# The eXteme Mesh deformation approach (X-Mesh) applied to the Porous Medium Equation

Alexandre Chemin, Jonathan Lambrechts, Nicolas Moës, Jean-François Remacle

# The eXteme Mesh deformation approach (X-Mesh) applied to the Porous Medium Equation

Alexandre Chemin* Jonathan Lambrechts† Nicolas Moës‡ Jean-François Remacle§

**Abstract**

Numerical simulation of the Porous Medium Equation (PME) is challenging due to the specific properties of its solutions. Notably, the PME exhibits a distinctive feature — a time-dependent interface emerges when the initial solution is zero in part of the spatial domain. We propose here to leverage the eXtreme Mesh deformation approach (X-Mesh), initially developed for two-phase problems, to efficiently and accurately compute solutions for the PME, tracking sharp interfaces and their potential topology changes without the need for remeshing or altering mesh topology.

## 1 Introduction

The Porous Medium Equation (PME) is a nonlinear partial differential equation of significant interest due to its widespread applicability in modeling various physical phenomena such as gas flow in porous medium, incompressible fluid dynamics and nonlinear heat transfer [6]. We consider a domain $\Omega \subset \mathbb{R}^d$, with $d$ the dimension of the problem ($d \in \{1, 2\}$) and a scalar function $u \in \mathcal{C}^0(\Omega \times [t_0, T])$.

The PME is defined as:

$$\frac{\partial u}{\partial t} = \nabla \cdot (\kappa u^m \nabla u), \quad m > 0 \tag{1.1}$$

In this equation, $u$ is a scalar function in space $x \in \mathbb{R}^d$ and time $t \in [t_0, T]$ and $\kappa u^m$ is homogeneous to a diffusivity coefficient. The function $u$ is frequently employed to denote physical quantities such as gas density or temperature, requiring the preservation of non-negativity in solutions.

The case $m \to 0$ leads to the heat equation:

$$\frac{\partial u}{\partial t} = \nabla \cdot (\kappa \nabla u) \tag{1.2}$$

PME is parabolic everywhere $u \neq 0$, and loses this property when $u = 0$. It is a degenerate parabolic equation [6]. What sets the PME's solutions apart from the heat equation's solutions are their distinctive solution behaviors. Unlike solutions from the heat equations, solutions to the PME can exhibit an interface $\Gamma$ separating regions where the solutions is strictly positive, and regions where the solution is null. Moreover, depending on initial solution and $m$ value, $\nabla u$ can be discontinuous across $\Gamma$.

As equation 1.1 is not defined for $u < 0$, it is usually rewritten for numerical solve purposes as:

$$\frac{\partial u}{\partial t} = \nabla \cdot (\kappa |u|^m \nabla u), \quad m > 0 \tag{1.3}$$

Equation 1.3 is called the signed PME. Solving 1.3 while ensuring non-negativity of the solution over the whole domain $\Omega$ is equivalent to solving 1.1. Conventional numerical methods, such as finite element methods (FEM), face notable challenges when applied to 1.3. Sharp interface in the solution domain lead to spurious oscillations breaking the non-negativity.

To overcome these challenges, we propose in this paper the use of X-Mesh method which is specifically designed to handle problems with sharp interfaces in the FEM context. X-Mesh has demonstrated its effectiveness in scenarios like the Stefan problem or two-phase flows [3, 5], excelling at capturing intricate interface dynamics, without the need of using enriched elements nor modifying the mesh topology. We will show that X-Mesh is a valuable tool for accurately simulating the PME while addressing the limitations of traditional FEM approaches.

## 2 The Porous Medium Equation

In this section we will detail important properties of PME and state of the art methods to solve it. We consider a domain $\Omega \subset \mathbb{R}^d$, and a known function $u_0 \in \mathcal{C}^1(\Omega)$ almost everywhere such as:

$$\begin{cases} u_0(\mathbf{x}) & \geq & 0 & , \forall \mathbf{x} \in \Omega \\ \dfrac{\partial u_0}{\partial \mathbf{n}}(\mathbf{x}) & = & 0 & , \forall \mathbf{x} \in \partial\Omega \end{cases} \tag{2.4}$$

*Université Catholique de Louvain
alexandre.chemin@uclouvain.be
†Université Catholique de Louvain
jonathan.lambrechts@uclouvain.be
‡École Centrale de Nantes
nicolas.moes@ec-nantes.fr
§Université Catholique de Louvain
jean-francois.remacle@uclouvain.be

The goal is, for fixed $m > 0$ and $\kappa > 0$, to find $u$ verifying:

(2.5)
$$\begin{cases} \dfrac{\partial u}{\partial t}(\mathbf{x},t) &= \nabla \cdot (\kappa |u(\mathbf{x},t)|^m \, \nabla u(\mathbf{x},t)) \,, \\ & \qquad \forall (\mathbf{x},t) \in \Omega \times [t_0, T] \\ u(\mathbf{x},t_0) &= u_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \\ \dfrac{\partial u(\mathbf{x},t)}{\partial \mathbf{n}} &= 0, \qquad \forall (\mathbf{x},t) \in \partial\Omega \times [t_0, T] \\ u(\mathbf{x},t) &\geq 0, \qquad \forall (\mathbf{x},t) \in \partial\Omega \times [t_0, T] \end{cases}$$

As already mentioned, solutions of the PME can exhibit an interface separating regions where the solution is null from region where the solution is strictly positive. Therefore, problems modeled with the PME fall into the categorie of *one-phase* problems. For clarity purposes, we are defining here $\mathcal{P}_{u(t)} = \{\mathbf{x} \in \Omega \text{ such as } u(\mathbf{x},t) > 0\}$ , $\Gamma_{u(t)} = \partial\mathcal{P}_{u(t)}$ and $\mathcal{Q}_{u(t)} = \Omega \setminus (\mathcal{P}_{u(t)} \cup \partial\mathcal{P}_{u(t)})$. $\mathcal{P}_{u(t)}$ will be refered to as *the phase*, $\Gamma_{u(t)}$ as *the interface* and $\mathcal{Q}_{u(t)}$ the *empty region* for a given solution $u(t)$ on $\Omega$. The PME has a lot of interesting properties and we will focus on two that a numerical simulation should be able to ensure (see [6] for a more extensive list).

**Mass conservation** An important property of the PME is that with homogeneous Neumann boundary conditions on $\partial\Omega$ is the invariance in time of the quantity $\int_\Omega u \, \mathrm{d}\Omega$. General proof can be found in [6]. In the case of the PME modeling non linear heat transfer, this correspond to internal energy conservation. In the case of the PME modeling a gas flow in porous medium, this corresponds to mass conservation. For convenience, this property will be refered to as mass conservation.

**Barenblatt-Pattle solution** Another important result is the existence of an analytic solution for Equation 1.3. The solution of the form:

(2.6)
$$u(\mathbf{x},t) = t^{-\alpha} \left( (C - k|\mathbf{x}|^2 t^{-2\beta})^+ \right)^{\frac{1}{m}}$$
$$\text{with } \alpha = \frac{d}{md+2}, \, \beta = \frac{\alpha}{d}, \, k = \frac{m\alpha}{2d}.$$

with $C$ an arbitrary constant is solution of the PME. This solution corresponds to an intial solution $u_0$ being a Dirac distribution for which the total integral $\int_\Omega u_0 \, \mathrm{d}\Omega = M(C)$ depends on $C$. For this solution, the interface localization is $f(t) = (\frac{C}{k})^{\frac{1}{2}} t^\beta$.

**Solving PME with Finite Element Method** When solving 2.5 with FEM, the property of non-negativity is broken and oscillations appear at interface $\Gamma_u$ which propagates in the whole *empty region*. An example of such oscillations is shown Figure 1 in 1D as illustration.

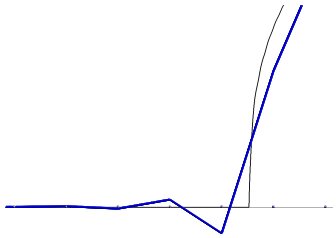

Figure 1: In black: analytical solution of equation 1.3. In blue: solution obtained with the FEM. The discretization induces oscillations propagating in the *empty region* which are breaking the non-negativity property.

To circumvent this issue, several approaches have been proposed. A first idea is to use a numerical scheme which ensure non-negativity of the solution. Such a method has been proposed in [7] where a local discontinuous Galerkin FEM coupled with a custom made limiter to ensure non-negativity of the solution. A demonstration is provided showing that this limiter ensure non-negativity of the solution for the discontinuous $\mathbb{P}_0$ finite elements, and provide numerical 1D results for discontinuous $\mathbb{P}_2$ finite elements in which non-negativity is ensured and convergence rate of 3 is retreived.

A second type of approach is to use a time dependant spatial discretization. In [4] is proposed an adaptative moving mesh FEM. Meshes used for solving the PME are time dependant but all have the same topology (same number of vertices and elements, and same connectivity). Each mesh is generated such as its elements' sizes respect an imposed metric and different metrics are investigated. They are showing that using a target metric based on the Hessian of the solution at previous time step allows to retreive a convergence rate in $\mathcal{O}(h^2)$ for $m = 1$ and $m = 2$ for the Barenblatt-Pattle solution. It is important to note that with this method, non-negativity of the solution is broken and oscillations at the interface $\Gamma_{u(t)}$ are still showing. However, the target metric used leads to a smaller element size near the interface decreasing significantly oscillations propagation in the *empty region*.

In [1] is proposed a moving mesh finite element algorithm for PDE with moving boundaries, and is applied to the Barenblatt-Pattle solution of the PME. In that case, only the part of the computational domain where $u > 0$ is meshed and the mesh evolves in time in order to ensure mass conservation of the solution. This allows to obtain a second-order convergence rate for $m = 1$ and a first-order convergence rate for $m = 3$. Moreover, this approach does not handle topology

changes of $\mathcal{P}_u(t)$ which can lead the mesh to overlap itself and need to be coupled with a remeshing algorithm for such cases, which is not shown in the paper.

The method proposed in this paper falls into the moving meshes category. The X-MESH method principle is to find at each time step $t_n$ the solution $u_n$ and the spatial discretization $\mathcal{T}_n$ such as $\mathcal{T}_n$ has the same topology as $\mathcal{T}_{n-1}$ and that $\Gamma_{u_n}$ is fully represented by vertices and edges of $\mathcal{T}_n$.

## 3 The X-Mesh approach

As previously mentioned, the goal of the X-MESH approach is to use a time dependant spatial discretization. In order to develop a discrete formulation of the problem with this constraint, we introduce the arbitrary Lagrangian Eulerian frame of reference [2]. We define the reference domain $\Omega_0$ as $\Omega_0 = \Omega$, and a mapping $\mathbf{X} \in (H^1(\Omega_0 \times [t_0, T]))^d$:

$$\begin{array}{rcl} \mathbf{X} : \Omega_0 \times [t_0, T] & \rightarrow & \mathbb{R}^d \\ (\mathbf{X}_0, t) & \mapsto & X(\mathbf{X}_0, t) \end{array} \tag{3.7}$$

We define $\Omega_t = \{\mathbf{X}(\mathbf{X}_0, t), \mathbf{X}_0 \in \Omega_0, t \in [t_0, T]\}$, $\mathbf{w} = \left.\frac{\partial \mathbf{X}}{\partial t}\right|_{\mathbf{X}_0}$, $\mathbf{F} = \left.\frac{\partial \mathbf{X}}{\partial \mathbf{X}_0}\right|_t$ and $J = \det \mathbf{F}$. The variational formulation obtained from 1.3 is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \bar{u}u \,\mathrm{d}\Omega = -\int_\Omega \kappa|u|^m \nabla\bar{u}.\nabla u \,\mathrm{d}\Omega - \int_\Omega u\mathbf{w}\nabla\bar{u} \,\mathrm{d}\Omega \tag{3.8}$$

The spatial discretization is done by partitioning the reference domain $\Omega_0$ into a triangular mesh $\mathcal{T}_0$ for which the set of nodes is denoted $\mathcal{N}_0$. The spatial discretization at $t$ is the triangulation $\mathcal{T}(t)$ obtained from applying the mapping $\mathbf{X}$ to $\mathcal{T}_0$. By construction, the mesh topology (number of nodes and connectivities) does not depend on time. The time integration of equation 3.8 between two instants $t_n$ and $t_{n+1}$ is done using a $\theta$ scheme.

The X-MESH approach is summed up in Algorithm 1, were *solve* refers to a classical finite element resolution with two fixed triangulations $\mathcal{T}_n$ and $\mathcal{T}_{n+1}$, *localize_interface* refers to a selection procedure flagging nodes of $\mathcal{T}_{n+1}$ which should belong to the interface, *update_interface* refers to a modification of $\mathcal{T}_{n+1}$ moving flagged nodes to an estimated interface position and $R$ refers to the residual evaluation.

**Algorithm 1:** Solving scheme for $[t_n, t_{n+1}]$ time step. Solution $(U_n, \mathbf{X}_n)$ at instant $t_n$ is known, *tol* is a user imposed tolerance for convergence.

```
k ← 0;
X^0_{n+1} ← X_0;
U^0_{n+1} ← solve((U_n, X_n), X^0_{n+1});
N_{Γ_{u^h_{n+1}}} ← localize_interface(U^0_{n+1}, X^0_{n+1});
while ε > tol do
    X^{k+1}_{n+1} ←
     update_interface(N_{Γ_{u^h_{n+1}}}, (U_n, X_n), (U^k_{n+1}, X^k_{n+1}));
    U^{k+1}_{n+1} ← solve((U_n, X_n), X^{k+1}_{n+1});
    ε ← R((U_n, X_n), (U^{k+1}_{n+1}, X^{k+1}_{n+1}));
    k ← k + 1;
(U_{n+1}, X_{n+1}) ← (U^k_{n+1}, X^k_{n+1});
```

**3.1 Interface location estimation and mesh update** Given a solution $(U_{n+1}, \mathbf{X}_{n+1})$, it is possible to split the set of nodes $\mathcal{N}$ into two subsets $\mathcal{N}_\mathcal{P}$ and $\mathcal{N}_\mathcal{Q}$:

$$\left\{ \begin{array}{rcl} \mathcal{N}_\mathcal{P} & = & \{i \in \mathcal{N} \mid U^i_{n+1} > 0\} \\ \mathcal{N}_\mathcal{Q} & = & \mathcal{N} \setminus \mathcal{N}_\mathcal{P} \end{array} \right. \tag{3.9}$$

$\mathcal{N}_\mathcal{P}$ corresponds to the set of nodes which are inside the phase at $t_{n+1}$ and $\mathcal{N}_\mathcal{Q}$ to the set of nodes which are outside the phase. Interface $\Gamma_{u^h_{n+1}}$ is localized inside triangles of $\mathcal{T}_{n+1}$ which vertices are belonging to $\mathcal{N}_\mathcal{P}$ and $\mathcal{N}_\mathcal{Q}$, but the exact location is not known a priori. We now define:

$$\mathcal{N}_{\Gamma_{u^h_{n+1}}} = \{i \in \mathcal{N}_\mathcal{Q} \mid i \text{ is connected to at least one node of } \mathcal{N}_\mathcal{P}\} \tag{3.10}$$

$\mathcal{N}_{\Gamma_{u^h_{n+1}}}$ is the set of nodes which will be used to describe $\Gamma_{u^h_{n+1}}$ (Figure 2). It is defined only once for time step $t_{n+1}$ and will remain unchanged during the successive updates of $\mathcal{T}_{n+1}$ nodes localization. This step is called *localize_interface* in Algorithm 1. During the mesh update procedure, nodes from $\mathcal{N}_{\Gamma_{u^h_{n+1}}}$ will be moved to $\Gamma_{u^h_{n+1}}$ estimated location.

It is important to note that moving all nodes from $\mathcal{N}_{\Gamma_{u^h_{n+1}}}$ to $\Gamma_{u^h_{n+1}}$ is an ill-posed nonlinear problem. The approach adopted in this work is to compute and update an estimated optimal position for each node of $\mathcal{N}_{\Gamma_{u^h_{n+1}}}$ independantly from each other and reiterate this process until convergence.

For $p \in \mathcal{N}_{\Gamma_{u^h_{n+1}}}$ a node which has been flagged to be belonging to the interface $\Gamma_{u^h_{n+1}}$. In order to get a good estimation of the interface $\Gamma_{u^h_{n+1}}$ location, we are considering the current solution $U_{n+1}$ only on the patch of triangles $\mathcal{T}^p_{n+1}$ sharing $p$ as a vertex. We denote $\mathcal{N}^p$ the set of nodes of $\mathcal{T}^p_{n+1}$ and $\bar{\mathcal{N}^p} = \mathcal{N}^p \setminus \mathcal{N}_{\Gamma_{u^h_{n+1}}}$ the set of nodes connected to $p$ which does not belong to $\mathcal{N}_{\Gamma_{u^h_{n+1}}}$.

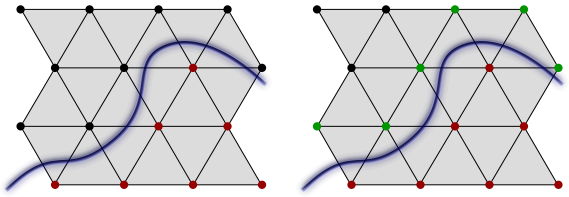

Figure 2: Left: in red nodes of $\mathcal{N}_\mathcal{P}$ and in black nodes of $\mathcal{N}_\mathcal{Q}$. Right: In green are highligted nodes selected to belong to $\mathcal{N}_{\Gamma_{u^h_{n+1}}}$. The blurred blue line represent the approximate location of $\Gamma_{u^h_{n+1}}$.

Finally, we define $\tilde{U}_{n+1}$ such as:

$$\begin{cases} \tilde{U}^i_{n+1} = U^i_{n+1}, \forall i \in \mathcal{N}^p, i \neq p \\ \tilde{U}^p_{n+1} = 0 \end{cases} \tag{3.11}$$

With $\phi_p$ the finite element basis function associated to node $p$, we have that $p$ belongs to interface $\Gamma_{u^h_{n+1}}$ is equivalent to:

$$\begin{aligned} & r(\phi_p, \tilde{U}_{n+1}, \mathbf{X}_{n+1}, U_n, \mathbf{X}_n) \\ = & \int_{t_n}^{t_{n+1}} \left( \frac{\mathrm{d}}{\mathrm{d}t} \int_\Omega \phi_p \tilde{u} \, \mathrm{d}\Omega \right. \\ & \left. + \int_\Omega \kappa |\tilde{u}|^m \nabla \phi_p . \nabla \tilde{u} \, \mathrm{d}\Omega + \int_\Omega \tilde{u} \mathbf{w} \nabla \phi_p \, \mathrm{d}\Omega \right) \mathrm{d}t \\ = & \ 0 \end{aligned} \tag{3.12}$$

We define now a vector $\mathbf{v} \in \mathbb{R}^d$ and a new triangulation $\tilde{\mathcal{T}}^\mathbf{v}_{n+1}$ obtained from the triangulation $\mathcal{T}_{n+1}$ where the node $p$ is translated by $\mathbf{v}$.

The problem of finding a new location for $p$ such as $p \in \Gamma_{u^h_{n+1}}$ can then be formulated as:

$$\begin{aligned} & \text{Find } \mathbf{v} \in \mathbb{R}^d \text{ such as} \\ & r(\phi_p, \tilde{U}_{n+1}, \tilde{\mathbf{X}}^\mathbf{v}_{n+1}, U_n, \mathbf{X}_n) = 0 \end{aligned} \tag{3.13}$$

This problem does not have a unique solution. In order to simplify the search for a valid $\mathbf{v}$, we are only considering translations along edges connecting $p$ to $\bar{\mathcal{N}}^p$. Therefore, we only consider vectors $\mathbf{v} = \eta_{pj}\mathbf{v}_{pj}$ where:

$$\begin{cases} \mathbf{v}_{pj} = \mathbf{X}_{n+1}(j) - \mathbf{X}_{n+1}(p), \ j \in \bar{\mathcal{N}}^p \\ \eta_{pj} \in [0, 1] \end{cases} \tag{3.14}$$

By doing so, solving problem 3.13 comes down to solve $\#(\bar{\mathcal{N}}^p)$ sub-problems:

$$\begin{cases} \text{With } p, \tilde{U}_{n+1}, U_n, \mathbf{X}_n, \mathbf{v}_{pj} \text{ fixed, and } j \in \bar{\mathcal{N}}^p \\ \text{find } \eta_{pj} \in [0, 1] \text{ such as} \\ r(\eta_{pj}) = r(\phi_p, \tilde{U}_{n+1}, \tilde{\mathbf{X}}_{n+1}(\eta_{pj}\mathbf{v}_{pj}), U_n, \mathbf{X}_n) = 0 \end{cases} \tag{3.15}$$

Defining $\mathcal{S}_p = \{\eta_{pj}\mathbf{v}_{pj} \mid (\eta_{pj}, \mathbf{v}_{pj}) \text{ verifies } 3.15\}$ the set of possible translations to move node $p$, the one inducing the smallest mesh deformation is selected.

## 4 Results and conclusion

To evaluate the X-Mesh approach performances, the method is applied to retreive the Barenblatt-Pattle solution 2.6 for $d = 2$, $\kappa = 1$ and various values of $m$. The quantities of interest are the interface localization and the mass conservation. At every time step, the error in interface localization relative to the mesh size ($f_r$) and the mass variation relative to the initial solution's mass ($M_r$) are computed for each obtained solution. These results are depicted Figure 3 and show for $m \geq 1$ an enhancement on interface localization by one order of magnitude compared to a classical FEM resolution, and that the mass conservation property is ensured up to the solver precision.

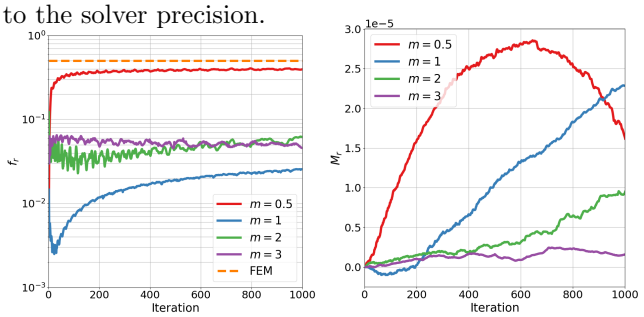


Figure 3: Left: error on interface localization relative to the mesh size for different values of $m$. The dashed orange line represent the average interface localization error relative to the mesh size obtained with a classical FEM. Right: Total relative mass variation for different values of $m$.

Figure 4 describes the simulation of a configuration were interfaces are coalescing. In that case, the *phase* $\mathcal{P}_{u(t)}$ changes topology during time, which the X-Mesh approach handles automatically without any difficulties.

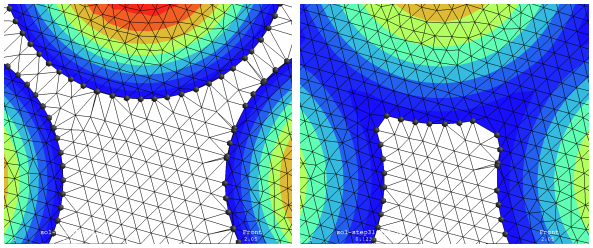

Figure 4: Two chronological time steps in the case of coalescing interfaces. Colored elements belong to the *phase*, white one to the *empty region* and gray spheres are nodes $\mathcal{N}_\Gamma$ representing the interface.

These primary results illustrate that X-Mesh approach is an interesting path to follow for one-phase problems' simulations.

## References


[1] M. J. Baines, M. Hubbard, and P. Jimack, *A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries*, Applied Numerical Mathematics, 54 (2005), pp. 450–469.

[2] D. Boffi and L. Gastaldi, *Stability and geometric conservation laws for ale formulations*, Computer methods in applied mechanics and engineering, 193 (2004), pp. 4717–4739.

[3] N. Moes, J.-F. Remacle, J. Lambrechts, B. Le, and N. Chevaugeon, *The eXtreme Mesh deformation approach (X-MESH) for the Stefan phase change model*, Journal of Computational Physics, (2023), p. 111878.

[4] C. Ngo and W. Huang, *A study on moving mesh finite element solution of the porous medium equation*, Journal of Computational Physics, 331 (2017), pp. 357–380.

[5] A. Quiriny, J. Lambrechts, N. Moës, and J.-F. Remacle, *X-mesh: A new approach for the simulation of two-phase flow with sharp interface*, arXiv preprint arXiv:2302.03983, (2023).

[6] J. L. Vázquez, *The porous medium equation: mathematical theory*, Oxford University Press, 2007.

[7] Q. Zhang and Z.-L. Wu, *Numerical simulation for porous medium equation by local discontinuous galerkin finite element method*, Journal of Scientific Computing, 38 (2009), pp. 127–148.