$  and re-use it to solve the system for multiple RHSs. In Fig. 12, the relative error of the von Mises stress (top) as well as the relative error in energy norm (bottom) are plotted against the condition number, applying three correction steps. It can be seen that the condition number is improved significantly without affecting the solution at all, provided that the load vector is corrected accordingly. In the nonlinear computations (Sects. 5.2 and 5.3), we correct the RHS during the iterative/incremental procedure using the last computed solution, as illustrated in Algorithm 1.



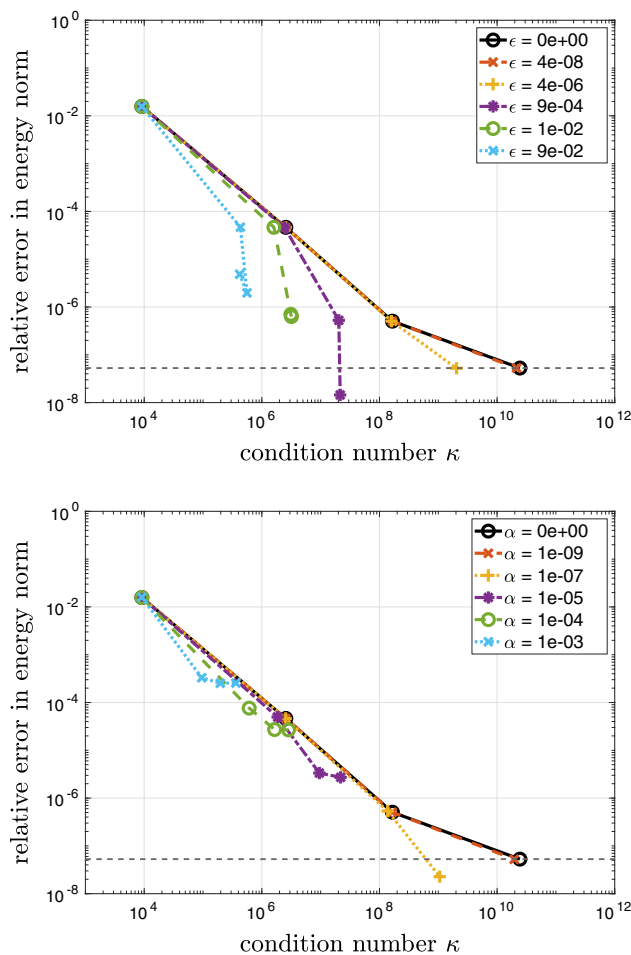
**Fig. 10** Relative error in energy norm. (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

### 5.1.3 Cost of the stabilization

In this section, we investigate the cost of the stabilization utilizing the  $\alpha$ -method as well as the eigenvalue approach. We define the cost of the stabilization by measuring the difference between the computation time of the whole simulation with stabilization ( $T_s$ ) and without stabilization ( $T$ ) as follows

$$x = \frac{T_s - T}{T} \times 100. \quad (26)$$

The same problem setup as described in (Sect. 5.1) is used here with  $\alpha = 1\text{e-}5$  and  $\epsilon = 9\text{e-}4$ . Here, a computer with 2 CPUs each of with 10 cores (40 threads) and 2.4 GHz is used for the simulations. In Fig. 13, the stabilization cost is plotted versus the number of degrees of freedom. It can be seen that the cost of the eigenvalue stabilization is comparable to the  $\alpha$ -method. This shows that the cost of computing



**Fig. 11** Relative error in energy norm versus condition number. (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

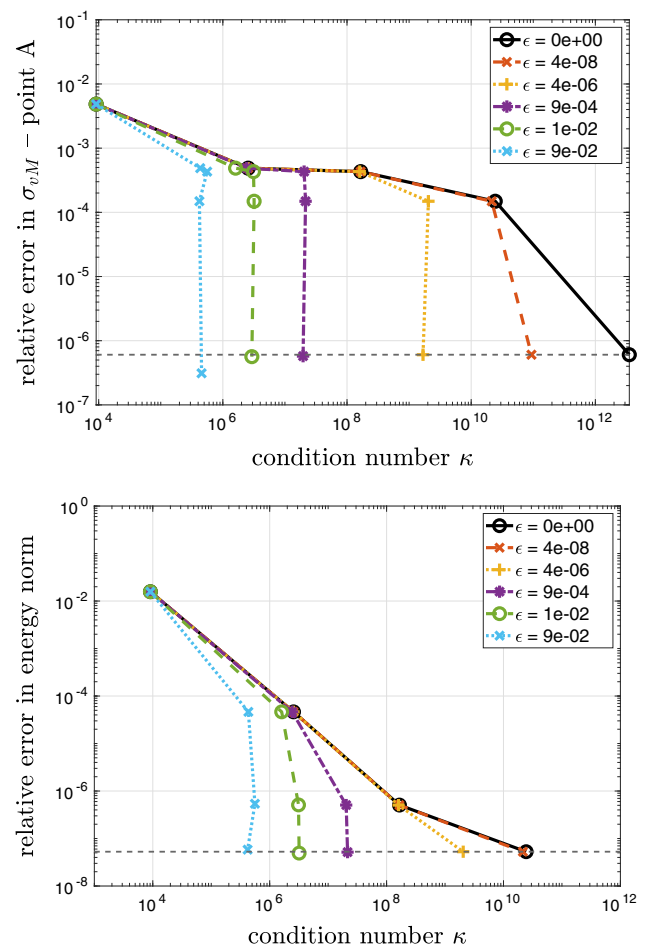
the eigenvalues of the cell stiffness matrices is negligible in comparison to the whole computation since the stabilization is applied on the cell level where every cell runs in parallel.

## 5.2 Single cube connector

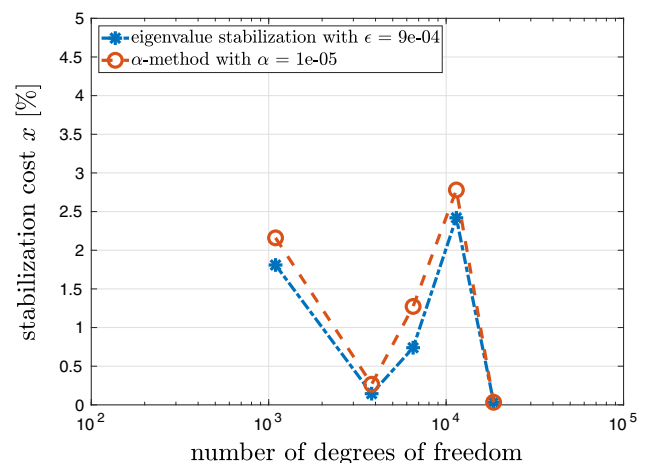
In the following application, we consider a single cube connector [17]. The motivation is to investigate the robustness of the *eigenvalue stabilization* technique for nonlinear computations undergoing large deformations—and to compare it to the  $\alpha$ -method. The geometry of the cube connector is defined using the following level set function

$$\begin{aligned} \phi(\mathbf{x}) = & [(x - x_c)^2 + (y - y_c)^2 - R^2]^2 \\ & + [(y - y_c)^2 + (z - z_c)^2 - R^2]^2 + [(z - z_c)^2 - r^2]^2 \\ & + [(x - x_c)^2 + (z - z_c)^2 - R^2]^2 \\ & + [(x - x_c)^2 - r^2]^2 + [(y - y_c)^2 - r^2]^2 - d^4 \geq 0, \quad (27) \end{aligned}$$

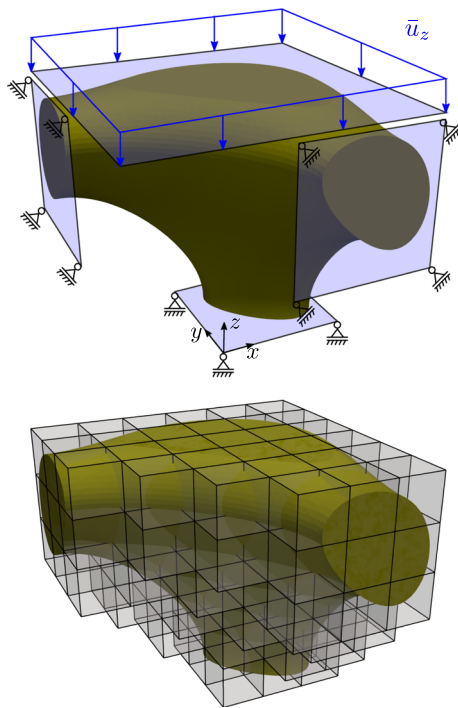
with an inner radius of  $r = 11.25$  mm, an outer radius of  $R = 15$  mm, and a design parameter  $d = 4.6 \times 10^4$  mm<sup>4</sup>. The



**Fig. 12** The eigenvalue stabilization with correction of the RHS in three steps. (top) Relative error in von Mises stress versus condition number. (bottom) Relative error in energy norm versus condition number



**Fig. 13** The computation cost of the stabilization based on the eigenvalues and the  $\alpha$ -method compared to the simulation without stabilization



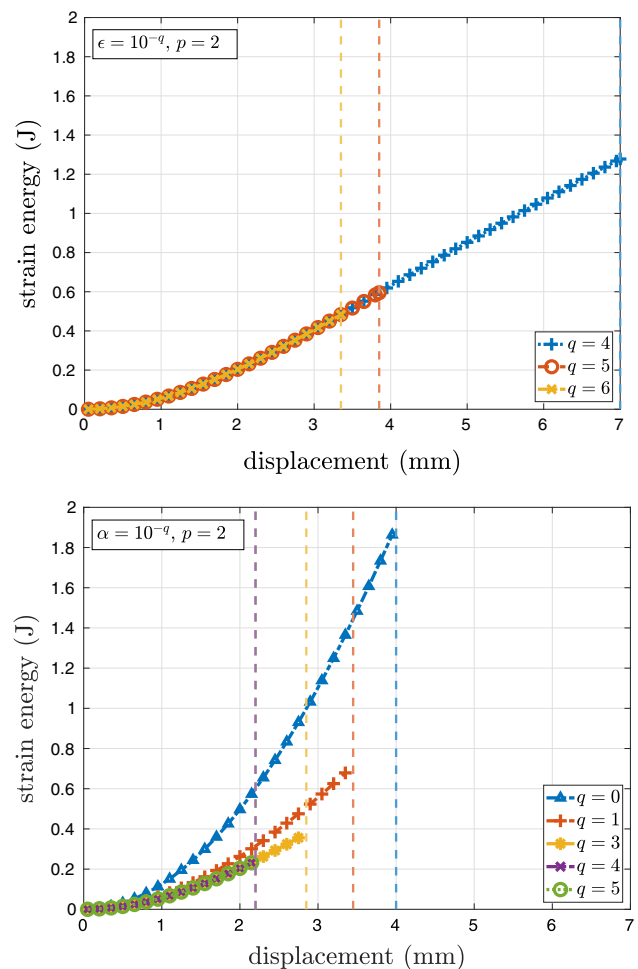
**Fig. 14** Single Cube Connector. Geometry, boundary conditions, and mesh

coordinates of the center are set to  $x_c = y_c = z_c = 0.0$  mm. The indicator function  $\alpha$  can now be defined based on  $\phi(\mathbf{x})$  as

$$\alpha = \begin{cases} 1, & \text{if } \phi(\mathbf{x}) \geq 0 \\ 10^{-q}, & \text{otherwise} \end{cases} \quad (28)$$

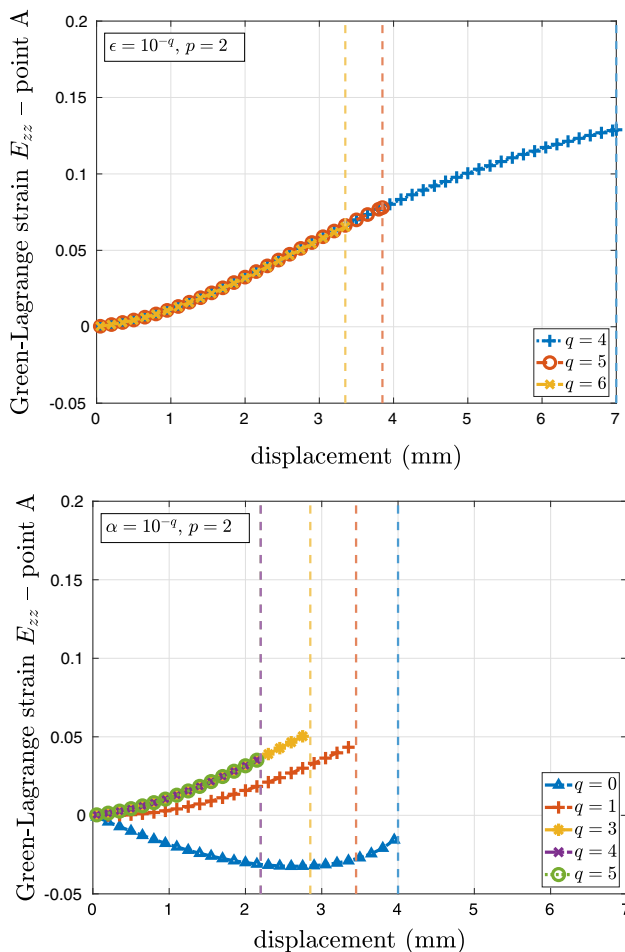
Here, every point  $\mathbf{x}$  within the reference domain ( $\mathbf{x} \in \Omega$ ) corresponds to a value of the level set function that is greater or equal to zero ( $\phi(\mathbf{x}) \geq 0$ ). The strain energy function of the hyperelastic material model is given in Eq. (18) and the corresponding material parameters are presented in Eqs. (19) and (20). Symmetry boundary conditions are applied together with a prescribed displacement at the top surface (compression). The geometry is discretized with 129 cells ( $6 \times 6 \times 6$  subdivisions), as illustrated in Fig. 14. For integrating the broken cells, the adaptive octree with a tree depth of  $k = 3$  is utilized and  $(p + 1)^3$  integration points are distributed in each sub-cell. For stabilizing the fictitious domain using the  $\alpha$ -method, the standard Gauss-quadrature is utilized with  $(p + 1)^3$  integration points, as can be seen in Fig. 3b. Applying the eigenvalue stabilization, a set of  $(p - 1)^3$  integration points is added in the fictitious domain with  $\alpha = 1e-7$ , as can be seen in Fig. 3c. We apply displacement increments of 0.05 mm in each load step, trying to compress the single cube connector as much as possible.

To this end, we start by setting the ansatz order to  $p = 2$  and apply different stabilization factors for the  $\alpha$ -method,



**Fig. 15** Energy-displacement curves for the single cube connector using ansatz order  $p = 2$ . (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

where  $\alpha = 10^{-q}$  with  $q = \{0, 1, 3, 4, 5\}$ , as well as for the eigenvalue stabilization, where  $\epsilon = 10^{-q}$  with  $q = \{4, 5, 6\}$ . It can be observed that using the  $\alpha$ -method, the structure can be deformed up to a displacement of  $\bar{u}_z = 4.0$  mm utilizing  $q = 0$ , as can be seen in Fig. 15 (bottom). However, it is clear that the structure becomes too stiff because of the large stiffness added to the system. For  $q$  values between 3 and 5, the  $\alpha$ -method produces reasonable results with a maximum deformation of  $\bar{u}_z = 2.85$  mm. On the other hand, utilizing the eigenvalue stabilization, we can reach the final deformation state with a displacement of  $\bar{u}_z = 7.00$  mm using  $q = 4$ , which is about 2.5 times more as compared to the  $\alpha$ -method. Additionally, it can be seen that the solution is not modified when using higher  $\epsilon$  values. Furthermore, the Green–Lagrange strain component  $E_{zz}$  is plotted for point A ( $x = 0.73$  mm,  $y = 0.73$  mm,  $z = 0.9$  mm) of a broken cell at every load step. It can be observed that using the  $\alpha$ -method with  $q = 0$  and  $q = 1$  the solution is affected largely for such a local quantity, see Fig. 16 (bottom). On

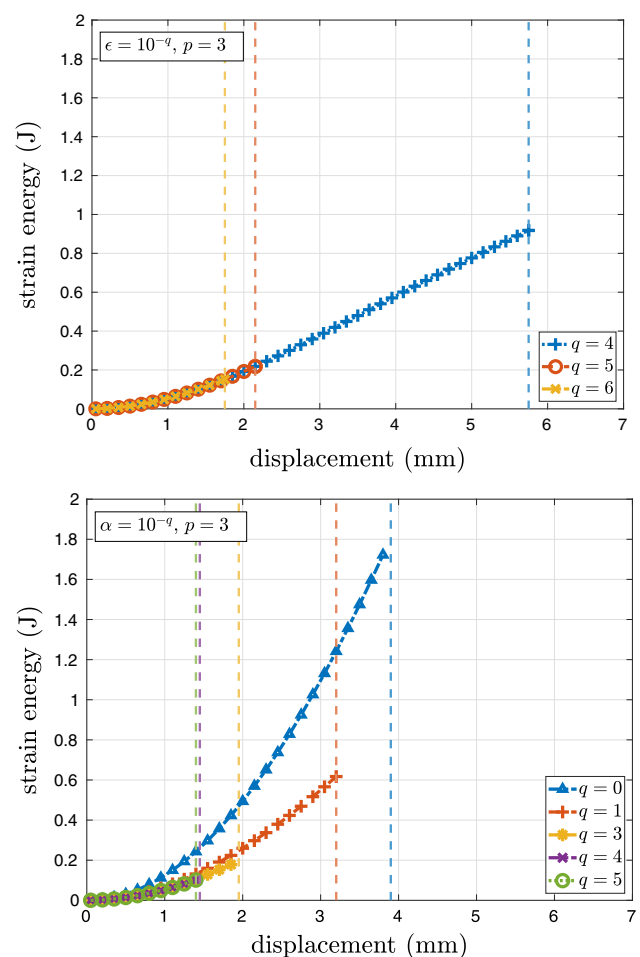


**Fig. 16** The Green–Lagrange strain component  $E_{zz}$  evaluated at point A versus the applied displacement for the single cube connector using ansatz order  $p = 2$ . (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

the other hand, utilizing the eigenvalue stabilization we can reach a much further deformation state without affecting the solution largely, as can be seen in Fig. 16 (top).

Next, we also investigate the strain energy using an ansatz order of  $p = 3$ , as illustrated in Fig. 17. It can be observed that using the  $\alpha$ -method with  $q = 0$  and  $q = 1$  results in a high loss of accuracy. Also, the structure becomes too stiff. However, for  $q$  values between 3 and 5, the  $\alpha$ -method produces reasonable results with a maximum deformation of  $\bar{u}_z = 1.95$  mm using  $q = 3$ , as can be seen in Fig. 17 (bottom). Utilizing the eigenvalue stabilization, we can reach a further deformation state with a displacement of  $\bar{u}_z = 5.75$  mm with  $q = 4$ , which is about 2.95 times more as compared to the  $\alpha$ -method, as can be seen in Fig. 17 (top).

These results show that the strategy of stabilizing the FCM for hyperelastic problems using the technique based on eigenvalues is more robust and accurate as compared to the  $\alpha$ -method. In Fig. 18, the von Mises stress of the single cube connector is plotted at different load steps using the ansatz order of  $p = 3$  and  $\epsilon = 1e-4$ .

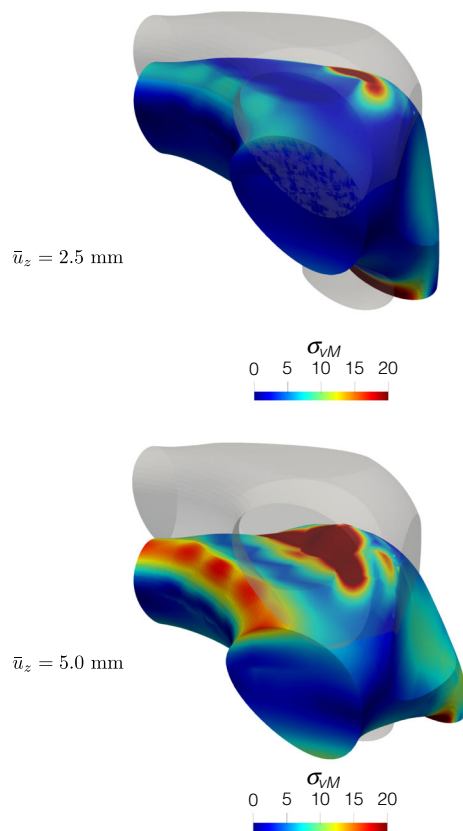


**Fig. 17** Energy-displacement curves for the single cube connector using ansatz order  $p = 3$ . (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method

### 5.3 Single pore of a foam

In the last application, we analyze the performance of the proposed *eigenvalue stabilization* for a more complex structure such as a pore of a foam [21]. The geometry of the foam is obtained from a CT-scan and then converted into a triangulated surface, as depicted in Fig. 19. For the computation, the bottom surface is fixed in all directions, while the top surface is fixed in  $x$  and  $z$ -directions with a prescribed displacement in the  $y$ -direction to compress the foam. The geometry is discretized with 1815 cells ( $20 \times 20 \times 20$  subdivisions). The numerical integration is done the same way as in the previous example (Sect. 5.2). We apply displacement increments of 0.02 mm at each load step and try to compress the geometry as much as possible. To this end, we set the ansatz order to  $p = 2$  and apply different stabilization factors for the  $\alpha$ -method with  $q = \{0, 1, 2, 4, 5\}$ , as well as for the eigenvalue stabilization with  $q = \{4, 6, 8\}$ .

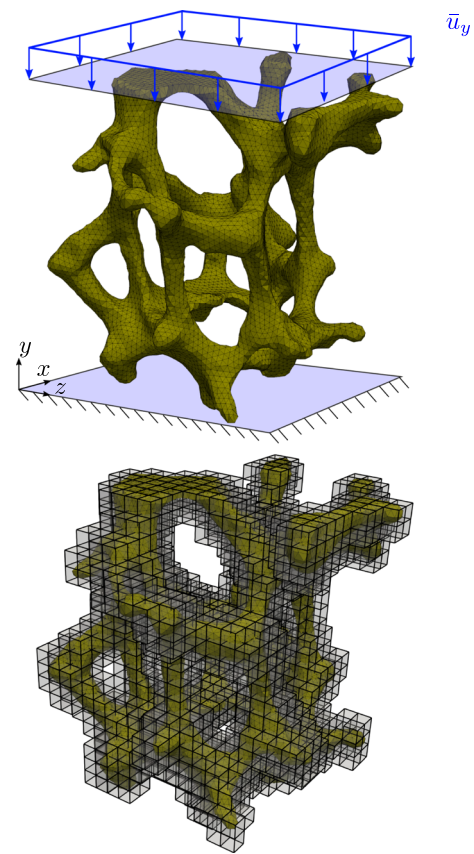
To this end, the strain energy function is plotted for the  $\alpha$ -method at different load steps, as shown in Fig. 20 (bot-



**Fig. 18** von Mises stress  $\sigma_{VM}$  (MPa) for the single cube connector with  $p = 3$  and  $\epsilon = 1e-4$  at different load steps

tom). It can be observed that using  $q = 0$  and  $q = 1$  results in a drastic change in the solution. For  $q = 4$  and  $q = 5$ , the accuracy obtained is reasonable. However, the structure could only be deformed up to a displacement of  $\bar{u}_y = 1.2$  mm using  $q = 4$ . On the other hand, utilizing the eigenvalue stabilization, we can reach a much more pronounced deformation state with a displacement of  $\bar{u}_y = 3.82$  mm with  $q = 4$ , which is about 3.2 times more as compared to the  $\alpha$ -method. Furthermore, we observe no high loss of accuracy when the  $\epsilon$  values increased from  $q = 8$  to  $q = 4$ , as can be seen in Fig. 20 (top).

Furthermore, the Green–Lagrange strain component  $E_{yy}$  is plotted for point A ( $x = 1.9$  mm,  $y = 3.45$  mm,  $z = 4.9$  mm) of a broken cell at every load step. It can be observed that using the  $\alpha$ -method with  $q = 4$  and  $q = 5$  does not affect the solution largely. However the simulations failed to converge much earlier as compared to the eigenvalue stabilization, as can be seen in Fig. 21. Finally, the von Mises stress for the single pore of a foam is plotted in Fig. 22 at different load steps for  $\epsilon = 1e-4$ .

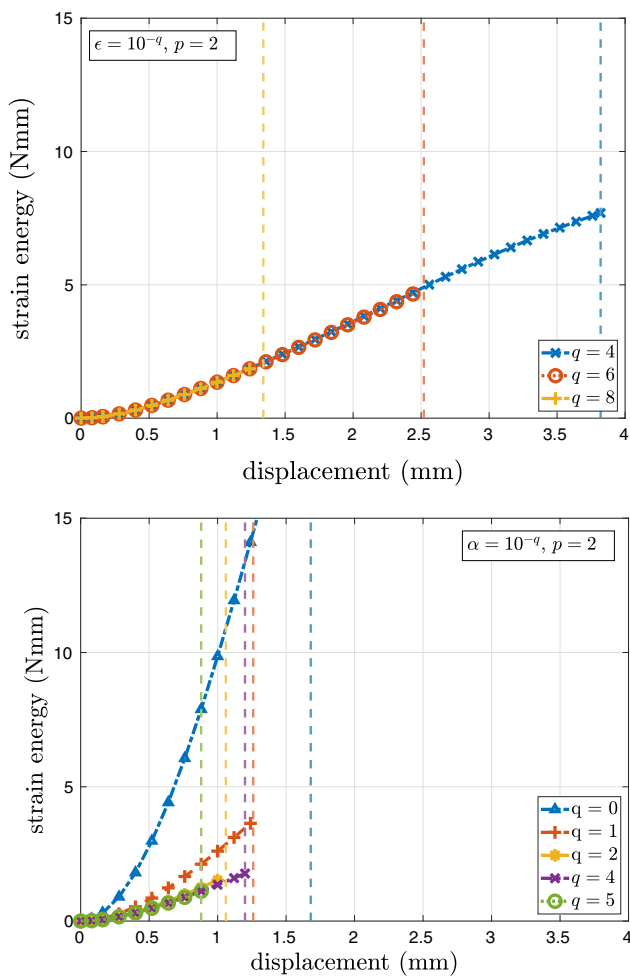


**Fig. 19** Single pore of a foam. Geometry, boundary conditions, and mesh

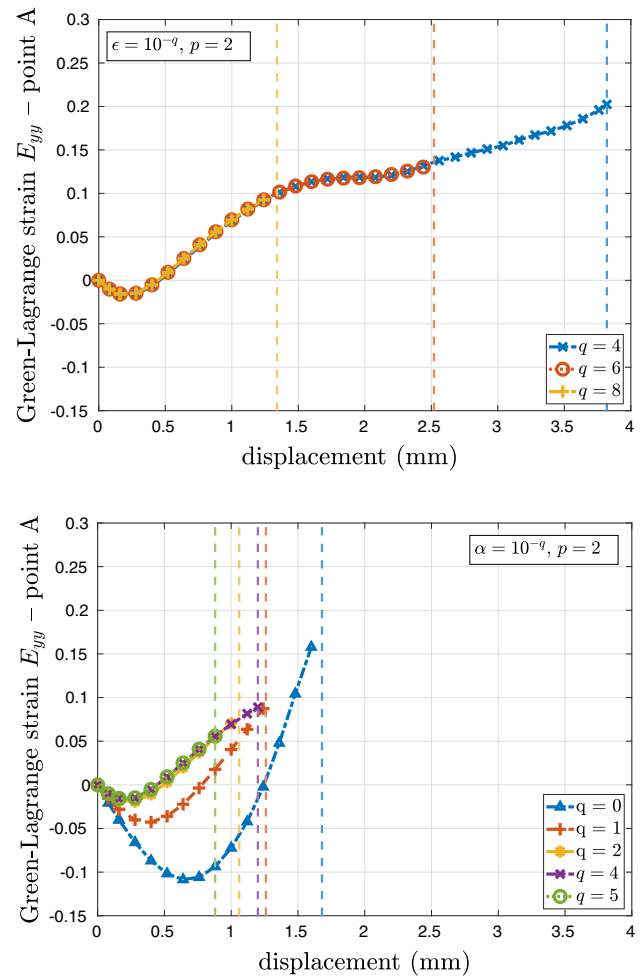
## 6 Conclusions

In this paper, we proposed an eigenvalue stabilization technique for the finite cell method in order to improve its robustness for nonlinear analysis at finite strains. The approach, originally proposed by Loehnert [29], is based on the eigenvalue decomposition of the stiffness matrix for cells that are cut by the boundary of the geometry. To this end, eigenvalues that have a smaller value than a certain tolerance are grouped together. Due to the linear dependencies between the modes, the eigenvalues of some of the modes can even be close to zero—which is why they need additional support. The zero modes that are related to rigid body translations and rotations are not considered for stabilization. Afterwards, the bad modes are supported by adding extra stiffness to the corresponding cell stiffness matrix. Additionally, a correction term is added to the load vector so as to avoid a large modification of the system. Furthermore, we proposed an adaptive stabilization criterion so that the modes with the worst eigenvalues receive more support than modes with better support.

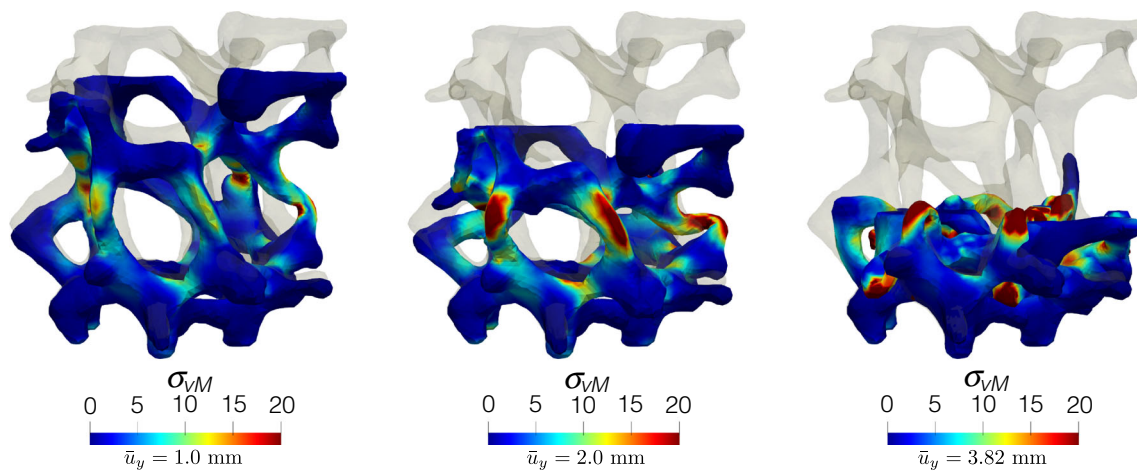




**Fig. 20** Energy-displacement curves for the single pore of a foam using ansatz order  $p = 2$ . (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method



**Fig. 21** The Green-Lagrange strain component  $E_{yy}$  evaluated at point A versus the applied displacement for the single pore of a foam using ansatz order  $p = 2$ . (top) The eigenvalue stabilization. (bottom) The  $\alpha$ -method



**Fig. 22** von Mises stress  $\sigma_{vM}$  (MPa) for the single pore of a foam with  $p = 2$  and  $\epsilon = 1e-4$  at different load steps

We investigated the approach using different numerical examples, comparing the results to the commonly used  $\alpha$ -method. To this end, we showed for the linear analysis that the condition number can be improved significantly without affecting the solution at all, provided that the load vector is corrected accordingly. Moreover, we showed that the method is suitable to improve the robustness of the FCM significantly for nonlinear analysis.

There are several aspects which are of interest for future work. One important point is to extend the proposed stabilization scheme to distinguish between numerical and geometrical or physical instabilities. The stabilization should be applied only to instabilities arising from badly broken cells. Geometrical instabilities related to local buckling or physical instabilities caused by the material behavior should not be affected by the proposed scheme. Furthermore, the stabilization scheme could be combined with iterative solvers acting as a kind of preconditioner for nonlinear problems to be solved with the FCM.

**Acknowledgements** The authors gratefully acknowledge support by the Deutsche Forschungsgemeinschaft in the Priority Program 1748 “Reliable simulation techniques in solid mechanics. Development of non-standard discretization methods, mechanical and mathematical analysis” under the project DU 405/8-2.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Abedian A, Düster A (2019) Equivalent Legendre polynomials: numerical integration of discontinuous functions in the finite element methods. *Comput Methods Appl Mech Eng* 343:690–720. <https://doi.org/10.1016/j.cma.2018.08.002>
2. Abedian A, Parvizia J, Düster A et al (2013) Performance of different integration schemes in facing discontinuities in the finite cell method. *Int J Comput Methods* 10(3):1350002. <https://doi.org/10.1142/S0219876213500023>
3. Abedian A, Parvizia J, Düster A et al (2013) The finite cell method for the  $J_2$  flow theory of plasticity. *Finite Elem Anal Des* 69:37–47. <https://doi.org/10.1016/j.finel.2013.01.006>
4. Baggio R, Franceschini A, Spiezia N et al (2017) Rigid body modes deflation of the preconditioned conjugate gradient in the solution of discretized structural problems. *Comput Struct* 185:15–26. <https://doi.org/10.1016/j.compstruc.2017.03.003>
5. Bathe KJ (1996) Finite element procedures. Prentice Hall, Englewood Cliffs
6. Burman E (2010) Ghost penalty. *CR Math* 348:1217–1220. <https://doi.org/10.1016/j.crma.2010.10.006>
7. Burman E, Hansbo P (2012) Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Appl Numer Math* 62(4):328–341. <https://doi.org/10.1016/j.apnum.2011.01.008>
8. Burman E, Claus S, Hansbo P et al (2015) CutFEM: discretizing geometry and partial differential equations. *Int J Numer Methods Eng* 104:472–501. <https://doi.org/10.1002/nme.4823>
9. Ciarlet PG (1988) Mathematical elasticity, vol 1. Elsevier, Amsterdam
10. Dauge M, Düster A, Rank E (2015) Theoretical and numerical investigation of the finite cell method. *J Sci Comput* 65:1039–1064. <https://doi.org/10.1007/s10915-015-9997-3>
11. Düster A, Allix O (2020) Selective enrichment of moment fitting and application to cut finite elements and cells. *Comput Mech* 65:429–450. <https://doi.org/10.1007/s00466-019-01776-2>
12. Düster A, Hubrich S (2020) Adaptive integration of cut finite elements and cells for nonlinear structural analysis. In: De Lorenzis L, Düster A (eds) Modeling in engineering using innovative numerical methods for solids and fluids. CISM international centre for mechanical sciences book series (CISM, volume 599). Springer, Berlin, chap 2, pp 31–73. [https://doi.org/10.1007/978-3-030-37518-8\\_2](https://doi.org/10.1007/978-3-030-37518-8_2)
13. Düster A, Parvizia J, Yang Z et al (2008) The finite cell method for three-dimensional problems of solid mechanics. *Comput Methods Appl Mech Eng* 197:3768–3782. <https://doi.org/10.1016/j.cma.2008.02.036>
14. Düster A, Rank E, Szabó B (2017) The  $p$ -version of the finite element and finite cell methods. In: Stein E, de Borst R, Hughes TJR (eds) Encyclopedia of computational mechanics second edition, vol Part 1. Solids and structures. Wiley, London, pp 137–171. <https://doi.org/10.1002/9781119176817.ecm2003g>
15. Elfverson D, Larson MG, Larsson K (2018) CutIGA with basis function removal. *Adv Model Simul Eng Sci* 5:2213–7467. <https://doi.org/10.1186/s40323-018-0099-2>
16. Elhaddad M, Zander N, Kollmannsberger S et al (2015) Finite cell method: high-order structural dynamics for complex geometries. *Int J Struct Stab Dyn* 15(7):1540018. <https://doi.org/10.1142/S0219455415400180>
17. Garhuom W, Hubrich S, Radtke L et al (2020) A remeshing strategy for large deformations in the finite cell method. *Comput Math Appl* 80(11):2379–2398. <https://doi.org/10.1016/j.camwa.2020.03.020>
18. Garhuom W, Hubrich S, Radtke L et al (2021) A remeshing approach for the finite cell method applied to problems with large deformations. *Proc Appl Math Mech* 21(1):e202100047. <https://doi.org/10.1002/pamm.202100047>
19. Garhuom W, Hubrich S, Radtke L et al (2022) Adaptive quadrature and remeshing strategies for the finite cell method at large deformations. In: Schröder J, Wriggers P (eds) Non-standard discretisation methods in solid mechanics. Lecture notes in applied and computational mechanics. Springer, Berlin, chap 12. [https://doi.org/10.1007/978-3-030-92672-4\\_12](https://doi.org/10.1007/978-3-030-92672-4_12)
20. Heinze S, Jouliaian M, Düster A (2015) Numerical homogenization of hybrid metal foams using the finite cell method. *Comput Math Appl* 70:1501–1517. <https://doi.org/10.1016/j.camwa.2015.05.009>
21. Heinze S, Bleistein T, Düster A et al (2018) Experimental and numerical investigation of single pores for identification of effective metal foams properties. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik* 98:682–695. <https://doi.org/10.1002/zamm.201700045>

22. Hubrich S, Düster A (2019) Numerical integration for nonlinear problems of the finite cell method using an adaptive scheme based on moment fitting. *Comput Math Appl* 77:1983–1997. <https://doi.org/10.1016/j.camwa.2018.11.030>
23. Jönsthövel TB, van Gijzen MB, Vuik C et al (2013) On the use of rigid body modes in the deflated preconditioned conjugate gradient method. *SIAM J Sci Comput* 35(1):B207–B225. <https://doi.org/10.1137/100803651>
24. Joulaian M, Duczek S, Gabbert U et al (2014) Finite and spectral cell method for wave propagation in heterogeneous materials. *Comput Mech* 54:661–675. <https://doi.org/10.1007/s00466-014-1019-z>
25. Joulaian M, Hubrich S, Düster A (2016) Numerical integration of discontinuities on arbitrary domains based on moment fitting. *Comput Mech* 57:979–999. <https://doi.org/10.1007/s00466-016-1273-3>
26. Kollmannsberger S, D'Angella D, Rank E et al (2019) Spline- and  $hp$ -basis functions of higher differentiability in the finite cell method. *GAMM-Mitteilungen*. <https://doi.org/10.1002/gamm.202000004>
27. Korshunova N, Jomo J, Lékó G et al (2020) Image-based material characterization of complex microarchitected additively manufactured structures. *Comput Math Appl* 80(11):2462–2480. <https://doi.org/10.1016/j.camwa.2020.07.018> (**High-Order Finite Element and Isogeometric Methods 2019**)
28. Kudela L, Zander N, Kollmannsberger S et al (2016) Smart octrees: accurately integrating discontinuous functions in 3D. *Comput Methods Appl Mech Eng* 306:406–426. <https://doi.org/10.1016/j.cma.2016.04.006>
29. Loehnert S (2014) A stabilization technique for the regularization of nearly singular extended finite elements. *Comput Mech* 54:523–533. <https://doi.org/10.1007/s00466-014-1003-7>
30. Loehnert S (2015) Stabilizing the xfem for static and dynamic crack simulations. *Proc Appl Math Mech* 15:137–138. <https://doi.org/10.1002/pamm.201510059>
31. Loehnert S, Beese S (2016) A regularization technique for the xfem: extension to finite deformations, inelastic material behaviour and multifield problems. *Proc Appl Math Mech* 16(1):153–154. <https://doi.org/10.1002/pamm.201610065>
32. Parvzian J, Düster A, Rank E (2007) Finite cell method— $h$ —and  $p$ -extension for embedded domain problems in solid mechanics. *Comput Mech* 41:121–133. <https://doi.org/10.1007/s00466-007-0173-y>
33. Petö M, Duvigneau F, Eisenträger S (2020) Enhanced numerical integration scheme based on image-compression techniques: application to fictitious domain methods. *Adv Model Simul Eng Sci* 7. <https://doi.org/10.1186/s40323-020-00157-2>
34. de Prenter F, Verhoosel CV, van Zwieten GJ et al (2017) Condition number analysis and preconditioning of the finite cell method. *Comput Methods Appl Mech Eng* 316:297–327. <https://doi.org/10.1016/j.cma.2016.07.006>
35. Ruess M, Tal D, Trabelsi N et al (2012) The finite cell method for bone simulations: verification and validation. *Biomech Model Mechanobiol* 11:425–437. <https://doi.org/10.1007/s10237-011-0322-2>
36. Schillinger D, Ruess M (2015) The finite cell method: a review in the context of higher-order structural analysis of CAD and image-based geometric models. *Arch Comput Methods Eng* 22:391–455. <https://doi.org/10.1007/s11831-014-9115-y>
37. Schillinger D, Ruess M, Zander N et al (2012) Small and large deformation analysis with the  $p$ - and B-spline versions of the finite cell method. *Comput Mech* 50:445–478. <https://doi.org/10.1007/s00466-012-0684-z>
38. Szabó B, Babuška I (1991) Finite element analysis. Wiley, London
39. Szabó B, Düster A, Rank E (2004) The  $p$ -version of the finite element method. In: Stein E, de Borst R, Hughes TJR (eds) *Encyclopedia of computational mechanics*, vol 1. Wiley, London, pp 119–139. <https://doi.org/10.1002/0470091355.ecm003g>
40. Taghipour A, Parvzian J, Heinze S et al (2018) The finite cell method for nearly incompressible finite strain plasticity problems with complex geometries. *Comput Math Appl* 75:3298–3316. <https://doi.org/10.1016/j.camwa.2018.01.048>
41. Ventura G, Benvenuti E (2015) Equivalent polynomials for quadrature in Heaviside function enrichment elements. *Int J Numer Methods Eng* 102:688–710. <https://doi.org/10.1002/nme.4679>
42. Verhoosel CV, van Zwieten GJ, Rietbergen B et al (2015) Image-based goal-oriented adaptive isogeometric analysis with application to the micro-mechanical modeling of trabecular bone. *Comput Methods Appl Mech Eng* 284:138–164. <https://doi.org/10.1016/j.cma.2014.07.009>
43. Wriggers P (2008) Nonlinear finite-element-methods. Springer, Berlin
44. Zander N, Kollmannsberger S, Ruess M et al (2012) The finite cell method for linear thermoelasticity. *Comput Math Appl* 64(11):3527–3541. <https://doi.org/10.1016/j.camwa.2012.09.002>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.