

Jan Valášek; Petr Sváček

Interpolation with restrictions – role of the boundary conditions and individual restrictions

In: Jan Chleboun and Pavel Kůs and Jan Papež and Miroslav Rozložník and Karel Segeth and Jakub Šístek (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Jablonec nad Nisou, June 19-24, 2022. Institute of Mathematics CAS, Prague, 2023. pp. 281–292.

Persistent URL: <http://dml.cz/dmlcz/703208>

Terms of use:

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library*
<http://dml.cz>

INTERPOLATION WITH RESTRICTIONS — ROLE OF THE BOUNDARY CONDITIONS AND INDIVIDUAL RESTRICTIONS

Jan Valášek¹, Petr Sváček²

¹ Institute of Mathematics, Czech Academy of Sciences
Žitná 25, 115 67 Praha 1, Czech Republic
valasek@math.cas.cz

² Faculty of Mechanical Engineering, CTU in Prague
Karlovo nám. 13, Praha 2, 121 35, Czech Republic
petr.svacek@fs.cvut.cz

Abstract: The contribution deals with the remeshing procedure between two computational finite element meshes. The remeshing represented by the interpolation of an approximate solution onto a new mesh is needed in many applications like e.g. in aeroacoustics, here we are particularly interested in the numerical flow simulation of a gradual channel collapse connected with a severe deterioration of the computational mesh quality.

Since the classical Lagrangian projection from one mesh to another is a dissipative method not respecting conservation laws, a conservative interpolation method introducing constraints is described. The constraints have form of Lagrange multipliers enforcing conservation of desired flow quantities, like e.g. total fluid mass, flow kinetic energy or flow potential energy. Then the interpolation problem turns into an error minimization problem, such that the resulting quantities of proposed interpolation satisfy these physical properties while staying as close as possible to the results of Lagrangian interpolation in the L2 norm. The proposed interpolation scheme does not impose any restrictions on mesh generation process and it has a relatively low computational cost. The implementation details are discussed and test cases are shown.

Keywords: interpolation, Lagrange multiplier, Lagrange projection

MSC: 65D05, 65M60

1. Introduction

Interpolation is one of the basic mathematical problems and therefore there are plenty of available methods. Here we consider an interpolation procedure between two 2D computational finite element meshes involved during the remeshing step. This is a typical task in engineering simulations of cutting, forging, casting, welding,

(see e.g. [1]), where material is processed and reshaped, or in multiphysics simulations like geophysics, aeroacoustics or fluid-structure interaction (FSI), see e.g. [7]. Particularly, motivation for this paper is provided by the FSI problem of flow-induced vibrations of vocal folds studied in [9, 8]. The implemented ALE method during large vibrations was not able to provide a computational flow mesh of sufficient quality and thus the remeshing is needed, see [9].

As the base of the available interpolation methods the scattered data interpolations can be regarded. Such approaches are realized in many packages as e.g. Matlab, SciPy. Another possibility available also for higher dimensional cases and unstructured grids is the use of the radial basis function approach, see e.g. [6]. Further, there are methods specially suited for ALE methods, see e.g. [4]. However, they are designed for meshes with the same topology based on the computation of the local fluxes. Another approach is represented by so called supermesh approach, see e.g. [2], where a superior mesh given by mesh intersections is constructed what results in a high computational cost albeit it guarantees a L2 accurate projection. More computationally favourable approach of [1] replaces supermesh approach used together with Galerkin projection by an approximate evaluation of involved integrals, where a relative lack of precise intersection information should be compensated by increase of number of quadrature points. Nevertheless the most suitable method for our purpose is the idea of [5] to combine a cheap interpolation with supplementary restrictions typically chosen such that conservation of quantities from the physical nature of investigated problem is required. Let us call this approach as interpolation with restrictions or Codina & Pont interpolation (CPI). This method has a great advantage of satisfying physical laws (in global meaning) what is a typical disadvantage of other methods which results do not respect physical laws. Disadvantage is that restrictions, i.e. conservation of selected quantities, are not valid locally.

Thus we will further deal only with the interpolation with restrictions, see [5], and we will focus on behaviour of this method near domain boundaries. Our aim is to improve CPI by using further information from boundaries, i.e. we assume that new target FE mesh occupies the same space as the old donor FE mesh and further that the vertex locations of the old and the new mesh on the mesh boundaries are identical. This assumption is motivated by implementation of our in-house FSI solver, [9, 8]. Then two methods of boundaries values treatment are compared and the interpolation error for case of small highly distorted domain contrary to case of larger domain with smaller distortion is calculated (motivated by different settings during construction of ALE mapping).

The structure of the paper is following. First the interpolation with restrictions is described and applied for the case of fluid flow. Further the implementation details are presented. Finally the errors of different interpolation settings are analyzed and summarized in conclusion.

2. Interpolation with restrictions

In the whole paper we consider two triangulations \mathcal{T}^o and \mathcal{T}^n of the same bounded physical domain Ω of \mathbb{R}^2 , see Figure 1, which moreover satisfy that their boundary vertices are identical. Here, \mathcal{T}^o is called the old (donor) mesh and \mathcal{T}^n is the new (target) mesh. By \mathcal{V}_h^o and \mathcal{V}_h^n the corresponding FE spaces constructed over the triangulations \mathcal{T}^o and \mathcal{T}^n are denoted, respectively. Further, we denote a FE function from FE space \mathcal{V}_h^o constructed over the FE mesh \mathcal{T}^o by u_h^o , i.e. $u_h^o(x) = \sum_j U_j^o \psi_j^o(x)$, where $\psi_j^o(x)$ are basis functions of the FE space \mathcal{V}_h^o and U_j^o are corresponding nodal values. Similarly, a function from \mathcal{V}_h^n can be written as $u_h^n(x) = \sum_k U_k^n \psi_k^n(x) \in \mathcal{V}_h^n$.

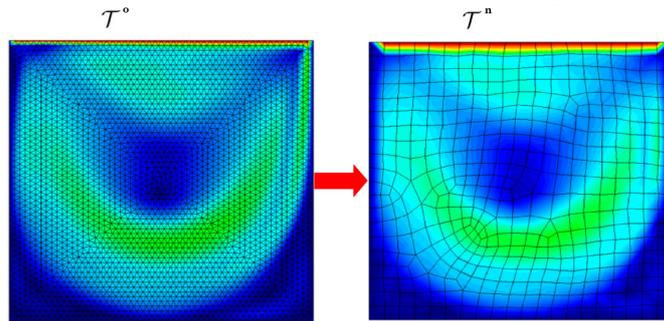


Figure 1: Illustration of interpolation from old to new FE mesh, [5].

2.1. Key idea of the method

The general procedure of interpolation with restrictions, see [5], is based on two steps. During the first step a function $u_h^o \in \mathcal{V}_h^o$ defined on the old mesh \mathcal{T}^o is projected on the new mesh. The commonly used projections are either Lagrange or Galerkin projections, [5]. The first one, the Lagrange projection, is based on the evaluation of the values U_A^n given by

$$U_A^n = u_h^n(X_A^n) = \sum_j U_j^o \psi_j^o(X_A^n), \quad (1)$$

where X_A^n denotes the coordinates of the point associated with the nodal value U_A^n . In the second (Galerkin) case, the L2 projection is applied leading to the integral identity

$$\int_{\Omega} u_h^n \psi^n dx = \int_{\Omega} u_h^o \psi^n dx \quad \forall \psi^n \in \mathcal{V}_h^n. \quad (2)$$

In order to precisely fulfill (2) one needs to compute elements intersections. Such a procedure can be computationally demanding and requires additional techniques to be applied as e.g. supermesh approach used in [2]. High computational costs of the Galerkin approach can be reportedly reduced by using numerical quadrature of high orders, see [1]. As this phenomenon was not observed for the considered numerical tests, the use of the Lagrangian interpolation is preferred.

As one of the biggest interpolation problems is the violation of physical nature of interpolated variable, see e.g. [4], in the second step appropriate restrictions are applied as a correction step of the projection. The idea of imposing additional restrictions with the help of Lagrangian multipliers is a key how to conserve quantities of the interest (in global sense). The presented two steps of the CPI algorithm is general and it can be potentially used in many different scenarios, [5]. The disadvantage of CPI is that local conservation of desired quantities is not guaranteed.

2.2. Application to fluid flow problem

The previous general concept is now applied for incompressible fluid flow problem with the constant density ρ . In this context we will use following notation: $\mathbf{v}^o \in \mathbf{V}_h^o = \mathcal{V}_h^o \times \mathcal{V}_h^o$ for the given velocity defined on the old mesh \mathcal{T}^o , $\tilde{\mathbf{v}}^n = \Pi_h \mathbf{v}^o \in \mathbf{V}_h^n$ for the Lagrangian projection of \mathbf{v}^o on the new mesh \mathcal{T}^n and \mathbf{v}^n for the sought interpolation with restrictions on the target mesh \mathcal{T}^n . The interpolation procedure is now described.

Based on the nature of the problem we impose conservation of the following quantities: 1) mass (through the conservation of the velocity divergence), 2) linear momenta and 3) kinetic energy. This leads to the following four restrictions:

$$\begin{aligned} 1) \quad & \int_{\Omega} \nabla \cdot \mathbf{v}^o \, dx = \int_{\Omega} \nabla \cdot \mathbf{v}^n \, dx, & 2) \quad & \int_{\Omega} \rho \mathbf{v}^o \cdot \mathbf{e}_i \, dx = \int_{\Omega} \rho \mathbf{v}^n \cdot \mathbf{e}_i \, dx, \quad i = \{1, 2\}, \\ 3) \quad & \frac{1}{2} \int_{\Omega} \rho |\mathbf{v}^o|^2 \, dx = \frac{1}{2} \int_{\Omega} \rho |\mathbf{v}^n|^2 \, dx, \end{aligned} \quad (3)$$

where vectors \mathbf{e}_i denotes standard basis. In what follows we set $\rho = 1$.

Then the problem of interpolation with restrictions reads: For the given velocity $\mathbf{v}^o \in \mathbf{V}_h^o$ find

$$[\mathbf{v}^n, \boldsymbol{\lambda}] = \arg \inf_{\mathbf{u}^n \in \mathbf{V}_h^n} \sup_{\boldsymbol{\mu} \in \mathbb{R}^4} L(\mathbf{u}^n, \boldsymbol{\mu}), \quad (4)$$

where $\boldsymbol{\mu}$ are Lagrangian multipliers and $L(\mathbf{u}^n, \boldsymbol{\mu})$ is Lagrangian function defined as

$$\begin{aligned} L(\mathbf{u}^n, \boldsymbol{\mu}) = & \frac{1}{2} \int_{\Omega} \left(\sum_k (U_k^n - \tilde{U}_k^n) \boldsymbol{\psi}_k^n \right)^2 \, dx - \mu_1 \int_{\Omega} \nabla \cdot \left(\sum_k U_k^n \boldsymbol{\psi}_k^n - \sum_j U_j^o \boldsymbol{\psi}_j^o \right) \, dx \\ & - \sum_{l=1}^2 \mu_l \int_{\Omega} \left(\sum_k U_k^n \boldsymbol{\psi}_k^n - \sum_j U_j^o \boldsymbol{\psi}_j^o \right) \cdot \mathbf{e}_l \, dx \\ & - \frac{\mu_4}{2} \int_{\Omega} \left(\sum_k U_k^n \boldsymbol{\psi}_k^n \right)^2 - \left(\sum_j U_j^o \boldsymbol{\psi}_j^o \right)^2 \, dx. \end{aligned} \quad (5)$$

The differentiation of the function L with respect to all unknowns U_i^n yields

$$\begin{aligned} \int_{\Omega} \sum_k U_k^n \boldsymbol{\psi}_k^n \boldsymbol{\psi}_i^n dx - \mu_1 \int_{\Omega} \nabla \cdot \boldsymbol{\psi}_i^n dx - \sum_{l=1}^2 \mu_l \int_{\Omega} \boldsymbol{\psi}_i^n dx \\ - \mu_4 \int_{\Omega} \sum_k U_k^n \boldsymbol{\psi}_k^n \boldsymbol{\psi}_i^n dx = \int_{\Omega} \sum_k \tilde{U}_k^n \boldsymbol{\psi}_k^n \boldsymbol{\psi}_i^n dx, \end{aligned} \quad (6)$$

and by the differentiation of L with respect to μ_i together with the condition given by Eq. (4) we get

$$\begin{aligned} \int_{\Omega} \nabla \cdot \left(\sum_k U_k^n \boldsymbol{\psi}_k^n \right) dx &= \int_{\Omega} \nabla \cdot \left(\sum_j U_j^o \boldsymbol{\psi}_j^o \right) dx, \\ \int_{\Omega} \left(\sum_k U_k^n \boldsymbol{\psi}_k^n \cdot \mathbf{e}_l \right) dx &= \int_{\Omega} \left(\sum_j U_j^o \boldsymbol{\psi}_j^o \cdot \mathbf{e}_l \right) dx, \quad l = \{1, 2\}, \\ \int_{\Omega} \left(\sum_k U_k^n \boldsymbol{\psi}_k^n \right)^2 dx &= \int_{\Omega} \left(\sum_j U_j^o \boldsymbol{\psi}_j^o \right)^2 dx. \end{aligned} \quad (7)$$

Previous equations written in the matrix notation reads

$$\begin{pmatrix} \mathbb{M}^n & -R_1 & -R_2 & -R_3 & -\mathbb{M}^n \mathbf{U}^n \\ R_1^T & 0 & 0 & 0 & 0 \\ R_2^T & 0 & 0 & 0 & 0 \\ R_3^T & 0 & 0 & 0 & 0 \\ (\mathbb{M}^n \mathbf{U}^n)^T & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{U}^n \\ \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{pmatrix} = \begin{pmatrix} \mathbb{M}^n \tilde{\mathbf{U}}^n \\ R_1^o \mathbf{U}^o \\ R_2^o \mathbf{U}^o \\ R_3^o \mathbf{U}^o \\ (\mathbf{U}^o)^T \mathbb{M}^o \mathbf{U}^o \end{pmatrix}, \quad (8)$$

where \mathbb{M}^n denotes mass matrix with components $m_{ij}^n = \int_{\Omega} \boldsymbol{\psi}_j^n \boldsymbol{\psi}_i^n dx$, \mathbb{M}^o is the mass matrix defined on the old mesh \mathcal{T}^o and vectors R_1, R_2, R_3 are given componentwise by

$$(R_1)_i = \int_{\Omega} \nabla \cdot \boldsymbol{\psi}_i^n dx, \quad (R_2)_i = \int_{\Omega} \boldsymbol{\psi}_i^n \cdot \mathbf{e}_1 dx, \quad (R_3)_i = \int_{\Omega} \boldsymbol{\psi}_i^n \cdot \mathbf{e}_2 dx. \quad (9)$$

Vectors R_i^o , $i \in \{1, 2, 3\}$ are defined similarly on the old mesh. Since problem (8) is nonlinear the Newton-Rhapson method is used for its numerical solution, see [5].

Pressure. The same concept is also used for the interpolation of the pressure obtained by the solution of the Navier-Stokes equations. In this case only the conservation of its L2 norm is considered.

3. Implementation

Although problem (8) has a saddle point structure the most computationally demanding part is the computation of the Lagrange projection. It is due to the

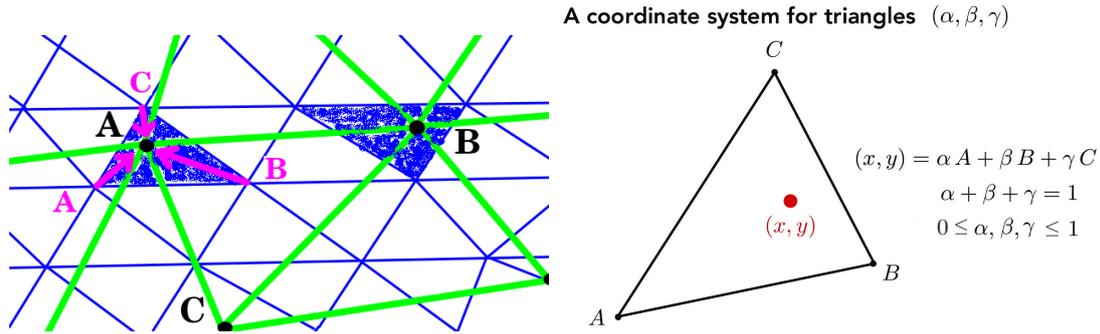


Figure 2: Illustration of Lagrange interpolation from the old FE mesh \mathcal{T}^o (blue) to the new FE mesh \mathcal{T}^n (green) with its vertices plotted in black colour. The filled blue triangles highlight the area, from which the final value in vertices A and B are computed.

necessity to find locations of the vertices X_k^n from \mathcal{T}^n in terms of the old mesh \mathcal{T}^o in order to evaluate $\psi^o(X_k^n)$ in Eq. (1). The way towards it is to determine at which triangles from \mathcal{T}^o points X_k^n lie and to find their barycentric coordinates inside these triangles, see Fig. 2. Then evaluation of Eq. (1) is straightforward.

There are more possible methods how to find such locations. In the work of [5] the octree parallel algorithm was employed, another possibility offers advancing front techniques, see e.g. [3]. Nevertheless here we adopted the procedure based on the computation of barycentric coordinates as it is implemented in software Octave. The algorithm is following: First prepare the list \mathcal{X} of vertices X_k^n of \mathcal{T}^n . Then in a loop over all triangles $T_i^o \in \mathcal{T}^o$ determine which points from the list \mathcal{X} lie in T_i^o :

1. Compute the barycentric coordinates $\alpha_j, \beta_j, \gamma_j$ for each $X_j^n \in \mathcal{X}$ by solving 3×3 matrix system with M right hand vectors, where M is the length of list \mathcal{X} .
2. If $0 \leq \alpha_j, \beta_j, \gamma_j \leq 1$ and $\alpha_j + \beta_j + \gamma_j = 1$ then point X_j^n belongs to triangle T_i^o . Save its barycentric coordinates and shorten list \mathcal{X} .

Complexity of this approach is almost quadratic, on the other hand this procedure can be well parallelized. Further, a division of list \mathcal{X} in short sub-list based e.g. on conditions $x \geq x_0, y \geq 0$ can speed up the algorithm.

4. Numerical simulations

Two tests of the interpolation with restrictions are performed.

4.1. First interpolation test – question of boundary values

The modified interpolation test of [1, 5] shows how an additional information from the boundary can improve the interpolation results. Let have a divergence-free function $\mathbf{F}(x, y)$ with the components $f_1(x, y) = 2x^2(x-1)^2y(y-1)(2y-1)$, $f_2(x, y) = -2y^2(y-1)^2x(x-1)(2x-1)$ and the donor and the target triangular meshes of the

domain $\langle 0, 1.1 \rangle^2$ with identical vertices at the boundary. Both meshes has the characteristic length $h = 0.033$ and the inner vertices of the target mesh are shifted by $h/2$ to the right. In total 20 pairs of interpolations between these meshes are performed and four different interpolation variants are compared. The first two are the classical Lagrange projection (LAG) and the interpolation with restrictions (CPI). Further two are the CPI modifications: by CPI_m the variant, where the known values at boundary vertices are eliminated from the final matrix system (8), is denoted. The CPI_bv denotes the CPI variant, where the results of system (8) are at the positions related to the boundary vertices overwritten by the known values.

Figure 3 shows the velocity magnitude after all 20 interpolation runs from the original to the target mesh and back. It is evident that the Lagrange projection performs badly and it is too diffusive. The results of the interpolations CPI_m, CPI_bv (not shown) and CPI are very similar each to the other as well to the original data. The behaviour of interpolations along two lines are shown in Figure 4. In the case of the top domain boundary only the CPI results do not correspond to the exact ones because other CPI variants benefit from the additional information at the boundary. The CPI behaviour along the middle line is the same as the CPI_bv and the CPI_m is even slightly closer to the exact values than CPI, the LAG results are the worst.

From the quantitative point of view the L_2 error of the Lagrange projection is higher by 38%, while both the CPI modifications outperforms the original by 13% (CPI_bv) and by 11% (CPI_m), respectively, see Table 1. The L_∞ error is for the considered interpolation methods similar. Nevertheless the disadvantage of the CPI_bv method is the violation of the conservation of the kinetic energy. This happens due to the modification of the CPI solution¹ at the positions related to the boundary vertices contrary to the CPI_m variant, where the matrix system is modified rather than the individual values of interpolation result. Consequently the CPI_m provides a very precise kinetic energy conservation. Thus better choice appears to be the CPI_m than the CPI_bv, the CPI interpolation performs also reasonably well.

method	$\max \mathbf{F} $	E_{kin}	L_2 error	L_∞ error
exact	$1.650 \cdot 10^{-2}$	$6.657 \cdot 10^{-5}$	0	0
Lagrange int.	$1.650 \cdot 10^{-2}$	$5.306 \cdot 10^{-5}$	$2.343 \cdot 10^{-6}$	$3.489 \cdot 10^{-3}$
CPI	$1.848 \cdot 10^{-2}$	$6.657 \cdot 10^{-5}$	$1.697 \cdot 10^{-6}$	$3.499 \cdot 10^{-3}$
CPI_bv	$1.650 \cdot 10^{-2}$	$6.592 \cdot 10^{-5}$	$1.436 \cdot 10^{-6}$	$3.211 \cdot 10^{-3}$
CPI_m	$1.650 \cdot 10^{-2}$	$6.657 \cdot 10^{-5}$	$1.520 \cdot 10^{-6}$	$3.221 \cdot 10^{-3}$

Table 1: Comparison of interpolation results of the first test.

¹The CPI interpolation preserves kinetic energy.

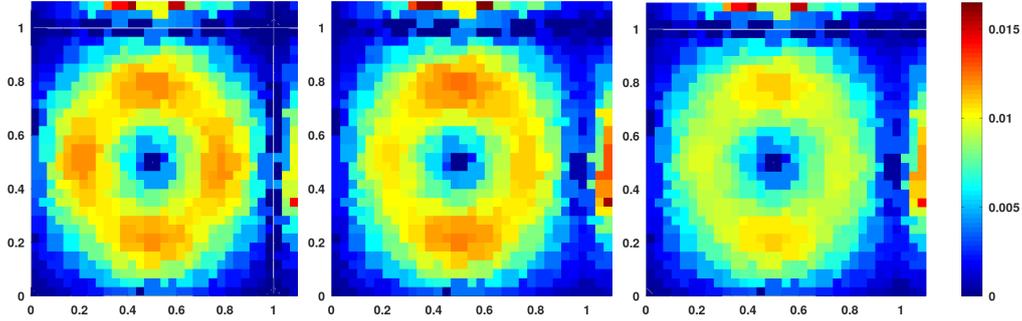


Figure 3: Magnitude of the interpolated vector field on the structured FE mesh after 20 runs. The exact values are shown on the left, the result of the interpolation with restrictions in the middle and the Lagrange interpolation on the right.

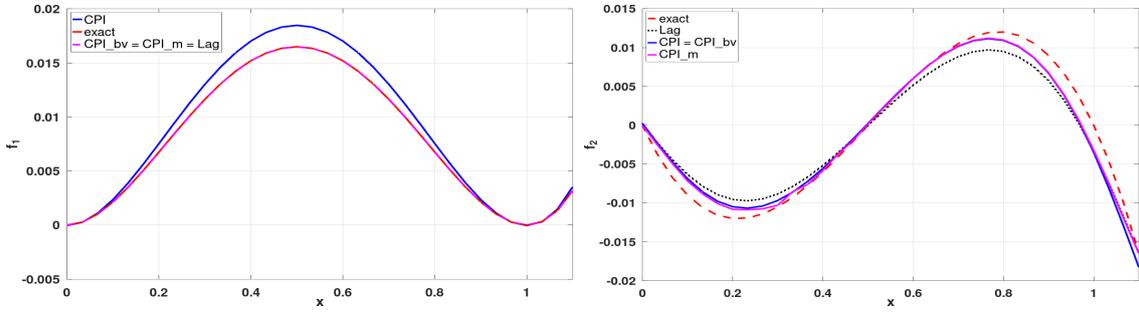


Figure 4: **Left:** Comparison of the interpolation of the first component of the velocity along the top boundary given by $y = 1.1, x \in \langle 0, 1.1 \rangle$. **Right:** Comparison of the second component of the velocity along line given by $y = 0.5, x \in \langle 0, 1.1 \rangle$.

4.2. Second interpolation test – question of interpolation domain

In the second test the interpolation results for different choices of interpolation domain are compared using an additional assumption of the following correspondence between the donor and the target mesh: The difference of the new target against the original mesh is the coarsened middle part around a channel constriction, see Figure 5, where the remaining parts of the target mesh are identical with the original one. Such mesh coarsening is motivated by the usage of our in-house solver FSIfem based on the ALE method (see [9]) in order to avoid deterioration of fluid mesh quality during simulations involving (almost complete) channel closing. Since the target domain of the ALE mapping can be chosen in the FSIfem solver we compare CPI interpolation on the following choices of the interpolation subdomains of the computational domain, see Fig. 5, with the aim to decrease interpolation error:

1. only the middle part of the constriction (CPI_sel)
2. the whole domain, but with the coincident mesh vertices outside of the middle part (CPI_coi)
3. the whole domain (here slightly shifted mesh vertices are used) (CPI.all).

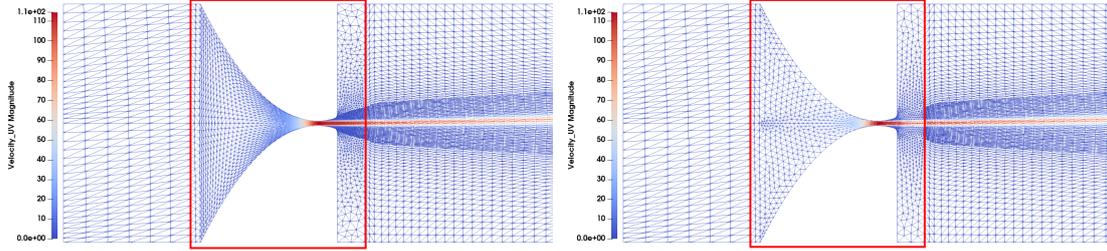


Figure 5: Meshes used in the second test together with initial and interpolated velocity distributions. The donor (original) mesh is shown on left and the target mesh on the right, only the middle part of both meshes differs and it is highlighted by the red square.

Here, the considered velocity field, which is obtained by FSIfem as part of the FSI solution, is once interpolated from the donor to the target mesh and vice versa.

Figure 6 illustrates the distribution of error after one pair of the interpolation runs. The results of the interpolations CPI_sel and CPI_coi are very close, while the error of CPI_all is a little higher. Moreover the error of CPI_all is distributed also significantly in the area right from the channel constriction contrary to the CPI_sel and CPI_coi results. The relative high interpolation error in the boundary layer is caused by the coarse target mesh at the region. In the case with a similarly dense target mesh the interpolation error can be expected to be significantly lower.

Interpolations obtained by CPI_sel and CPI_coi have similar L_2 and L_∞ errors, see Table 2, however slightly smaller L_2 error of CPI_sel is redeemed by the inconsistency in the maximal value and in the total kinetic energy. The CPI_all presents the largest L_2 and L_∞ errors.

method	$\max \mathbf{F} $	E_{kin}	L_2 error	L_∞ error
exact	$1.142 \cdot 10^2$	$9.644 \cdot 10^{-2}$	0	0
Lagrange int.	$1.141 \cdot 10^2$	$9.580 \cdot 10^{-2}$	$9.914 \cdot 10^{-5}$	$3.512 \cdot 10^1$
CPI_sel	$1.184 \cdot 10^2$	$9.646 \cdot 10^{-2}$	$9.086 \cdot 10^{-5}$	$3.369 \cdot 10^1$
CPI_coi	$1.145 \cdot 10^2$	$9.644 \cdot 10^{-2}$	$9.838 \cdot 10^{-5}$	$3.498 \cdot 10^1$
CPI_all	$1.153 \cdot 10^2$	$9.644 \cdot 10^{-2}$	$1.323 \cdot 10^{-4}$	$3.472 \cdot 10^1$

Table 2: Comparison of interpolation results of the second test.

5. Conclusion

The article presents the general concept of the interpolation between FEM meshes based on paper [5]. The idea of the interpolation with restrictions is to improve the commonly available interpolation procedure by a restriction of the conservation of additional physical quantities. Such approach has the advantage of the relatively

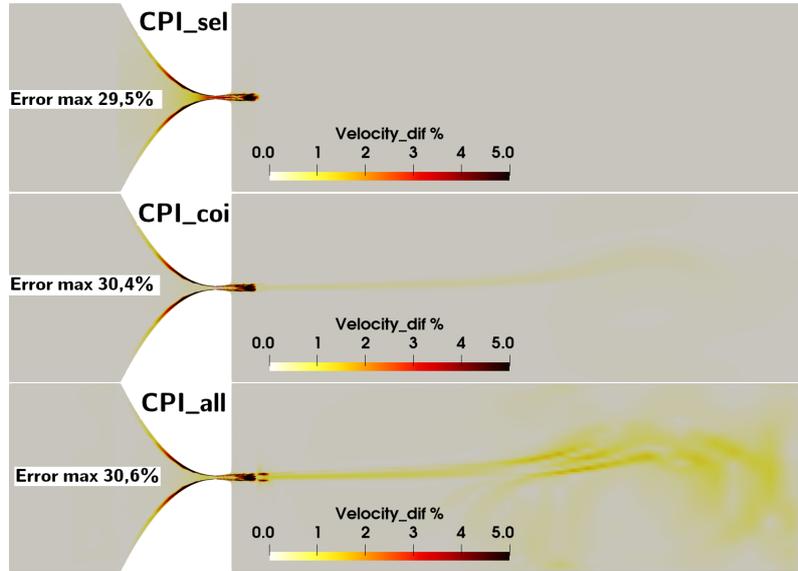


Figure 6: Distributions of relative error magnitudes after one pair of interpolations. Results for CPI_sel on the top, for CPI_coi in the middle and error of the CPI_all interpolation on the bottom. The maximal error is located for all methods similarly in a few elements inside boundary layer (out of color scale).

computationally cheap method which moreover respects physical nature of the problem. Here for the case of the fluid flow the conservation of the linear momenta, the divergence and the kinetic energy is considered. Our implementation based on the barycentric coordinates is described.

Two interpolation tests are performed in order to compare different modifications of this method motivated by the different settings of the FSI solver based on the ALE method. First, the different treatments of the nonzero boundary values are studied. The interpolation results can be slightly improved by inclusion of the information from the boundary into the resulting matrix system (variant CPI_m). Second, the interpolation error is calculated for different choices of the interpolation domain. The best results are obtained for the interpolation in the whole domain where the mesh distortion is highly localized around the channel constriction (variant CPI_coi). The interpolation of only the distorted part of the domain violates the conservation of the total kinetic energy in the whole domain.

Acknowledgements

The work was supported from European Regional Development Fund – Project “Center for Advanced Applied Science” (No.CZ.02.1.01/0.0/0.0/16-019/0000778) and from Premium Academiae of Prof. Nečasová.

References

- [1] Bussetta, P., Boman, R., and Ponthot, J.P.: Efficient 3D data transfer operators based on numerical integration. *Internat. J. Numer. Methods Engrg.* **102** (2015), 892–929.
- [2] Farrell, P.E. and Maddison, J.R.: Conservative interpolation between volume meshes by local galerkin projection. *Comput. Methods Appl. Mech. Engrg.* **200** (2011), 89–100.
- [3] Gander, M.J. and Japhet, C.: An algorithm for non-matching grid projections with linear complexity. In: *Proc. of 18th Intern. Conf. on Domain Decomposition Methods*. Springer, 2009.
- [4] Klíma, M., Kuchařík, M., and Shashkov, M.: Combined swept region and intersection-based single-material remapping method. *Internat. J. Numer. Methods Fluids* **85** (2017), 363–382.
- [5] Pont, A., Codina, R., and Baiges, J.: Interpolation with restrictions between finite element meshes for flow problems in an ALE setting. *Internat. J. Numer. Methods Engrg.* **110** (2017), 1203–1226.
- [6] Schoder, S., Roppert, K., Weitz, M., Junger, C., and Kaltenbacher, M.: Aeroacoustic source term computation based on radial basis functions. *Internat. J. Numer. Methods Engrg.* (2019).
- [7] Schoder, S. et al.: Application limits of conservative source interpolation methods using a low Mach number hybrid aeroacoustic workflow. *J. Theor. Comput. Acoust.* **29** (2021), 2050 032.
- [8] Sváček, P. and Horáček, J.: FE numerical simulation of incompressible airflow in the glottal channel periodically closed by self-sustained vocal folds vibration. *J. Comput. Appl. Math.* **393** (2021), 113 529.
- [9] Valášek, J., Sváček, P., and Horáček, J.: On suitable inlet boundary conditions for fluid-structure interaction problems in a channel. *Appl. Math.* **64** (2019), 225–251.

