

Chapter 4

An Implementational Framework for the DG-PEM

The implementation of any partitioned element method must address two primary challenges: the subdivision of an element into cells, and the solution of shape function-specific boundary value problems by way of the chosen PEM formulation. The first of these two tasks may be accomplished by various means. However, careful attention must be paid to the stability requirements of the PEM, such that the resulting quadrature cell partition yields stable approximations to harmonic shape functions, and an equally stable integration rule for the element. Specifying such a discretization is not trivial for arbitrary shapes. For this reason, we will confine our attention to strictly star-convex elements, for which we propose a relatively simple (edge-based) element partitioning algorithm.

The present implementational framework is directed at obtaining DG-PEM approximations to harmonic shape functions on arbitrary polytopes – i.e. solving (3.34) on an appropriately defined partition of the element. A discussion of the pertinent data structures and solution methods is provided, and a number of concluding remarks are given with regard to numerical robustness of the proposed methodology.

4.1 Arbitrary Polytopal Meshes

Traditional finite element methods typically permit the elements to only take the form of certain canonical shapes with fixed topology (most commonly tetrahedra or hexahedra).

This yields several benefits with regard to the data and storage requirements necessary to represent a given element in a computational setting. Frequently, it suffices to describe the geometry of a canonical shape (such as a hexahedron) by providing a list of the element's nodal coordinates, along with an ordered sequence of the element's nodal IDs, which fully determines its resulting topology according to some conventional node numbering scheme (refer to Figure 4.1 as a representative example.)

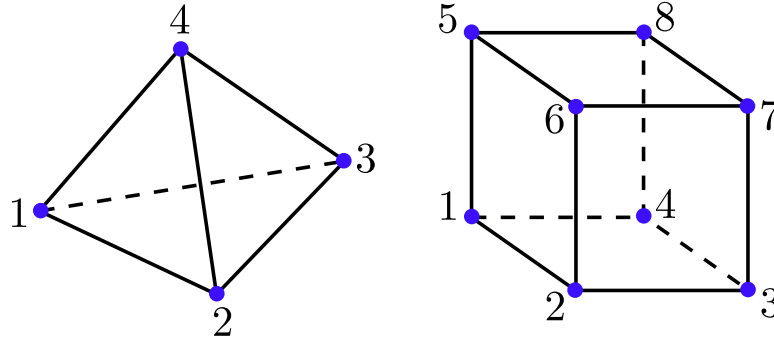


Figure 4.1: Canonical node numbering schemes for a linear tetrahedral element (left) and a linear hexahedral element (right). Each of these standard orderings induce corresponding nodal orderings for each triangular or quadrilateral face of the element, as well.

Polytopal elements with arbitrary topology (with a variable number of nodes, edges, and faces) cannot be represented in this fashion. As a consequence, more descriptive data structures are necessary to fully determine a polytopal element's nodal connectivity. A particular scheme to represent an arbitrary polyhedron within a finite element mesh is described in the following section.

Geometric Data Structures for Arbitrary Polytopes

There are multiple ways in which the geometry of a given polyhedral element Ω may be represented within a finite element code. Ideally, however, the chosen data structure should be made as compact as possible, for the sake of minimizing the storage requirements of a single element. This section describes a few basic data structures for storing arbitrary polygonal and polyhedral shapes within an unstructured finite element mesh.

The geometric data used to describe a typical finite element mesh consists of the following:

- A list of the spatial coordinates $\{\mathbf{X}_a\}_{a=1}^{N_V^{\mathcal{B}_0}}$ for all nodes in the mesh. The sub-index $a \in 1, \dots, N_V^{\mathcal{B}_0}$ induces a *global node ID* ascribed to each node.
- A list of all elements $\{\Omega_e\}_{e=1}^{N_\Omega^{\mathcal{B}_0}}$ where $\Omega_e \subset \mathcal{B}_0$. The sub-index $e \in 1, \dots, N_\Omega^{\mathcal{B}_0}$ induces an associated *element ID*.
- A list of all faces $\{F_b\}_{b=1}^{N_F^{\Gamma_0^N}}$ where $F_b \subset \Gamma_0^N$ to which traction boundary conditions are assigned. Likewise, the sub-index $b \in 1, \dots, N_F^{\Gamma_0^N}$ induces a *boundary face ID*.

In a finite element mesh consisting of arbitrary polyhedral elements, it is suggested that each element Ω_e be represented by the following data:

- A list of the global node IDs $\{a_i\}_{i=1}^{N_V^{\Omega_e}}$ comprising the set of nodes belonging to Ω_e . The sub-index $i \in 1, \dots, N_V^{\Omega_e}$ induces a *local node ID*, particular to the element Ω_e .
- A list of the polygonal faces $\{F_j\}_{j=1}^{N_F^{\Omega_e}}$ which belong to $\partial\Omega_e$; each polygonal face F_j is in turn represented by a cycle of local node IDs denoted $\{n_i\}_{i=1}^{N_V^{F_j}}$, which further determine each face's outward orientation with respect to the element Ω_e .

An illustration of this collection of data for a given polyhedron Ω_e is provided in Figure 4.2.

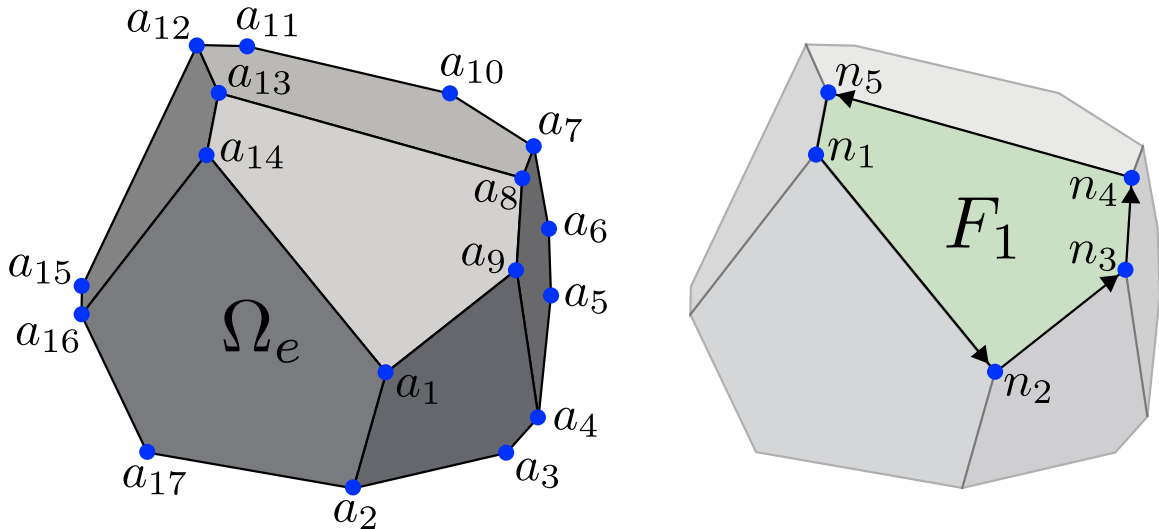


Figure 4.2: Illustration of the data necessary to describe an arbitrary polyhedral element Ω_e . The local node ID ordering for the face F_1 shown would be $\{n_i\}_{i=1}^5 = \{14, 1, 9, 8, 13\}$.

A given polygonal face $F_b \subset \Gamma_0^N$ will similarly be represented by:

- A list of the global node IDs $\{a_i\}_{i=1}^{N_V^{F_b}}$ which belong to F_b .
- A list of the linear edges $\{E_c\}_{c=1}^{N_E^{F_b}}$ which belong to ∂F_b ; each linear edge E_c is in turn represented as an ordered list of local node IDs $\{n_i\}_{i=1}^{N_V^{E_c}}$.

Unlike canonical finite element shapes, the ordering of the element's nodal IDs is effectively arbitrary, and does not induce a topology. Rather, the element's topology is determined by virtue of its polygonal faces, and their respective (conventionally clockwise) local node orderings. Consequently, each edge of a given polyhedron is defined implicitly as the intersection of two adjacent faces' ordered node sets. Such a scheme may be easily generalized to accommodate serendipity elements containing additional nodes along element edges, as illustrated in Figure 4.3.

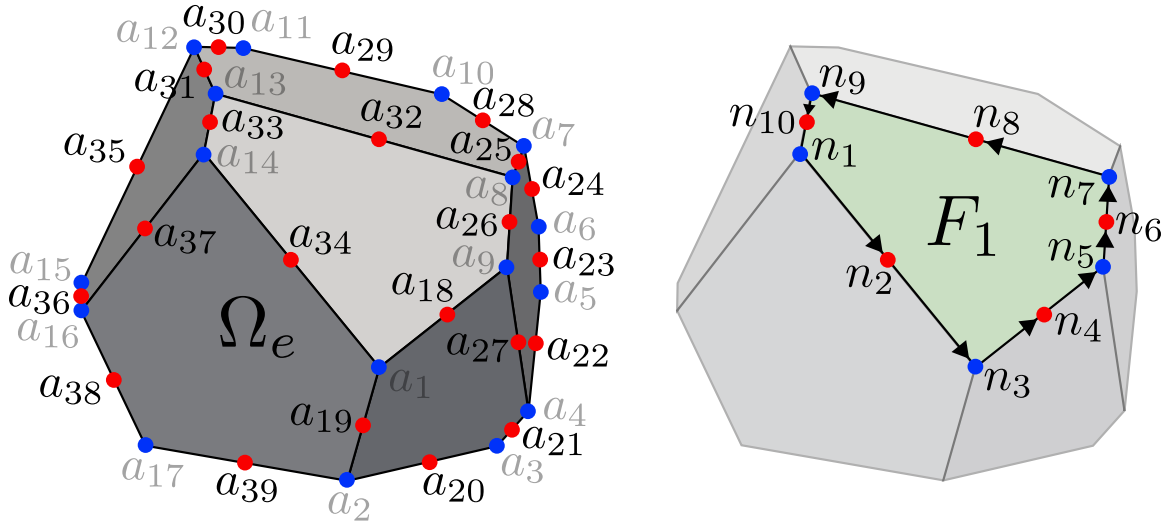


Figure 4.3: Illustration of the data necessary to describe a quadratic serendipity polyhedral element Ω_e . The local node ID ordering for the face F_1 shown would be $\{n_i\}_{i=1}^{10} = \{14, 34, 1, 18, 9, 26, 8, 32, 13, 33\}$.

Finite Element Data for Arbitrary Polytopes

Traditional Lagrangian finite element methods require each element $\Omega_e \subset \mathcal{B}_0$ to carry the following data (at a minimum) for the purposes of evaluating weak form integrals:

- A list of quadrature weights $\{w_q\}_{q=1}^{N_{qp}^{\Omega_e}}$ associated with the quadrature points of the element. The sub-index q induces a local *quadrature point ID*.

- Evaluations of the element's nodal shape functions $\{\varphi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{\Omega_e}} \forall a = 1, \dots, N_V^{\Omega_e}$ at each quadrature point location \mathbf{X}_q .
- Gradients (with respect to the element's reference coordinates \mathbf{X} at time $t = 0$) of the element's nodal shape functions $\{\nabla_X \varphi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{\Omega_e}} \forall a = 1, \dots, N_V^{\Omega_e}$.
- (Optionally) if the element relies upon some form of gradient correction scheme (or more generally, if a Petrov-Galerkin method is employed): evaluations and/or gradients of the element's test functions $\{\phi_a(\mathbf{X}_q), \nabla_X \phi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{\Omega_e}} \forall a = 1, \dots, N_V^{\Omega_e}$ must be stored, as well.

For solid mechanics applications, material state data (e.g. material properties, internal variables, Cauchy stress) would also need to be stored at each quadrature point location.

Similarly, each boundary face $F_b \subset \Gamma_0^N$ must carry:

- A list of quadrature weights $\{w_q\}_{q=1}^{N_{qp}^{F_b}}$ for each quadrature point of the face.
- Evaluations of the face's nodal shape functions $\{\varphi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{F_b}} \forall a = 1, \dots, N_V^{F_b}$ at each quadrature point location \mathbf{X}_q .
- Gradients (with respect to the face's in-plane reference coordinates \mathbf{X} at $t = 0$) of the face's nodal shape functions $\{\nabla_X \varphi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{F_b}} \forall a = 1, \dots, N_V^{F_b}$.
- If a Petrov-Galerkin method is employed: evaluations and/or gradients of the face's test functions $\{\phi_a(\mathbf{X}_q), \nabla_X \phi_a(\mathbf{X}_q)\}_{q=1}^{N_{qp}^{F_b}} \forall a = 1, \dots, N_V^{F_b}$.
- Outward unit normals $\{\mathbf{N}_q\}_{q=1}^{N_{qp}^{F_b}}$ to the face F_b at each quadrature point.

The data enumerated above must be determined via a specified *element formulation*: the method by which the element's shape functions and quadrature rule are constructed. Partitioned element methods address precisely this task. Given an abstract representation for the geometry of a given polyhedral element Ω (as discussed in the previous section), a PEM formulation proceeds in a number of distinct steps:

- 1.) The element (and its faces, edges) are appropriately partitioned into cells (facets, segments, vertices).

- 2a.) Individual nodal shape functions are constructed along each edge E of the element.
- 2b.) Individual nodal shape functions are constructed on each face F of the element.
- 2c.) Individual nodal shape functions are constructed on the interior of the element Ω .
- 3.) The discrete finite element data (including quadrature point evaluations of the shape functions and their gradients) are computed and stored by the element.
- 4.) Any auxiliary data (regarding the element's partitioned geometry, etc.) is discarded.

Depending on how the chosen PEM is carried out, the above process can amount to a relatively large computational expense. However, if a total Lagrangian approach is employed within the associated finite element analysis, then the above methodology would only need to be carried out once for each element, at the beginning of the simulation (prior to the first time step). Consequently, the cost of constructing element shape functions in this manner is amortized over the duration of the analysis.

The subsequent sections of this chapter are dedicated to a more detailed discussion of the aforementioned steps taken to construct a given element's partition, and its shape functions.

4.2 Element Partitioning Schemes

The process of obtaining a suitable partition for a given element constitutes the greatest challenge facing partitioned element methods – an issue of computational geometry, primarily. The secondary complication arises from the conditions of stability that the element must satisfy: the shape function approximations, and the corresponding quadrature rules defined on the element's partition must guarantee a sufficiently stable integration of the weak form. For relatively simple shapes, a number of stable partitioning schemes exist. However, for arbitrary shapes, it becomes difficult – if not impossible – for a heuristically-driven discretization scheme to guarantee that the resulting partition will satisfy the aforementioned stability requirements.

For this reason, we will limit our subsequent discussion to *star-convex* element geometries. For such shapes, a relatively simple partitioning scheme is suggested, resembling the

decomposition used in the edge-based smoothed finite element method. In the absence of degenerate features, the resulting partition is verified to provide sufficiently stable shape function approximations and quadrature rules.

Edge-Based Partitioning for Star-Convex Elements

A star-convex shape Ω is one for which there exists some interior point $\mathbf{X}_0^\Omega \in \Omega$ such that the line segment connecting any point $\mathbf{X} \in \Omega$ to \mathbf{X}_0^Ω is entirely contained within Ω . If $\Omega \subset \mathbb{R}^3$ refers to a polyhedron which is star-convex, this further implies that each polygonal face $F \subset \partial\Omega$ is also star-convex with respect to some point $\mathbf{X}_0^F \in F$. Any linear edge $E \subset \partial F$ is star-convex with respect to any point $\mathbf{X} \in E$.

A simple and efficient partitioning scheme is described for polyhedral elements Ω (and their polygonal faces F_j) which are star-convex with respect to their vertex-averaged centroids $\bar{\mathbf{X}}_0^\Omega$ (or $\bar{\mathbf{X}}_0^{F_j}$), i.e.

$$\bar{\mathbf{X}}_0^\Omega = \frac{1}{N_V^\Omega} \sum_{i=1}^{N_V^\Omega} \mathbf{X}_i, \quad \text{and} \quad \bar{\mathbf{X}}_0^{F_j} = \frac{1}{N_V^{F_j}} \sum_{i=1}^{N_V^{F_j}} \mathbf{X}_i. \quad (4.1)$$

For a given polyhedral element Ω , the algorithm proceeds in several steps:

- 1.) Identify all edges of the polyhedron, and subdivide them into linear segments; each segment should join two adjacent nodes of a given edge.
- 2.) For each face, compute its vertex-averaged centroid $\bar{\mathbf{X}}_0^{F_j}$, and subdivide the face into triangular facets which emanate from $\bar{\mathbf{X}}_0^{F_j}$; each facet should share at most two segments (at least one segment) with ∂F_j .
- 3a.) Compute the element's vertex-averaged centroid $\bar{\mathbf{X}}_0^\Omega$, and subdivide the element into tetrahedra which emanate from $\bar{\mathbf{X}}_0^\Omega$; each tetrahedron should share a single facet with $\partial\Omega$.
- 3b.) Lump into cells any tetrahedra which share a segment belonging to an edge of Ω ; each cell should consist of exactly two tetrahedra.

An illustration of this process is depicted in Figure 4.4. A similar algorithm may be applied to each boundary face $F_b \subset \Gamma_0^N$, entailing only the first two steps described above.

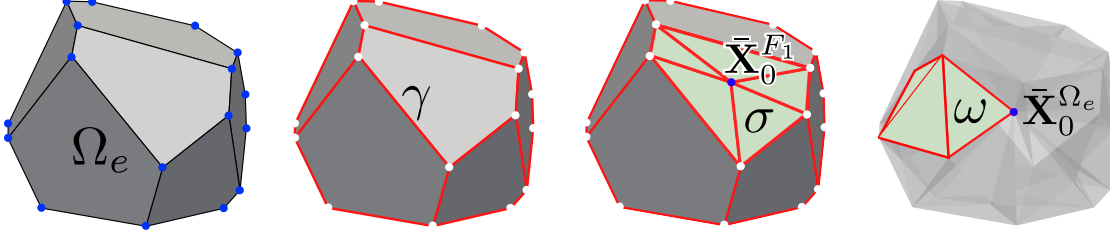


Figure 4.4: The resulting segment, facet, and cell decomposition for the proposed edge-based partitioning algorithm.

The algorithm can also be naturally extended to accommodate serendipity polyhedral elements, with additional nodes belonging to element edges, as depicted in Figure 4.5.

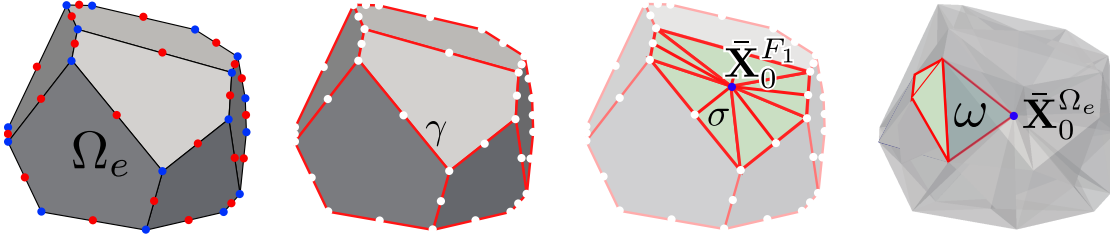


Figure 4.5: The resulting segment, facet, and cell decomposition for the proposed edge-based partitioning algorithm, applied to a quadratic serendipity polyhedral element.

A crucial advantage of this approach is that the partition deterministically obtained for each edge (face) will be identical between all elements that share that edge (face), leading to direct satisfaction of (3.67). As such, the gradient correction schemes discussed in section 3.6 may be applied independently to each element. Moreover, because each element may be partitioned in an autonomous fashion (without knowledge of any neighboring elements), the computations required to construct each element's shape functions can be directly parallelized.

4.3 Abstract Geometric Data Structures

Because partitioned element methods consist of solving a set of 1D, 2D, and 3D problems on each edge, face, and element, there arise a number of similarities between these

problems of variable dimension. Namely, the geometric data describing each cell, facet, segment and vertex may be abstracted through the use of generic parent-child (tree-based) data structures. Instead of requiring a separate implementation for the solution of each 1D, 2D, 3D problem, generic programming paradigms are exploited to facilitate a single, unified implementation which is agnostic to the dimensionality of the problem being solved. The proposed organization shares many similarities with the generic programming approaches presented in [18]. A number of definitions for the abstract data structures used henceforth are given in the following sections.

Geometric Entities

Geometric entities (or simply *entities*) are defined as the atomic units of the element's geometric partition. An entity may be: a polyhedral cell, a polygonal facet, a linear segment, or a vertex. Entities are defined in terms of their relationship to other geometric entities. Specifically, a 3-dimensional entity (a polyhedral cell ω) is uniquely defined in terms of its 2-dimensional facets $\sigma \subset \partial\omega$ (termed the “children” of ω). Each facet σ is in turn considered a 2-dimensional entity, defined in terms of its 1-dimensional children (its bounding segments $\gamma \subset \partial\sigma$). An entity of dimension 0 (a vertex) is identified as having no children.

Each geometric entity may be viewed as a “node” in a corresponding tree diagram, as illustrated in Figure 4.6, where the height of a given entity within the tree structure indicates its dimensionality d .

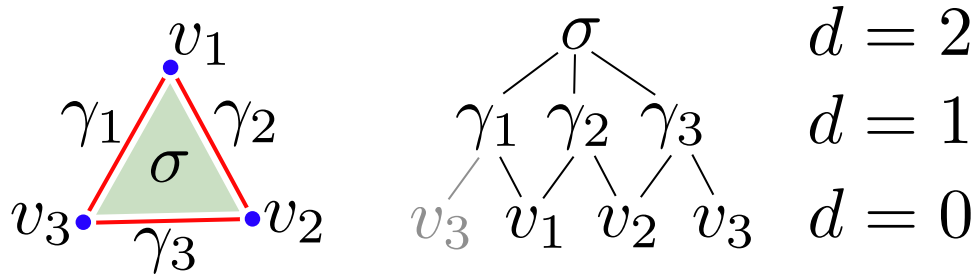


Figure 4.6: A representative entity tree diagram for a 2-dimensional facet σ and its children.

Henceforth, we denote by ε any generic d -dimensional entity, and by ζ any $(d - 1)$ -dimensional child of ε (such that $\zeta \subset \partial\varepsilon$). The data stored on a given entity ε consists

of:

- Information (pointers or entity IDs) referring to the $(d - 1)$ -dimensional children of ε , and/or information (pointers or entity IDs) referring to the $(d + 1)$ -dimensional parents of ε .
- An (optional) orientation/unit direction \mathbf{N}_ε .
- A quadrature rule $\{\mathbf{X}_q, w_q\}_{q=1}^{N_{qp}^\varepsilon}$ defined on ε , or pre-computed monomial integrals $\int_\varepsilon \mathbf{X}^\alpha d\varepsilon$ for all $|\alpha| \leq k$ (alternatively, the shifted monomial moments $\int_\varepsilon (\mathbf{X} - \mathbf{X}_\varepsilon)^\alpha d\varepsilon$ may be computed and stored, instead.)
- A list of the DG-PEM basis function IDs which possess compact support over ε .

If quadrature rules are to be defined on each geometric entity, we may exploit the particular choice made regarding the element's edge-based partition, noting that every polygonal facet will be a triangle, and every polyhedral cell will consist of two adjoining tetrahedra, allowing for the use of standard Dunavant [23] and Grundmann-Möller [36] quadrature rules for triangles and tetrahedra, respectively. Standard Gaussian quadrature rules may be specified on each linear segment.

The advantage of defining entities in this fashion is that it affords greater flexibility in solving the DG-PEM problem (3.34) on elements with arbitrary dimensionality.

Sub-Elements

Herein, a *sub-element* (generically denoted as \mathcal{E}) is defined as a d -dimensional polytope (a polyhedral element, a polygonal face, a linear edge, or a node) upon which nodal shape functions are locally constructed and defined as the solution to a d -dimensional boundary value problem. Sub-elements consist of a partition $\mathcal{T}_\varepsilon(\mathcal{E})$ of \mathcal{E} into d -dimensional entities ε . The boundary of each sub-element \mathcal{E} is comprised of $(d - 1)$ -dimensional sub-elements, denoted $e \subset \partial\mathcal{E}$ (called the “children” of \mathcal{E} , in analog to the terminology used for geometric entities). A sub-element of dimension 0 (a node) refers to a single vertex, and possesses no children.

As a representative example, consider the face $F \subset \partial\Omega$ shown in Figure 4.7, which is classified as a 2-dimensional sub-element, whose partition $\mathcal{T}_\sigma(F)$ consists of 2-dimensional facets $\sigma \subset F$. The children of F are comprised of 1-dimensional sub-elements – edges $E \subset \partial F$; in turn, the children of each edge E are the nodes of the element. As with geometric entities, the parent-child relationships between different sub-elements induces a tree-like data structure, where the height of a given sub-element within the tree corresponds to its dimensionality d .

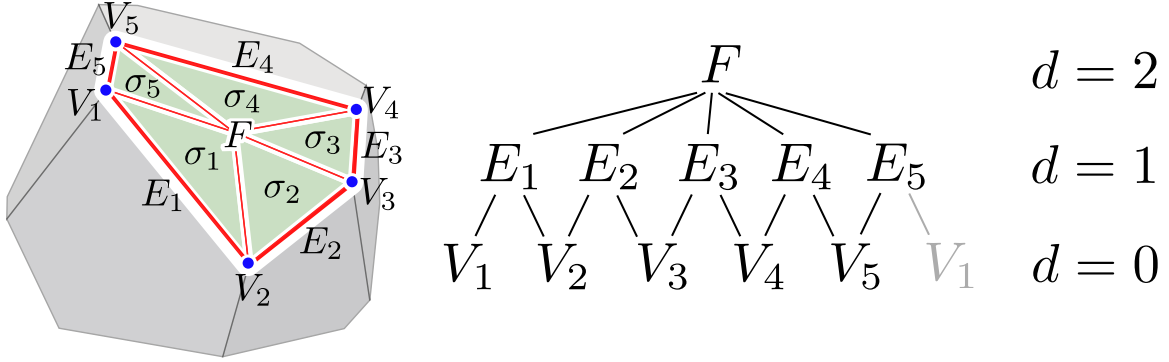


Figure 4.7: A representative 2-dimensional sub-element $F \subset \partial\Omega$ (a polygonal face), and its corresponding sub-element tree diagram. The partition of F consists of polygonal facets $\sigma_i \in \mathcal{T}_\sigma(F) \forall i = 1, \dots, 5$.

The data stored for a given sub-element \mathcal{E} consists of:

- Information (pointers or entity IDs) referring to the d -dimensional entities $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$ belonging to the partition of \mathcal{E} .
- Information (pointers or sub-element IDs) referring to the $(d - 1)$ -dimensional children of \mathcal{E} (the sub-elements $e \subset \partial\mathcal{E}$ which constitute the boundary of \mathcal{E}).
- A list of the DG-PEM basis function IDs which are defined on $\mathcal{T}_\varepsilon(\mathcal{E})$.

A more thorough discussion is dedicated to the subject of DG-PEM basis functions in the following section.

4.4 Partition-Based Approximation Spaces

As discussed in chapter 3, partitioned element methods consider a weak formulation of a given PDE (e.g. Laplace's equation), whose solution is approximated through the spec-

ification of a finite dimensional function space $\mathcal{U}^h(\mathcal{E})$ defined on the partition $\mathcal{T}_\varepsilon(\mathcal{E})$ of a given sub-element \mathcal{E} . The (sub-)element's shape functions are selected as the “best approximations” to (harmonic) functions which are contained within the chosen approximation space $\mathcal{U}^h(\mathcal{E})$.

Herein, we consider $\mathcal{U}^h(\mathcal{E}) = \mathcal{D}_k^h(\mathcal{E}) = \{\varphi \in L^2(\mathcal{E}) : \varphi|_\varepsilon \in P^k(\varepsilon) \forall \varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})\}$ to be spanned by a set of basis functions $\{\psi_A\}_{A=1}^{N_{bf}^\mathcal{E}}$, such that each basis function $\psi \in \mathcal{U}^h(\mathcal{E})$ is compactly supported over a given geometric entity $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$ (or patch of entities).

Any function $\varphi^h \in \mathcal{U}^h(\mathcal{E})$ may be written as a linear combination of the basis functions which span $\mathcal{U}^h(\mathcal{E})$, i.e.

$$\varphi^h(\mathbf{X}) = \sum_{A=1}^{N_{bf}^\mathcal{E}} \psi_A(\mathbf{X}) \varphi_A \quad \forall \mathbf{X} \in \mathcal{E}. \quad (4.2)$$

The restriction of φ^h to a given geometric entity ε may be written in terms of only those basis functions $\{\psi_a^\varepsilon\}_{a=1}^{N_{bf}^\varepsilon} \subset \mathcal{U}^h(\Omega)$ which possess compact support over ε , such that

$$\varphi^h|_\varepsilon(\mathbf{X}) = \sum_{a=1}^{N_{bf}^\varepsilon} \psi_a^\varepsilon(\mathbf{X}) \varphi_a \quad \forall \mathbf{X} \in \varepsilon. \quad (4.3)$$

To guarantee $P^k(\mathcal{E}) \subset \mathcal{U}^h(\mathcal{E})$ for some desired degree of polynomial completeness k , it is necessary for a given entity's local basis $\{\psi_a^\varepsilon\}_{a=1}^{N_{bf}^\varepsilon}$ to span $P^k(\varepsilon)$. Arguably the simplest such basis consists of the monomials through order k defined on each entity $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$. However, the use of (unscaled) monomial bases can lead to ill-conditioning of the DG-PEM problem, particularly as the maximal polynomial degree k is increased. This fact has been well-established in the literature pertaining to discontinuous Galerkin methods (see for example [39].) For this reason, other (well-scaled) polynomial bases are recommended, such as the Lagrange polynomials defined on each entity (for linear segments and triangular facets), or the orthogonal polynomials obtained via the methodology proposed in [8]. An exploration of the condition number for the resulting DG-PEM systems of equations is presented at the end of this chapter.

Irrespective of the chosen basis, each basis function $\psi(\mathbf{X})$ may be expressed as a low-order polynomial (of maximal degree k) via a linear combination of (possibly shifted)

monomials:

$$\psi(\mathbf{X}) = \sum_{|\alpha| \leq k} c_\alpha (\mathbf{X} - \mathbf{X}_0)^\alpha, \quad (4.4)$$

where $\alpha = \alpha_1, \dots, \alpha_d$ is a multi-index, such that $|\alpha| = \alpha_1 + \dots + \alpha_d$ and $\mathbf{X}^\alpha = X_1^{\alpha_1} \dots X_d^{\alpha_d}$. Consequently, a given basis function is uniquely defined in terms of its shifted coordinate \mathbf{X}_0 , and its monomial coefficients c_α . The gradient of a given basis function $\nabla \psi(\mathbf{X})$ may in turn be computed via

$$\frac{\partial \psi(\mathbf{X})}{\partial X_i} = \sum_{|\alpha| \leq k} \alpha_i c_\alpha (\mathbf{X} - \mathbf{X}_0)^{\alpha_1, \dots, (\alpha_i - 1), \dots, \alpha_d}. \quad (4.5)$$

4.5 Linearization and Assembly of the DG-PEM Systems of Equations

The DG-PEM problem of (3.34) may be written in terms of the chosen basis $\{\psi_A\}_{A=1}^{N_{bf}^\varepsilon}$ for $\mathcal{D}_k^h(\mathcal{E})$, such that the basis function coefficients $\varphi_A \ \forall A = 1, \dots, N_{bf}^\varepsilon$ representing a given shape function $\varphi^h(\mathbf{X}) = \sum_{A=1}^{N_{bf}^\varepsilon} \psi_A(\mathbf{X}) \varphi_A$ may be determined as the solution to a linear system of equations:

$$\sum_{A=1}^{N_{bf}^\varepsilon} J_{AB} \varphi_A = F_B \quad \forall B = 1, \dots, N_{bf}^\varepsilon, \quad (4.6)$$

where J_{AB} and F_B are computed via an entity-wise assembly process following to the methodology described in [61]. In addition to assembling local contributions J_{ab}^ε and F_b^ε to J_{AB} and F_B from all d -dimensional entities $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$, it is also necessary to assemble appropriate contributions from all $(d-1)$ -dimensional interfaces $\zeta \in \Gamma_\varepsilon \cup \partial \mathcal{E}$, as well.

Specifically, each d -dimensional entity $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$ contributes the following local arrays:

$$J_{ab}^\varepsilon = \int_\varepsilon \nabla \psi_a^\varepsilon \cdot \nabla \psi_b^\varepsilon d\varepsilon, \quad \forall a, b = 1, \dots, N_{bf}^\varepsilon, \quad (4.7)$$

$$F_b^\varepsilon = \int_\varepsilon f_\varepsilon \psi_b^\varepsilon d\varepsilon \quad \forall b = 1, \dots, N_{bf}^\varepsilon. \quad (4.8)$$

Each $\zeta \in \partial \mathcal{E}$ (which borders a single d -dimensional entity ε) contributes:

$$J_{ab}^\varepsilon = \int_\zeta \left(\epsilon (\mathbf{N}_\zeta \cdot \nabla \psi_b^\varepsilon) \psi_a^\varepsilon - \psi_b^\varepsilon (\mathbf{N}_\zeta \cdot \nabla \psi_a^\varepsilon) \right) d\zeta + \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta \psi_a^\varepsilon \psi_b^\varepsilon d\zeta \quad \forall a, b = 1, \dots, N_{bf}^\varepsilon, \quad (4.9)$$

$$F_b^\varepsilon = \int_\zeta \epsilon (\mathbf{N}_\zeta \cdot \nabla \psi_b^\varepsilon) \bar{\varphi} d\zeta + \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta \bar{\varphi} \psi_b^\varepsilon d\zeta \quad \forall b = 1, \dots, N_{bf}^\varepsilon, \quad (4.10)$$

and each $\zeta \in \Gamma_\varepsilon$ (which borders two d -dimensional entities, ε_1 and ε_2) contributes:

$$J_{ab}^{\varepsilon_i \varepsilon_j} = -\frac{1}{2} \int_\zeta \left((-1)^i \epsilon (\mathbf{N}_\zeta \cdot \nabla \psi_b^{\varepsilon_j}) \psi_a^{\varepsilon_i} - (-1)^j \psi_b^{\varepsilon_j} (\mathbf{N}_\zeta \cdot \nabla \psi_a^{\varepsilon_i}) \right) d\zeta \quad (4.11)$$

$$+ (-1)^{(i+j)} \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta \psi_a^{\varepsilon_i} \psi_b^{\varepsilon_j} d\zeta \quad (4.12)$$

$$+ (-1)^{(i+j)} \frac{\alpha_{\zeta 1}}{|\zeta|^{\beta_1}} \int_\zeta \nabla \psi_a^{\varepsilon_i} \cdot (\mathbf{N}_\zeta \otimes \mathbf{N}_\zeta) \cdot \nabla \psi_b^{\varepsilon_j} d\zeta \quad (4.13)$$

for all $i, j = 1, 2$, and $a = 1, \dots, N_{bf}^{\varepsilon_i}, b = 1, \dots, N_{bf}^{\varepsilon_j}$. For the remainder of our discussions, we will consider only the case where $f_\mathcal{E} \equiv 0$ for all sub-elements $\mathcal{E} \subset \Omega$.

For a given d -dimensional sub-element \mathcal{E} , the assembly of the above terms entails an entity-wise integration of products between basis functions (and their gradients) over d -dimensional entities $\varepsilon \in \mathcal{T}_\varepsilon(\mathcal{E})$ and their $(d-1)$ -dimensional children $\zeta \in \Gamma_\varepsilon \cup \partial\mathcal{E}$. This may be effected through the use of appropriately defined quadrature rules on each entity, or by pre-computing the monomial integrals over each entity up to some sufficiently high degree (nominally $2k$).

Additionally, it is remarked that the boundary condition $\bar{\varphi}$ is defined independently on every $(d-1)$ -dimensional sub-element $e \subset \partial\mathcal{E}$, such that its restriction $\bar{\varphi}|_\zeta$ may be written in terms of the local basis functions $\{\psi_c^\zeta\}_{c=1}^{N_{bf}^\zeta}$ belonging to a given boundary entity $\zeta \in \mathcal{T}_\zeta(e)$, i.e.

$$\bar{\varphi}|_\zeta(\mathbf{X}) = \sum_{c=1}^{N_{bf}^\zeta} \psi_c^\zeta(\mathbf{X}) \bar{\varphi}_c. \quad (4.14)$$

For a given $\zeta \in \partial\mathcal{E}$, this yields:

$$F_b^\varepsilon = \sum_{c=1}^{N_{bf}^\zeta} \bar{J}_{bc}^\varepsilon \bar{\varphi}_c \quad \forall b = 1, \dots, N_{bf}^\varepsilon, \quad (4.15)$$

which may be used in lieu of (4.10), where

$$\bar{J}_{bc}^\varepsilon = \int_\zeta \epsilon (\mathbf{N}_\zeta \cdot \nabla \psi_b^\varepsilon) \psi_c^\zeta d\zeta + \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta \psi_c^\zeta \psi_b^\varepsilon d\zeta \quad \forall b = 1, \dots, N_{bf}^\varepsilon, c = 1, \dots, N_{bf}^\zeta. \quad (4.16)$$

Consequently, we may re-write the right-hand side of (4.6) as a linear mapping from the basis coefficients $\bar{\varphi}_C$ which determine the boundary function $\bar{\varphi}(\mathbf{X}) = \sum_{C=1}^{N_{bf}^{\partial\mathcal{E}}} \psi_C(\mathbf{X}) \bar{\varphi}_C$:

$$\sum_{A=1}^{N_{bf}^{\mathcal{E}}} J_{AB} \varphi_A = \sum_{C=1}^{N_{bf}^{\partial\mathcal{E}}} \bar{J}_{BC} \bar{\varphi}_C \quad \forall B = 1, \dots, N_{bf}^{\mathcal{E}}. \quad (4.17)$$

Alternatively, in matrix-vector format:

$$\mathbf{J} \boldsymbol{\varphi} = \bar{\mathbf{J}} \bar{\boldsymbol{\varphi}}. \quad (4.18)$$

The above system of equations can be solved to obtain a linear mapping from $\bar{\boldsymbol{\varphi}}$ to $\boldsymbol{\varphi}$, denoted $\mathbf{M}^{\partial\mathcal{E} \mapsto \mathcal{E}} = \mathbf{J}^{-1} \bar{\mathbf{J}} : \mathbb{R}^{N_{bf}^{\partial\mathcal{E}}} \mapsto \mathbb{R}^{N_{bf}^{\mathcal{E}}}$. This mapping may be utilized to define the shape functions in a recursive fashion (over the children of each sub-element, in succession), ultimately yielding a representation for the shape functions on \mathcal{E} in terms of only nodal evaluations $\varphi|_V$. This process is discussed in greater detail in the following section.

Hierarchical Construction of Shape Functions

For a given polyhedral element $\Omega \subset \mathbb{R}^d$, we presume that each shape function φ_A is associated with a particular node V_A of the element, such that

$$\varphi_A|_{V_B} = \delta_{AB} \quad \forall A, B = 1, \dots, N_V^{\Omega}. \quad (4.19)$$

Additionally, for the sake of simplicity (and computational efficiency), it is suggested that the shape functions along each linear edge E be constructed from the standard Lagrange polynomials which interpolate the nodal values of that edge.

For each polygonal face F , linear mappings $\mathbf{M}^{\partial F \mapsto F} : \mathbb{R}^{N_V^F} \mapsto \mathbb{R}^{N_{bf}^F}$ may be constructed according to the process described in the previous section, yielding

$$\varphi_A|_F(\mathbf{X}) = \sum_{b=1}^{N_{bf}^F} \sum_{c=1}^{N_V^F} \psi_b(\mathbf{X}) M_{bc}^{\partial F \mapsto F} \varphi_A|_{V_c}. \quad (4.20)$$

Subsequently, a mapping $\mathbf{M}^{\partial\Omega \mapsto \Omega} : \mathbb{R}^{N_{bf}^{\partial\Omega}} \mapsto \mathbb{R}^{N_{bf}^{\Omega}}$ is constructed to determine the representation for the shape functions on the interior of the element, which may be composed with the mappings $\mathbf{M}^{\partial F \mapsto F}$ for each face to determine a final mapping $\mathbf{M}^{V \mapsto \Omega} : \mathbb{R}^{N_V^{\Omega}} \mapsto \mathbb{R}^{N_{bf}^{\Omega}}$ which yields

$$\varphi_A|_{\Omega}(\mathbf{X}) = \sum_{b=1}^{N_{bf}^{\Omega}} \sum_{c=1}^{N_V^{\Omega}} \psi_b(\mathbf{X}) M_{bc}^{V \mapsto \Omega} \varphi_A|_{V_c}. \quad (4.21)$$

Once the above representations for the element's shape function have been obtained, the discrete data required of the element (evaluations of the shape functions and their gradients at a discrete number of quadrature points $\{\mathbf{X}_q\}_{q=1}^{N_{qp}}$) can be obtained through direct evaluation of (4.21), i.e.

$$\varphi_A|_{\Omega}(\mathbf{X}_q) = \sum_{b=1}^{N_{bf}^{\Omega}} \sum_{c=1}^{N_V^{\Omega}} \psi_b(\mathbf{X}_q) M_{bc}^{V \mapsto \Omega} \varphi_A|_{V_c}, \quad (4.22)$$

and

$$\nabla \varphi_A|_{\Omega}(\mathbf{X}_q) = \sum_{b=1}^{N_{bf}^{\Omega}} \sum_{c=1}^{N_V^{\Omega}} \nabla \psi_b(\mathbf{X}_q) M_{bc}^{V \mapsto \Omega} \varphi_A|_{V_c}. \quad (4.23)$$

Once the discrete quadrature point evaluations for the shape functions have been obtained, there is no need to retain any information regarding the DG-PEM basis functions, or the transitional mappings; these artifacts and their corresponding data structures may therefore be discarded.

The specification of the element's quadrature rule is discussed in the following section.

Construction of Quadrature Rules

The quadrature rule for an element $\{\mathbf{X}_q, w_q\}_{q=1}^{N_{qp}}$ nominally consists of quadrature points \mathbf{X}_q located at the centroids of each cell $\omega \in \mathcal{T}_{\Omega}(\Omega)$, whose corresponding quadrature weights w_q are equal to the volumes of each respective cell $|\omega|$. This results in the composite mid-point rule discussed in chapter 3.

As previously discussed, the volumes $|\omega|$ and centroids $\bar{\mathbf{X}}_{\omega}$ of each cell may be obtained via the exact integration formulas provided by Chin et al. in [16]. Alternatively, we may exploit the geometric simplicity of the chosen edge-based partitioning scheme to compute these quantities more easily, by considering the fact that each edge-centered cell ω consists of two tetrahedral sub-domains (i.e. $T_1 \cup T_2 = \omega$). Volumes and centroids for each tetrahedron can be easily computed using standard formulas, and the resulting quantities of interest may be expressed as

$$|\omega| = |T_1| + |T_2|, \quad \bar{\mathbf{X}}_{\omega} = \frac{|T_1| \bar{\mathbf{X}}_{T_1} + |T_2| \bar{\mathbf{X}}_{T_2}}{|\omega|}. \quad (4.24)$$

Straightforward integration rules $\{\mathbf{X}_q, w_q\}_{q=1}^{N_{qp}^F}$ for each face $F \subset \partial\Omega$ are more easily obtained by considering the corresponding areas $|\sigma|$ and centroids $\bar{\mathbf{X}}_{\sigma}$ of each (strictly

triangular) facet $\sigma \in \mathcal{T}_\sigma(F)$. Outward unit normals \mathbf{N}_σ to each facet may also be readily computed and stored, as needed.

With these quantities in hand, the element's shape functions (and their gradients) may be evaluated at the quadrature points of the element, as described in the previous section. Subsequently, a consistent integration scheme (as discussed in 3) may be constructed from the discrete data stored at the element's quadrature points. If a gradient correction scheme is adopted, the corrected test function gradients $\nabla\phi_A(\mathbf{X}_q)$ at all quadrature points may be computed and stored, as needed.

4.6 Numerical Conditioning of the DG-PEM Systems of Equations

As a representative example, consider a given sub-element $\mathcal{E} \subset \mathbb{R}^d$ and its corresponding partition into d -dimensional entities $\varepsilon \subset \mathcal{T}_\varepsilon(\mathcal{E})$. Recall that the representation of a given shape function $\varphi \in \mathcal{D}_k^h(\mathcal{E})$ is piecewise polynomial in each entity, i.e. $\varphi|_\varepsilon \in P^k(\varepsilon)$.

A reasonable estimate for the condition number $\kappa(\mathbf{J})$ of the SPD matrix \mathbf{J} appearing in (4.18) may be obtained via

$$\frac{\max_a |J_{aa}^\varepsilon|}{\min_a |J_{aa}^\varepsilon|} \leq \kappa(\mathbf{J}), \quad (4.25)$$

where

$$\begin{aligned} J_{aa}^\varepsilon &= \int_\varepsilon \nabla \psi_a^\varepsilon \cdot \nabla \psi_a^\varepsilon d\varepsilon + (\epsilon - 1) \sum_{\zeta \in \partial\varepsilon \cap \partial\mathcal{E}} \int_\zeta (\psi_a^\varepsilon) (\mathbf{N}_\zeta \cdot \nabla \psi_a^\varepsilon) d\zeta \\ &\quad + \frac{1}{2}(\epsilon - 1) \sum_{\zeta \in \partial\varepsilon \cap \Gamma_\varepsilon} \int_\zeta (\psi_a^\varepsilon) (\mathbf{N}_\zeta \cdot \nabla \psi_a^\varepsilon) d\zeta \\ &\quad + \sum_{\zeta \in \partial\varepsilon} \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta (\psi_a^\varepsilon)^2 d\zeta + \sum_{\zeta \in \partial\varepsilon \cap \Gamma_\varepsilon} \frac{\alpha_{\zeta 1}}{|\zeta|^{\beta_1}} \int_\zeta (\mathbf{N}_\zeta \cdot \nabla \psi_a^\varepsilon)^2 d\zeta. \end{aligned} \quad (4.26)$$

A brief remark should be made regarding the positivity of J_{aa}^ε , and its relation to the particular choice of DG method (i.e. whether the SIPG, NIPG, or IIPG is chosen). As discussed in [61], the NIPG method with $\epsilon = +1$ will yield strictly positive diagonal entries provided $\alpha_{\zeta 0} > 0$, resulting in the coercivity of \mathbf{J} . If the SIPG ($\epsilon = -1$) or IIPG ($\epsilon = 0$) methods are employed, then the penalty parameters must be made sufficiently large enough to guarantee coercivity, stability.

The subsequent discussion examines the case where $\epsilon = +1$, yielding strictly positive diagonal entries:

$$J_{aa}^\epsilon = \int_\epsilon \nabla \psi_a^\epsilon \cdot \nabla \psi_a^\epsilon d\epsilon + \sum_{\zeta \in \partial\epsilon} \frac{\alpha_{\zeta 0}}{|\zeta|^{\beta_0}} \int_\zeta (\psi_a^\epsilon)^2 d\zeta + \sum_{\zeta \in \partial\epsilon \cap \Gamma_\epsilon} \frac{\alpha_{\zeta 1}}{|\zeta|^{\beta_1}} \int_\zeta (\mathbf{N}_\zeta \cdot \nabla \psi_a^\epsilon)^2 d\zeta. \quad (4.27)$$

Let h_ϵ denote the diameter of ϵ , such that

$$h_\epsilon = \sup_{\mathbf{X}_1, \mathbf{X}_2 \in \epsilon} \|\mathbf{X}_1 - \mathbf{X}_2\|_2. \quad (4.28)$$

Using the following trace inequalities:

$$\int_\zeta (\psi_a^\epsilon)^2 d\zeta \leq C h_\epsilon^{-1/2} \int_\epsilon (\psi_a^\epsilon)^2 d\epsilon \quad \forall \zeta \subset \partial\epsilon, \quad (4.29)$$

$$\int_\zeta (\mathbf{N}_\zeta \cdot \nabla \psi_a^\epsilon)^2 d\zeta \leq C h_\epsilon^{-1/2} \int_\epsilon \nabla \psi_a^\epsilon \cdot \nabla \psi_a^\epsilon d\epsilon \quad \forall \zeta \subset \partial\epsilon, \quad (4.30)$$

and observing that $|\zeta|^{\beta_0} \leq h_\epsilon$, $|\zeta|^{\beta_1} \geq h_\epsilon^{-1}$ if $\beta_0 = (d-1)^{-1}$, $\beta_1 = -(d-1)^{-1}$ (assuming $h_\epsilon \geq 1$, without loss of generality), then

$$J_{aa}^\epsilon \leq (1 + C_1 h_\epsilon^{1/2}) \int_\epsilon \nabla \psi_a^\epsilon \cdot \nabla \psi_a^\epsilon d\epsilon + C_2 h_\epsilon^{-3/2} \int_\epsilon (\psi_a^\epsilon)^2 d\epsilon, \quad (4.31)$$

for some $C_1, C_2 \in \mathbb{R}$.

If $\varphi|_\epsilon$ is represented in terms of the (unshifted and unscaled) monomial basis (i.e. $\{\varphi_a^\epsilon\}_{a=1}^{N_{bf}^\epsilon} = \{\mathbf{X}^\alpha\}_{|\alpha| \leq k}$), then by application of the max-min inequality:

$$\int_\epsilon \mathbf{X}^{2\alpha} d\epsilon \leq h_\epsilon^d \max_{\mathbf{X} \in \epsilon} \{\mathbf{X}^{2\alpha}\} \leq h_\epsilon^d R_\epsilon^{2|\alpha|}, \quad (4.32)$$

$$\int_\epsilon \nabla \mathbf{X}^\alpha \cdot \nabla \mathbf{X}^\alpha d\epsilon \leq h_\epsilon^d \max_{\mathbf{X} \in \epsilon} \{\nabla \mathbf{X}^\alpha \cdot \nabla \mathbf{X}^\alpha\} \leq h_\epsilon^d R_\epsilon^{2(|\alpha|-1)}, \quad (4.33)$$

where it is assumed that

$$R_\epsilon = \max_{\mathbf{X} \in \epsilon} \|\mathbf{X}\|_2, \quad (4.34)$$

and $R_\epsilon \geq 1$, without loss of generality. Under these conditions, it suffices to assert that $\exists C > 0$ which may be used to establish a uniform lower bound on J_{aa}^ϵ , such that

$$C [(h_\epsilon^d + C_1 h_\epsilon^{d+1/2}) + C_2 h_\epsilon^{d-3/2}] \leq J_{aa}^\epsilon \leq (h_\epsilon^d + C_1 h_\epsilon^{d+1/2}) R_\epsilon^{2\langle k-1 \rangle} + C_2 h_\epsilon^{d-3/2} R_\epsilon^{2k}. \quad (4.35)$$

If the penalty parameter α_{γ_0} is sufficiently small, we obtain the estimate $\kappa(\mathbf{J}) \approx O(R_{\max}^{2\langle k-1 \rangle})$, where $R_{\max} = \max \{ \max_{\varepsilon \in \mathcal{E}} R_{\varepsilon}, \max_{\varepsilon \in \mathcal{E}} R_{\varepsilon}^{-1} \}$. Otherwise, if α_{γ_0} is sufficiently large, then $\kappa(\mathbf{J}) \approx O(R_{\max}^{2k})$; a similar estimate may be obtained for the SIPG and IIPG methods. The latter estimate is also applicable to the pure penalty DG-PEM, provided sufficient stability is supplied by the inclusion of additional penalty terms (3.43).

Clearly, the condition number of \mathbf{J} under these circumstances may become unacceptably large for increasing values of k and R_{\max} , leading to numerical inaccuracies in the resulting shape functions. This can ultimately degrade the degree of precision achieved in finite element patch tests.

To mitigate this issue, the (sub-)element's shape functions may be computed with respect to a shifted and scaled coordinate system, e.g. $\mathbf{X}' = h_{\mathcal{E}}^{-1}(\mathbf{X} - \bar{\mathbf{X}}_0^{\mathcal{E}})$, where

$$h_{\mathcal{E}} = \sup_{\mathbf{X}_1, \mathbf{X}_2 \in \mathcal{E}} \|\mathbf{X}_1 - \mathbf{X}_2\|_2. \quad (4.36)$$

This assists in limiting the worst-case value of R_{\max} appearing in the previous estimate, yielding a somewhat improved bound on the condition number: $\kappa(\mathbf{J}) \approx O(\rho^{2k})$. We define $\rho = h_{\mathcal{E}}/h_{\min}$, and

$$h_{\min} = \inf_{\varepsilon \in \mathcal{T}_{\varepsilon}(\mathcal{E})} \sup_{B_r(\mathbf{X}_B) \subset \varepsilon} |2r| \quad (4.37)$$

denotes the smallest diameter of the largest ball $B_r(\mathbf{X}_B) = \{ \mathbf{X} \in \mathbb{R}^d : \|\mathbf{X} - \mathbf{X}_B\|_2 \leq r \}$ fully contained within any single entity $\varepsilon \in \mathcal{T}_{\varepsilon}(\mathcal{E})$. Nonetheless, the condition number of \mathbf{J} can become large if: a relatively refined partition of the element is utilized; if the aspect ratio of the element becomes large; or if the element's partition contains “sliver cells” (i.e. if an edge-based partition is employed on an element with nearly degenerate geometric features.) Some improvements in the conditioning of \mathbf{J} may be obtained by considering the DG basis described in [8]. The Lagrange polynomials (for triangles and tetrahedra) can also be used to obtain well-scaled bases for each entity.

REFERENCES

- [1] *LS-DYNA Keyword User's Manual, Volume II: Material Models.*
- [2] D. N. Arnold, D. Boffi, and R. S. Falk. Approximation by quadrilateral finite elements. *Mathematics of Computation*, 71(239):909–922, 2002.
- [3] D. N. Arnold, D. Boffi, R. S. Falk, and L. Gastaldi. Finite element approximation on quadrilateral meshes. *Communications in Numerical Methods in Engineering*, 17(11):805–812, 2001.
- [4] Ivo Babuška. Error-bounds for finite element method. *Numerische Mathematik*, 16:322–333, 1971.
- [5] Ivo Babuška and Manil Suri. Locking effects in the finite element approximation of elasticity problems. *Numerische Mathematik*, 62:439–463, 1992.
- [6] Ivo Babuška and Manil Suri. On locking and robustness in the finite element method. *SIAM Journal on Numerical Analysis*, 29(5):1261–1293, 1992.
- [7] Biswajit Banerjee. Taylor impact tests: Detailed report. *C-SAFE Internal Report*, Report No. C-SAFE-CD-IR-05-001, 2005.
- [8] F. Bassi, L. Botti, A. Colombo, D.A. Di Pietro, and P. Tesini. On the flexibility of agglomeration based physical space discontinuous galerkin discretizations. *Journal of Computational Physics*, 231(1):45 – 65, 2012.
- [9] G.P. Bazeley, Y.K. Cheung, B.M. Irons, and O.C. Zienkiewicz. Triangular elements in bending. conforming and non-conforming solutions. *Proc. Conf. Matrix Methods in Structural Mechanics*, 1965.
- [10] J. E. Bishop. A displacement-based finite element formulation for general polyhedra using harmonic shape functions. *International Journal for Numerical Methods in Engineering*, 97:1–31, 2014.
- [11] Franco Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers. *ESAIM: Mathematical Modelling and Numerical Analysis - Modlisation Mathmatique et Analyse Numrique*, 8:129–151, 1974.
- [12] Erik Burman. A penalty-free nonsymmetric nitsche-type method for the weak imposition of boundary conditions. *SIAM Journal on Numerical Analysis*, 50(4):1959–1981, 2012.
- [13] Jiun-Shyan Chen, Michael Hillman, and Marcus Rüter. An arbitrary order variationally consistent integration for galerkin meshfree methods. *International Journal for Numerical Methods in Engineering*, 95:387–418, 2013.

- [14] H. Chi, L. Beirão da Veiga, and G. H. Paulino. Some basic formulations of the virtual element method (vem) for finite deformations. *Computer Methods in Applied Mechanics and Engineering*, 318:148–192, 2017.
- [15] Heng Chi, Cameron Talischi, Oscar Lopez-Pamies, and Glaucio H. Paulino. A paradigm for higher-order polygonal elements in finite elasticity using a gradient correction scheme. *Computer Methods in Applied Mechanics and Engineering*, 306:216–251, 2016.
- [16] Eric B. Chin, Jean B. Lasserre, and N. Sukumar. Numerical integration of homogeneous functions on convex and nonconvex polygons and polyhedra. *Computational Mechanics*, 56:967–981, 2015.
- [17] Zhong ci Shi. The f-e-m-test for convergence of nonconforming finite elements. *Mathematics of Computation*, 49(180):391–405, 1987.
- [18] M. Cicuttin, D.A. Di Pietro, and A. Ern. Implementation of discontinuous skeletal methods on arbitrary-dimensional, polytopal meshes using generic programming. *Journal of Computational and Applied Mathematics*, 2017.
- [19] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo. Basic principles of virtual element methods. *Computer Methods in Applied Mechanics and Engineering*, 23:199–214, 2013.
- [20] L. Beirão da Veiga, C. Lovadina, and D. Mora. A virtual element method for elastic and inelastic problems on polytope meshes. *Computer Methods in Applied Mechanics and Engineering*, 295:327–346, 2015.
- [21] E. A. de Souza Neto, D. Perić, M. Dukto, and D. R. J. Owen. Design of simple low order finite elements for large strain analysis of nearly incompressible solids. *International Journal of Solids and Structures*, 33:3277–3296, 1996.
- [22] C. R. Dohrmann and M. M. Rashid. Polynomial approximation of shape function gradients from element geometries. *International Journal for Numerical Methods in Engineering*, 53:945–958, 2002.
- [23] D. A. Dunavant. High degree efficient symmetrical gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, 21(6):1129–1148, 1985.
- [24] Standard test methods for tension testing of metallic materials. Standard, ASTM International, 100 Barr Harbor Drive, PO Box C700, West Conshohocken, PA 19428-2959, United States, 2008.
- [25] Mohamed S. Ebeida and Scott A. Mitchell. Uniform random voronoi meshes. *Proceedings of the 20th International Meshing Roundtable*, pages 273–290, 2011.
- [26] Mohamed Salah Ebeida. Vorocrust v. 1.0, Jul 2017.

- [27] Carlos A. Felippa. *Introduction to Finite Element Methods*. University of Colorado, Boulder, 2004.
- [28] Sonia Fernández-Méndez and Antonio Huerta. Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering*, 193(12):1257 – 1275, 2004. Meshfree Methods: Recent Advances and New Applications.
- [29] D. P. Flanagan and T. Belytschko. A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. *International Journal for Numerical Methods in Engineering*, 17:679–706, 1981.
- [30] Michael S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, 20:19–27, 2003.
- [31] Jouni Freund and Rolf Stenberg. On weakly imposed boundary conditions for second order problems. pages 327–336, 01 1995.
- [32] Arun Gain, Cameron Talischi, and Glaucio H. Paulino. On the virtual element method for three-dimensional elasticity problems on arbitrary polyhedral meshes. *Computer Methods in Applied Mechanics and Engineering*, 282, 11 2013.
- [33] Xifeng Gao, Wenzel Jakob, Marco Tarini, and Daniele Panozzo. Robust hex-dominant mesh generation using field-guided polyhedral agglomeration. *ACM Transactions on Graphics*, 36, 2017.
- [34] W. W. Gerberich, H. E. Martens, and R. A. Boundy. Tensile properties of five low-alloy and stainless steels under high-heating-rate and constant-temperature conditions. *Jet Propulsion Laboratory*, Technical Report No. 32-222, 1962.
- [35] William J. Gordon and James A. Wixom. Pseudo-harmonic interpolation on convex domains. *SIAM Journal on Numerical Analysis*, 11(5):909–933, 1974.
- [36] Axel Grundmann and H. M. Möller. Invariant integration formulas for the n-simplex by combinatorial methods. *SIAM Journal on Numerical Analysis*, 15(2):282–290, 1978.
- [37] W. H. Gust. High impact deformation of metal cylinders at elevated temperatures. *Journal of Applied Physics*, 53(5):3566–3575, 1982.
- [38] S. S. Hecker. Influence of deformation history on the yield locus and stress-strain behavior of aluminum and copper. *Metallurgical Transactions*, 4:985–989, 1973.
- [39] Jan S. Hesthaven and Tim Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2010.

- [40] Hans M. Hilber, Thomas J. R. Hughes, and Robert L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5(3):283–292, 1977.
- [41] Thomas J. R. Hughes. *The Finite Element Method—Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000.
- [42] G.R. Johnson and W.H. Cook. A constitutive model and data for metals subjected to large strains, high strain rates, and high temperatures. *Proceedings 7th International Symposium on Ballistics, The Hague*, 19-21:541–547, April 1983.
- [43] Pushkar Joshi, Mark Meyer, Tony DeRose, Brian Green, and Tom Sanocki. Harmonic coordinates for character articulation. *ACM Transactions on Graphics*, 26, 2007.
- [44] Mika Juntunen and Rolf Stenberg. Nitsche’s method for general boundary conditions. *Mechanics of Computation*, 78(267):1353–1374, 2009.
- [45] Nam-Sua Lee and Klaus-Jürgen Bathe. Effects of element distortions on the performance of isoparametric elements. *International Journal for Numerical Methods in Engineering*, 36:3553–3576, 1993.
- [46] Qiaoluan H. Li and Junping Wang. Weak galerkin finite element methods for parabolic equations. *Numerical Methods for Partial Differential Equations*, 29:2004–2024, 2013.
- [47] Guang Lin, Jiangguo Liu, and Farrah Sadre-Marandi. A comparative study on the weak galerkin, discontinuous galerkin, and mixed finite element methods. *Journal of Computational and Applied Mathematics*, 273:346–362, 2015.
- [48] Richard H. MacNeal. A theorem regarding the locking of tapered four-noded membrane elements. *International Journal for Numerical Methods in Engineering*, 24:1793–1799, 1987.
- [49] David S. Malkus and Thomas J.R. Hughes. Mixed finite element methods reduced and selective integration techniques: A unification of concepts. *Computer Methods in Applied Mechanics and Engineering*, 15(1):63 – 81, 1978.
- [50] Sebastian Martin, Peter Kaufmann, Mario Botsch, Martin Wicke, and Markus Gross. Polyhedral finite elements using harmonic basis functions. *Eurographics Symposium on Geometry Processing 2008*, 27(5), 2008.
- [51] L. Mu, J. Wang, and Y. Wang. A computational study of the weak galerkin method for second-order elliptic equations. *Numerical Algorithms*, 63:753, 2013.
- [52] L. Mu, J. Wang, and X. Ye. Weak galerkin finite element method for second-order elliptic problems on polytopal meshes. *International Journal of Numerical Analysis and Modeling*, 12:31–53, 2015.

- [53] Lin Mu, Junping Wang, and Xiu Ye. A weak galerkin finite element method with polynomial reduction. *Journal of Computational and Applied Mathematics*, 285:45–58, 2015.
- [54] J.Tinsley Oden, Ivo Babuška, and Carlos Erik Baumann. A discontinuous finite element method for diffusion problems. *J. Comput. Phys.*, 146(2):491–519, October 1998.
- [55] Daniel Pantuso and Klaus-Jürgen Bathe. On the stability of mixed finite elements in large strain analysis of incompressible solids. *Finite Elements in Analysis and Design*, 28:83–104, 1997.
- [56] M. M. Rashid. Incremental kinematics for finite element applications. *International Journal for Numerical Methods in Engineering*, 36:3937–3956, 1993.
- [57] M. M. Rashid and P. M. Gullett. On a finite element method with variable element topology. *Computer Methods in Applied Mechanics and Engineering*, 190:1509–1527, 2000.
- [58] M. M. Rashid and A. Sadri. The partitioned element method in computational solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 237–240:152–165, 2012.
- [59] M. M. Rashid and M. Selimotić. A three-dimensional finite element method with arbitrary polyhedral elements. *International Journal for Numerical Methods in Engineering*, 67:226–252, 2006.
- [60] J.-F. Remacle, J. Lambrechts, B. Seny, E. Marchandise, A. Johnen, and C. Geuzainet. Blossom-quad: a non-uniform quadrilateral mesh generator using a minimum cost perfect matching algorithm. *International Journal for Numerical Methods in Engineering*, 89:1102–1119, 2012.
- [61] Béatrice Rivière. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*. Society for Industrial and Applied Mathematics, 3600 Market Street, 6th Floor, Philadelphia, PA 19104-2688 USA, 2008.
- [62] Béatrice Rivière, Mary F. Wheeler, and Vivette Girault. A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems. *SIAM Journal on Numerical Analysis*, 39(3):902–931, 2001.
- [63] A. C. Robinson, Martin W. Heinstein, and Guglielmo Scovazzi. Lagrangian modeling at sandia national laboratories: Current status and future directions. *Proceedings of 5LC*, 19-21:541–547, 2005.
- [64] Mili Selimotić. *Polyhedral Finite-Element Approximants in 3D Solid Mechanics*. PhD thesis, University of California, Davis, 2008.

- [65] J. C. Simo and F. Armero. Geometrically non-linear enhanced strain mixed methods and the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 33:1413–1449, 1992.
- [66] J. C. Simo, F. Armero, and R. L. Taylor. Improved versions of assumed enhanced strain tri-linear elements for 3d finite deformation problems. *Computer Methods in Applied Mechanics and Engineering*, 110:359–386, 1993.
- [67] J. C. Simo and M. S. Rifai. A class of mixed assumed strain methods and the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 29:1595–1638, 1990.
- [68] Friedrich Stummel. The generalized patch test. *SIAM Journal on Numerical Analysis*, 16(3):449–471, 1979.
- [69] Friedrich Stummel. The limitations of the patch test. *International Journal for Numerical Methods in Engineering*, 16:177–188, 1980.
- [70] N. Sukumar. Construction of polygonal interpolants: a maximum entropy approach. *International Journal for Numerical Methods in Engineering*, 61:2159–2181, 2004.
- [71] Manil Suri. On the robustness of the h - and p -versions of the finite-element method. *Journal of Computational and Applied Mathematics*, 35:303–310, 1991.
- [72] Theodore Sussman and Klaus-Jürgen Bathe. Spurious modes in geometrically non-linear small displacement finite elements with incompatible modes. *Computers & Structures*, 140:14–22, 2014.
- [73] Cameron Talischi and Glaucio H. Paulino. Addressing integration error for polygonal finite elements through polynomial projections: A patch test connection. *Mathematical Models and Methods in Applied Sciences*, 24:1701–1727, 2014.
- [74] Cameron Talischi, Glaucio H. Paulino, Anderson Pereira, and Ivan F. M. Menezes. Polymesher: a general-purpose mesh generator for polygonal elements written in matlab. *Structural and Multidisciplinary Optimization*, 45(3):309–328, Mar 2012.
- [75] Cameron Talischi, Anderson Pereira, Ivan F. M. Menezes, and Glaucio H. Paulino. Gradient correction for polygonal and polyhedral finite elements. *International Journal for Numerical Methods in Engineering*, 102:728–747, 2015.
- [76] R. L. Taylor, J. C. Simo, O. C. Zienkiewicz, and A. C. H. Chan. The patch test condition for assessing fem convergence. *International Journal for Numerical Methods in Engineering*, 22(1):39–62, 1986.
- [77] Eugene L. Wachspress. *A Rational Finite Element Basis*. Academic Press, 1975.
- [78] J. Wang and X. Ye. A weak galerkin finite element method for second-order elliptic problems. *Journal of Computational and Applied Mathematics*, 241:103–115, 2013.

- [79] X. Wang, N.S. Malluwawadu, F. Gao, and T.C. McMillan. A modified weak galerkin finite element method. *Journal of Computational and Applied Mathematics*, 271:319–329, 2014.
- [80] Wikiversity. Introduction to elasticity/plate with hole in tension, 2017.
- [81] E.L. Wilson, R. Taylor, W.P. Doherty, and Jamshid Ghaboussi. *Incompatible Displacement Models*. 12 1973.
- [82] Robert Winkler. Comments on membrane locking. *Proceedings in Applied Mathematics and Mechanics*, 10:229–230, 2010.
- [83] Steven R. Wopschall. *A Novel Face-on-Face Contact Method for Nonlinear Solid Mechanics*. PhD thesis, 2017.