Fast formation and assembly for spline-based fictitious domain methods

Benjamin Marussig
Institute of Applied Mechanics
Graz Center of Computational Engineering
Graz University of Technology

Outline

1. Fast formation and assembly
2. Application to fictitious domains
3. Numerical results

Fast formation and assembly – key ingredients

- Sum factorization (2D tensor product basis) (P. Antolin et al., CMAME 2015)

$$m_{\boldsymbol{i},\boldsymbol{j}} = \int_{\Omega} B_{\boldsymbol{i}}(\boldsymbol{\xi}) B_{\boldsymbol{j}}(\boldsymbol{\xi}) c(\boldsymbol{\xi}) d\boldsymbol{\xi}$$

$$= \int_{\Omega_1} B_{i_1}(\xi_1) B_{j_1}(\xi_1) \times \left[\int_{\Omega_2} B_{i_2}(\xi_2) B_{j_2}(\xi_2) c(\xi_1, \xi_2) d\xi_2 \right] d\xi_1$$

Fast formation and assembly – key ingredients

- Sum factorization (2D tensor product basis) (P. Antolin et al., CMAME 2015)

$$m_{\mathbf{i},\mathbf{j}} = \int_{\Omega} B_{\mathbf{i}}(\boldsymbol{\xi}) B_{\mathbf{j}}(\boldsymbol{\xi}) c(\boldsymbol{\xi}) d\boldsymbol{\xi}$$

$$= \int_{\Omega_1} B_{i_1}(\xi_1) B_{j_1}(\xi_1) \times \left[\int_{\Omega_2} B_{i_2}(\xi_2) B_{j_2}(\xi_2) c(\xi_1, \xi_2) d\xi_2 \right] d\xi_1$$

- Weighted quadrature $\mathbb{Q}_i$ (for smooth splines) (F. Calabrò et al., CMAME 2017)

$$\mathbb{Q} = \sum_k B_i(x_k) B_j(x_k) w_k := \int_{\Omega} B_i(\xi) B_j(\xi) d\xi$$

$$\Rightarrow \mathbb{Q}_i = \sum_k B_j(x_k) w_{k,i} := \int_{\Omega} B_j(\xi) \left(B_i(\xi) d\xi \right)$$

Fast formation and assembly – key ingredients

- Sum factorization (2D tensor product basis) (P. Antolin et al., CMAME 2015)

$$m_{\boldsymbol{i},\boldsymbol{j}} = \int_{\Omega} B_{\boldsymbol{i}}(\boldsymbol{\xi}) B_{\boldsymbol{j}}(\boldsymbol{\xi}) c(\boldsymbol{\xi}) d\boldsymbol{\xi}$$

$$= \int_{\Omega_1} B_{i_1}(\xi_1) B_{j_1}(\xi_1) \times \left[\int_{\Omega_2} B_{i_2}(\xi_2) B_{j_2}(\xi_2) c(\xi_1, \xi_2) d\xi_2 \right] d\xi_1$$

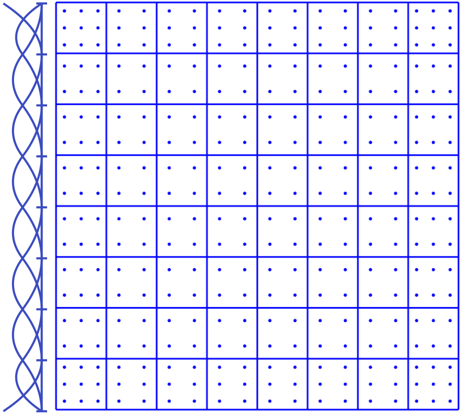
- Weighted quadrature $\mathbb{Q}_i$ (for smooth splines) (F. Calabrò et al., CMAME 2017)

$$\mathbb{Q} = \sum_k B_i(x_k) B_j(x_k) w_k := \int_{\Omega} B_i(\xi) B_j(\xi) d\xi$$

$$\Rightarrow \mathbb{Q}_i = \sum_k B_j(x_k) w_{k,i} := \int_{\Omega} B_j(\xi) \left(B_i(\xi) d\xi \right)$$

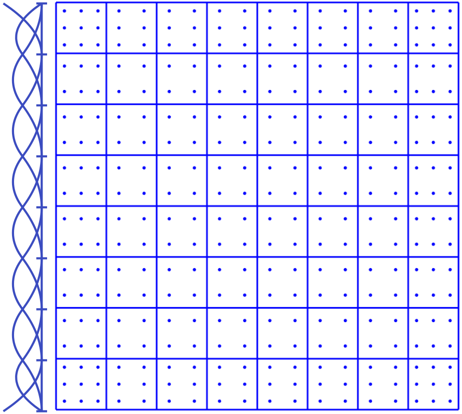
- Row/column-based assembly

Weighted quadrature point distribution

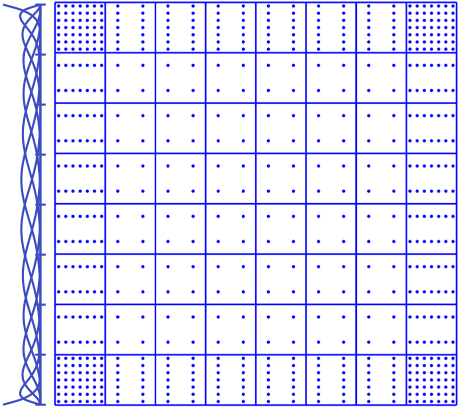


Bi-degree $p = 2$, C^{p-1} B-spline basis

Weighted quadrature point distribution

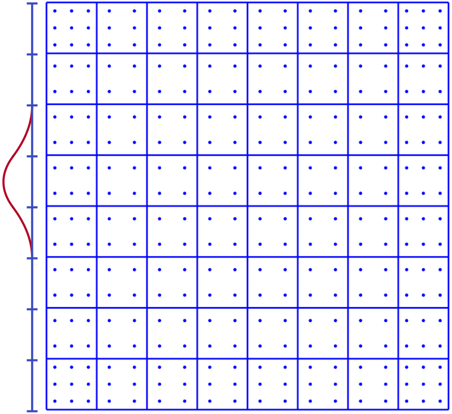


Bi-degree $p = 2$, C^{p-1} B-spline basis

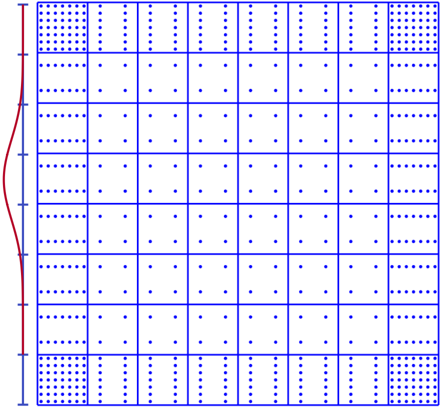


Bi-degree $p = 6$, C^{p-1} B-spline basis

Weighted quadrature point distribution



Bi-degree $p = 2$, C^{p-1} B-spline basis



Bi-degree $p = 6$, C^{p-1} B-spline basis

Fast formation and assembly – impact

FLOPS for 3D mass matrix for C^{p-1} splines

(R. Hiemstra et al., CMAME 2019)

- $\mathcal{O}(p^3)$ quadrature points per element

Assembly \ Formation	Quadrature loop	Sum factorization
Element loop	$c \cdot p^9$	$c_1 \cdot p^5 + c_2 \cdot p^6 + c_3 \cdot p^7$
Row/column loop	$c \cdot p^9$	$c_1 \cdot p^7 + c_2 \cdot p^6 + c_3 \cdot p^5$

Fast formation and assembly – impact

FLOPS for 3D mass matrix for C^{p-1} splines

(R. Hiemstra et al., CMAME 2019)

- $\mathcal{O}(p^3)$ quadrature points per element

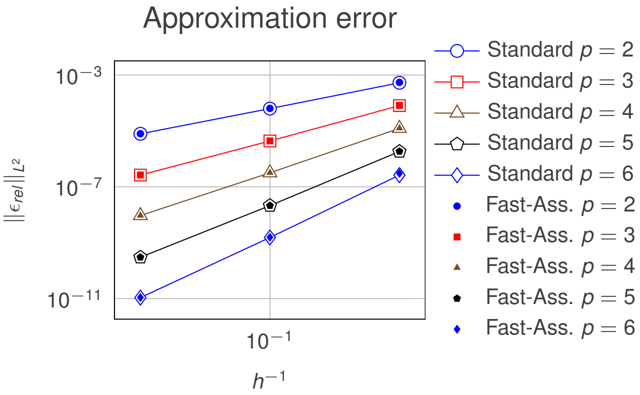
Assembly \ Formation	Quadrature loop	Sum factorization
Element loop	$c \cdot p^9$	$c_1 \cdot p^5 + c_2 \cdot p^6 + c_3 \cdot p^7$
Row/column loop	$c \cdot p^9$	$c_1 \cdot p^7 + c_2 \cdot p^6 + c_3 \cdot p^5$

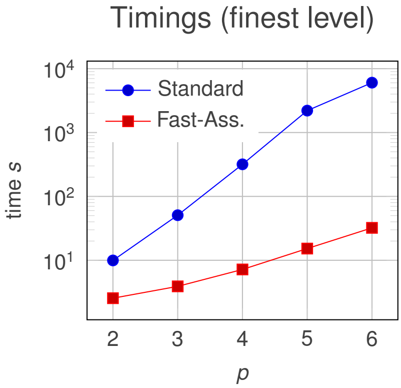
- $\mathcal{O}(1)$ quadrature points per element

Assembly \ Formation	Quadrature loop	Sum factorization
Element loop	$c \cdot p^6$	$c_1 \cdot p^2 + c_2 \cdot p^4 + c_3 \cdot p^6$
Row/column loop	$c \cdot p^6$	$c_1 \cdot p^4 + c_2 \cdot p^4 + c_3 \cdot p^4$

Fast formation and assembly – impact

Single threat formation of a 3D mass matrix



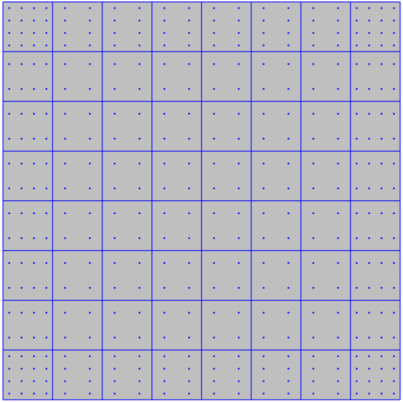


Outline

1. Fast formation and assembly
2. Application to fictitious domains
3. Numerical results

The challenge of fictitious domains

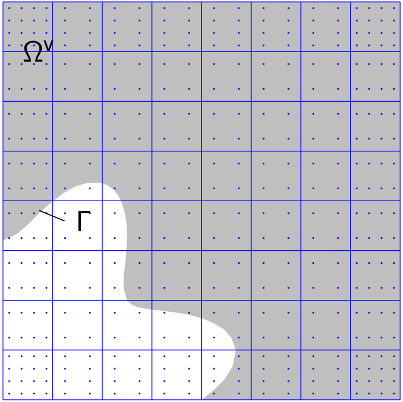
- Sum factorization and weighted quadrature take full advantage of the tensor product structure ...



Bi-cubic C^{p-1} background mesh

The challenge of fictitious domains

- Sum factorization and weighted quadrature take full advantage of the tensor product structure ...
- ... which is violated by an arbitrarily located interface Γ.

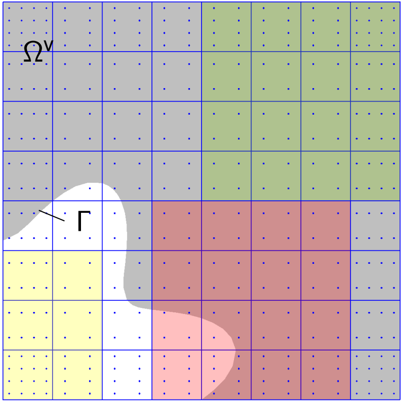


Cut bi-cubic C^{p-1} background mesh

Function types of cut background meshes

Classification by valid support size $\mathcal{S}_i^{\mathrm{v}} := \mathrm{supp}\{B_i\} \cap \overline{\Omega^{\mathrm{v}}}$

- **Exterior** if $\mathcal{S}_i^{\mathrm{v}} = \emptyset$
- **Interior** if $\mathcal{S}_i^{\mathrm{v}} = \mathrm{supp}\{B_i\}$
- **Cut** if $0 < |\mathcal{S}_i^{\mathrm{v}}| < |\mathrm{supp}\{B_i\}|$



Cut bi-cubic C^{p-1} background mesh

Integration of cut basis functions

Split of the basis function's valid support $\mathcal{S}_i^{\mathrm{v}}$ into...

$$\int_{\mathcal{S}_i^{\mathrm{v}}} B_i(\xi)d\xi = \int_{\mathcal{S}_i^{\mathrm{c}}} B_i(\xi)d\xi + \int_{\mathcal{S}_i^{\mathrm{r}}} B_i(\xi)d\xi$$

Integration of cut basis functions

Split of the basis function's valid support $\mathcal{S}_i^{\text{v}}$ into...

$$\int_{\mathcal{S}_i^{\text{v}}} B_i(\xi) d\xi = \boxed{\int_{\mathcal{S}_i^{\text{c}}} B_i(\xi) d\xi} + \int_{\mathcal{S}_i^{\text{r}}} B_i(\xi) d\xi$$

- **Cut** part $\mathcal{S}_i^{\text{c}}$:
  Element-wise assembly using **higher-order accurate** quadrature
  (e.g., T.-P. Fries et al., CMAME 2017; R.I. Saye, SISC 2015; R.I. Saye, J. Comput. Phys. 2022)

Integration of cut basis functions

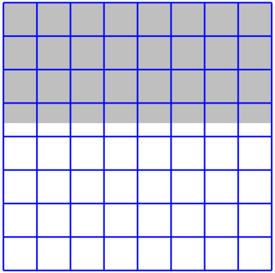
Split of the basis function's valid support $\mathcal{S}_{\boldsymbol{i}}^{\text{v}}$ into...

$$\int_{\mathcal{S}_{\boldsymbol{i}}^{\text{v}}} B_{\boldsymbol{i}}(\boldsymbol{\xi})d\boldsymbol{\xi} = \boxed{\int_{\mathcal{S}_{\boldsymbol{i}}^{\text{c}}} B_{\boldsymbol{i}}(\boldsymbol{\xi})d\boldsymbol{\xi}} + \boxed{\int_{\mathcal{S}_{\boldsymbol{i}}^{\text{r}}} B_{\boldsymbol{i}}(\boldsymbol{\xi})d\boldsymbol{\xi}}$$

- **Cut** part $\mathcal{S}_{\boldsymbol{i}}^{\text{c}}$:
  Element-wise assembly using **higher-order accurate** quadrature
  (e.g., T.-P. Fries et al., CMAME 2017; R.I. Saye, SISC 2015; R.I. Saye, J. Comput. Phys. 2022)
- **Regular** part $\mathcal{S}_{\boldsymbol{i}}^{\text{r}}$:
  Integration that **exploits the tensor product structure**

Integration of $\mathcal{S}_i^r$: weighted quadrature (WQ)

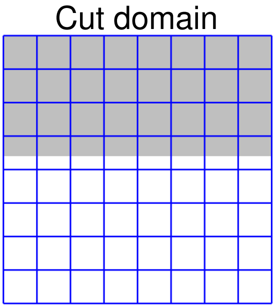
Cut domain



Mass matrix deviation

p	Error
1	3.330e-03
2	3.389e-04
3	3.038e-04
4	1.782e-04
5	1.599e-04
6	1.282e-04

Integration of $\mathcal{S}_i^r$: weighted quadrature (WQ)



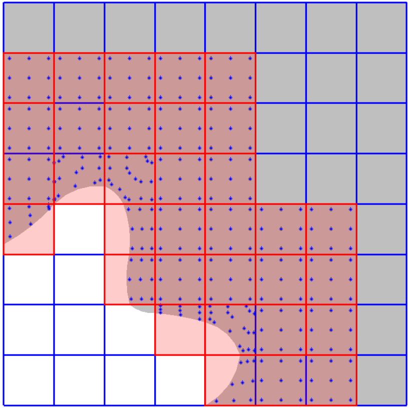
Mass matrix deviation

p	Error
1	3.330e-03
2	3.389e-04
3	3.038e-04
4	1.782e-04
5	1.599e-04
6	1.282e-04

Straightforward use of weighted quadrature is **not** possible since a **discontinuity** is introduced in the integration domain.

Integration of $\mathcal{S}_i^r$: Gauss quadrature (GQ)

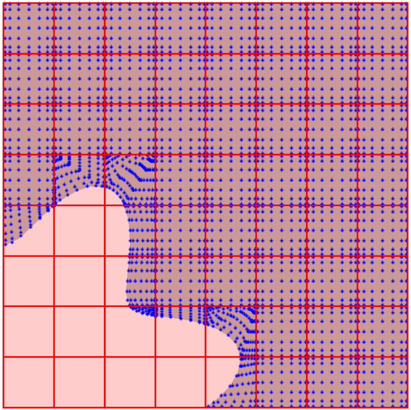
Gauss points for all cut basis functions



Bi-degree $p = 2$

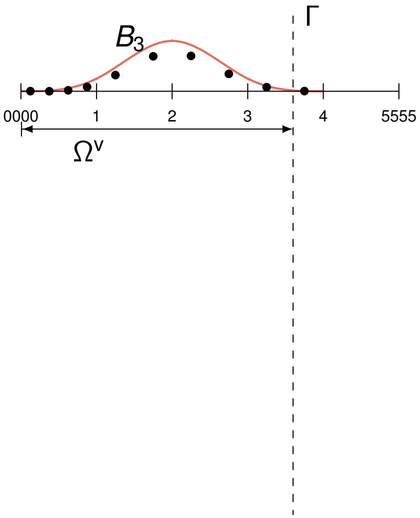
Integration of $\mathcal{S}_i^r$: Gauss quadrature (GQ)

Gauss points for all cut basis functions



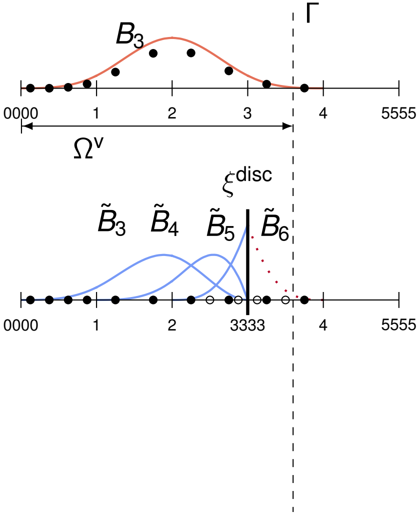
Bi-degree $p = 6$

Discontinuous weighted quadrature (DWQ)



Cut B-spline and
its WQ rule

Discontinuous weighted quadrature (DWQ)

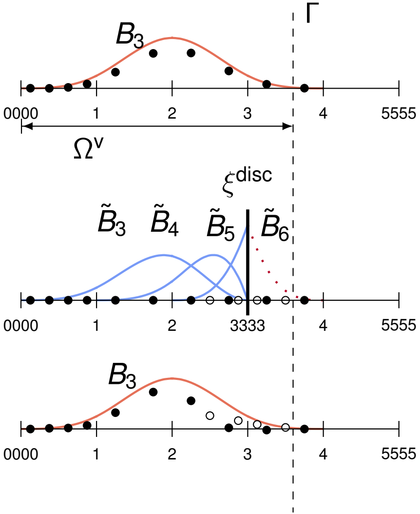


Cut B-spline and its WQ rule

Knot insertion at ξ^{disc} and subdivision matrix **S**

Disc. basis and additional nested quadrature points

Discontinuous weighted quadrature (DWQ)



Cut B-spline and its WQ rule

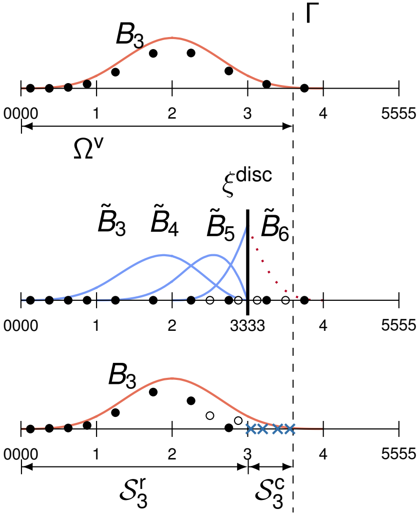
Knot insertion at ξ^{disc} and subdivision matrix **S**

Disc. basis and additional nested quadrature points

Compute weights $\tilde{w}$ and multiply them by $\mathbf{S}^T$

Initial cut B-spline and its DWQ rule

Discontinuous weighted quadrature (DWQ)



Cut B-spline and its WQ rule

Knot insertion at ξ^{disc} and subdivision matrix $\mathbf{S}$

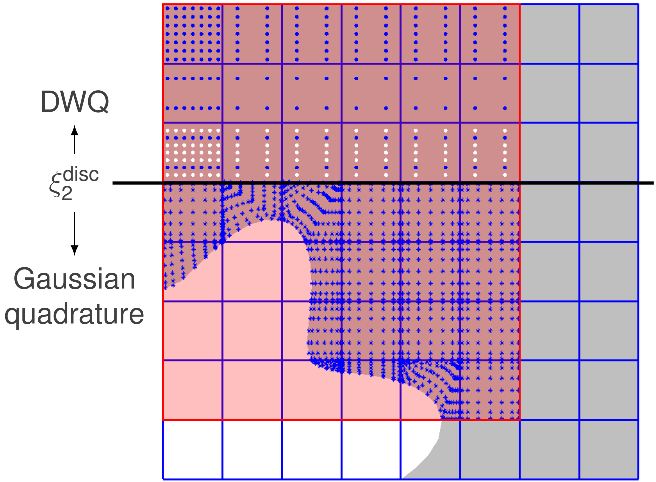
Disc. basis and additional nested quadrature points

Compute weights $\tilde{w}$ and multiply them by $\mathbf{S}^{\mathrm{T}}$

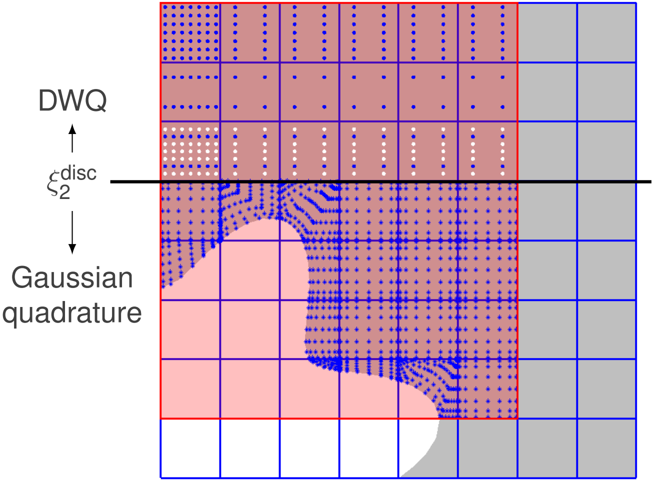
Initial cut B-spline and its DWQ rule

- • DWQ point
- × Gauss point

Discontinuous weighted quadrature (DWQ)



Discontinuous weighted quadrature (DWQ)



- Implementation remark: Use Gauss point locations for DWQ to reuse evaluations

Outline

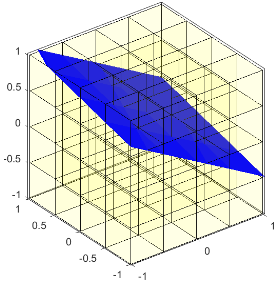
1. Fast formation and assembly
2. Application to fictitious domains
3. Numerical results

3D mass matrix

- L^2-projection of the target function

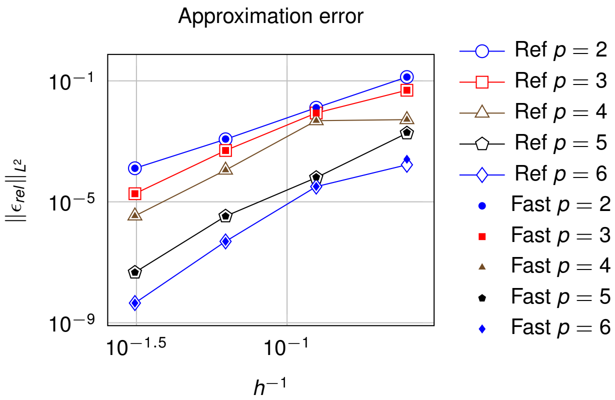
$$f = \sin(2xz)\cos(3yz)$$

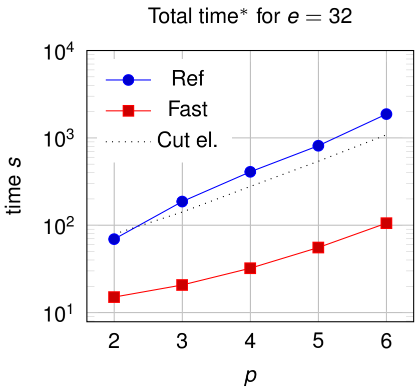
- Degrees $p = \{2, \ldots, 6\}$
- Elements per dimension $e = \{4, 8, 16, 32\}$
- Ill-conditioning due to cut elements is treated by extended B-splines (K. Höllig et al., SINUM 2002)



Domain of interest

3D mass matrix: Convergence and timings





* Timings without cut elements

Conclusions

Fast formation and assembly key ingredients:

- Spline-based background mesh
- Sum factorization (P. Antolin et al., CMAME 2015)
- Row/column assembly
- (Discontinuous[1]) weighted quadrature (F. Calabrò et al., CMAME 2017)

[1] BM, Fast formation and assembly of isogeometric Galerkin matrices for trimmed patches, INdAM Series 2021

Conclusions

Fast formation and assembly key ingredients:

- Spline-based background mesh
- Sum factorization (P. Antolin et al., CMAME 2015)
- Row/column assembly
- (Discontinuous[1]) weighted quadrature (F. Calabrò et al., CMAME 2017)

Impact

- Significant reduction of FLOPS for setting up system matrices
- Overall performance depends on degree, interface shape, and mesh size

[1] BM, Fast formation and assembly of isogeometric Galerkin matrices for trimmed patches, INdAM Series 2021

Conclusions

Fast formation and assembly key ingredients:

- Spline-based background mesh
- Sum factorization (P. Antolin et al., CMAME 2015)
- Row/column assembly
- (Discontinuous[1]) weighted quadrature (F. Calabrò et al., CMAME 2017)

Impact

- Significant reduction of FLOPS for setting up system matrices
- Overall performance depends on degree, interface shape, and mesh size

Potential drawbacks

- Weighted quadrature does not preserve symmetry
- Code requires a completely new structure
- Many quantities have to be pre-computed for sum factorization

[1] BM, Fast formation and assembly of isogeometric Galerkin matrices for trimmed patches, INdAM Series 2021

Fast formation and assembly for spline-based fictitious domain methods

Benjamin Marussig
Institute of Applied Mechanics
Graz Center of Computational Engineering
Graz University of Technology