Computational Mechanics (2023) 72:743–764
https://doi.org/10.1007/s00466-023-02305-y

ORIGINAL PAPER

Boolean finite cell method for multi-material problems including local enrichment of the Ansatz space

Márton Petö[1] · Sascha Eisenträger[2] · Fabian Duvigneau[1] · Daniel Juhre[1]

Received: 16 November 2022 / Accepted: 17 February 2023 / Published online: 31 March 2023

Abstract

The Finite Cell Method (FCM) allows for an efficient and accurate simulation of complex geometries by utilizing an unfitted discretization based on rectangular elements equipped with higher-order shape functions. Since the mesh is not aligned to the geometric features, cut elements arise that are intersected by domain boundaries or internal material interfaces. Hence, for an accurate simulation of multi-material problems, several challenges have to be solved to handle cut elements. On the one hand, special integration schemes have to be used for computing the discontinuous integrands and on the other hand, the weak discontinuity of the displacement field along the material interfaces has to be captured accurately. While for the first issue, a space-tree decomposition is often employed, the latter issue can be solved by utilizing a local enrichment approach, adopted from the extended finite element method. In our contribution, a novel integration scheme for multi-material problems is introduced that, based on the B-FCM formulation for porous media, originally proposed by Abedian and Düster (Comput Mech 59(5):877–886, 2017), extends the standard space-tree decomposition by Boolean operations yielding a significantly reduced computational effort. The proposed multi-material B-FCM approach is combined with the local enrichment technique and tested for several problems involving material interfaces in 2D and 3D. The results show that the number of integration points and the computational time can be reduced by a significant amount, while maintaining the same accuracy as the standard FCM.

Keywords Finite cell method · Local enrichment · Material interfaces · Discontinuous integrals · Extended finite element method

1 Introduction

1.1 Finite cell method

The finite cell method (FCM) is an embedded domain approach based on an unfitted discretization and high-order shape functions [1–3], enabling an efficient and accurate simulation of geometrically complex structures including hole regions [4,5] and inclusions of additional material [6,7]. As point of departure, a general multi-material problem is depicted in Fig. 1a. Here, $\Omega = \Omega_1 \cup \Omega_2$ is the physical domain of interest, $\partial\Omega$ its boundary, and Γ_D together with Γ_N the Dirichlet and Neumann boundaries, respectively. Finally, Γ_{12} is the material interface between Ω_1 and Ω_2. In the FCM framework, Ω is embedded into a larger domain Ω_e (Fig. 1b), giving rise to the fictitious domain $\Omega_\mathrm{fict} = \Omega_\mathrm{e} \backslash \Omega$ of theoretically zero stiffness. Due to its simple shape, a straightforward discretization of Ω_e is possible by Cartesian meshes (Fig. 1c). Thus, the often cumbersome and error-prone generation of a geometry-conforming discretization by means of distorted finite elements can be avoided.

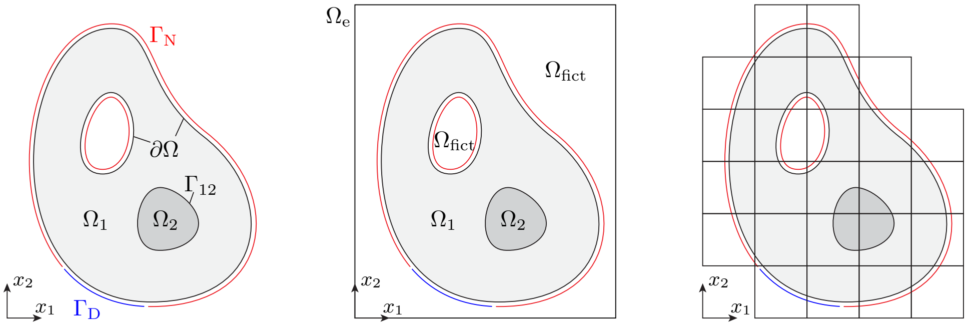
Note that due to the unfitted discretization, cut cells arise, that require extra attention regarding the implementation of boundary conditions [8,9], the condition number of the global system of equations [10–12], and the numerical integration of cell matrices (Sect. 1.2). For the fictitious domains, the volume integrals are penalized by the indicator function

$$\alpha = \begin{cases} 1 & \text{in } \Omega \\ 10^{-q} & \text{in } \Omega_\mathrm{fict} \end{cases} \tag{1}$$

✉ Márton Petö
marton.petoe@ovgu.de

[1] Institute of Mechanics, Otto von Guericke University Magdeburg, Magdeburg, Germany

[2] Institute for Mechanics, Technical University of Darmstadt, Darmstadt, Germany



(a) Original problem (b) Embedded problem (c) Finite cell discretization

Fig. 1 FCM concept

in order to avoid a large energy contribution of Ω_{fict} to the total system (for a more detailed mathematical formulation of the FCM involving void regions, see Sect. 4.1). In contrast to void regions, the displacement field along Γ_{12} is generally C^0-continuous. Such displacement fields are poorly approximated by a linear combination of smooth shape functions over the cut cells. Regardless of the polynomial degree, the insufficiently captured kink in the displacement field causes severe oscillations in the strain and stress fields, which lead to a severely deteriorated accuracy of the simulation and sub-optimal convergences rates [6].

In the *extended finite element method* (XFEM) [13–15], an accurate approximation of the displacement field for multi-material elements is achieved by a *local enrichment* of the Ansatz space. Here, the key idea is based on introducing additional shape functions that are constructed to be C^0-continuous along Γ_{12}. In the context of the FCM, Joulaian and Düster proposed the *hp-d*-FCM and *hp-d*/PUM-FCM approaches for dealing with material interfaces [6,7]. In both cases, the discontinuity is dealt with on a superimposed overlay mesh. In the former approach, the local behavior around the interface is captured by a geometry-conforming mesh, while in the latter one, the local enrichment of the XFEM is adopted. Note however, that in the *hp-d*/PUM-FCM, the unfitted overlay mesh is not necessarily aligned to the base mesh. Thus, it is possible to define larger overlay patches that span multiple cells in the enrichment zone. The weak form corresponding to the multi-material FCM in conjunction with the local enrichment approach is given in Sect. 4.2.

1.2 Integration of discontinuous functions

For an accurate FCM simulation, a proper integration of the cut cell matrices is of crucial importance. Since for discontinuous integrals, the integration accuracy of Gaussian quadrature rules is severely deteriorated, more suitable numerical integration schemes are required. Approaches being available for this purpose can be classified as follows:

1. A main branch is based on partitioning the integration domain into several integration sub-cells using quadtree/octree-decomposition techniques (QTD/OTD). Here, rectangular [5,16] or tetrahedral sub-cells [17,18] are often used, however, geometry-conforming sub-cells with linear [19,20], high-order [21,22] or blending mapping functions [23–25] are also possible.
2. Another branch is represented by the *Moment Fitting* (MF) approach, where in each cut cell, a unique quadrature rule is derived [26,27]. By pre-defining the position of the integration points, the computational complexity of this method can be significantly reduced [28–31]. Furthermore, by appropriate combination with an adaptive space-decomposition technique [32], or by using a non-negative least square solver [33,34], application of MF to non-linear problems is also possible. Finally Düster and Allix proposed and investigated the combination of moment fitting with the local enrichment approach [35].
3. A different idea is seen in the *equivalent polynomials* [36–38] method, which is based on the replacement of the discontinuous integrand by a continuous one yielding the same integral value. Thus, similar to the MF, the computationally often expensive space-partitioning can be avoided.
4. Further methods use the *Divergence Theorem* for reducing higher dimensional integrals to surface and line integrals [39]. Here, radial basis functions [40,41] or pre-derivation can be used [42] for evaluating the antiderivatives required by this method.

Provided that the embedded geometry is accurately captured during the integration phase, exponential convergence rates are possible [10]. For further reading on the key concepts of the FCM, we refer the readers to Refs. [3,43].

1.3 Boolean finite cell method

Due to its robustness and straightforward implementation, the QTD/OTD-based integration scheme was already utilized in the early days of FCM [5,16,44,45] and is still widely used by the community nowadays [46–50]. However, if a high integration accuracy is desired, it requires a large number of integration points, having a significant impact on the computational time. In our previous articles, we introduced an effective approach based on merging the integration sub-cells both in 2D [51,52] and 3D [53] for reduced computational time.

In this contribution, the *Boolean FCM* (B-FCM) [1], whose aim is also the reduction of integration points, is investigated and extended for multi-material problems. The B-FCM for single material problems was also investigated in Ref. [51], where promising results were obtained. The B-FCM extends the integration procedure by Boolean operations and can be easily combined with standard space-decomposition techniques. Originally, it was developed for single-material domains with hole regions. The advantage of the method is best demonstrated by Fig. 2, where Fig. 2a, depicts a cut cell, over which the function αf should be integrated. The standard FCM approach follows the QTD-based integration scheme as depicted in Fig. 2b, requiring 10 sub-cells. In the context of the B-FCM, the same integration accuracy can be obtained for the given example by employing 2 sub-cells only. This is achieved by first, integrating over the entire domain, where α is set to a value of 1 and second, by subtracting the contribution of the fictitious domain, which means that α is set to 0 in the physical part of the sub-cell and takes the value of 1 in the fictitious part. It was shown in Refs. [1] and [51] that by using this approach a significant amount of integration points can be saved.

1.4 Motivation

As mentioned in Sect. 1.3, the B-FCM was originally developed for problems that (i) solely consist of Ω and Ω_{fict}, where (ii) the discontinuity in the integrals is caused by the piecewise constant indicator function α. On the contrary, in case of multi-material problems, the physical domain is composed by n_{d} material sub-domains

$$\Omega = \bigcup_{i=1}^{n_{\text{d}}} \Omega_i, \tag{2}$$

and the discontinuity in the integrand is caused by an arbitrary function D resulting from the enriched Ansatz space. Thus, for extending the single-material B-FCM to a multi-material version, two main steps are needed: (i) a generalization of the Boolean space-partitioning technique for decomposing the multi-material integration domain and (ii) an adjustment of the integrands over the Boolean sub-cells, which depend on the chosen enrichment type.

2 Simplifying integrals with Boolean operations

In this section, the key idea of the Boolean integration scheme is presented for embedded problems that contain several sub-domains with arbitrary functions defined over them.

2.1 Two domains

Let $\Omega \in \mathbb{R}^1$ be a union of the disjoint domains Ω_1 and Ω_2. Over Ω, we consider the continuous and discontinuous functions $f(\xi)$ and $D(\xi)$, respectively, where the latter one is defined as

$$D(\xi) = \begin{cases} D_1(\xi) & \forall \xi \in \Omega_1 \\ D_2(\xi) & \forall \xi \in \Omega_2 \end{cases} . \tag{3}$$

Then, the integral of the product Df over Ω is composed by integrating over the sub-domains Ω_1 and Ω_2 as

$$\int_{\Omega} Df \, \mathrm{d}\xi = \int_{\Omega_1} D_1 f \, \mathrm{d}\xi + \int_{\Omega_2} D_2 f \, \mathrm{d}\xi . \tag{4}$$

Assuming that D_1 is a well defined and known function not only over Ω_1, but also over Ω, Eq. (4) can be reformulated, such that

$$\int_{\Omega} Df \, \mathrm{d}\xi = \int_{\Omega} D_1 f \, \mathrm{d}\xi + \int_{\Omega_2} \big(D_2 - D_1\big) f \, \mathrm{d}\xi , \tag{5}$$

where the integral of the term Df is computed by the following two steps: First, the continuous function $D_1 f$ is integrated over the *entire* domain Ω and second, the integral value of the first term is augmented by $(D_2 - D_1) f$ over Ω_2.

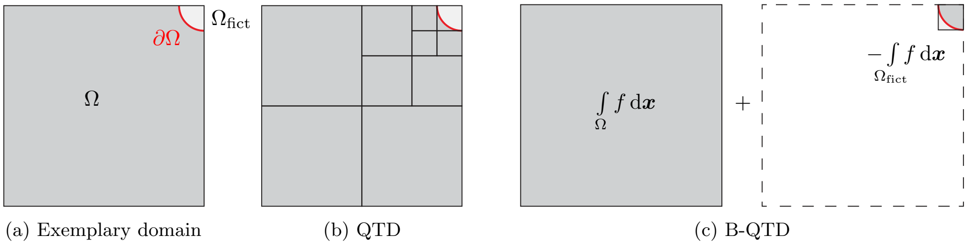
Example: An example visualizing Eq. (5) is given in Fig. 3, where

$$\Omega_1 = \{\xi \mid \xi \in [-1, 0.25]\} \tag{6}$$

$$\Omega_2 = \{\xi \mid \xi \in]0.25, 1]\}. \tag{7}$$

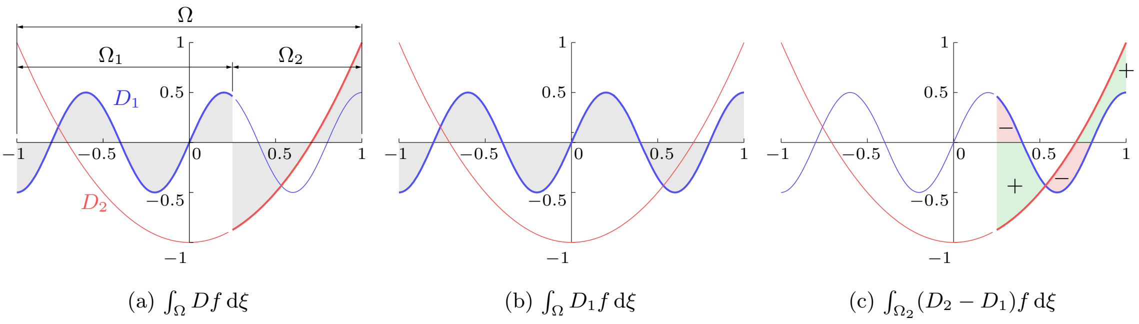
For simplicity, $f(\xi) = 1$ is chosen,[1] and $D(\xi)$ defined as $D_1 = \sin(8\xi)$ in Ω_1 and $D_2 = \xi^2$ in Ω_2. The total integral value of the function Df is depicted by the gray area in Fig. 3a. In Fig. 3b, the first integral on the right hand side of Eq. (5) is visualized. Finally, Fig. 3c depicts the contribution of the second term, where the red (green) color indicates regions that need be subtracted from (added to) the gray area in Fig. 3b in order to obtain the correct integral value.

[1] Note that, as long as it is a smooth function, the complexity of f does not exert any influence on the proposed idea and on the algorithm discussed in the next sections.


Ω_{fict}
$\partial\Omega$
Ω
$\int_\Omega f \, d\boldsymbol{x}$
$+$
$-\int_{\Omega_{\text{fict}}} f \, d\boldsymbol{x}$

(a) Exemplary domain (b) QTD (c) B-QTD

Fig. 2 Two-dimensional example for visualizing the basic concept of the Boolean FCM approach (reproduced from Ref.[1])



(a) $\int_\Omega Df \, d\xi$ (b) $\int_\Omega D_1 f \, d\xi$ (c) $\int_{\Omega_2} (D_2 - D_1) f \, d\xi$

Fig. 3 One-dimensional example for visualizing Eq. (5)

Of course, there is nothing special about choosing Ω_1 and D_1 as the base term in Eq. (5). The formulation works equally well for Ω_2 and D_2. In this case, however, it must hold that D_2 is defined not only over Ω_2 but also over Ω

$$\int_\Omega Df \, d\xi = \int_\Omega D_2 f \, d\xi + \int_{\Omega_1} (D_1 - D_2) f \, d\xi. \tag{8}$$

2.2 Multiple domains

The formulation given in Sect. 2.1 can be also extended to higher dimensional domains composed by $n_d > 2$ disjoint sub-domains according to Eq. (2) as well, where the discontinuous function $D(\boldsymbol{\xi})$ is defined individually over each sub-domain, such that

$$D(\boldsymbol{\xi}) = D_i(\boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Omega_i. \tag{9}$$

Then, for a chosen sub-domain i, Eq. (5) generalizes to

$$\int_\Omega Df \, d\boldsymbol{\xi} = \int_\Omega D_i f \, d\boldsymbol{\xi} + \sum_{\substack{j=1 \\ j \neq i}}^{n_d} \int_{\Omega_j} (D_j - D_i) f \, d\boldsymbol{\xi}, \tag{10}$$

where in the second term, $j = i$ is excluded due to the integral value being zero. Note that each sub-domain i can be chosen as the basis of the formulation, assuming that D_i can be extended from Ω_i to Ω. This is fairly simple for functions that are explicitly given, such as, piece-wise continuous material properties. However, for locally enriched multi-material problems, more steps are required for identifying D_i over Ω (see Sect. 5).

Example: In Fig. 4, a two-dimensional example is given for Eq. (10), where a rectangular domain $\Omega \in \mathbb{R}^2$ consists of three sub-domains ($n_d = 3$). In the current example, the discontinuous function D over Ω is a piece-wise constant function with the values $D_1 = 2$, $D_2 = 4$ and $D_3 = 1$ over the sub-domains Ω_1, Ω_2, and Ω_3, respectively (Fig. 4a)

$$\Omega := \{\boldsymbol{\xi} = [\xi_1, \xi_2]^T \mid (-1 \le \xi_1 \le 1 \text{ and } -1 \le \xi_2 \le 1\} \tag{11}$$

$$\Omega_1 := \{\boldsymbol{\xi} = [\xi_1, \xi_2]^T \mid (\xi_1 + 1)^2 + (\xi_2 + 1)^2 \le 1\} \tag{12}$$

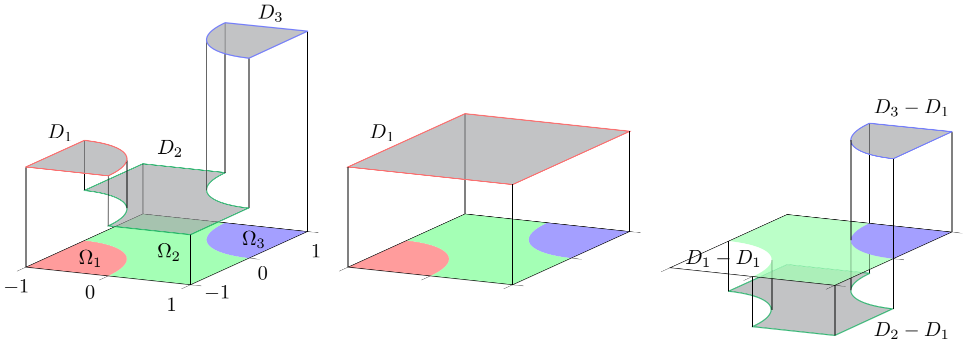
$$\Omega_3 := \{\boldsymbol{\xi} = [\xi_1, \xi_2]^T \mid (\xi_1 - 1)^2 + (\xi_2 - 1)^2 \le 1\} \tag{13}$$

$$\Omega_2 := \Omega \backslash (\Omega_1 \cup \Omega_3). \tag{14}$$

Also in this example, $f = 1$ is selected for simplicity. Choosing the sub-domain $i = 1$ as the basis for the formulation, the first and second terms in Eq. (10) are depicted in Figs. 4b and 4c, respectively. It is easy to see that the integral values based on the direct integration over the individual sub-domains (I) and on the Boolean formulation (I_B) are indeed equal $I = I_B = 4 + \pi$

$$I = D_1 \cdot (\pi/4) + D_2 \cdot (4 - \pi/2) + D_3 \cdot (\pi/4) \tag{15}$$

$$I_B = D_1 \cdot (4) + (D_2 - D_1) \cdot (4 - \pi/2) + (D_3 - D_1) \cdot (\pi/4). \tag{16}$$



(a) $\int_{\Omega} Df\,\mathrm{d}\boldsymbol{\xi}$

(b) $\int_{\Omega} D_1 f\,\mathrm{d}\boldsymbol{\xi}$

(c) $\sum_{j=1}^{3}\int_{\Omega_j}(D_j - D_1)f\,\mathrm{d}\boldsymbol{\xi}$

Fig. 4 Two-dimensional example for visualizing Eq. (10)

3 Combination of Boolean operations and space-tree decomposition techniques

Generally, using Eq. (10) over Eq. (4) does not have major advantages, when Ω_i has a simple shape (e.g., quadrilateral, triangular, hexahedral, tetrahedral, or even circular domains [1]), such that the integrals can be computed either analytically, or via readily available numerical quadrature rules. However, when Ω_i has a complex shape, Eq. (4) is often computed using the QTD and OTD space-tree decomposition techniques. As it will be shown later, if Eq. (10) is combined with a suitable space-decomposition strategy, the Boolean approach is more beneficial than the pure QTD/OTD, as it leads to a smaller number of integration points, while maintaining the integration accuracy.

3.1 Standard space-tree decomposition

The computation of Eq. (4) using the QTD/OTD approach is based on a recursive decomposition of the integration domain into a disjoint set of n_{sc} quadratic/cubic sub-cells $\Omega = \bigcup_{k=1}^{n_{\mathrm{sc}}} \omega_k$, where ω_k is the domain of the kth sub-cell. The partition procedure is applied only to the cut sub-cells, leading to a set of integration domains with decreased size and increased density in the vicinity of the discontinuity, as depicted in Fig. 5 for the domains defined by Eqs. (11)–(14). The tree-depth is controlled by the refinement level $\mathcal{R}$. Finally, the total integral value is computed by integrating the product Df via Gaussian quadrature over the individual sub-cells

$$\int_{\Omega} Df\,\mathrm{d}\boldsymbol{\xi} = \sum_{k=1}^{n_{\mathrm{sc}}} \int_{\omega_k} Df\,\mathrm{d}\boldsymbol{\xi}. \tag{17}$$

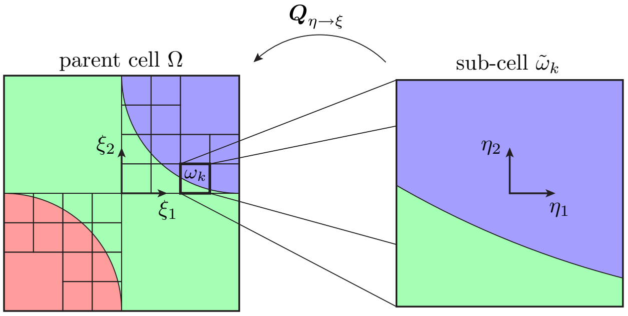


Fig. 5 Partitioning of the the parent cell via QTD with $\mathcal{R} = 3$ and geometry mapping of the sub-cell ω_k

Each sub-cell has its own local coordinate system $\boldsymbol{\eta} \in \mathbb{R}^d$, where d is the dimensionality of the problem. Furthermore, the geometry mapping to the parent cell is established by the transformation $\boldsymbol{\eta} = \boldsymbol{Q}_{\eta\to\xi}(\boldsymbol{\xi})$ involving a uniform scaling and translation. Since integrals in Eq. (17) are computed in the local space of the individual sub-cells, the appropriate change of integration limits has to be taken into account by

$$\int_{\omega_k} D(\boldsymbol{\xi})f(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi} = \int_{\tilde{\omega}_k} D(\boldsymbol{Q}_{\eta\to\xi}(\boldsymbol{\xi}))\,f(\boldsymbol{Q}_{\eta\to\xi}(\boldsymbol{\xi}))\,\det(\boldsymbol{J}_{\eta\to\xi})\,\mathrm{d}\boldsymbol{\eta}, \tag{18}$$

where $\tilde{\omega}_k = [-1, 1] \times [-1, 1]$ denotes the sub-cell in its local coordinate system $\boldsymbol{\eta}$. Furthermore, both the Jacobian matrix $\boldsymbol{J}_{\eta\to\xi} = \mathrm{grad}_{\eta}(\boldsymbol{Q}_{\eta\to\xi})$ and its determinant are constants due to the nature of the geometry mapping. For sake of compactness, wherever it is possible in this article, the integrals over the sub-cells will be formulated w.r.t. $\mathrm{d}\boldsymbol{\xi}$ rather than to $\mathrm{d}\boldsymbol{\eta}$.

3.2 Boolean space-tree decomposition

The proposed Boolean integration scheme extends the conventional QTD/OTD algorithm three-fold, such that the resulting sub-cells are still square/cube-shaped, but (i) they are typically overlapping, (ii) have special labels that contain instructions for computing the integrals over the individual

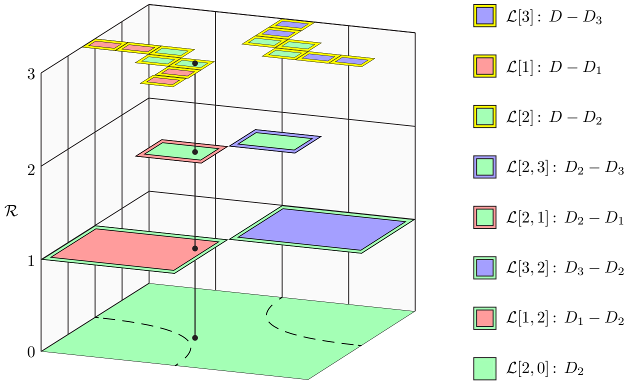

3
2
1
0
$\mathcal{R}$
$\mathcal{L}[3]:\ D - D_3$
$\mathcal{L}[1]:\ D - D_1$
$\mathcal{L}[2]:\ D - D_2$
$\mathcal{L}[2,3]:\ D_2 - D_3$
$\mathcal{L}[2,1]:\ D_2 - D_1$
$\mathcal{L}[3,2]:\ D_3 - D_2$
$\mathcal{L}[1,2]:\ D_1 - D_2$
$\mathcal{L}[2,0]:\ D_2$

Fig. 6 Overlapping sub-cells resulting from the B-QTD algorithm when applied for integrating the function depicted in Fig. 4a

sub-cells (Sect. 3.2.1), and (iii) steer the space-partitioning process (Sect. 3.2.2). In the remainder of this paper, to emphasize the different nature of Boolean sub-cells, they are denoted by ω^{B}, while the QTD/OTD algorithm enhanced by the Boolean approach is referred to as B-QTD/B-OTD. The typical hierarchic structure of the Boolean sub-cells and the distribution of integration points are depicted in Figs. 6 and 7, respectively. The actual meaning of the labels denoted by $\mathcal{L}$ and the color-coding associated with them will be discussed in the following sub-sections. From Fig. 5 vs. Figure 6 and Fig. 7a vs. Figure 7b, the key idea and the potential of the Boolean approach for reducing the number of sub-cells n_{SC} and integration points n_{IP} is already visible. In fact, in Sect. 3.3 and 6, it will be shown that the much more efficient distribution of integration points via the B-QTD/B-OTD algorithm leads to reduction in computational time by approximately 70–80%, while maintaining the same accuracy.

3.2.1 Sub-cell labels

The Boolean approach sketched in Eq. (10) is based on integrating a discontinuous function in two steps: (i) a continuous function is integrated over a simple square/cube-shaped domain and (ii) a discontinuous function is integrated over a specific sub-domain, where the integrand results from the subtraction of two known functions. Although in the B-QTD/B-OTD, the integral value is not computed in two steps like in case of Eq. (10), but in n_{sc} steps

$$\int_{\Omega} Df \,\mathrm{d}\boldsymbol{\xi} = \sum_{k=1}^{n_{\mathrm{sc}}} \int_{\omega_k^{\mathrm{B}}} \mathcal{D} f \,\mathrm{d}\boldsymbol{\xi}, \tag{19}$$

each sub-cell is directly connected to one of the two terms in Eq. (10). Thus, although multiple calculations are required, in essence, the same procedure as described before applies. The actual purpose of a given sub-cell and the form of the discontinuous function $\mathcal{D}$ in the integrand for that specific sub-cell, is explicitly encoded in the developed labeling system. In particular, two different label types have been developed, which are comprehensively discussed in the following paragraphs.

Label type 1: The label of type 1 is composed by two domain IDs i and j in a specific order, and refers to the following integral over the given sub-cell

$$\mathcal{L}[i,j]: \quad +\int_{\omega_k^{\mathrm{B}}} (D_i - D_j) f \,\mathrm{d}\boldsymbol{\xi}. \tag{20}$$

The above integral essentially corresponds to the first term on the right hand side in Eq. (10), where the whole domain of interest is integrated and the discontinuity is ignored, regardless of whether the sub-cell is cut or not. However, instead of just using a single value of D_i like in Eq. (10), the term $D_i - D_j$ is inserted, such that an overlapping sub-cell structure can be exploited. Note that label type 1 is always assigned to sub-cells created at the levels $\mathcal{R} < \mathcal{R}_{\max}$.

Additionally, we would like to express integrals, where only D_i or $-D_j$ is used. Therefore, an additional zero value $D_0 := 0$ is declared, while keeping in mind that the actual domain IDs start from 1, not from 0, cf. Eq. (2). Thus, using $i = 0$ or $j = 0$, special cases of Eq. (20) are also possible, which are needed for the algorithm to work

$$\mathcal{L}[i,0]: +\int_{\omega_k^{\mathrm{B}}} D_i f \,\mathrm{d}\boldsymbol{\xi}, \tag{21}$$

$$\mathcal{L}[0,j]: -\int_{\omega_k^{\mathrm{B}}} D_j f \,\mathrm{d}\boldsymbol{\xi}. \tag{22}$$

Label type 2: The second label type contains only a single domain ID j, and refers to the integral

$$\mathcal{L}[j]: \quad \int_{\omega_k^{\mathrm{B}}} (D - D_j) f \,\mathrm{d}\boldsymbol{\xi}, \tag{23}$$

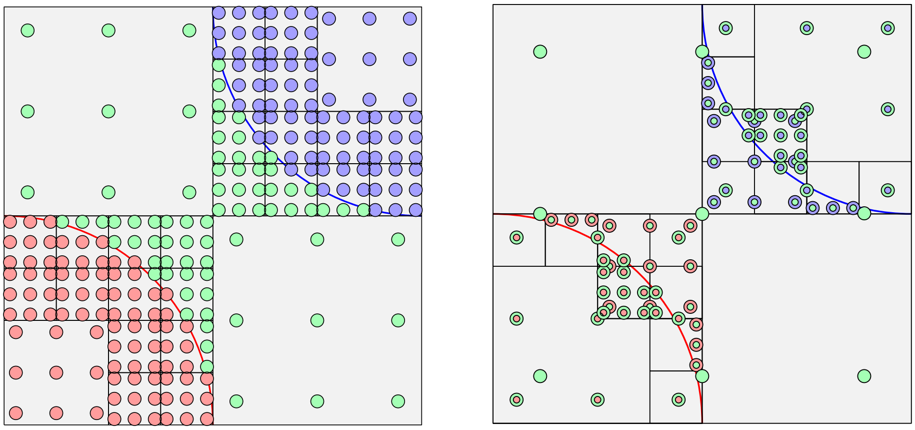
where, unlike for type 1 labels, the discontinuity is taken into account due to the presence of D. The above integral corresponds to the second term on the right hand side in Eq. (10). Here, the discontinuity is taken into account and hence, the integration error resulting from the label type 1 sub-cells is corrected. Note that for points lying in Ω_j, the integrand is vanishing due to Eq. (9)

$$D - D_j = 0 \text{ in } \Omega_j. \tag{24}$$

This fact constitutes a key feature of the B-QTD/B-OTD algorithm, since every sub-cell and integration point located in Ω_j can be directly discarded.

Similar to Eqs. (21) and (22), if $j = 0$ is used, Eq. (23) turns into simply integrating the discontinuous function Df over ω_k^{B}

Fig. 7 Distribution of integration points using the QTD and B-QTD methods with $\mathcal{R} = 3$where each sub-cell contains 3×3 integration points



(a) QTD ($n_{\mathrm{SC}} = 28$, $n_{\mathrm{IP}} = 252$)

(b) B-QTD ($n_{\mathrm{SC}} = 19$, $n_{\mathrm{IP}} = 81$)

$$\mathcal{L}[0]: \quad \int_{\omega_k^{\mathrm{B}}} (D - D_0) f \, \mathrm{d}\boldsymbol{\xi} \quad = \quad \int_{\omega_k^{\mathrm{B}}} D f \, \mathrm{d}\boldsymbol{\xi}. \tag{25}$$

due to the applied definition $D_0 = 0$.

3.2.2 Labelling of new sub-cells

The labeling procedure is illustrated in Fig. 8 for the partitioning step of a quadratic sub-cell $\omega_k \in \mathbb{R}^2$ with the type 2 label $\mathcal{L}[j]$. In principle, there are three cases how the B-QTD can proceed from this stage, which are elaborated below. Note that at the very beginning of the Boolean decomposition algorithm, $j = 0$ is set for the label.

Case 1: If the given sub-cell is not cut, it must completely belong to a single domain $\omega_k \subset \Omega_i$. The sub-cell is excluded from further partitioning and its label modifies to $\mathcal{L}[i, j]$ (orange box in Fig. 8).

If the sub-cell is cut, four children are created and the following two additional steps are executed for the Boolean labeling process: (i) Find the material ID i for the sub-cell ω_k^{B}, for which $\Omega_i^k = \omega_k^{\mathrm{B}} \cap \Omega_i$ is the largest. In the current implementation, this is achieved by testing a set of sample points. (ii) Test whether Ω_i^k occupies at least $^1/_4{}^{\mathrm{th}}$ of ω_k^{B} in 2D and $^1/_8{}^{\mathrm{th}}$ in 3D. This condition is denoted by $\mathcal{C}$ in Fig. 8.

Case 2: If $\mathcal{C} = \texttt{false}$, the parent sub-cell is deleted and all of its children inherit its label (green box in Fig. 8). Essentially, this scenario corresponds to the general procedure of the standard QTD, with the difference, that the sub-cells are also labeled. Since $\mathcal{C}$ is violated, even the largest sub-domain has an area less then 25% of that of the parent sub-cell. Thus, none of the children can belong to a single sub-domain only. Note that $\mathcal{C} = \texttt{false}$ only applies in cases where the sub-cells have at least 5 sub-domains.

Case 3: If $\mathcal{C} = \texttt{true}$ for a cut sub-cell, the following procedure applies (blue box in Fig. 8): (i) Keep the parent sub-cell and assign label $\mathcal{L}[i, j]$ and (ii) assign the label $\mathcal{L}[i]$ to the children.

The benefit of the labeling procedure in this case is demonstrated in Fig. 9, where the example domain is discretized by a single cell with $\mathcal{R} = 1$. Since $\mathcal{C} = \texttt{true}$ for Ω_2, the parent cell is kept and the labels are assigned according to Fig. 9b, where the white text boxes are used for numbering the sub-cells. In Fig. 9c, the vanishing integrand property given in Eq. (24) is utilized: Firstly, since sub-cells No. 3 and 4 lie completely in Ω_2, these can be directly discarded. Secondly, while sub-cells No. 2 and 5 are kept, a further reduction of integration points is still possible due to the integrand beeing $D_2 - D_2$ in $\omega_2^{\mathrm{B}} \cap \Omega_2$ and $\omega_5^{\mathrm{B}} \cap \Omega_2$ (green domain in Fig. 9c). Both, discarding the entire sub-cells and the integration points are possible due to the profound interplay between the assigned labels.

Remark 1 Note that performing a standard QTD and then assigning labels to the sub-cells is not possible. The labeling process has to be carried out parallel to the QTD.

The procedure depicted in Fig. 8 is repeated for each cut sub-cell until the final refinement level $\mathcal{R}_{\max}$ is reached. Every time $\mathcal{C} = \texttt{true}$, both a parent sub-cell and some of its children are kept. Using $\mathcal{R}_{\max} = 3$ for Fig. 9a results in the set of overlapping sub-cells depicted in Fig. 6, where the $\mathcal{R}$-axis indicates the refinement levels the individual sub-cells were generated on. A special color-coding is used to indicate the different sub-cell labels, whose corresponding integrands are depicted in the right side of Fig. 6.

Examining an arbitrary point $P = [-0.1, -0.35] \in \Omega_2$ in Fig. 6, the integrand at P should evaluate to $I = D_2$ (cf. Fig. 4a). By adding up the integrands represented by the Boolean sub-cells containing the given point,

$$I = \underbrace{D_2}_{\mathcal{R}=0} + \underbrace{(D_1 - D_2)}_{\mathcal{R}=1} + \underbrace{(D_2 - D_1)}_{\mathcal{R}=2} + \underbrace{(D_2 - D_2)}_{\mathcal{R}=3} = D_2, \tag{26}$$

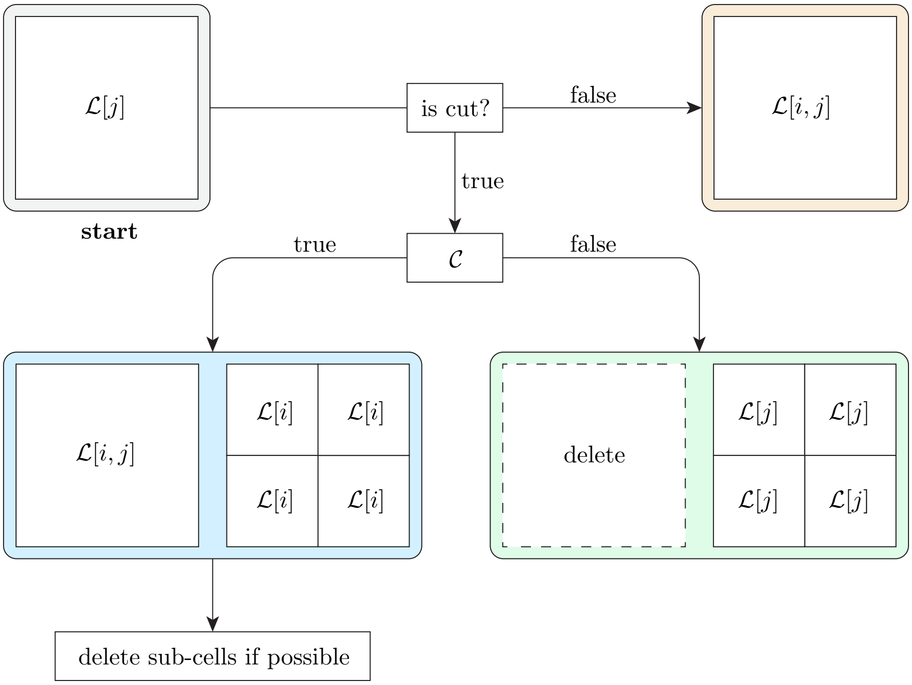
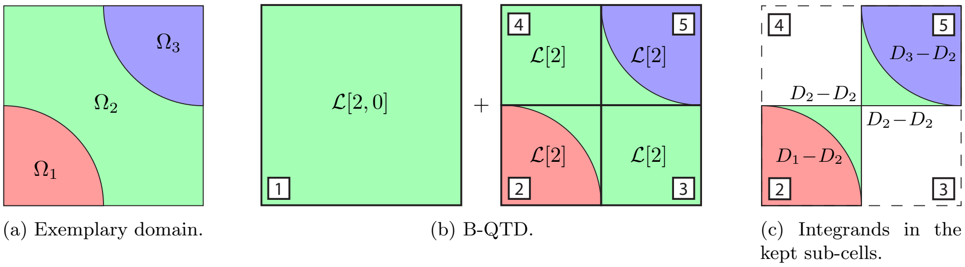


Fig. 8 Illustration of the partitioning step and labeling of the parent sub-cell and its children



(a) Exemplary domain.

(b) B-QTD.

(c) Integrands in the kept sub-cells.

Fig. 9 Two-dimensional example for visualizing Eq. (10)

it is evident, that the equality indeed holds. Note that on $\mathcal{R} = 3$, the type 2 label is used as defined in Eq. (23).

Finally, the effect of the labeling concept on the integration points is depicted in Fig. 10, where 3 × 3 integration points are distributed in each sub-cell of Fig. 6 and the same coloring scheme is used. Additionally, for each integration point, the corresponding integrand is given. The difference between the different label types is clear: Since sub-cell labels at $\mathcal{R} < \mathcal{R}_{\max}$ are not taking the discontinuity into account (red and blue dashed lines), all integration points are used in the sub-cells. For the sub-cells at $\mathcal{R} = \mathcal{R}_{\max}$, label type 2 is used, where the discontinuity (red and blue solid lines) is taken into account during the integration. These cells generally contain integration points with vanishing integrand, such that they can be excluded from the integration procedure due to Eq. (24).

3.3 Performance

In this sub-section, the performance of the B-QTD and B-OTD algorithms is investigated in terms of the reduction of integration points and achieved integration accuracy. Due to the Boolean sub-cell labels, the B-QTD has a much more efficient distribution of the integration points than the standard QTD. This can be clearly recognized in Fig. 7 presented earlier, where the integration points for the B-QTD are based on Fig. 10. Expressed quantitatively, the number of integration points using the QTD integration scheme $n_{\mathrm{IP}}^{\mathrm{QTD}} = 252$ drops to $n_{\mathrm{IP}}^{\mathrm{B-QTD}} = 81$ for the current example, when the QTD is extended by the Boolean approach. Using these quantities, the reduction of integration points is introduced as

$$r_{\mathrm{IP}} = \left(1 - \frac{n_{\mathrm{IP}}^{\mathrm{B-QTD}}}{n_{\mathrm{IP}}^{\mathrm{QTD}}}\right) \times 100\%, \qquad (27)$$

where $r_{\mathrm{IP}} > 0\,\%$ indicates a meaningful reduction and $r_{\mathrm{IP}} = 0\,\%$ means no reduction. For the example given in Fig. 7, a significant saving by $r_{\mathrm{IP}} = 67.86\%$ can be obtained. Similar to r_{IP}, a reduction rate of sub-cells r_{SC} can also be defined. Both of these quantities are depicted in Fig. 11a for various refinement levels, where a reduction rate up to $r_{\mathrm{IP}} \approx 80\%$ is observed. Note that due to the additional savings of integration points in the leaf sub-cells (cf. Fig. 10d), $r_{\mathrm{IP}} \geq r_{\mathrm{SC}}$.

Furthermore, let e_{I} express the relative integration error

$$e_{\mathrm{I}} = \left(1 - \frac{I}{I_{\mathrm{ref}}}\right) \times 100\%, \tag{28}$$

where I is the integral value obtained by different numerical methods and I_{ref} the reference value. For the current example I_{ref} is given in Eq. (15). Figure. 11b reveals that despite the significant reduction of integration points, the B-QTD yields the same integration accuracy as the QTD.

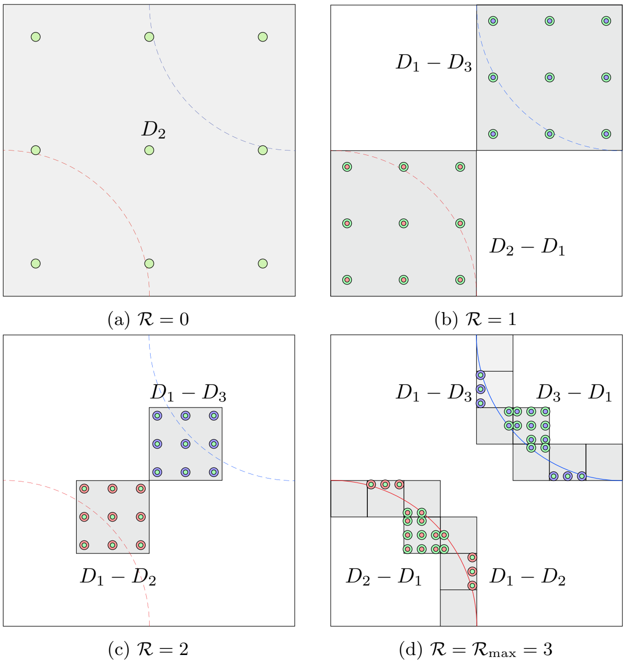
The same significant reduction rate and no loss of accuracy can be observed for 3D problems as well, as depicted in Fig. 12 for a cubic domain composed by the sub-domains Ω_1, Ω_2, Ω_3, defined below.

$$\Omega := \{\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]^{\mathrm{T}} \mid (-1 \leq \xi_i \leq 1 \text{ for } i = 1, 2, 3\} \tag{29}$$

$$\begin{aligned}\Omega_1 := \{\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]^{\mathrm{T}} \mid (\xi_1 + 1)^2 \\ + (\xi_2 + 1)^2 + (\xi_3 + 1)^2 \leq 1\}\end{aligned} \tag{30}$$

$$\begin{aligned}\Omega_2 := \{\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]^{\mathrm{T}} \mid (\xi_1 - 1)^2 \\ + (\xi_2 - 1)^2 + (\xi_3 - 1)^2 \leq 1\}\end{aligned} \tag{31}$$

$$\Omega_3 := \Omega \backslash (\Omega_1 \cup \Omega_2) \tag{32}$$



(a) $\mathcal{R} = 0$

(b) $\mathcal{R} = 1$

(c) $\mathcal{R} = 2$

(d) $\mathcal{R} = \mathcal{R}_{\max} = 3$

Fig. 10 Integration points in the overlapping sub-cells of Fig. 6 and their corresponding integrands

4 Multi-material finite cell method

In this section, the discussion in Sect. 1.1 is extended by the required mathematical formulation for multi-material problems. Let us consider a material domain Ω that is composed by n_{d} disjoint sub-domains $\Omega = \cup_{i=1}^{n_{\mathrm{d}}} \Omega_i$, which are either void regions or other material domains.

4.1 Finite cell method

The original FCM formulation is concerned with porous media, i.e., the given problem consists of a single material and multiple void regions [3–5,43]. It has been shown, that exponential convergence rates can be obtained [10]. Without derivation, the weak formulation of linear elasticity in the context of the FCM reads

$$\textit{Find } \boldsymbol{u} \in \mathcal{S}^{\mathrm{h}} \textit{ such that, } \mathcal{B}_{\mathrm{e}}(\boldsymbol{u}, \boldsymbol{v}) = \mathcal{F}_{\mathrm{e}}(v), \quad \forall \boldsymbol{v} \in \mathcal{V}^{\mathrm{h}}. \tag{33}$$

Here, $\boldsymbol{u}$ is the displacement field, $\boldsymbol{v}$ denotes the test function, and $\mathcal{S}^{\mathrm{h}}$ together with $\mathcal{V}^{\mathrm{h}}$ represent the appropriate finite dimensional function spaces, which in this article, are based on the spectral element concept [54–56]. Furthermore, $\mathcal{B}_{\mathrm{e}}$ and $\mathcal{F}_{\mathrm{e}}$ are bilinear and linear functionals, respectively, defined over the extended domain Ω_{e}

$$\mathcal{B}_{\mathrm{e}}(\boldsymbol{u}, \boldsymbol{v}) = \int\limits_{\Omega_{\mathrm{e}}} \alpha \rho \ddot{\boldsymbol{u}} \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} + \int\limits_{\Omega_{\mathrm{e}}} (\boldsymbol{L}\boldsymbol{u})^{\mathrm{T}} \alpha \boldsymbol{C} \boldsymbol{L} \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} \tag{34}$$

$$\mathcal{F}_{\mathrm{e}}(\boldsymbol{v}) = \int\limits_{\Omega_{\mathrm{e}}} \boldsymbol{v}^{\mathrm{T}} \alpha \boldsymbol{b} \,\mathrm{d}\boldsymbol{x} + \int\limits_{\Gamma_{\mathrm{N}}} \boldsymbol{v}^{\mathrm{T}} \hat{\boldsymbol{t}} \,\mathrm{d}\boldsymbol{x}. \tag{35}$$

In the above equations, α is the indicator function defined in Eq. (1) for distinguishing between the physical and fictitious domains, while ρ and $\boldsymbol{C}$ are the material density and stiffness, respectively. Furthermore, $\ddot{\boldsymbol{u}}$ is the acceleration field, $\boldsymbol{b}$ stands for the body load vector, and $\hat{\boldsymbol{t}}$ represents the prescribed tractions along the Neumann boundary Γ_{N}. Considering a two-dimensional setting for the sake of simplicity, in each finite cell, the displacement field $\boldsymbol{u}^{(c)} = [u_1^{(c)}, u_2^{(c)}]^{\mathrm{T}}$ is approximated using smooth shape functions

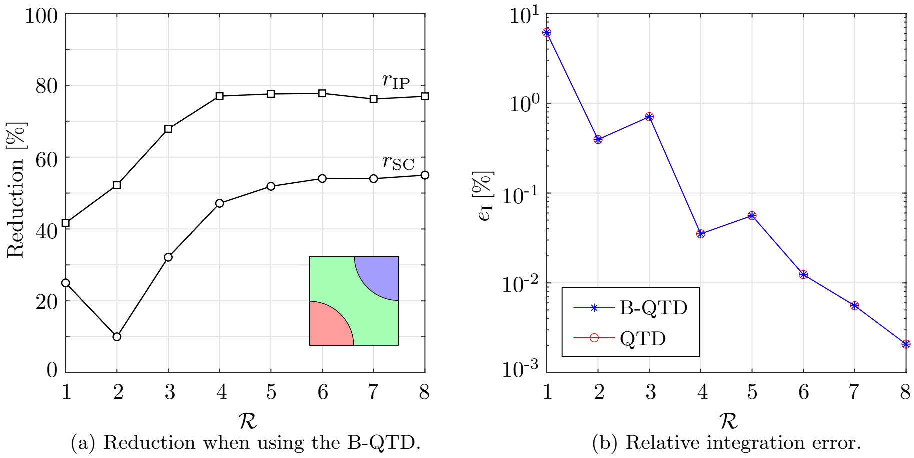
$$\boldsymbol{u}^{(c)} = \boldsymbol{N}_{\mathrm{u}} \boldsymbol{U}^{(\mathrm{c})}, \tag{36}$$

where $\boldsymbol{N}_{\mathrm{u}} \in \mathbb{R}^{2 \times 2n_{\mathrm{N}}}$ is a matrix containing the shape functions associated with the n_{N} nodes of the given cell

$$\boldsymbol{N}_{\mathrm{u}} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & \cdots & N_{n_{\mathrm{N}}} & 0 \\ 0 & N_1 & 0 & N_2 & 0 & \cdots & N_{n_{\mathrm{N}}} \end{bmatrix}, \tag{37}$$

and $\boldsymbol{U}^{(\mathrm{c})}$ contains the unknown nodal displacements. The cell matrices as well as the global system matrices of the

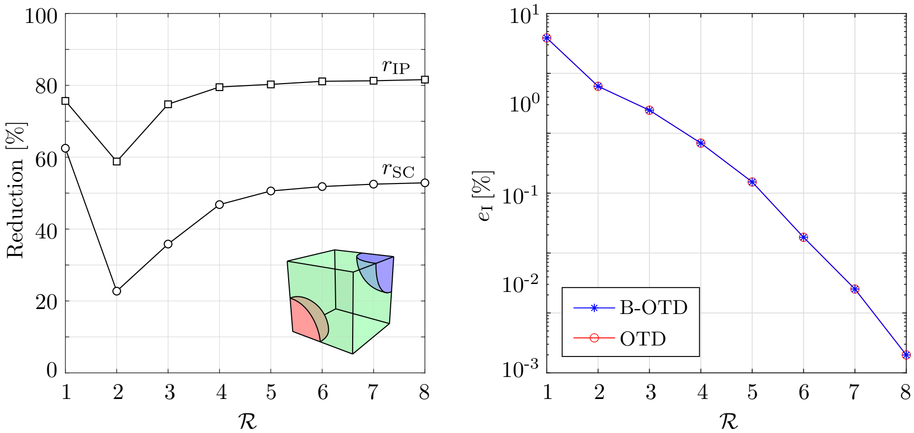
Fig. 11 Comparison of the QTD ond B-QTD algorithms



(a) Reduction when using the B-QTD.

(b) Relative integration error.

Fig. 12 Comparison of the QTD ond B-QTD algorithms



(a) Reduction when using the B-OTD.

(b) Relative integration error.

discretized problem will be given in Sect. 4.3 after a brief introduction to the local enrichment approach in Sect. 4.2.

4.2 Local enrichment

In the following, the focus is kept on multi-material problems without any void regions. Thus, $\alpha = 1$ applies to all cases. Nonetheless, the material properties are still discontinuous due to $\boldsymbol{C}(\boldsymbol{x}) = \boldsymbol{C}_i(\boldsymbol{x})$ and $\rho(\boldsymbol{x}) = \rho_i(\boldsymbol{x})\ \forall \boldsymbol{x} \in \Omega_i$. Furthermore, at the material interfaces, the displacements are generally C^0-continuous while stress and strains obey certain jump conditions. For locally enriched problems, the weak formulation reads

$$\textit{Find } \boldsymbol{u} \in \mathcal{S}^{\mathrm{h}},\ \textit{such that} \\ \begin{cases} \mathcal{B}_{\mathrm{e}}(\boldsymbol{u}_{\mathrm{u}} + \boldsymbol{u}_{\mathrm{a}}, \boldsymbol{v}_{\mathrm{u}}) = \mathcal{F}_{\mathrm{e}}(\boldsymbol{v}_{\mathrm{u}}), & \forall \boldsymbol{v}_{\mathrm{u}} \in \mathcal{V}_{\mathrm{u}}^{\mathrm{h}} \\ \mathcal{B}_{\mathrm{e}}(\boldsymbol{u}_{\mathrm{u}} + \boldsymbol{u}_{\mathrm{a}}, \boldsymbol{v}_{\mathrm{a}}) = \mathcal{F}_{\mathrm{e}}(\boldsymbol{v}_{\mathrm{a}}), & \forall \boldsymbol{v}_{\mathrm{a}} \in \mathcal{V}_{\mathrm{a}}^{\mathrm{h}} \end{cases}, \tag{38}$$

where $\boldsymbol{u}$ is composed of the smooth $\boldsymbol{u}_{\mathrm{u}}$ and enriched displacements $\boldsymbol{u}_{\mathrm{a}}$, such that $\boldsymbol{u} = \boldsymbol{u}_{\mathrm{u}} + \boldsymbol{u}_{\mathrm{a}}$. Furthermore, $\boldsymbol{v}_{\mathrm{u}}$ and $\boldsymbol{v}_{\mathrm{a}}$ are the test functions associated with the base $\mathcal{V}_{\mathrm{u}}^{\mathrm{h}}$ and enriched function spaces $\mathcal{V}_{\mathrm{a}}^{\mathrm{h}}$, respectively. Following from Eq. (38), for the approximation of $\boldsymbol{u}^{(c)}$, Eq. (36) turns into

$$\boldsymbol{u}^{(c)} = \boldsymbol{N}_{\mathrm{u}} \boldsymbol{U}^{(\mathrm{c})} + \boldsymbol{N}_{\mathrm{a}} \boldsymbol{A}^{(\mathrm{c})}, \tag{39}$$

where the first term is identical to the standard approximation in Eq. (36) and the second term includes the matrix of enrichment shape functions

$$\boldsymbol{N}_{\mathrm{a}} = \psi \boldsymbol{N}_{\mathrm{u}} \tag{40}$$

and enrichement DOFs $\boldsymbol{A}$. There are multiple ways how the enrichment function ψ in Eq. (40) can be defined. In this contribution, we follow the *modified-abs enrichment* proposed by Moës et al. [14]

$$\psi = \boldsymbol{N}|\boldsymbol{\varphi}| - |\boldsymbol{N}\boldsymbol{\varphi}|, \tag{41}$$

which is based on a level-set function φ_i of the embedded geometries, where the domain and boundary of the ith geometric entity is defined by

$$\Omega_i = \{\boldsymbol{x} \mid \varphi_i(\boldsymbol{x}) < 0\}, \tag{42}$$

$$\Gamma_i = \{\boldsymbol{x} \mid \varphi_i(\boldsymbol{x}) = 0\}. \tag{43}$$

In Eq. (41), $\boldsymbol{N}$ and $\boldsymbol{\varphi}$ are vectors containing the shape functions and level-set values associated with the individual nodes, respectively

$$\boldsymbol{N} = \left[N_1 \; N_2 \; \cdots \; N_{n_\mathrm{N}}\right]^\mathrm{T}, \tag{44}$$

$$\boldsymbol{\varphi} = \left[\varphi_1 \; \varphi_2 \; \cdots \; \varphi_{n_\mathrm{N}}\right]^\mathrm{T}. \tag{45}$$

Furthermore, the last term in Eq. (41) ensures the C^0-continuity of ψ along the interface, and subtracted from the first term, it yields a ψ that vanishes on the element edges that are not intersected by the boundary. Thus, parasitic terms and partially enriched transition elements can be avoided [57].

Note that in the general case, $\boldsymbol{N}_\mathrm{u}$, $\boldsymbol{N}_\mathrm{a}$ and $\boldsymbol{N}$ can be based on different Ansatz spaces and meshes [6]. In this contribution, for all of these quantities, Lagrangian shape functions are used based on the Gauss–Legendre–Lobatto (GLL) nodal distribution. While for $\boldsymbol{N}_\mathrm{u}$, a user defined polynomial order p is used, the polynomial degree p_ψ of $\boldsymbol{N}$ depends on the complexity of intersecting interface [6]. Finally, following from Eq. (40), the polynomial degree of $\boldsymbol{N}_\mathrm{a}$ is $p_\mathrm{a} = p + p_\psi$.

4.3 Discretization

Discretization of the coupled weak form leads to cell-specific stiffness matrices including terms that are only related to (i) the smooth displacements $\boldsymbol{K}^\mathrm{c}_\mathrm{uu}$, (ii) the enriched displacements $\boldsymbol{K}^\mathrm{c}_\mathrm{aa}$, and (iii) appropriate coupling terms $\boldsymbol{K}^\mathrm{c}_\mathrm{ua} = (\boldsymbol{K}^\mathrm{c}_\mathrm{au})^\mathrm{T}$. Following the QTD/OTD-based integration scheme often used in the FCM (see Sect. 3.1), $\boldsymbol{K}^c$ is computed by integrating over the n_sc sub-cells

$$\boldsymbol{K}^c = \begin{bmatrix} \boldsymbol{K}^\mathrm{c}_\mathrm{uu} & \boldsymbol{K}^\mathrm{c}_\mathrm{ua} \\ \boldsymbol{K}^\mathrm{c}_\mathrm{au} & \boldsymbol{K}^\mathrm{c}_\mathrm{aa} \end{bmatrix} = \sum_{k=1}^{n_\mathrm{sc}} \int_{\omega_k} \begin{bmatrix} \boldsymbol{B}_\mathrm{u}^\mathrm{T}\boldsymbol{C}\boldsymbol{B}_\mathrm{u} & \boldsymbol{B}_\mathrm{u}^\mathrm{T}\boldsymbol{C}\boldsymbol{B}_\mathrm{a} \\ \boldsymbol{B}_\mathrm{a}^\mathrm{T}\boldsymbol{C}\boldsymbol{B}_\mathrm{u} & \boldsymbol{B}_\mathrm{a}^\mathrm{T}\boldsymbol{C}\boldsymbol{B}_\mathrm{a} \end{bmatrix} \mathrm{d}\boldsymbol{x}. \tag{46}$$

In the above equation, $\boldsymbol{B}_\mathrm{u}$ and $\boldsymbol{B}_\mathrm{a}$ are the strain-gradient operators associated with the standard and enriched displacements, respectively. Similar to $\boldsymbol{B}_\mathrm{u}$, $\boldsymbol{B}_\mathrm{a}$ is constructed as

$$\boldsymbol{B}_\mathrm{a} = [\boldsymbol{B}_\mathrm{a}^{(1)},\; \boldsymbol{B}_\mathrm{a}^{(2)}, \ldots, \boldsymbol{B}_\mathrm{a}^{(n_\mathrm{N})}], \tag{47}$$

where $\boldsymbol{B}_\mathrm{a}^{(l)}$ is associated with the lth enrichment shape function, and for two-dimensional linear elastic problems it is of the form

$$\boldsymbol{B}_\mathrm{a}^{(l)} = \begin{bmatrix} \dfrac{\partial(\psi N_l)}{\partial x_1} & 0 \\ 0 & \dfrac{\partial(\psi N_l)}{\partial x_2} \\ \dfrac{\partial(\psi N_l)}{\partial x_2} & \dfrac{\partial(\psi N_l)}{\partial x_1} \end{bmatrix}. \tag{48}$$

The differentiation of the individual entries w.r.t. x_m is realized using the product rule

$$\frac{\partial(\psi N_l)}{\partial x_m} = \frac{\partial \psi}{\partial x_m} N_l + \frac{\partial N_l}{\partial x_m}\psi, \tag{49}$$

with $m = 1, 2$. Based on Eq. (41), the derivative of the enrichment term ψ is

$$\frac{\partial \psi}{\partial x_m} = \frac{\partial \boldsymbol{N}}{\partial x_m}|\boldsymbol{\varphi}| - \mathtt{sgn}(\boldsymbol{N}\boldsymbol{\varphi})\frac{\partial \boldsymbol{N}}{\partial x_m}\boldsymbol{\varphi}. \tag{50}$$

Note that here the chain rule is utilized again and that the sgn-function results from differentiating the abs-function. Similar to Eq. (46), the cell-specific mass matrix has the structure

$$\boldsymbol{M}^c = \begin{bmatrix} \boldsymbol{M}^\mathrm{c}_\mathrm{uu} & \boldsymbol{M}^\mathrm{c}_\mathrm{ua} \\ \boldsymbol{M}^\mathrm{c}_\mathrm{au} & \boldsymbol{M}^\mathrm{c}_\mathrm{aa} \end{bmatrix} = \sum_{k=1}^{n_\mathrm{sc}} \int_{\omega_k} \rho \begin{bmatrix} \boldsymbol{N}_\mathrm{u}^\mathrm{T}\boldsymbol{N}_\mathrm{u} & \boldsymbol{N}_\mathrm{u}^\mathrm{T}\boldsymbol{N}_\mathrm{a} \\ \boldsymbol{N}_\mathrm{a}^\mathrm{T}\boldsymbol{N}_\mathrm{u} & \boldsymbol{N}_\mathrm{a}^\mathrm{T}\boldsymbol{N}_\mathrm{a} \end{bmatrix} \mathrm{d}\boldsymbol{x}. \tag{51}$$

Due to the enrichment, the highest polynomial degree in the integrands of $\boldsymbol{K}^c$ and $\boldsymbol{M}^c$ is $2p_\mathrm{a} = 2(p + p_\psi)$. Thus, for an accurate numerical integration[2], instead of the standard $p + 1$ integration points per direction, $p + p_\psi + 1$ integration points are required. Finally, after assembly, the coupled global equation system without damping reads

$$\begin{bmatrix} \boldsymbol{M}_\mathrm{uu} & \boldsymbol{M}_\mathrm{ua} \\ \boldsymbol{M}_\mathrm{au} & \boldsymbol{M}_\mathrm{aa} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{U}}_\mathrm{u} \\ \ddot{\boldsymbol{U}}_\mathrm{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_\mathrm{uu} & \boldsymbol{K}_\mathrm{ua} \\ \boldsymbol{K}_\mathrm{au} & \boldsymbol{K}_\mathrm{aa} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_\mathrm{u} \\ \boldsymbol{U}_\mathrm{a} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_\mathrm{u} \\ \boldsymbol{F}_\mathrm{a} \end{bmatrix}. \tag{52}$$

5 B-FCM for multi-material problems

In this section, the Boolean integration approach given in Sect. 2 is combined with the local enrichment of the displacement field over cut cell. For simplicity, the focus is kept on cases where cut cells are intersected by nothing else but a single material interface. Thus, the cells contain two material domains Ω_1 and Ω_2, without the presence of any void regions. However, more complex scenarios are possible and constitute a straightforward extension of the proposed approach.

5.1 Intuitive implementation of the cell matrices

In order to apply the features presented in Sect. 3.2, a clear separation of the discontinuous (D) and continuous terms (f) is required. In the case of local enrichment, the discontinuity in Eq. (46) is not only caused by the strongly discontinuous material properties related to $\boldsymbol{C}$, but also by $\boldsymbol{B}_\mathrm{a}$, that is

[2] Using Gaussian quadrature, with n^d integration points, polynomial degrees up to $(2n-1)^d$ can be exactly integrated, where d is the dimensionality of the problem.

derived from the weakly discontinuous enrichment function ψ. Thus, when integrating the enriched stiffness matrix over a given Boolean sub-cell ω_k^{B}, the integration labels defined in Eqs. (20) and (23) lead to

$$\mathcal{L}[i,j]: \begin{cases} \boldsymbol{K}^{\mathrm{c}}_{\mathrm{uu}} &= \int\limits_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \Big(\boldsymbol{C}_i - \boldsymbol{C}_j \Big) \boldsymbol{B}_{\mathrm{u}} \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{ua}} &= \int\limits_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \Big(\boldsymbol{C}_i \boldsymbol{B}_{\mathrm{a}i} - \boldsymbol{C}_j \boldsymbol{B}_{\mathrm{a}j} \Big) \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{au}} &= \int\limits_{\omega_k^{\mathrm{B}}} \Big(\boldsymbol{B}_{\mathrm{a}i}^{\mathrm{T}} \boldsymbol{C}_i - \boldsymbol{B}_{\mathrm{a}j}^{\mathrm{T}} \boldsymbol{C}_j \Big) \boldsymbol{B}_{\mathrm{u}} \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{aa}} &= \int\limits_{\omega_k^{\mathrm{B}}} \Big(\boldsymbol{B}_{\mathrm{a}i}^{\mathrm{T}} \boldsymbol{C}_i \boldsymbol{B}_{\mathrm{a}i} - \boldsymbol{B}_{\mathrm{a}j}^{\mathrm{T}} \boldsymbol{C}_j \boldsymbol{B}_{\mathrm{a}j} \Big) \,\mathrm{d}\boldsymbol{x} \end{cases} \tag{53}$$

and

$$\mathcal{L}[j]: \begin{cases} \boldsymbol{K}^{\mathrm{c}}_{\mathrm{uu}} &= \int\limits_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \Big(\boldsymbol{C} - \boldsymbol{C}_j \Big) \boldsymbol{B}_{\mathrm{u}} \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{ua}} &= \int\limits_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \Big(\boldsymbol{C} \boldsymbol{B}_{\mathrm{a}} - \boldsymbol{C}_j \boldsymbol{B}_{\mathrm{a}j} \Big) \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{au}} &= \int\limits_{\omega_k^{\mathrm{B}}} \Big(\boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} - \boldsymbol{B}_{\mathrm{a}j}^{\mathrm{T}} \boldsymbol{C}_j \Big) \boldsymbol{B}_{\mathrm{u}} \,\mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}^{\mathrm{c}}_{\mathrm{aa}} &= \int\limits_{\omega_k^{\mathrm{B}}} \Big(\boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{a}} - \boldsymbol{B}_{\mathrm{a}j}^{\mathrm{T}} \boldsymbol{C}_j \boldsymbol{B}_{\mathrm{a}j} \Big) \,\mathrm{d}\boldsymbol{x} \end{cases} . \tag{54}$$

Note that in the above equations, the terms are separated such that all weakly/strongly discontinuous terms are within the brackets. As a next step, the extension of these terms to the entire integration domain is required. On the one hand, this is not an issue for $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$, which are often piece-wise constant functions. On the other hand, $\boldsymbol{B}_{\mathrm{a}1}$ and $\boldsymbol{B}_{\mathrm{a}2}$ are polynomial functions and form a piece-wise polynomial $\boldsymbol{B}_{\mathrm{a}}$, such that

$$\boldsymbol{B}_{\mathrm{a}} = \begin{cases} \boldsymbol{B}_{\mathrm{a}1} & \text{in } \Omega_1^k \subset \omega_k^{\mathrm{B}} \\ \boldsymbol{B}_{\mathrm{a}2} & \text{in } \Omega_2^k \subset \omega_k^{\mathrm{B}} \end{cases} . \tag{55}$$

Since the discontinuity in $\boldsymbol{B}_{\mathrm{a}}$ is caused by the enrichment function ψ, the computation of ψ_i and its spatial derivatives are required for calculating $\boldsymbol{B}_{\mathrm{a}i}$. Based on Eqs. (48) and (49), for the sub-matrix of $\boldsymbol{B}_{\mathrm{a}i}$ associated with the lth enriched shape function,

$$\boldsymbol{B}_{\mathrm{a}i}^{(l)} = \begin{bmatrix} \dfrac{\partial \psi_i}{\partial x_1} N_l + \dfrac{\partial N_l}{\partial x_1} \psi_i & 0 \\ 0 & \dfrac{\partial \psi_i}{\partial x_2} N_l + \dfrac{\partial N_l}{\partial x_2} \psi_i \\ \dfrac{\partial \psi_i}{\partial x_2} N_l + \dfrac{\partial N_l}{\partial x_2} \psi_i & \dfrac{\partial \psi_i}{\partial x_1} N_l + \dfrac{\partial N_l}{\partial x_1} \psi_i \end{bmatrix} . \tag{56}$$

When using the modified abs-enrichment approach to construct ψ, (cf. Eq. (41)), the weak discontinuity is introduced by the term $|\boldsymbol{N}\boldsymbol{\varphi}|$. Thus, the enrichment function is divided along the isoline $\boldsymbol{N}\boldsymbol{\varphi} = 0$ into ψ_1 and ψ_2, such that

$$\psi(\boldsymbol{\xi}) = \begin{cases} \psi_1(\boldsymbol{\xi}) & \forall \boldsymbol{\xi},\ \boldsymbol{N}\boldsymbol{\varphi} \le 0 \\ \psi_2(\boldsymbol{\xi}) & \forall \boldsymbol{\xi},\ \boldsymbol{N}\boldsymbol{\varphi} > 0 \end{cases} . \tag{57}$$

Since $|\boldsymbol{N}\boldsymbol{\varphi}| = \boldsymbol{N}\boldsymbol{\varphi}$ for $\boldsymbol{N}\boldsymbol{\varphi} > 0$ and $|\boldsymbol{N}\boldsymbol{\varphi}| = -\boldsymbol{N}\boldsymbol{\varphi}$ for $\boldsymbol{N}\boldsymbol{\varphi} < 0$, ψ_1 and ψ_2 read

$$\psi_1 = \boldsymbol{N}|\boldsymbol{\varphi}| - \boldsymbol{N}\,\boldsymbol{\varphi} \;=\; \boldsymbol{N} \cdot (|\boldsymbol{\varphi}| - \boldsymbol{\varphi}), \tag{58}$$

$$\psi_2 = \boldsymbol{N}|\boldsymbol{\varphi}| + \boldsymbol{N}\,\boldsymbol{\varphi} \;=\; \boldsymbol{N} \cdot (|\boldsymbol{\varphi}| + \boldsymbol{\varphi}). \tag{59}$$

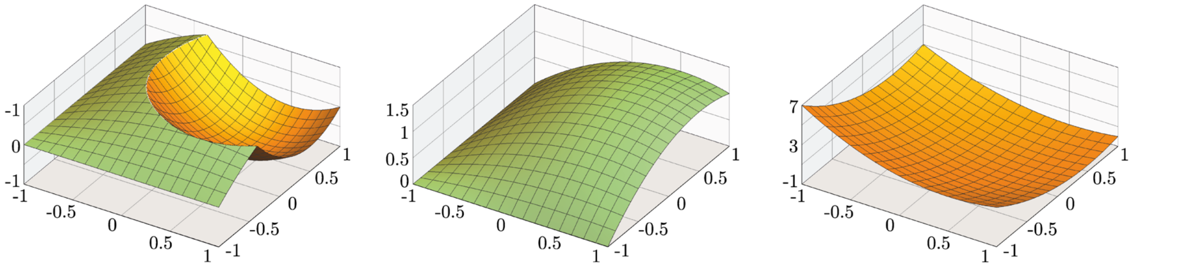
An example for this is given in Fig. 13, where Fig. 13a depicts a specific ψ whose parts over Ω_1^k and Ω_2^k are color-coded by green and orange. The extensions of ψ_1 and ψ_2 to the entire domain $\omega_k^{\mathrm{B}} = \Omega_1^k \cup \Omega_2^k$ are depicted in Figs. 13b and 13c, respectively. For the current example, $\omega_k^{\mathrm{B}} = [-1, 1] \times [-1, 1]$. Furthermore, Ω_1^k and Ω_2^k are the domains exterior and interior to a circle of radius $R = 1$, whose origin is located at $\boldsymbol{x}_{\mathrm{o}} = \{0.5, 0.5\}$.

Finally, since in Eqs. (58) and (59), the vectors $\boldsymbol{N}$ and $\boldsymbol{\varphi}$ contain polynomial and constant entries, respectively, the required partial derivatives of ψ_1 and ψ_2 in Eq. (56) are computed as

$$\frac{\partial \psi_1}{\partial x_m} = \frac{\partial \boldsymbol{N}}{\partial x_m} (|\boldsymbol{\varphi}| - \boldsymbol{\varphi}), \tag{60}$$

$$\frac{\partial \psi_2}{\partial x_m} = \frac{\partial \boldsymbol{N}}{\partial x_m} (|\boldsymbol{\varphi}| + \boldsymbol{\varphi}). \tag{61}$$

The formulation presented in this section sketches an *intuitive* concept of formulating the integrands over the Boolean sub-cells. Although it utilizes a reduced set of integration points, Eqs. (53) and (54) contain additional matrix operations. An example for this can be seen in the computation of $\boldsymbol{K}^{\mathrm{c}}_{\mathrm{aa}}$ by means of Eq. (53), where instead of $\boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{a}}$, $\boldsymbol{B}_{\mathrm{a}i}^{\mathrm{T}} \boldsymbol{C}_i \boldsymbol{B}_{\mathrm{a}i} - \boldsymbol{B}_{\mathrm{a}j}^{\mathrm{T}} \boldsymbol{C}_j \boldsymbol{B}_{\mathrm{a}j}$ is computed, i.e., the number of matrix operations is increased by a factor of two. For avoiding this feature, the next sub-section presents a more compact formulation, reducing the number of unnecessary matrix operations.



(a) ψ (b) ψ_1 (c) ψ_2

Fig. 13 Example for ψ_1 and ψ_2 over the entire element domain. Note that for visualization purposes, that vertical axes are scaled differently

5.2 Compact implementation of the cell matrices

In the current sub-section, the notion of Sect. 5.1 is extended by an improved separation of the continuous and discontinuous terms, such that unnecessary or redundant matrix operations can be avoided and the computational time of the Boolean approach is further reduced. More details concerning the reduction in computational effort are given at the end of this section.

5.2.1 More efficient separation of terms

As a first step, Eq. (48) is rewritten, such that

$$\boldsymbol{B}_{\mathrm{a}}^{(l)} = \begin{bmatrix} \dfrac{\partial \psi}{\partial x_1} & 0 \\ 0 & \dfrac{\partial \psi}{\partial x_2} \\ \dfrac{\partial \psi}{\partial x_2} & \dfrac{\partial \psi}{\partial x_1} \end{bmatrix} N_l + \psi \begin{bmatrix} \dfrac{\partial N_l}{\partial x_1} & 0 \\ 0 & \dfrac{\partial N_l}{\partial x_2} \\ \dfrac{\partial N_l}{\partial x_2} & \dfrac{\partial N_l}{\partial x_1} \end{bmatrix} = N_l \boldsymbol{\Psi} + \psi \boldsymbol{B}_{\mathrm{u}}^{(l)}, \quad (62)$$

where $\boldsymbol{\Psi}$ is a 3×2 matrix containing the spacial derivatives of ψ. The same approach also applies when computing the entire matrix

$$\boldsymbol{B}_{\mathrm{a}} = \boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}. \quad (63)$$

Note that in this case $\boldsymbol{B}_{\mathrm{a}}$ is reproduced by using the terms $\boldsymbol{N}_{\mathrm{u}}$ and $\boldsymbol{B}_{\mathrm{u}}$, which are generated only once for a given sub-cell. Using Eq. (63), let us rewrite the purely enriched term $\boldsymbol{K}_{\mathrm{aa}}^{\mathrm{c}}$ in Eq. (46):

$$\boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{a}} = \left(\boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}\right)^{\mathrm{T}} \boldsymbol{C} \left(\boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}\right) \quad (64)$$

$$= \Big(\boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{\Psi}^{\mathrm{T}} \psi \boldsymbol{C} \boldsymbol{B}_{\mathrm{u}} + \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \psi \boldsymbol{C} \boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi^2 \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{u}}\Big). \quad (65)$$

The above equation can be also expressed in a matrix form as

$$\boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{a}} = \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \cdot \underbrace{\begin{bmatrix} \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{\Psi} & \psi \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C} \\ \psi \boldsymbol{C} \boldsymbol{\Psi} & \psi^2 \boldsymbol{C} \end{bmatrix}}_{\boldsymbol{D}^{\mathrm{aa}}} \cdot \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix}, \quad (66)$$

where $\boldsymbol{D}^{\mathrm{aa}} \in \mathbb{R}^{5\times 5}$ contains all discontinuous properties associated with $\boldsymbol{K}_{\mathrm{aa}}^{\mathrm{c}}$ in a compact form. Since $\boldsymbol{C}$ is symmetric, $(\psi \boldsymbol{C} \boldsymbol{\Psi})^{\mathrm{T}} = \psi \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C}$, and the matrix $\boldsymbol{D}^{\mathrm{aa}}$ is also symmetric

$$\boldsymbol{D}^{\mathrm{aa}} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{\Psi} & \psi (\boldsymbol{C} \boldsymbol{\Psi})^{\mathrm{T}} \\ \psi \boldsymbol{C} \boldsymbol{\Psi} & \psi^2 \boldsymbol{C} \end{bmatrix}. \quad (67)$$

Using the same idea, the following two equations redefine the mixed terms associated with $\boldsymbol{K}_{\mathrm{ua}}^{\mathrm{c}}$ and $\boldsymbol{K}_{\mathrm{au}}^{\mathrm{c}}$:

$$\begin{aligned} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{a}} &= \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \left(\boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}\right) \\ &= \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{\Psi} \boldsymbol{N}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{u}} = \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \underbrace{\begin{bmatrix} \boldsymbol{C} \boldsymbol{\Psi} & \psi \boldsymbol{C} \end{bmatrix}}_{\boldsymbol{D}^{\mathrm{ua}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix}, \quad (68) \\ \boldsymbol{B}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{u}} &= \left(\boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{\Psi}^{\mathrm{T}} + \psi \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}}\right) \boldsymbol{C}\, \boldsymbol{B}_{\mathrm{u}} \\ &= \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C}\, \boldsymbol{B}_{\mathrm{u}} + \psi \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}}\, \boldsymbol{C}\, \boldsymbol{B}_{\mathrm{u}} = \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \underbrace{\begin{bmatrix} \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{C} \\ \psi \boldsymbol{C} \end{bmatrix}}_{\boldsymbol{D}^{\mathrm{au}}} \boldsymbol{B}_{\mathrm{u}}. \quad (69) \end{aligned}$$

Finally, the definition is also applied to $\boldsymbol{K}_{\mathrm{uu}}^{\mathrm{c}}$, where due to $\boldsymbol{C}$ being the only discontinuous term, $\boldsymbol{D}^{\mathrm{uu}} = \boldsymbol{C}$ applies

$$\boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{C} \boldsymbol{B}_{\mathrm{u}} = \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{D}^{\mathrm{uu}} \boldsymbol{B}_{\mathrm{u}}. \quad (70)$$

Collecting all matrices containing the discontinuous terms in Eqs. (67)–(70), the set $\mathbb{D}$ is defined, which is, in the remainder of this section, associated with the enriched $\boldsymbol{K}^c$ of Eq. (46):

$$\mathbb{D} = \{\boldsymbol{D}^{\mathrm{uu}},\ \boldsymbol{D}^{\mathrm{ua}},\ \boldsymbol{D}^{\mathrm{au}},\ \boldsymbol{D}^{\mathrm{aa}}\}. \quad (71)$$

5.2.2 Boolean operations

Let us now combine the formulation from the previous subsection with the Boolean approach. All discontinuous terms in $\mathbb{D}$ are composed by individually continuous parts, such that

$$\mathbb{D} = \begin{cases} \mathbb{D}_1 & \text{for } \boldsymbol{x} \in \Omega_1 \\ \mathbb{D}_2 & \text{for } \boldsymbol{x} \in \Omega_2 \end{cases}, \tag{72}$$

where

$$\mathbb{D}_i = \{\boldsymbol{D}_i^{\mathrm{uu}},\ \boldsymbol{D}_i^{\mathrm{ua}},\ \boldsymbol{D}_i^{\mathrm{au}},\ \boldsymbol{D}_i^{\mathrm{aa}}\}. \tag{73}$$

The above expression only requires evaluating the matrices $\boldsymbol{C}_i$ and the scalars ψ_i and $\partial \psi_i / \partial x_m$, for $i = 1, 2$ and $m = 1, 2$, where the latter ones constitute $\boldsymbol{\Psi}_i$. Using this formulation, Eqs. (53) and (54) can be reformulated as

$$\mathcal{L}[i, j]: \begin{cases} \boldsymbol{K}_{\mathrm{uu}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \left(\boldsymbol{D}_i^{\mathrm{uu}} - \boldsymbol{D}_j^{\mathrm{uu}}\right) \boldsymbol{B}_{\mathrm{u}}\, \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{ua}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \left(\boldsymbol{D}_i^{\mathrm{ua}} - \boldsymbol{D}_j^{\mathrm{ua}}\right) \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{au}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \left(\boldsymbol{D}_i^{\mathrm{au}} - \boldsymbol{D}_j^{\mathrm{au}}\right) \boldsymbol{B}_{\mathrm{u}}\, \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{aa}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \left(\boldsymbol{D}_i^{\mathrm{aa}} - \boldsymbol{D}_j^{\mathrm{aa}}\right) \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x} \end{cases} \tag{74}$$

and

$$\mathcal{L}[j]: \begin{cases} \boldsymbol{K}_{\mathrm{uu}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \left(\boldsymbol{D}^{\mathrm{uu}} - \boldsymbol{D}_j^{\mathrm{uu}}\right) \boldsymbol{B}_{\mathrm{u}}\, \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{ua}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \left(\boldsymbol{D}^{\mathrm{ua}} - \boldsymbol{D}_j^{\mathrm{ua}}\right) \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{au}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \left(\boldsymbol{D}^{\mathrm{au}} - \boldsymbol{D}_j^{\mathrm{au}}\right) \boldsymbol{B}_{\mathrm{u}}\, \mathrm{d}\boldsymbol{x} \\ \boldsymbol{K}_{\mathrm{aa}}^{\mathrm{c}} &= \displaystyle\int_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} & \boldsymbol{B}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix} \left(\boldsymbol{D}^{\mathrm{aa}} - \boldsymbol{D}_j^{\mathrm{aa}}\right) \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}} \\ \boldsymbol{B}_{\mathrm{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x} \end{cases} \tag{75}$$

The application of the same approach to the enriched mass matrix $\boldsymbol{M}^c$ and body load vector $\boldsymbol{F}^c$ is presented in Appendices A.2 and A.3, respectively.

Recall that computing $\boldsymbol{K}_{\mathrm{aa}}^{\mathrm{c}}$ in Eq. (53) involves evaluating the matrices $\boldsymbol{B}_{\mathrm{a}1}$ and $\boldsymbol{B}_{\mathrm{a}2}$, which are in turn used to determine the terms $\boldsymbol{B}_{\mathrm{a}1}^{\mathrm{T}} \boldsymbol{C}_1 \boldsymbol{B}_{\mathrm{a}1}$ and $\boldsymbol{B}_{\mathrm{a}2}^{\mathrm{T}} \boldsymbol{C}_2 \boldsymbol{B}_{\mathrm{a}2}$ as well as their difference (cf. end of Sect. 5.1). In the more compact version presented by Eq. (74), the matrices $\boldsymbol{B}_{\mathrm{a}1}$ and $\boldsymbol{B}_{\mathrm{a}2}$ are not computed explicitly and the Boolean operation is performed on significantly smaller matrices using $\boldsymbol{D}_1^{\mathrm{aa}}$ and $\boldsymbol{D}_2^{\mathrm{aa}}$. Thus, while the intuitive approach is more straightforward in its implementation, the more compact approach presented in this section generally leads to a decreased numerical effort.

6 Numerical examples

6.1 Circular plate with inclusion

In this section, a benchmark problem with a circular inclusion Ω_1 embedded into a circular plate Ω_2 is considered [58,59]. Along the outer boundary of the plate, radial displacement boundary conditions are applied

$$u_{\mathrm{r}}(r) = \begin{cases} \left[\left(1 - \dfrac{b^2}{a^2}\right)\beta + \dfrac{b^2}{a^2}\right] r & 0 \le r \le a \\ \left(r - \dfrac{b^2}{r}\right)\beta + \dfrac{b^2}{r} & a \le r \le b \end{cases}, \tag{76}$$

$$u_{\varphi} = 0, \tag{77}$$

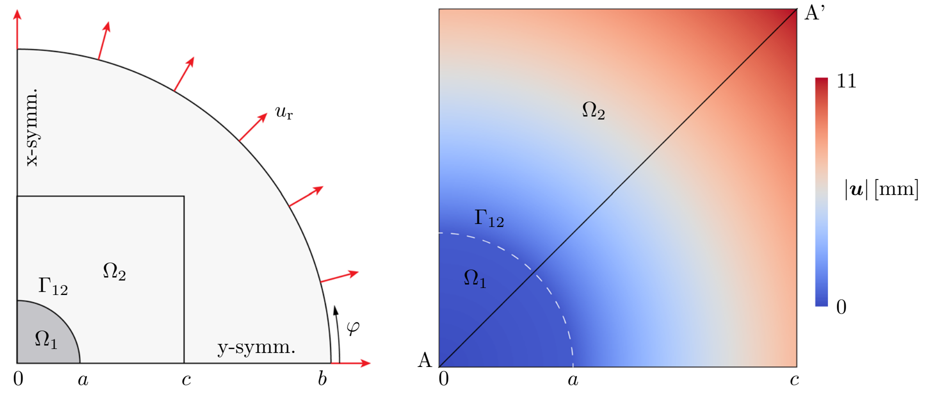
where a and b are the radii of the inclusion and embedding plate, respectively. The parameter β is determined by a and b, together with the Lamé constants of the inclusion $\{r_1, \mu_1\}$ and the matrix $\{r_2, \mu_2\}$

$$\beta = \frac{(r_1 + \mu_1 + \mu_2)b^2}{(r_2 + \mu_2)a^2 + (r_1 + \mu_1)(b^2 - a^2) + \mu_2 b^2}. \tag{78}$$

Instead of simulating the entire domain, a quadratic domain with a side length of c is considered with appropriate non-homogeneous Dirichlet boundary conditions (Fig. 14a). Regarding the geometry and material properties, the following parameters are used: $a = 3\,\mathrm{mm}$, $c = 8\,\mathrm{mm}$, $b = 15\,\mathrm{mm}$, $E_1 = 10{,}000\,\mathrm{MPa}$, $\nu_1 = 0.3$, $E_2 = 0.1E_1$ and $\nu_2 = 0.27$. Furthermore, a plane strain state is assumed. The analytical reference strain energy for the simulated domain is $1/2\,\mathcal{B}(\boldsymbol{u}_{\mathrm{ref}}, \boldsymbol{u}_{\mathrm{ref}}) = 1.109766574913341 \times 10^6$. For simplicity, the quadratic domain is discretized by a single finite cell and spectral shape functions with polynomial degrees from $p = 1$ to $p = 8$ are tested. For ψ in Eq. (41), quadratic shape functions are used, which are sufficient for approximating the level-set function of the circle. For each step of the p-refinement, the QTD is performed with $\mathcal{R} = p + 2$ and $(p + 10)^2$ integration points per sub-cell are used[3].

[3] While less integration points also suffice when reasonable refinement levels are used (cf. Sect. 4.3), according to our studies, utilizing different quadrature orders leads to basically the same reductions in integration points and computation time. Therefore, the savings presented later on are representative for other quadrature orders as well.

Fig. 14 Circular plate with circular inclusion



(a) Problem setup.

(b) Displacement field.

When using the B-FCM approach, the simulation accuracy is maintained when compared to the standard FCM, resulting in basically identical convergence of the relative error in the energy norm

$$e = \sqrt{\frac{|\mathcal{B}(\boldsymbol{u}_{\text{ref}}, \boldsymbol{u}_{\text{ref}}) - \mathcal{B}(\boldsymbol{u}, \boldsymbol{u})|}{\mathcal{B}(\boldsymbol{u}_{\text{ref}}, \boldsymbol{u}_{\text{ref}})}} \cdot 100\% \tag{79}$$

as depicted in Fig. 15a. Here, the B-FCM versions 1 and 2 refer to the different versions of the method introduced in Sects. 5.1 and 5.2, respectively. Additionally, Fig. 15a also depicts the severely deteriorated convergence curve of the case when no enrichment is used. Furthermore, the accuracy of the simulation is also demonstrated in Fig. 15b, where local values are evaluated for $p = 8$ along the cut line A–A', indicated in Fig 14b. The red curve depicts the radial displacements, in which the sharp kink at the material interface is well captured, while the blue curve depicts the non-oscillatory strain energy density $\psi = 1/2\,\boldsymbol{\sigma} : \boldsymbol{\varepsilon}$.

Thus, it can be stated that the B-FCM achieves very accurate results, while requiring significantly less integration points and computational time compared to the standard FCM. Figure 16a compares the wall-clock times spent on the cut cells in an absolute, while Fig. 16b in a relative manner

$$r_t = \left(1 - \frac{t_{\text{B-FCM}}}{t_{\text{FCM}}}\right) \cdot 100\%. \tag{80}$$

Additionally, the reduction of integration points is also depicted, which is fairly constant for the given example within a typical refinement range of $\mathcal{R} = 3...10$: $r_{\text{IP}} = 70{-}76\%$. The reduction of computational time r_t starts at about 30%, however with increasing p and $\mathcal{R}$, it reaches values of $r_t \approx 60\%$ for the B-FCM 1st version and $r_t \approx 76\%$ for the B-FCM 2nd version. The reason for the larger difference between r_{IP} and r_t is due to the fact, that for low values of p and $\mathcal{R}$, the integration time of $\boldsymbol{K}^c$ accounts for a relatively smaller chunk of the overall computational time spent on the cut cell. For higher values of p and $\mathcal{R}$, the numerical integration is more costly, and the effect of reducing the integration points is more prominent. The additional time reduction by the blue curve is due to the more efficient formulation of the matrix operations in case of the 2nd version when compared to the 1st one. Note however, that despite these differences, both versions of the B-FCM (i) operate on the same number of integration points, (ii) reduce the computational time by a significant amount, while (iii) yielding the same accuracy as the standard FCM.

6.2 Cube with spherical inclusion

Next, a 3D test is conducted involving a cube (Ω_2) of side length $a = 4$ m and a spherical inclusion (Ω_1) with a radius of $R = 2$ m, as depicted in Fig. 17a. The problem is formulated using the *method of manufactured solutions* (MoMS) [60,61], i.e., a displacement field $\boldsymbol{u}^*$ is given, for which the strong form of linear elastostatics[4] yields the corresponding body loads $\boldsymbol{b} = -\text{div}(\boldsymbol{\sigma})$. These body loads, together with the appropriate boundary conditions are then given as input for the FCM simulation and the resulting $\boldsymbol{u}$ is compared to $\boldsymbol{u}^*$. Using this approach, the code functionality can be tested based on an analytical reference solution, and no overkill FEM solution is required. For the given example, a radial displacement field $\boldsymbol{u}^* = [u_r^*,\ u_\varphi^*,\ u_\theta^*]$ is chosen in spherical coordinates

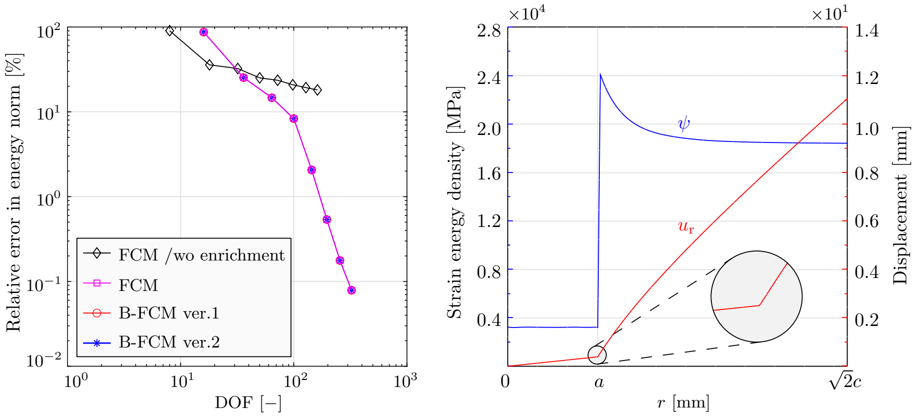
$$u_r^*(r) = c^*(r)\frac{1}{1600}\left(r^5 - 4r^3\right), \tag{81}$$

$$u_\varphi^* = 0, \tag{82}$$

$$u_\theta^* = 0, \tag{83}$$

[4] It is remarked that MoMS can be also applied to more complex problems PDEs and is generally available for all sorts of different physics.

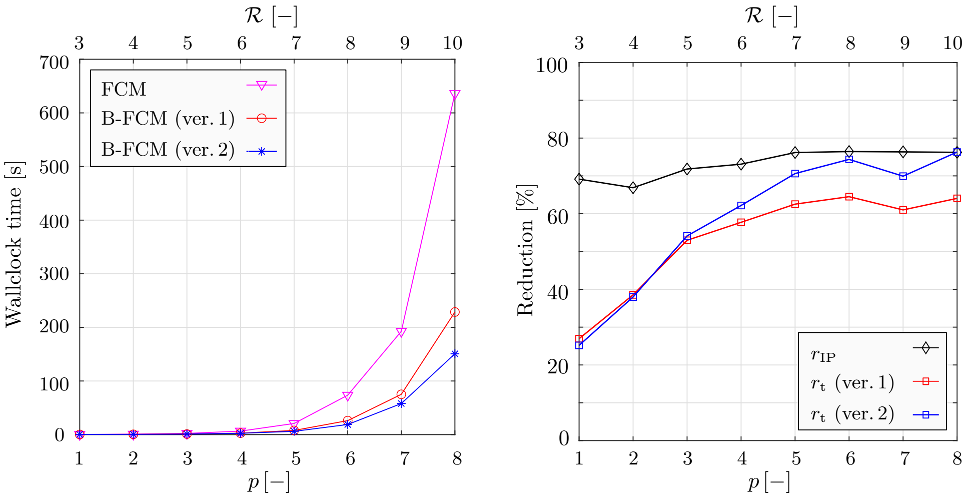
Fig. 15 Global and local results



(a) Global convergence.

(b) Results of ψ and u along the cut line A–A' indicated in Fig. 14b. (B-FCM, $p = 8$).

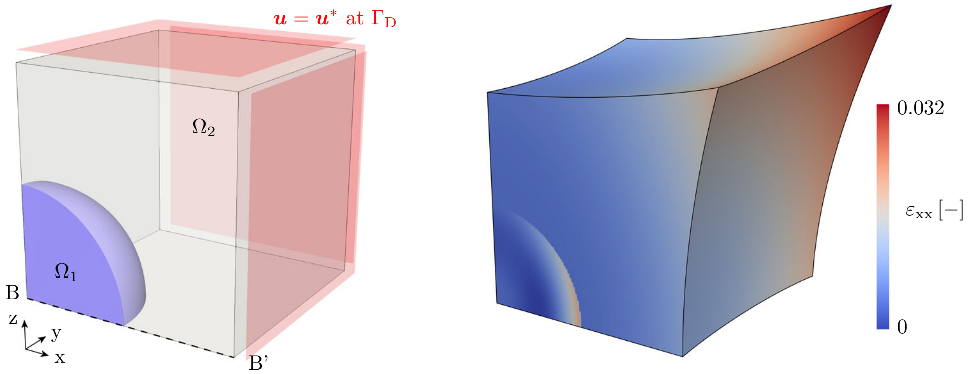
Fig. 16 Reduction of integration points and computational time when using the B-FCM



(a) Computational time for single cut cell using different polynomial degrees.

(b) Reduction of integration points and computational time.

Fig. 17 Simulation of a 3D inclusion problem with a single finite cell



(a) Problem setup

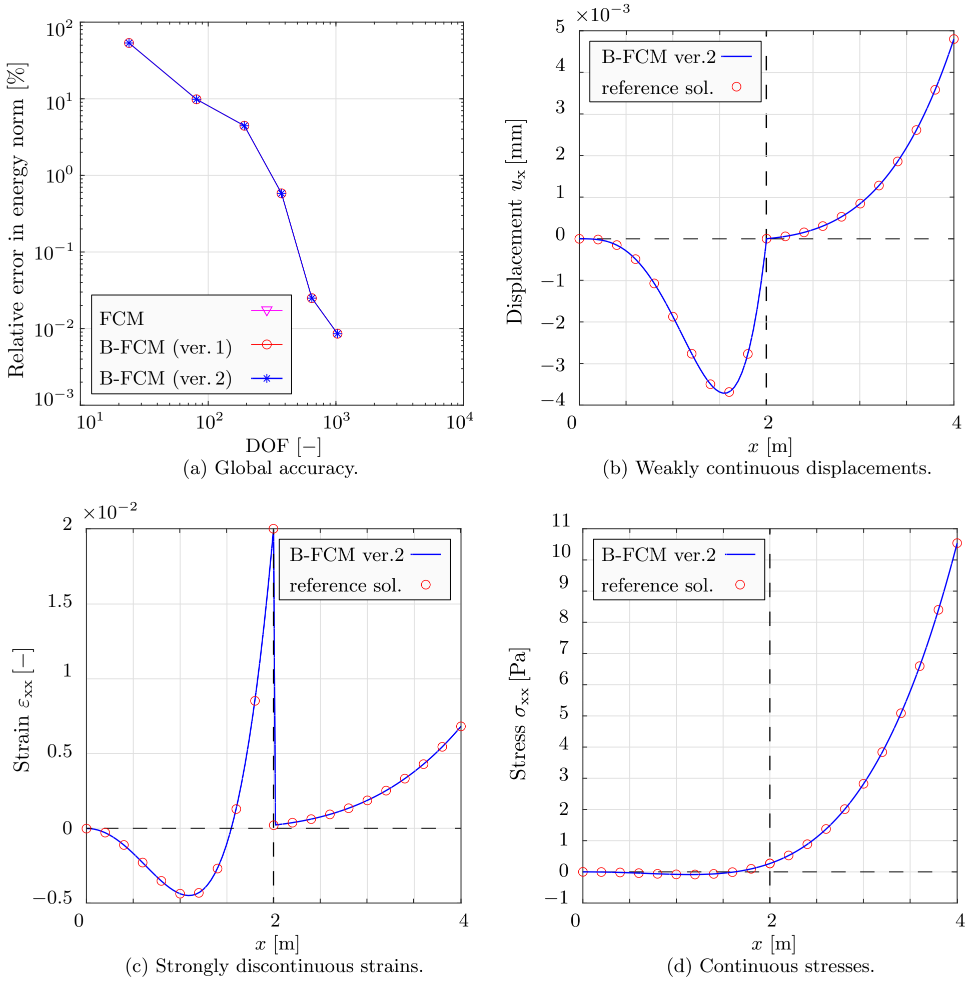
(b) ε_{xy} for the deformed shape $p = 6$.

where

$$c^*(r) = \begin{cases} 1 & \forall r < R \\ c & \forall r > R \end{cases} \quad . \tag{84}$$

Note that for $u_r^*(r = R) = 0$ and for any $c \neq 1$, the manufactured displacement field exhibits a kink at the interface (Fig. 18b). In the current example, $c = 0.01$ is chosen and the deformed shape of the simulation domain is depicted in Fig. 17b. For the material properties, $E_1 = 0.1$ Pa, $E_2 = 10$ Pa and $\nu_1 = \nu_2 = 0.3$ are chosen. The body loads,

Fig. 18 Global results and local field values along the cut line indicated in Fig. 17a



(a) Global accuracy.

(b) Weakly continuous displacements.

(c) Strongly discontinuous strains.

(d) Continuous stresses.

expressed in a spherical coordinate system, read

$$b_{r1} = -\frac{E_1 r(\nu_1 - 1)(7r^2 - 10)}{400(2\nu_1^2 + \nu_1 - 1)}, \tag{85}$$

$$b_{r2} = -\frac{cE_2 r(\nu_2 - 1)(7r^2 - 10)}{400(2\nu_2^2 + \nu_2 - 1)}, \tag{86}$$

for the inclusion and matrix, respectively. Due to our specific choice of c, $E_1 = cE_2$, and thus, $b_{r1} = b_{r2}$. Consequently, tractions are continuous along the interface and jump conditions are naturally fulfilled (see Fig. 18d). In this special case, when following the MoMS approach, no weak boundary conditions along the interface are required and the problem complexity is kept at the necessary level.[5] The body load vectors for the B-FCM are given in Eqs. (100) and (101). Finally, the analytically evaluated reference strain energy for the manufactured solution reads

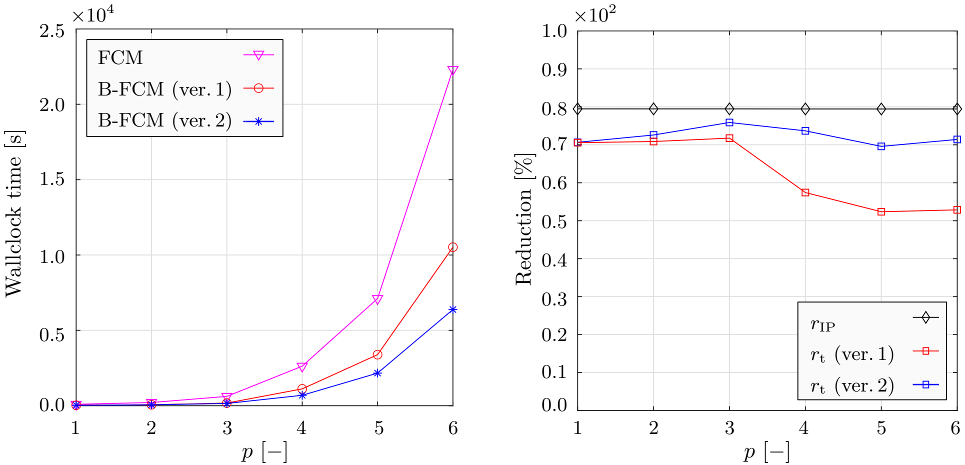
$$\frac{1}{2}\mathcal{B}(\boldsymbol{u}^*, \boldsymbol{u}^*) = 9.664790469963024 \cdot 10^{-2}\,\mathrm{J}. \tag{87}$$

For the current example, the simulation domain is discretized by a single finite cell only and the polynomial degrees $p = 1 \ldots 6$ are tested. Furthermore, a constant $\mathcal{R} = 8$ is chosen for all polynomial degrees and $(p + 10)^3$ integration points are used per sub-cell. The global simulation accuracy measured by the relative error in the energy norm, is depicted Fig. 18a, where similar to the 2D case, the B-FCM approach yields the same convergence curve for the current 3D problem as the FCM. Furthermore, for the B-FCM simulation with $p = 6$, Fig. 18b–d depict local quantities of the solution, such as the radial displacement, strain and stress fields along the cut line B–B’ indicated in Fig. 17a. For all cases, the analytical reference solutions are also shown, demonstrating the high accuracy of B-FCM, while using a single finite cell only.

Analogous to Fig. 16 of the previous example, Fig. 19 depicts the absolute computational times as well as the reduc-

[5] It is noted that in a forthcoming publication, the application of MoMS to embedded domain methods is studied in detail. To this end, special considerations needed in a embedded domain framework are comprehensively discussed and algorithms to make MoMS available for verification purposes are presented.

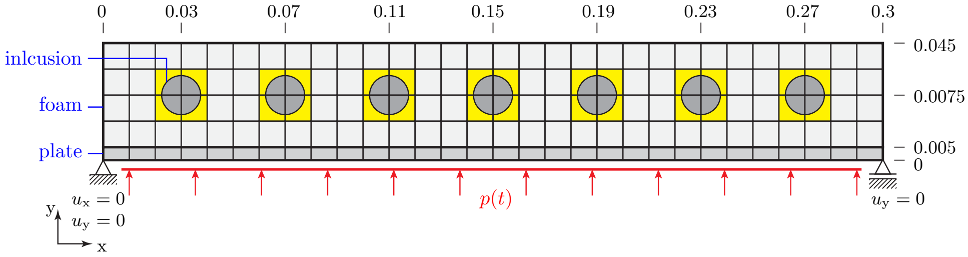
Fig. 19 Numerical efforts required for problem given in Fig. 17a using the FCM and B-FCM approaches



(a) Computational time for different values of polynomial degrees.

(b) Reduction of computational time.

Fig. 20 Problem setup for an acoustic meta-material discretized by finite cells. All dimensions are given in [m]



tion of those (r_{t}) and of the integration points (r_{IP}). Due to the constant refinement level, the reduction of integration points is basically constant, $r_{\mathrm{IP}} \approx 80\%$. However, since (i) generating $\boldsymbol{K}^c$ in 3D is computationally more intensive, and additionally, (ii) $\boldsymbol{F}^c_{\mathrm{body}}$ also has to be computed, the difference between the presented B-FCM versions is more pronounced. For the 2nd version (blue curve), r_{t} is about 70% for all values of p, while for the 1st version (red curve), the unnecessary matrix operations in $\boldsymbol{K}^c$ lead to additional computational overhead for higher values of p, rendering the method less effective. Nonetheless, even in this case, 50–70% of the computational time is saved.

Table 1 Material properties for the plate with meta-material depicted in Fig. 20

Domain	E [MPa]	ν [−]	ρ [kg/m^3]	η [−]
Plate	70,000	0.3	2700	0.05
Foam	5	0.3	5	0.01
Inclusions	500	0.3	500	0.00

6.3 Acoustic meta-material

Finally, the proposed multi-material B-FCM approach is applied to the analysis of an acoustic meta-material placed on a metal plate, where the meta-material consists of a foam matrix and several inclusions (Fig. 20). The simulation of the surrounding acoustic field is omitted at this point for simplicity, and the focus is kept on the vibrating structure. For testing the proposed integration scheme, a harmonic analysis is conducted on the structure using the material properties given in Table 1.

The simulation domain is discretized by 5×30 finite cells, among which 28 are cut by the embedded material interfaces (yellow cells). The plate is subjected to a harmonic excitation along its bottom edge

$$p(t) = \hat{p}\,\mathrm{e}^{\mathrm{i}\Omega t}, \tag{88}$$

where $\hat{p}$ is the pressure amplitude, i the imaginary unit and $\Omega = 2\pi f$ the angular excitation frequency. The system answer $\hat{\boldsymbol{U}}$ is computed in the frequency domain by

$$(-\Omega^2 \boldsymbol{M} + \mathrm{i}\Omega \boldsymbol{D} + \boldsymbol{K})\hat{\boldsymbol{U}} = \hat{\boldsymbol{F}}, \tag{89}$$

where a stiffness proportional damping is used

$$\boldsymbol{D} = \frac{\eta \boldsymbol{K}}{\Omega}. \tag{90}$$

Note that since the damping factor η is also yet another discontinuous material property (Table 1), Eq. (90), cannot be applied directly to generate the global stiffness matrix $\boldsymbol{K}$. Instead, it is realized on the sub-cell level

Table 2 Material properties for the plate with meta-material depicted in Fig. 20

Method	Number of... [−]		Time required for... [s]		
	SC	IP	Create LIM	Cut Cell	Assembly
FCM	3871	1,548,400	0.54	418.40	433.72
B-FCM	2639	401,072	0.68	146.48	147.15
Reduction [%]	31.82	74.10	−25.93	65.00	66.07

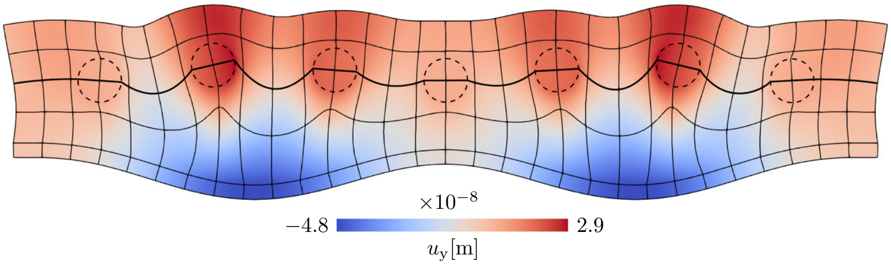


Fig. 21 Dynamic response of the system to a mono-frequent harmonic excitation with an excitation frequency of $f = 2710$ Hz. The domain is discretized by 150 unfitted cells with Boolean integration approach. The dashed lines indicate the positions of the embedded inclusions

$$\boldsymbol{D}^c = \frac{1}{\Omega} \sum_{k=1}^{n_{\mathrm{sc}}} \sum_{i=1}^{n_{\mathrm{d}}^k} \eta_i \boldsymbol{K}^c_{k,i}, \tag{91}$$

where $\boldsymbol{K}^c_{k,i}$ is the contribution to the stiffness matrix $\boldsymbol{K}^c$ over $\Omega_i^k \subset \omega_k^{\mathrm{B}}$, i.e., the ith material domain in the kth sub-cell, and η_i is the damping factor in that material region. Furthermore, n_{d}^k is the number of sub-domains in ω_k^{B}. Note that Eq. (91) holds both for the FCM and B-FCM approaches, i.e., regardless of how $\boldsymbol{K}^c$ is computed.

Table 2 compares the FCM and B-FCM approaches in terms of the number of sub-cells (SC) and integration points (IP) generated in the cut cells, as well as the computation times required for creating the local integration mesh (LIM), the cut cell procedure, assembly, and solver. In both cases, $p = 5$ and $\mathcal{R} = 5$ is used with $(p+10)^2$ integration points in each sub-cell. In the last row, the reductions of the different values are computed. Similar to the previous examples, a high reduction of integration points is obtained, $r_{\mathrm{IP}} = 74.10\%$. While creating the LIM requires additional time in the B-FCM, its cost is negligible to the time spent on numerical integration. Thus, the overall time spent on the cut cells, that includes other miscellaneous tasks besides creating the LIM and integration, is reduced by 65%. Since the cut cells represent the computationally most intensive part of the assembly procedure, the assembly time is reduced by a similar amount. The deformed shape of the vibrating meta-material computed with the B-FCM is depicted in Fig. 21.

7 Conclusion

In this contribution, the B-FCM formulation for porous media (Sect. 1.3) is extended to multi-material problems for efficient simulation of complex structures with material interfaces in the FCM framework. The numerical integration in the proposed B-FCM approach is based on a quadtree-/octree-decomposition (QTD/OTD) of the cut cells (Sect. 3.1), as it is often done in the FCM. However, in the B-FCM implementation, sub-cells are equipped with special labels (Sect. 3.2), whose purpose is twofold:

1. They steer the decomposition, such that it generally results in an overlapping structure of integration domains (Figs. 6 and 10), leading to significantly less sub-cells and integration points.
2. The labels hold specific information regarding the form of the integrand over the individual sub-cells, such that Boolean operations can be used to successively compute the integral value over the overlapping sub-cells.

In our implementation, the C^0-continuous displacement field along the material interfaces is captured by a local enrichment approach. The multi-material B-FCM requires certain changes to the integrand such that the Boolean nature of the sub-cells can be exploited. While these changes are straightforward for discontinuities in the material properties, further steps are needed for discontinuities caused by the chosen enrichment function, as derived in Eqs. (58)–(61).

Numerical examples are conducted both in 2D and 3D, and the multi-material B-FCM concept is applied to the computation of stiffness and mass matrices as well as to the body load vector. The results show that compared to the conventional QTD/OTD-based FCM, its Boolean version requires up to 80% less integration points for reasonable refinement levels. Furthermore, despite a significant reduction in the computational effort, the global and local accuracy of the simulation is not altered. These statements apply to both presented B-FCM formulations in this paper. While the 1st version enables a more intuitive understanding of the integration over the

Boolean sub-cells (Sect. 5.1), the 2nd version (Sect. 5.2) is based on a more efficient separation of continuous and discontinuous terms and thus, reduces the number of redundant matrix multiplications. For simpler problems, the difference in the two approaches is less prominent, as they reduce the time spent on the cut cells by ∼55–65% and ∼55–75%, respectively (Fig. 16). However, for problems with larger cell matrices, e.g., in case of 3D examples or when using high polynomial orders, the 2nd version is superior, leading to ∼70% time reduction, while the 1st version achieves roughly 50%.

In conclusion, while the implementation of the B-FCM requires slight modification of the decomposition scheme and integrands over the sub-cells, it reduces the computational effort of the QTD/OTD-based integration schemes by a significant amount, while enjoying the same level of robustness in combination with no loss in accuracy. If the simplest implementation of the B-FCM is required (e.g., for testing purposes), we recommend its 1st version, and if the maximum efficiency is desired, its 2nd version should be exploited.

Funding Open Access funding enabled and organized by Projekt DEAL.

A Appendix

A.1 BFCM for porous media

The proposed multi-material B-FCM approach can reproduce the special case of the B-FCM for porous media [1], where only two domain types are present, Ω_{phys} and Ω_{fict}, and the discontinuity is caused by the pieace-wise constant indicator function α given in Eq. (1). In their paper, Abedian and Düster discuss four different cases for the Boolean sub-cells when integrating the discontinuous function αf over the cut cell:

1. Integrate f over the given sub-cell while neglecting the discontinuity (left part of Fig. 2c)
2. Integrate $-f$ over the given sub-cell while neglecting the discontinuity
3. Integrate f over the physical part of the given sub-cell
4. Integrate $-f$ over the fictitious part of the given sub-cell (right part of Fig. 2c)

Following the notion used in this paper, in the porous case, $\Omega_1 = \Omega_{\text{phys}}$, $\Omega_2 = \Omega_{\text{fict}}$, $D_1 = 1$ and $D_2 = 0$. Furthermore, the Boolean sub-cells contain parts of Ω_1 and Ω_2, such that $\omega_k^{\text{B}} = \Omega_1^k \cup \Omega_2^k$. Based on these, the above four integrals are reproduced by using the labels

$$\mathcal{L}[1,2]\ :\ \int_{\omega_k^{\text{B}}} (D_1 - D_2) f \,\mathrm{d}\boldsymbol{x} \ =\ \int_{\omega_k^{\text{B}}} f \,\mathrm{d}\boldsymbol{x}\,, \tag{92}$$

$$\mathcal{L}[2,1]\ :\ \int_{\omega_k^{\text{B}}} (D_2 - D_1) f \,\mathrm{d}\boldsymbol{x} \ =\ -\int_{\omega_k^{\text{B}}} f \,\mathrm{d}\boldsymbol{x}\,, \tag{93}$$

$$\mathcal{L}[2]\ :\ \int_{\omega_k^{\text{B}}} (D - D_2) f \,\mathrm{d}\boldsymbol{x} \ =\ \int_{\Omega_1^k} f \,\mathrm{d}\boldsymbol{x}\,, \tag{94}$$

$$\mathcal{L}[1]\ :\ \int_{\omega_k^{\text{B}}} (D - D_1) f \,\mathrm{d}\boldsymbol{x} \ =\ -\int_{\Omega_2^k} f \,\mathrm{d}\boldsymbol{x}\,, \tag{95}$$

where in Eqs. (94) and (95), the vanishing integrand property related to the definition of Eq. (24) is utilized.

A.2 Mass matrix

With the relation for $\boldsymbol{N}_{\text{a}}$ given in Eq. (40), the enriched mass matrix over the sub-cell ω_k^{B} in the cell c is given by

$$\begin{aligned}\boldsymbol{M}^c &= \int\limits_{\omega_k^{\text{B}}} \begin{bmatrix} \rho \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & \rho \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{a}} \\ \rho \boldsymbol{N}_{\text{a}}^{\text{T}} \boldsymbol{N}_{\text{u}} & \rho \boldsymbol{N}_{\text{a}}^{\text{T}} \boldsymbol{N}_{\text{a}} \end{bmatrix} \mathrm{d}\boldsymbol{x} \\ &= \int\limits_{\omega_k^{\text{B}}} \begin{bmatrix} \rho \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & \rho \psi \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \\ \rho \psi \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & \rho \psi^2 \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x}\,.\end{aligned} \tag{96}$$

Using the proposed integration approach, $\boldsymbol{M}^c$ over the Boolean sub-cells is computed by

$$\mathcal{L}[i,j] : \boldsymbol{M}^c = \int\limits_{\omega_k^{\text{B}}} \begin{bmatrix} (\rho_i - \rho_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & (\rho_i \psi_i - \rho_j \psi_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \\ (\rho_i \psi_i - \rho_j \psi_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & (\rho_i \psi_i^2 - \rho_j \psi_j^2) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x}, \tag{97}$$

$$\mathcal{L}[j] : \boldsymbol{M}^c = \int\limits_{\omega_k^{\text{B}}} \begin{bmatrix} (\rho - \rho_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & (\rho \psi - \rho_j \psi_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \\ (\rho \psi - \rho_j \psi_j) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} & (\rho \psi^2 - \rho_j \psi_j^2) \boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}} \end{bmatrix} \mathrm{d}\boldsymbol{x}, \tag{98}$$

for the different label types. In the brackets, the discontinuity is taken into account, which in case of bi-material problems, requires the evaluation of the scalar terms ρ_1, ρ_2, ψ_1, ψ_2, ψ_1^2 and ψ_2^2. Note that the product $\boldsymbol{N}_{\text{u}}^{\text{T}} \boldsymbol{N}_{\text{u}}$ must be computed only once for the given sub-cell.

A.3 Body load vector

The enriched body load vector over the sub-cell ω_k^{B} is computed by

$$\boldsymbol{F}^c_{\mathrm{body}} = \begin{bmatrix} \boldsymbol{F}^c_{\mathrm{u}} \\ \boldsymbol{F}^c_{\mathrm{a}} \end{bmatrix} = \int\limits_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \boldsymbol{b} \\ \boldsymbol{N}_{\mathrm{a}}^{\mathrm{T}} \boldsymbol{b} \end{bmatrix} \mathrm{d}\boldsymbol{x}. \tag{99}$$

Using Boolean sub-cells for the numerical integration and the relation for $\boldsymbol{N}_{\mathrm{a}}$ given in Eq. (40), F^c_{body} over ω_k^{B} reads for the two different label types

$$\mathcal{L}[i,j] : \boldsymbol{F}^c_{\mathrm{body}} = \int\limits_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \big(\boldsymbol{b}_i - \boldsymbol{b}_j\big) \\ \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \big(\psi_i \boldsymbol{b}_i - \psi_j \boldsymbol{b}_j\big) \end{bmatrix} \mathrm{d}\boldsymbol{x}, \tag{100}$$

$$\mathcal{L}[j] : \boldsymbol{F}^c_{\mathrm{body}} = \int\limits_{\omega_k^{\mathrm{B}}} \begin{bmatrix} \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \big(\boldsymbol{b} - \boldsymbol{b}_j\big) \\ \boldsymbol{N}_{\mathrm{u}}^{\mathrm{T}} \big(\psi \boldsymbol{b} - \psi_j \boldsymbol{b}_j\big) \end{bmatrix} \mathrm{d}\boldsymbol{x}. \tag{101}$$

References

1. Abedian A, Düster A (2017) An extension of the finite cell method using Boolean operations. Comput Mech 59(5):877–886
2. Szabó B, Düster A, Rank E (2004) The *p*-version of the finite element method, chapter 5. Encyclopedia of Computational Mechanics
3. Düster A, Rank E, Szabó B (2017) The *p*-version of the finite element and finite cell methods. Encyclopedia of Computational Mechanics, pp 1–35
4. Parvizian 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
5. Düster A, Parvizian J, Yang Z, Rank E (2008) The finite cell method for three-dimensional problems of solid mechanics. Comput Methods Appl Mech Eng 197:3768–3782
6. Joulaian M, Düster A (2013) Local enrichment of the finite cell method for problems with material interfaces. Comput Mech 52(4):741–762
7. Joulaian M (2017) The hierarchical finite cell method for problems in structural mechanics. PhD thesis, Hamburg Technical University
8. Ramière I, Angot P, Belliard M (2007) A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. Comput Methods Appl Mech Eng 196(4–6):766–781
9. Ruess M, Schillinger D, Bazilevs Y, Varduhn V, Rank E (2013) Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method. Int J Numer Methods Eng 95(10):811–846
10. Dauge M, Düster A, Rank E (2015) Theoretical and numerical investigation of the finite cell method. J Sci Comput 65(3):1039–1064
11. de Prenter F, Verhoosel CV, van Zwieten GJ, van Brummelen EH (2017) Condition number analysis and preconditioning of the finite cell method. Comput Methods Appl Mech Eng 316:297–327
12. de Prenter F, Verhoosel CV, van Brummelen EH (2019) Preconditioning immersed isogeometric finite element methods with application to flow problems. Comput Methods Appl Mech Eng 348:604–631
13. Fries TP, Belytschko T (2010) The extended/generalized finite element method: an overview of the method and its applications. Int J Numer Methods Eng 84:253–304
14. Moës N, Cloirec M, Cartraud P, Remacle J-F (2003) A computational approach to handle complex microstructure geometries. Comput Methods Appl Mech Eng 192(28–30):3163–3177
15. Chin EB, Sukumar N (2019) Modeling curved interfaces without element-partitioning in the extended finite element method. Int J Numer Methods Eng 120(5):607–649
16. Abedian A, Parvizian J, Düster A, Rank E (2013) The finite cell method for the J_2 flow theory of plasticity. Finite Elem Anal Des 69:37–47
17. Duczek S, Duvigneau F, Gabbert U (2016) The finite cell method for tetrahedral meshes. Finite Elem Anal Des 121:18–32
18. Xu F, Schillinger D, Kamensky D, Varduhn V, Wang C, Hsu M-C (2016) The tetrahedral finite cell method for fluids: immersogeometric analysis of turbulent flow around complex geometries. Comput Fluids 141:135–154
19. Nadal E, Ródenas JJ, Albelda J, Tur M, Tarancón JE, Fuenmayor FJ (2013) Efficient finite element methodology based on cartesian grids: application to structural shape optimization. Abstr Appl Anal 1–19:2013
20. Legrain G, Moës N (2018) Adaptive anisotropic integration scheme for high-order fictitious domain methods: application to thin structures. Int J Numer Methods Eng 114(8):882–904
21. Cheng KW, Fries T-P (2009) Higher-order XFEM for curved strong and weak discontinuities. Int J Numer Methods Eng 82:564–590
22. Fries T-P, Omerović S (2015) Higher-order accurate integration of implicit geometries. Int J Numer Methods Eng 106(5):323–371
23. Strouboulis T, Copps K, Babuška I (2001) The generalized finite element method. Comput Methods Appl Mech Eng 190(32–33):4081–4193
24. Kudela L, Zander N, Bog T, Kollmannsberger S, Rank E (2015) Efficient and accurate numerical quadrature for immersed boundary methods. Adv Model Simul Eng Sci 2(10):1–22
25. Kudela L, Zander N, Kollmannsberger S, Rank E (2016) Smart octrees: Accurately integrating discontinuous functions in 3D. Comput Methods Appl Mech Eng 306:406–426
26. Mousavi SE, Xiao H, Sukumar N (2009) Generalized Gaussian quadrature rules on arbitrary polygons. Int J Numer Methods Eng 82:99–113
27. Xiao H, Gimbutas Z (2010) A numerical algorithm for the construction of efficient quadrature rules in two and higher dimensions. Comput Math Appl 59(2):663–676
28. Müller B, Kummer F, Oberlack M (2013) Highly accurate surface and volume integration on implicit domains by means of moment-fitting. Int J Numer Methods Eng 96(8):512–528
29. Joulaian M, Hubrich S, Düster A (2016) Numerical integration of discontinuities on arbitrary domains based on moment fitting. Comput Mech 57(6):979–999
30. Mousavi SE, Sukumar N (2010) Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. Comput Mech 47(5):535–554
31. Sudhakar Y, Wall WA (2013) Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods. Comput Methods Appl Mech Eng 258:39–54
32. 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
33. Legrain G (2021) Non-negative moment fitting quadrature rules for fictitious domain methods. Comput Math Appl 99:270–291
34. Garhuom W, Düster A (2022) Non-negative moment fitting quadrature for cut finite elements and cells undergoing large deformations. Comput Mech 70:1059–181

35. Düster A, Allix O (2019) Selective enrichment of moment fitting and application to cut finite elements and cells. Comput Mech 65(2):429–450
36. Ventura G (2006) On the elimination of quadrature subcells for discontinuous functions in the eXtended finite-element method. Int J Numer Methods Eng 66(5):761–795
37. Ventura G, Benvenuti E (2014) Equivalent polynomials for quadrature in heaviside function enriched elements. Int J Numer Methods Eng 102(3–4):688–710
38. 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
39. Dasgupta G (2003) Integration within polygonal finite elements. J Aerosp Eng 16(1):9–18
40. Gao X-W (2002) The radial integration method for evaluation of domain integrals with boundary-only discretization. Eng Anal Boundary Elem 26(10):905–916
41. Sudhakar Y, Moitinho de Almeida JP, Wall WA (2014) An accurate, robust, and easy-to-implement method for integration over arbitrary polyhedra: application to embedded interface methods. J Comput Phys 273:393–415
42. Duczek S, Gabbert U (2015) Efficient integration method for fictitious domain approaches. Comput Mech 56(4):725–738
43. Schillinger D, Ruess M (2014) 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(3):391–455
44. Düster A, Sehlhorst H-G, Rank E (2012) Numerical homogenization of heterogeneous and cellular materials utilizing the finite cell method. Comput Mech 50(4):413–431
45. Zander N, Kollmannsberger S, Ruess M, Yosibash Z, Rank E (2012) The finite cell method for linear thermoelasticity. Comput Math Appl 64(11):3527–3541
46. Taghipour A, Parvizian J, Heinze S, Düster A (2018) The finite cell method for nearly incompressible finite strain plasticity problems with complex geometries. Comput Math Appl 75(9):3298–3316
47. Kudela L, Kollmannsberger S, Almac U, Rank E (2020) Direct structural analysis of domains defined by point clouds. Comput Methods Appl Mech Eng 358:112581
48. Wassermann B, Korshunova N, Kollmannsberger S, Rank E, Elber G (2020) Finite cell method for functionally graded materials based on V-models and homogenized microstructures. Adv Model Simul Eng Sci 7(49):1–33
49. Kollmannsberger S, D'Angella D, Rank E, Garhuom W, Hubrich S, Düster A, Di Stolfo P, Schröder A (2019) Spline- and hp-basis functions of higher differentiability in the finite cell method. GAMM-Mitteilungen 43:e202000004
50. Zakian P, Nadi M, Tohidi M (2021) Finite cell method for detection of flaws in plate structures using dynamic responses. Structures 34:327–338
51. 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(1):1–42
52. Petö M, Duvigneau F, Juhre D, Eisenträger S (2020) Enhanced numerical integration scheme based on image compression techniques: application to rational polygonal interpolants. Arch Appl Mech 91(2):753–775
53. 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. Comput Methods Appl Mech Eng 401:115565
54. Gopalakrishnan S, Chakraborty A, Mahapatra DR (2008) Spectral finite element method. Springer, Berlin
55. Duczek S, Liefold S, Gabbert U (2014) The finite and spectral cell methods for smart structure applications: transient analysis. Acta Mech 226(3):845–869
56. Nicoli S, Agathos K, Chatzi E (2022) Moment fitted cut spectral elements for explicit analysis of guided wave propagation. Comput Methods Appl Mech Eng 398:115140
57. Khoei AR, Vahab M, Ehsani H, Rafieerad M (2015) X-FEM modeling of large plasticity deformation; a convergence study on various blending strategies for weak discontinuities. Eur J Comput Mech 24(3):79–106
58. Sukumar N, Chopp DL, Moës N, Belytschko T (2001) Modeling holes and inclusions by level sets in the extended finite-element method. Comput Methods Appl Mech Eng 190(46–47):6183–6200
59. Schröder J, Wick T, Reese S, Wriggers P, Müller R, Kollmannsberger S, Kästner M, Schwarz A, Igelbüscher M, Viebahn N, Bayat HR, Wulfinghoff S, Mang K, Rank E, Bog T, D'Angella D, Elhaddad M, Hennig P, Düster A, Garhuom W, Hubrich S, Walloth M, Wollner W, Kuhn C, Heister T (2020) A selection of benchmark problems in solid mechanics and applied mathematics. Arch Comput Methods Eng 28:713–751
60. Salari K, Knupp P (2000) Code verification by the method of manufactured solutions. Technical report, Sandia National Laboratories
61. Roache PJ (2001) Code verification by the method of manufactured solutions. J Fluids Eng 124(1):4–10