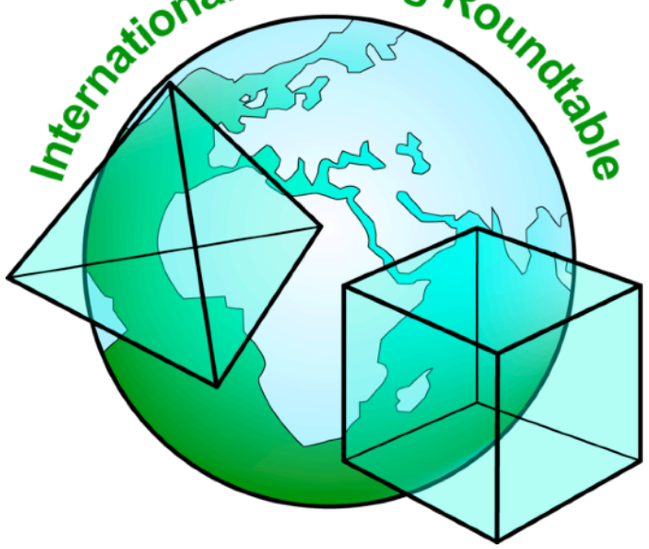
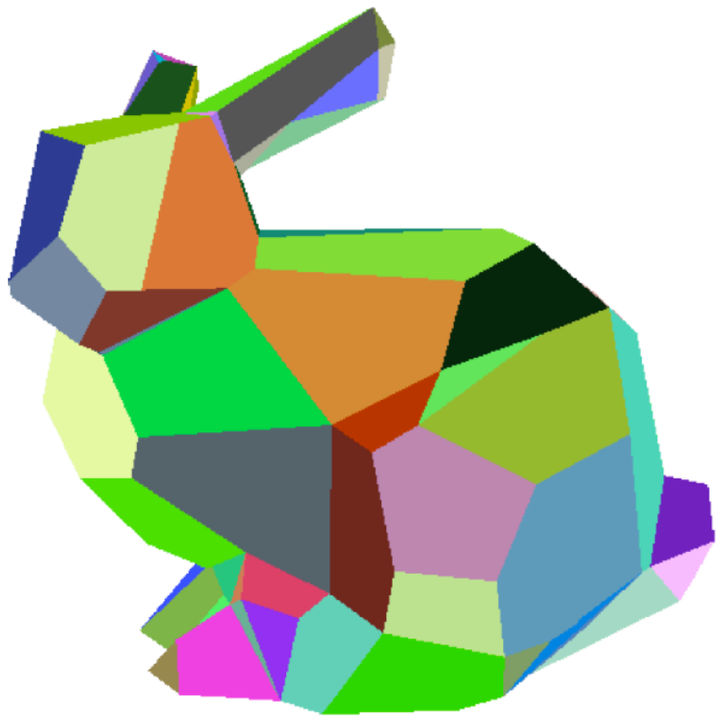


A Short Course on

High-order polytopal methods

Joaquim Peiró

Imperial College London



Objectives of the course

- To give a brief overview of current low-order polyhedral mesh generation and PDE discretization methods.
- To introduce high-order polytopal discretization methods.
- To identify geometric requirements for the generation of suitable high-order polytopal meshes.
- To show how current *a posteriori* curvilinear mesh generation methods can be adapted to the generation of such meshes.

Landesgartenschau Exhibition Hall (University of Stuttgart)

Table of contents

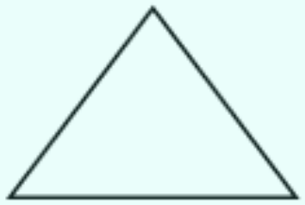
Topics this course will cover:

1. **What are polytopal meshes and why use them?**
2. **Low-order discretization methods in polytopal meshes:**
    a. Finite volume methods (FVM)
    b. Finite element methods (FEM)
    c. Polytopal mesh generation and optimization
3. **High-order discretization methods in polytopal meshes:**
    a. High-order finite volume methods (HOFVM)
    b. High-order hybrid methods (HOHM)
    c. Virtual element methods (VEM)
    d. *A posteriori* high-order polytopal mesh generation

Politopal elements

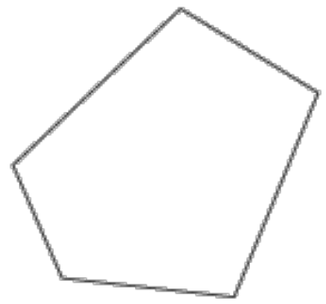
2D

Triangle

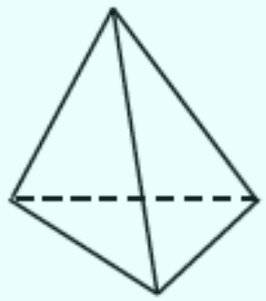


Quadrilateral

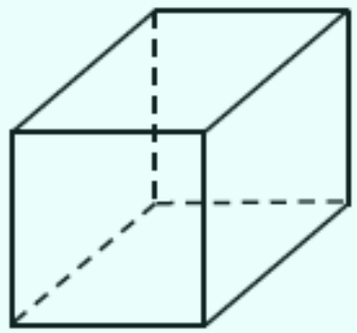
Polygon



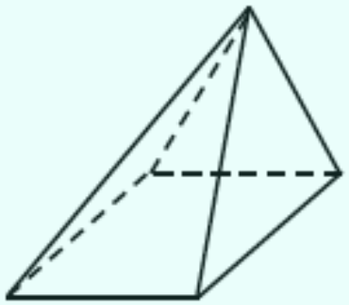
3D



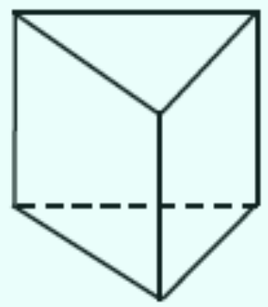
Tetrahedron



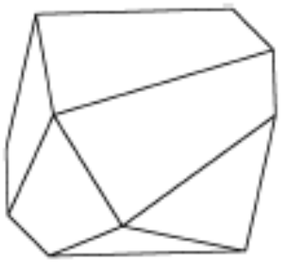
Hexaedron



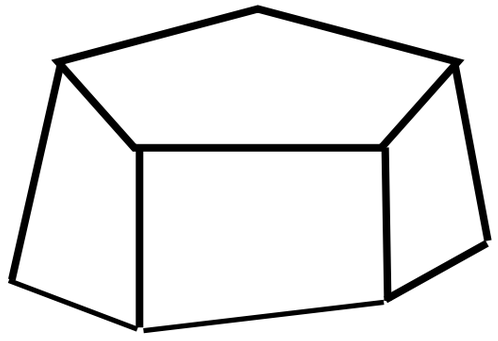
Square pyramid



Triangular prism or “wedge”



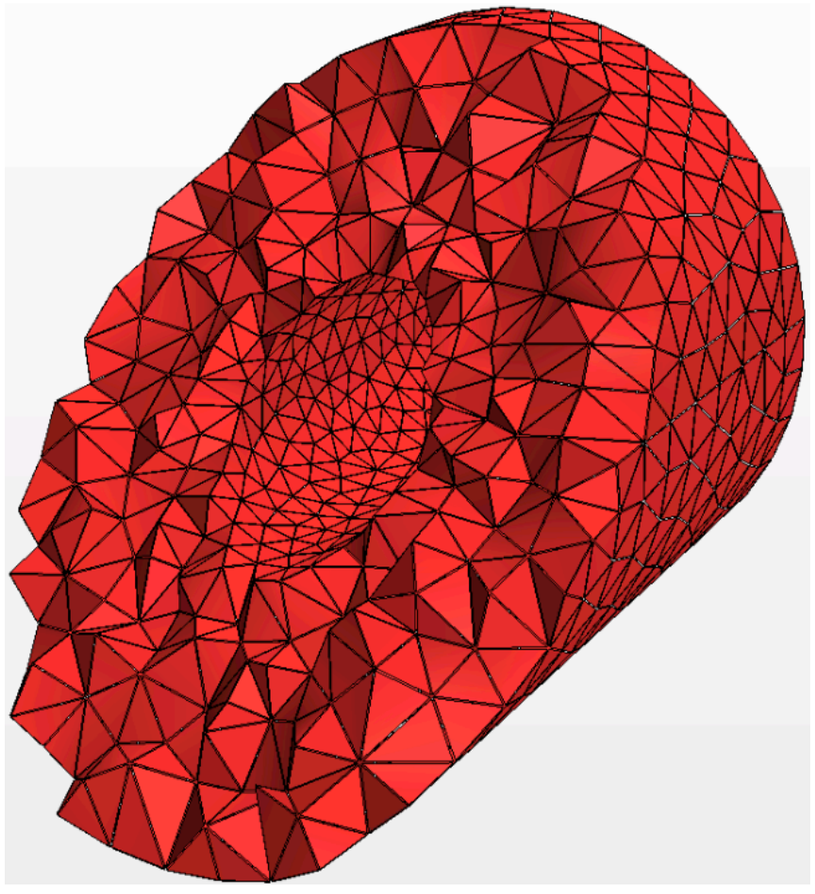
Polyhedron



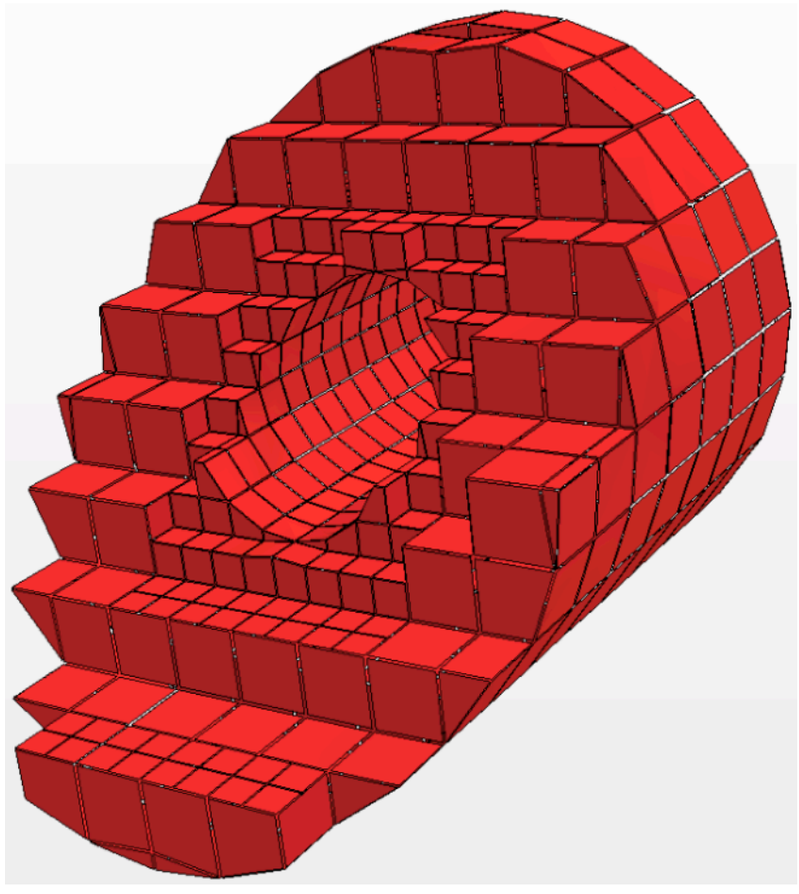
Polygonal Prism

Polytopal meshes are spatial partitions of a domain into non-overlapping elements, polygons in 2D and polyhedra in 3D (not necessarily regular), that share common sides and faces.

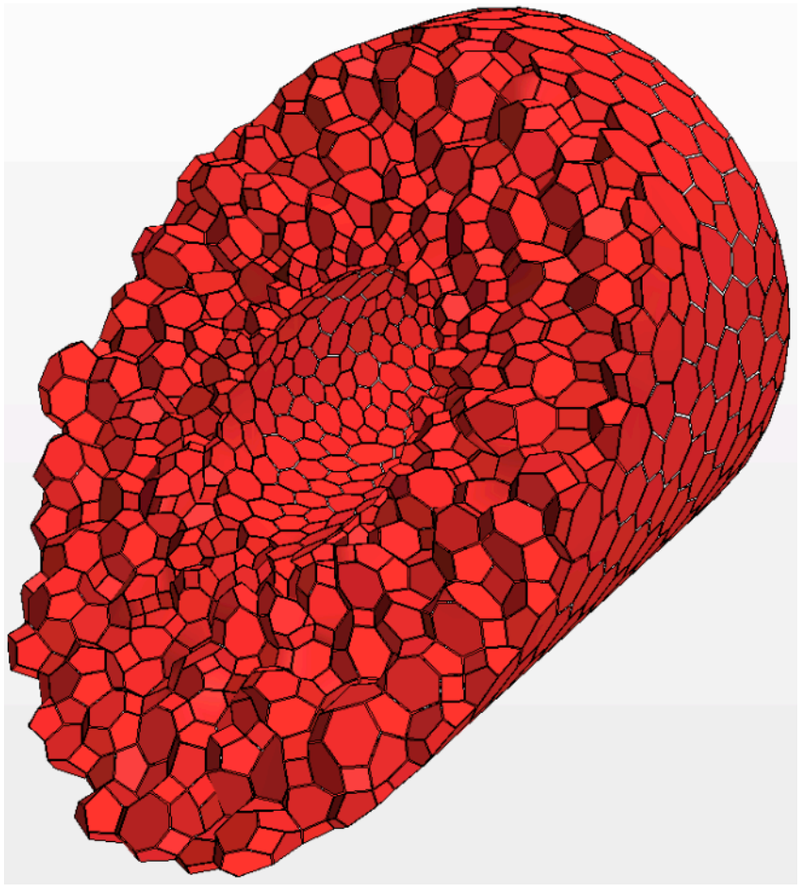
Unstructured mesh typologies



Tetrahedral



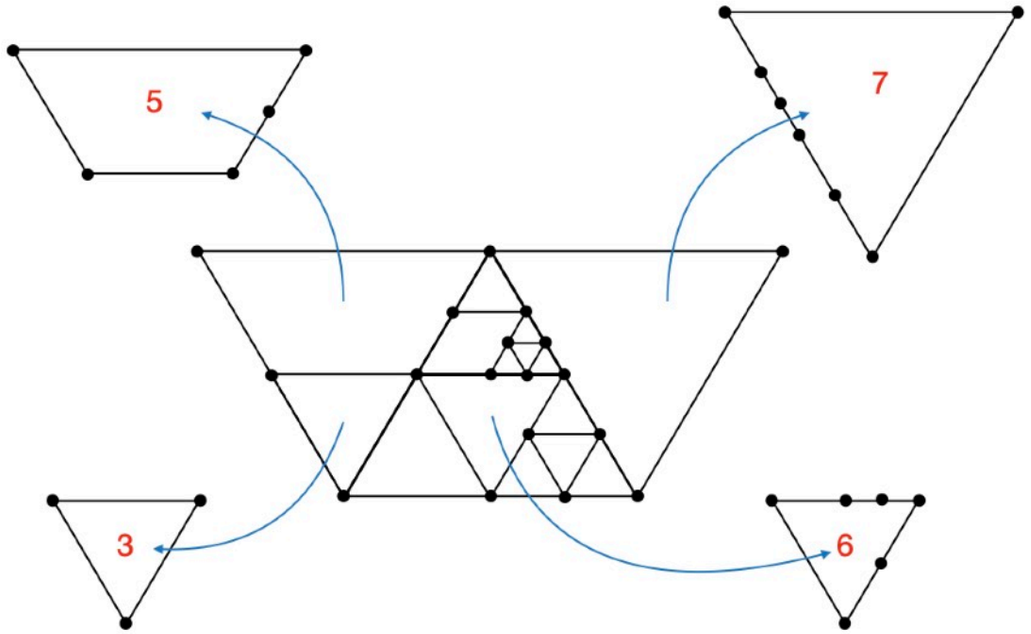
Hex-dominant



Polyhedral

Why polytopal meshes?

- Routinely used in second-order finite volume methods for CFD.
- General: encompass conventional cells, e.g. Tet, Hex.
- Handle complex geometries: distorted conventional meshes are usual.
- Facilitate automatic mesh generation.
- More d.o.f. per element means fewer elements.
- Robustness to mesh distortion.
- Ease coarsening by agglomeration (multigrid).
- Easier/better meshing of domain features.
- Generic shapes facilitate adaptive refinement.
- Automatic inclusion of “hanging nodes”.



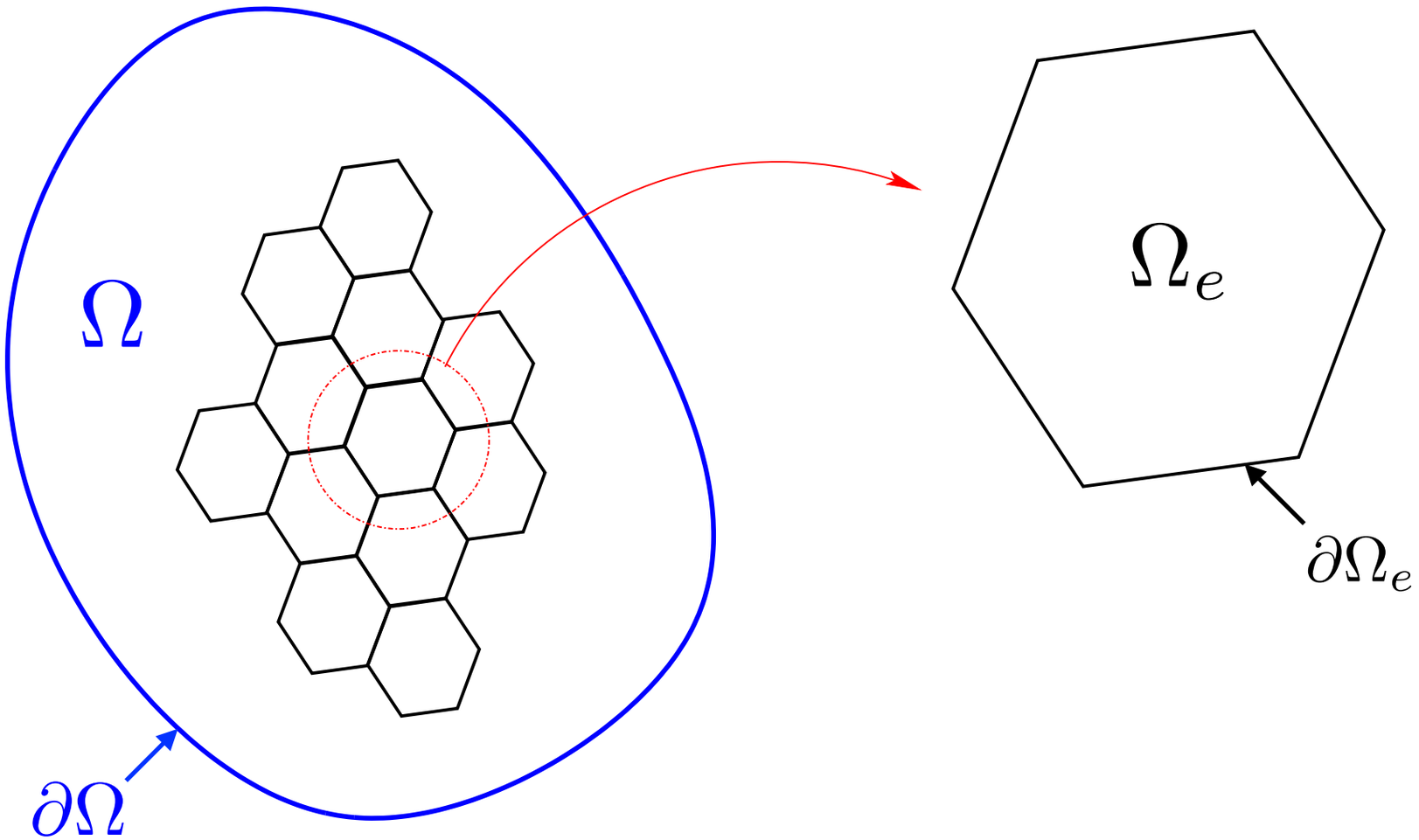
LOW-ORDER DISCRETIZATION METHODS

a. Finite volume methods (FVM)
b. Finite element methods (FEM)
c. Polytopal mesh generation and optimization

Model PDE: Poisson's equation

$$-\nabla^2 u = f \quad \text{in } \Omega$$

$$u = 0 \quad \text{in } \partial\Omega$$



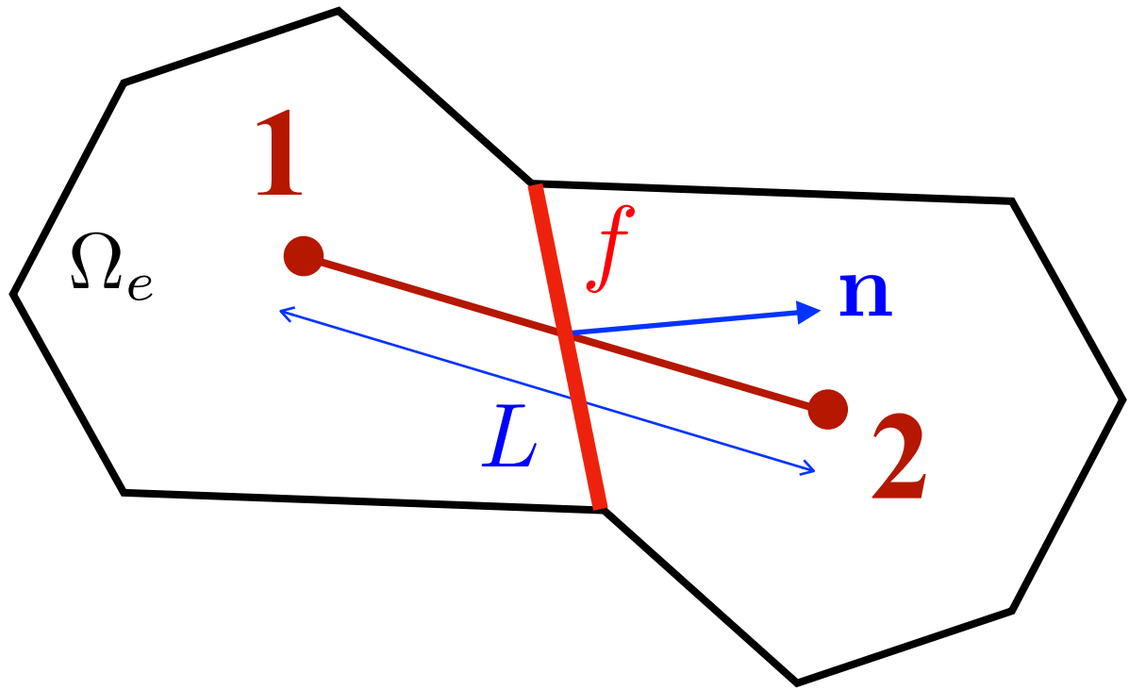
Finite volume method (FVM)

Integral form of Poisson's equation

$$-\int_\Omega \nabla^2 u \, d\mathbf{x} = \int_\Omega f \, d\mathbf{x}$$

Apply Gauss' theorem

$$-\int_{\partial\Omega} \nabla u \cdot \mathbf{n} \, d\mathbf{x} = \int_\Omega f \, d\mathbf{x}$$



Numerical approximation

$$\int_{\partial\Omega_e} \nabla u \cdot \mathbf{n} \, d\mathbf{x} \approx \sum_f \frac{u_2 - u_1}{L} A_f$$

$$\int_{\Omega_e} f \, d\mathbf{x} \approx f_1 V_e$$

$$A_f = \int_{\partial\Omega_e} d\mathbf{x}$$

$$V_e = \int_{\Omega_e} d\mathbf{x}$$

Finite element method (FEM)

Weak form of Poisson's equation

$$-\int_\Omega \nabla^2 u\, v\, d\mathbf{x} = \int_\Omega f\, v\, d\mathbf{x} \qquad v = 0 \text{ in } \partial\Omega$$

Apply Gauss' theorem

$$\int_\Omega \nabla u \cdot \nabla v\, d\mathbf{x} - \underbrace{\int_{\partial\Omega} v\, \nabla u \cdot \mathbf{n}}_{=0} = \int_\Omega f\, v\, d\mathbf{x}$$

Numerical approximation $u \approx u_h \in V_h$

$$\int_\Omega \nabla u_h \cdot \nabla v_h\, d\mathbf{x} = \int_\Omega f v_h\, d\mathbf{x} \qquad v_h \in V_h$$

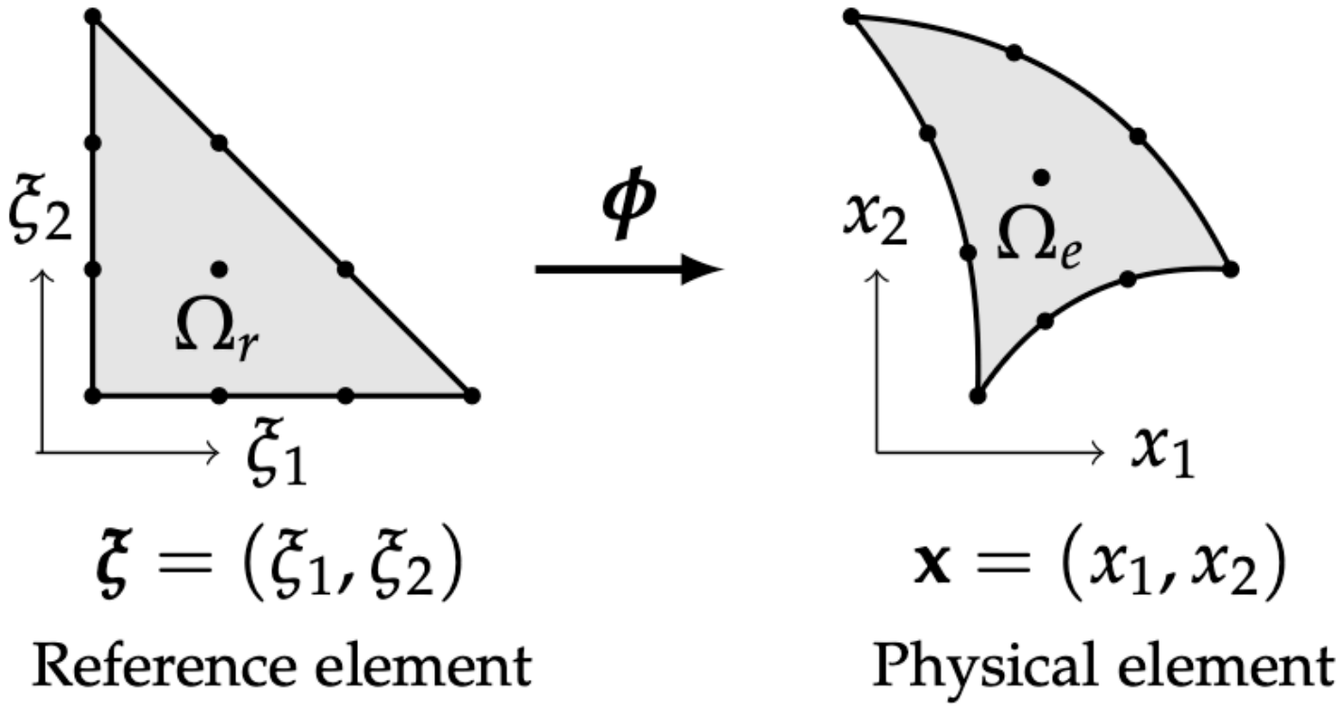
Chose a basis $\{\varphi_1(\mathbf{x}), \ldots, \varphi_N(\mathbf{x})\}$ in V_h $\longrightarrow$ $u_h = \sum_{i=1}^{N} u_i \varphi_i$ Shape functions

$$\sum_{i=1}^{N} u_i \int_\Omega \nabla \varphi_i \cdot \nabla \varphi_j\, d\mathbf{x} = \int_\Omega f \varphi_j\, d\mathbf{x} \quad j = 1, \ldots, N \quad \equiv \quad \mathbf{KU} = \mathbf{F}$$

Linear system of equations

Reference-to-physical mappings

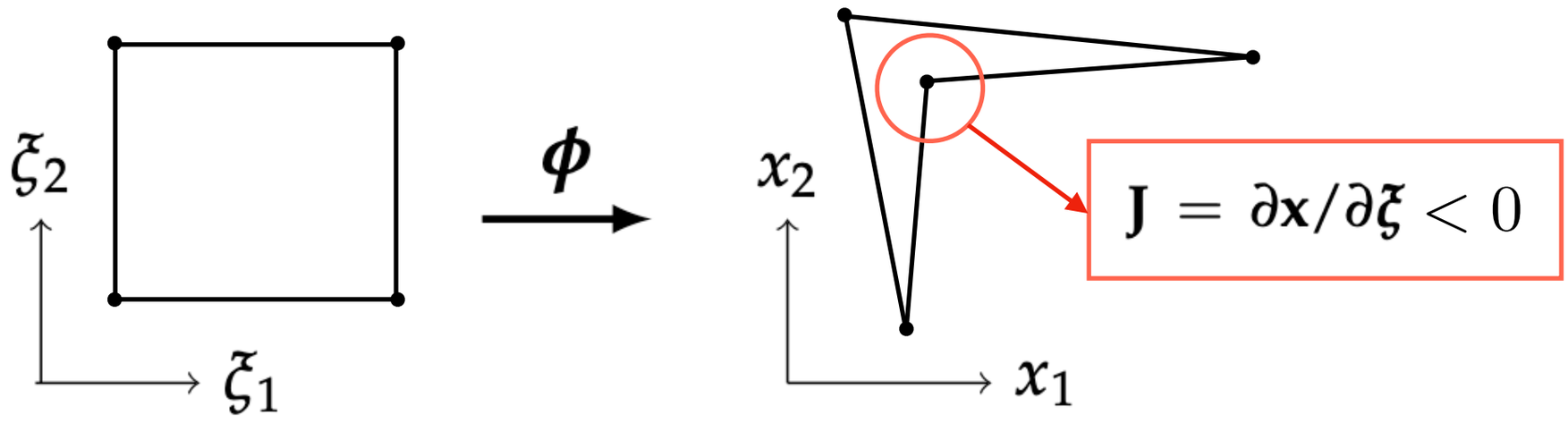
To have integration quadrature rules that are independent of element geometry, we introduce a mapping $\boldsymbol{\phi} : \Omega_r \rightarrow \Omega_e$

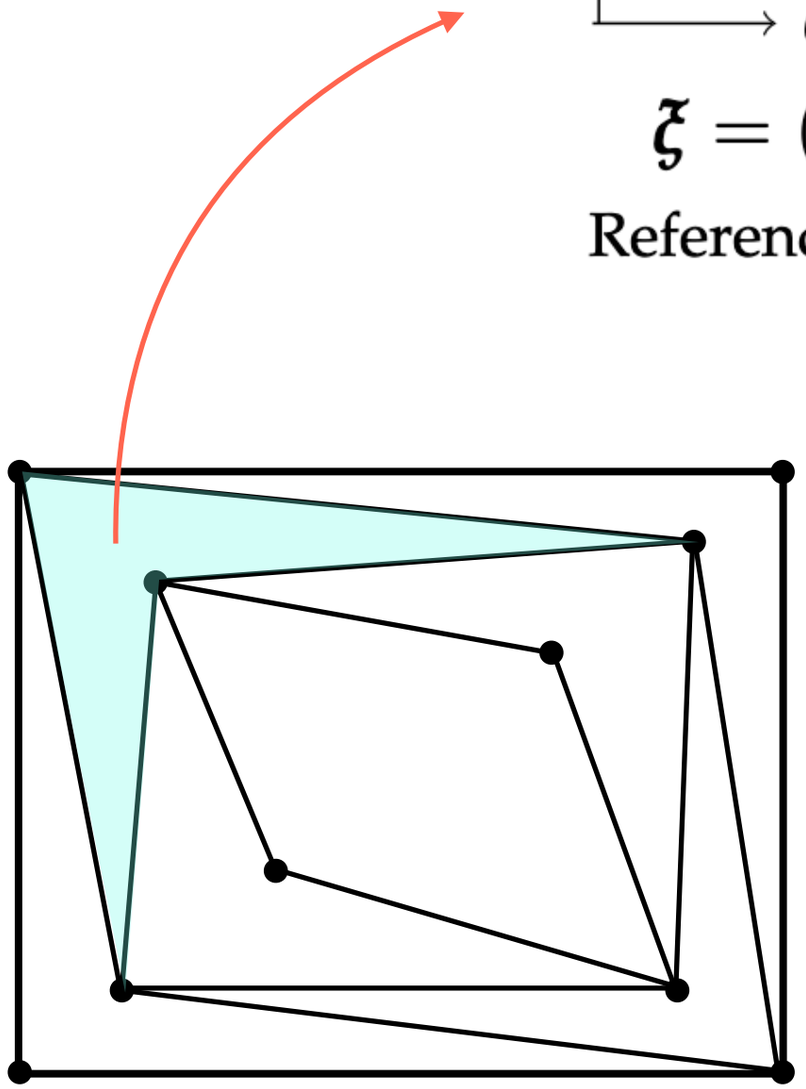


Integration in the reference element

$$\int_{\Omega_e} f(\mathbf{x}) \varphi_j^e(\mathbf{x})\, d\mathbf{x} = \int_{\Omega_r} f(\mathbf{x}(\boldsymbol{\xi}))\, \varphi_j^e(\mathbf{x}(\boldsymbol{\xi}))\, \underbrace{\det(\mathbf{J})\, d\boldsymbol{\xi}}_{\mathbf{J} = \partial \mathbf{x}/\partial \boldsymbol{\xi}}$$

Distorted/invalid elements





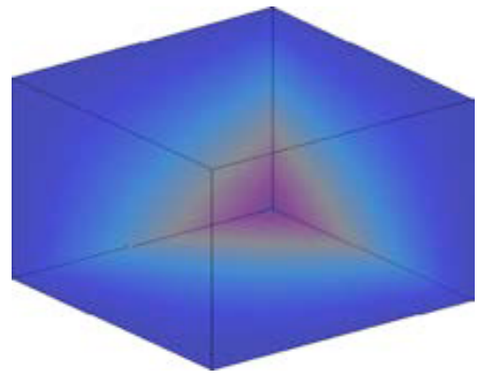
Distortions in the mapping may lead to inaccuracies, even invalidity, of the numerical approximation.

Note: HHO and VEM are more tolerant as they do not use the mapping.

Polytopal shape functions: Barycentric coordinates

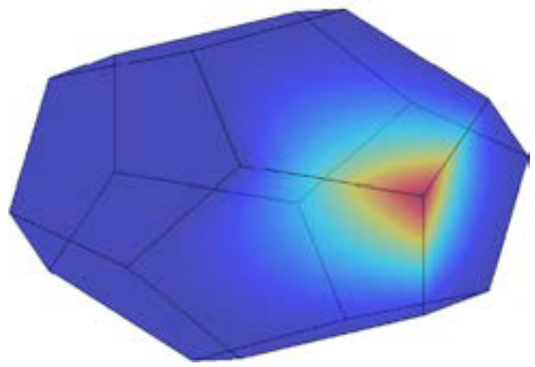
Generalized barycentric coordinates are defined by a set of weights $w_i(\mathbf{x});\ i = 1, \ldots, n$, to form shape functions:

$$\varphi_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum_{j=1}^{n} w_j(\mathbf{x})}$$



This bases should be a *partition of unity*

$$\sum_{i=1}^{n} \varphi_i(\mathbf{x}) = 1$$



For a polytope of n vertices with coordinates $\mathbf{x}_i;\ \ i = 1, \ldots, n$ coordinates are an *affine combination* of the vertex coordinates

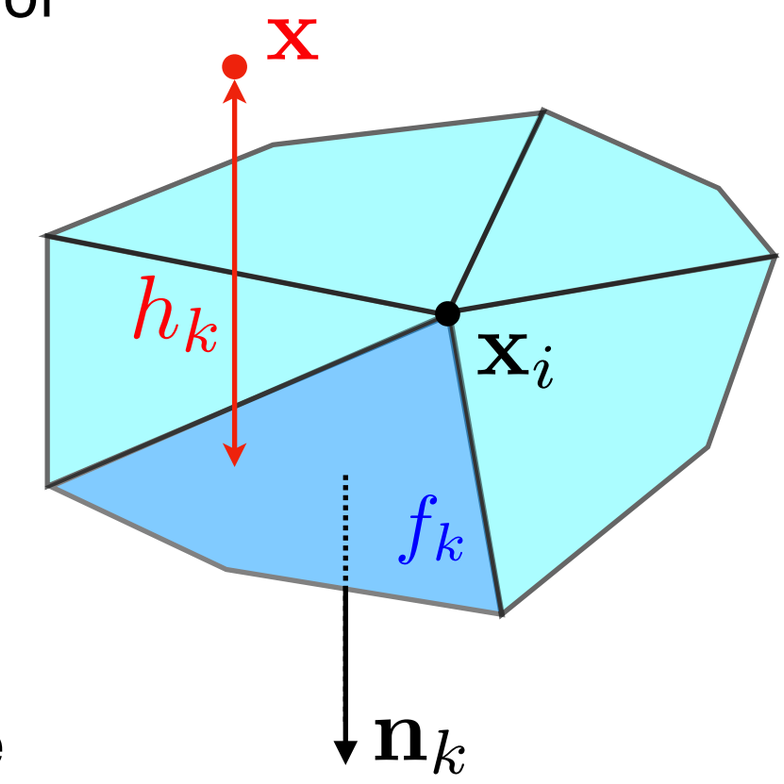
$$\mathbf{x} = \sum_{i=1}^{n} \mathbf{x}_i\, \varphi_i(\mathbf{x}) \qquad \text{with} \qquad \varphi_i(\mathbf{x}_j) = \delta_{ij}$$

If $\varphi(\mathbf{x}) > 0$ the interpolant $u_h(\mathbf{x}) = \sum_{i=1}^{N} u_i \varphi_i(\mathbf{x})$ is within the convex hull of the coefficients $\{u_1, \ldots, u_N\}$.

M.S. Floater. Generalized barycentric coordinates and applications. Acta Numerica, 24:161—-214, 2015.

Wachspress barycentric coordinates

Consider the K faces, $\{f_1, \ldots, f_K\}$, incident on a vertex of coordinates $\mathbf{x}_i$ with outer normal vectors $\{\mathbf{n}_1, \ldots, \mathbf{n}_K\}$.



Perpendicular distance of a point (coordinate $\mathbf{x}$) interior to the polytope to the face f_k is $h_k > 0$.

Denoting $\mathbf{p}_k(\mathbf{x}) := \mathbf{n}_k / h_k(\mathbf{x})$, the shape functions are

$$\varphi_j = \frac{w_{\mathbf{x}_j}}{\sum_{i=1}^{n} w_{\mathbf{x}_i}}; \qquad w_{\mathbf{x}_i} = \sum_{k=1}^{K-2} \det(\mathbf{p}_k, \mathbf{p}_{k+1}, \mathbf{p}_K)$$

Note: The values of the Wachspress shape functions are *strictly positive*, $\varphi(\mathbf{x}) > 0$, only for convex polytopes.

E. Wachspress, Rational Bases and Generalized Barycentrics, Springer (2016)

Barycentric coordinates for non-convex polytopes

Harmonic shape functions

Solutions of the Laplace's equation within the polytope Ω_e with suitable Dirichlet boundary conditions prescribed on the boundary

$$\nabla^2 \varphi_i(\mathbf{x}) = 0 \quad \text{in } \Omega_e; \qquad \varphi_i(\mathbf{x}) = g_i(\mathbf{x}) \quad \text{in } \partial\Omega_e$$

Here $g_i(\mathbf{x})$ is a piecewise linear function on the boundary with $g_i(\mathbf{x}_j) = \delta_{ij}$.

Maximum principle for Laplace's equation ensures $\varphi_i > 0$.

Maximum-entropy shape functions

Maxima of the "Shannon entropy" functional

$$\mathcal{L}(\varphi, \lambda_0, \boldsymbol{\lambda}) = -\sum_{i=1}^{n} \varphi_i(\mathbf{x}) \ln\left(\frac{\varphi_i(\mathbf{x})}{w_i(\mathbf{x})}\right)$$
$$- \lambda_0 \left(\sum_{i=1}^{n} \varphi_i(\mathbf{x}) - 1\right) - \boldsymbol{\lambda} \cdot \left(\sum_{i=1}^{n} \varphi_i(\mathbf{x})(\mathbf{x}_i - \mathbf{x})\right)$$

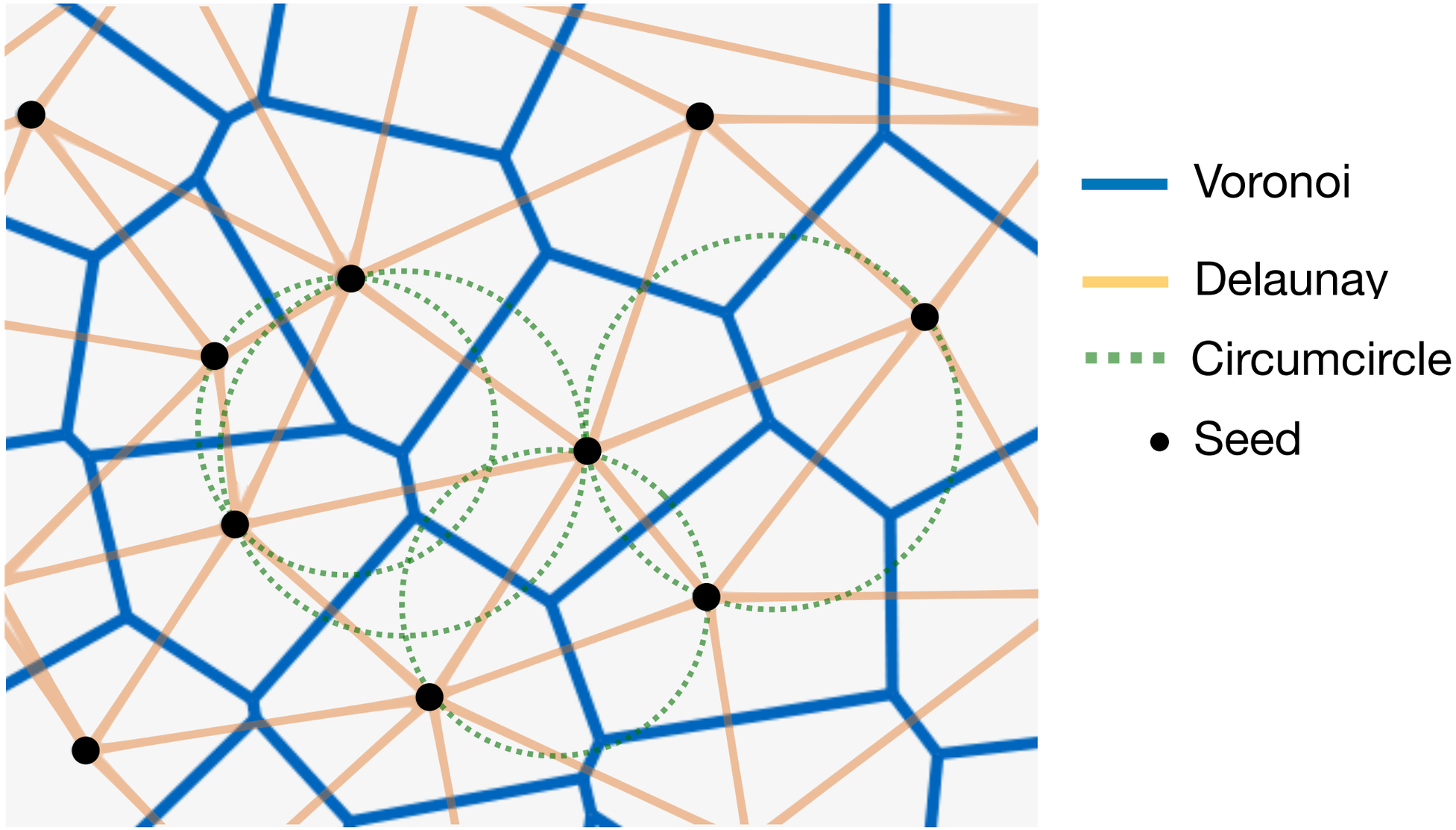
Lagrange multipliers used to impose the *partition of unity* and *affine combination* constraints

These functions are very expensive to evaluate!

POLYTOPAL MESH GENERATION

a. Voroni-based approach
b. Polytopal-dual meshes
c. Cut-cell meshes
d. Elemental agglomeration
e. Mesh quality and enhancement

Preliminaries: Delaunay tessellations and Voronoi cells

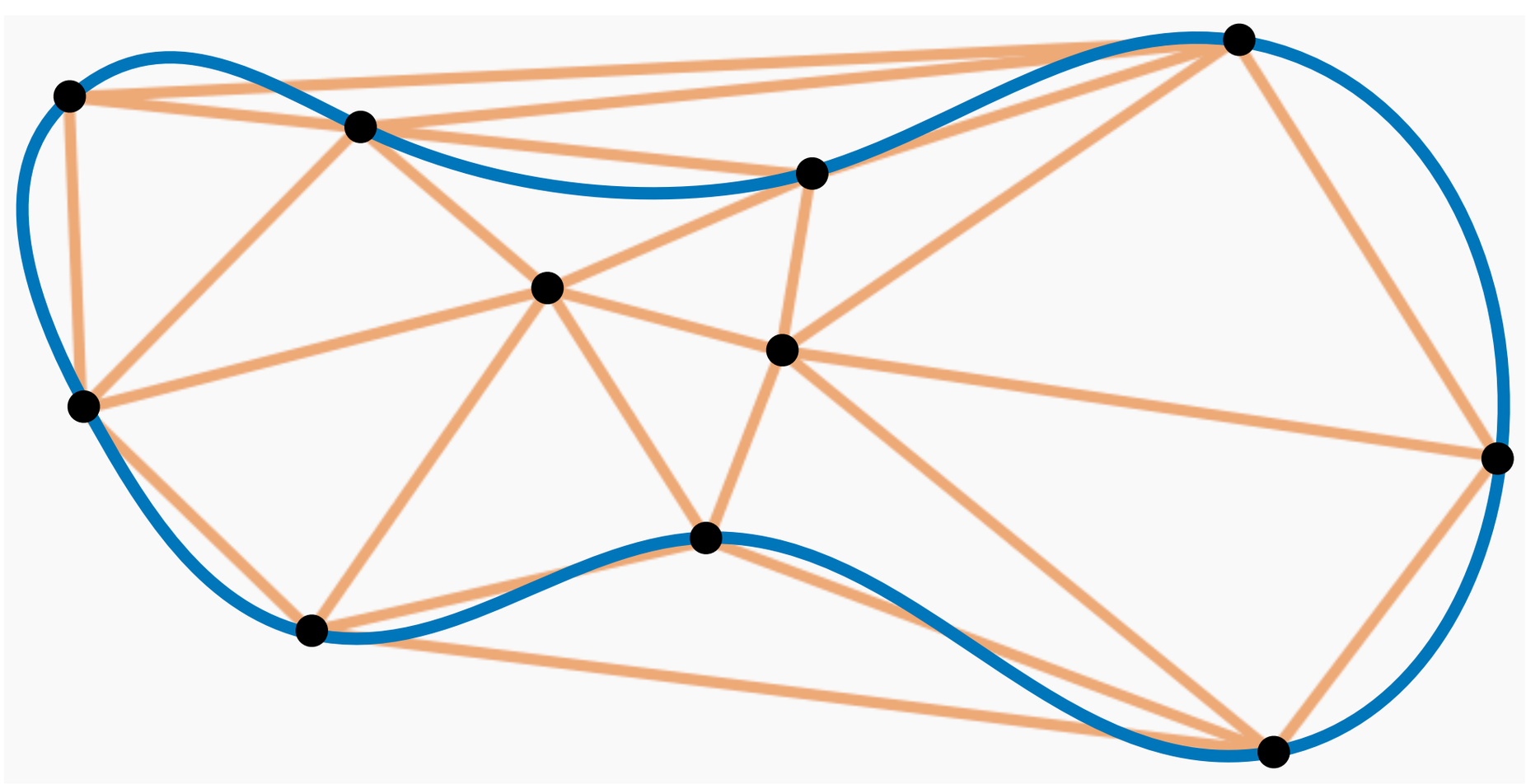


Voronoi cell: set of points closer to a seed than any other seeds. Convex cells by construction!

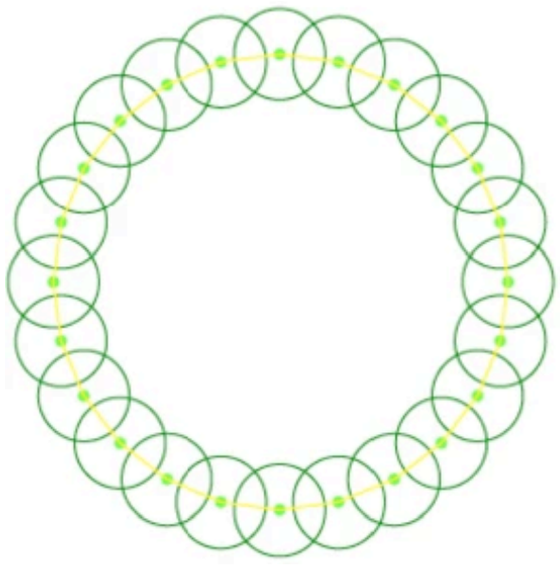
Circumcircle property: a circle passing through three seeds does not contain any other seeds in the set.

Caveat: Boundary-conforming meshes

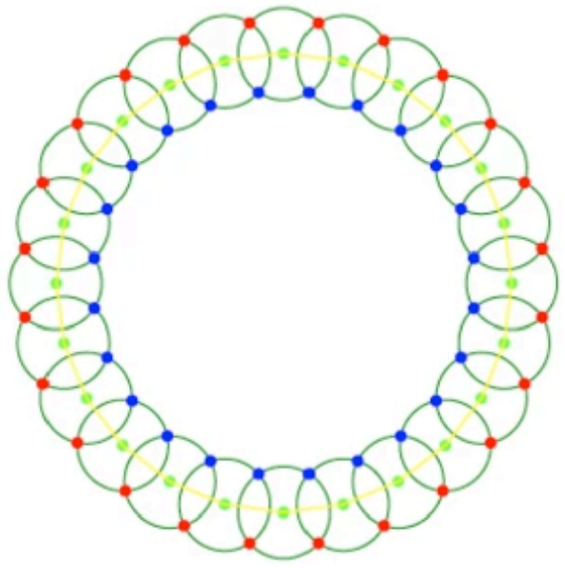
- A Delaunay tessellation meshes the convex hull of the set of seeds.
- It does not preserve the boundary of non-convex domains!
- Boundary-recovery procedures required for non-convex domains.



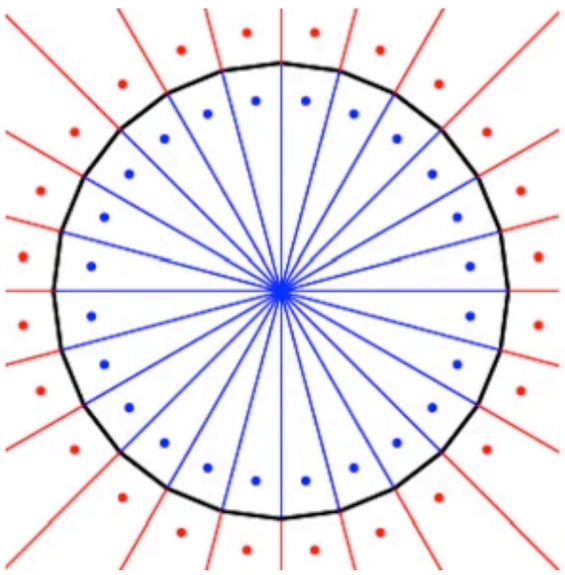
Voronoi-based approach: VoroCrust



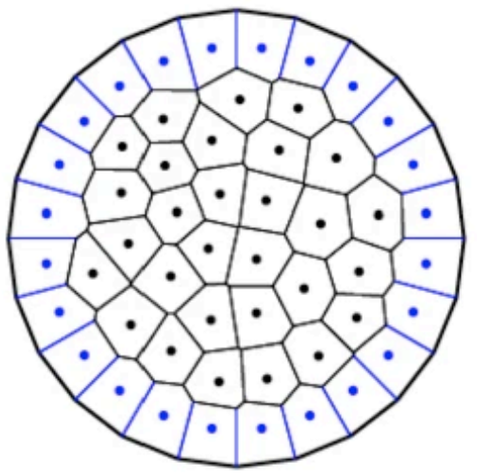
1. Cover boundary with overlapping circles



2. Place seeds on the circle both sides of the boundary (at their intersection)

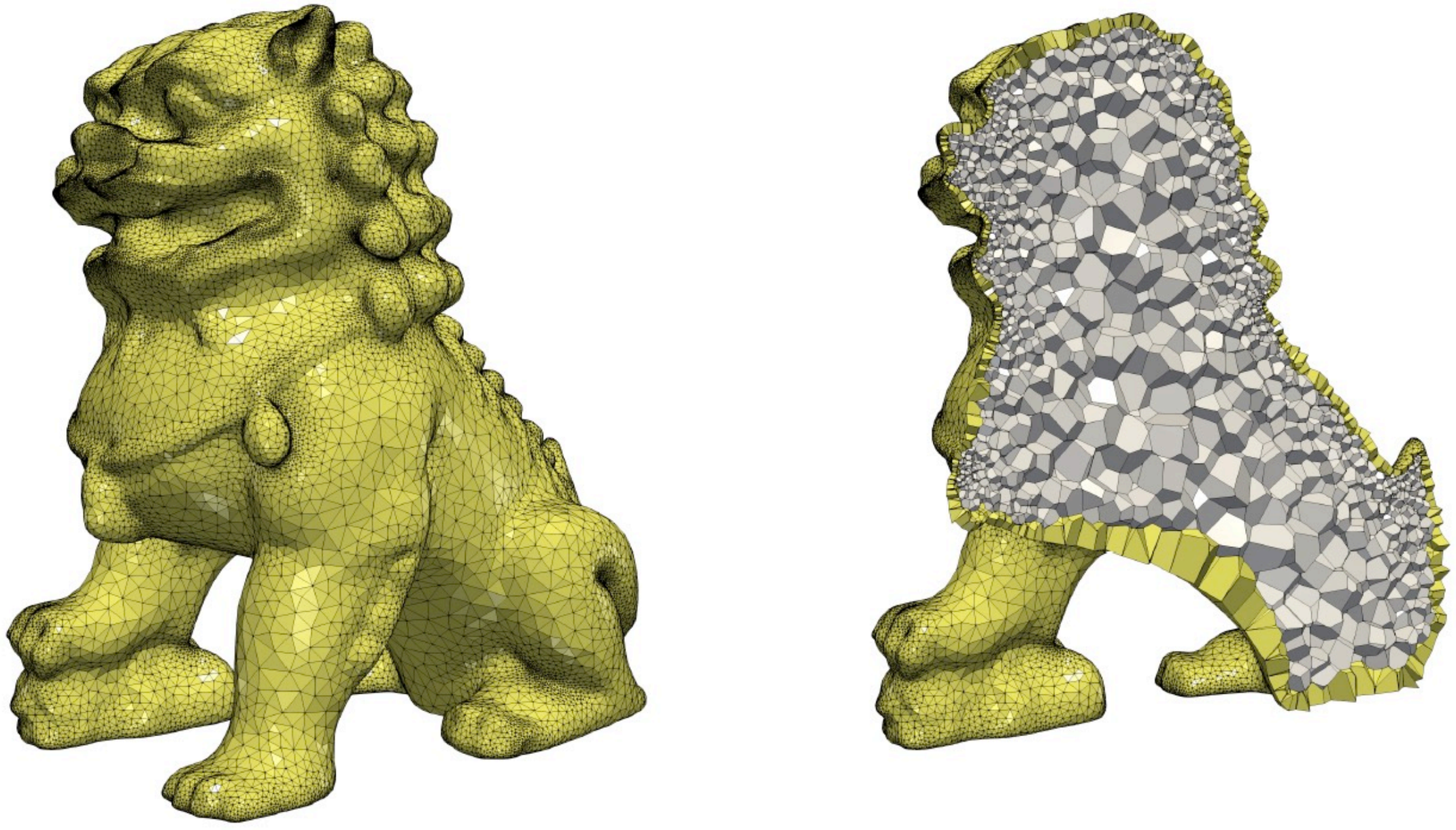


3. Generate Voronoi cells from boundary seeds



4. Seed interior points and generate the final Voronoi mesh

VoroCrust mesh example



VoroCrust mesh of a Chinese dragon

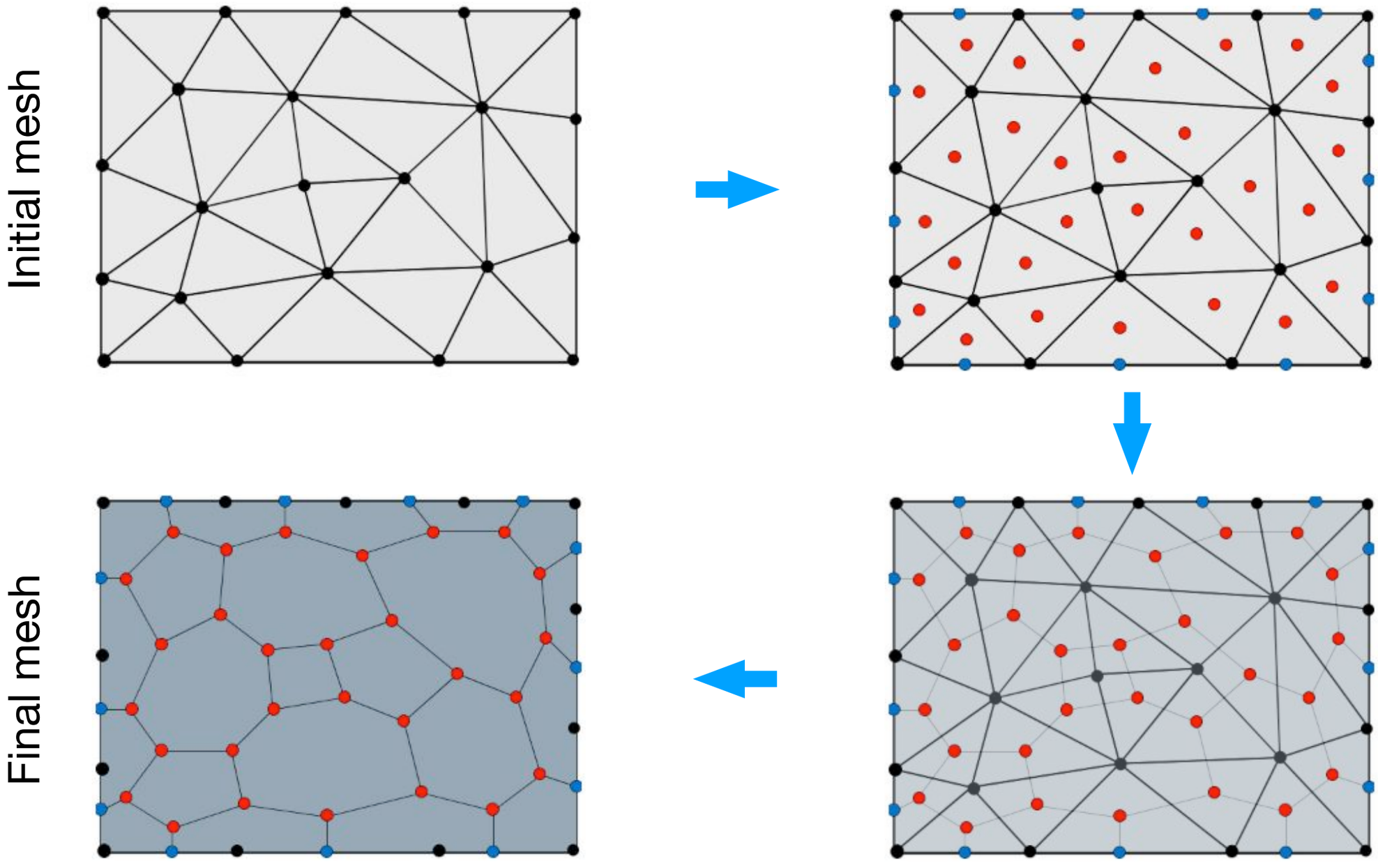
A. Abdelkader, C.L. Bajaj, M.S. Ebeida, A.H. Mahmoud, S.A. Mitchell, J.D. Owens, and A.A. Rushdi, “VoroCrust: Voronoi meshing without clipping”, ACM Trans Graph., **39**(3):1–40, 2020.

Polytopal-dual meshing

1. Subdivide elements

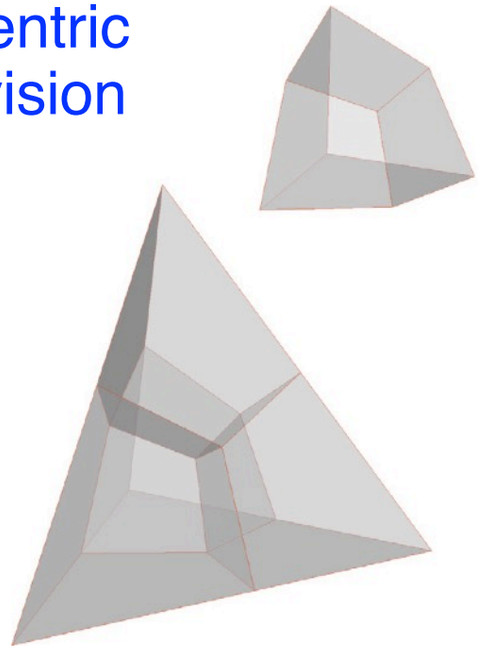
2. Link neighbours

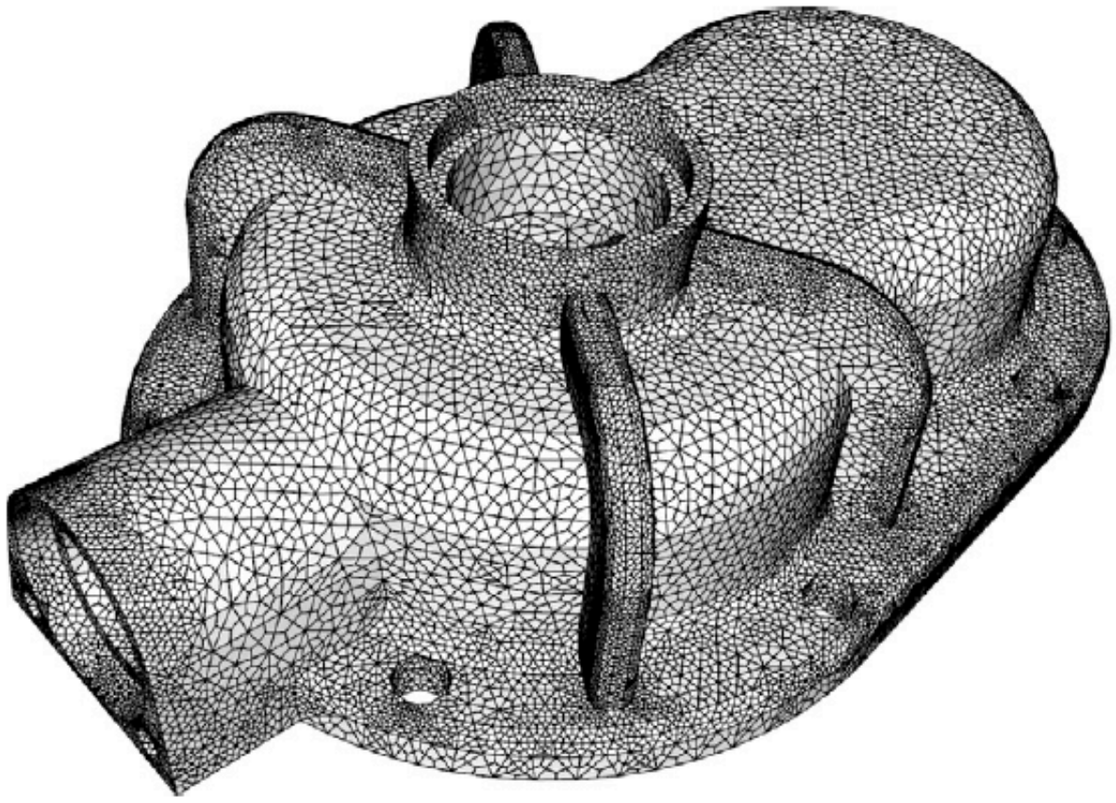
3. Create elements



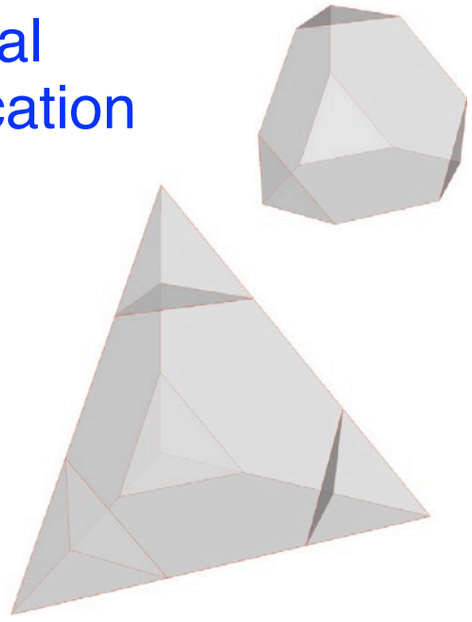
Polytopal-dual mesh example

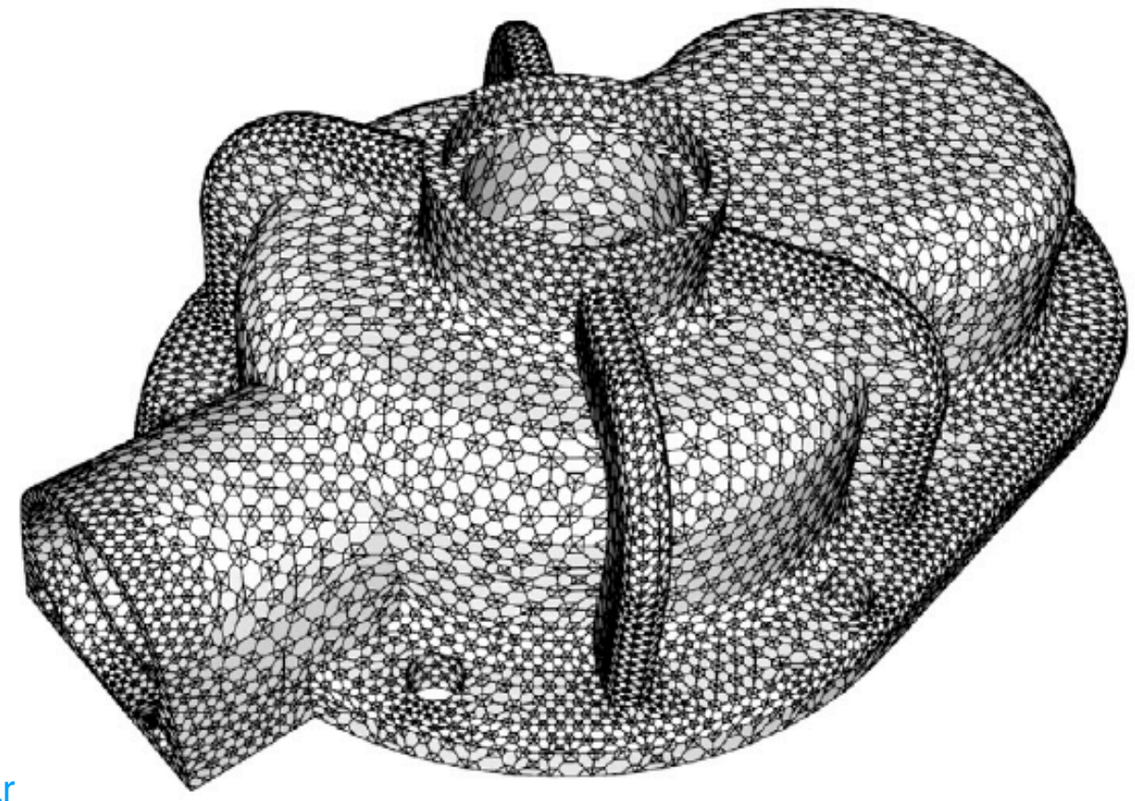
Barycentric subdivision





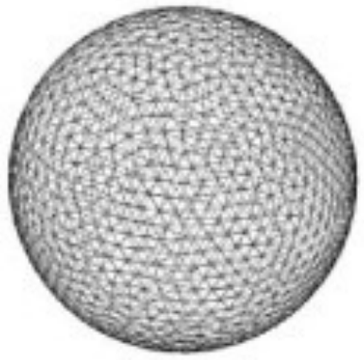
Partial truncation



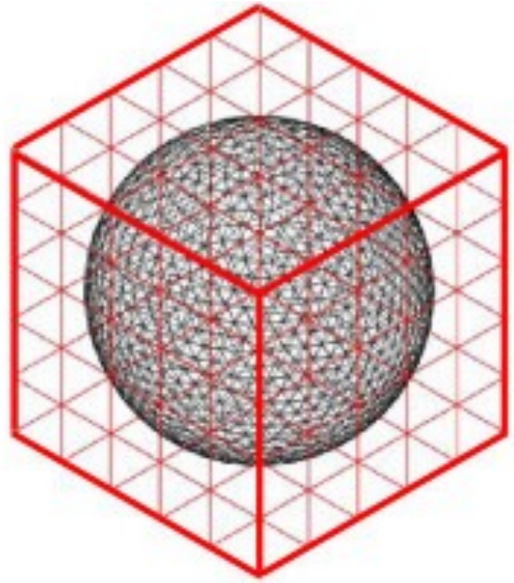


J.E. Bishop and N. Sukumar, “Polyhedral finite elements for nonlinear solid mechanics using tetrahedral subdivisions and dual-cell aggregation”, Computer Aided Geometric Design **77**, 101812 (2020)

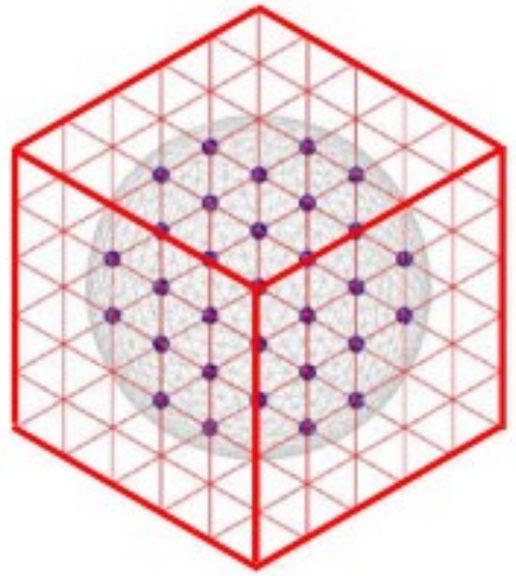
Cut-cell mesh generation workflow



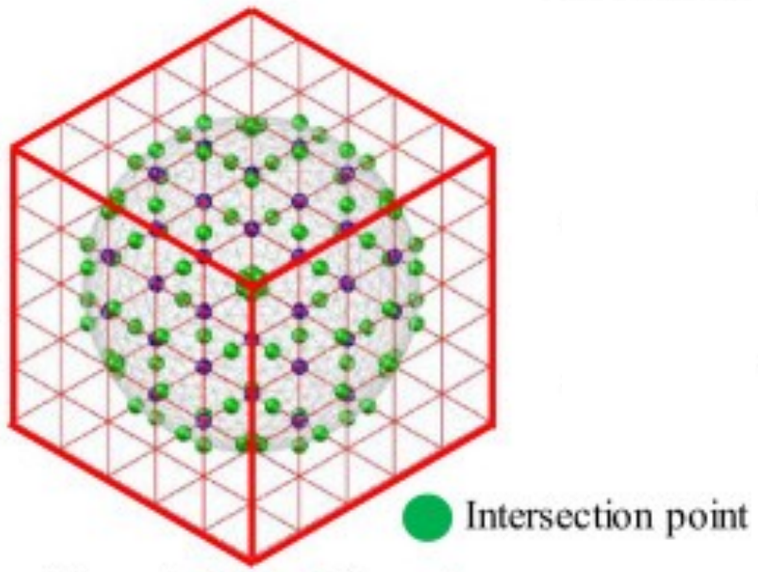
Surface information of 3D target geometry



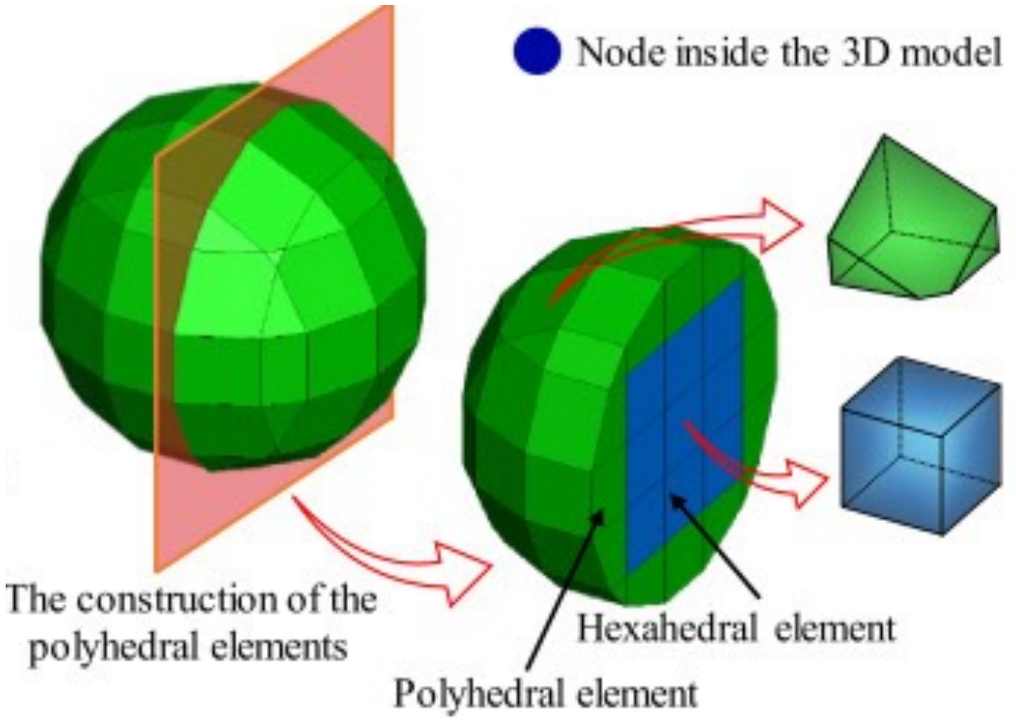
The constitution of background hexahedral element



The categorization of the state of the node inside the 3D target geometry



The calculating intersection point between 3D geometry and background element

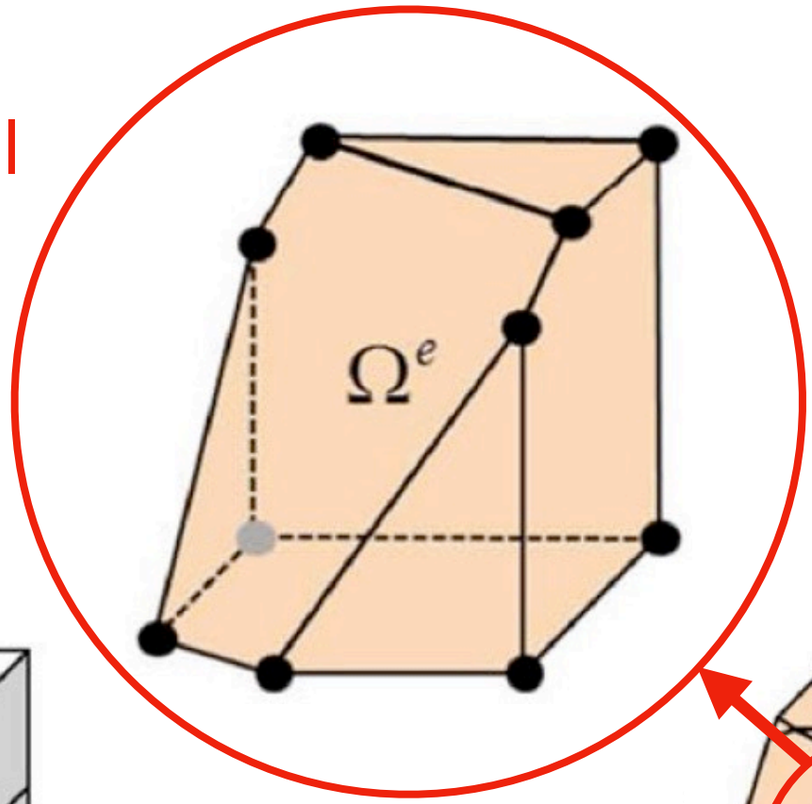


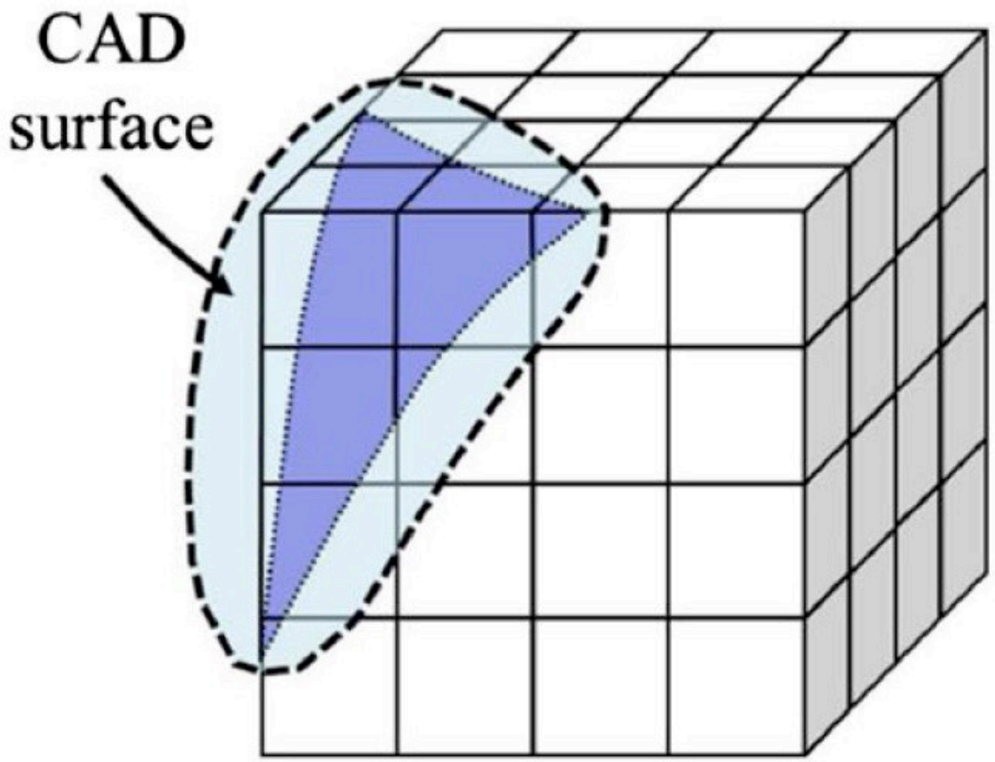
S. Kim, D. Sohn, and S. Im, “Construction of polyhedral finite element meshes based upon marching cube algorithm”, Advances in Engineering Software, **128**:98–112 (2019)

Cut-cell mesh at boundaries

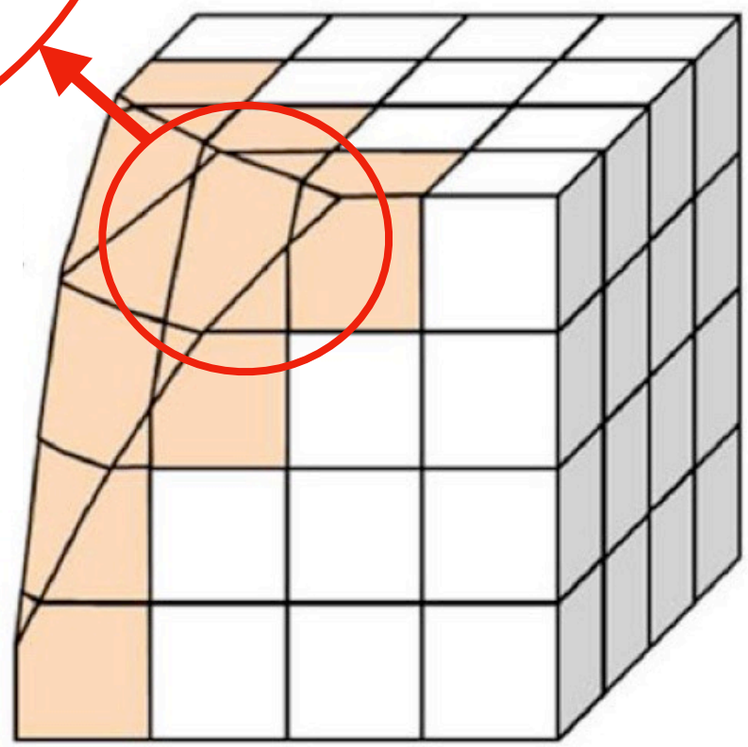
The shape of cells at the boundary is unrestricted thus allowing more flexibility, but avoid small lengths!

Cut-cell

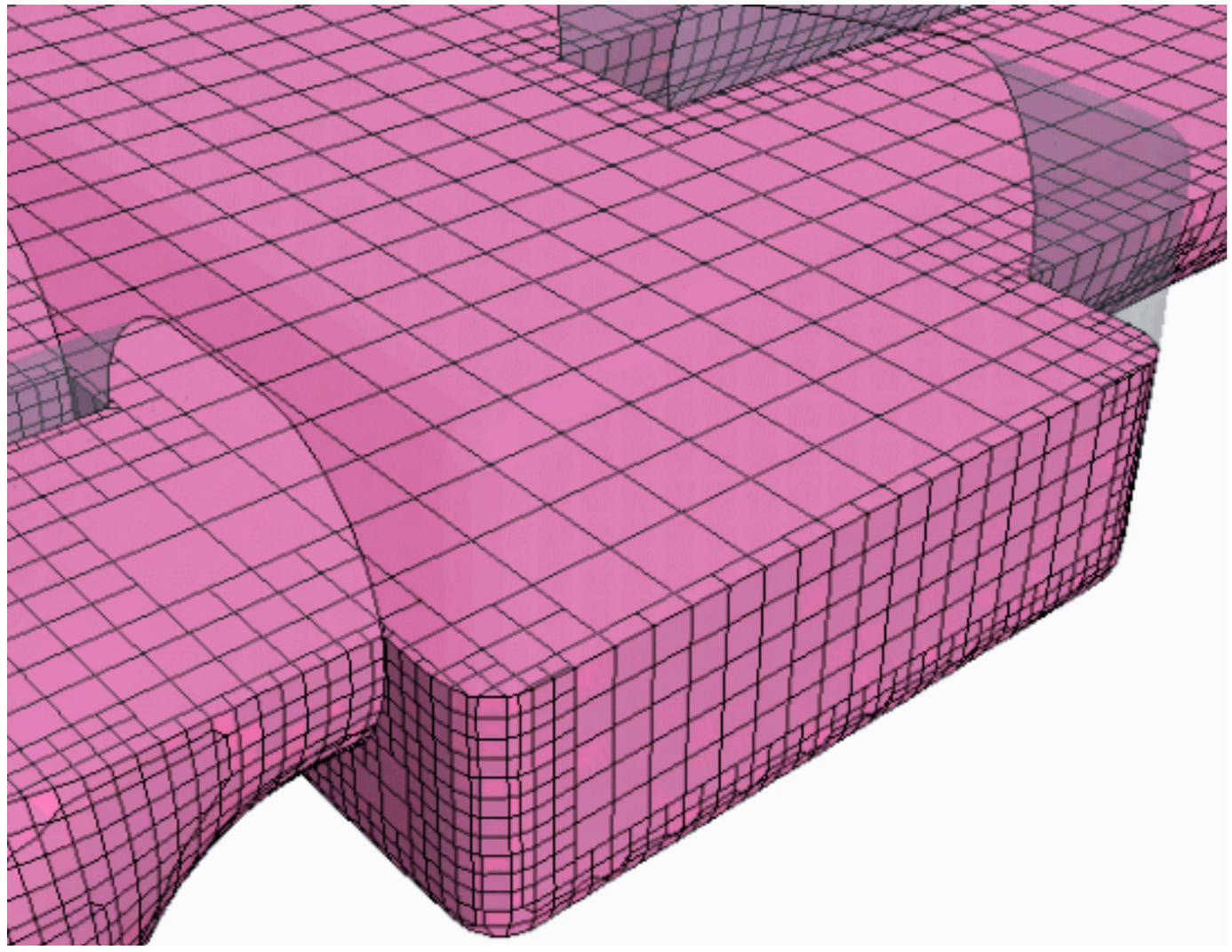




Trim elements



Cut-cell mesh example

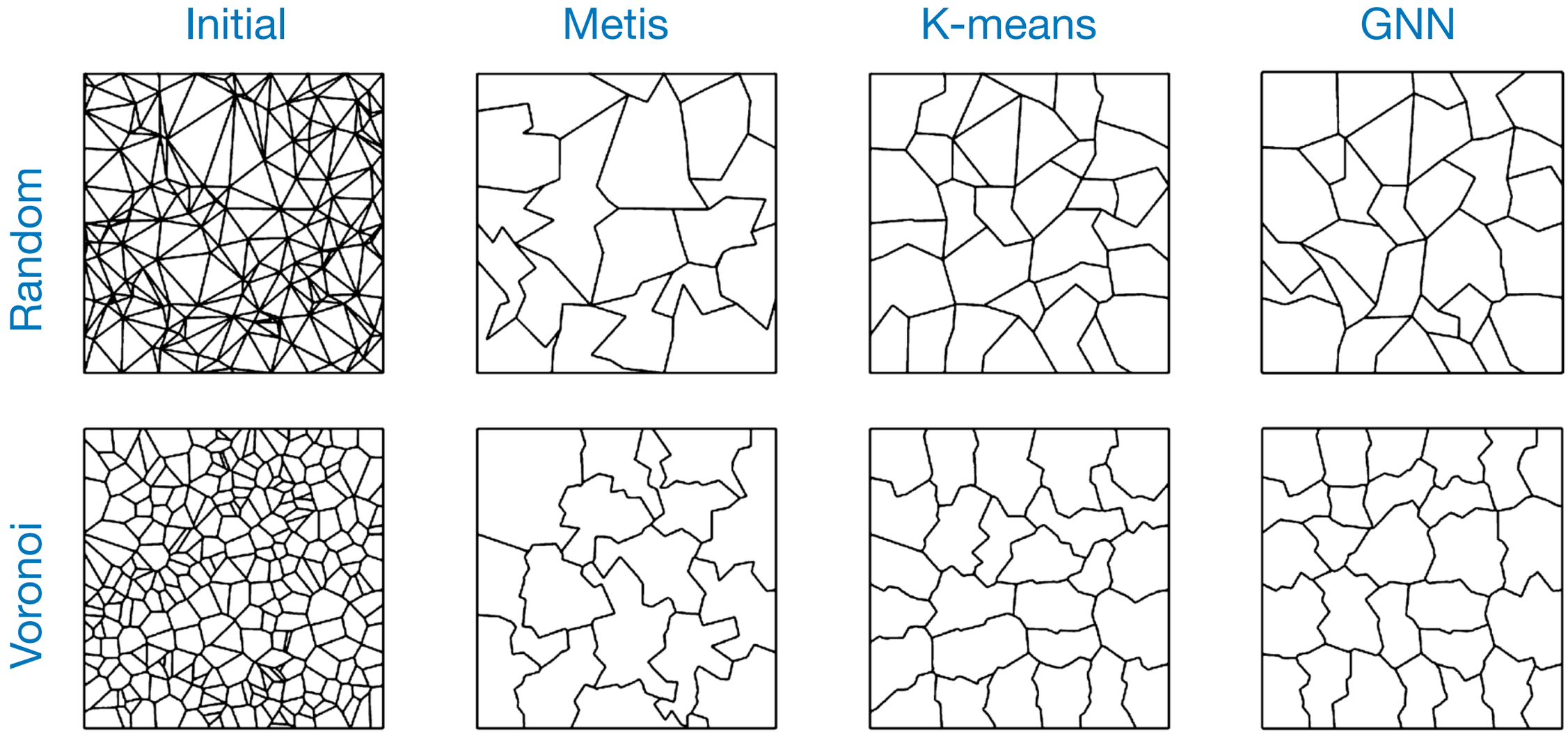


Cut-cell mesh obtained from a background octree mesh (STAR-CCM+)

Elemental agglomeration

Elements, or subdivisions of elements, of a conventional mesh are aggregated into polytopes.

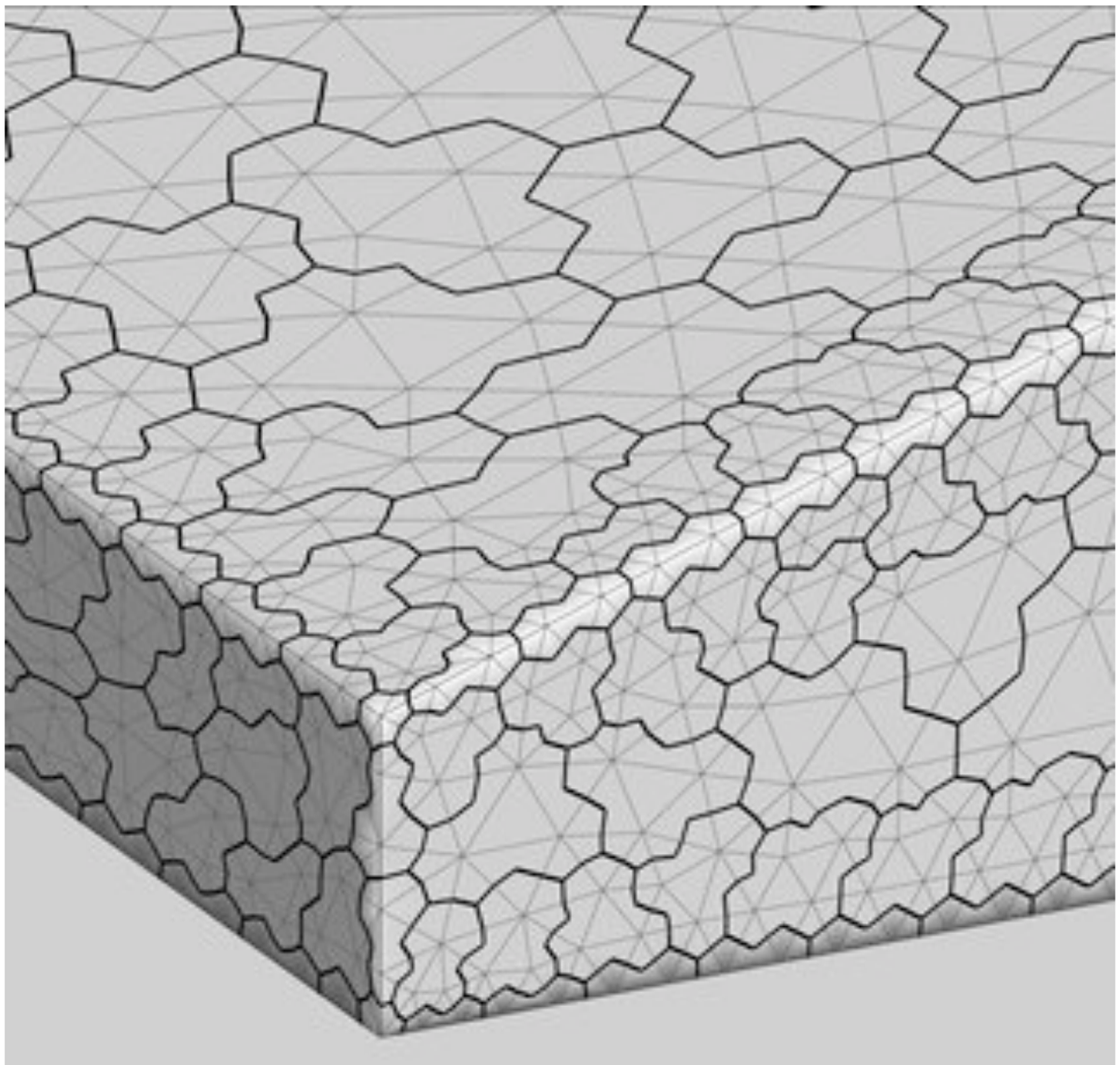
The mesh graph, composed of vertices and edges, is partitioned to group elements into larger polytopal elements.



Useful for adaptive refinement and multi-grid solvers

P.F. Antoniettiet al., "Agglomeration of polygonal grids using graph neural networks with applications to multigrid solvers, Computers & Mathematics with Applications, **154** (2024)

Agglomerated mesh example



Agglomerated mesh near wing tip

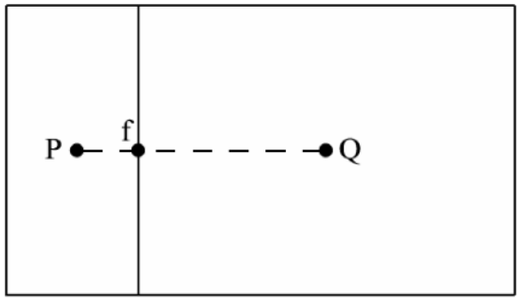
H. Nishikawa and B. Diskin, “Development and Application of Parallel Agglomerated Multigrid Methods for Complex Geometries”, AIAA paper 2011-3232.

Polytopal mesh quality

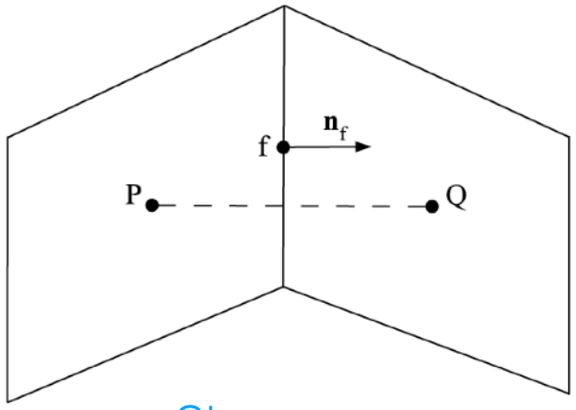
GOOD MESH → GOOD SOLUTION, but what is a “good” mesh?

Without solution behaviour knowledge, we rely on
a priori shape-based mesh quality measures.

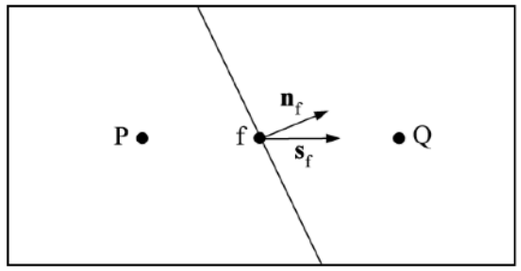
FV discretization errors associated to mesh geometry



Non-uniformity



Skewness

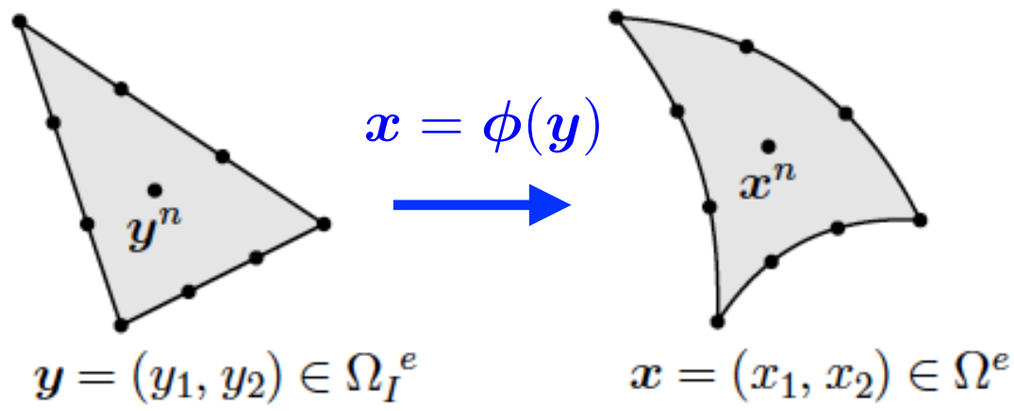


Non-orthogonality

Mapping-based quality metric

Distortion of the element shape with reference to an “ideal” shape is evaluated using:

- Jacobian: $J = \det(\boldsymbol{J})$; $\boldsymbol{J} = \partial\boldsymbol{\phi}/\partial\boldsymbol{y}$
- Metric tensor: $\boldsymbol{G} = \boldsymbol{G}^T\boldsymbol{G}$



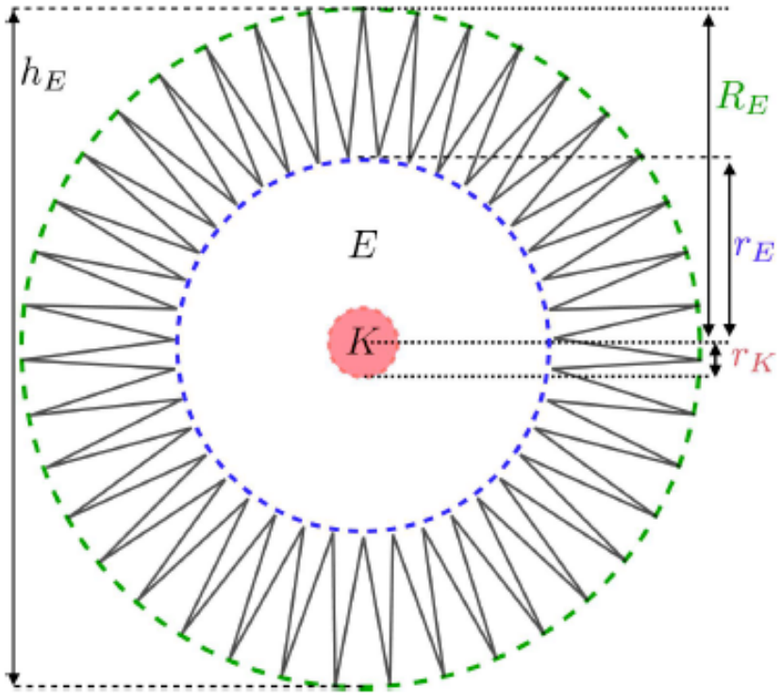
Ideal element

Current element

A recent survey of polytopal mesh quality

2D

	polygons
warping	$\max_{i=1,\dots,n}\left\{\sin^{-1}\left(\frac{\|\mathbf{v}_i-\mathbf{v}_i'\|}{l}\right)\right\}$
aspect ratio	$\frac{r}{R}$, $\frac{r}{h_E}$, $\frac{h_E}{R}$, $\frac{r_K}{r_E}$, $\frac{A_K}{A_E}$
skewness	$\sin(\theta_{\min})$, $\sin(\theta_{\max})$, $\frac{\sin(\theta_{\min})}{\sin(\theta_{\max})}$
taper	-
interpolation quality	$\sqrt{\frac{\rho_1\rho_2+\rho_1\rho_3+\rho_1\rho_4}{3}}$
mean ratio	$\min_k\left\{\frac{2}{\kappa(T_k)}\right\}$
shape regularity	$\frac{\int_E \|x-\bar{x}\|^2 dx}{2\left(\int_E dx\right)^2}$



3D

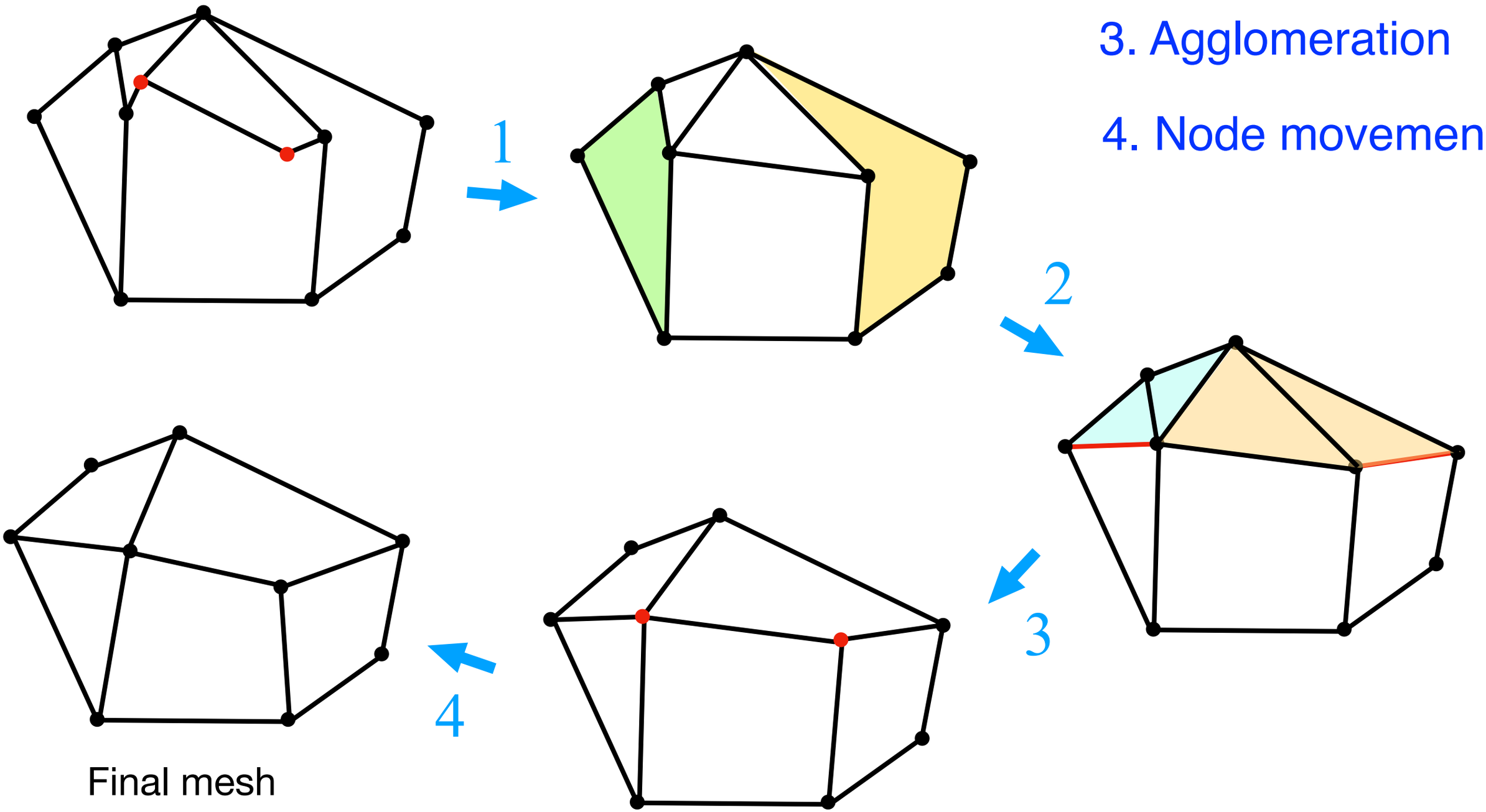
	polyhedra
warping	$\max_{i=1,\dots,n}\left\{\sin^{-1}\left(\frac{\|\mathbf{v}_i-\mathbf{v}_i'\|}{l}\right)\right\}$
aspect ratio	$\frac{r}{R}$, $\frac{r}{h_E}$, $\frac{h_E}{R}$, $\frac{r_K}{r_E}$, $\frac{V_K}{V_E}$
skewness	$\min_{i=1,\dots,n}\{\sin(\theta_i/2)\}$
taper	-
interpolation quality	$\sqrt{\frac{\rho_1\rho_2+\rho_1\rho_3}{2}}$
mean ratio	$\min_k\left\{\frac{3}{\kappa(T_k)}\right\}$
shape regularity	$\frac{\int_E \|x-\bar{x}\|^2 dx}{3\left(\int_E dx\right)^{5/2}}$

T. Sorgente et al., “A Survey of Indicators for Mesh Quality Assessment”, Computer Graphics forum, **42**(2), 2023.

Mesh modification techniques

Chose a mesh quality metric and improve the mesh by:

1. Side collapsing
2. Element splitting
3. Agglomeration
4. Node movement



HIGH-ORDER DISCRETIZATION METHODS

a. High-order finite volume methods (HOFVM)
b. Discontinuous Galerkin methods (DGM)
c. High-order hybrid methods (HOHM)
d. Virtual element methods (VEM)
e. A posteriori high-order polytopal mesh generation

An “alphabet soup” of methods

- CDO = Compatible discrete operator schemes
- DEC = Discrete exterior calculus
- DPG = Discontinuous Petrov-Galerkin
- FES = Finite element systems
- GBC = Generalized barycentric coordinate methods
- GS = Gradient schemes
- HOFV = High-order finite volume methods
- HDG = Hybrid discontinuous Galerkin methods
- HHO = Hybrid higher-order methods
- MFD = Mimetic finite difference
- VEM = Virtual element methods
- WG = Weak Galerkin methods
- VCFEM = Voronoi cell FEM
- DGFEM = Discontinuous Galerkin FEM
- PFEM = Conforming polygonal FEM
- nSFEM = n-Sided polygonal smoothed FEM
- PSBFEM = Polygonal scaled boundary FEM
- BFEM = Base forces element method
- BEM-based FEM = Boundary element based FEM
- VNM = Virtual node method (VNM)
- T/HT-FEM = Trefftz/Hybrid Trefftz polygonal FEM
- TDGFEM = Trefftz Discontinuous Galerkin FEM
- HS-F = Hybrid stress-function polygonal element

High-order finite volume methods

Compute the gradient term in the boundary integral of the FV formulation with a higher degree of accuracy.

Reconstruct a polynomial of degree $k \geq 1$ $\quad u_h(\mathbf{x}) = u(\mathbf{y}) + \mathbb{P}^k(\mathbf{x}-\mathbf{y})$ where $\boldsymbol{y}$ are the centroid coordinates using e.g. a Taylor series

$$u_h(x_1,x_2) = u(y_1,y_2) + \sum_{i=1}^{2} \left.\frac{\partial u}{\partial x_i}\right|_{(y_1,y_2)} (x_i - y_i) + \frac{1}{2}\sum_{i,j=1}^{2} \left.\frac{\partial^2 u}{\partial x_i x_j}\right|_{(y_1,y_2)} (x_i - y_i)(x_j - y_j) + \frac{1}{6}\sum_{i,j,l=1}^{2} \left.\frac{\partial^3 u}{\partial x_i x_j x_l}\right|_{(y_1,y_2)} (x_i - y_i)(x_j - y_j)(x_l - y_l) + \ldots$$

In practice, it is often desirable to use elemental averages

$$u_h(\mathbf{x}) = \bar{u} + \bar{\mathbb{P}}^k(\mathbf{x}-\mathbf{y})$$

with

$$\bar{u} = \frac{1}{|\Omega_e|}\int_{\Omega_e} u\, d\mathbf{x}; \quad \bar{\mathbb{P}}^k = \frac{1}{|\Omega_e|}\int_{\Omega_e} \mathbb{P}^k(\mathbf{x}-\mathbf{y})\, d\mathbf{x}; \quad |\Omega_e| = \int_{\Omega_e} d\mathbf{x}$$

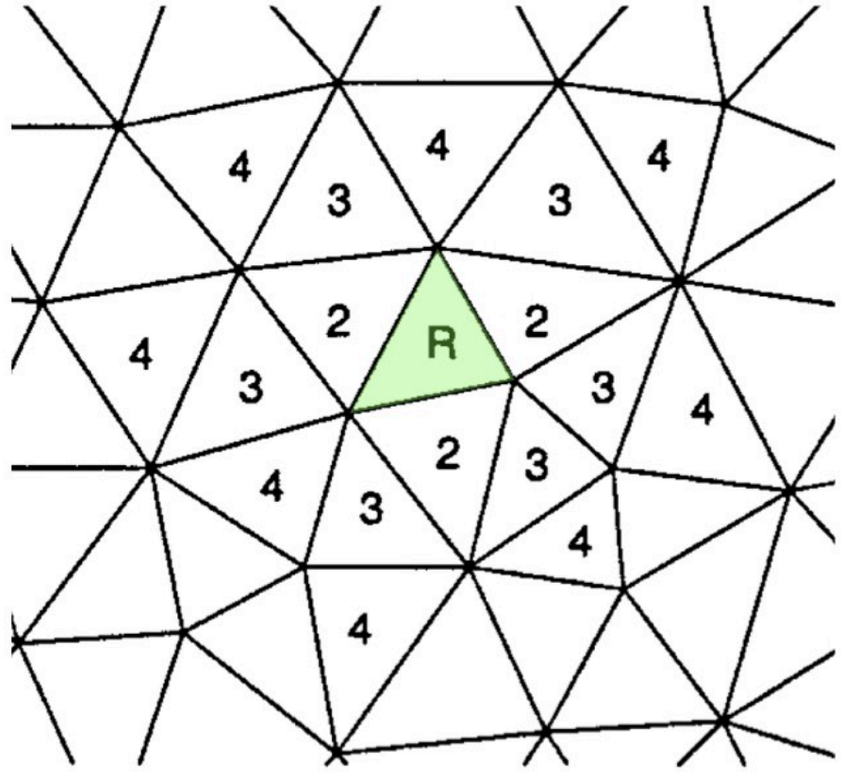
HOFV gradient reconstruction

The *coefficients of the polynomial* $\mathbb{P}^k$ are the n_d derivatives in the Taylor series

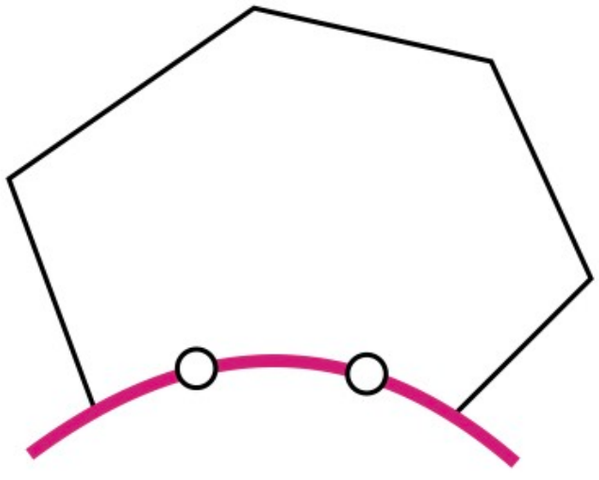
Their value is calculated by a least-squares fitting to the values of $\bar{u}$ in n_n neighbouring elements, $n_n > n_d$.

For instance, $n_n = 3$ for second-order accuracy, $n_n = 9$ for third order, and $n_n = 14$ for fourth order.

The face gradient is discontinuous, it is averaged at the quadrature points from the reconstructed values from the two adjacent cells.



Boundary conditions



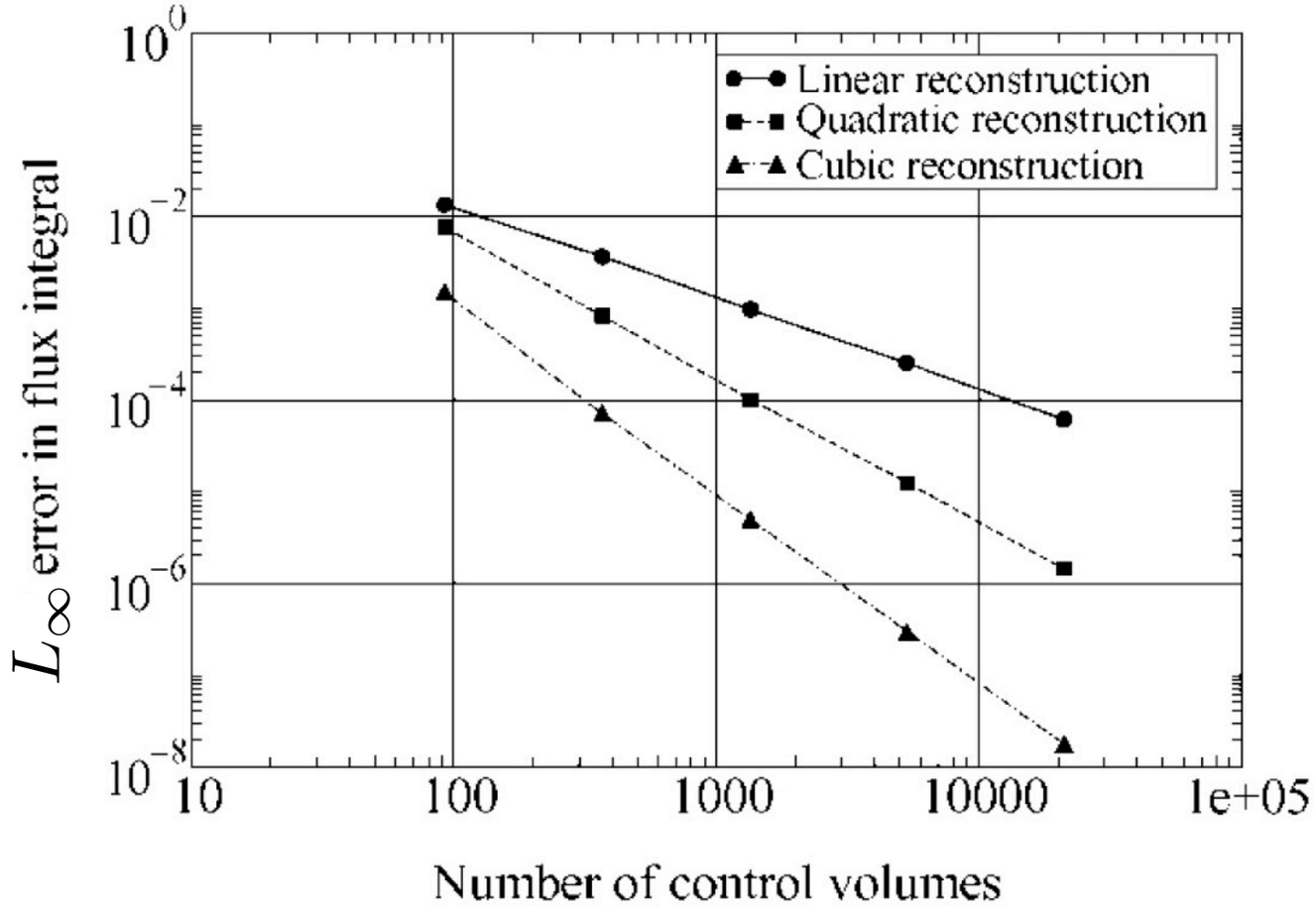
A quadrature rule is used to evaluate boundary integrals. Values of the function (Dirichlet) or normal derivative (Neumann) at the quadrature points *are imposed as constraints* in the least-squares fitting process.

C. Ollivier-Gooch and M. Van Altena, "A High-Order-Accurate Unstructured Mesh Finite-Volume Scheme for the Advection–Diffusion Equation", Journal of Computational Physics **181** (2002)

HOFVM convergence

Solution to Poisson equation in unit square

$$u(x_1, x_2) = \sin(\pi x_1) \sin(\pi x_2)$$



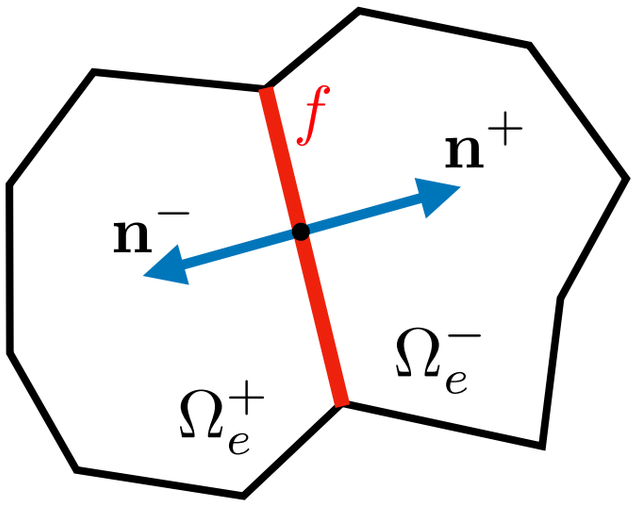
Increasing the degree k results in very large stencils. This requires special care for elements near boundaries.

Alternative compact stencils minimize a functional that measures the jumps of the reconstruction polynomial and its derivatives on each face.

High-order discontinuous Galerkin

DG methods allow for discontinuous solution with a more compact stencil

$$\int_{\Omega_e} \nabla u_h \cdot \nabla v_h \, d\mathbf{x} + \int_{\partial\Omega_e} \{\{\nabla u_h\}\} \, [\![v_h]\!] \, d\mathbf{x} = \int_{\Omega_e} f v_h \, d\mathbf{x}$$

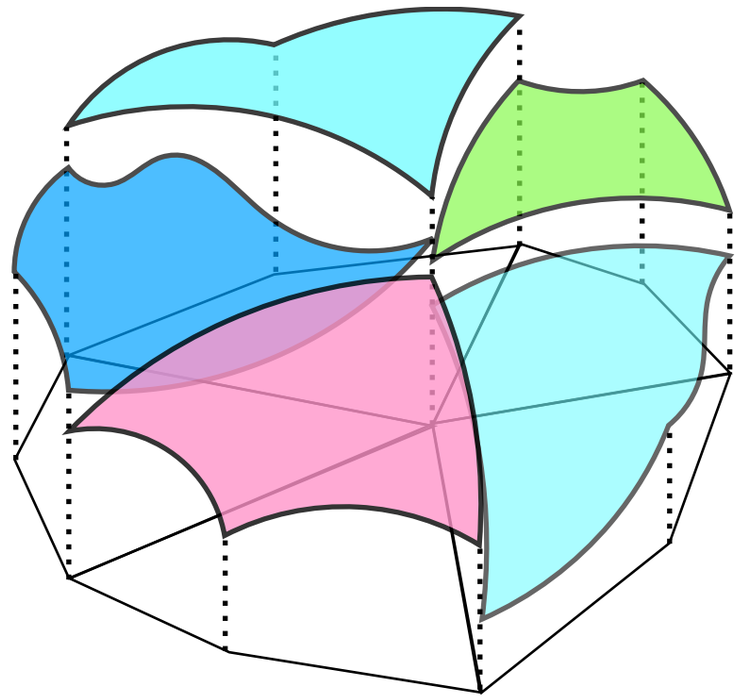


Average operator:

$$\{\{w\}\} = \frac{1}{2}(w^+ + w^-)$$

Jump operator:

$$[\![w]\!] = w^+\mathbf{n}^+ + w^-\mathbf{n}^-$$



Polytopal DG methods construct the *elemental bases in physical space*, without requiring a reference element.

Physical-space basis

Monomial bases

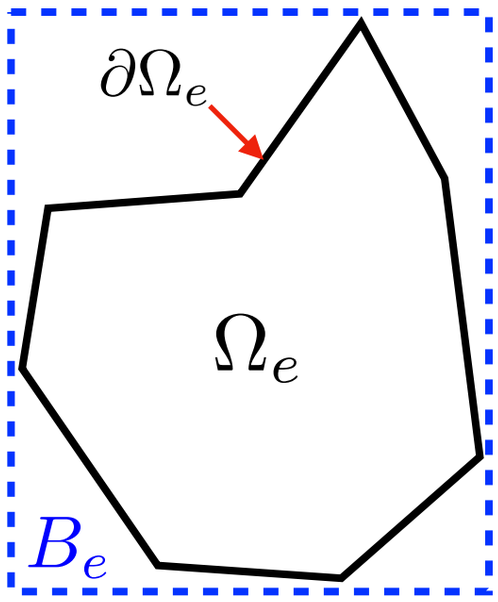
If d is the dimension of the problem, monomials are defined as

$$\mathbf{x}^{\boldsymbol{\alpha}} = \prod_{i=1}^{d} x_i^{\alpha_i} = x_1^{\alpha_1}\, x_2^{\alpha_2}\, \ldots\, x_d^{\alpha_d} \qquad \text{with} \quad \begin{cases} \text{Integer indices} \\ \boldsymbol{\alpha} = \{\alpha_i; i = 1, \ldots d\} \\ \text{Cartesian coordinates} \\ \mathbf{x} = \{x_i; i = 1, \ldots d\} \end{cases}$$

A Gram-Smith orthogonalization is use to improve the conditioning of the matrices that become stiffer as the polynomial order increases.

Tensor-product bases

Attach a Cartesian bounding box to each element and define a tensor-product of 1D polynomial bases within the box.

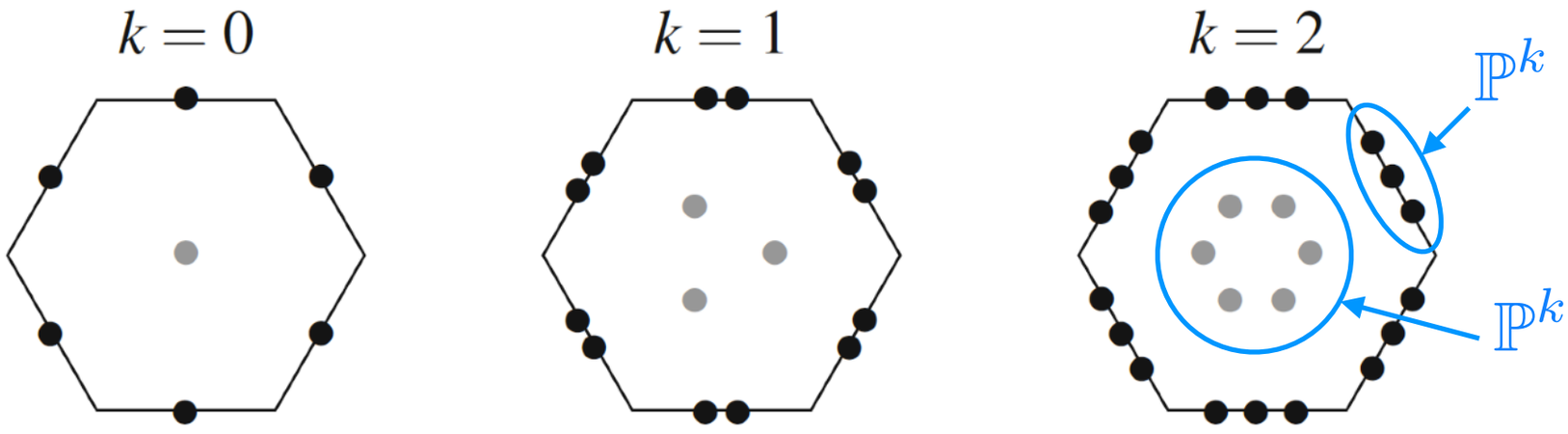


In both cases, the challenge is now to devise efficient and accurate quadrature rules for the approximation of elemental and face integrals.

A. Cangiani, Z. Dong, E. H. Georgoulis, and P Houston. "hp-Version Discontinuous Galerkin Methods on Polygonal and Polyhedral Meshes". Springer (2017)

High-order hybrid method

An approximation of degree $k+1$, an *elliptic projection*, of a function can be computed from L^2 *projections* of degree k on the polytopal element and on each of its faces.



The HHO method requires:

- Polynomial spaces of degree $k>0$ in the element and its faces, possibly discontinuous.
- A local interpolator constructed from L^2 projections of degree k.
- A local potential reconstruction operator of degree $k+1$.
- A stabilization term to match the elements trace values with the face values.

M. Cicuttin, A. Ern, and N. Pignet, “Hybrid High-Order Methods – A Primer with Applications to Solid Mechanics”, Springer (2021)

HHO building blocks

Inner product

$$(a,b)_X = \int_X a\,b\,d\mathbf{x} \qquad X = e\,(\text{element}),\, f\,(\text{face})$$

L^2-orthogonal projection $\pi_X^{0,k}$ $(X = e, f)$

$$(\pi_X^{0,k} v - v, w)_X = 0 \quad w \in \mathbb{P}^k$$

Elliptic projection $\pi_X^{1,k+1}$

$$(\nabla(\pi_X^{1,k} v - v), \nabla w)_X = 0 \quad \text{and} \quad (\pi_X^{1,k} v - v, 1)_X = 0 \quad w \in \mathbb{P}^k$$

The first condition defines the polynomial up a constant and the second one fixes the value of the constant.

Gradient reconstruction

To reconstruct the gradient, the starting point is the expression

$$(\nabla v, \nabla w)_e = -(v, \nabla^2 w)_e + \sum_f (v, \nabla w \cdot \mathbf{n})_f$$

Replace v by suitable projections

$$(\nabla \pi_e^{1,k+1} v, \nabla w)_e = -(\pi_e^{0,k} v, \nabla^2 w)_e + \sum_f (\pi_f^{0,k} v, \nabla w \cdot \mathbf{n})_f \qquad w \in \mathbb{P}^{k+1}$$

The elliptic projection $\pi_e^{1,k+1}$ does not require the full knowledge of the function v, only its projection, since

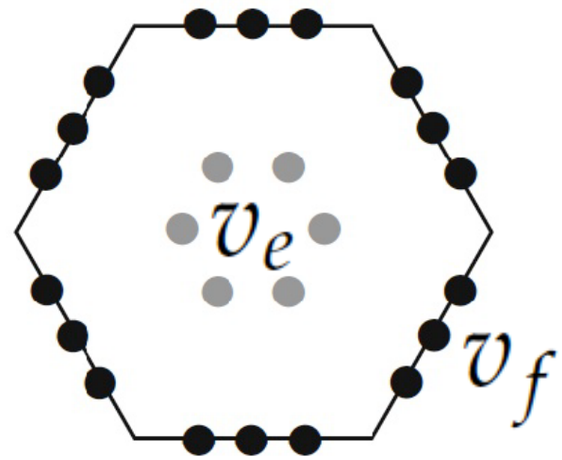
$$(\pi_e^{1,k+1} v - v, 1)_e = (\pi_e^{1,k+1} v - \pi_e^{0,k} v, 1)_e = 0$$

We *only require the projections* $\pi_e^{0,k} v$ and $\pi_f^{0,k} v$ (for all faces).

HHO discretization

Consider a set of elemental unknowns which are polynomials of degree k on the element and its faces

$$\hat{v}_e := [v_e, v_f]; \qquad v_e \in \mathbb{P}^k(e),\ v_f \in \mathbb{P}^k(f)$$



A natural choice is the local interpolator

$$\hat{v}_e \approx \hat{I}_e v = [\pi_e^{0,k} v, \pi_f^{0,k} v]$$

Calculate an approximation of the elemental gradient through the local potential reconstruction operator R_e^{k+1} such that $R_e^{k+1}\hat{v}_e \in \mathbb{P}^{k+1}$

$$(\nabla R_e^{k+1}\hat{v}_e, \nabla w)_e = -(v_e, \nabla^2 w)_e + \sum_f (v_f, \nabla w \cdot \mathbf{n})_f; \quad w \in \mathbb{P}^{k+1}$$

again, to fix the constant, we impose

$$(R_e^{k+1}\hat{v}_e - v_e, 1)_e = 0$$

This construction leads to optimal approximation in $\mathbb{P}^{k+1}$ since we have

$$R_e^{k+1}(\hat{I}_e v) = \pi_e^{1,k+1} v$$

Stabilization

An approximation of the form

$$a_e(u,v) = \int_{\Omega_e} \nabla u \cdot \nabla v \, d\mathbf{x} \approx (\nabla R_e^{k+1} \hat{u}_e, \nabla R_e^{k+1} \hat{v}_e)_e$$

is not stable in general because $\nabla R_e^{k+1} \hat{v} = 0 \not\Longrightarrow v_e = v_f = \text{cons.}$

To fix the mismatch, a stabilization term, S_f, is added of the form

$$a_e(\hat{u}_e, \hat{v}_e) = (\nabla R_e^{k+1} \hat{u}_e, R_e^{k+1} \hat{v}_e)_e + S_f(\hat{u}_e, \hat{v}_e)$$

Introducing two operators that capture the difference of these values as

$$\delta_e^k \hat{u}_e = \pi_e^{0,k}(R_e^{k+1} \hat{v}_e - v_e), \qquad \delta_f^k \hat{u}_e = \pi_f^{0,k}(R_e^{k+1} \hat{v}_e - v_f)$$

we build a symmetric and positive definite form of S_f that ensures polynomial consistency and stability

$$S_f(\hat{u}_e, \hat{v}_e) = \sum_f h_f^{-1} \left((\delta_f^k \hat{u}_e - \delta_e^k \hat{u}_e), (\delta_f^k \hat{v}_e - \delta_e^k \hat{v}_e) \right)_f$$

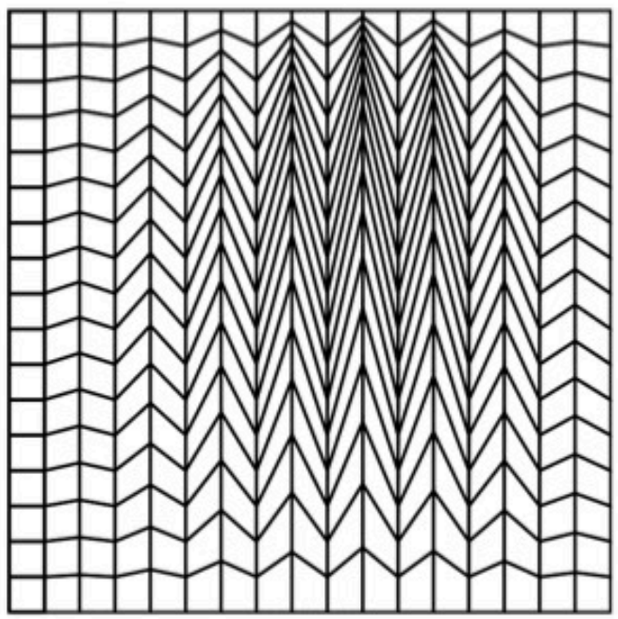
The final discrete problem is

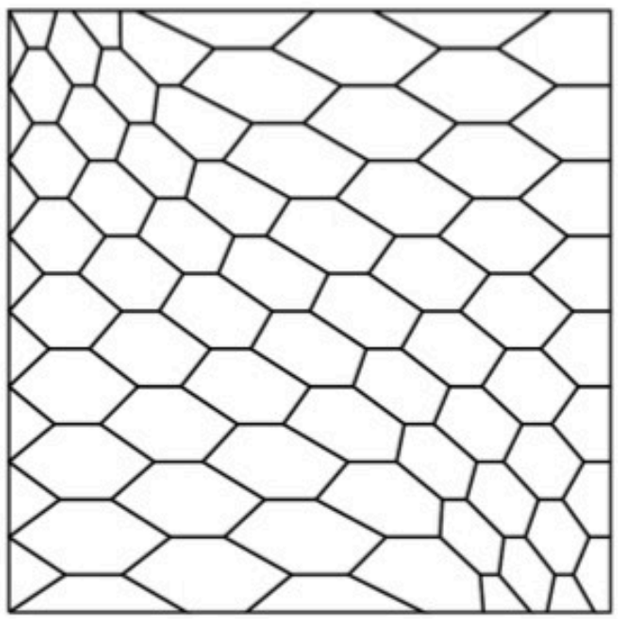
$$a_h(u_h, v_h) = \sum_e a_e(\hat{u}_e, \hat{v}_e) = \sum_e (f, v_e)_e$$

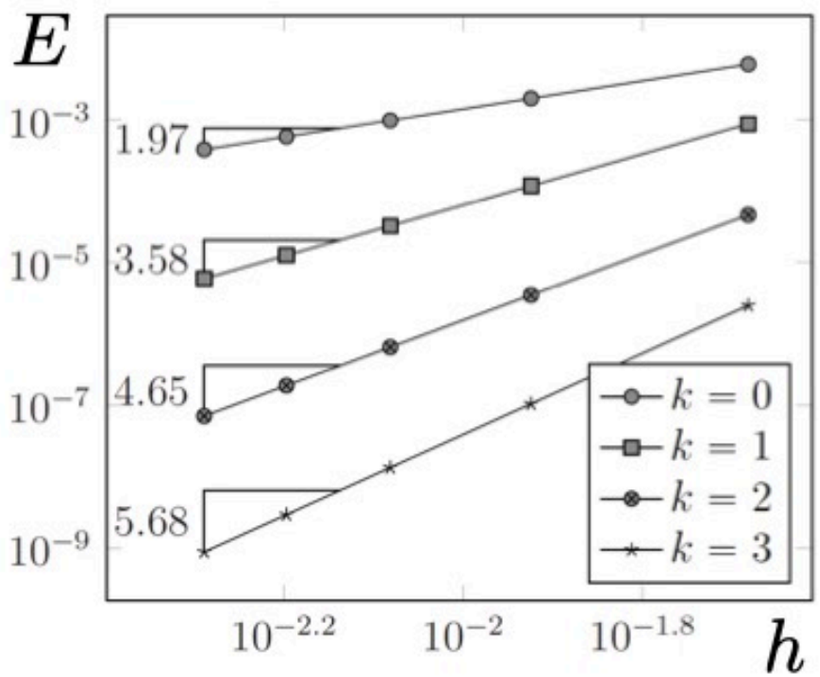
HHO convergence

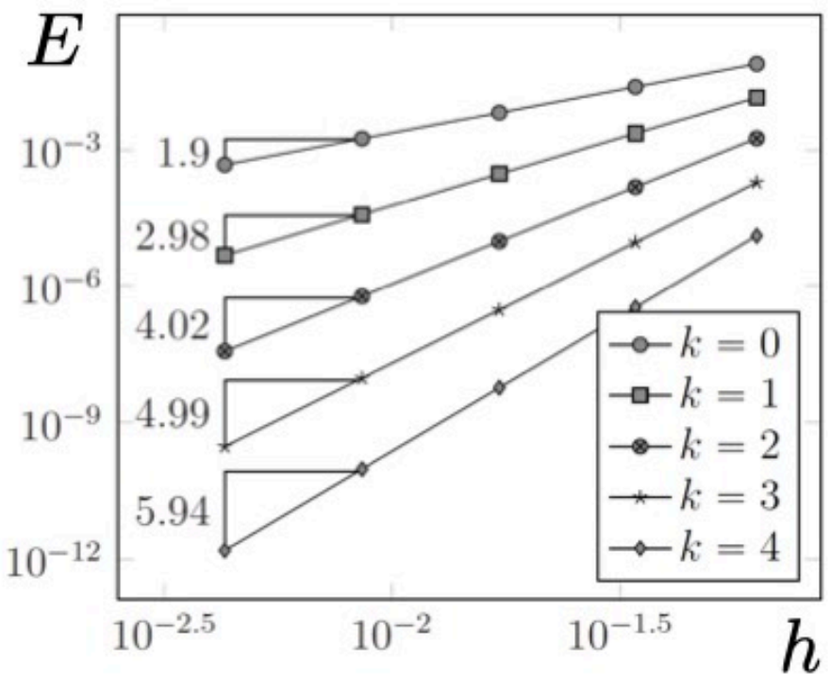
Solution to Poisson equation in unit square $u(x_1, x_2) = \sin(\pi x_1)\sin(\pi x_2)$

$$E = \|u - u_h\|_{L^2}$$









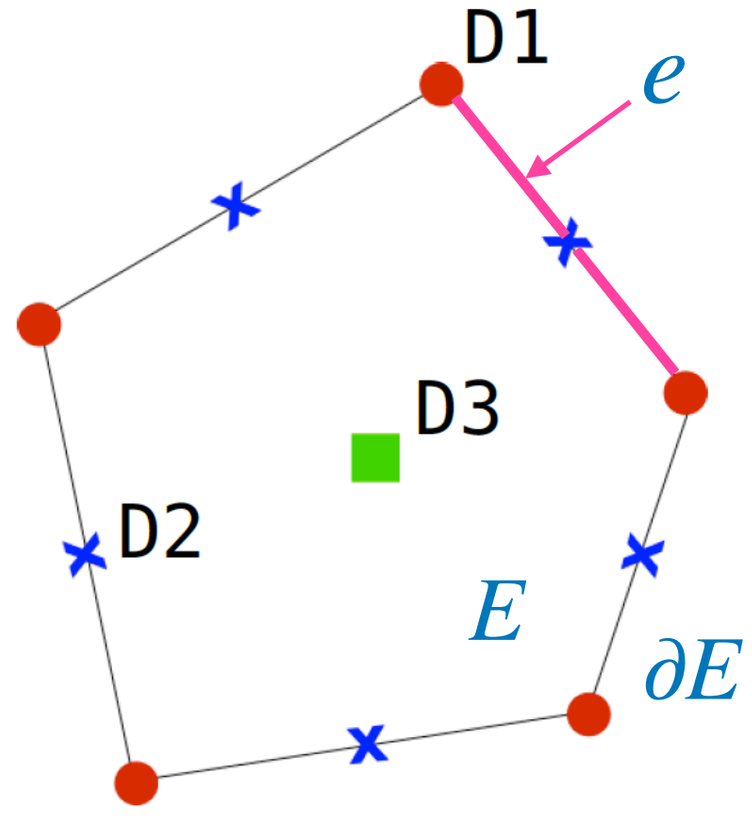
Virtual element method

VEM functions can be interpreted as the solution of $\nabla^2 u_h = p_l;\ p_l \in \mathbb{P}^l$ with prescribed values at the element's boundary.

The solution is not a polynomial and the VEM space may contain non-polynomials.

We define the 2D VEM space as

$$V_h^k := \{v : \Delta v \in \mathbb{P}_{k-2}(E),\ v|_e \in \mathbb{P}_k(e)\}$$



A function v is uniquely determined by:

- D1: value of v at the vertices.
- D2: $k+1$ values of v on the edge nodes.
- D3: $k(k+1)/2$ moments $(v, p_k)_E$.

The function v is not known a priori, it may not have a explicit expression, it is virtual: never computed explicitly, but known only through its d.o.f.

L. Beirão da Veiga et al., "Basic principles of virtual element methods", Math. Models Methods Appl. Sci., **23** (2013)

VEM building blocks

To obtain a VEM discretization of the form

$$a(v_h, w_h) = \sum_E (\nabla v_h, \nabla w_h)_E = \sum_E (f, w_h)_E$$

we use (again) the expression

$$(\nabla v, \nabla p_k)_E = -(v, \nabla^2 p_k)_E + (v, \mathbf{n} \cdot \nabla p_k)_{\partial E}$$

Elliptic projection: $v \longrightarrow \pi_E^{1,k} v$ can be calculated from the d.o.f.

To obtain a *computable* approximation, dropping indices, consider

$$v_h = \pi v_h + v_h - \pi v_h = \pi v_h + (I - \pi) v_h$$

Projection error

so that

Computable

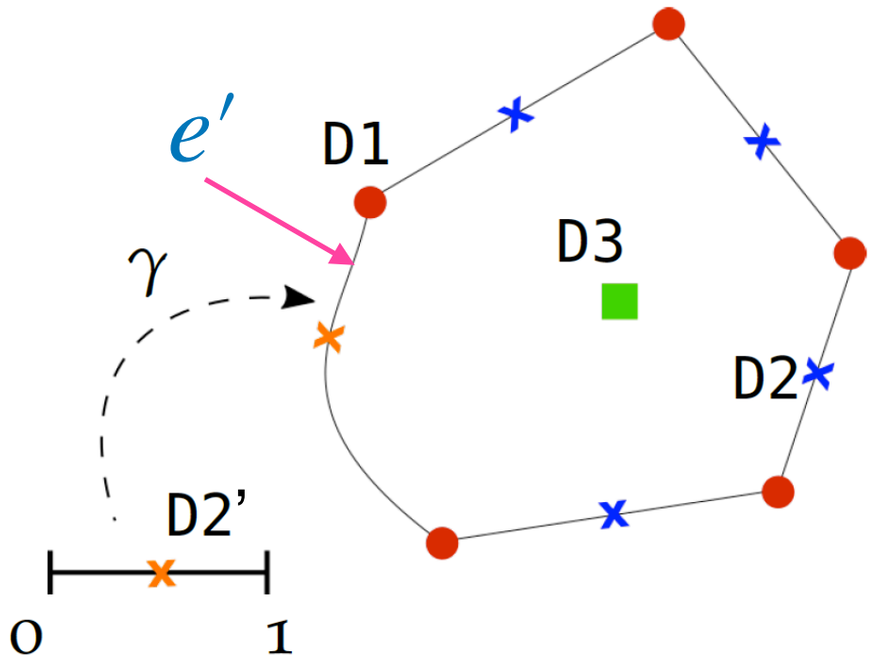
Not computable

$$a(v_h, w_h) = a(\pi v_h, \pi w_h) + a\left((I - \pi) v_h, (I - \pi) w_h\right)$$

Replace by a computable symmetric bilinear *stabilisation term* $S(.,.) \approx a(.,.)$

$$a(v_h, w_h) = a(\pi v_h, \pi w_h) + S\left((I - \pi) v_h, (I - \pi) w_h\right)$$

Curvilinear VEM



Define a curved edge by a map $\mathbf{x} = \boldsymbol{\gamma}(t);\ 0 \leq t \leq 1$

Polynomial space in the curved edge:

$$v|_{e'} \in \mathbb{P}^k([0,1]) \circ \boldsymbol{\gamma}$$

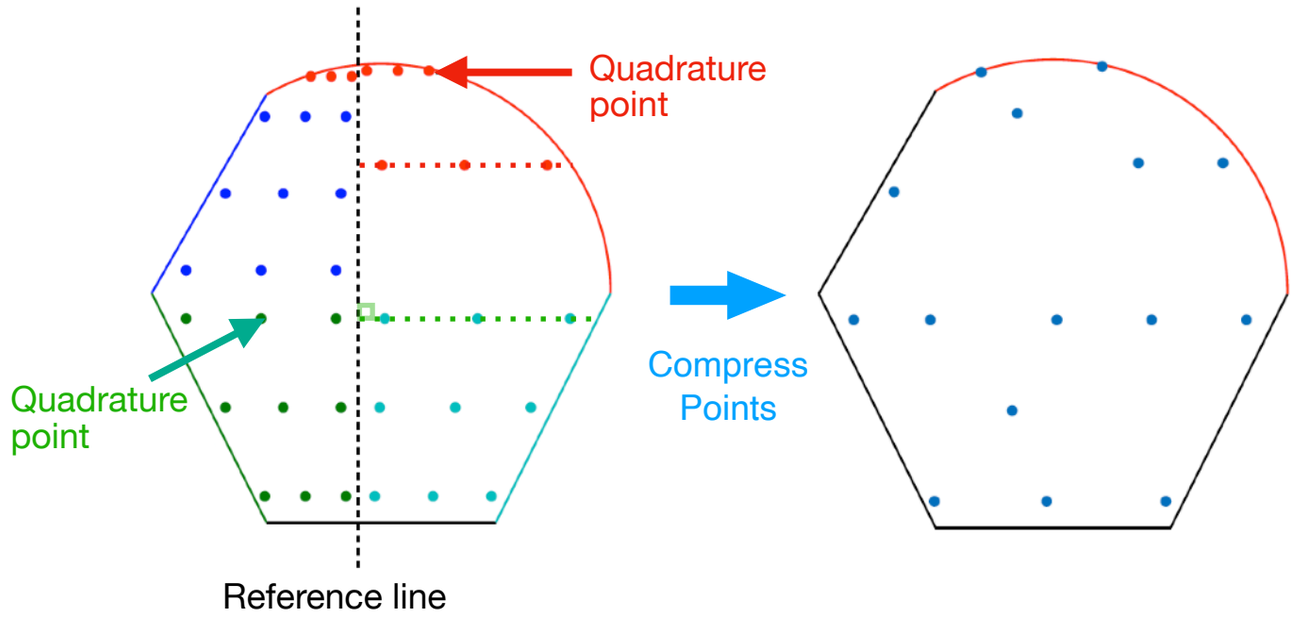
A function v is now determined by:

- D1: value of v at the vertices.
- D2: $k+1$ values of v on the edge nodes.
- D2': k – 1 values on the parameter space [0, 1]. associated with the curved edge e'.
- D3: $k(k+1)/2$ moments $\left(v, p_k\right)_E$.

Integration in E

Define a reference line.

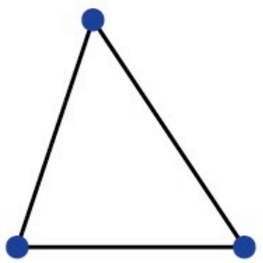
Locate quadrature points on the (orthogonal) segment within the edge quadrature point and the intersection point with the reference line.



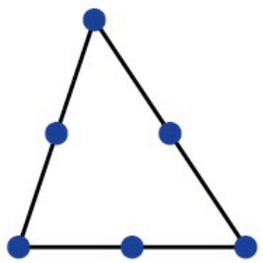
VEM vs FEM

Triangular Elements

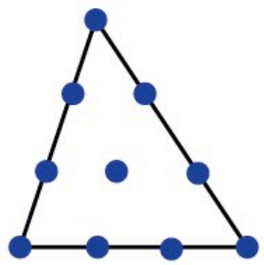
SAME



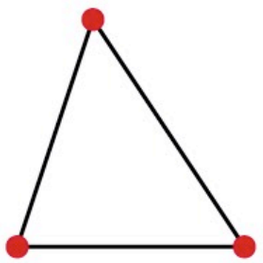
FEM $k = 1$



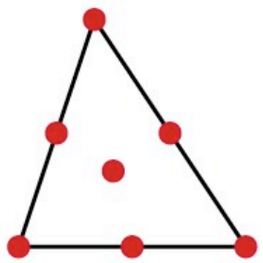
FEM $k = 2$



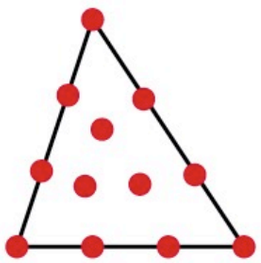
FEM $k = 3$



VEM $k = 1$

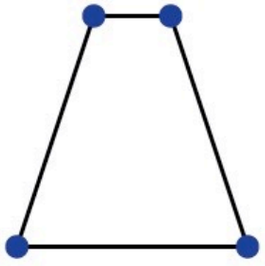


VEM $k = 2$

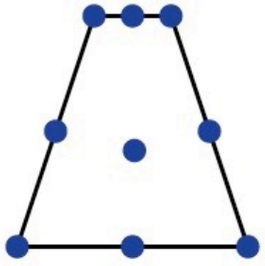


VEM $k = 3$

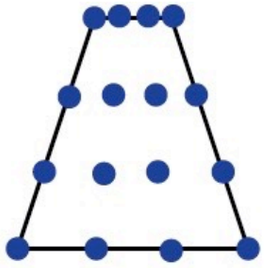
SAME



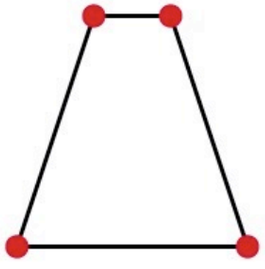
FEM $k = 1$



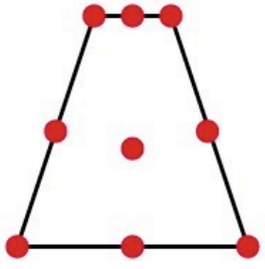
FEM $k = 2$



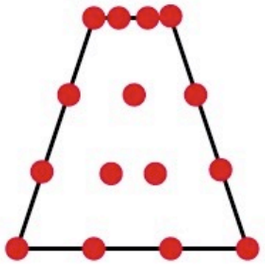
FEM $k = 3$



VEM $k = 1$



VEM $k = 2$



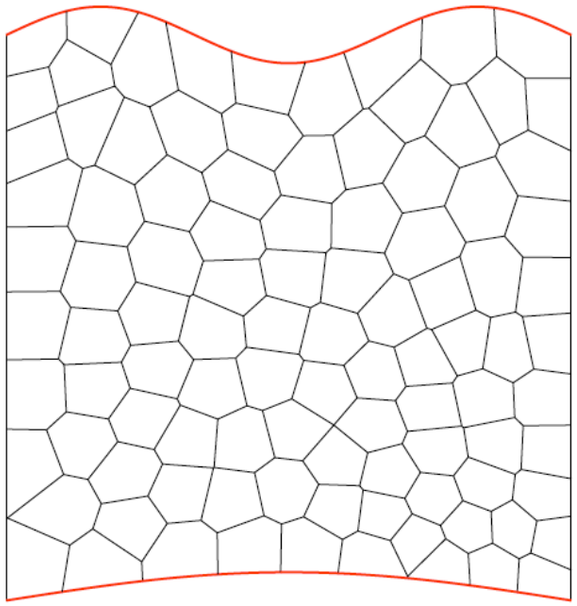
VEM $k = 3$

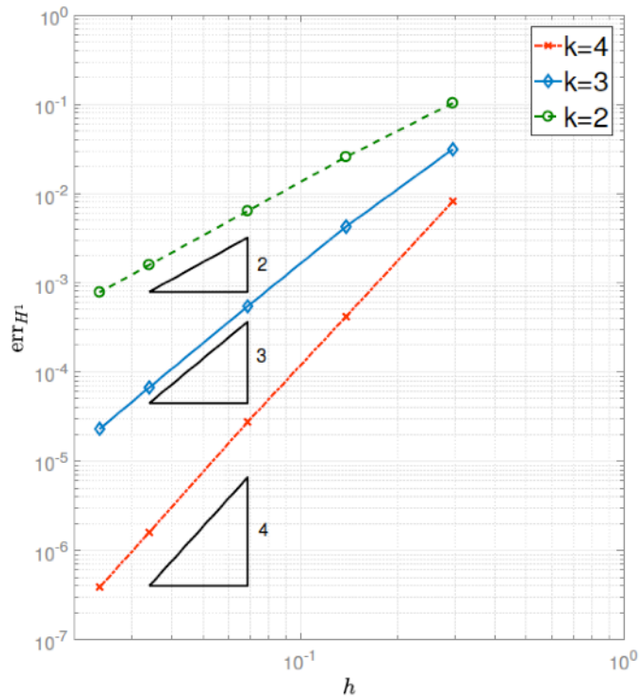
Quadrilateral Elements

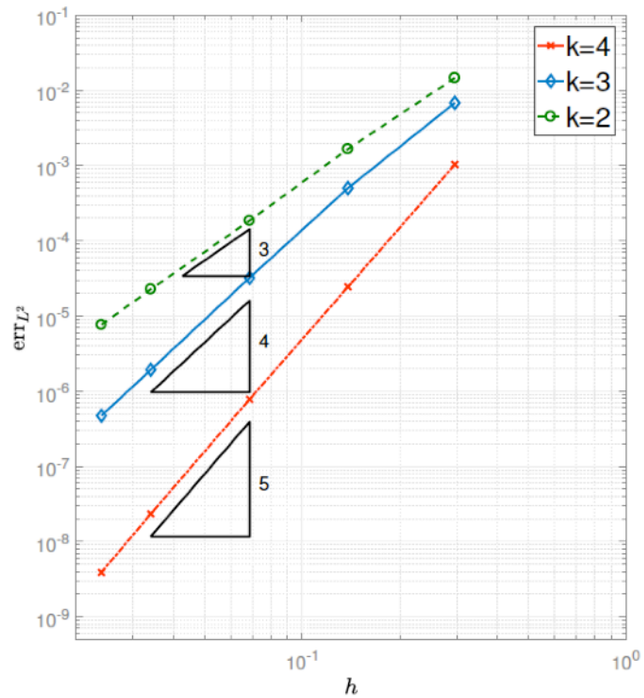
Boundary conformity matters!

$$u(x_1, x_2) = -\left[x_2 - \tfrac{1}{20}\sin(\pi x_1)\right]\left[x_2 - 1 - \tfrac{1}{20}\sin(3\pi x_1)\right]\left[3 + \sin(5x_1)\sin(7x_2)\right]$$

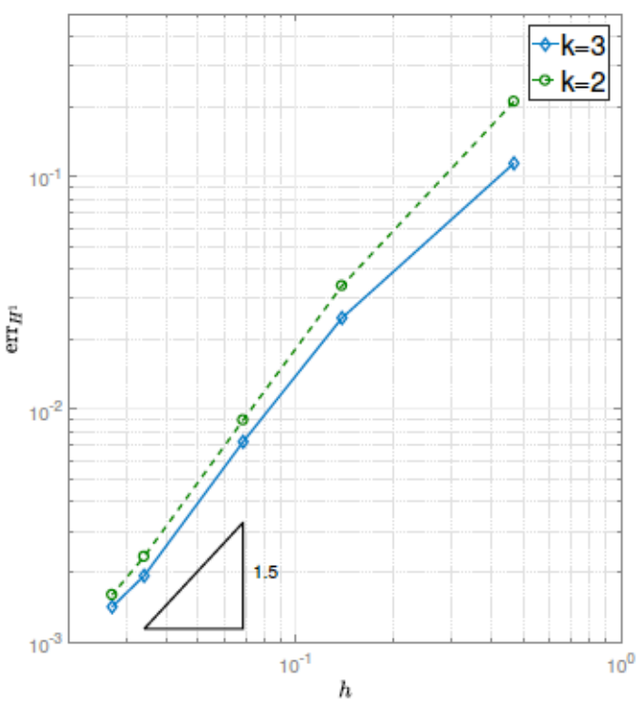
Convergence in boundary non-conforming meshes is **sub-optimal**!

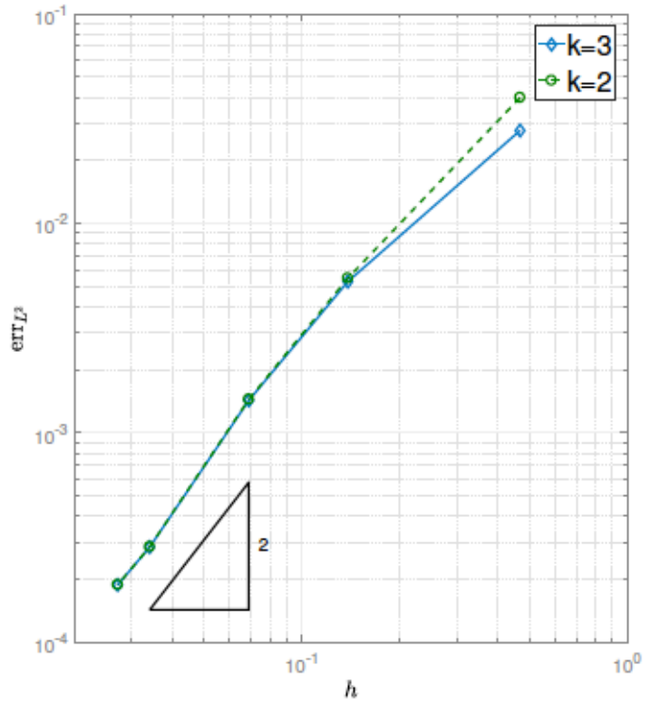






Conforming ✓





Non-conforming ✗

Meshing requirements for high-order methods

A curvilinear discretization requires this geometrical information:

1. The *coordinates of a set of points on the vertices and edges* of the polygonal mesh which are used to interpolate the numerical solution v.

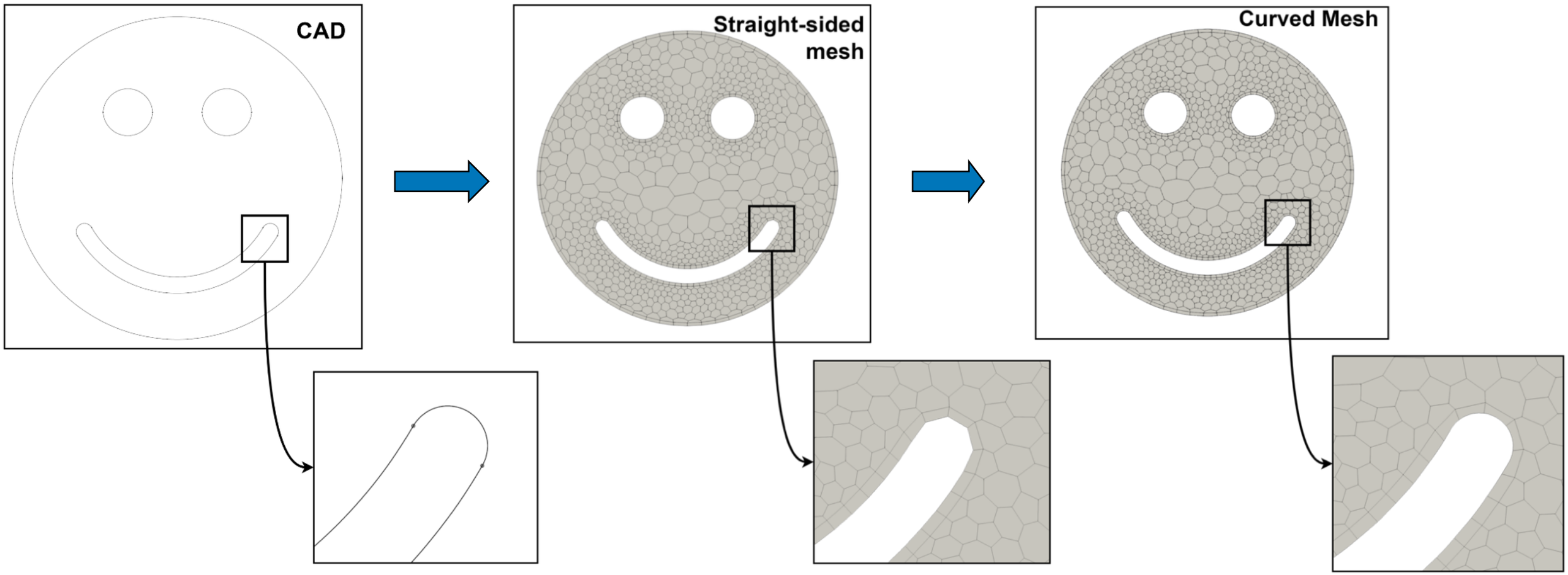
2. The *coordinates of a set of quadrature points* and their corresponding weights for evaluating integrals over:

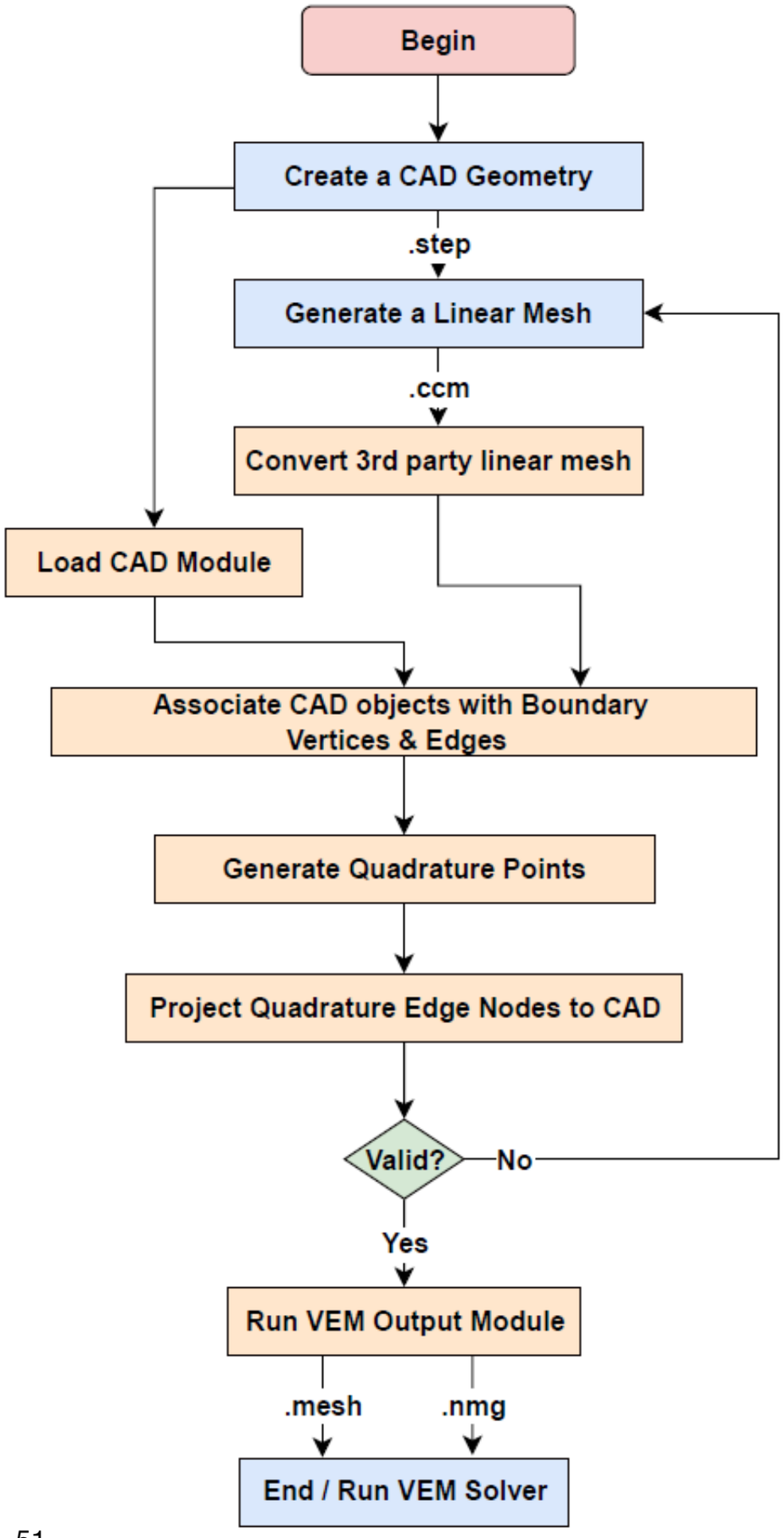
    a. Curved edges, and

    b. Polygons with curved edges

3. The *mapping* $\gamma(t)$ *defining the curved edge* which is used to compute tangent and normal vectors appearing in some of the integrals.

A posteriori high-order polytopal mesh generation

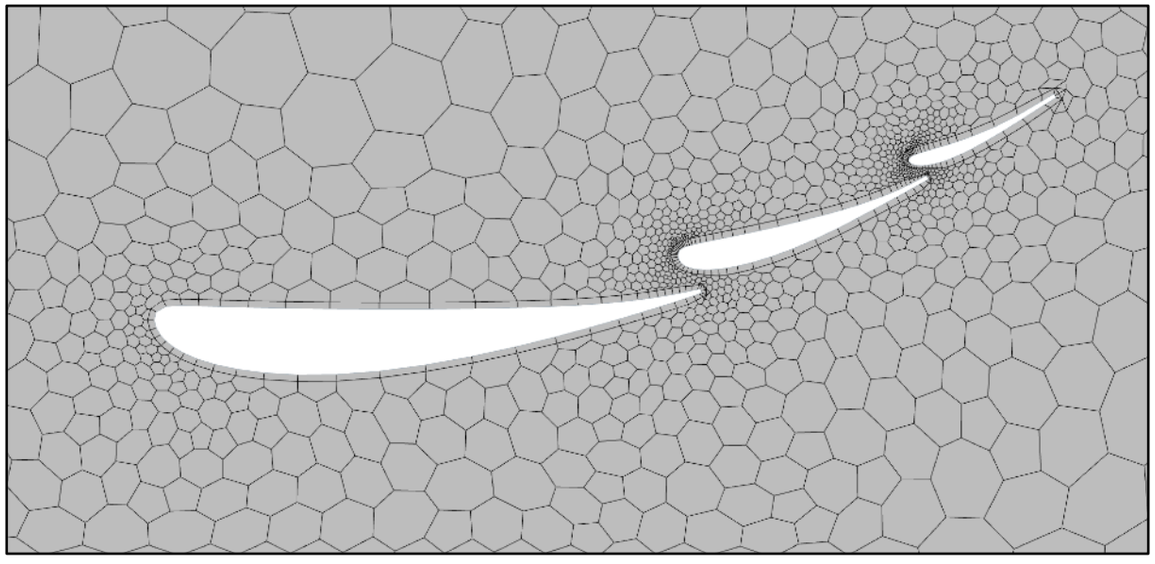




High-order mesh generation workflow

1. CAD geometry input via API.
2. Straight-sided (linear) mesh generation.
3. CAD-mesh reconstruction.
4. VEM edge quadrature and interpolation points.
5. Projections onto CAD.
6. Output VEM geometrical information.

CAD queries and “linear” mesh generation

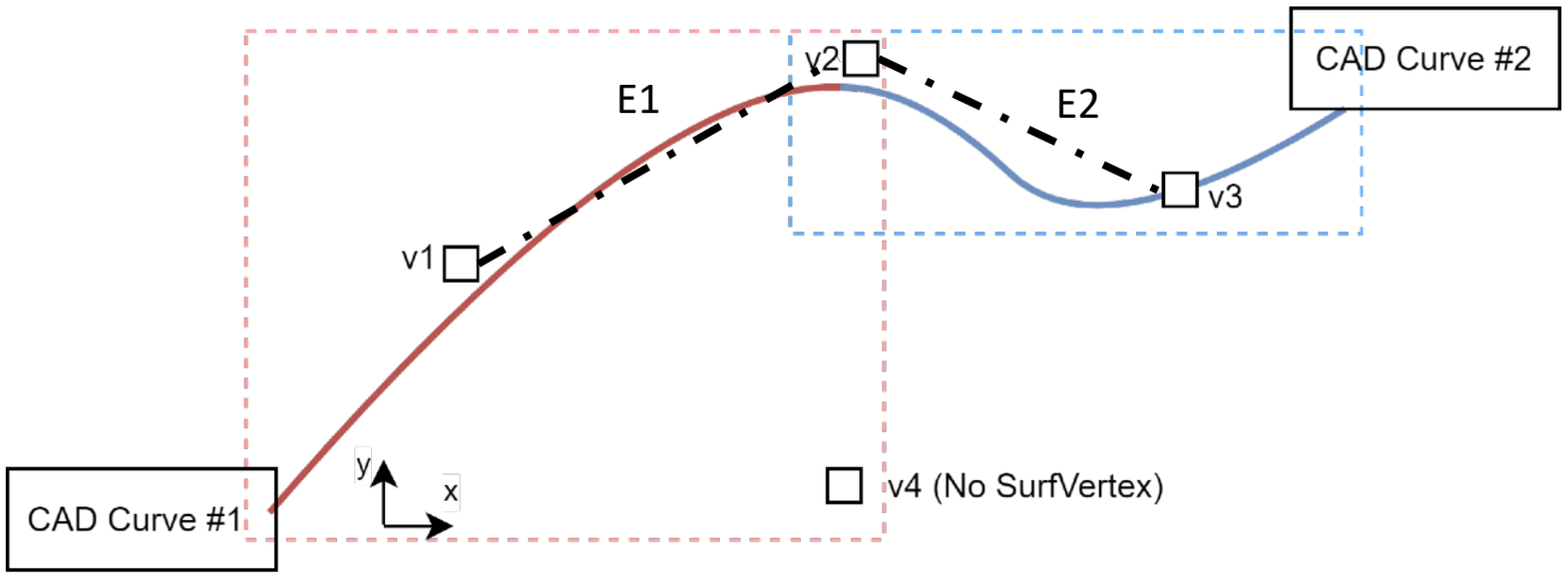


1. **CAD API - NekMesh**
   - BRep - OpenCascade or CADfix.
   - STEP (CATIA, SolidWorks, etc.)
2. **“Linear” mesh generation**
   - External polyhedral mesh generator (STAR-CCM+).
   - Mesh conversion to NekMesh linear N-sided polygonal element.
   - Applicable to other linear polygonal mesh generators.

Recovering the CAD missing link

Frequently when importing third-party meshes:

- Vertices of the linear mesh are not on the CAD.
- No CAD-mesh connectivity is available.

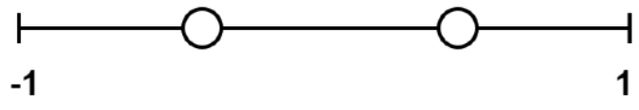


CAD recovery steps:

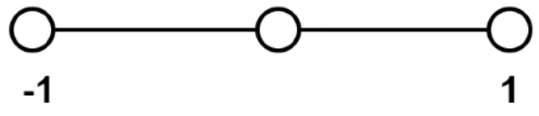
1. Construct bounding boxes around the CAD curves.
2. Assign CAD curves to vertices.
3. Project mesh vertices to the CAD curve.
4. Assign CAD curve to edges.

Quadrature point projection onto CAD

1. VEM edge quadrature rules (Arbitrary order)

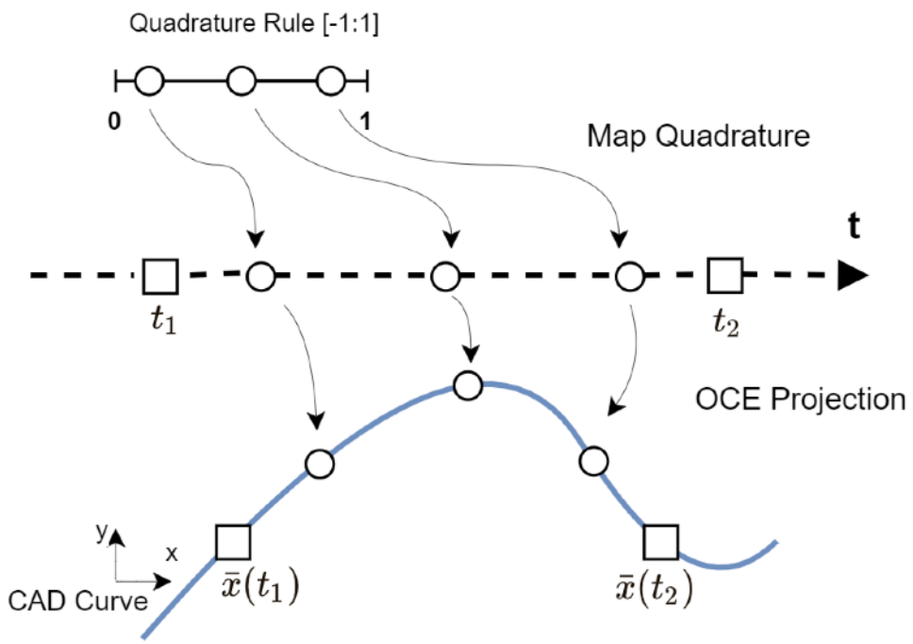


Quadrature Points (Gauss-Legendre)

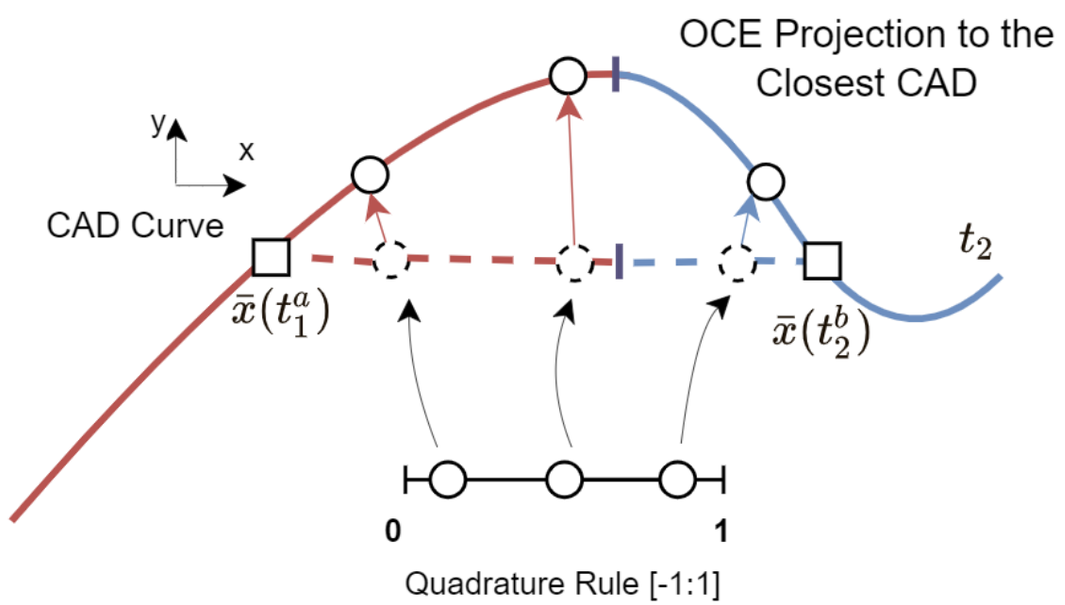


Interpolation Points (Gauss-Lobbato-Legendre)

2. Parametric projection



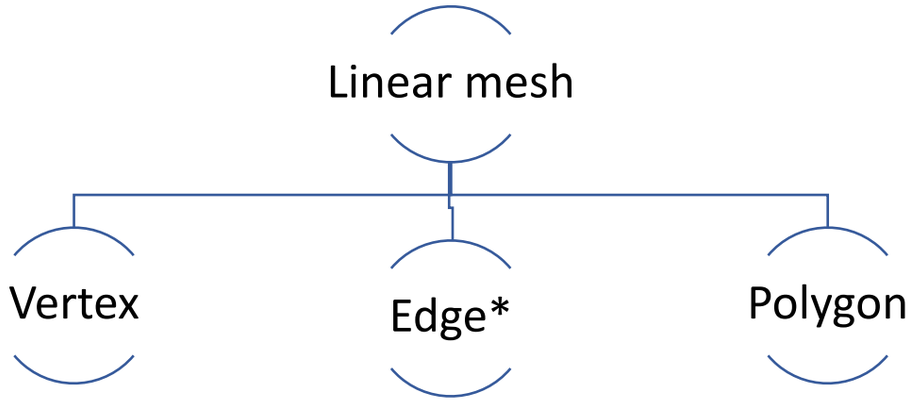
3. Multi-NURBS edge projection

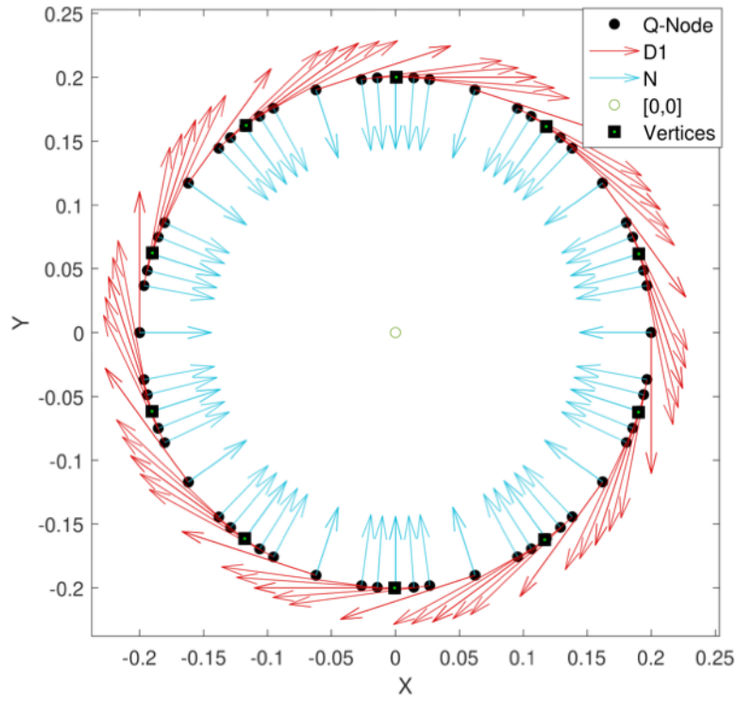


Fail-safe approach

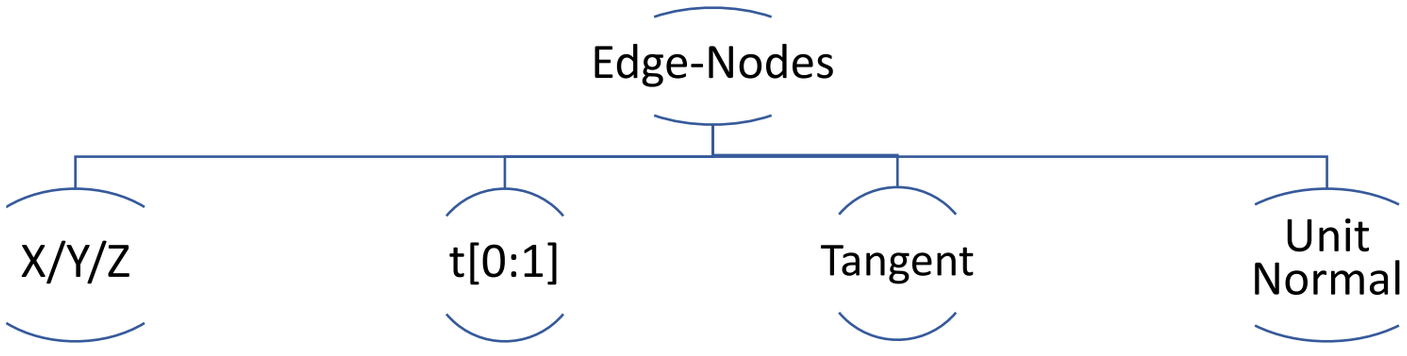
Geometrical queries and VEM integration

Linear mesh output





High-order geometrical information



All geometrical queries are processed by the CAD-API.

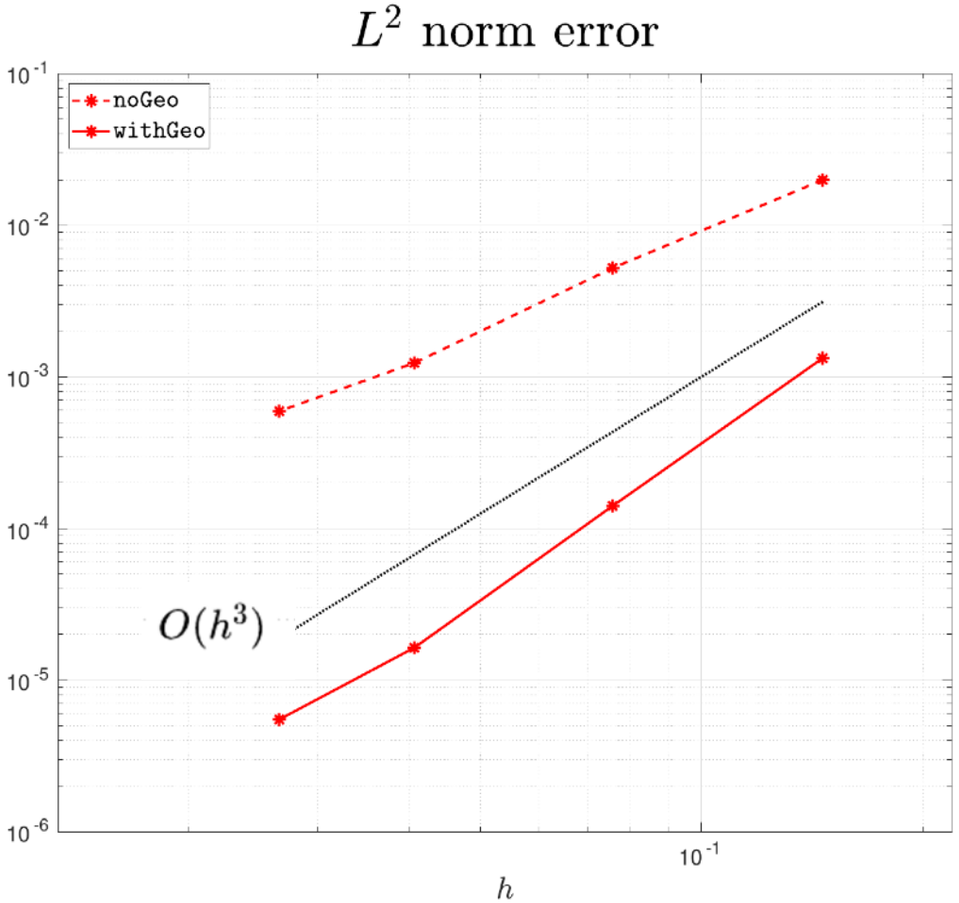
VEM solution

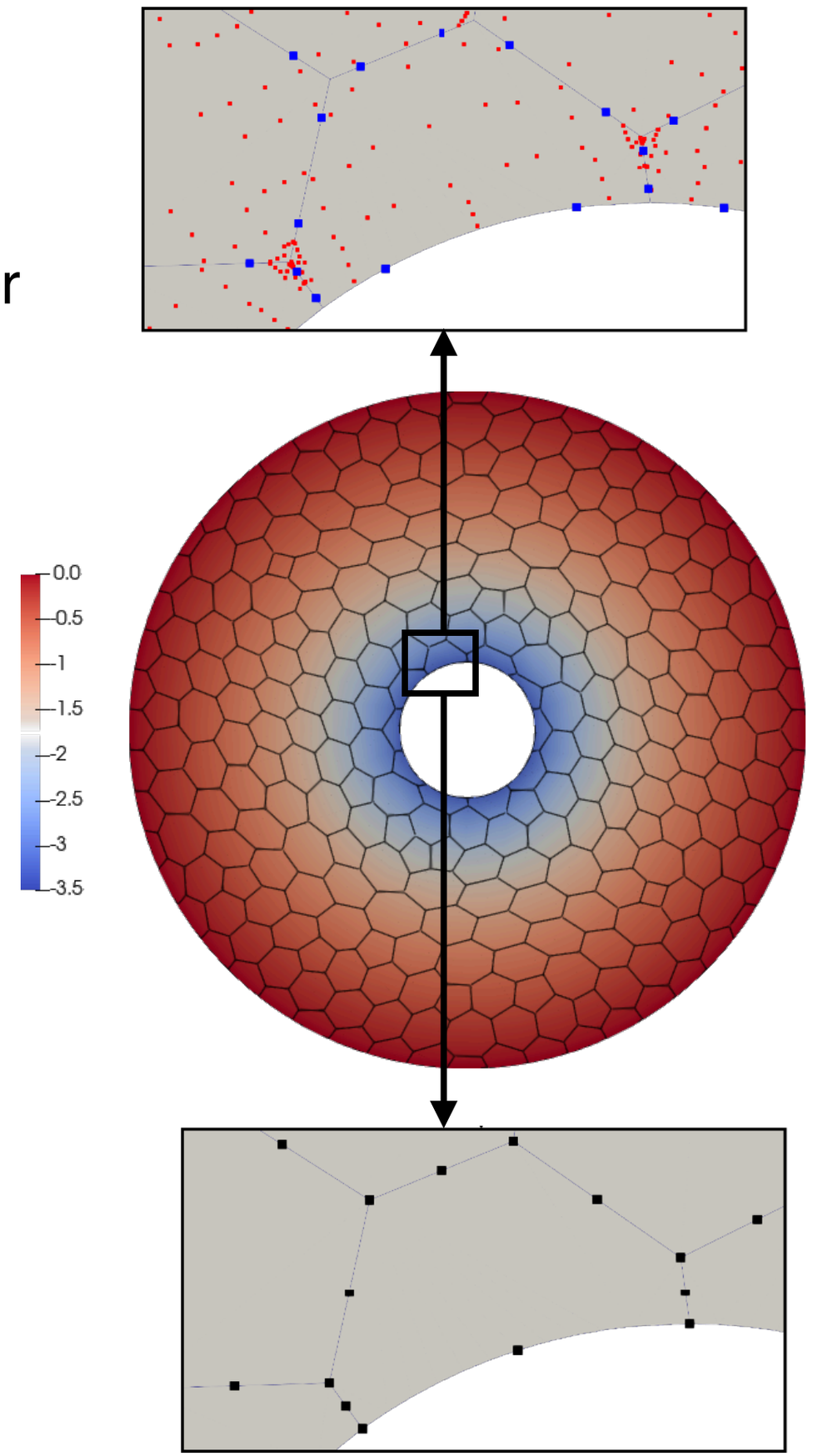
$$u(x, y) = \log(x^2 + y^2)$$

1. Curvilinear mesh accuracy is better than for straight-sided mesh for the VEM ($k = 2$)

2. Mesh convergence (h):

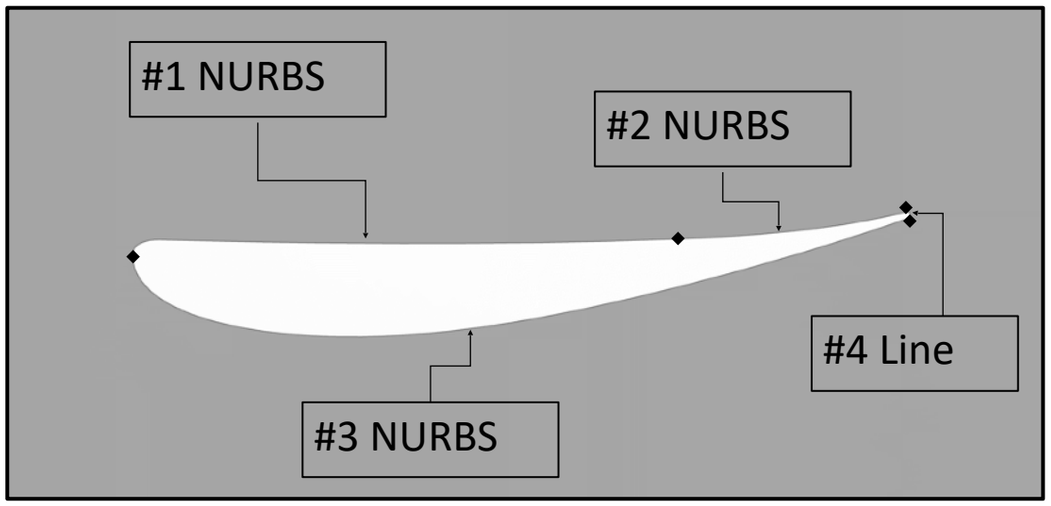
   L^2 norm error: $O(h^3)$



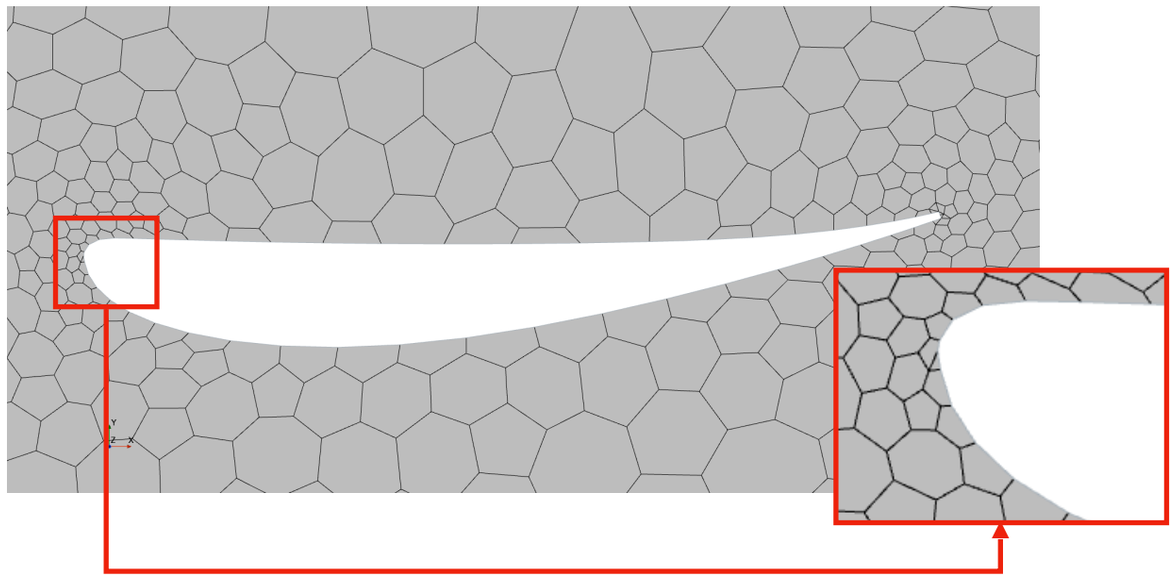


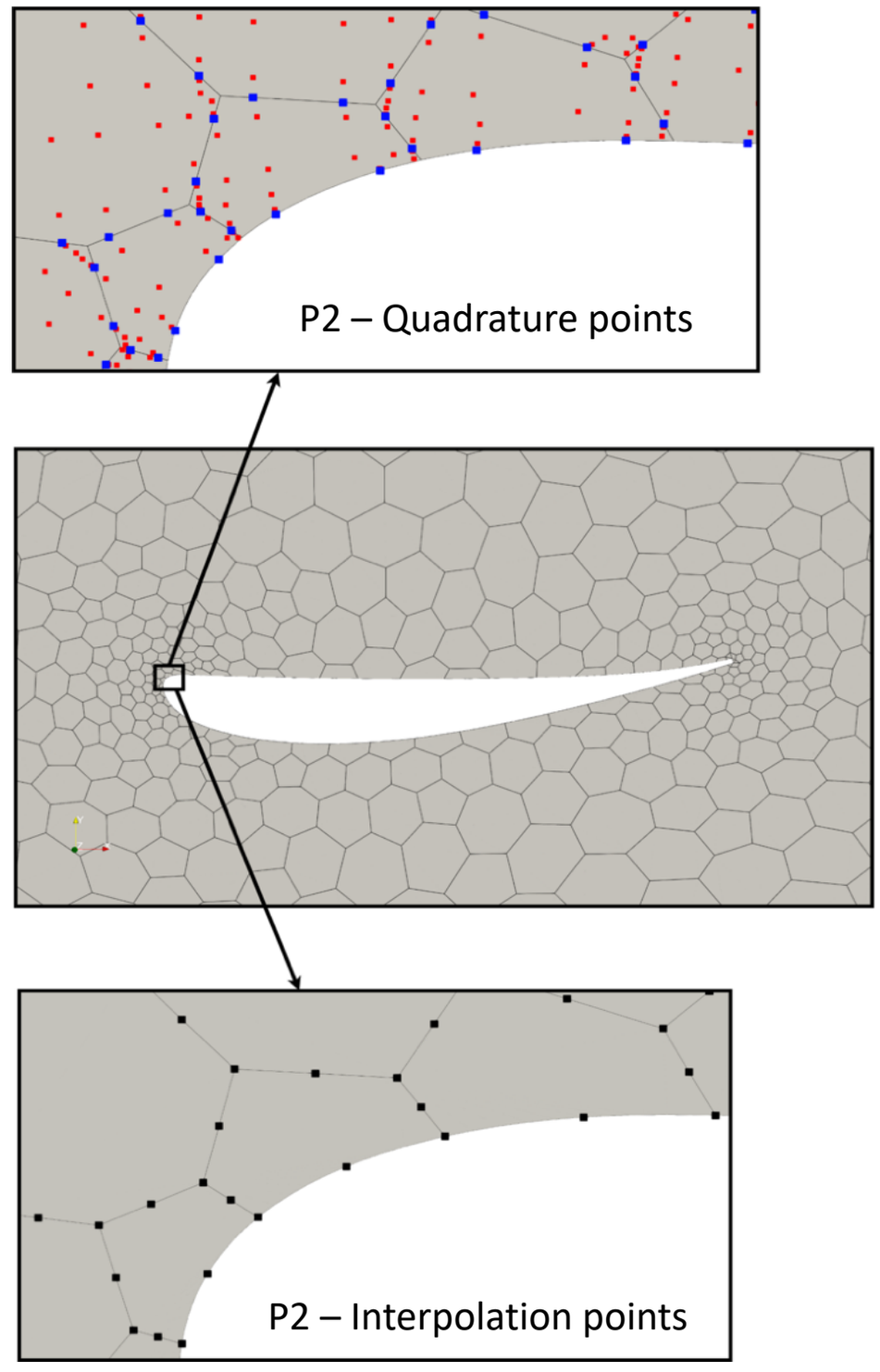
Automotive aerofoil

CAD



Linear mesh

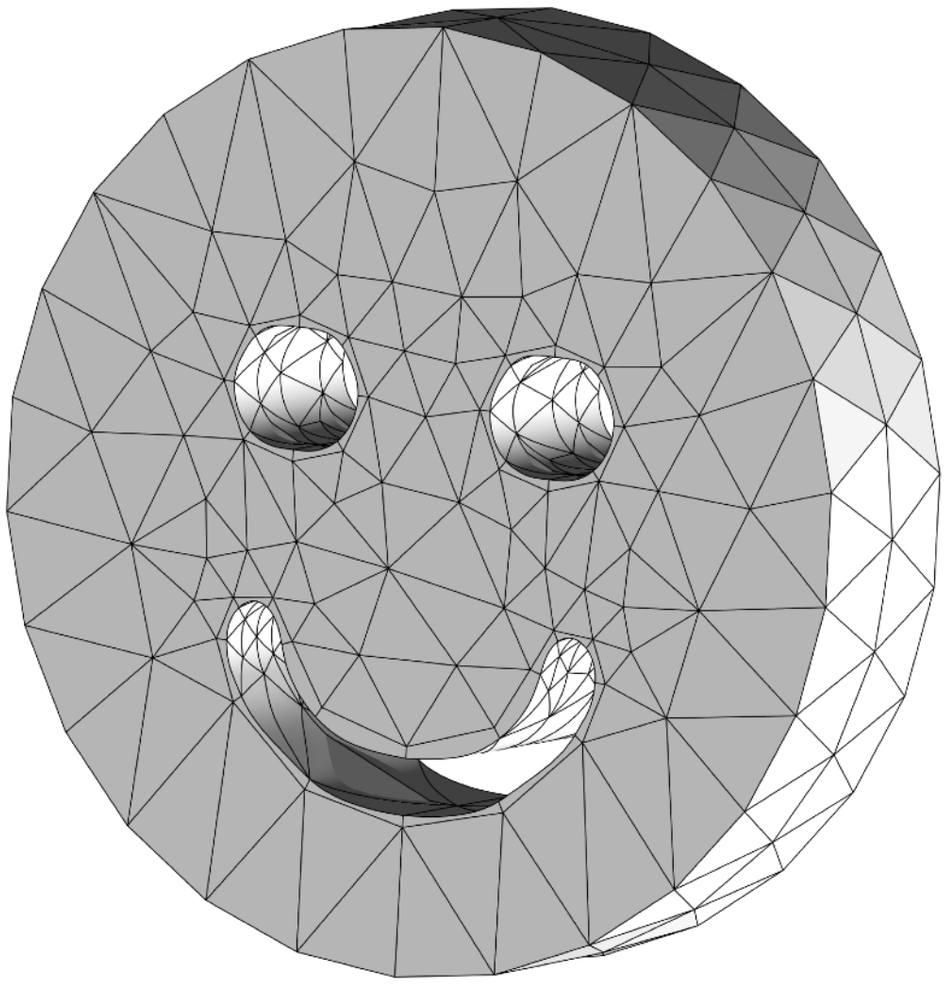




What next for HO polytopal methods?

- Streamlining of high-order methods.
- Integration rules on curvilinear polytopes.
- Robust 3D mesh generation.
- Polytopal mesh quality metrics.
- CFD implementations: hyperbolic/non-linear.
- Large-scale simulation and adaptation.
- General-purpose implementations/libraries.
- High-order FE exterior calculus.

?



http://www.nektar.info/

Acknowledgements

Kaloyan Kirilov, Jingtian Zhou and Spencer Sherwin

(Imperial College London, UK)

Mashy Green and David Moxey

(Kings College London, UK)

Lourenço Beirão da Veiga, Franco Dassi and Alessandro Russo

(University of Milano-Bicocca, Italy)

The authors of the references I have borrowed material from!

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 955923.