

Tree-based adaptive finite element methods for deformable image registration

Nicolás A. Barnafi^{a,b}, Alberto F. Martín^c, and Ricardo Ruiz-Baier^{d,e,*}

^aFacultad de Ciencias Biológicas, Instituto de Ingeniería Matemática y Computacional, Pontificia Universidad Católica de Chile, Santiago, Chile

^bCenter for Mathematical Modeling, Santiago, Chile

^cSchool of Computing, Australian National University, Acton, ACT, Australia

^dSchool of Mathematics, Monash University, Melbourne, VIC, Australia

^eUniversidad Adventista de Chile, Chillán, Chile

*Corresponding author. e-mail address: ricardo.ruizbaier@monash.edu

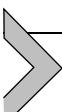
Contents

1. Introduction	2
2. Problem setting	4
3. Adaptive discretisation scheme	7
4. Solution strategy	12
4.1 Time acceleration	14
4.2 The adaptive solver	14
5. Numerical tests	16
5.1 Convergence verification against manufactured solutions	18
5.2 Anderson acceleration for DIR	20
5.3 Adaptivity performance on brain images	23
5.4 Adaptivity performance on OC images	24
5.5 Quadrature sensitivity	26
6. Conclusions and future perspectives	28
Funding	30
References	31

Abstract

In this work we propose an adaptive Finite Element Method (FEM) formulation for the Deformable Image Registration problem (DIR) together with a residual-based *a posteriori* error estimator, whose efficiency and reliability are theoretically established. This estimator is used to guide Adaptive Mesh Refinement and coarsening (AMR). The nonlinear Euler–Lagrange equations associated with the minimisation of the relevant functional are solved with a pseudo time-stepping fixed-point scheme which is further accelerated using Anderson Acceleration (AA). The efficient implementation of these solvers relies on an efficient adaptive mesh data structure

based on forests-of-octrees endowed with space-filling-curves. Several numerical results illustrate the performance of the proposed methods applied to adaptive DIR in application-oriented problems.



1. Introduction

Image registration problems consist in finding an appropriate mapping of a given template image so that it resembles another reference image. These images can originate from several different applications, and may represent physical parameters, locations, or even time. See ([Sotiras et al., 2013](#)) for a review in medical imaging. There are many ways of defining such a problem, including variational formulations ([Christensen et al., 1996](#); [Henn & Witsch, 2004](#)), level-set methods, grid deformation methods ([Lee & Gunzburger, 2010](#)), learning processes ([De Vos et al., 2019](#)) and Bayesian techniques ([Deshpande & Bhatt, 2019](#)), among others. All these approaches differ in how they measure image alignment, with such measure being referred to as *similarity*. Image registration is named *deformable* whenever the mapping between images is allowed to vary arbitrarily between pixels, which results in a highly nonlinear problem.

Image registration remains a challenging problem in computational imaging due to three inherent difficulties: its ill-posed nature, the selection of appropriate similarity metrics, and the high computational demands of numerical optimisation. First, the problem's ill-posedness necessitates regularisation strategies, such as elastic potential energy constraints ([Sotiras et al., 2013](#)), to ensure physically plausible solutions. Second, defining effective similarity measures—whether through energy-based criteria or intensity difference metrics like the L^2 norm—requires careful consideration of image modality and noise characteristics. Third, solving the Euler–Lagrange equations derived from the energy minimisation framework imposes significant computational costs ([Haber et al., 2007](#)), particularly when capturing localised deformations. These deformations often exhibit high spatial gradients to resolve fine anatomical or structural details, further amplifying discretisation challenges.

Adaptive numerical methods naturally present a promising approach to address these issues. By dynamically refining computational grids or basis functions in regions of sharp deformation, adaptive techniques hold the promise to achieve a remarkable trade-off among accuracy and computational efficiency. Recent advances in AMR demonstrate the potential of such strategies to overcome traditional limitations in registration tasks, for which we

highlight (Barnafi et al., 2021; Haber et al., 2007, 2008; Pawar et al., 2016; Zhang et al., 2013). With this in mind, in this work we leverage hierarchically-adapted non-conforming forest-of-octrees meshes endowed with Morton (a.k.a., Z-shaped) space-filling-curves for storage and data partitioning; see, e.g., (Badia et al., 2020; Burstedde et al., 2011). These n -cube meshes (made of quadrilaterals or cubes in 2D and 3D, resp.) can be very efficiently handled (i.e. refined, coarsened, re-partitioned, etc.) using high-performance and low-memory footprint algorithms (Burstedde et al., 2011). However, they are non-conforming at the interfaces of cells with different refinement levels as they yield hanging vertices, edges and faces. In order to enforce the conformity (continuity) of Finite Element (FE) spaces built out of this kind of meshes, it is standard practice to equip the space with additional linear multi-point constraints. The structure and set up of such constraints is well-established knowledge, and thus not covered here; see (Badia et al., 2020) and references therein for further details.

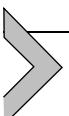
In this work, we consider the DIR with an L^2 similarity measure and a regularisation given by the (linear) elastic energy of the deformation with Robin and pure Neumann boundary conditions. (We note that nonlinear elasticity has also been considered as an alternative regularization term in other works, such as, e.g. (Genet et al., 2018).) This formulation was analysed in (Barnafi et al., 2018), and an efficient and reliable *a-posterior* error estimator was developed for it in (Barnafi et al., 2021) for the pure-Neumann case, where uniqueness was obtained by imposing orthogonality against the kernel of the regularisation operator. We extend that analysis to the case of Robin boundary conditions. This efficient and reliable *a-posterior* error estimator is used at a given solution of the DIR on a given mesh in order to automatically adapt it, and the process is repeated across several AMR iterations to successively improve the accuracy of the DIR solution.

As mentioned above, solving DIR is a challenging task. It is essentially a nonlinear inverse problem. A well-established strategy to solve it is by means of a pseudo-time formulation (Modersitzki, 2003), which is an IMEX approximation of a proximal point algorithm (Kaplan & Tichatschke, 1998). In this work, we leverage AA (Walker & Ni, 2011) to improve its robustness and efficiency in terms of the iteration count. Anderson acceleration is a method that considers a fixed-point iteration and yields another one with improved convergence, both in terms of the radius of convergence and the convergence rate (Toth & Kelley, 2015). This technique has already been successfully validated for proximal-point

algorithms in the context of inverse problems (Mai & Johansson, 2020), which is the method that most resembles the scheme we use.

The main contributions of this work are: (i) the extension of the efficiency and reliability proof of the existing *a posteriori* error indicator for DIR to Robin boundary conditions; (ii) the use of AA to accelerate the pseudo-time solution strategy commonly used in DIR; (iii) the use of the computed *a posteriori* error estimator to guide efficient AMR using octree-based meshes; and (iv) the combination of AA and AMR into a novel AA-AMR algorithm for solving DIR. These methodologies result in significant computational savings and increased accuracy, as shown in our numerical tests. Our software is implemented in the Julia programming language (Bezanson et al., 2017) using the FE tools provided by the Gridap ecosystem of Julia packages (Verdugo & Badia, 2022; Badia et al., 2022; Martín, 2025).

Structure. The remainder of the work has been organised as follows. In Section 2, we formulate the DIR problem. In Section 3, we propose FEM discretisation for DIR and its related robust (in the sense of reliability and efficiency) *a-posteriori* error estimator driving the AMR process. In Section 4, we show how we orchestrate the solution of the nonlinear problem with both AA and AMR. In Section 5 we provide several numerical tests to validate our approach. We conclude our work with some comments and an outlook for future work in Section 6.



2. Problem setting

Consider a domain $\Omega \subset \mathbb{R}^{d=2,3}$, and two fields $R: \Omega \rightarrow \mathbb{R}$ and $T: \Omega \rightarrow \mathbb{R}$ referred to as *reference* and *target* (or template) images, where $R(\mathbf{x})$ and $T(\mathbf{x})$ denote the *image intensity* at point \mathbf{x} . DIR consists in finding a transformation under which the template image resembles the reference (Modersitzki, 2003), that is to find a mapping of T onto R by means of a warping \mathbf{u} such that

$$T(\mathbf{x} + \mathbf{u}(\mathbf{x})) \approx R(\mathbf{x}) \quad \forall \mathbf{x} \in \Omega. \quad (1)$$

This can be reformulated as a variational problem

$$\inf_{\mathbf{u} \in \mathcal{V}} \alpha \mathcal{D}[\mathbf{u}; R, T] + \mathcal{S}[\mathbf{u}], \quad (2)$$

where \mathcal{V} models the admissible deformations space, $\mathcal{D}: \mathcal{V} \rightarrow \mathbb{R}$ is the aforementioned *similarity measure* which attains its minimum when (1) holds, \mathcal{S} is a regulariser, and α is a regularisation parameter that balances the effects of \mathcal{D} and \mathcal{S} . In this work we consider $\mathcal{V} = \mathbf{H}^1(\Omega)$, the L^2 -norm of the error for the similarity, given by

$$\mathcal{D}[\mathbf{u}; R, T] = \int_{\Omega} (T(\mathbf{x} + \mathbf{u}(\mathbf{x})) - R(\mathbf{x}))^2 d\mathbf{x},$$

and as a regulariser we use the elastic deformation energy, defined in the following manner

$$\mathcal{S}[\mathbf{u}] := \frac{1}{2} \int_{\Omega} C \mathbf{e}(\mathbf{u}) : \mathbf{e}(\mathbf{u}) d\mathbf{x}.$$

Here $\mathbf{e}(\mathbf{u}) = \frac{1}{2}\{\nabla \mathbf{u} + (\nabla \mathbf{u})^t\}$ is the infinitesimal strain tensor, i.e., the symmetric component of the displacement field gradient, and C is the elasticity tensor for isotropic solids:

$$C\boldsymbol{\tau} = \lambda \text{tr}(\boldsymbol{\tau}) \mathbb{I} + 2\mu \boldsymbol{\tau} \quad \forall \boldsymbol{\tau} \in \mathbb{R}^{d \times d}.$$

Naturally, one may consider many other types of regularisations. See (Modersitzki, 2003) for a review. Assuming that (2) has at least one solution with sufficient regularity, the associated Euler–Lagrange equations deliver the following strong problem with Robin or Neumann boundary conditions (representing springs of stiffness $\kappa \geq 0$): Find \mathbf{u} in $\mathbf{H}^1(\Omega)$ such that

$$\begin{aligned} -\mathbf{div}(C \mathbf{e}(\mathbf{u})) &= \alpha f_u && \text{in } \Omega, \\ C \mathbf{e}(\mathbf{u}) \mathbf{n} + \kappa \mathbf{u} &= \mathbf{0} && \text{on } \partial\Omega, \end{aligned} \tag{3}$$

where

$$f_u(\mathbf{x}) = \{T(\mathbf{x} + \mathbf{u}(\mathbf{x})) - R(\mathbf{x})\} \nabla T(\mathbf{x} + \mathbf{u}(\mathbf{x})) \quad \forall \mathbf{x} \in \Omega. \tag{4}$$

For the analysis we assume that there are positive constants L_f and M_f such that the nonlinear load term f_u is Lipschitz continuous and uniformly bounded:

$$\begin{aligned} |f_u(\mathbf{x}) - f_v(\mathbf{x})| &\leq L_f |\mathbf{u}(\mathbf{x}) - \mathbf{v}(\mathbf{x})|, \\ |f_u(\mathbf{x})| &\leq M_f \quad \forall \mathbf{x} \in \Omega \text{ a.e.} \end{aligned} \tag{5}$$

Under the previous considerations, (2) becomes

$$\min_{\boldsymbol{v} \in \mathbf{H}^1(\Omega)} \left\{ \alpha \int_{\Omega} |T(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) - R(\boldsymbol{x})|^2 d\boldsymbol{x} + \frac{1}{2} \int_{\Omega} C \mathbf{e}(\boldsymbol{u}) : \mathbf{e}(\boldsymbol{u}) d\boldsymbol{x} \right\} \\ + \kappa \int_{\partial\Omega} |\boldsymbol{u}|^2 ds,$$

with first order conditions given by the primal variational formulation for the registration problem: Find $\boldsymbol{u} \in \mathbf{H}^1(\Omega)$ such that

$$a(\boldsymbol{u}, \boldsymbol{v}) = \alpha F_{\boldsymbol{u}}(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in \mathbf{H}^1(\Omega), \quad (6)$$

where $a: \mathbf{H}^1(\Omega) \times \mathbf{H}^1(\Omega) \rightarrow \mathbb{R}$ is the bilinear form defined by

$$a(\boldsymbol{u}, \boldsymbol{v}) := \int_{\Omega} C \mathbf{e}(\boldsymbol{u}) : \mathbf{e}(\boldsymbol{v}) + \kappa \int_{\partial\Omega} \boldsymbol{u} \cdot \boldsymbol{v} ds \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \mathbf{H}^1(\Omega), \quad (7)$$

and for every $\boldsymbol{u} \in \mathbf{H}^1(\Omega)$, $F_{\boldsymbol{u}}: \mathbf{H}^1(\Omega) \rightarrow \mathbb{R}$ is the linear functional given by

$$F_{\boldsymbol{u}}(\boldsymbol{v}) := - \int_{\Omega} \boldsymbol{f}_{\boldsymbol{u}} \cdot \boldsymbol{v} \quad \forall \boldsymbol{v} \in \mathbf{H}^1(\Omega). \quad (8)$$

The conditions (5) imply the Lipschitz continuity and uniform boundedness of $F_{\boldsymbol{u}}$, that is

$$\|F_{\boldsymbol{u}} - F_{\boldsymbol{v}}\|_{\mathbf{H}^1(\Omega)'} \leq L_F \|\boldsymbol{u} - \boldsymbol{v}\|_{0,\Omega}, \quad (9)$$

$$\|F_{\boldsymbol{u}}\|_{\mathbf{H}^1(\Omega)'} \leq M_F \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \mathbf{H}^1(\Omega),$$

respectively. We recall the results concerning the solvability of (6), as developed in (Barnafi et al., 2018, Section 3). For it, whenever $\kappa = 0$ we need to modify the solution space in order to guarantee uniqueness. We will denote with $\mathbb{RM} = \ker(\mathcal{S})$ as the space of rigid body modes, and thus consider the solution space given by

$$\mathbf{V} = \begin{cases} \mathbf{H}^1(\Omega) & \kappa > 0 \\ \mathbf{H}^1(\Omega) \cap [\mathbb{RM}]^\perp & \kappa = 0 \end{cases}$$

With it, we define the following linear auxiliary problem: Given $\boldsymbol{\xi} \in \mathbf{H}^1(\Omega)$, find $\boldsymbol{u} \in \mathbf{V}$ such that

$$a(\boldsymbol{u}, \boldsymbol{v}) = \alpha F_{\boldsymbol{\xi}}(\boldsymbol{v}), \quad \boldsymbol{v} \in \mathbf{V}. \quad (10)$$

Theorem 1. Given $\boldsymbol{\xi} \in \mathbf{H}^1(\Omega)$, problem (10) has a unique solution $\boldsymbol{u} \in \mathbf{V}$, and there exists $C_p > 0$ such that

$$\|\boldsymbol{u}\|_{1,\Omega} \leq \alpha C_p \|F_{\boldsymbol{\xi}}\|_{\mathbf{H}^1(\Omega)'}. \quad (11)$$

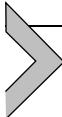
We now define the operator $\widehat{\mathbf{T}}: \mathbf{V} \rightarrow \mathbf{V}$ given by $\widehat{\mathbf{T}}(\boldsymbol{\zeta}) = \mathbf{u}$, where \mathbf{u} is the unique solution to problem (10) and thus rewrite (6) as the fixed-point equation: Find \mathbf{u} in \mathbf{V} such that

$$\widehat{\mathbf{T}}(\mathbf{u}) = \mathbf{u}. \quad (11)$$

The following result, also proven in Barnafi et al. (2018, Theorem 3), establishes the existence and uniqueness of solution to the fixed-point equation (11) whenever $\kappa = 0$.

Theorem 2. Under data assumptions (5), the operator $\widehat{\mathbf{T}}$ has at least one fixed point. Moreover, if $\alpha C_p L_F < 1$, the fixed point is unique.

Throughout the manuscript, we will use \mathbf{V} to write our problem in order to avoid technicalities regarding the implementation of rigid body modes \mathbb{RM} whenever $\kappa = 0$. We enforce it in practice by means of an adequate Lagrange multiplier as in (Barnafi et al., 2018).



3. Adaptive discretisation scheme

Let \mathbf{V}_h be a finite dimensional subspace of \mathbf{V} built out of a suitable mesh partition \mathcal{T}_h of Ω into quadrilateral or hexahedral elements, where h denotes the mesh size. The primal nonlinear discrete problem consists in finding $\mathbf{u}_h \in \mathbf{V}_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}_h) = \alpha F_{\mathbf{u}_h}(\mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h. \quad (12)$$

Analogously to the continuous case, we consider the auxiliary linear problem: Given $\boldsymbol{\zeta}_h \in \mathbf{V}_h$, find $\mathbf{u}_h \in \mathbf{V}_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}_h) = \alpha F_{\boldsymbol{\zeta}_h}(\mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h, \quad (13)$$

and also let $T_h: \mathbf{V}_h \rightarrow \mathbf{V}_h$ be the discrete operator given by $T_h(\boldsymbol{\zeta}_h) = \mathbf{u}_h$, where \mathbf{u}_h is the solution to problem (13). Considering the same data assumptions as in the continuous case, as well as the continuity and bound obtained before, we arrive at the following result proven in Barnafi et al. (2018, Theorem 5), adapted to include the terms arising from $\kappa \neq 0$.

Theorem 3. Assume that data assumptions (5) hold. Then, the operator T_h has at least one fixed point. Moreover, if $\alpha C_p L_F < 1$, then such fixed point is unique.

Next, we define for each $K \in \mathcal{T}_h$ the *a-posteriori* error indicator

$$\begin{aligned}\Theta_K^2 &:= h_K^2 \|\alpha f_{\mathbf{u}_h} - \mathbf{div}(C\mathbf{e}(\mathbf{u}_h))\|_{0,K}^2 \\ &\quad + \sum_{e \in \mathcal{E}(K) \cap \mathcal{E}_h(\Omega)} h_e \| [C\mathbf{e}(\mathbf{u}_h) \mathbf{n}_e] \|_{0,e}^2 \\ &\quad + \sum_{e \in \mathcal{E}(K) \cap \mathcal{E}_h(\Gamma)} h_e \| C\mathbf{e}(\mathbf{u}_h) \mathbf{n}_e + \kappa \mathbf{u}_h \|_{0,e}^2,\end{aligned}\tag{14}$$

where, according to (4),

$$f_{\mathbf{u}_h}|_K(\mathbf{x}) := \{T(\mathbf{x} + \mathbf{u}_h(\mathbf{x})) - R(\mathbf{x})\} \nabla T(\mathbf{x} + \mathbf{u}_h(\mathbf{x})) \quad \forall \mathbf{x} \in K,$$

and introduce the global *a-posteriori* error estimator

$$\Theta := \left\{ \sum_{K \in \mathcal{T}_h} \Theta_K^2 \right\}^{1/2}.$$

The following theorem constitutes the main result of this section.

Theorem 4. Let $\mathbf{u} \in \mathbf{V}$ and $\mathbf{u}_h \in \mathbf{V}_h$ be the solutions of (6) and (12), respectively, and assume that $\alpha C_p L_F < 1/2$. Then, there exist positive constants h_0 , C_{rel} , C_{eff} independent of h such that for $h \leq h_0$ there holds

$$C_{\text{eff}} \Theta \leq \|\mathbf{u} - \mathbf{u}_h\|_{1,\Omega} \leq C_{\text{rel}} \Theta. \tag{15}$$

The reliability (upper bound in (15)) and the efficiency (lower bound in (15)) of Θ are established separately in the following two lemmas.

Lemma 1. Assume that $\alpha C_p L_F < 1/2$. Then, there exist positive constants h_0 , C_{rel} independent of h , such that for $h \leq h_0$ there holds

$$\|\mathbf{u} - \mathbf{u}_h\|_{\mathbf{V}} \leq C_{\text{rel}} \Theta.$$

Proof. Let us first define

$$\mathcal{R}_h(\mathbf{w} - \mathbf{w}_h) := \alpha F_{\mathbf{u}}(\mathbf{w} - \mathbf{w}_h) - a(\mathbf{u}_h, \mathbf{w} - \mathbf{w}_h) \quad \forall \mathbf{w}_h \in \mathbf{V}_h.$$

As a consequence of the ellipticity of a (c.f. (7)) with ellipticity constant $\bar{\alpha}$ (c.f. Brenner, 2008, Corollary 11.2.22), we obtain the following condition

$$\bar{\alpha} \|\mathbf{v}\|_{1,\Omega} \leq \sup_{\substack{\mathbf{w} \in \mathbf{H}^1(\Omega) \\ \mathbf{w} \neq 0}} \frac{a(\mathbf{v}, \mathbf{w})}{\|\mathbf{w}\|_{\mathbf{V}}} \quad \forall \mathbf{v} \in \mathbf{H}^1(\Omega).$$

In particular, for $\mathbf{v} := \mathbf{u} - \mathbf{u}_h \in \mathbf{H}^1(\Omega)$, we notice from (6) and (12) that

$a(\mathbf{u} - \mathbf{u}_h, \mathbf{w}_h) = 0 \quad \forall \mathbf{w}_h \in \mathbf{V}_h$, and hence we obtain $a(\mathbf{u} - \mathbf{u}_h, \mathbf{w}) = a(\mathbf{u} - \mathbf{u}_h, \mathbf{w} - \mathbf{w}_h) = \mathcal{R}_h(\mathbf{w} - \mathbf{w}_h)$, which yields

$$\bar{\alpha} \|\mathbf{u} - \mathbf{u}_h\|_{1,\Omega} \leq \sup_{\substack{\mathbf{w} \in \mathbf{H}^1(\Omega) \\ \mathbf{w} \neq 0}} \frac{\mathcal{R}_h(\mathbf{w} - \mathbf{w}_h)}{\|\mathbf{w}\|_{1,\Omega}} \quad \forall \mathbf{w}_h \in \mathbf{V}_h. \quad (16)$$

From the definition of $\mathcal{R}_h(\mathbf{w} - \mathbf{w}_h)$, integrating by parts on each $K \in \mathcal{T}_h$, and adding and subtracting a suitable term, we can write

$$\begin{aligned} \mathcal{R}_h(\mathbf{w} - \mathbf{w}_h) &= \alpha F_{\mathbf{u}_h}(\mathbf{w} - \mathbf{w}_h) + \alpha F_{\mathbf{u}}(\mathbf{w} - \mathbf{w}_h) - a(\mathbf{u}_h, \mathbf{w} - \mathbf{w}_h) \\ &\quad - \alpha F_{\mathbf{u}_h}(\mathbf{w} - \mathbf{w}_h) \\ &= \alpha \{ F_{\mathbf{u}}(\mathbf{w} - \mathbf{w}_h) - F_{\mathbf{u}_h}(\mathbf{w} - \mathbf{w}_h) \} - \alpha \int_{\Omega} f_{\mathbf{u}_h} \cdot (\mathbf{w} - \mathbf{w}_h) \\ &\quad - \sum_{K \in \mathcal{T}_h} \int_K C\mathbf{e}(\mathbf{u}_h) : \mathbf{e}(\mathbf{w} - \mathbf{w}_h) \\ &= \alpha \{ (F_{\mathbf{u}} - F_{\mathbf{u}_h})(\mathbf{w} - \mathbf{w}_h) \} - \alpha \int_{\Omega} f_{\mathbf{u}_h} \cdot (\mathbf{w} - \mathbf{w}_h) \\ &\quad - \sum_{K \in \mathcal{T}_h} \left\{ - \int_K \mathbf{div}(C\mathbf{e}(\mathbf{u}_h)) \cdot (\mathbf{w} - \mathbf{w}_h) \right. \\ &\quad \left. + \int_{\partial K} (C\mathbf{e}(\mathbf{u}_h) \mathbf{n}_e) \cdot (\mathbf{w} - \mathbf{w}_h) \right\}, \quad (17) \\ &= \alpha \{ (F_{\mathbf{u}} - F_{\mathbf{u}_h})(\mathbf{w} - \mathbf{w}_h) \} + \sum_{K \in \mathcal{T}_h} \int_K (\mathbf{div}(C\mathbf{e}(\mathbf{u}_h))) \\ &\quad - \alpha f_{\mathbf{u}_h} \cdot (\mathbf{w} - \mathbf{w}_h) \\ &\quad - \sum_{e \in \mathcal{E}_h(\Omega)} \int_e [(C\mathbf{e}(\mathbf{u}_h) \mathbf{n}_e)] \cdot (\mathbf{w} - \mathbf{w}_h) \\ &\quad - \sum_{e \in \mathcal{E}_h(\Gamma)} \int_e (C\mathbf{e}(\mathbf{u}_h) \mathbf{n}_e + \kappa \mathbf{u}_h) \cdot (\mathbf{w} - \mathbf{w}_h). \end{aligned}$$

Then, choosing \mathbf{w}_h as the Clément interpolant of \mathbf{w} , that is $\mathbf{w}_h := I_h(\mathbf{w})$, the approximation properties of I_h yield (Clément, 1975)

$$\|\mathbf{w} - \mathbf{w}_h\|_{0,K} \leq c_1 h_K \|\mathbf{w}\|_{1,\square(K)}, \quad \|\mathbf{w} - \mathbf{w}_h\|_{0,e} \leq c_2 h_e \|\mathbf{w}\|_{1,\square(e)}, \quad (18)$$

where $\square(K) := \cup\{K' \in \mathcal{T}_h: K' \cap K \neq \emptyset\}$ and $\square(e) := \cup\{K' \in \mathcal{T}_h: K' \cap e \neq \emptyset\}$. In this way, applying the Cauchy–Schwarz inequality to each term (17), and making use of (18) together with the Lipschitz continuity of $F_{\mathbf{u}}$ (cf. (9)), we obtain

$$\begin{aligned} \mathcal{R}_h(\mathbf{w} - \mathbf{w}_h) &\leq \alpha c_1 L_F h_K \| \mathbf{u} - \mathbf{u}_h \|_{1,\Omega} \| \mathbf{w} \|_{1,\square(K)} \\ &+ \widehat{C} \left\{ \sum_{K \in \mathcal{T}_h} \Theta_K^2 \right\}^{1/2} \left\{ \sum_{K \in \mathcal{T}_h} \| \mathbf{w} \|_{1,\square(K)}^2 + \sum_{e \in \mathcal{E}_h(\Omega)} \| \mathbf{w} \|_{1,\square(e)}^2 \right\}^{1/2}, \end{aligned}$$

where \widehat{C} is a constant depending on c_1 and c_2 and Θ_K^2 defined by (14). Additionally using the fact that the number of elements in $\square(K)$ and $\square(e)$ is bounded, we have

$$\sum_{K \in \mathcal{T}_h} \| \mathbf{w} \|_{1,\square(K)}^2 \leq C_1 \| \mathbf{w} \|_{1,\Omega}^2 \quad \text{and} \quad \sum_{e \in \mathcal{E}_h(\Omega)} \| \mathbf{w} \|_{1,\square(e)}^2 \leq C_2 \| \mathbf{w} \|_{1,\Omega}^2,$$

where C_1, C_2 are positive constants, and using that $\alpha C_p L_F \leq 1/2$, it follows that $h_0 := 1/(2c_1 \alpha L_F)$. Finally, substituting in (16) we conclude that

$$\| \mathbf{u} - \mathbf{u}_h \|_{1,\Omega} \leq C_{\text{rel}} \Theta,$$

where C_{rel} is independent of h . \square

The efficiency bound requires using a localisation technique based on element-bubble and edge-bubble functions. Given $K \in \mathcal{T}_h$ and $e \in \mathcal{E}(K)$, we define ψ_K and ψ_e the typical element- and edge-bubble functions [Verfürth, 1999, eqs. (1.5)-(1.6)], which satisfy:

- (i) $\psi_K \in P_3(K)$, $\psi_K = 0$ on ∂K , $\text{supp}(\psi_K) \subseteq K$, and $0 \leq \psi_K \leq 1$ in K ,
- (ii) $\psi_e \in P_2(K)$, $\psi_e = 0$ on ∂K , $\text{supp}(\psi_e) \subseteq \omega_e$, and $0 \leq \psi_e \leq 1$ in ω_e ,

where $\omega_e := \cup\{K' \in \mathcal{T}_h : e \in \mathcal{E}(K')\}$. Additional properties of ψ_K and ψ_e are collected in the following lemma (c.f. Verfürth (1994, Lemma 1.3), Verfürth (1996, Section 3.4) or Verfürth (1999, Section 4)).

Lemma 2. Given $k \in \mathbb{N} \cup \{0\}$, there exist positive constants $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ and γ_5 , depending only on k and the shape regularity of the triangulations, such that for each $K \in \mathcal{T}_h$ and $e \in \mathcal{E}(K)$, there hold

$$\begin{aligned} \gamma_1 \| q \|_{0,K}^2 &\leq \| \psi_K^{1/2} q \|_{0,K}^2 \quad \forall q \in P_k(K), \\ \| \psi_K q \|_{1,K} &\leq \gamma_2 h_K^{-1} \| q \|_{0,K} \quad \forall q \in P_k(K), \\ \gamma_3 \| p \|_{0,e}^2 &\leq \| \psi_e^{1/2} p \|_{0,e}^2 \quad \forall p \in P_k(e), \\ \| \psi_e p \|_{1,\omega_e} &\leq \gamma_4 h_e^{-1/2} \| p \|_{0,e} \quad \forall p \in P_k(e), \\ \| \psi_e p \|_{0,\omega_e} &\leq \gamma_5 h_e^{1/2} \| p \|_{0,e} \quad \forall p \in P_k(e). \end{aligned} \tag{19}$$

The efficiency (lower bound in (15)) is established with the help of the following lemma whose proof is a slight modification of Verfürth (1999, Section 6).

Lemma 3. There exist constants $\eta_1, \eta_2, \eta_3 > 0$ and $C_{\text{eff}} > 0$, independent of h , but depending on γ_i for $i \in \{1, \dots, 5\}$ from Lemma 2, such that for each $K \in \mathcal{T}_h$ there holds

$$\begin{aligned} h_K \| \alpha f_{u_h} - \mathbf{div}(C\mathbf{e}(u_h)) \|_{0,K} &\leq \eta_1 \| u - u_h \|_{0,K}, \\ h_e^{1/2} \| [C\mathbf{e}(u_h) \cdot \mathbf{n}_e] \|_{0,e} &\leq \eta_2 \left\{ \| u - u_h \|_{0,\omega_e} + \sum_{K \in \omega_e} h_K \| u - u_h \|_{0,K} \right\}, \\ h_e^{1/2} \| C\mathbf{e}(u_h) \cdot \mathbf{n}_e \|_{0,e} &\leq \eta_3 \| u - u_h \|_{0,K}, \end{aligned}$$

where $\omega_e := \cup \{K' \in \mathcal{T}_h : e \in \mathcal{E}(K')\}$. Further, it holds that

$$C_{\text{eff}} \Theta \leq \| u - u_h \|_{1,\Omega}.$$

Proof. Using the properties of bubble functions, and letting $R_K(u_h) := \alpha f_{u_h} - \mathbf{div}(C\mathbf{e}(u_h))$ we have

$$\begin{aligned} \| R_K(u_h) \|_{0,K}^2 &\leq \gamma_1^{-1} \| \psi_K^{1/2} R_K(u_h) \|_{0,K}^2 \\ &= \frac{1}{\gamma_1} \int_K \alpha \psi_K R_K(u_h) \{ f_{u_h} - f_u \} \\ &\quad - \frac{1}{\gamma_1} \int_K \psi_K R_K(u_h) \{ \mathbf{div}(C\mathbf{e}(u_h)) - C\mathbf{e}(u) \}, \\ &= \frac{1}{\gamma_1} \int_K \alpha \psi_K R_K(u_h) \{ f_{u_h} - f_u \} + \frac{1}{\gamma_1} \int_K (C\mathbf{e}(u_h) \\ &\quad - C\mathbf{e}(u)) \cdot \nabla (\psi_K R_K(u_h)), \\ &\leq \frac{\alpha}{\gamma_1} \| R_K(u_h) \|_{0,K} \| f_{u_h} - f_u \|_{0,K} \\ &\quad + \frac{\gamma_2}{\gamma_1 h_K} \| C\mathbf{e}(u_h) - C\mathbf{e}(u) \|_{0,K} \| R_K(u_h) \|_{0,K}, \end{aligned}$$

where, for the last inequality we used the inverse inequality. Next, we have

$$h_K \| R_K(u_h) \|_{0,K} \leq \alpha h_K \gamma_1^{-1} \| f_{u_h} - f_u \|_{0,K} + \gamma_1^{-1} \gamma_2 \| C\mathbf{e}(u_h) - C\mathbf{e}(u) \|_{0,K},$$

now, using (5) and grouping terms, we conclude with $\eta_1 > 0$ independent of h , that

$$h_K \| \alpha f_{u_h} - \mathbf{div}(C\mathbf{e}(u_h)) \|_{0,K} \leq \eta_1 \| u - u_h \|_{0,K}.$$

We omit further details and repeated arguments used for the remaining inequalities. \square

4. Solution strategy

In this section we present how we propose to solve (3), which is mainly based on an accelerated proximal-point algorithm formulated in an infinite-dimensional setting. The *proximal point method* (Rockafellar, 1976) consists in adding a convex term to the original problem, so that given a solution \mathbf{u}^k , we can rewrite the minimisation problem as

$$\inf_{\mathbf{u} \in \mathcal{V}} \alpha \mathcal{D}[\mathbf{u}; R, T] + \mathcal{S}[\mathbf{u}] + \frac{1}{\Delta t} \|\mathbf{u} - \mathbf{u}^k\|_{\mathbf{V}}, \quad (20)$$

where the pseudo-timestep Δt is a regularisation parameter, and we denote the solution as \mathbf{u}^{k+1} . This scheme can be shown to converge under typically mild hypotheses (Kaplan & Tichatschke, 1998), and its related Euler–Lagrange equations are given by

$$\frac{1}{\Delta t} \mathcal{L}(\mathbf{u}^{k+1} - \mathbf{u}^k) - \operatorname{div}(C\epsilon(\mathbf{u}^{k+1})) = \alpha f_{\mathbf{u}^{k+1}}, \quad (21)$$

where \mathcal{L} is the operator induced by the inner product defining the norm in \mathbf{V} . This can possibly induce a modification in the boundary conditions that is proportional to the velocity term $\Delta t^{-1}(\mathbf{u}^{k+1} - \mathbf{u}^k)$. This problem is still highly nonlinear because of the nonlinear term $f_{\mathbf{u}^{k+1}}$, and so a common strategy is to treat it explicitly (Modersitzki, 2003), which yields the following semi-implicit (or IMEX) problem:

$$\frac{1}{\Delta t} \mathcal{L}\mathbf{u}^{k+1} - \operatorname{div}(C\epsilon(\mathbf{u}^{k+1})) = \frac{1}{\Delta t} \mathcal{L}\mathbf{u}^k + \alpha f_{\mathbf{u}^k}. \quad (22)$$

This formulation has been shown to be stable under the timestep condition $\Delta t \approx 1/\alpha$ (Barnafi et al., 2018). We now describe all the numerical choices that yield an efficient solution strategy of problem (3), based on the iterated solution of (22). The use of a semi-implicit formulation yields a simpler problem to solve at each instant, but in turn it implies a CFL condition relating the pseudo-timestep and the spatial discretisation in order to have stability. Such a stability condition has not been established rigorously for this type of problem to the best of the authors knowledge.

The proximal operator. We consider two options for the operator \mathcal{L} . On the one hand, an L^2 -norm in the regularised formulation (20), which results in $\mathcal{L} = I$ the identity operator. On the other hand, an H^1 -norm which results in $\mathcal{L} = -\Delta + I$, which gives rise to Sobolev

gradient stabilisation in the context of Levemberg–Marquardt methods ([Kazemi & Renka, 2012](#)). We then note that (22) written in terms of the increment $\delta\mathbf{u}^{k+1}$ results in

$$\left(\frac{1}{\Delta t} \mathcal{L} - \mathbf{div} C\epsilon \right) \delta\mathbf{u}^{k+1} = \mathbf{div} (C\epsilon(\mathbf{u}^k)) + \alpha f_{\mathbf{u}^k},$$

where the right-hand side $\mathbf{div} (C\epsilon(\mathbf{u}^k)) + \alpha f_{\mathbf{u}^k}$ is the residual of the original problem (3). This shows that this method can also be interpreted as a Levemberg–Marquardt type of algorithm ([Nocedal & Wright, 1999](#)). The stabilisation matrix in this case is the operator \mathcal{L} , and the gradient $\nabla_{\mathbf{u}} f_{\mathbf{u}}$ is neglected from the complete Jacobian, given by

$$\begin{aligned} \mathcal{J}(\mathbf{u}) = d^2 \mathcal{S} + \alpha (d^2 \mathcal{D}(\mathbf{u})) = \\ -\mathbf{div} C\epsilon + \alpha \underbrace{(\nabla T(\mathbf{l} + \mathbf{u}) \otimes \nabla T(\mathbf{l} + \mathbf{u}) + (T(\mathbf{l} + \mathbf{u}) - R) \mathbf{H}T(\mathbf{l} + \mathbf{u}))}_{\nabla_{\mathbf{u}} f_{\mathbf{u}}} \end{aligned}$$

where \mathbf{HT} stands for the Hessian of T and \mathbf{l} is the identity function.

Discrete spaces. The left-hand side operator in (22) consists in a linear elasticity operator plus an identity, so we can use \mathbf{H}^1 -conforming FEs. In particular, we consider a vector-valued Lagrangian FE space. For a quadrilateral ($d = 2$) or hexahedral ($d = 3$) discretization \mathcal{T}_h of Ω , this space is given by

$$Q_h^k = \{\mathbf{v} \in C(\Omega, \mathbb{R}^d) : \mathbf{v}|_K \in Q_k^d(K) \quad \forall K \in \mathcal{T}_h\},$$

where $Q_k^d = Q_k \otimes \dots \otimes Q_k$, and Q_k is the space of d -variate tensor-product polynomials of partial degree at most k with respect to each variable, for each of the d components of the vector-valued field.

Solvers in use. To solve the linear system arising from (22), we use an efficient sparse LU factorisation, which we compute once and then reuse it throughout all time iterations.

4.1 Time acceleration

Note that we can rewrite (22) as

$$G(\mathbf{u}^{k+1}) = F(\mathbf{u}^k), \tag{23}$$

and, as G is invertible, the solution can be characterised as a fixed point of the operator $T = G^{-1} \circ F$. This motivates using AA as a fixed-point acceleration algorithm. The acceleration of time iterations to compute steady state computations has already been successfully used in nonlinear

poroelasticity (Barnafi et al., 2024) using AA, so we adopt the same strategy. AA of depth m consists in the following: consider a fixed-point iteration function g , whose iterations are given by $x^{k+1} = g(x^k)$. Then, at each iteration k :

1. Define the matrix $F_k = [f_{k-m}, \dots, f_k]$, where $f_\ell = g(x^\ell) - x^\ell$.
2. Determine $\alpha = (\alpha_0, \dots, \alpha_k)$ that solves

$$\sum_{j=0}^k \alpha_j = 1 \quad \min \|F_k \alpha\|^2.$$

3. Compute the accelerated update

$$x^{k+1} = \sum_{i=0}^k \alpha_i g(x^{k-m+i}). \quad (24)$$

We will denote, for each iterate \mathbf{u}^k , as $AA_m(\mathbf{u}^k)$ the accelerated solution obtained with (24), where the fixed-point map g is given by our solution map $G^{-1} \circ F$.

4.2 The adaptive solver

The steps in our adaptive solver are summarized in Algorithm 1. The algorithm leverages non-conforming forest-of-octrees meshes; see, e.g., (Badia et al., 2020). Forest-of-octrees meshes can be seen as a two-level decomposition of the computational domain (typically an square or cube in the case of DIR) referred to as macro and micro level, respectively. The macro level is a suitable *conforming* partition C_h of into quadrilateral ($d = 2$) or hexahedral cells ($d = 3$). This mesh, which may be generated using for instance an unstructured mesh generator, is referred to as the coarse mesh. At the micro level, each of the cells of C_h becomes the root of an adaptive octree with cells that can be recursively and dynamically refined or coarsened using the so-called 1: 2^d uniform partition rule. If a cell is marked for refinement, then it is split into 2^d children cells by subdividing all parent cell edges. If all children cells of a parent cell are marked for coarsening, then they are collapsed into the parent cell. The union of all leaf cells in this hierarchy forms the decomposition of the domain at the micro level.

Algorithm 1. AA-AMR solution strategy (α , m , N_{ref}^0 , N_{ref} , θ_{coarsen} , θ_{refine}).

```

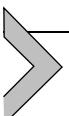
1: Refine a one-element mesh  $N_{\text{ref}}^0$  times to get initial mesh  $M^0$ 
2: Initialise solution vector  $\mathbf{u}$  in  $M^0$ 
3: for  $k$  in  $1, \dots, N_{\text{ref}}$  do
4:   Interpolate solution  $\mathbf{u}$  to current mesh  $\mathbf{u}^0 = \Pi_{M^{k-1}} \mathbf{u}$ 
5:   while Solution not converged do
6:     Set  $\tilde{\mathbf{u}}^{k+1}$  the solution of (4.2) given  $\mathbf{u}^k$ 
7:     Set  $\mathbf{u}^{k+1} = AA_m(\tilde{\mathbf{u}}^{k+1})$  the accelerated solution
8:   end while
9:   Set solution  $\mathbf{u}$  as the last iteration
10:  Compute error estimate (3.3)
11:  Refine  $\theta_{\text{refine}}$  fraction of elements with largest estimator in  $M^{k-1}$ 
12:  Coarsen  $\theta_{\text{coarsen}}$  fraction of elements with smallest estimator in  $M^{k-1}$ 
13:  Set new mesh  $M^k$  and adapt discrete spaces
14: end for
15: return  $\mathbf{u}$ 

```

One important consideration is that AMR with forest-of-octrees allows for coarsening only if there is an initial mesh hierarchy to be coarsened. Because of this, we consider for all problems an initial coarse mesh C_h with only one coarse element (where the images fit), and perform some initial uniform refinements that give us such initial hierarchy. This is relevant as images have in many cases a dark background where accuracy is not important, and thus we expect our error estimator to detect this and coarsen such areas. Because of these considerations, our proposed algorithm consists in the following steps: (i) build an initial mesh by uniformly refining a single-element coarse mesh, (ii) on each mesh solve DIR (3) using scheme (21), (iii) after computing a solution for a given mesh, a θ_{refine} percentage of elements are refined and a θ_{coarsen} percentage of elements are coarsened, and (iv) stop after a given number of mesh adaptations has been performed. We stress that this approach resembles the octree-based approach from [Haber et al. \(2007, Sect. 3\)](#), but with the following key differences: (i) we use a FEM formulation (instead of finite differences), (ii) we guide adaptivity using a theoretically derived residual-based error indicator, and (iii) we combine AMR with AA to speed-up the convergence of the pseudo-time formulation at each mesh level in the hierarchy.

Given that our target problem is (3), the algorithm performs adaptivity only after the pseudo-time simulation based on (22) has reached a stationary state. We acknowledge that further optimisation can be obtained by performing adaptivity instead every fixed number of timesteps, but as the number of timesteps is highly unpredictable, we preferred not to pursuit

this strategy. In addition, the use of AA for the pseudo-time iterations would be otherwise incompatible with adaptivity due to the change of dimensions between different adaptivity steps.



5. Numerical tests

In this section we present numerical examples to illustrate the performance of the proposed adaptive DIR solver. The realisation of this solver is conducted using the tools provided by the open source scientific software packages in the Gridap ecosystem (Badia et al., 2022; Verdugo & Badia, 2022). The sparse linear systems were solved using UMFPACK, as provided by Julia. We used the GridapP4est.jl (Martín, 2025) Julia package in order to handle forest-of-octrees meshes (including facet integration on non-conforming interfaces as per required by the computation of the a posteriori error estimator) and FE space constraints. This package, built upon the p4est meshing engine (Burstedde et al., 2011), is endowed with Morton space-filling curves, and it provides high-performance and low-memory footprint algorithms to handle forest-of-octrees. All numerical tests were performed on a supercomputer node equipped with Intel Xeon Platinum 8274 CPU cores, with Julia 1.10.4 and IEEE double precision. We used `-O3` as the optimisation flag for the Julia compiler.

For simplicity, we will test our algorithm in two brain images, obtained from the BrainWeb Database (Cocosco, 1997; Collins et al., 1998; Kwan et al., 1996, 1999) and both of 129×129 pixels, and on a standard benchmark test known as the OC images, which we stored as 600×400 pixels images. We show both image pairs in Fig. 1. The brain test does not require large deformations to take place, but is nonetheless a challenging problem as brain images have regions with large contrasts, mainly near the cortex. The OC test is instead much more challenging, as the registration we look for require very large deformations and is guided only through the discrete gradient of a field that is fundamentally discontinuous. Unless otherwise stated, all images will be pre-processed with a Gaussian kernel using a variance parameter of $\sigma = 1$ to avoid having discontinuous derivatives, and then interpolated globally using linear B-Splines (De Boor, 1978), so that we can compute the gradient of the template image as required by the ∇T term.

A fundamental issue that we do not address in detail in this work is that whenever we use elements that have more than one image pixel inside,

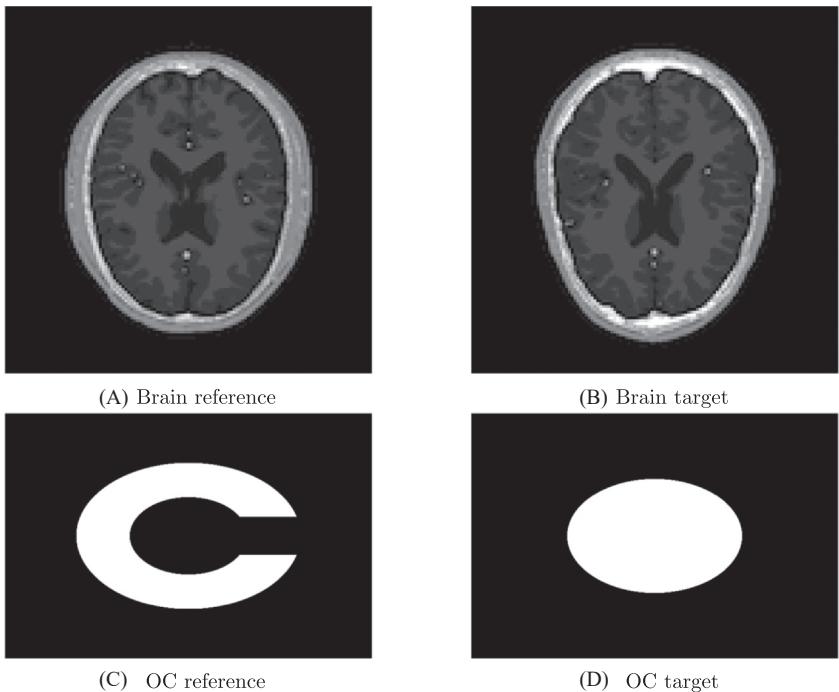


Fig. 1 Reference (left) and target (right) images used in the brain (top) and OC (bottom) tests.

there could be additional numerical integration errors involved in our computations, in particular for those terms involving images. In preliminary studies we have observed this to have a non-negligible influence on the performance of the solvers, and thus we show a sensitivity analysis regarding this issue in [Section 5.5](#) using first order Lagrangian FE. In the experiments, we chose a quadrature order of 6 for all integrals involving images (see the aforementioned section for a formal definition of quadrature order, and the particular kind of quadratures that we used in this work). We envision that the use of adaptive integration rules for n -cubes (such as, e.g., ([Genz & Malik, 1980](#); [Johnson, 2018](#))) for terms involving images may lead to a better trade-off among integration accuracy and computational effort (see also ([Bull & Freeman, 1995](#))). Because of this issue, we have performed all of our tests using only first order elements, as higher order approximations would require further tuning the quadrature rules.

5.1 Convergence verification against manufactured solutions

In order to confirm the accuracy of the proposed schemes we perform two simple tests of convergence with exact solutions. We consider the unit square domain $\Omega = (0, 1)^2$, use synthetic images

$$R(\mathbf{x}) = |\mathbf{x} - (0.2, 0.2)|^2, \quad T(\mathbf{x}) = |\mathbf{x} - (0.8, 0.8)|^2,$$

and use the following smooth and non-smooth displacement solution

$$\begin{aligned} \mathbf{u}_{\text{ex}} &= \frac{1}{10} \begin{pmatrix} \left[-\sin(\pi x) + \frac{1}{\lambda} \cos(\pi x) \right] \sin(\pi y) + \frac{4}{\pi^2} \\ \left[-\cos(\pi x) + \frac{1}{\lambda} \sin(\pi x) \right] \cos(\pi y) \end{pmatrix}, \\ \text{and } \mathbf{u}_{\text{ex}} &= \frac{r^\beta}{10} \begin{pmatrix} \cos(\beta\theta) \\ \sin(\beta\theta) \end{pmatrix}, \end{aligned}$$

respectively. For the non-smooth case $r = \sqrt{x^2 + y^2}$, $\theta = \text{atan2}(y, x)$ are the polar coordinates and $\beta = \frac{2}{3}$ is a regularity index that yields $\mathbf{u} \in \mathbf{H}^{1+\beta}(\Omega)$, and therefore we expect (under uniform refinement) a suboptimal convergence of $O(h^\beta)$. We consider the parameters $\Delta t = \alpha = E = 1$, $\nu = \frac{1}{4}$, and $\beta = \frac{2}{3}$. For the smooth case we take $\kappa = 0.5$ and for the non-smooth case we consider the pure-traction boundary condition (setting the boundary stiffness parameter $\kappa = 0$). Note that the exact solutions above do not induce zero traction boundary conditions for planar elasticity, so we need to also manufacture an exact traction imposed weakly in the formulation. Similarly, the manufactured solution may have a component of rigid body motions and so we also include on the right-hand side a contribution taking into account this part of the kernel. Likewise, an additional contribution is required as manufactured load on the right-hand side of the momentum equation, as well as in the element contribution to the volume error estimator and in the edge/boundary contribution (since the manufactured normal stress is non-homogeneous). In particular, this gives an estimator of the form

$$\begin{aligned} \Theta_K^2 &:= h_K^2 \left\| \frac{1}{\Delta t} \mathbf{u}_h - \mathbf{f}_{\text{ex}} + \mathbf{g}_{\text{ex}} - \alpha \mathbf{f}_{\mathbf{u}_h} - \mathbf{div}(\mathbf{C}\mathbf{e}(\mathbf{u}_h)) \right\|_{0,K}^2 \\ &\quad + \sum_{e \in \mathcal{E}(K) \cap \mathcal{E}_h(\Omega)} h_e \left\| [(\mathbf{C}\mathbf{e}(\mathbf{u}_h) - \boldsymbol{\sigma}_{\text{ex}}) \mathbf{n}_e + \kappa (\mathbf{u}_h - \mathbf{u}_{\text{ex}})] \right\|_{0,e}^2 \\ &\quad + \sum_{e \in \mathcal{E}(K) \cap \mathcal{E}_h(\Gamma)} h_e \left\| (\mathbf{C}\mathbf{e}(\mathbf{u}_h) - \boldsymbol{\sigma}_{\text{ex}}) \mathbf{n}_e + \kappa (\mathbf{u}_h - \mathbf{u}_{\text{ex}}) \right\|_{0,e}^2, \end{aligned}$$

with $\mathbf{f}_{\text{ex}} = -\mathbf{div} \boldsymbol{\sigma}_{\text{ex}}$ and $\mathbf{g}_{\text{ex}} = \alpha \{T(\mathbf{x} + \mathbf{u}_{\text{ex}}) - R\} \nabla T(\mathbf{x} + \mathbf{u}_{\text{ex}})$.

The numerical results of the uniform refinement tests are reported in Tables 1 and 2 for the smooth and non-smooth solutions, respectively. We construct six levels of uniform mesh refinement of the domain, on which we compute approximate solutions and the associated errors in the norm $|\mathbf{v}|_{1,\Omega} = \|\boldsymbol{\epsilon}(\mathbf{v})\|_{0,\Omega}$. Convergence rates are calculated as usual:

$$\text{rate} = \log(e/\hat{e}) [\log(h/\hat{h})]^{-1},$$

where e and \hat{e} denote errors produced on two consecutive meshes of sizes h and \hat{h} , respectively. For the smooth case we see optimal convergence of order $O(h^{k+1})$ and bounded effectivity indexes indepen-

Table 1 Convergence tests against smooth manufactured solutions. Error history of the method for two polynomial degrees and effectivity index associated with the *a posteriori* error estimator on uniform mesh refinement.

Degrees of freedom (DoFs)	h	$ \mathbf{u} - \mathbf{u}_h _{1,\Omega}$	rate	eff (Θ)
------------------------------	-----	--	------	------------------

Uniform mesh refinement, with $k = 1$				
21	0.7071	6.48e-02	*	0.234
53	0.3536	3.29e-02	0.977	0.231
165	0.1768	1.66e-02	0.991	0.214
581	0.0884	8.30e-03	0.997	0.206
2181	0.0442	4.15e-03	0.999	0.202
8453	0.0221	2.08e-03	1.000	0.200
Uniform mesh refinement, with $k = 2$				
53	0.7071	1.32e-02	*	0.062
165	0.3536	3.34e-03	1.985	0.063
581	0.1768	8.39e-04	1.995	0.063
2181	0.0884	2.10e-04	1.996	0.064
8453	0.0442	5.33e-05	1.979	0.065
33285	0.0221	1.51e-05	1.946	0.077

Table 2 Convergence tests. Error history of the method for two polynomial degrees, using a non-smooth manufactured solution, and effectivity index associated with the *a posteriori* error estimator on uniform and adaptive mesh refinement.

DoFs	h	$ \mathbf{u} - \mathbf{u}_h _{1,\Omega}$	rate	eff (Θ)
Uniform mesh refinement, with $k = 1$				
21	0.7071	1.30e-02	*	0.178
53	0.3536	8.60e-03	0.599	0.215
165	0.1768	5.61e-03	0.616	0.228
581	0.0884	3.63e-03	0.628	0.233
2181	0.0442	2.34e-03	0.634	0.237
8453	0.0221	1.50e-03	0.638	0.240
33285	0.0110	9.65e-04	0.640	0.244
Adaptive mesh refinement, with $k = 1$				
53	0.7071	8.60e-03	*	0.215
71	0.3536	6.20e-03	2.240	0.229
97	0.1768	4.65e-03	1.848	0.234
137	0.0884	3.64e-03	1.408	0.237
187	0.0442	2.87e-03	1.530	0.242
259	0.0221	2.37e-03	1.180	0.239
367	0.0110	1.88e-03	1.334	0.227
511	0.0055	1.55e-03	1.156	0.221
Uniform mesh refinement, with $k = 2$				
53	0.7071	6.09e-03	*	0.111
165	0.3536	4.13e-03	0.558	0.184
581	0.1768	2.80e-03	0.563	0.246
2181	0.0884	1.88e-03	0.571	0.284
8453	0.0442	1.26e-03	0.580	0.308

33285	0.0221	8.39e-04	0.588	0.327
132101	0.0110	5.56e-04	0.594	0.345
Adaptive mesh refinement, with $k = 2$				
165	0.7071	4.13e-03	*	0.184
231	0.3536	2.83e-03	2.253	0.235
327	0.1768	1.96e-03	2.100	0.259
477	0.0884	1.37e-03	1.927	0.267
699	0.0442	9.34e-04	1.987	0.269
963	0.0221	6.53e-04	2.231	0.293
1373	0.0110	4.32e-04	2.335	0.294
1981	0.0055	2.77e-04	2.423	0.284

dently of the refinement level. For the non-smooth case we observe the expected sub-optimal convergence under a uniform mesh refinement.

On the other hand, the numerical results for the adaptive mesh refinement case are reported in [Table 2](#). We take (for the two polynomial degrees tested here) the same refinement fraction of 15 %. For the experimental convergence rates of the adaptive case we use the alternative form

$$\text{rate} = -2 \log(e/\hat{e}) [\log(\text{DoFs}/\widehat{\text{DoFs}})]^{-1}.$$

One can see in [Table 2](#) that the optimal convergence is attained under adaptive mesh refinement guided by the *a posteriori* error estimator. Again, the effectivity index remains bounded in all cases. Also, we can readily see that the same level of energy error is reached with the adaptive case using roughly 10 % of the number of DoFs required in the uniform mesh refinement. This is consistent in both first and second order schemes.

To exemplify the performance of the method in the non-smooth solution regime, we plot in [Fig. 2](#) the approximate solution (displacement magnitude warped) in a coarse adapted mesh, the synthetic images (in the undeformed mesh), as well as depictions of the adapted meshes after several steps of refinement, which indicate the expected agglomeration of elements near the origin (where the singularity is).

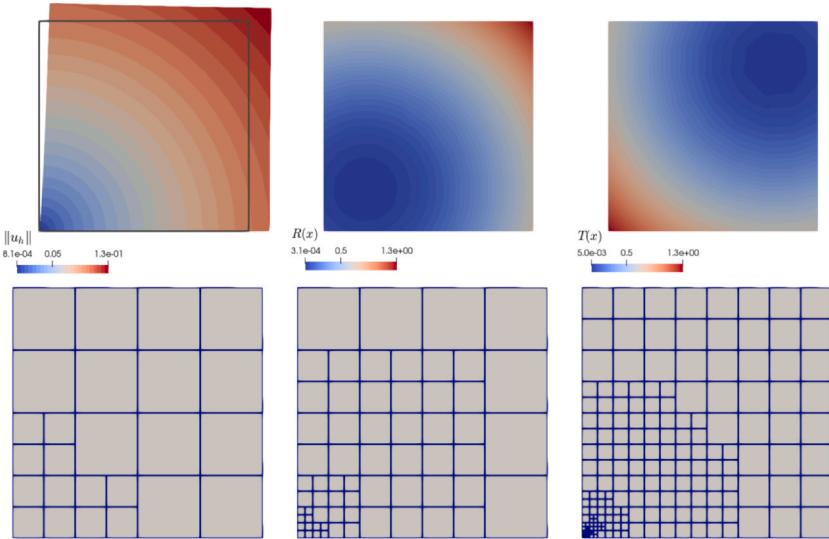


Fig. 2 Convergence test against a singular solution. Approximate displacement magnitude (warped), synthetic reference and target images all on the final adapted mesh (top row), and sample of adaptively refined meshes guided by the *a posteriori* error estimator Θ after 3,5,7 refinement steps (bottom row).

Table 3 Acceleration test, brain images: Similarity measure, iteration count and elapsed time required for convergence for varying AA depth, given by the parameter m .

Scheme	Similarity	Iterations	Elapsed time (s)
No accel	0.0412	2289	3520.81
AA(2)	0.0412	382	651.43
AA(5)	0.0412	581	997.12
AA(10)	0.0412	203	332.04
AA(20)	0.0412	99	157.61

5.2 Anderson acceleration for DIR

In this section we study the performance of the time acceleration method through AA described in Section 4.1. We do this for both the brain and OC images, to evaluate the technique under small and large deformations.

We show the similarity measure, total number of iterations and elapsed time for the brain images registration in Table 3. For this test, we used $\alpha = 10^4$

and $\Delta t = 10^{-5}$. The mesh resolution was set such that there is a single pixel image per each cell. To avoid ambiguities regarding convergence, we used the relative Euclidean vector norm of the stationary residual as a convergence criterion, with a tolerance of 10^{-4} . We highlight that better results were obtained using an L^2 stabilisation in the pseudo-time terms from (21), i.e. $\mathcal{L} = I$ the identity operator. In this case, AA is extremely convenient, as it provides a significant reduction in the number of iterations that increases as the depth parameter m increases, except for the case $m = 5$, which is still more convenient than the non-accelerated strategy. We registered the results until a wider depth was not convenient anymore, as iterations started increasing again, which typically happens because the conditioning of the related least squares problem starts deteriorating. For the largest depth parameter, using AA can yield time accelerations up to a factor of 22 with respect to a non-accelerated approach.

For the OC images test we consider two cases for the values of α given by 10^4 and 10^5 , and as the images have a sharp discontinuity, we considered $\sigma = 4$. We set the size of the pseudo time-step to $\Delta t = 10^{-2}$ and $\Delta t = 10^{-3}$, respectively. The mesh resolution was set in order to have 4×4 image pixels per each cell. As with the brain, we used the relative Euclidean vector norm of the stationary residual as a convergence criterion for this test, but with a tolerance of 10^{-2} . For this problem, solutions are obtained using $\mathcal{L} = I - \Delta$ from (21), as using only the identity yielded diverging iterations in our preliminary tests. This benchmark requires far larger deformations to obtain a satisfactory registration, which we depict by showing the warped target images next to the reference image for both values of α in Fig. 3. For the smaller α case, we show the results in Table 4A, where the results obtained are very similar to those of the brain. Acceleration is very convenient as it drastically reduces the number of iterations with minimal overhead per iteration. This yields elapsed time reductions of up to a factor of 3.7. In spite of this, we note that, as shown in Table 4B, acceleration is ineffective when larger displacements are involved, so that all accelerated iterations achieved the maximum number of iterations allowed (10, 000).

5.3 Adaptivity performance on brain images

Starting from a mesh of one element, we considered 4 initial uniform refinements, and perform 5 adaptive mesh refinements with refine and coarse parameters given by $\theta_{\text{refinement}} = 0.4$ and $\theta_{\text{coarsen}} = 0.2$

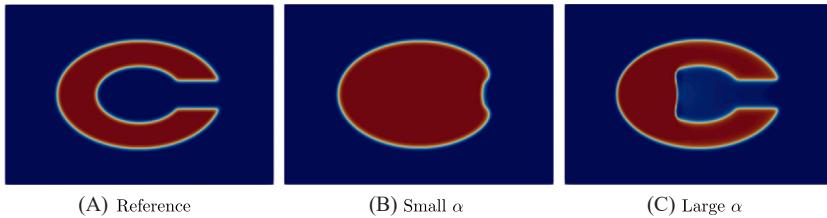


Fig. 3 OC test solutions for small and large similarity parameter (10^4 and 10^5 respectively).

Table 4 Acceleration test, OC images: Similarity measure, iteration count and elapsed time required for convergence for varying AA depth, given by the parameter m .

Scheme	Similarity	Iterations	Time (s)
No accel	0.263	113	78.43
AA(2)	0.263	39	28.58
AA(5)	0.263	37	27.03
AA(10)	0.264	28	21.02
AA(20)	0.264	28	21.93

(A) Small α

Scheme	Similarity	Iterations	Time (s)
Scheme	Similarity	Iterations	time (s)
No accel	0.0846	8927	6123
AA(2)	—	—	—
AA(5)	—	—	—
AA(10)	—	—	—
AA(20)	—	—	—

(B) Large α

respectively. We considered a tolerance of 10^{-4} for the velocity $\frac{\mathbf{u}^{k+1} - \mathbf{u}^k}{\Delta t}$ in the reference (non-adapted) case, and a tolerance of 10^{-2} for all the adaptive cases. We performed all tests using AA with a depth parameter of

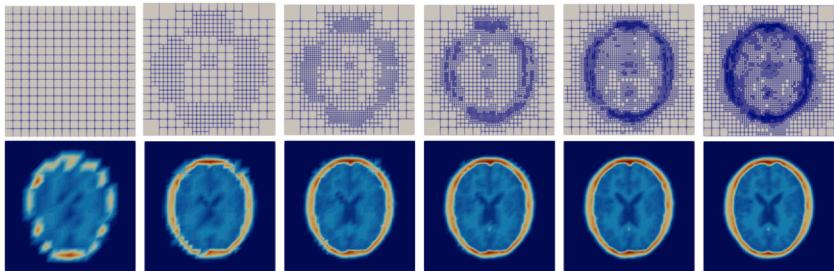


Fig. 4 Brain AMR test: evolution of AMR solution through all adaptive steps from left to right. In the top row we show the evolution of the adapted grid, and on the bottom row we show the evolution of the solution.

$m = 10$. We have seen that considering an equally precise solution in all levels of the adaptive solver leads to over-solving, and thus the resulting scheme that we consider resembles an inexact-Newton procedure. To obtain a solution with better similarity than the one in [Table 3](#) we used $\alpha = 10^5$, and set $\Delta t = 10^{-6}$ as larger timesteps resulted in non-convergence (in the form of oscillating iterations).

In [Fig. 4](#) we show the evolution of the meshes obtained from our algorithm, and compare the solution obtained with a reference mesh of one-element-per-pixel in [Fig. 5](#). The solutions are indistinguishable to the eye, so we provide further comparison information in [Table 5](#), where we show the final number of DoFs, similarity, total iterations, and required elapsed time. While the AMR solution requires a larger total number of iterations, most of these are performed on coarser meshes where iterations are much faster. This results in the overall solution time being accelerated by a factor of 13.62. This faster solution yielded a mildly lower similarity measure, and this was additionally achieved with less DoFs. We stress that this test does not converge with the parameter values at hand if AA is not used.

5.4 Adaptivity performance on OC images

The setup of the experiments for the OC images in this section is almost equivalent to that of the brain images, with the following differences. First, we considered a tolerance of 10^{-2} for the residual in the reference (non-adapted) case, and a tolerance of 10^{-1} for all the adaptive cases. Second, we used $\alpha = 10^8$ to obtain a more accurate solution than the one in

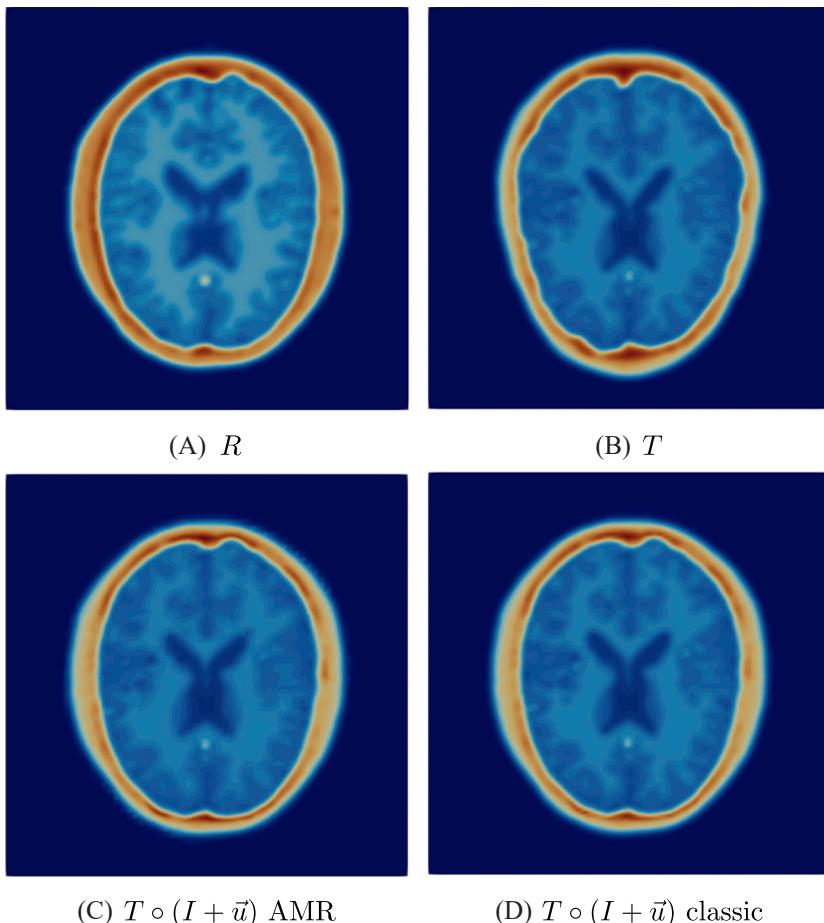


Fig. 5 Brain AMR test: comparison of AMR and classic solutions (bottom left and bottom right respectively) in comparison to original images (top row).

Table 5 Brain AMR test: solution metrics, given by (A) total degrees of freedom (DoFs), (B) final similarity, (C) iterations, where we display the iterations incurred by the adaptive solver in each level, and (D) the overall elapsed time.

Strategy	DoFs	Similarity	Iterations	Elapsed time (seconds)	
Classic	33,803	0.0263		927	1649.94
Adaptive	24,843	0.0260	(616, 321, 76, 43, 27, 24)	1107	121.14

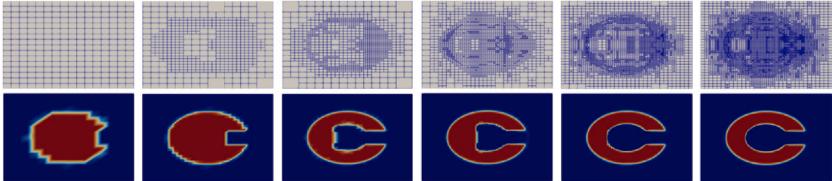


Fig. 6 OC AMR test: evolution of AMR solution through all adaptive steps from left to right. In the top row we show the evolution of the adapted grid, and on the bottom row we show the evolution of the solution.

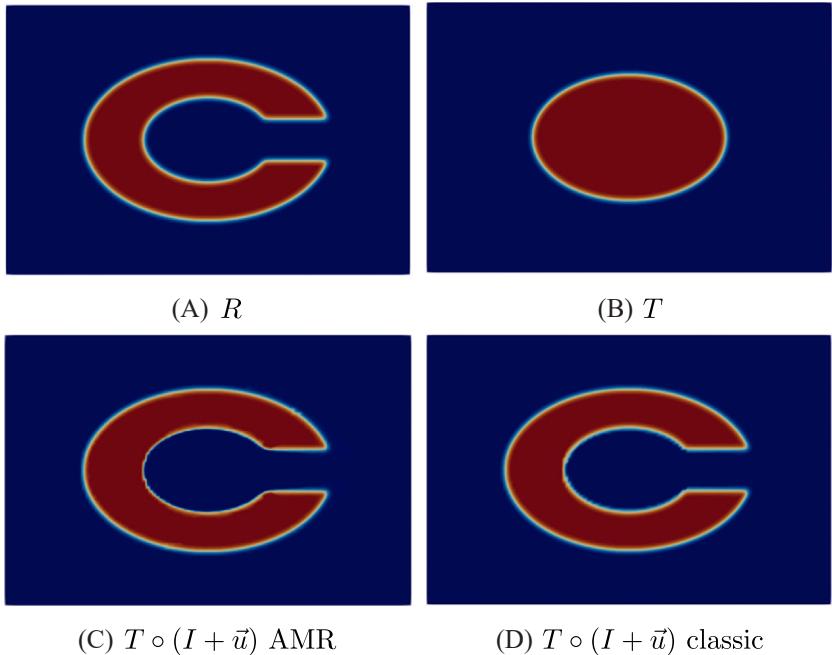


Fig. 7 OC AMR test: comparison of AMR and classic solutions (bottom left and bottom right respectively) in comparison to original images (top row).

[Table 4a](#) and [4b](#), and set $\Delta t = 10^{-6}$, which yields convergence (without AA). Third, we did not use AA, as it did not yield to a convergent fixed-point iteration scheme for the values of α and Δt at hand.

In [Fig. 6](#) we show the evolution of the meshes obtained from our algorithm, and compare the solution obtained with a reference mesh of four-pixels-per-element in [Fig. 7](#). The solutions are almost indistinguishable to the eye, so we provide further comparison information in [Table 6](#)

Table 6 OC AMR test: solution metrics, given by (A) total degrees of freedom (dofs), (B) final similarity, (C) iterations, where we display the iterations incurred by the adaptive solver in each level in gray, and (D) the overall elapsed time.

Strategy	DoFs	Similarity	Iterations	Elapsed time (seconds)
Classic	30,505	0.0196	1478	1003.88
Adaptive	23,953	0.0156	(21, 124, 549, 217, 662, 295) 1868	478.57

(see description of Table 5 above for a description of the different fields in the table). We highlight that the overall solution time was accelerated by a factor of 2.09. This faster solution yielded a lower similarity measure, and this was achieved with less DoFs.

5.5 Quadrature sensitivity

Image functions are highly nonlinear, which makes the computation of integrals that depend on them highly prone to numerical integration (quadrature) errors. It is difficult to obtain analytic results that can give sharp estimates for this, so instead we tried the following approach: depending of the pixels-per-element that one can have, we compute the quadrature order (i.e. maximum polynomial degree that is integrated exactly (Ern & Guermond, 2004)) that provides minimal variations of the residual. Naturally, one would expect that the larger the number of pixels-per-element, the larger the required quadrature order to commit approximately the same amount of numerical integration error. To test this, we considered as the ground truth the residual vector

$$[W(q)]_i = \int_{\Omega} (T(\mathbf{x}) - R(\mathbf{x})) \nabla T(\mathbf{x}) \cdot \mathbf{v}_i \, d\mathbf{x}_q,$$

where \mathbf{v}_i is a basis function of the considered FE space, and we denote with $d\mathbf{x}_q$ the numerical integration performed with a quadrature of order q . With it, the error will be denoted with

$$e(q) = \frac{|W(q) - W(q_{\text{truth}})|}{|W(q_{\text{truth}})|}.$$

We consider the ground truth integration value to be $q_{\text{truth}} = 201$. In this setting, we computed the error for various numbers of pixels-per-element, and display the errors for different Gaussian smoothing levels and

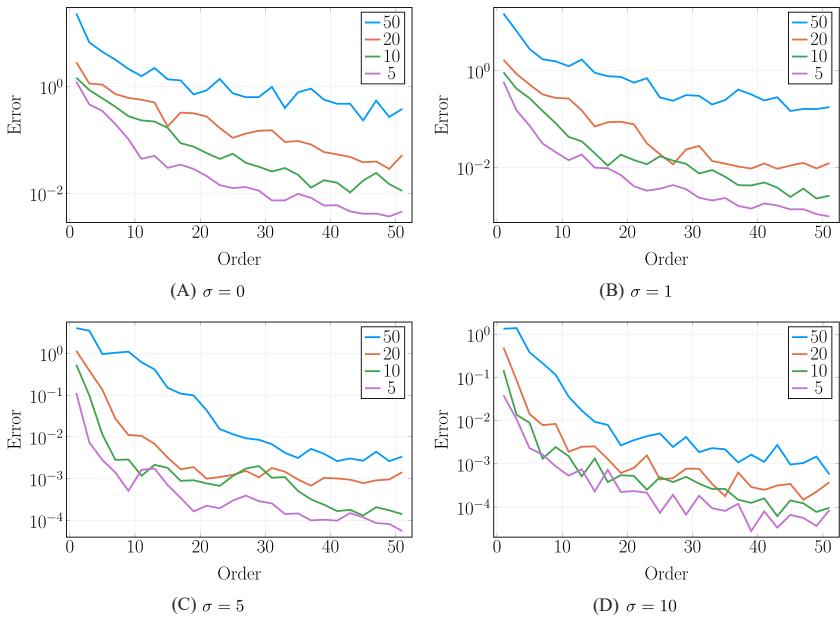


Fig. 8 Integration error for 5×5 , 10×10 , 20×20 , and 50×50 pixels per element considering odd quadrature orders ranging from 1 to 51. From top to bottom, and left to right, we consider this setting using a Gaussian smoothing of 0, 1, 5, and 10 with the brain images.

the brain images in Fig. 8. We use standard tensor-product Gauss-Legendre quadratures for the experiment. Integration error is thus significant, and it becomes much more relevant when no smoothing is used. Considering that with AMR we have coarser elements where there are no complex feature of the images, we have found that a good compromise is considering $q = 6$. Still, this will be addressed more effectively in future work.



6. Conclusions and future perspectives

In this work, we have extended the *a-posteriori* error analysis of DIR with linear elastic regularisation to the case of Robin boundary conditions. We have formulated an efficient strategy for leveraging octree-based adaptivity that supports both coarsening and refinement, and have tested our strategy on realistic brain images and on the challenging OC

benchmark. The DIR problem is highly nonlinear, and so we solve it with an IMEX formulation of a proximal-point algorithm, and further proposed to accelerate this algorithm with AA. We have observed that acceleration can be very convenient computationally, as it significantly reduced the elapsed time of the brain test. Still, the methodology is highly problem dependent, as it was ineffective for the OC test when using larger values of α . Putting everything together, we were able to significantly reduce the computational time required to solve DIR. The proposed AMR approach is able to provide better resolution on difficult domain regions and instead relax it where not required, which results in better solutions obtained with fewer DoFs and in less time.

We will focus our future work on developing black-box nonlinear solvers for the DIR problem, so that we do not need to compute a pseudo-time step for convergence. We will provide a deeper analysis of the interplay between mesh size and solver performance, as the sensitivity of proximal-point algorithms with respect to mesh-size remains understudied, even more so in DIR. Then, we will leverage such robust solvers to develop an automated computation of the similarity parameter α , and extend our software to the 3D case. Our long term goal is that of developing an open source software that provides a robust and efficient solution of DIR without the requirement of intense parameter tuning.

Funding

This work has been partially supported by the Monash Mathematics Research Fund S05802–3951284; by the Australian Research Council through the FUTURE FELLOWSHIP grant FT220100496 and DISCOVERY PROJECT grant DP22010316; and by the National Research and Development Agency (ANID) of the Ministry of Science, Technology, Knowledge and Innovation of Chile through the grant FONDECYT de Postdoctorado No. 3230326 and the Center for Mathematical Modeling (CMM), Proyecto Basal FB210005. This work was also supported by computational resources provided by the Australian Government through NCI under the ANU Merit Allocation Scheme (ANUMAS).

References

- Badia, S., Martín, A. F., Neiva, E., & Verdugo, F. (2020). A generic finite element framework on parallel tree-based adaptive meshes. *SIAM Journal on Scientific Computing*, 42, C436–C468.
- Badia, S., Martín, A. F., & Verdugo, F. (2022). GridapDistributed: A massively parallel finite element toolbox in julia. *Journal of Open Source Software*, 7, 4157.
- Barnafi, N. A., Gatica, G. N., & Hurtado, D. E. (2018). Primal and mixed finite element methods for deformable image registration problems. *SIAM Journal on Imaging Sciences*, 11, 2529–2567.

- Barnafi, N. A., Gatica, G. N., Hurtado, D. E., Miranda, W., & Ruiz-Baier, R. (2021). Adaptive mesh refinement in deformable image registration: A posteriori error estimates for primal and mixed formulations. *SIAM Journal on Imaging Sciences*, 14, 1238–1272.
- Barnafi, N. A., Petras, A., & Gerardo-Giorda, L. (2024). Fully nonlinear inverse poroelasticity: Stress-free configuration recovery. *Computer Methods in Applied Mechanics and Engineering*, 425, 116960.
- Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, 59, 65–98.
- Brenner, S. C. (2008). *The mathematical theory of finite element methods*. Springer.
- Bull, J., & Freeman, T. (1995). Parallel globally adaptive algorithms for multi-dimensional integration. *Applied Numerical Mathematics*, 19, 3–16.
- Burstedde, C., Wilcox, L. C., & Ghantas, O. (2011). p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing*, 33, 1103–1133.
- Christensen, G. E., Rabbitt, R. D., & Miller, M. I. (1996). Deformable templates using large deformation kinematics. *IEEE Transactions on Image Processing*, 5, 1435–1447.
- Clément, P. (1975). Approximation by finite element functions using local regularization. *RAIRO: Analyse Numérique*, 9, 77–84.
- Cocosco, C. A., Kollokian, V., Kwan, R. K.-S., & Evans, A. C. (1997). “BrainWeb: Online Interface to a 3D MRI Simulated Brain Database” NeuroImage, vol. 5, no. 4, part 2/4, S425, 1997 – Proceedings of 3-rd International Conference on Functional Mapping of the Human Brain, Copenhagen.
- Collins, D. L., Zijdenbos, A. P., Kollokian, V., Sled, J. G., Kabani, N. J., Holmes, C. J., & Evans, A. C. (1998). Design and construction of a realistic digital brain phantom. *IEEE Transactions on Medical Imaging*, 17, 463–468.
- De Boor, C. (1978). *A practical guide to splines*. New York: Springer.
- De Vos, B. D., Berendsen, F. F., Viergever, M. A., Sokooti, H., Staring, M., & Išgum, I. (2019). A deep learning framework for unsupervised affine and deformable image registration. *Medical Image Analysis*, 52, 128–143.
- Deshpande, V. S., & Bhatt, J. S. (2019). Bayesian deep learning for deformable medical image registration. *International conference on pattern recognition and machine intelligence*. Springer, 41–49.
- Ern, A., & Guermond, J.-L. (2004). *Theory and practice of finite elements*. Springer.
- Genet, M., Stoeck, C. T., von Deuster, C., Lee, L. C., & Kozerke, S. (2018). Equilibrated warping: Finite element image registration with finite strain equilibrium gap regularization. *Medical Image Analysis*, 50, 1–22.
- Genz, A. C., & Malik, A. A. (1980). Remarks on algorithm 006: An adaptive algorithm for numerical integration over an n-dimensional rectangular region. *Journal of Computational and Applied Mathematics*, 6, 295–302.
- Haber, E., Heldmann, S., & Modersitzki, J. (2007). An octree method for parametric image registration. *SIAM Journal on Scientific Computing*, 29, 2008–2023.
- Haber, E., Heldmann, S., & Modersitzki, J. (2008). Adaptive mesh refinement for non-parametric image registration. *SIAM Journal on Scientific Computing*, 30, 3012–3027.
- Henn, S., & Witsch, K. (2004). Multimodal image registration using a variational approach. *SIAM Journal on Scientific Computing*, 25, 1429–1447.
- Johnson, R. W. (2018). Algorithm 988: AMGKQ: An efficient implementation of adaptive multivariate Gauss-Kronrod quadrature for simultaneous integrands in Octave/MATLAB. *ACM Transactions on Mathematical Software*, 44, 1–19.
- Kaplan, A., & Tichatschke, R. (1998). Proximal point methods and nonconvex optimization. *Journal of Global Optimization*, 13, 389–406.
- Kazemi, P., & Renka, R. J. (2012). A Levenberg–marquardt method based on sobolev gradients. *Nonlinear Analysis, Theory, Methods and Applications*, 75, 6170–6179.

- Kwan, R. K.-S., Evans, A. C., & Pike, G. B. (1996). An extensible mri simulator for post-processing evaluation. *International conference on visualization in biomedical computing*. Springer, 135–140.
- Kwan, R.-S., Evans, A. C., & Pike, G. B. (1999). Mri simulation-based evaluation of image-processing and classification methods. *IEEE Transactions on Medical Imaging*, 18, 1085–1097.
- Lee, E., & Gunzburger, M. (2010). An optimal control formulation of an image registration problem. *Journal of Mathematical Imaging and Vision*, 36, 69–80.
- Mai, V., & Johansson, M. (2020). Anderson acceleration of proximal gradient methods. In *International conference on machine learning*, PMLR (pp. 6620–6629).
- Martín, A. F. (2025). Gridapp4est.jl. Available at <https://github.com/gridap/GridapP4est.jl>.
- Modersitzki, J. (2003). *Numerical methods for image registration*. Oxford: Oxford Science Publications.
- Nocedal, J., & Wright, S. J. (1999). *Numerical optimization*. Springer.
- Pawar, A., Zhang, Y., Jia, Y., Wei, X., Rabczuk, T., Chan, C. L., & Anitescu, C. (2016). Adaptive FEM-based nonrigid image registration using truncated hierarchical B-splines. *Computational and Applied Mathematics*, 72, 2028–2040.
- Rockafellar, R. T. (1976). Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14, 877–898.
- Sotiras, A., Davatzikos, C., & Paragios, N. (2013). Deformable medical image registration: A survey. *IEEE Transactions on Medical Imaging*, 32, 1153–1190.
- Toth, A., & Kelley, C. T. (2015). Convergence analysis for anderson acceleration. *SIAM Journal on Numerical Analysis*, 53, 805–819.
- Verdugo, F., & Badia, S. (2022). The software design of gridap: A finite element package based on the julia JIT compiler. *Computer Physics Communications*, 276, 108341.
- Verfürth, R. (1994). A posteriori error estimation and adaptive mesh-refinement techniques. *Journal of Computational and Applied Mathematics*, 50, 67–83.
- Verfürth, R. (1996). *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. Oxford University Press.
- Verfürth, R. (1999). A review of a posteriori error estimation techniques for elasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 176, 419–440.
- Walker, H. F., & Ni, P. (2011). Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis*, 49, 1715–1735.
- Zhang, J., Wang, J., Wang, X., & Feng, D. (2013). The adaptive FEM elastic model for medical image registration. *Physics in Medicine and Biology*, 59, 97.