Numerical integration of weak forms in Embedded Interface Methods

Lehrstuhl für Numerische Mechanik

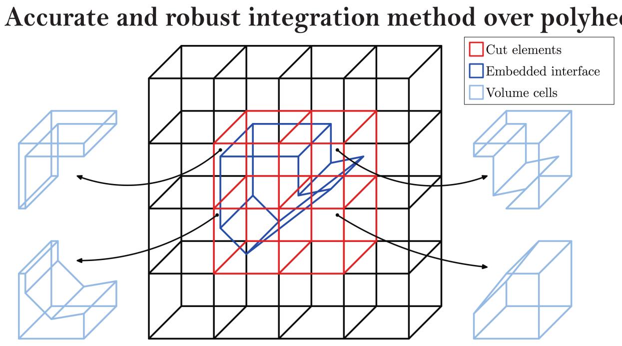
Y. Sudhakar and **Wolfgang A. Wall**

Institute for Computational Mechanics, Technische Universität München, Germany

www.lnm.mw.tum.de

Introduction

- Embedded interface introduces arbitrary polyhedral shaped volume cells
- Weak form integration over volume cells — stiffness matrix of cut elements
- Accurate and robust integration method over polyhedra is essential for EIM

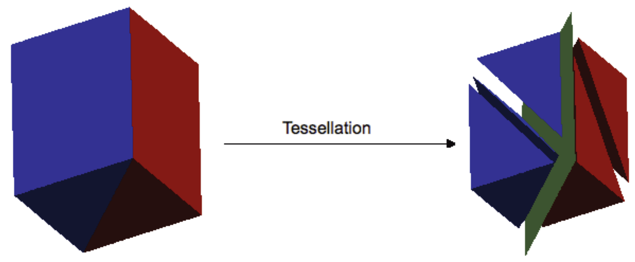


Methods available in 'Baci'

- Tessellation
- Moment fitting method
- Direct divergence method

Tessellation

- Decompose a polyhedron into a number of tetrahedra [1, 2, 3]
- Integrate over each tet and sum it up



Direct divergence method

- Use divergence theorem to convert volume integral into surface integrals [4]
- The divergence theorem

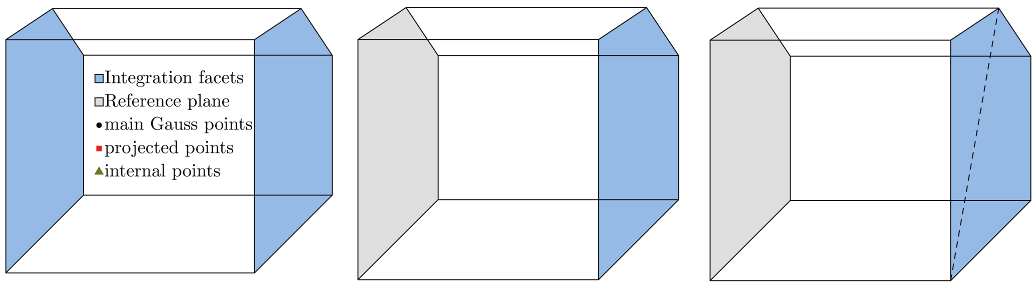
$$\int_{\mathcal{R}} \nabla\cdot\mathbf{F}\; d\mathcal{R} = \int_{\mathcal{S}} \mathbf{F}\cdot\hat{\mathbf{n}}\; d\mathcal{S}$$

- Applying it for a scalar integrand $\mathcal{F}$

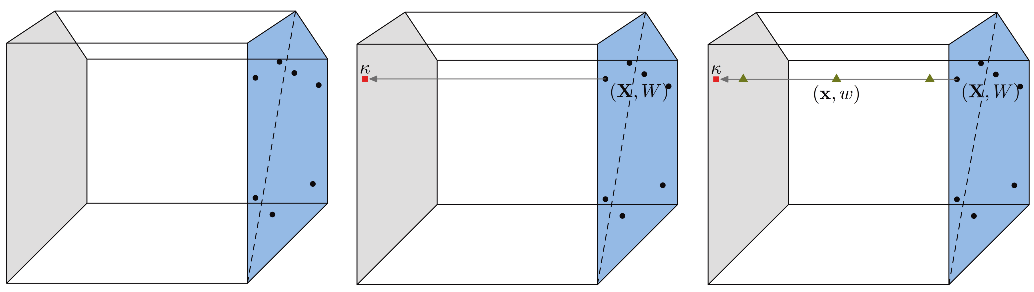
$$\int_{\mathcal{R}} \mathcal{F} d\mathcal{R} = \int_{\mathcal{S}} \mathcal{G} n_x\; d\mathcal{S} \qquad\qquad \mathcal{G} = \int \mathcal{F} dx$$

Surface integral — Line integral

- Two sets of Gauss points to evaluate the two integrals separately

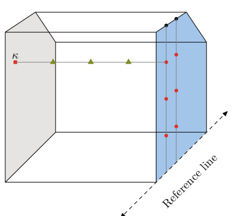


Identify integration facets — Mark reference plane — Split integration facets



Distribute main Gauss points — Project main Gauss points to reference plane — Distribute internal Gauss points

- Evaluate line integrals using internal Gauss points $\quad \mathcal{G}(\mathbf{X}_i) = \sum_{j=1}^{NI_i} \mathcal{F}(\mathbf{x}_{i,j}) w_{i,j}$
- Compute surface integrals with main Gauss points $\quad I_{\mathcal{R}} = \sum_{i=1}^{NM} \mathcal{G}(\mathbf{X}_i) n_{xi} W_i$
- Possibility of converting the surface integrals into contour integrals is avoided



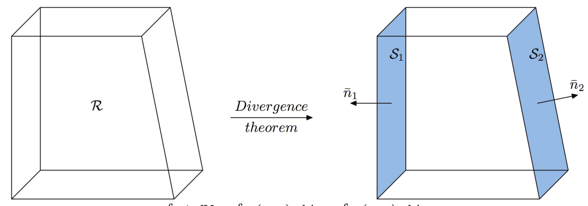
Moment fitting method

- Construct quadratures for polyhedra by solving moment fitting equations [5]

$$\begin{Bmatrix} \int_\Omega \omega(\mathbf{x})\phi_1(\mathbf{x})d\mathbf{x} \\ \int_\Omega \omega(\mathbf{x})\phi_2(\mathbf{x})d\mathbf{x} \\ \vdots \\ \int_\Omega \omega(\mathbf{x})\phi_m(\mathbf{x})d\mathbf{x} \end{Bmatrix} = \begin{bmatrix} \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \dots & \phi_1(\mathbf{x}_N) \\ \phi_2(\mathbf{x}_1) & \phi_2(\mathbf{x}_2) & \dots & \phi_2(\mathbf{x}_N) \\ \vdots & & & \\ \phi_m(\mathbf{x}_1) & \phi_m(\mathbf{x}_2) & \dots & \phi_m(\mathbf{x}_N) \end{bmatrix} \begin{Bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{Bmatrix} \quad \text{.........} \quad (1)$$

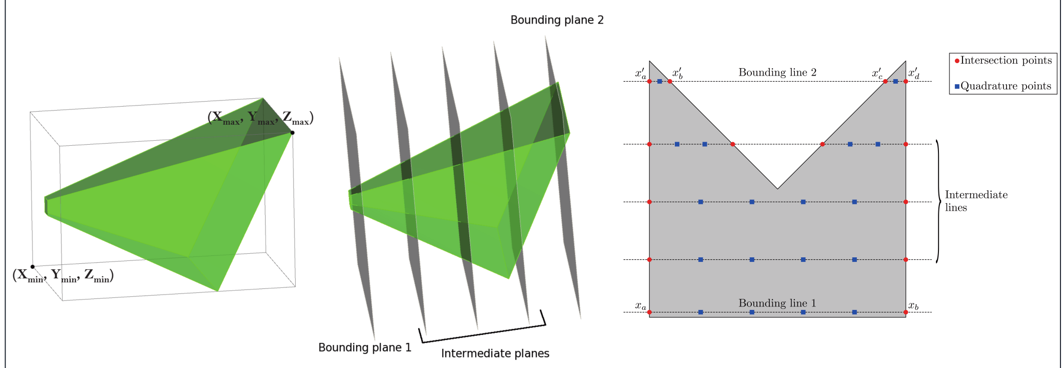
Integrate base functions — Distribute integration points — Solve for weights

- Step 1: predefine base functions $\quad \phi = \{x^i y^j z^k, i+j+k \le 6\}$
- Step 2: integrate base functions and get l.h.s of (1) using divergence theorem



$$\int_{\mathcal{R}} 1\; dV = \int_{\mathcal{S}_1} (xn_x)_1 dA_1 + \int_{\mathcal{S}_2} (xn_x)_2 dA_2$$

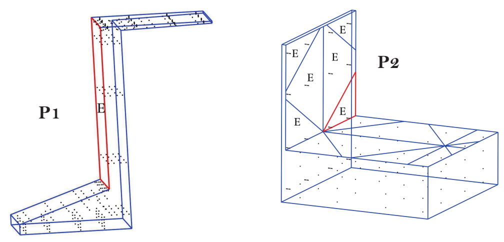
- Step 3: distribute Gauss points inside polyhedra and evaluate the matrix in (1)



- Step 4: Solve eqn (1) to obtain weights of quadrature points

Results

- **Robustness**

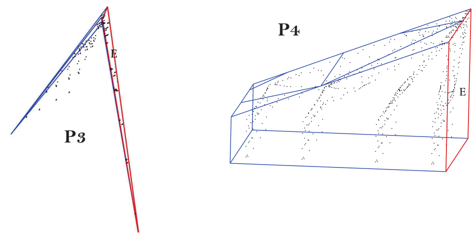


Polyhedron	Tessellation	Moment fitting	Direct divergence
P1	1248 (52 Tet)	756	752
P2	1032 (43 Tet)	1232	128
P3	48 (2 Tet)	125	220
P4	456 (19 Tet)	424	620

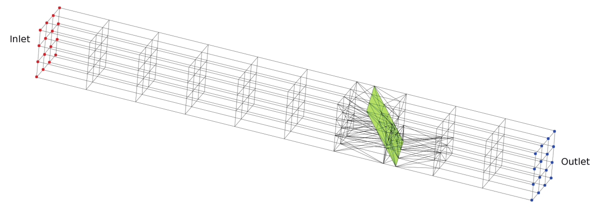
Number of integration points

Polyhedron	Tessellation	Moment fitting	Direct divergence
P1	3.7345×10^{-12}	1.5544×10^{-11}	3.7374×10^{-12}
P2	1.0472×10^{-15}	4.5992×10^{-13}	2.5132×10^{-15}
P3	1.7522×10^{-15}	4.4125×10^{-06}	6.3159×10^{-15}
P4	5.5618×10^{-11}	3.8046×10^{-11}	1.8687×10^{-13}

Error in integrating a 6th order polynomial



- **Accuracy of weak form integration**



Quantity	Tessellation	Moment fitting	Direct divergence
u	8.88×10^{-16}	2.22×10^{-10}	$< 10^{-18}$
v	9.19×10^{-16}	6.87×10^{-12}	1.11×10^{-17}
w	5.01×10^{-16}	3.95×10^{-12}	1.62×10^{-17}
p	1.23×10^{-13}	5.96×10^{-09}	8.88×10^{-15}

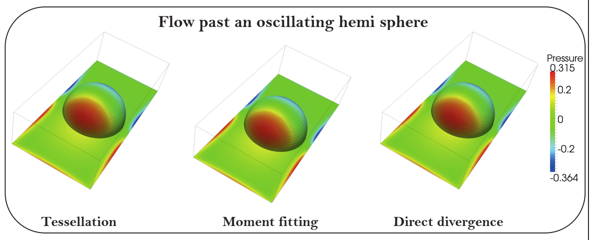
Error in weak form integration of Navier-Stokes equations

- **Computational efficiency**

	Tessellation	Moment Fitting	Direct divergence
Stationary cylinder	0.4003	3.78	0.048
Stationary star	8.422	11.97	0.2312
Oscillating hemisphere	925.71	6424.69	105.47

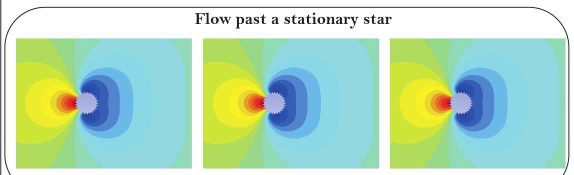
Quadrature construction time

Flow past an oscillating hemi sphere



Tessellation — Moment fitting — Direct divergence

Flow past a stationary star



Tessellation — Moment fitting — Direct divergence

	Tessellation	Moment Fitting	Direct divergence
Stationary cylinder	1.486×10^4	1.310×10^4	1.487×10^4
Stationary star	10.35×10^4	9.407×10^4	10.72×10^4
Oscillating hemisphere	12.85×10^4	13.41×10^4	14.49×10^4

Total simulation time

Conclusion

- **Tessellation**
  - computationally efficient but not robust
  - volume decomposition process fails in complex situations
- **Moment fitting method**
  - most efficient for stationary interface simulations, but expensive for moving boundary problem
  - less accurate for certain polyhedra due to ill-conditioned moment fitting equation system
- **Direct divergence method**
  - involves neither volume decomposition nor special equations
  - slightly expensive than tessellation
  - robust, accurate and easy to implement

References

[1] N. Sukumar, N. Moes, B. Moran, and T. Belytschko. International Journal for Numerical Methods in Engineering, 48:1549–1570, 2000.
[2] J. P. Peraira, C. A. Duarte, D. Guoy, and X. Jiao. International Journal for Numerical Methods in Engineering, 77:601–633, 2009.
[3] A. Gerstenberger and W. A. Wall. International Journal for Numerical Methods in Engineering, 82:537–563, 2010.
[4] Y. Sudhakar, J. P. M. D. Almeida and W. A. Wall. Journal of Computational Physics, Submitted.
[5] Y. Sudhakar and W. A. Wall. Computer Methods in Applied Mechanics and Engineering, 258:39–54, 2013.