A projective hybridizable discontinuous Galerkin mixed method for second-order diffusion problems

Loïc Dijoux, Vincent Fontaine, Thierry A. Mara

A projective hybridizable Raviart–Thomas method for second-order diffusion problems

Loic Dijoux[a], Vincent Fontaine[a,*], Thierry Alex Mara[b]

[a]*Laboratoire* PIMENT, *Université de La Réunion, France*
[b]*Joint Research Centre, European Commission, Ispra (VA), Italy*

Abstract

In this paper, we present a projective hybridizable Raviart-Thomas mixed method (H-RT$^{\mathfrak{p}}$) for second-order diffusion problems. The proposed method is inspired by the hybridizable discontinuous Galerkin (HDG) formalism, as it introduces a residual flux terms in the (hybridized version of) Raviart–Thomas mixed method. Specifically, we add a projective-type stabilization function in the definition of the normal trace of the flux on the mesh skeleton. Indeed, we use *broken* Raviart–Thomas space of degree $k \geq 0$ for the flux, a piecewise polynomial of degree $k+1$ for the potential, and a piecewise polynomial of degree k for its numerical trace. This *unconventional* polynomial combination is made possible in the general framework of HDG methods thanks to the *projective-based* stabilization function introduced. The convergence and accuracy of the H-RT$^{\mathfrak{p}}$ method are investigated through numerical experiments in two-dimensional space by using h- and p-refinement strategies. An optimal convergence order $(k+1)$ for the $\boldsymbol{H}(\text{div})$-conforming flux variable (obtained after a straightforward reconstruction) and superconvergence $(k+2)$ for the potential (without any postprocessing) is observed. Comparative tests with the classical H-RT method and the well-known hybridizable local discontinuous Galerkin (H-LDG) method are also performed and exposed in terms of CPU time and mesh refinement.

Keywords: Hybridizable discontinuous Galerkin, projective-based stabilization function, higher-order Raviart–Thomas space, simplified postprocessing, h- and p-refinements

1. Introduction

Consider the second-order diffusion model problem

$$\nabla \cdot (-\boldsymbol{\kappa}^{-1}\boldsymbol{\nabla}u) = f \quad \text{in} \quad \Omega, \tag{1a}$$

$$u = g \quad \text{on} \quad \partial\Omega. \tag{1b}$$

where $\Omega \subset \mathbb{R}^d$ is a bounded polyhedral domain $(d \geq 2)$ with boundary $\partial\Omega \subset \mathbb{R}^{d-1}$. The boldface fonts are used throughout this paper to characterize any vector-valued or matrix-valued functions. To ensure that the problem (1) is well posed, the following assumptions

are assumed to be satisfied: $\boldsymbol{\kappa} \in [L^\infty(\Omega)]^{d,d}$ is a matrix-valued function that is symmetric and positive definite in Ω, $f \in L^2(\Omega)$, and $g \in H^{1/2}(\partial\Omega)$ is a prescribed Dirichlet boundary condition. System (1) can be used to model, for instance, the groundwater flow in porous media in which the potential u represents the pressure head, f a sink/source term and $\boldsymbol{\kappa}^{-1}$ the permeability tensor [1]. In this physical framework, it is convenient to introduce the Darcy velocity $\boldsymbol{\sigma}$ (i.e, the flux) as a supplementary unknown such that the problem (1) can be rewritten as a first-order system,

$$\boldsymbol{\kappa}\boldsymbol{\sigma} = -\boldsymbol{\nabla}u \quad \text{in} \quad \Omega, \tag{2a}$$

$$\nabla \cdot \boldsymbol{\sigma} = f \quad \text{in} \quad \Omega, \tag{2b}$$

$$u = g \quad \text{on} \quad \partial\Omega. \tag{2c}$$

The $\boldsymbol{H}(\text{div})$-conforming mixed finite element (MFE) methods are very widely used for solving the system (2) (see, e.g., [1, 2, 3, and references therein]). Several successful combinations of compatible polynomial interpolation functions for the flux $\boldsymbol{\sigma} \in \boldsymbol{H}(\text{div}; \Omega)$ and the potential $u \in L^2(\Omega)$ satisfying the discrete *inf-sup* condition have been proposed in the literature (e.g., the Raviart–Thomas (**RT**) [3] and the Brezzi–Douglas–Marini (**BDM**) [2] families). These formulations provide numerical solutions $(\boldsymbol{\sigma}_h, u_h)$ with optimal convergence in L^2 and continuous normal components for $\boldsymbol{\sigma}_h$ at the interelement boundaries [4, 2, 1]. However, MFE methods lead to a saddle-point problem with large coupled degrees of freedom, which is quite challenging and time consuming to solve (see, e.g. [2, 5, 6, and references therein]).

To circumvent these issues, mixed methods using the nonconforming Galerkin finite elements have been extensively studied over the last two decades for stationary diffusion problems (see, e.g., [7, and references therein]). The term *nonconforming* means that no regularity assumptions are made on the discrete variables. These methods were developed within the general framework of discontinuous Galerkin (DG) formalism and consider a combination of completely discontinuous approximation spaces for both the flux and potential [8]. Among the DG methods, the local discontinuous Galerkin (LDG) method is of particular interest due to its local conservation properties and its flexibility in handling adaptive hp refinement. In contrast to the standard MFE methods, the discrete flux variable $\boldsymbol{\sigma}_h$ can be easily eliminated locally, thus reducing the linear system to only the primal variable u_h unknowns. However, relative to standard $\boldsymbol{H}(\text{div})$-conforming MFE methods, the LDG method provides a less accurate approximation of discrete variables $(u_h, \boldsymbol{\sigma}_h)$. According to Corkburn *et al.* [9], this lack of robustness is due to the fact that the numerical trace $\hat{u}_h$ is expressed solely in terms of u_h, independent of $\boldsymbol{\sigma}_h$.

In response to these drawbacks, the *hybridization technique* has been introduced in [10, 11] using Lagrange multipliers and static condensation and, reducing the global coupled linear system to only unknowns located on the skeleton of the mesh [12]. This condition is achieved by introducing a new discrete single-valued variable $\hat{u}_h$ that is merely the so-called *numerical trace* of the potential defined at the element boundaries [11]. This technique allows one to establish *local solvers* in each element with $\hat{u}_h$ playing the role of the Dirichlet boundary conditions. Therefore, the interior unknowns $(\boldsymbol{\sigma}_h, u_h)$ can be easily eliminated (by static condensation) from the set of algebraic equations that are now expressed in terms of Lagrange multipliers $\hat{u}_h$ only. The problem is then closed and the global algebraic linear system is assembled by imposing *transmission conditions* throughout the Lagrange multipliers. The hybridizable discontinuous Galerkin (HDG) formalism introduced first by Cockburn *et al.* (2009)

2

represents a unified framework [11] that includes the well-established hybridized mixed finite element (H-MFE) methods by using a null-stabilization function [2, 3, 10], the hybridizable LDG (H-LDG) methods by using a multiplicative stabilization function [9, 11, 12, 13], and the recently developed hybrid higher-order (HHO) methods by using a projective stabilization function [14, 15, 16, 17]. All of these hybrid methods are established by suitably choosing the stabilization function and subsequently the local approximation spaces in addition to the postprocessing step. Note that the HDG formalism has several advantageous features and is well suited for parallel computing. Since its introduction, HDG methods have benefited from intensive research and development (see, e.g., [12, 13, 17, 18, and references therein]) and have been applied to a large class of physical problems [19, 20, 21, 22].

The aim of the present paper is to introduce novel perspectives in the hybridization of mixed finite element methods directly inspired by the HDG formalism. More precisely, we propose a *penalized* formulation of the Hybridized Raviart–Thomas method, denoted H-RT$^\mathbb{p}$, by adding a projective-type stabilization function in the definition of the normal trace of the flux on the mesh skeleton. As a result, the proposed method differs from the standard H-RT method and indeed belongs to the general class of HDG mixed methods. The introduction of these additional stability terms modifies the conventional polynomial combination commonly used for the approximation of the discrete variables. Thus, on each simplex A of the mesh, the higher-order Raviart–Thomas space $\mathbf{RT}_k(A)$ with $k \geq 0$ for the flux $\boldsymbol{\sigma}_h$ is used, a piecewise polynomial of degree $k + 1$ (i.e., one degree higher than for the standard H-RT$_k$ method) for the potential u_h and a piecewise polynomial of degree k for its numerical trace $\hat{u}_h$ on ∂A. This *unconventional* polynomial combination is made possible by the *Lehrenfeld–Schöberl* stabilization function [19]. One distinctive feature of this stabilization function is that it uses a suitable L^2-projection operator, allowing for the consideration of polynomials of higher degree for the approximation of the potential. Thus, no postprocessing is required to improve the accuracy of the potential u_h since it converges naturally at the order $k + 2$, and a straightforward flux reconstruction is sufficient to obtain an $\boldsymbol{H}(\mathrm{div})$–conforming variable $\boldsymbol{\sigma}_h^\star$ converging at the order $k + 1$. Note that the approach presented here can be extended to any other compatible mixed finite element such as those proposed by the authors in [2] (e.g., BDM or BDFM spaces) and to any other stabilization function (e.g., multiplicative-type).

The rest of this paper is organized as follows. In Section 2, we describe the general formalism of HDG methods, define the local solvers and the global problem and present the static condensation from an algebraic viewpoint. In Section 3, we review selected existing stabilization functions and provide details concerning the postprocessing of $\boldsymbol{\sigma}_h$ and u_h. In Section 4, we describe the projective H-RT$^\mathbb{p}$ method; in particular, we review the *Lehrenfeld–Schöberl* stabilization function, define the local approximation spaces and describe the simplified postprocessing to estimate the flux. In Section 5, numerical experiments are investigated using h- and p-refinement strategies, and comparisons are exposed in terms of memory requirements, CPU time and convergence properties. Finally, we end with concluding remarks and perspectives.

2. Hybridizable discontinuous Galerkin methods

2.1. Notations

Let us now introduce the notation that will be used throughout this paper. We denote by $\Omega_h := \{A\}$ (the mesh) a triangulation of the domain Ω into affine-mapped simplexes (triangles

3

if $d = 2$ and tetrahedron if $d = 3$) and by $\partial\Omega_h := \{\partial A\}$. Let $\mathcal{E}_h^{\mathfrak{a}} = \mathcal{E}_h^{\mathfrak{i}} \cup \mathcal{E}_h^{\mathfrak{b}}$ (the mesh skeleton) be the set of all facets (edges if $d = 2$ and faces if $d = 3$) of Ω_h, where $\mathcal{E}_h^{\mathfrak{i}}$ and $\mathcal{E}_h^{\mathfrak{b}}$ denote the set of interior and boundary facets, respectively. We say that $\mathcal{F}$ is an interior facet $\mathcal{F} \in \mathcal{E}_h^{\mathfrak{i}}$ if there exists A_1 and A_2 in Ω_h such that $\mathcal{F} = A_1 \cap A_2$. Furthermore, we assume that the $(d-1)$-Lebesgue measure of $\mathcal{F}$ is not null. For all $A \in \Omega_h$ and $\mathcal{F} \in \mathcal{E}_h^{\mathfrak{a}}$, $|A|$ and $|\mathcal{F}|$ represent the measure of A and $\mathcal{F}$, respectively. We set $h_A := \mathrm{diam}(A)$ and $h := \max_{A \in \Omega_h}(h_A)$ i.e., the maximal element diameter. Hereafter, for any domain $D \subset \mathbb{R}^d$, we denote by $(\cdot, \cdot)_D$ and by $\langle \cdot, \cdot \rangle_{\partial D}$ the standard L^2-inner products in $L^2(D)$ and $L^2(\partial D)$, respectively. Let us introduce the following notations associated with the weak formulation:

$$(\cdot, \cdot)_{\Omega_h} := \sum_{A \in \Omega_h} (\cdot, \cdot)_A, \quad \langle \cdot, \cdot \rangle_{\partial\Omega_h} := \sum_{A \in \Omega_h} \langle \cdot, \cdot \rangle_{\partial A} \quad \text{and} \quad \langle \cdot, \cdot \rangle_{\mathcal{E}_h^{\mathfrak{s}}} := \sum_{\mathcal{F} \in \mathcal{E}_h^{\mathfrak{s}}} \langle \cdot, \cdot \rangle_{\mathcal{F}}, \qquad (3)$$

where the index $\mathfrak{s} = \{\mathfrak{a}, \mathfrak{i}, \mathfrak{b}\}$. Let $\mathcal{P}_k^d(D)$ denote the space of polynomial functions of degree not exceeding $k \geq 0$ on D, and let $\boldsymbol{\mathcal{P}}_k^d(D) := [\mathcal{P}_k^d(D)]^d$. Similarly, we set $\boldsymbol{L}^2(\Omega_h) := [L^2(\Omega_h)]^d$. Let us now introduce the *jump* $[\![\cdot]\!]$ and *weighted average* $\{\cdot\}_{\boldsymbol{\omega}}$ trace operators defined on interior facets as follows:

$$[\![\boldsymbol{v}]\!] := \boldsymbol{v}_1 \cdot \boldsymbol{n}_1 + \boldsymbol{v}_2 \cdot \boldsymbol{n}_2, \qquad (4)$$

$$\{\varphi\}_{\boldsymbol{\omega}} := \omega_1 \varphi_1 + \omega_2 \varphi_2, \qquad (5)$$

for all $(\boldsymbol{v}, \varphi) \in \boldsymbol{L}^2(\Omega_h) \times L^2(\Omega_h)$, where the vector of weights $\boldsymbol{\omega} := (\omega_1, \omega_2)$ is such that $\omega_1 + \omega_2 = 1$. Similarly, we introduce the complement of $\boldsymbol{\omega}$ denoted $\overline{\boldsymbol{\omega}}$ defined as follows:

$$\overline{\boldsymbol{\omega}} := \mathbf{1} - \boldsymbol{\omega} = (\omega_2, \omega_1). \qquad (6)$$

If $\boldsymbol{\omega} = (1/2, 1/2)$, we then recover the classical average operator; we omit the subscript $\boldsymbol{\omega}$ in its definition. For the HDG discretization, two types of finite element spaces are required. The first ones for $\boldsymbol{\sigma}_h$ and u_h are defined inside the elements:

$$\boldsymbol{\mathcal{V}}_h := \{\boldsymbol{v} \in \boldsymbol{L}^2(\Omega_h) : \boldsymbol{v}|_A \in \boldsymbol{\mathcal{V}}(A), \ \forall A \in \Omega_h\}, \qquad (7)$$

$$\mathcal{Q}_h := \{q \in L^2(\Omega_h) : q|_A \in \mathcal{Q}(A), \ \forall A \in \Omega_h\}, \qquad (8)$$

and the second ones for the trace function $\hat{u}_h$ are defined on the mesh skeleton:

$$\Lambda_h^g := \{\mu \in L^2(\mathcal{E}_h^{\mathfrak{a}}) : \mu|_{\mathcal{F}} \in \Lambda(\mathcal{F}) \ \forall \mathcal{F} \in \mathcal{E}_h^{\mathfrak{i}} \ \text{and} \ \mu|_{\mathcal{F}} = \Pi_h g \ \forall \mathcal{F} \in \mathcal{E}_h^{\mathfrak{b}}\}, \qquad (9)$$

where Π_h denotes the L^2-projection onto $\Lambda(\mathcal{F})$. Following (9), the numerical trace of the potential is defined as follows:

$$\hat{u}_h = \begin{cases} \lambda_h & \text{on} \quad \mathcal{E}_h^{\mathfrak{i}} \\ \Pi_h g & \text{on} \quad \mathcal{E}_h^{\mathfrak{b}} \end{cases}, \qquad (10)$$

where $\lambda_h \in \Lambda_h^0$ denotes the Lagrange multipliers. For convenience, let us introduce the local space associated with the degrees of freedom of $\hat{u}_h$ on the boundary of the element A,

$$\Lambda(\partial A) := \{\mu \in L^2(\partial A) : \mu|_{\mathcal{F}} \in \Lambda(\mathcal{F}) \ \forall \mathcal{F} \subset \partial A\}. \qquad (11)$$

Furthermore, we assume for the moment that $\boldsymbol{\mathcal{V}}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\partial A)$ are suitably chosen local approximation spaces of finite dimensions that we define precisely below.

4

2.2. Local solvers & Global problem

The HDG methods provide an approximation of $(\boldsymbol{\sigma}_h, u_h)$ in terms of solutions of a local Dirichlet boundary value problem on each element of the mesh, which are then patched together by *transmission conditions* across interelement boundaries. In other words, for all $A \in \Omega_h$, we assume that we know the numerical trace of the potential $\hat{u}_h$ on ∂A, and we initially determine $(\boldsymbol{\sigma}_h, u_h)$ in terms of $(\hat{u}_h, f)$ by solving the local problem

$$
\begin{aligned}
(\boldsymbol{\kappa}\boldsymbol{\sigma}_h, \boldsymbol{v}_h)_A - (u_h, \nabla \cdot \boldsymbol{v}_h)_A + \langle \hat{u}_h, \boldsymbol{v}_h \cdot \boldsymbol{n}\rangle_{\partial A} &= 0, \\
-(\boldsymbol{\sigma}_h, \nabla q_h)_A + \langle \hat{\boldsymbol{\sigma}}_h \cdot \boldsymbol{n}, q_h\rangle_{\partial A} &= (f, q_h)_A,
\end{aligned}
\tag{12}
$$

for all $(q_h, \boldsymbol{v}_h) \in \mathcal{Q}(A) \times \boldsymbol{\mathcal{V}}(A)$. Here, the numerical trace $\hat{\boldsymbol{\sigma}}_h \cdot \boldsymbol{n}$ represents an approximation of $\boldsymbol{\sigma} \cdot \boldsymbol{n}$ over ∂A, and we assume that this trace is consistent and depends linearly on $\boldsymbol{\sigma}_h$, u_h and $\hat{u}_h$. Following [13], we then assume that the trace has the following simple form:

$$
\hat{\boldsymbol{\sigma}}_h \cdot \boldsymbol{n} := \boldsymbol{\sigma}_h \cdot \boldsymbol{n} + \tau(u_h - \hat{u}_h) \quad \text{on} \quad \partial A,
\tag{13}
$$

where τ is a linear local stabilization function that we describe below. In practice, its choice is quite delicate since it can strongly affect the stability and accuracy of the HDG method. Inserting (13) into (12), the weak form of the local problem in each element $A \in \Omega_h$ is found to be as follows: for any given $\hat{u}_h \in \Lambda(\partial A)$, find $(u_h, \boldsymbol{\sigma}_h) \in \mathcal{Q}(A) \times \boldsymbol{\mathcal{V}}(A)$ such that

$$
\begin{aligned}
(\boldsymbol{\kappa}\boldsymbol{\sigma}_h, \boldsymbol{v}_h)_A - (u_h, \nabla \cdot \boldsymbol{v}_h)_A + \langle \hat{u}_h, \boldsymbol{v}_h \cdot \boldsymbol{n}\rangle_{\partial A} &= 0, \\
(\nabla \cdot \boldsymbol{\sigma}_h, q_h)_A + \langle \tau(u_h - \hat{u}_h), q_h\rangle_{\partial A} &= (f, q_h)_A,
\end{aligned}
\tag{14}
$$

for all $(q_h, \boldsymbol{v}_h) \in \mathcal{Q}(A) \times \boldsymbol{\mathcal{V}}(A)$. Note that the local problem (14) can be solved in an element-by-element fashion to derive the variables $(u_h, \boldsymbol{\sigma}_h)$ solely in terms of $(\hat{u}_h, f)$. The global problem is then closed by imposing weakly the *transmission conditions* i.e., the single-valuedness of the normal component of the discrete variable $\hat{\boldsymbol{\sigma}}_h$, on each interior facet of the mesh that is as follows: find $\hat{u}_h \in \Lambda_h^g$ such that

$$
\langle \boldsymbol{\sigma}_h \cdot \boldsymbol{n} + \tau(u_h - \hat{u}_h), \mu_h\rangle_{\partial\Omega_h} = 0,
\tag{15}
$$

for all $\mu_h \in \Lambda_h^0$, where $\boldsymbol{\sigma}_h$ and u_h are solutions of the local problem (14). Substituting the solutions of (14) into (15), we obtain the global linear system that involves only the numerical trace $\hat{u}_h$. This step completes the presentation of the HDG methods. All hybrid methods referenced in the literature, such as hybridized Raviart–Thomas (H-RT) or Brezzi–Douglas–Marini (H-BDM) methods [2, 10], H-LDG methods [11, 9, 23] or, more recently, HHO methods [16, 15, 17], are established by properly choosing the stabilization function τ and subsequently the local spaces $\boldsymbol{\mathcal{V}}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\mathcal{F})$ in addition to the postprocessing step. Before describing some of these choices, let us briefly review the well-known technique of *static condensation* used to reduce the stiffness matrix of the HDG methods.

2.3. Static condensation

Note that the HDG method just described above consists of seeking an approximation $(u_h, \boldsymbol{\sigma}_h, \hat{u}_h) \in \mathcal{Q}_h \times \boldsymbol{\mathcal{V}}_h \times \Lambda_h^g$ satisfying the equations

$$
\begin{cases}
\boldsymbol{a}(\boldsymbol{\sigma}_h, \boldsymbol{v}_h) + \boldsymbol{b}(u_h, \boldsymbol{v}_h) - \boldsymbol{c}(\hat{u}_h, \boldsymbol{v}_h) &= 0, \\
\boldsymbol{b}(q_h, \boldsymbol{\sigma}_h) + \boldsymbol{d}_\tau(u_h, q_h) - \boldsymbol{r}_\tau(\hat{u}_h, q_h) &= (f, q_h)_{\Omega_h}, \\
\boldsymbol{c}(\mu_h, \boldsymbol{\sigma}_h) + \boldsymbol{r}_\tau(\mu_h, u_h) + \boldsymbol{s}_\tau(\hat{u}_h, \lambda_h) &= 0,
\end{cases}
\tag{16}
$$

for all $(q_h, \boldsymbol{v}_h, \mu_h) \in \mathcal{Q}_h \times \boldsymbol{\mathcal{V}}_h \times \Lambda_h^0$. Notably, the bilinear functionals used in (16) can be decomposed into two classes: those who are independent of the stabilization function τ such as

$$\boldsymbol{a}(\boldsymbol{\sigma}_h, \boldsymbol{v}_h) := -(\boldsymbol{\kappa}\boldsymbol{\sigma}_h, \boldsymbol{v}_h)_{\Omega_h},$$
$$\boldsymbol{b}(u_h, \boldsymbol{v}_h) := (u_h, \nabla \cdot \boldsymbol{v}_h)_{\Omega_h},$$
$$\boldsymbol{c}(\hat{u}_h, \boldsymbol{v}_h) := \langle \hat{u}_h, \boldsymbol{v}_h \cdot \boldsymbol{n} \rangle_{\partial\Omega_h},$$

and those depending on it:

$$\boldsymbol{d}_\tau(u_h, q_h) := \langle \tau(u_h), q_h \rangle_{\partial\Omega_h},$$
$$\boldsymbol{r}_\tau(\hat{u}_h, q_h) := \langle \tau(\hat{u}_h), q_h \rangle_{\partial\Omega_h},$$
$$\boldsymbol{s}_\tau(\mu_h, \hat{u}_h) := -\langle \tau(\hat{u}_h), \mu_h \rangle_{\partial\Omega_h}.$$

Hence, the corresponding algebraic system can be written as follows:

$$\begin{bmatrix} \mathcal{A} & \mathcal{B}^t & -\mathcal{C}^t \\ \mathcal{B} & \mathcal{D}_\tau & -\mathcal{R}_\tau^t \\ \mathcal{C} & \mathcal{R}_\tau & \mathcal{S}_\tau \end{bmatrix} \cdot \begin{bmatrix} \Sigma_h \\ \mathrm{U}_h \\ \hat{\mathrm{U}}_h \end{bmatrix} = \begin{bmatrix} \mathcal{G} \\ \mathcal{F} \\ \mathcal{H} \end{bmatrix}, \tag{17}$$

where Σ_h, U_h and $\hat{\mathrm{U}}_h$ are vectors of degrees of freedom associated with the discrete variables $\boldsymbol{\sigma}_h$, u_h and λ_h, respectively. To eliminate $(\boldsymbol{\sigma}_h, u_h)$ locally, we introduce the corresponding vector of interior degrees of freedom $\tilde{\Sigma}_h^t = [\Sigma_h^t, \mathrm{U}_h^t]$. Then, the linear system (17) can be rewritten in a compact form:

$$\begin{bmatrix} \tilde{\mathcal{A}} & -\tilde{\mathcal{B}}^t \\ \tilde{\mathcal{B}} & \mathcal{S}_\tau \end{bmatrix} \cdot \begin{bmatrix} \tilde{\Sigma}_h \\ \hat{\mathrm{U}}_h \end{bmatrix} = \begin{bmatrix} \tilde{\mathcal{F}} \\ \mathcal{H} \end{bmatrix}, \tag{18}$$

where $\tilde{\mathcal{A}}$, $\tilde{\mathcal{B}}$ and $\tilde{\mathcal{F}}$ are defined as follows:

$$\tilde{\mathcal{A}} = \begin{bmatrix} \mathcal{A} & \mathcal{B}^t \\ \mathcal{B} & \mathcal{D}_\tau \end{bmatrix}, \quad \tilde{\mathcal{B}}^t = \begin{bmatrix} \mathcal{C}^t \\ \mathcal{R}_\tau^t \end{bmatrix}, \quad \text{and} \quad \tilde{\mathcal{F}} = \begin{bmatrix} \mathcal{G} \\ \mathcal{F} \end{bmatrix}. \tag{19}$$

The associated matrix $\tilde{\mathcal{A}}$ has a block-diagonal structure due to the discontinuous nature of $\boldsymbol{\mathcal{V}}_h$ and $\mathcal{Q}_h$, and its inverse can be easily computed by using a Cholesky factorization. Thus, the elimination of $\tilde{\Sigma}_h$ follows immediately, and we obtain a single matrix equation only for the multipliers $\hat{\mathrm{U}}_h$,

$$\mathbb{A}\hat{\mathrm{U}}_h = \tilde{\mathcal{H}}, \tag{20}$$

where $\mathbb{A}$ is a symmetric positive definite matrix and $\tilde{\mathcal{H}}$, the corresponding right-hand side, is given by

$$\mathbb{A} = \mathcal{S}_\tau + \tilde{\mathcal{B}}\tilde{\mathcal{A}}^{-1}\tilde{\mathcal{B}}^t \quad \text{and} \quad \tilde{\mathcal{H}} = \mathcal{H} - \tilde{\mathcal{B}}\tilde{\mathcal{A}}^{-1}\tilde{\mathcal{F}}. \tag{21}$$

Once the solution $\hat{u}_h$ is obtained, the discrete interior variables $(\boldsymbol{\sigma}_h, u_h)$ can be computed by solving the local problem (14) on each element of the mesh. This step completes the reduction technique by *static condensation* of the HDG methods. It is clear that the choice of local spaces $\boldsymbol{\mathcal{V}}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\mathcal{F})$ has a direct impact on the dimension of the local solvers and of the global problem (20). For instance, the size of $\mathbb{A}$ is directly proportional to the dimension of $\Lambda(\mathcal{F})$. However, no assumptions have been made about local spaces nor on the stabilization function τ used to characterize the HDG method. This point is discussed in detail in the next section.

3. Stabilization functions and postprocessing techniques

3.1. Local stabilization functions

The choice of the stabilization function remains an open question. Several formulations have been proposed in the literature, some more intuitive [10, 11], and others more sophisticated [16, 17, 19], but all of them initially assume that τ is a linear, nonnegative function and that the local spaces $\mathcal{Q}(A)$ and $\boldsymbol{\mathcal{V}}(A)$ cannot be chosen arbitrarily and must satisfy inclusion constraints. A remarkable feature of these properties is that HDG methods can be well-defined completely independently of the choice of the local approximation space $\Lambda(\partial A)$, but in practice, it will be assumed to be sufficiently rich. As suggested by Cockburn in [13], *"the role of τ is to transform the discrepancy between u_h and $\hat{u}_h$ on ∂A into an energy"* to enhance the stability of HDG methods and to ensure that the local problem (14) and, hence, the global problem (16) are well-defined. We then focus on two well-known variants that we describe in detail below.

3.1.1. Null stabilization

In the simplest form of the *null–stabilization function*, we suppose that

$$\tau^{\mathtt{n}}(u_h - \hat{u}_h) := 0 \quad \text{on} \quad \partial\Omega_h. \tag{22}$$

Assuming (22), the bilinear functionals $\boldsymbol{d}_\tau$, $\boldsymbol{r}_\tau$ and $\boldsymbol{s}_\tau$ become nulls, and the HDG formulation (17) consists of seeking an approximation $(u_h, \boldsymbol{\sigma}_h, \hat{u}_h) \in \mathcal{Q}_h \times \boldsymbol{\mathcal{V}}_h \times \Lambda_h^g$ satisfying the equations

$$\begin{cases} (\boldsymbol{\kappa}\boldsymbol{\sigma}_h, \boldsymbol{v}_h)_{\Omega_h} - (u_h, \nabla \cdot \boldsymbol{v}_h)_{\Omega_h} + \langle \hat{u}_h, \boldsymbol{v}_h \cdot \boldsymbol{n} \rangle_{\partial\Omega_h} = 0, \\ (\nabla \cdot \boldsymbol{\sigma}_h, q_h)_{\Omega_h} = (f, q_h), \\ \langle \boldsymbol{\sigma}_h \cdot \boldsymbol{n}, \mu_h \rangle_{\partial\Omega_h} = 0, \end{cases} \tag{23}$$

for all $(q_h, \boldsymbol{v}_h, \mu_h) \in \mathcal{Q}_h \times \boldsymbol{\mathcal{V}}_h \times \Lambda_h^0$. This simplified formulation is typically considered in the so-called hybridized version of mixed finite element methods [1, 2]. In this case, the choice of appropriate finite dimensional spaces $\mathcal{Q}_h$, $\boldsymbol{\mathcal{V}}_h$ and Λ_h is not evident, but *stability conditions* provide many successful combinations of them such as the **RT** and **BDM** families [2, 3]. Hence, the local problems (14) as well as the global problem (23) are well defined without any stabilization procedure. The expected convergence rates of u_h and $\boldsymbol{\sigma}_h$ for the H-RT$_k$ and H-BDM$_k$ methods are summarized in Table 2.

3.1.2. Multiplicative stabilization

The *multiplicative–stabilization function* was suggested first by Cockburn *et al.* [11] to establish the original H-LDG$^{\mathtt{m}}$ method, namely,

$$\tau^{\mathtt{m}}(u_h - \hat{u}_h) := \tau_{\mathcal{F}} \cdot (u_h - \hat{u}_h) \quad \text{on} \quad \partial\Omega_h, \tag{24}$$

where $\tau_{\mathcal{F}}$ is a strictly positive constant. By imposing transmission conditions $[\![\hat{\boldsymbol{\sigma}}_h]\!] = 0$ on interior facets $\mathcal{F} \in \mathcal{E}_h^{\mathtt{i}}$, we obtain an explicit expression of the numerical traces $(\hat{u}_h, \hat{\boldsymbol{\sigma}}_h)$ in terms of $(u_h, \boldsymbol{\sigma}_h)$:

$$\begin{bmatrix} \hat{u}_h \\ \hat{\boldsymbol{\sigma}}_h \end{bmatrix} = \begin{bmatrix} \{u_h\}_{\boldsymbol{\omega}} \\ \{\boldsymbol{\sigma}_h\}_{\overline{\boldsymbol{\omega}}} \end{bmatrix} + \begin{bmatrix} 0 & \alpha_{\mathcal{F}} \\ \beta_{\mathcal{F}} & 0 \end{bmatrix} \cdot \begin{bmatrix} [\![u_h]\!] \\ [\![\boldsymbol{\sigma}_h]\!] \end{bmatrix}, \tag{25}$$

where $\boldsymbol{\omega}$, $\alpha_{\mathcal{F}}$ and $\beta_{\mathcal{F}}$ are positive constants defined as follows:

$$\boldsymbol{\omega} = \left(\frac{\tau_1}{\tau_1 + \tau_2}, \frac{\tau_2}{\tau_1 + \tau_2}\right), \quad \alpha_{\mathcal{F}} = \frac{1}{\tau_1 + \tau_2} \quad \text{and} \quad \beta_{\mathcal{F}} = \frac{\tau_1 \tau_2}{\tau_1 + \tau_2}. \tag{26}$$

Here, τ_1 and τ_2 represent the stabilization parameters associated with the common facet $\mathcal{F}$ adjacent to A_1 and A_2, respectively. Let us specify that the H-LDG$^{\mathbf{m}}$ method cannot be considered as a LDG method since the parameter $\alpha_{\mathcal{F}}$ is not null for any finite value of $\tau_{1,2}$. In this case, the local spaces $\mathcal{Q}(A)$, $\boldsymbol{V}(A)$ and $\Lambda(\partial A)$ must satisfy the following inclusion constraints:

$$\boldsymbol{\nabla}\mathcal{Q}(A) \subset \boldsymbol{V}(A), \quad \mathcal{Q}(A)|_{\partial A} \subset \Lambda(\partial A) \quad \text{and} \quad \boldsymbol{V}(A) \cdot \boldsymbol{n}|_{\partial A} \subset \Lambda(\partial A). \tag{27}$$

Several combinations of admissible local spaces have been studied in the literature [11], as summarized in Table 1. Previous reports have concluded that both discrete variables converge optimally $(k+1)$ in L^2 assuming that u_h and $\boldsymbol{\sigma}_h$ are approximated locally by polynomials of degree k (cf. Table 1, Variant 2) and that $\tau_{\mathcal{F}}$ is taken to be of order one (cf. Table 2, line 3).

H-LDG	$\boldsymbol{V}(A)$	$\mathcal{Q}(A)$	$\Lambda(\mathcal{F})$
Variant 1.	$\boldsymbol{\mathcal{P}}_{k-1}^d(A)$	$\mathcal{P}_k^d(A)$	$\mathcal{P}_k^{d-1}(\mathcal{F})$
Variant 2.	$\boldsymbol{\mathcal{P}}_k^d(A)$	$\mathcal{P}_k^d(A)$	$\mathcal{P}_k^{d-1}(\mathcal{F})$
Variant 3.	$\boldsymbol{\mathcal{P}}_k^d(A)$	$\mathcal{P}_{k-1}^d(A)$	$\mathcal{P}_k^{d-1}(\mathcal{F})$

Table 1: Admissible polynomial spaces $\boldsymbol{V}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\mathcal{F})$ for the H-LDG method with $k \geq 1$.

3.2. Postprocessing techniques

Until 1985, the *hybridization* was considered only as an implementation trick to overcome the saddle-point problem provided by MFE methods. However, Arnold and Brezzi established in [4] that the additional information $\hat{u}_h$ defined on the mesh skeleton can be exploited to construct locally a new approximated variable $u_h^{\star}$ converging faster than u_h in L^2. We end this section by showing how to postprocess both interior variables $(u_h, \boldsymbol{\sigma}_h)$ in conjunction with their numerical traces $(\hat{u}_h, \hat{\boldsymbol{\sigma}}_h)$ to obtain (i) a better approximation of the potential $u_h^{\star}$ that converges one order higher than u_h and (ii) a $\boldsymbol{H}(\mathrm{div})$-conforming variable $\boldsymbol{\sigma}_h^{\star}$ that ensures the continuity of its normal across interelement boundaries.

3.2.1. Postprocessing of the flux variable

We now show how to postprocess $\boldsymbol{\sigma}_h$ and $\hat{\boldsymbol{\sigma}}_h$ to obtain an optimal convergent approximation of $\boldsymbol{\sigma}$, denoted $\boldsymbol{\sigma}_h^{\star}$ and belonging to $\boldsymbol{H}(\mathrm{div}; \Omega)$. On each simplex $A \in \Omega_h$, we initially assume that the velocity $\boldsymbol{\sigma}_h \cdot \boldsymbol{n} \in \mathcal{P}_k^{d-1}(\partial A)$, and we define the variable $\boldsymbol{\sigma}_h^{\star}$ as the only element of the local Raviart–Thomas space $\mathbf{RT}_k(A) := \boldsymbol{\mathcal{P}}_k^d(A) \oplus \boldsymbol{x}\mathcal{P}_k^d(A)$ verifying that, for any $k \geq 0$, then

$$\langle(\boldsymbol{\sigma}_h^{\star} - \hat{\boldsymbol{\sigma}}_h) \cdot \boldsymbol{n}, \mu_h\rangle_{\partial A} = 0 \quad \forall \mu_h \in \mathcal{P}_k^{d-1}(\partial A), \tag{28a}$$

$$(\boldsymbol{\sigma}_h^{\star} - \boldsymbol{\sigma}_h, \boldsymbol{\xi}_h)_A = 0 \quad \forall \boldsymbol{\xi}_h \in \boldsymbol{\mathcal{P}}_{k-1}^d(A). \tag{28b}$$

We specify that (28a) and (28b) are simply an L^2-projection of the discrete variable $\boldsymbol{\sigma}_h$ and its trace $\hat{\boldsymbol{\sigma}}_h$ onto the interior and normal basis functions of $\mathbf{RT}_k(A)$, respectively. Note that for $k = 0$, the orthogonal projection (28b) is empty.

3.2.2. Postprocessing of the scalar variable

Here, we show how to obtain the postprocessed variable $u_h^\star$. To this aim, we assume that the potential is approximated by a polynomial of degree at most k in A, i.e., $u_h \in \mathcal{P}_k^d(A)$. For all $A \in \Omega_h$, find $u_h^\star \in \mathcal{P}_{k+1}^d(A)$ such that

$$(\boldsymbol{\nabla} u_h^\star, \boldsymbol{\nabla} q_h)_A = -(\boldsymbol{\kappa}\boldsymbol{\sigma}_h^\star, \boldsymbol{\nabla} q_h)_A, \tag{29a}$$

$$(u_h^\star, 1)_A = (u_h, 1)_A, \tag{29b}$$

for all $q_h \in \mathcal{P}_{k+1}^d(A)$. The *closure condition* (29b) is an additional solution constraint that fixes the mean value of $u_h^\star$ on A. The postprocessing technique used here relies on the superconvergence of the average value of the potential u_h and is a slight modification of the postprocessing introduced in [9], which consists of using the $\boldsymbol{H}(\text{div})$-conforming variable $\boldsymbol{\sigma}_h^\star$ instead of $\boldsymbol{\sigma}_h$.

4. Projective hybridizable Raviart–Thomas method

To our knowledge, only two well-established primal formulations referenced in the literature belong to the subclass of the projective HDG methods: first, the HDG method with reduced stabilization (H-LDG$^\mathfrak{p}$) proposed initially by Lehrenfeld in his thesis [19] (see, e.g., [13, 18, 24]), and second, the novel HHO method proposed by Di Pietro and Ern [17, 25]. The error analysis of the H-LDG$^\mathfrak{p}$ method was carried out by Oikawa in 2015 [18], who established the optimal convergence of both $\boldsymbol{\sigma}_h$ and u_h for general polyhedral elements: we have included in Table 2 (Line 4) the analogous mixed formulation of the primal method proposed by these authors. Recently, Cockburn *et al.* established bridges between the HHO method and the general framework of HDG methods [16] (see, e.g., [17, 15]): they identify the numerical traces of the primal HHO method in the general HDG mixed format and establish that both the local approximation spaces and the stabilization function suggested by Di Pietro and Ern are novel and distinctive choices that enrich the family of HDG methods. More precisely, the stabilization functions as used by both H-LDG$^\mathfrak{p}$ and HHO methods rely on specific L^2-projection operators, allowing for the consideration of polynomials of a higher degree for the approximation of the potential. Recently, Lederer *et al.* proposed in a series of original papers [26, 27] a new discretization method for incompressible flow problems based on an $\boldsymbol{H}(\text{div})$-conforming finite element space (BDM space) and an HDG method. The key idea is to relax the $\boldsymbol{H}(\text{div})$-conformity constraint so that only reduced unknowns are involved by imposing transmission conditions. Hence, the general HDG framework opens up novel perspectives by expanding the permissible polynomial combinations for approximations of discrete variables through a large choice of stabilization function. The projective hybridizable Raviart–Thomas (H-RT$^\mathfrak{p}$) formulation that we develop in this paper can be considered an inspired variation of the H-LDG$^\mathfrak{p}$ method described above and the traditional H-RT mixed method. Before describing our projective variant, let us review some definitions concerning the projective stabilization function introduced in [24].

4.1. The Lehrenfeld–Schöberl stabilization function

As mentioned above, Lehrenfeld and Schöberl observed that (i) the size of the stiffness matrix is directly proportional to the dimension of the local space $\Lambda(\mathcal{F})$ and that (ii) the quality of the discrete variables u_h and $\boldsymbol{\sigma}_h$ relies on polynomials used to build them. They

Method	Degree	$\boldsymbol{\mathcal{V}}(A)$	$\mathcal{Q}(A)$	τ	u_h	$\boldsymbol{\sigma}_h$	$u_h^\star$	$\boldsymbol{\sigma}_h^\star$
H-RT$_k$	$k\geq 0$	$\boldsymbol{\mathcal{P}}_k^d(A)\oplus\boldsymbol{x}\mathcal{P}_k^d(A)$	$\mathcal{P}_k^d(A)$	$\tau^{\mathtt{n}}$	$k+1$	$k+1$	$k+2$	–
H-BDM$_k$	$k\geq 2$	$\boldsymbol{\mathcal{P}}_k^d(A)$	$\mathcal{P}_{k-1}^d(A)$	$\tau^{\mathtt{n}}$	k	$k+1$	$k+2$	–
H-LDG$_k^{\mathtt{m}}$	$k\geq 1$	$\boldsymbol{\mathcal{P}}_k^d(A)$	$\mathcal{P}_k^d(A)$	$\tau^{\mathtt{m}}$	$k+1$	$k+1$	$k+2$	$k+1$
H-LDG$_k^{\mathtt{p}}$	$k\geq 0$	$\boldsymbol{\mathcal{P}}_k^d(A)$	$\mathcal{P}_{k+1}^d(A)$	$\tau^{\mathtt{p}}$	$k+2$	$k+1$	–	$k+1$
H-RT$_k^{\mathtt{p}}$	$k\geq 0$	$\boldsymbol{\mathcal{P}}_k^d(A)\oplus\boldsymbol{x}\mathcal{P}_k^d(A)$	$\mathcal{P}_{k+1}^d(A)$	$\tau^{\mathtt{p}}$	$k+2$	$k+1$	–	$k+1$

Table 2: Comparison of various hybrid mixed methods using the same local approximation space $\Lambda(\mathcal{F}) := \mathcal{P}_k^{d-1}(\mathcal{F})$. These methods are defined by their local approximation spaces $\mathcal{Q}(A)$ and $\boldsymbol{\mathcal{V}}(A)$ and the stabilization function $\tau^{\mathtt{n}}$, $\tau^{\mathtt{m}}$ or $\tau^{\mathtt{p}}$. The columns u_h, $\boldsymbol{\sigma}_h$, $u_h^\star$ and $\boldsymbol{\sigma}_h^\star$ indicate the expected convergence rate of $\|u-u_h\|_{L^2(\Omega_h)}$, $\|\boldsymbol{\sigma}-\boldsymbol{\sigma}_h\|_{L^2(\Omega_h)}$, $\|u-u_h^\star\|_{L^2(\Omega_h)}$ and $\|\boldsymbol{\sigma}-\boldsymbol{\sigma}_h^\star\|_{L^2(\Omega_h)}$, respectively.

improve the convergence of interior variables, without increasing the size of the final system (20), by simply modifying the multiplicative stabilization function $\tau^{\mathtt{m}}$ (24) as follows:

$$\tau^{\mathtt{p}}(u_h - \hat{u}_h) := \tau_{\mathcal{F}} \cdot (\pi_{\Lambda_h}(u_h) - \hat{u}_h) \quad \text{on} \quad \partial\Omega_h, \tag{30}$$

where π_{Λ_h} is the L^2-projection operator onto $\Lambda(\partial A)$. Following (30), the discrete variable u_h can now be approximated in a larger space $\mathcal{Q}(A)$. However, as suggested by the authors, the enrichment is limited and conditioned by the following inclusion constraints:

$$\boldsymbol{\nabla}\mathcal{Q}(A) \subset \boldsymbol{\mathcal{V}}(A), \quad \pi_{\Lambda_h}(\mathcal{Q}(A))|_{\partial A} \subset \Lambda(\partial A) \quad \text{and} \quad \boldsymbol{\mathcal{V}}(A)\cdot\boldsymbol{n}|_{\partial A} \subset \Lambda(\partial A). \tag{31}$$

Assuming that $\Lambda(\partial A)$ and $\boldsymbol{\mathcal{V}}(A)$ are the set of polynomials of degree k, the authors conclude that $\mathcal{Q}(A)$ can be composed at most of polynomial of degree $k+1$. Then, assuming that the stabilization parameter $\tau_{\mathcal{F}}$ in (30) is taken to be of order $1/h$, they observe that the discrete variable u_h converges at the order $k+2$ in the L^2-norm without any postprocessing [18, 19]. There is a price to pay in return for using the projective *Lehrenfeld–Schöberl* stabilization function. Indeed, by increasing the polynomial degree of the scalar variable u_h (just one degree higher), the *static condensation* clearly becomes heavier due to the larger number of degrees of freedom per element (cf., e.g. Table 3). However, this trick avoids any postprocessing of the scalar variable u_h (29).

4.2. Local approximation spaces

A characteristic feature of the standard H-MFE methods is that their discrete solutions coincide with those of the original $\boldsymbol{H}(\text{div})$-conforming MFE methods [10, 11]. This equivalence rests on two key ingredients: (i) the choice of the null-stabilization function $\tau^{\mathtt{n}}$ which amounts to forcing the flux residual $R_{\partial A}^{\sigma} := (\hat{\boldsymbol{\sigma}}_h - \boldsymbol{\sigma}_h) \cdot \boldsymbol{n}$ to be zero and (ii) the choice of the local approximation spaces as indicated in Table 2. The proposed H-RT$^{\mathtt{p}}$ method is obtained by modifying each of these two points. First, we relax the residual constraint (i.e., $R_{\partial A}^{\sigma} \neq 0$) by replacing $\tau^{\mathtt{n}}$ by the *Lehrenfeld–Schöberl* stabilization function $\tau^{\mathtt{p}}$ (30); second, we adapt the polynomial combination of the discrete variables to respect the inclusion constraints (31). Consequently, the H-RT$^{\mathtt{p}}$ method differs fundamentally from the standard H-RT method as it introduces additional stabilization terms and uses a different polynomial combination set for approximating discrete variables. To complete our presentation, we select (i) the smaller admissible space $\Lambda(\partial A)$ to minimize the size of the stiffness matrix $\mathbb{A}$ in (20) and (ii) the larger space $\mathcal{Q}(A)$ to optimize the expected convergence rate of u_h. We then define the projective

H-RT$_k^\flat$ method by choosing the numerical trace of flux and local polynomial spaces as follows:

$$\hat{\boldsymbol{\sigma}}_h \cdot \boldsymbol{n} := \boldsymbol{\sigma}_h \cdot \boldsymbol{n} + \tau_{\mathcal{F}} \cdot (\pi_{\Lambda_h}(u_h) - \hat{u}_h) \quad \text{on} \quad \mathcal{F} \in \partial A, \tag{32}$$

$$\boldsymbol{\mathcal{V}}(A) := \mathbf{RT}_k(A), \quad \mathcal{Q}(A) := \mathcal{P}_{k+1}^d(A) \quad \text{and,} \quad \Lambda(\partial A) := \mathcal{P}_k^{d-1}(\partial A), \tag{33}$$

for all $A \in \Omega_h$. It can be easily checked that the chosen local spaces (33) with respect to the normal trace definition (32) verify all inclusion constraints (31). We have illustrated in Table 3, the various local approximation spaces, i.e., $\boldsymbol{\mathcal{V}}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\partial A)$, used to construct the H-RT$_k^\flat$ method by specifying the location of their corresponding degrees of freedom. Thus, the resulting method can be considered as a *penalized* variant of the original H-RT method where the postprocess of the potential is now disabled and the flux reconstruction is simplified, as we will see below.

4.3. Simplified postprocess of the flux

The key idea of this simplified postprocessing procedure relies on properties of local basis functions $\boldsymbol{v}_h$ that span the Raviart-Thomas element $\mathbf{RT}_k(A)$. Let us remind that a unisolvent finite element $\mathbf{RT}_k(A)$ can be derived by fixing (i) the moments of the normal component up to order k on facets and (ii) the interior moments up to order $k-1$ [2]. As a result, the set of basis functions can be divided into two categories: first, the subset of *normal basis functions*, denoted $\boldsymbol{v}_h^\partial$, having nonvanishing normal traces on ∂A which are L^2-orthogonal to any polynomial $\boldsymbol{\xi}_h \in \mathcal{P}_{k-1}^d(A)$ and, second, the subset of *interior basis functions*, denoted $\boldsymbol{v}_h^i$, having vanishing normal traces on the boundary ∂A. Thanks to these last two properties, we will be able to guarantee the existence and uniqueness of the simplistic flux reconstruction. As explained in the previous section, the postprocess procedure (28) used to construct $\boldsymbol{\sigma}_h^\star$ relies on the L^2-orthogonal projection of the discrete variable $\boldsymbol{\sigma}_h$ and its trace $\hat{\boldsymbol{\sigma}}_h$ onto basis functions that span $\mathbf{RT}_k(A)$. Hence, assuming that both $\boldsymbol{\sigma}_h$ and $\boldsymbol{\sigma}_h^\star$ belong to the same local space $\mathbf{RT}_k(A)$ and using the L^2-orthogonality property of functions $\boldsymbol{v}_h^\partial$ and $\boldsymbol{\xi}_h$, the second equation (28b) reduces to,

$$(\boldsymbol{\sigma}_h^{\star,i}, \boldsymbol{\xi}_h)_A = (\boldsymbol{\sigma}_h^i, \boldsymbol{\xi}_h)_A \quad \forall \boldsymbol{\xi}_h \in \mathcal{P}_{k-1}^d(A), \tag{34}$$

where $\boldsymbol{\sigma}_h^{\star,i}$ and $\boldsymbol{\sigma}_h^i$ denote the restriction of $\boldsymbol{\sigma}_h^\star$ and $\boldsymbol{\sigma}_h$ to the *interior basis functions*, respectively. Thus, the projection operation (34) consists of substituting the interior nodal values of $\boldsymbol{\sigma}_h^\star$ by those of $\boldsymbol{\sigma}_h$. In other words, if we denote by $\Sigma_A^\star$ (resp. Σ_A) the local vector of degrees of freedom of $\boldsymbol{\sigma}_h^\star$ (resp. $\boldsymbol{\sigma}_h$), we then deduce easily from (34) that

$$\Sigma_A^\star = \begin{pmatrix} \Sigma_A^{\star,\partial} \\ \Sigma_A^i \end{pmatrix}, \tag{35}$$

where $\Sigma_A^{\star,\partial}$ and Σ_A^i correspond to the restriction of $\Sigma_A^\star$ and Σ_A to the *normal* and *interior basis functions* of $\mathbf{RT}_k(A)$, respectively. Similarly, by using that interior basis functions $\boldsymbol{v}_h^i$ have no normal trace on ∂A, the first equation (28a) reduces to the following equation,

$$\langle \boldsymbol{\sigma}_h^{\star,\partial} \cdot \boldsymbol{n}, \mu_h \rangle_{\partial A} = \langle \boldsymbol{\sigma}_h^\partial \cdot \boldsymbol{n}, \mu_h \rangle_{\partial A} + \langle \tau_{\mathcal{F}} \cdot (\pi_{\Lambda_h}(u_h) - \hat{u}_h), \mu_h \rangle_{\partial A} \quad \forall \mu_h \in \mathcal{P}_k^{d-1}(\partial A). \tag{36}$$

The corresponding matrix equation (36) is smaller than (28) since its dimension is equal here to the number of normal basis functions per element (cf., e.g., Tables 3 and 4). Note, however, that this simplified flux reconstruction requires computing beforehand the projection of the potential $\pi_{\Lambda_h}(u_h)$ onto $\Lambda(\partial A)$. This step can easily be done by means of the quadrature formula in the two-dimensional case [18, 24].
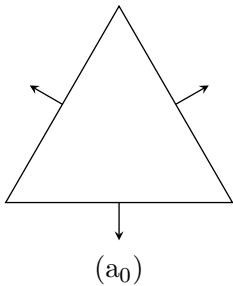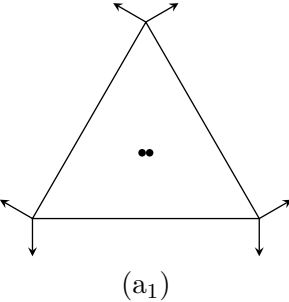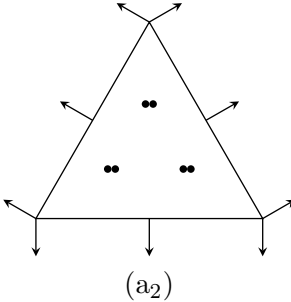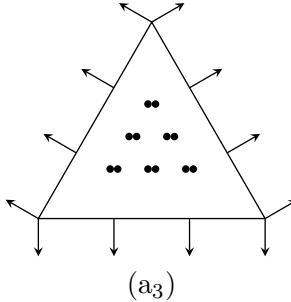
11
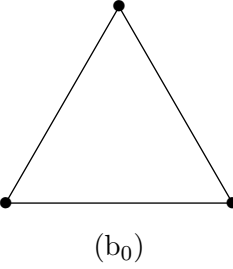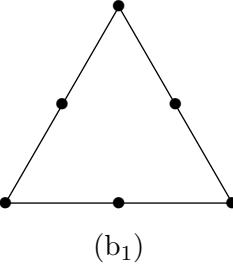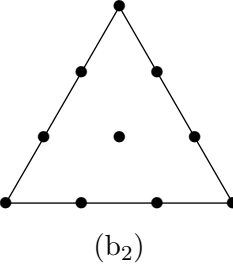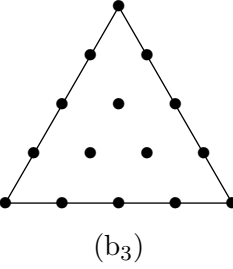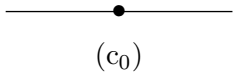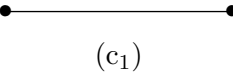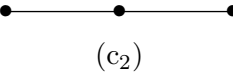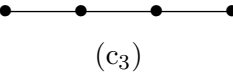
H-RT$_k^{\mathfrak{p}}$ method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
$\boldsymbol{\mathcal{V}}(A) := \mathbf{RT}_k(A)$	(a$_0$)	(a$_1$)	(a$_2$)	(a$_3$)
$\mathcal{Q}(A) := \mathcal{P}_{k+1}^d(A)$	(b$_0$)	(b$_1$)	(b$_2$)	(b$_3$)
$\Lambda(\mathcal{F}) := \mathcal{P}_k^{d-1}(\mathcal{F})$	(c$_0$)	(c$_1$)	(c$_2$)	(c$_3$)

Table 3: Localization of degrees of freedom of the local approximation spaces $\boldsymbol{\mathcal{V}}(A)$, $\mathcal{Q}(A)$ and $\Lambda(\mathcal{F})$ used by the H-RT$_k^{\mathfrak{p}}$ method for different polynomial order $k = 0, \ldots, 3$. Here, A is a triangle ($d = 2$), and $\mathcal{F}$ is an edge of ∂A. For the local space $\boldsymbol{\mathcal{V}}(A) = \mathbf{RT}_k(A)$, the arrows indicate the value of the normal component ($3(k+1)$-normal basis functions), and the double dots indicate the value of both components ($k(k+1)$-interior basis functions).

Method	dim $\boldsymbol{\mathcal{V}}(A)$	dim $\mathcal{Q}(A)$	$u_h^\star$	$\boldsymbol{\sigma}_h^\star$
H-RT$_k$	$(k+1)(k+3)$	$(k+1)(k+2)/2$	$(k+2)(k+3)/2$	$-$
H-LDG$_k^{\mathrm{m}}$	$(k+1)(k+2)$	$(k+1)(k+2)/2$	$(k+2)(k+3)/2$	$(k+1)(k+3)$
H-LDG$_k^{\mathrm{p}}$	$(k+1)(k+2)$	$(k+2)(k+3)/2$	$-$	$(k+1)(k+3)$
H-RT$_k^{\mathrm{p}}$	$(k+1)(k+3)$	$(k+2)(k+3)/2$	$-$	$3k+3$

Table 4: The first and second columns indicate the dimension of local spaces $\boldsymbol{\mathcal{V}}(A)$ and $\mathcal{Q}(A)$, respectively. The dimension of the local solver is equal to the sum of these two quantities. The third and fourth columns indicate the size of the local matrix to compute the postprocessed variables $u_h^\star$ and $\boldsymbol{\sigma}_h^\star$, respectively.

5. Numerical experiments

The performance of the novel H-RT$_k^{\mathrm{p}}$ method is now investigated for the following three numerical examples in two-dimensional space in terms of convergence, accuracy, stability and efficiency. In particular, we compare the projective H-RT$_k^{\mathrm{p}}$ variant with the standard H-RT$_k$ method without any stabilization and the well-established H-LDG$_k^{\mathrm{m}}$ method using a multiplicative stabilization function. For our experiments, we focused on three hybrid mixed methods, each using distinctive stabilization functions. These methods are all implemented in a single Fortran 2003 program, and a direct solver using an unsymmetric-pattern multifrontal method and direct sparse LU factorization (UMFPACK) is considered for solving the associated linear system [28]. In the rest of this paper, we focus only on the *best* approximated variables, i.e., $(u_h, \boldsymbol{\sigma}_h^\star)$ for the H-RT$_k^{\mathrm{p}}$ method, $(u_h^\star, \boldsymbol{\sigma}_h)$ for the H-RT$_k$ method and $(u_h^\star, \boldsymbol{\sigma}_h^\star)$ for the H-LDG$_k^{\mathrm{m}}$ method, and on the CPU time required to obtain them: let us presice that the CPU time does not take into account the time required to read mesh files and write output files. The numerical errors and corresponding estimated convergence rates (ECRs) are computed in the usual L^2-norm. In particular, we denote by e_{u_h} and $e_{\boldsymbol{\sigma}_h}$ (resp. $e_{u_h^\star}$ and $e_{\boldsymbol{\sigma}_h^\star}$), the L^2-error of discrete variables u_h and $\boldsymbol{\sigma}_h$ (resp. postprocessed variables $u_h^\star$ and $\boldsymbol{\sigma}_h^\star$),

$$e_{u_h} = \|u - u_h\|_{L^2(\Omega_h)} \quad \text{and} \quad e_{\boldsymbol{\sigma}_h} = \|\boldsymbol{\sigma} - \boldsymbol{\sigma}_h\|_{L^2(\Omega_h)}, \tag{37}$$

and similarly for $e_{u_h^\star}$ and $e_{\boldsymbol{\sigma}_h^\star}$, respectively. We consider two types of triangular meshes (structured/unstructured) in these experiments as illustrated in Figure 1, and standard h- and p-refinement strategies are used to compute the discrete solutions and numerical errors over the entire domain. For instance, regular meshes are obtained by discretizing the unit domain Ω with uniform squares of length $h = 1/n$ for $n = \{4, 8, 16, 32, 64\}$, which are then divided into two triangles as depicted in Figure 1.a. The mesh refinement is then characterized by n. However, let us precise here that for a given mesh and for any fixed value of k, these three variants have a global matrix equation (20) of identical size and sparsity structure (cf. Table 5). It should also be noted that the computational effort to calculate postprocessing variables is generally negligible thanks to parallelization strategies, but in our case, all these calculations are done sequentially. The source term and Dirichlet boundary conditions are taken such that the proposed analytical solution is exact.

5.1. Test A – Homogeneous isotropic flow

The first test case considered here is used to evaluate the performance of hybrid mixed methods in the simplest possible setting, i.e., *Poisson's equation*. We then consider the homogeneous Dirichlet boundary value problem corresponding to the exact solution $u(x,y) = \sin(2\pi x)\sin(2\pi y)$ in the unit domain $\Omega = (0,1)^2$. Here, the material is assumed to be the
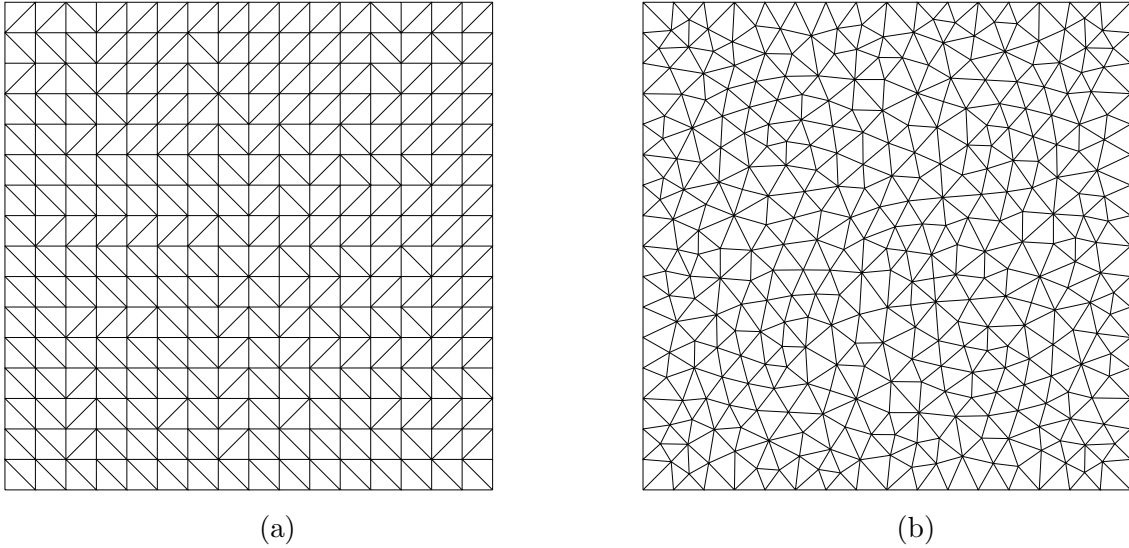
Figure 1: Regular mesh (a) and irregular mesh (b) for $n = 16$.

$k \backslash n$	4	8	16	32	64
0	40	176	736	3008	12160
1	80	352	1472	6016	24320
2	120	528	2208	9024	36480
3	160	704	2944	12032	48640

Table 5: Total number of interior Lagrange multipliers, i.e., dimension of the stiffness matrix $\mathbb{A}$, as a function of the mesh refinement n and the polynomial degree k of the hybrid method on regular meshes.

homogeneous and isotropic $\boldsymbol{\kappa} = \boldsymbol{I}_2$, where $\boldsymbol{I}_d$ denotes the $d \times d$ identity matrix. A history of convergence of the H-RT$_k^{\flat}$, H-RT$_k$ and H-LDG$_k^{\mathrm{m}}$ methods is summarized in Table 6 for regular meshes ($n = 4, \ldots, 64$) and different polynomial degrees ($k = 0, 1, 2$). We emphasize here that we do not report in the Table 6 the history of convergence of selected discrete variables; however, we analyze below their behavior. For all degrees of approximation k, we note that the discrete variables u_h and $\boldsymbol{\sigma}_h$ converge both optimally with order $k + 1$ for the H-RT$_k$ method, whereas the postprocessed variable $u_h^{\star}$ converges with order $k + 2$ in the L^2-norm. Similar results are obtained for the H-LDG$_k^{\mathrm{m}}$ method except for the lowest degree ($k = 0$), where $u_h^{\star}$ converges only with order $k + 1$, as is the case for u_h. These results are in accordance with the theoretical estimates of convergence rates given by Brezzi *et al.* in [2] for mixed methods and by Cockburn *et al.* in [11] for HDG methods. Concerning the H-RT$_k^{\flat}$ method and for all degrees of approximation k, we observe that the scalar variable u_h converges (without any postprocessing) with order $k + 2$, including the particular case $k = 0$, and that both $\boldsymbol{\sigma}_h$ and $\boldsymbol{\sigma}_h^{\star}$ converge with optimal order $k + 1$. The substantial gain in accuracy induced by the postprocessing technique is clearly illustrated here; however, the discrete variable u_h of the H-RT$_k^{\flat}$ method has a slightly smaller error than the postprocessed scalar variable $u_h^{\star}$ of the H-RT$_k$ and H-LDG$_k^{\mathrm{m}}$ methods (cf. Table 6). To complete our analysis, we also include comparisons in terms of the CPU time, which corresponds here to the simulation time needed to obtain the discrete variables referenced in Table 6 for each method. Despite the fact that all hybrid methods are built according to a unified principle, each of these three variants uses different

14

		H-RT$_k^{\mathfrak{p}}$				H-RT$_k$				H-LDG$_k^{\mathfrak{m}}$			
k	n	e_{u_h}	ECR	$e_{\sigma_h^\star}$	ECR	$e_{u_h^\star}$	ECR	e_{σ_h}	ECR	$e_{u_h^\star}$	ECR	$e_{\sigma_h^\star}$	ECR
0	4	2.10E−1	−	1.99E−0	−	1.27E−1	−	1.96E−0	−	1.27E−1	−	1.97E−0	−
	8	4.41E−2	2.25	1.01E−0	0.98	3.11E−2	2.03	1.01E−0	0.96	6.63E−1	0.94	1.01E−0	0.96
	16	9.83E−3	2.16	5.06E−1	1.00	7.79E−3	2.00	5.05E−1	1.00	3.41E−1	0.96	5.06E−1	1.00
	32	2.38E−3	2.04	2.52E−1	1.00	1.95E−3	2.00	2.52E−1	1.00	1.73E−1	0.98	2.52E−1	1.01
	64	5.95E−4	2.00	1.26E−1	1.00	4.90E−4	2.00	1.26E−1	1.00	8.72E−2	0.99	1.26E−1	1.00
1	4	3.64E−2	−	4.31E−1	−	2.27E−2	−	4.13E−1	−	2.67E−2	−	4.24E−1	−
	8	3.09E−3	3.56	1.14E−1	1.92	3.15E−3	2.85	1.13E−1	1.87	3.38E−3	2.98	1.16E−1	1.87
	16	3.38E−4	3.19	2.87E−2	1.99	3.99E−4	2.98	2.86E−2	1.98	4.21E−4	3.00	2.92E−2	1.99
	32	3.96E−5	3.09	7.04E−3	2.01	4.86E−5	3.04	7.02E−3	2.02	5.15E−5	3.03	7.17E−3	2.03
	64	4.80E−6	3.04	1.74E−3	2.00	5.98E−6	3.02	1.74E−3	2.01	6.33E−6	3.02	1.77E−3	2.02
2	4	4.59E−3	−	8.65E−2	−	5.21E−3	−	8.58E−2	−	5.26E−3	−	8.75E−2	−
	8	2.35E−4	4.29	9.62E−3	3.17	2.80E−4	4.22	9.54E−3	3.17	2.80E−4	4.23	9.83E−3	3.15
	16	1.29E−5	4.18	1.18E−3	3.03	1.76E−5	4.00	1.17E−3	3.03	1.76E−5	3.99	1.21E−3	3.02
	32	7.99E−7	4.01	1.51E−4	2.97	1.14E−6	3.95	1.51E−4	2.96	1.15E−6	3.94	1.55E−4	2.96
	64	5.08E−8	3.98	1.92E−5	2.98	7.33E−8	3.96	1.92E−5	2.98	7.36E−8	3.97	2.01E−5	2.95

Table 6: Test A – Homogeneous and isotropic test case on regular meshes

local assemblies and local techniques to reconstruct the discrete variables as indicated in Table 4. To achieve this aim, we plot in Figure 2 the errors in u and in $\boldsymbol{\sigma}$ as a function of the CPU time. For a wide range of degree of approximation k, we observe that both the H-RT$_k^{\mathfrak{p}}$ and H-LDG$_k^{\mathfrak{m}}$ methods yield very similar results with a slight advantage for the projective variant and that they outperform the standard H-RT$_k$ method, which clearly requires more CPU time (see., for instance, Figures 2-a$_{1,2}$ and 2-b$_{0,1,2}$). The situation is somewhat different for the lowest degree ($k = 0$), and in this latter case only, we observe that the numerical approximation of the scalar variable of both the H-RT$_0^{\mathfrak{p}}$ and H-RT$_0$ methods outperforms that of the H-LDG$_0^{\mathfrak{m}}$ method; this result arises mainly because the H-LDG$_0^{\mathfrak{m}}$ method converges only with order one (cf. Figure 2-a$_0$). This experiment also illustrates the benefits of using higher-degree approximations. For instance, we can achieve better accuracy with the H-RT$_k^{\mathfrak{p}}$ method (with approximately half the CPU time) with a square approximation ($k = 2$) in a coarser mesh ($n = 8$) than with a constant approximation ($k = 0$) in the finest mesh ($n = 64$).

5.2. Test B – Heterogeneous anisotropic flow

The second test case is used to analyze the behavior of hybrid methods in the context of heterogeneity and anisotropy of the material. To this aim, we consider the Dirichlet problem corresponding to the exact solution $u(x, y) = \sin(\pi x)\cos(\pi y)$ on the domain $\Omega \in (0, 1)^2$. Here, we assume that the diffusion tensor is given by

$$\boldsymbol{\kappa}^{-1} = \begin{bmatrix} e^{x+y} & 0 \\ 0 & e^{x-y} \end{bmatrix}, \tag{38}$$

which corresponds to a heterogeneous and *smooth* anisotropic medium, where the magnitude of diffusion coefficients along each principal direction x and y, respectively, vary continuously at each point of the domain; here, the strongest anisotropic ratio is approximately e^2. This test case is inspired from [29]. A history of convergence is presented in Table 7 for a wide range of polynomial degree $0 \leq k \leq 2$ and for irregular meshes ($n = 4, \ldots, 64$).

All conclusions established in the previous experiment (Test A – Homogeneous isotropic flow) are also observed here despite the presence of heterogeneity and anisotropy.
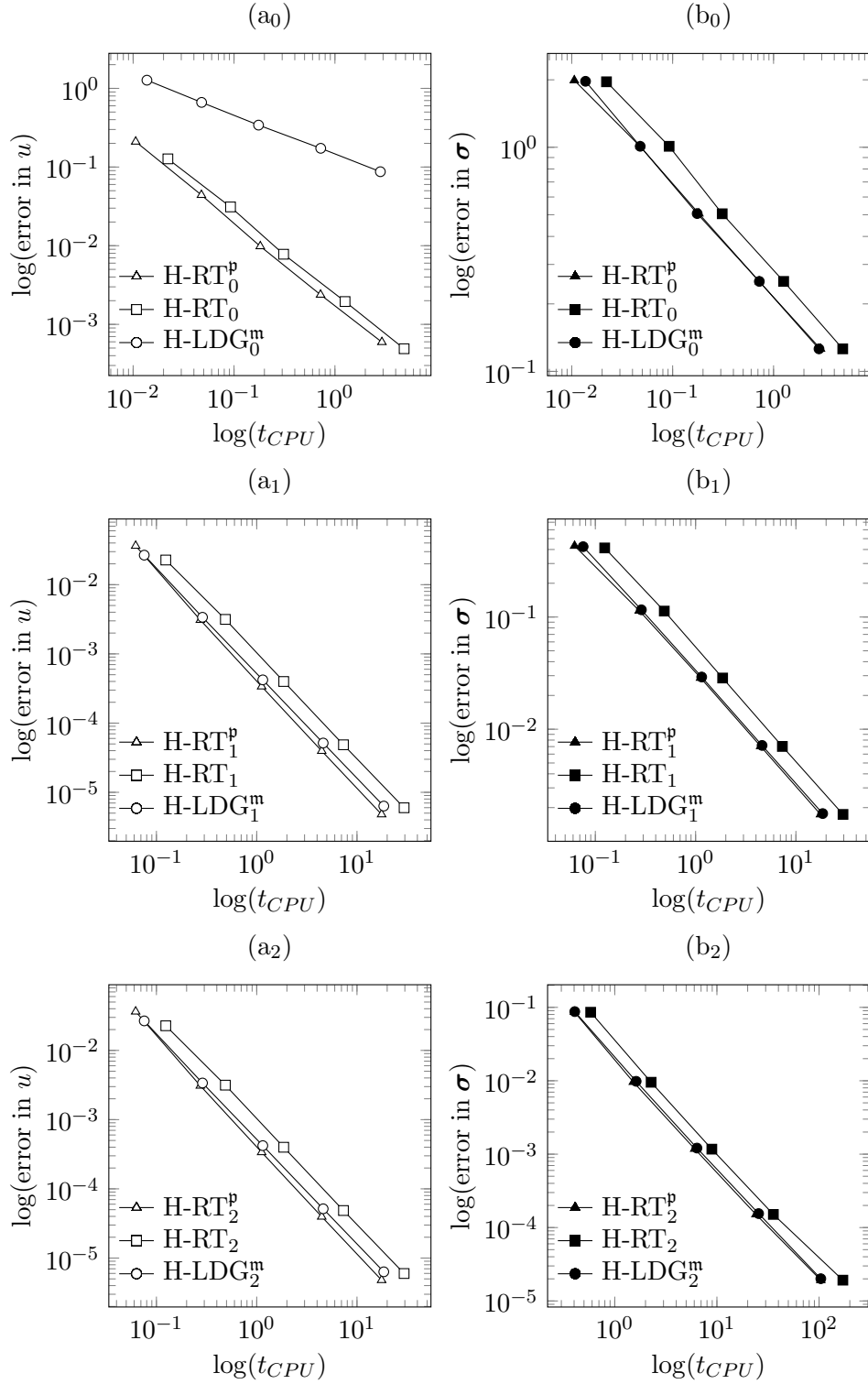
Figure 2: Test A – History of convergence of the discrete variables for the H-RT$_k^{\mathfrak{p}}$ ($\triangle$: e_{u_h} and $\blacktriangle$: $e_{\boldsymbol{\sigma}_h^\star}$), the H-RT$_k$ ($\square$: $e_{u_h^\star}$ and $\blacksquare$: $e_{\boldsymbol{\sigma}_h}$) and the H-LDG$_k^{\mathfrak{m}}$ ($\circ$: $e_{u_h^\star}$ and $\bullet$: $e_{\boldsymbol{\sigma}_h^\star}$) methods as a function of the CPU time t_{CPU} for different polynomial degrees $0 \leq k \leq 2$ on regular meshes.

		H-RT$_k^{\mathfrak{p}}$				H-RT$_k$				H-LDG$_k^{\mathfrak{m}}$			
k	n	e_{u_h}	ECR	$e_{\sigma_h^\star}$	ECR	$e_{u_h^\star}$	ECR	e_{σ_h}	ECR	$e_{u_h^\star}$	ECR	$e_{\sigma_h^\star}$	ECR
0	4	6.77E−2	−	1.80E−0	−	4.55E−2	−	1.77E−0	−	3.36E−1	−	1.85E−0	−
	8	1.54E−2	2.14	8.99E−1	1.00	1.41E−2	1.69	8.95E−1	0.98	1.72E−1	0.97	9.10E−1	1.02
	16	3.95E−3	1.96	4.46E−1	1.01	3.83E−3	1.88	4.45E−1	1.00	8.67E−2	0.99	4.49E−1	1.02
	32	1.00E−3	1.98	2.21E−1	1.01	9.82E−4	1.96	2.21E−1	1.00	4.36E−2	0.99	2.23E−1	1.01
	64	2.51E−4	2.00	1.10E−1	1.00	2.48E−4	1.99	1.10E−1	1.00	2.19E−2	0.99	1.10E−1	1.01
1	4	6.01E−3	−	2.27E−1	−	5.54E−3	−	2.25E−1	−	5.08E−3	−	2.12E−1	−
	8	7.34E−4	3.03	5.83E−2	1.96	7.82E−4	2.82	5.82E−2	1.95	6.60E−4	2.94	5.67E−2	1.90
	16	8.90E−5	3.04	1.46E−2	2.00	9.75E−5	3.00	1.46E−2	2.00	8.24E−5	3.00	1.43E−2	1.99
	32	1.09E−5	3.03	3.62E−3	2.01	1.21E−5	2.96	3.62E−3	2.01	1.03E−5	3.00	3.62E−3	1.98
	64	1.37E−6	3.00	9.10E−4	2.00	1.52E−6	3.04	9.11E−4	1.99	1.30E−6	2.99	9.07E−4	2.00
2	4	4.60E−4	−	2.31E−2	−	3.65E−4	−	2.30E−2	−	4.28E−4	−	2.12E−2	−
	8	2.27E−5	4.34	2.90E−3	2.99	2.32E−5	3.98	2.90E−3	2.99	2.86E−5	3.90	3.19E−3	3.02
	16	1.38E−6	4.04	3.58E−4	3.02	1.52E−6	3.94	3.58E−4	3.02	1.77E−6	4.01	3.95E−4	3.01
	32	8.72E−8	3.99	4.60E−5	2.96	9.72E−8	3.97	4.61E−5	2.96	1.11E−7	3.99	4.94E−5	3.00
	64	5.48E−9	4.00	5.74E−6	3.00	6.15E−9	3.99	5.75E−6	3.00	6.94E−9	4.00	6.17E−6	3.00

Table 7: Test B – Heterogeneous and smooth anisotropic test case on regular meshes

5.3. Test C – Homogeneous anisotropic flow

The last experiment was mentioned in [30] as causing some numerical locking problems for a large class of numerical methods. Then, the exact solution is taken to be

$$u(x,y) = \sin(2\pi x)e^{-\frac{2\pi}{\sqrt{\delta}}y},$$

and the permeability tensor is assumed to be equal to

$$\boldsymbol{\kappa}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & \delta \end{bmatrix}. \tag{39}$$

Here, the *locking parameter* δ is assumed to be the sufficiently large $\delta = 10^6$, and the stabilization parameter is chosen to be of order one, that is, $\tau = \mathcal{O}(1)$. In the last example, we wish to illustrate the benefits of using a stabilization function. Thus, we focus only on the H-RT$_k^{\mathfrak{p}}$ method using the *Lehrenfeld–Schöberl* stabilization function and the standard H-RT$_k$ method without any stabilization. As indicated above, for any fixed values of k and n, we observe that the projective H-RT$_k^{\mathfrak{p}}$ variant requires less CPU time than the standard H-RT$_k$ method in this latter example as well. A history of convergence is plotted in Figure 3 for different polynomial degrees $(k = 0, \ldots, 3)$ on irregular meshes $(n = 8, \ldots, 64)$. For the lowest degree $(k = 0)$, we observe that both the H-RT$_0^{\mathfrak{p}}$ and H-RT$_0$ methods yield similar results and that the use of the stabilization function provides no significant improvements (cf. Figures 3-a$_0$ and 3-b$_0$): the velocity error does not converge, and the estimated convergence rate (ECR) of the potential decreases significantly as the mesh is refined. We clearly observe a *numerical locking phenomena* for both the H-RT$_0^{\mathfrak{p}}$ and H-RT$_0$ methods. The situation is somewhat different for higher-degree approximations $(k \geq 1)$, for which the gap is now more significant. As shown in Figure 3, the numerical errors computed by the H-RT$_k^{\mathfrak{p}}$ method differ significantly and are more accurate by several orders of magnitude than those given by the H-RT$_k$ method (cf., e.g., Figures 3-a$_{1,2}$ and 3-b$_{1,2}$). In particular (for $k = 3$), we clearly observe *problematic numerical locking* for the H-RT$_3$ method only (cf. Figures 3-a$_3$ and 3-b$_3$). Note that this is not the case for the H-RT$_3^{\mathfrak{p}}$ method and that the computed errors decrease as expected, i.e., $k+2$ for the potential and $k+1$ for the flux. Therefore, we conclude that the projective H-RT$_k^{\mathfrak{p}}$ variant outperforms the standard H-RT$_k$ method, particularly in highly anisotropic porous media. In the latter case, we highlight the beneficial role of using a stabilization function to restore the convergence of the method in severe cases. Let us point out that any other HDG methods using a non-null stabilization function will also be able to treat this rough case.

6. Conclusion & Perspectives

In summary, we have presented a projective hybridizable Raviart–Thomas method for second-order diffusion problems. The proposed method is inspired by the HDG formalism as it introduces residual flux terms into the original H-RT method. Specifically, we add a projective-type stabilization function in the definition of the normal trace of the flux on the mesh skeleton allowing to consider polynomials of higher degree for the approximation of the potential. Extensive numerical experiments have been performed to prove the accuracy, efficiency and robustness of the H-RT$^{\mathfrak{p}}$ method. To this aim, comparisons with the standard H-RT method and the well-known H-LDG method are exposed in terms of mesh and polynomial refinements, and CPU time. For any fixed-value of k (including the particular case $k = 0$), the H-RT$^{\mathfrak{p}}_k$ method achieves an optimal convergence order for (i) the approximate flux, $k+1$, after a straighforward reconstruction step and (ii) the potential, $k + 2$, without any postprocess. These comparisons also indicate that the H-RT$^{\mathfrak{p}}$ method requires less CPU time and is more accurate than the standard H-RT method, particularly in highly anisotropic porous media.

The key idea explored in this paper is that H-MFE methods are also amenable to stabilization strategies within the general framework of HDG methods. From our viewpoint, this is a promising direction to explore, as the resulting method inherits all benefits of the original methods. Notably, the resulting method developed in this paper for second-order diffusion problems is not limited to only the Raviart–Thomas finite element and the projective *Lehrenfeld–Schöberl* stabilization function but can be easily extended to any other class of problems, (broken) $\boldsymbol{H}$(div)-conforming finite elements and stabilization functions. Accordingly, an extension to more general advection-diffusion-reaction problems (with nonnegative characteristic form) will be considered in a forthcoming paper.

Acknowledgments

References

[1] G. Chavent, J. Jaffre, Mathematical models and Finite Elements for Reservoir Simulation, vol. 17, Elsevier Science Publishers B. V., ISBN 0-444-70099-4, 1986.

[2] F. Brezzi, M. Fortin, Mixed and hybrid finite element methods, Springer-Verlag New York, Inc., New York, NY, USA, ISBN 0-387-97582-9, 1991.

[3] P. Raviart, J. Thomas, A mixed finite element method for second order elliptic problems, in Mahematical Aspects of Finite Element Method, Lecture Notes in Mathematics, Springer-Verlag (606) (1977) 292–315.

[4] D. Arnold, F. Brezzi, Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates, RAIRO Modél. Math. Anal. Numér. 19 (1) (1985) 7–32.

[5] V. Fontaine, A. Younes, Computational issues of hybrid and multipoint mixed methods for groundwater flow in anisotropic media, Computational Geosciences 14 (1) (2010) 171–181.

[6] A. Younes, V. Fontaine, Hybrid and multi-point formulations of the lowest-order mixed methods for Darcy's flow on triangles, International Journal for Numerical Methods in Fluids 58 (9) (2008) 1041–1062.

[7] D. Arnold, F. Brezzi, B. Cockburn, L. Marini, Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems, SIAM Journal on Numerical Analysis 39 (5) (2002) 1749–1779.

[8] R. M. Kirby, S. J. Sherwin, B. Cockburn, To CG or to HDG: A Comparative Study, Journal of Scientific Computing 51 (1) (2012) 183–212.

[9] B. Cockburn, W. Qiu, K. Shi, Conditions for superconvergence of HDG methods for second-order elliptic problems, Mathematics of Computation 81 (279) (2012) 1327–1353.

[10] B. Cockburn, J. Gopalakrishnan, A Characterization of Hybridized Mixed Methods for Second Order Elliptic Problems, SIAM Journal on Numerical Analysis 42 (1) (2004) 283–301.

[11] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems, SIAM Journal on Numerical Analysis 47 (2) (2009) 1319–1365.

[12] B. Cockburn, B. Dong, J. Guzmán, A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems, Mathematics of Computation 77 (264) (2008) 1887–1916.

[13] B. Cockburn, Static Condensation, Hybridization, and the Devising of the HDG Methods, Springer International Publishing, Cham, 129–177, 2016.

[14] D. Boffi, D. Di Pietro, Unified formulation and analysis of mixed and primal discontinuous skeletal methods on polytopal meshes, ESAIM: M2AN 52 (1) (2018) 1–28, doi:10.1051/m2an/2017036, URL https://doi.org/10.1051/m2an/2017036.

[15] D. Di Pietro, J. Droniou, G. Manzini, Discontinuous Skeletal Gradient Discretisation methods on polytopal meshes, Journal of Computational Physics 355 (2018) 397–425.

[16] B. Cockburn, D. Di Pietro, A. Ern, Bridging the hybrid high-order and hybridizable discontinuous Galerkin methods, ESAIM: M2AN 50 (3) (2016) 635–650.

[17] D. Di Pietro, A. Ern, S. Lemaire, A Review of Hybrid High-Order Methods: Formulations, Computational Aspects, Comparison with Other Methods, Springer International Publishing, Cham, 205–236, 2016.

[18] I. Oikawa, A Hybridized Discontinuous Galerkin Method with Reduced Stabilization, Journal of Scientific Computing 65 (1) (2015) 327–340.

[19] C. Lehrenfeld, Hybrid Discontinuous Galerkin methods for incompressible flow problems, Ph.D. thesis, Diploma thesis, Rheinisch-Westfälischen Technischen Hochschule Aachen, 2010.

[20] N. Nguyen, J. Peraire, B. Cockburn, A hybridizable discontinuous Galerkin method for Stokes flow, Computer Methods in Applied Mechanics and Engineering 199 (912) (2010) 582–597.

[21] N. Nguyen, J. Peraire, B. Cockburn, An Implicit High-order Hybridizable Discontinuous Galerkin Method for the Incompressible Navier-Stokes Equations, J. Comput. Phys. 230 (4) (2011) 1147–1170, ISSN 0021-9991.

[22] W. Qiu, K. Shi, An HDG Method for Convection Diffusion Equation, Journal of Scientific Computing 66 (1) (2016) 346–357.

[23] N. Nguyen, J. Peraire, B. Cockburn, An Implicit High-order Hybridizable Discontinuous Galerkin Method for Linear Convection-diffusion Equations, J. Comput. Phys. 228 (9) (2009) 3232–3254, ISSN 0021-9991.

[24] C. Lehrenfeld, J. Schöberl, High order exactly divergence-free Hybrid Discontinuous Galerkin Methods for unsteady incompressible flows, Computer Methods in Applied Mechanics and Engineering 307 (2016) 339 – 361, ISSN 0045-7825, doi:http://dx.doi.org/10.1016/j.cma.2016.04.025, URL `http://www.sciencedirect.com/science/article/pii/S004578251630264X`.

[25] D. Di Pietro, A. Ern, Hybrid High-Order methods for variable diffusion problems on general meshes, Comptes Rendus Mathématique 353 (2014) 31–34.

[26] P. L. Lederer, C. Lehrenfeld, J. Schöberl, Hybrid Discontinuous Galerkin methods with relaxed H(div)-conformity for incompressible flows. Part I, SIAM J. Numer. Anal. 56 (2018) 2070–2094, doi: 10.1137/17M1138078, URL `https://arxiv.org/abs/1707.02782`.

[27] P. L. Lederer, C. Lehrenfeld, J. Schöberl, Hybrid Discontinuous Galerkin methods with relaxed H(div)-conformity for incompressible flows. Part II, ESAIM: M2AN doi:10.1051/m2an/2018054, URL `https://doi.org/10.1051/m2an/2018054`.

[28] T. A. Davis, Direct Methods for Sparse Linear Systems (Fundamentals of Algorithms), Society for Industrial and Applied Mathematic, ISBN 9780898716139,0898716136, 2006.

[29] A. Samii, C. Michoski, C. Dawson, A parallel and adaptive hybridized discontinuous Galerkin method for anisotropic nonhomogeneous diffusion, Computer Methods in Applied Mechanics and Engineering 304 (2016) 118 – 139, ISSN 0045-7825.

[30] R. Herbin, F. Hubert, Benchmark on Discretization Schemes for Anisotropic Diffusion Problems on General Grids, in: Finite volumes for complex applications V, Wiley, France, 659–692, 2008.

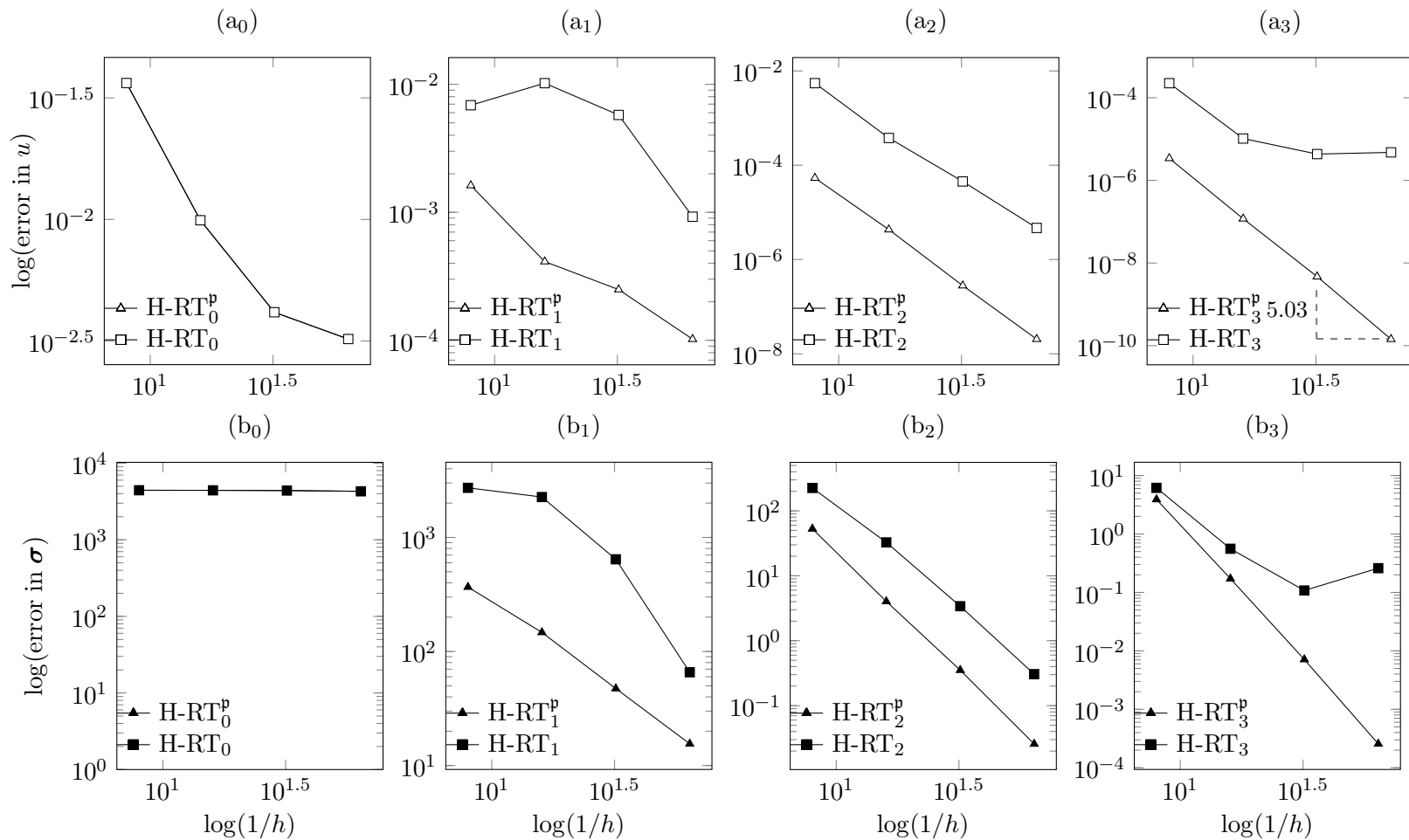Figure 3: Test C – History of convergence of the discrete variables for the H-RT$_k^{\mathfrak{p}}$ ($\triangle$: e_{u_h} and $\blacktriangle$: $e_{\boldsymbol{\sigma}_h^\star}$) and the H-RT$_k$ ($\square$: $e_{u_h^\star}$ and $\blacksquare$: $e_{\boldsymbol{\sigma}_h}$) methods as a function of the mesh size $1/h$ for different polynomial orders $0 \leq k \leq 3$ on irregular meshes.