Commun. Optim. Theory 2023 (2023) 8 https://doi.org/10.23952/cot.2023.8

MATHRES Communications in Optimization Theory
Available online at http://cot.mathres.org

NUMERICAL APPROXIMATION OF RIGID MAPS IN ORIGAMI THEORY

ALEXANDRE CABOUSSAT[1,*], DIMITRIOS GOURZOULIDIS[1,2]

[1]Geneva School of Business Administration (HEG), University of Applied Sciences and Arts Western Switzerland (HES-SO), Switzerland
[2]Institute of Mathematics, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Dedicated to the memory of Professor Roland Glowinski

Abstract. In origami theory, the problem of rigid maps consists in finding a paper folding from the two-dimensional space onto the three-dimensional space. This problem is an example of a first-order fully nonlinear equation. In this article, we present a general variational framework to solve the problem of rigid maps with Dirichlet boundary conditions. The numerical framework relies on the introduction of a regularized objective function and the penalization of the constraints. A splitting algorithm is advocated for the corresponding flow problem. The iterations sequence consists of local nonlinear problems and a global linear variational problem at each step. Numerical experiments validate the efficiency of the method for piecewise smooth exact solutions.

Keywords. Fully nonlinear equations; Origami; Rigid maps; Relaxation algorithm; Splitting algorithm; Variational principles.

2020 Mathematics Subject Classification. 65N30, 65K10, 65M60, 49M20, 35G30.

1. INTRODUCTION

Origami is the art of folding paper. It has received a lot of attention from the analysis and geometry communities [7, 9, 11, 12], and more recently algorithms have been proposed to mimic origami constructions in simple cases.

An origami is nothing but the immersion and folding of a two-dimensional sheet of paper into the three-dimensional space. The main feature to consider is the rigidity of the paper to fold. The tangential rigidity can be expressed by requiring that the folding map $\mathbf{u}$ is a *rigid map* of $\Omega \subset \mathbb{R}^2$ into $\mathbb{R}^3$. This means that the gradient of the map, $\nabla \mathbf{u}(\mathbf{x})$, is an orthogonal 3×2 matrix for all $\mathbf{x} \in \Omega$. Moreover, since origami is about folding paper, this map cannot be smooth everywhere; it is only piecewise smooth, with discontinuities of the gradient created by the folding lines. In this work, we are going to assume that the discontinuities are a set

*Corresponding author.
E-mail address: alexandre.caboussat@hesge.ch (A. Caboussat), dimitrios.gourzoulidis@hesge.ch (D. Gourzoulidis).

Received May 11, 2022; Accepted December 22, 2022.

of straight segments, even though these creasing lines could be curved [11]. Note that a map with an orthogonal gradient is not necessarily piecewise linear, as we will see in the numerical experiments. Functions with orthogonal gradients also appear in other applications in physics and engineering, such as bending [2, 3], computational geometry [1, 13], or smart materials [14].

In this article, we rely on a calculus of variations approach to solve the rigid maps problem. In particular, we extend the numerical method used for the so-called orthogonal maps problem [5] to the rigid maps problem. The objective of introducing a general numerical framework is to avoid making a priori assumptions on the shape or structure of the solution, as, e.g., in [8, 10] where the piecewise linear solution is assumed to either have a given expression, or follow a given pyramidal structure.

The main ingredients of the methodology include a *penalization* method to relax the orthogonality condition, the derivation of the Euler-Lagrange equation of a penalized and regularized problem, and the introduction of a related initial value *dynamical flow* problem. The discretization of the flow problem is addressed with an *operator-splitting* scheme (à la Marchuk-Yanenko) for the time-discretization, and a low order C^0-conforming finite element approximation. The operator-splitting approach allows the decoupling of the differential operators from the nonlinearities of the problem.

This work is organized as follows. In Section 2, we quickly develop a general mathematical methodology to solve first-order fully nonlinear equations. Then we use the rigid maps problem to illustrate practically our methodology. Section 3 covers the variational methodology, while Section 4 addresses the finite element discretization. Numerical experiments are presented in Section 5.

2. Mathematical formulation

2.1. **General framework.** Let Ω be an open bounded regular domain of $\mathbb{R}^2$. The general formulation of a vectorial first-order fully nonlinear equation can be expressed (in three space dimensions) as: find $\mathbf{u}:\Omega\to\mathbb{R}^3$ such that

$$F(\mathbf{x},\mathbf{u},\nabla\mathbf{u})=0, \tag{2.1}$$

where $\mathbf{x}\in\Omega, \mathbf{u}\in\mathbb{R}^3, \nabla\mathbf{u}\in\mathbb{R}^{3\times 2}$, and where there is a strong, or implicit, nonlinearity involving $\nabla\mathbf{u}$. Equation (2.1) is augmented with boundary conditions $\mathbf{u}=\mathbf{g}$ on $\partial\Omega$ when suitable, where $\mathbf{g}:\partial\Omega\to\mathbb{R}^3$ is a sufficiently smooth given function.

2.2. **Rigid maps.** Let us consider the unit square $\Omega=(-1,1)^2$. The problem of so-called rigid maps [9] consists in finding a map $\mathbf{u}:\Omega\subset\mathbb{R}^2\to\mathbb{R}^3$ such that

$$\begin{cases} \nabla\mathbf{u}\in\mathscr{O}(3,2) & \text{in } \Omega,\\ \mathbf{u}=\mathbf{g} & \text{on } \partial\Omega, \end{cases} \tag{2.2}$$

where in (2.2)

$$\mathbf{u}(x,y) = \begin{pmatrix} u_1(x,y) \\ u_2(x,y) \\ u_3(x,y) \end{pmatrix}, \qquad \nabla\mathbf{u}(x,y) = \begin{pmatrix} \frac{\partial u_1}{\partial x_1}(x,y) & \frac{\partial u_1}{\partial x_2}(x,y) \\ \frac{\partial u_2}{\partial x_1}(x,y) & \frac{\partial u_2}{\partial x_2}(x,y) \\ \frac{\partial u_3}{\partial x_1}(x,y) & \frac{\partial u_3}{\partial x_2}(x,y) \end{pmatrix},$$

and $\mathscr{O}(3,2) = \left\{\mathbf{A} : \Omega \to \mathbf{R}^{3\times 2} : \mathbf{A}^T\mathbf{A} = \mathbf{I}_2 \text{ in } \Omega\right\}$. The solution to this equivalent problem consists in finding the mapping $\mathbf{u}$ that embeds the paper Ω into its image $\mathbf{u}(\Omega) \subset \mathbb{R}^3$. A graphical representation of problem (2.2) is illustrated in Figure 1 (top). This application actually maps a two-dimensional paper into the three-dimensional space, unlike the so-called orthogonal maps that considers the mapping $\tilde{\mathbf{u}}$ of a two-dimensional paper into a two-dimensional space representing the fully-folded paper, see Figure 1 (bottom).

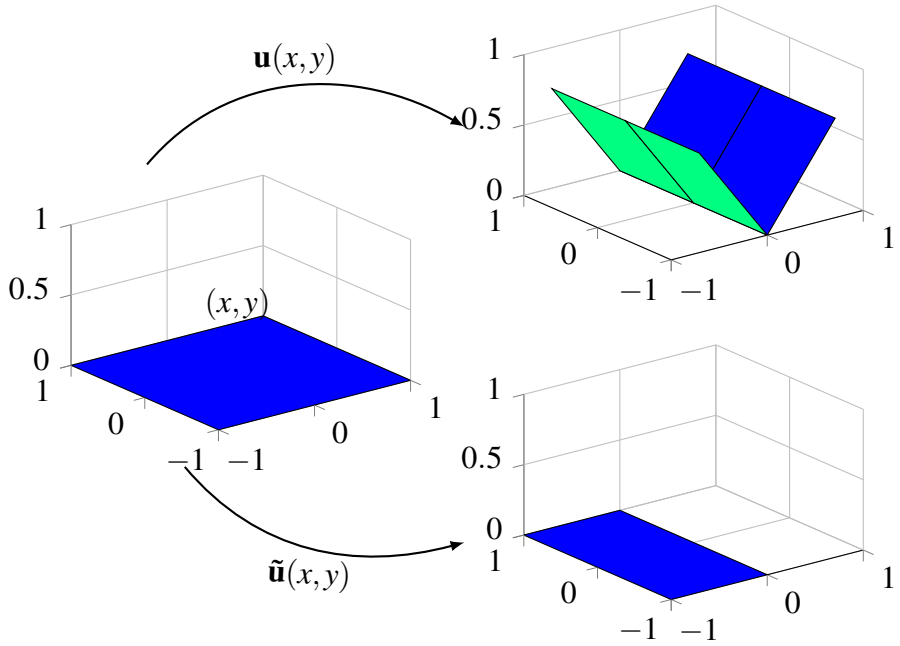

FIGURE 1. Visualization (top) of the rigid map $\mathbf{u} : \mathbb{R}^2 \to \mathbb{R}^3$, where $\mathbf{u}(x,y) = (x/\sqrt{2}, y, |x|/\sqrt{2})$ is the solution to (2.2) when folding a paper sheet in $\mathbb{R}^3$. Visualization (bottom) of the orthogonal map $\tilde{\mathbf{u}} : \mathbb{R}^2 \to \mathbb{R}^2$, where $\tilde{\mathbf{u}}(x,y) = (-|x|, y)$ is the solution to (2.5) when folding a paper sheet along a middle axis oriented with the Ox axis.

Problem (2.2) can be equivalently written as: Find $\mathbf{u} : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ such that

$$\begin{cases} \nabla\mathbf{u}^T\nabla\mathbf{u} = \mathbf{I}_2 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{g} & \text{on } \partial\Omega, \end{cases} \tag{2.3}$$

or, equivalently, with $\mathbf{u} = [u_1, u_2, u_3]^T$:

$$\begin{cases} \left(\dfrac{\partial u_1}{\partial x_1}\right)^2 + \left(\dfrac{\partial u_2}{\partial x_1}\right)^2 + \left(\dfrac{\partial u_3}{\partial x_1}\right)^2 = 1 & \text{in } \Omega \\ \left(\dfrac{\partial u_1}{\partial x_2}\right)^2 + \left(\dfrac{\partial u_2}{\partial x_2}\right)^2 + \left(\dfrac{\partial u_3}{\partial x_2}\right)^2 = 1 & \text{in } \Omega \\ \dfrac{\partial u_1}{\partial x_1}\dfrac{\partial u_1}{\partial x_2} + \dfrac{\partial u_2}{\partial x_1}\dfrac{\partial u_2}{\partial x_2} + \dfrac{\partial u_3}{\partial x_1}\dfrac{\partial u_3}{\partial x_2} = 0 & \text{in } \Omega \\ \mathbf{u} = \mathbf{g} & \text{on } \partial\Omega. \end{cases} \tag{2.4}$$

As illustrated in Figure 1 (bottom), the rigid maps problem can also be re-formulated as an *orthogonal maps* problem, when considering the resulting folded paper in the two-dimensional space. The problem of orthogonal maps has been treated in [5, 6]. It reads: find $\tilde{\mathbf{u}} = [u_1, u_2]^T : \Omega \to \mathbb{R}^2$ satisfying

$$\begin{cases} \nabla\tilde{\mathbf{u}} \in \mathscr{O}(2) & \text{in } \Omega, \\ \tilde{\mathbf{u}} = \tilde{\mathbf{g}} & \text{on } \partial\Omega. \end{cases} \tag{2.5}$$

Actually $\tilde{\mathbf{u}} = [\tilde{u}_1, \tilde{u}_2]^T$ is a mapping that maps Ω (the original sheet) into another domain of $\mathbb{R}^2$. The image $\tilde{\mathbf{u}}(\Omega)$ of Ω through this mapping corresponds to the resulting paper sheet, described in the two-dimensional space, after the folding. It is also called *flat origami*. Problem (2.5) can be equivalently written, component-wise, as:

$$\begin{cases} |\nabla\tilde{u}_1| = 1 & \text{a.e in } \Omega, \\ |\nabla\tilde{u}_2| = 1 & \text{a.e in } \Omega, \\ \nabla\tilde{u}_1 \cdot \nabla\tilde{u}_2 = 0 & \text{a.e in } \Omega, \\ \tilde{\mathbf{u}} = \tilde{\mathbf{g}} & \text{on } \partial\Omega. \end{cases} \tag{2.6}$$

3. A VARIATIONAL FRAMEWORK

3.1. **General principles.** The proposed solution method relies on a mix of classical variational techniques. Let us first sketch the general approach to solve (2.1). A general variational approach consists in considering the following variational problem under constraints : Find $\mathbf{u} : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ such that:

$$\begin{array}{ll} \min\limits_{\mathbf{u}\in V} & J(\mathbf{u}) \\ \text{s.t.} & F(\mathbf{x}, \mathbf{u}, \nabla\mathbf{u}) = 0, \end{array} \tag{3.1}$$

where V is a given functional space taking into account boundary conditions. Penalizing the equality constraint, together with introducing a smoothing term, leads to the following unconstrained variational problem: Find $\mathbf{u} : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ such that:

$$\min_{\mathbf{u}\in V} \left\{ J(\mathbf{u}) + \varepsilon_1 \tilde{J}(\mathbf{u}) + \frac{1}{2\varepsilon_2} F(\mathbf{x}, \mathbf{u}, \nabla\mathbf{u})^2 \right\}, \tag{3.2}$$

where $\varepsilon_1, \varepsilon_2$ are two (small) parameters, and $\tilde{J}(\cdot)$ is a smooth positive convex term. Next, we compute the first-order optimality conditions relative to (3.2). This leads to: find $\mathbf{u} \in V$ such that:

$$\langle J'(\mathbf{u}), \mathbf{v} \rangle + \langle \varepsilon_1 \tilde{J}'(\mathbf{u}), \mathbf{v} \rangle + \frac{1}{\varepsilon_2} \langle F(\mathbf{x}, \mathbf{u}, \nabla \mathbf{u}) \nabla F(\mathbf{x}, \mathbf{u}, \nabla \mathbf{u}), \mathbf{v} \rangle = 0, \tag{3.3}$$

for all test functions $\mathbf{v}$. Next, we introduce a pseudo-time $t \in (0, T)$, with $T > 0$ given, and we consider a so-called *flow* problem. Starting with $\mathbf{u}(0) = \mathbf{u}^0$ given, we solve:

$$\left\langle \frac{\partial \mathbf{u}}{\partial t}, \mathbf{v} \right\rangle + \langle J'(\mathbf{u}), \mathbf{v} \rangle + \langle \varepsilon_1 \tilde{J}'(\mathbf{u}), \mathbf{v} \rangle + \frac{1}{\varepsilon_2} \langle F(\mathbf{x}, \mathbf{u}, \nabla \mathbf{u}) \nabla F(\mathbf{x}, \mathbf{u}, \nabla \mathbf{u}), \mathbf{v} \rangle = 0. \tag{3.4}$$

Problem (3.4) can be addressed with time-stepping algorithm, as advocated in the sequel, to decouple the final term (the penalization term) from the other two terms, which are variational terms. Let us focus on the case of rigid maps for a more practical development and implementation.

3.2. **Rigid Maps.** In order to solve (2.3) and to enforce the uniqueness of the solution in some sense, we consider the following variational problem. Find $\mathbf{u} \in \mathbf{E_g}$ satisfying

$$J(\mathbf{u}) \leq J(\mathbf{v}), \ \forall \mathbf{v} \in \mathbf{E_g}, \tag{3.5}$$

where

$$J(\mathbf{v}) = \frac{C}{2} \int_\Omega |\mathbf{v} - \mathbf{f}|^2 \, d\mathbf{x} + \frac{1}{2} \int_\Omega \nabla \mathbf{v}^T \nabla \mathbf{v} d\mathbf{x}, \tag{3.6}$$

and

$$\mathbf{E_g} = \{\mathbf{v} \in (H^1(\Omega))^3, \ \mathbf{v} = \mathbf{g}, \text{ on } \partial\Omega, \ \nabla \mathbf{v}^T \nabla \mathbf{v} = \mathbf{I}_2 \text{ a.e. in } \Omega\}. \tag{3.7}$$

Here $C > 0$ is a given positive constant, $\mathbf{f}$ is a given vector-valued function (a potential *fidelity* term). The choice of objective function is not unique.

Note that the equation (2.3) itself has been transferred into a constraint in (3.7). To handle such constraints, we use a penalization approach that has been successful with the scalar Eikonal equation in [4]. Let $\varepsilon_1 > 0$ be a regularization parameter, and $\varepsilon_2 > 0$ be a given (penalization) parameter. We denote $\varepsilon := (\varepsilon_1, \varepsilon_2)$. The modified objective function is defined as follows:

$$\begin{aligned} J_\varepsilon(\mathbf{v}) \ := \ & J(\mathbf{v}) + \frac{\varepsilon_1}{2} \int_\Omega |\nabla^2 \mathbf{v}|^2 d\mathbf{x} \\ & + \frac{1}{4\varepsilon_2} \int_\Omega \left[\left(\left(\frac{\partial u_1}{\partial x_1} \right)^2 + \left(\frac{\partial u_2}{\partial x_1} \right)^2 + \left(\frac{\partial u_3}{\partial x_1} \right)^2 - 1 \right)^2 + \right. \\ & \left(\left(\frac{\partial u_1}{\partial x_2} \right)^2 + \left(\frac{\partial u_2}{\partial x_2} \right)^2 + \left(\frac{\partial u_3}{\partial x_2} \right) - 1 \right)^2 + \\ & \left. \left(\frac{\partial u_1}{\partial x_1} \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_1} \frac{\partial u_3}{\partial x_2} \right)^2 \right] d\mathbf{x}. \end{aligned} \tag{3.8}$$

The second term is a biharmonic regularization introduced to improve the convergence of the numerical algorithm. It artificially smoothes the folding lines of the solution, but it improves the general convergence behavior of the algorithm in stringent cases. The variational problem (3.5) becomes: Find $\mathbf{u}^\varepsilon \in \mathbf{V_g}$ satisfying

$$J_\varepsilon(\mathbf{u}^\varepsilon) \leq J_\varepsilon(\mathbf{v}),\ \forall \mathbf{v} \in \mathbf{V_g}, \tag{3.9}$$

where

$$\mathbf{V_g} = \{\mathbf{v} \in (H^2(\Omega))^3, \mathbf{v} = \mathbf{g} \text{ on } \partial\Omega\}. \tag{3.10}$$

The numerical approach to solve (3.9) relies on an appropriate reformulation of the problem when considering the first order optimality conditions, together with the introduction of a flow problem. Following the approach in [5], we define the tensor-valued function $\mathbf{p}^\varepsilon := \nabla \mathbf{u}^\varepsilon$. Problem (3.9) is equivalent to: Find $\mathbf{p}^\varepsilon \in \mathbf{Q}_4$ satisfying

$$j_\varepsilon(\mathbf{p}^\varepsilon) \leq j_\varepsilon(\mathbf{q}),\ \forall \mathbf{q} \in \mathbf{Q}_4, \tag{3.11}$$

where

$$\begin{aligned} j_\varepsilon(\mathbf{q}) =& \frac{1}{2}\int_\Omega |\mathbf{q}|^2 + \frac{C}{2}\int_\Omega |\mathbf{B}\mathbf{q} + \mathbf{u_g} - \mathbf{f}|^2 \,\mathrm{d}\mathbf{x} + I_\nabla(\mathbf{q}) \\ &+ \frac{1}{4\varepsilon_2}\int_\Omega \left[(|q_1|^2-1)^2 + (|q_2|^2-1)^2 + |q_1\cdot q_2|^2\right]\mathrm{d}\mathbf{x}, \end{aligned} \tag{3.12}$$

where $\mathbf{Q}_4 = (L^4(\Omega))^{3\times 2}$ and q_1, q_2 are the two 3×1 column vectors of $\mathbf{q}$. Here we define:

(i) the function $\mathbf{u_g} \in (H^1(\Omega))^3$ as the unique solution (harmonic extension) of

$$\begin{cases} \nabla^2 \mathbf{u_g} = \mathbf{0} & \text{in } \Omega, \\ \mathbf{u_g} = \mathbf{g} & \text{on } \partial\Omega. \end{cases}$$

(ii) the function $\mathbf{B}\mathbf{q}$ as the unique solution in $(H_0^1(\Omega))^3$ of

$$\nabla^2 \mathbf{B}\mathbf{q} = \nabla\cdot\mathbf{q} \text{ in } \Omega.$$

(iii) the functional $I_\nabla(\mathbf{q})$ as

$$I_\nabla(\mathbf{q}) = \begin{cases} \dfrac{\varepsilon_1}{2}\displaystyle\int_\Omega |\nabla\cdot\mathbf{q}|^2 d\mathbf{x} & \text{if } \mathbf{q} \in \nabla\mathbf{V_g}, \\ +\infty & \text{otherwise.} \end{cases}$$

Based on this *change of variables*, the first order optimality conditions (Euler-Lagrange equations) relative to (3.12) read as follows: Find $\mathbf{p}^\varepsilon \in \mathbf{Q}_4$ such that:

$$\begin{aligned} &\int_\Omega \mathbf{p}^\varepsilon : \mathbf{q}\mathrm{d}\mathbf{x} + C\int_\Omega (\mathbf{B}\mathbf{p}^\varepsilon + \mathbf{u_g} - \mathbf{f})\cdot\mathbf{B}\mathbf{q}\mathrm{d}\mathbf{x} + \langle \partial I_\nabla(\mathbf{p}^\varepsilon), \mathbf{q}\rangle \\ &+\frac{1}{\varepsilon_2}\int_\Omega \Big[(|p_1^\varepsilon|^2-1)p_1^\varepsilon\cdot q_1 + (|p_2^\varepsilon|^2-1)p_2^\varepsilon\cdot q_2 \\ &+\frac{1}{2}p_1^\varepsilon\cdot p_2^\varepsilon(p_2^\varepsilon\cdot q_1 + p_1^\varepsilon\cdot q_2)\Big]\mathrm{d}\mathbf{x} = 0, \quad \forall \mathbf{q} \in \mathbf{Q}_4. \end{aligned} \tag{3.13}$$

Here, $\partial I_\nabla(\cdot)$ denotes the subdifferential of the non-smooth proper lower semi-continuous (l.s.c.) convex functional I_∇. In the sequel, the superscript ε will be dropped for simplicity.

The solution method to solve (3.13) relies on the associated *flow problem*. Namely, we introduce a pseudo-time and we consider the initial-value evolutive problem to be integrated from $t=0$ to $t=+\infty$. Ultimately, the solution of (3.13) corresponds to the stationary solution of the flow problem. This initial value problem is defined as follows: Find $\mathbf{p}(t)\in\mathbf{Q}_4$ for a.e. $t\in(0,+\infty)$ satisfying

$$\begin{aligned}&\int_\Omega \frac{\partial \mathbf{p}(t)}{\partial t}:\mathbf{q}\,\mathrm{d}\mathbf{x}+\int_\Omega \mathbf{p}(t):\mathbf{q}\,\mathrm{d}\mathbf{x}+C\int_\Omega(\mathbf{B}\mathbf{p}(t)+\mathbf{u_g}-\mathbf{f})\cdot\mathbf{B}\mathbf{q}\,\mathrm{d}\mathbf{x}+\langle\partial I_\nabla(\mathbf{p}(t)),\mathbf{q}\rangle\\&+\frac{1}{\varepsilon_2}\int_\Omega\Big[(|p_1(t)|^2-1)p_1(t)\cdot q_1+(|p_2(t)|^2-1)p_2(t)\cdot q_2\\&+\frac{1}{2}p_1(t)\cdot p_2(t)(p_2(t)\cdot q_1+p_1(t)\cdot q_2)\Big]\,\mathrm{d}\mathbf{x}=0,\qquad\forall\mathbf{q}\in\mathbf{Q}_4,\end{aligned}\tag{3.14}$$

together with the initial condition $\mathbf{p}(0)=\mathbf{p}_0$ given. Following [5], the initial condition $\mathbf{p}_0=\nabla\mathbf{u}_0$, where $\mathbf{u}_0\in\mathbf{V_g}$ satisfies $-\nabla^2\mathbf{u}_0=\delta$ in Ω, where $\delta=(0.1,0.1,0.1)^T$ is a small right-hand side (quasi-harmonic extension).

We apply an operator-splitting strategy to solve (3.14) (namely, a first-order Marchuk-Yanenko scheme). Let $\Delta t>0$ be a constant given time step, $t^n=n\Delta t$, $n=1,2,\ldots$, to define the approximations $\mathbf{p}^n\simeq\mathbf{p}(t^n)$. Starting from the initial condition $\mathbf{p}^0=\mathbf{p}_0$, the Marchuk-Yanenko scheme allows, using $\mathbf{p}^n$ for all $n\geq 0$, to compute successively $\mathbf{p}^{n+1/2}$ and $\mathbf{p}^{n+1}$ using the two following intermediate steps:

(A) Prediction step (local optimization problem): Find $\mathbf{p}^{n+1/2}\in\mathbf{Q}_4$ satisfying

$$\begin{aligned}&\int_\Omega\frac{\mathbf{p}^{n+1/2}-\mathbf{p}^n}{\Delta t}:\mathbf{q}\,\mathrm{d}\mathbf{x}+\int_\Omega\mathbf{p}^{n+1/2}:\mathbf{q}\,\mathrm{d}\mathbf{x}\\&+\frac{1}{\varepsilon_2}\int_\Omega\Big[(|p_1^{n+1/2}|^2-1)p_1^{n+1/2}\cdot q_1+(|p_2^{n+1/2}|^2-1)p_2^{n+1/2}\cdot q_2\\&+\frac{1}{2}p_1^{n+1/2}\cdot p_2^{n+1/2}(p_2^{n+1/2}\cdot q_1+p_1^{n+1/2}\cdot q_2)\Big]\,\mathrm{d}\mathbf{x}=0,\end{aligned}\tag{3.15}$$

for all $\mathbf{q}\in\mathbf{Q}_4$.This problem does not involve any derivatives of the variable $\mathbf{p}$, which allows it to be solved locally (see Section 3.3).

(B) Correction step (variational problem): Find $\mathbf{p}^{n+1}\in\mathbf{Q}_4$ satisfying

$$\int_\Omega\frac{\mathbf{p}^{n+1}-\mathbf{p}^{n+1/2}}{\Delta t}:\mathbf{q}\,\mathrm{d}\mathbf{x}+C\int_\Omega(\mathbf{B}\mathbf{p}^{n+1}+\mathbf{u_g}-\mathbf{f})\cdot\mathbf{B}\mathbf{q}\,\mathrm{d}\mathbf{x}+\langle\partial I_\nabla(\mathbf{p}^{n+1}),\mathbf{q}\rangle=0,\tag{3.16}$$

for all $\mathbf{q}\in\mathbf{Q}_4$. This problem is actually an elliptic linear variational problem whose solution will be addressed in Section 3.4.

3.3. **Local Optimization Problems.** The sub-problem (3.15) that arises in the splitting algorithm does not involve any derivatives of the variable $\mathbf{p}^{n+1/2}$. Therefore, it can be solved

point-wise a.e. in Ω. Suppose that $\mathbf{p} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \\ p_{31} & p_{32} \end{bmatrix}$ and $\Delta t = \varepsilon_2/2$. Using the above notation, a more explicit formulation of (3.15) reads as:

$$\begin{aligned}
&(2+\varepsilon_2)\, p_{11}^{n+1/2} - 2\, p_{11}^{n} + \left(\mu^{n+1/2} - 1\right) p_{11}^{n+1/2} + \frac{\kappa^{n+1/2} p_{12}^{n+1/2}}{2} = 0,\\
&(2+\varepsilon_2)\, p_{21}^{n+1/2} - 2\, p_{21}^{n} + \left(\mu^{n+1/2} - 1\right) p_{21}^{n+1/2} + \frac{\kappa^{n+1/2} p_{22}^{n+1/2}}{2} = 0,\\
&(2+\varepsilon_2)\, p_{31}^{n+1/2} - 2\, p_{31}^{n} + \left(\mu^{n+1/2} - 1\right) p_{31}^{n+1/2} + \frac{\kappa^{n+1/2} p_{32}^{n+1/2}}{2} = 0,\\
&(2+\varepsilon_2)\, p_{12}^{n+1/2} - 2\, p_{12}^{n} + \left(\lambda^{n+1/2} - 1\right) p_{12}^{n+1/2} + \frac{\kappa^{n+1/2} p_{11}^{n+1/2}}{2} = 0,\\
&(2+\varepsilon_2)\, p_{22}^{n+1/2} - 2\, p_{22}^{n} + \left(\lambda^{n+1/2} - 1\right) p_{22}^{n+1/2} + \frac{\kappa^{n+1/2} p_{21}^{n+1/2}}{2} = 0,\\
&(2+\varepsilon_2)\, p_{32}^{n+1/2} - 2\, p_{32}^{n} + \left(\lambda^{n+1/2} - 1\right) p_{32}^{n+1/2} + \frac{\kappa^{n+1/2} p_{31}^{n+1/2}}{2} = 0,
\end{aligned} \tag{3.17}$$

where

$$\begin{aligned}
\mu^{n+1/2} &:= \left(\left(p_{11}^{n+1/2}\right)^2 + \left(p_{21}^{n+1/2}\right)^2 + \left(p_{31}^{n+1/2}\right)^2\right),\\
\lambda^{n+1/2} &:= \left(\left(p_{12}^{n+1/2}\right)^2 + \left(p_{22}^{n+1/2}\right)^2 + \left(p_{32}^{n+1/2}\right)^2\right),\\
\kappa^{n+1/2} &:= \left(p_{11}^{n+1/2} p_{12}^{n+1/2} + p_{21}^{n+1/2} p_{22}^{n+1/2} + p_{31}^{n+1/2} p_{32}^{n+1/2}\right).
\end{aligned}$$

System (3.17) can be reformulated in a more condensed form. Let us denote $\alpha := [p_{11}, p_{21}, p_{31}]^T$, and $\beta := [p_{12}, p_{22}, p_{32}]^T$; then (3.17) becomes

$$\begin{aligned}
&(2+\varepsilon_2)\alpha^{n+1/2} + \left(\mu^{n+1/2} - 1\right)\alpha^{n+1/2} + \frac{\kappa^{n+1/2}\beta^{n+1/2}}{2} = 2\alpha^n,\\
&(2+\varepsilon_2)\beta^{n+1/2} + \left(\lambda^{n+1/2} - 1\right)\beta^{n+1/2} + \frac{\kappa^{n+1/2}\alpha^{n+1/2}}{2} = 2\beta^n.
\end{aligned} \tag{3.18}$$

The above nonlinear system consists of six cubic equations. It is solved using a Newton-Raphson method. The initial guess is chosen as the value at the previous iteration, namely $\mathbf{p}^n$. Algorithm (3.15) (3.16) requires the condition $\Delta t \leq \varepsilon_2$ for system (3.18) to have a unique solution and guarantee the convergence as $n \to +\infty$, so here we took $\Delta t = \varepsilon_2/2$. Safeguarding is not needed in our numerical experiments.

In practice (see Section 4), once a finite element discretization of Ω is constructed, (3.18) is solved point-wise on each element of the discretization. The number of such nonlinear systems to solve thus depends on the number of elements of the triangulation.

3.4. **Variational Problems.** The sub-problem (3.16) that arises in the splitting algorithm is a well-posed, classical, linear elliptic variational problem. In order to highlight this statement, let us consider the reverse change of variable and take $\nabla \mathbf{u}^{n+1} := \mathbf{p}^{n+1}$. Problem (3.16) can be rewritten as follows : Find $\mathbf{u}^{n+1} \in \mathbf{V_g}$ such that

$$\begin{aligned} &\varepsilon_1 \Delta t \int_\Omega (\nabla^2 \mathbf{u}^{n+1}) \cdot (\nabla^2 \mathbf{v}) d\mathbf{x} + \int_\Omega \nabla \mathbf{u}^{n+1} : \nabla \mathbf{v} d\mathbf{x} + C\Delta t \int_\Omega \mathbf{u}^{n+1} \cdot \mathbf{v} d\mathbf{x} = \\ &C\Delta t \int_\Omega \mathbf{f} \cdot \mathbf{v} d\mathbf{x} + \int_\Omega \mathbf{p}^{n+1/2} : \nabla \mathbf{v} d\mathbf{x}, \quad \mathbf{v} \in (H^2(\Omega) \cap H_0^1(\Omega))^3. \end{aligned} \tag{3.19}$$

Problem (3.19) is a fourth-order linear elliptic variational problem of the biharmonic type. It can be solved directly, or via a coupled problem, introducing an auxiliary variable as in [5].

4. Mixed finite element discretization

4.1. **Generalities.** The solution of the time-stepping algorithm (3.15)-(3.16) is approximated with piecewise linear continuous finite elements. As mentioned earlier, the use of low-order finite elements is appropriate for problems such as (2.5), since the solution is piecewise smooth but presents a set of discontinuities along the folding lines anyway. Let us denote by $h > 0$ a space discretization step, together with an associated triangulation $\mathscr{T}_h$ that satisfies the usual compatibility conditions (see, e.g., [15]). From the triangulation $\mathscr{T}_h$, we define the following finite element spaces:

$$\begin{aligned} \mathbf{V}_h &= \{\mathbf{v} \in (C^0(\overline{\Omega}))^3, \mathbf{v}|_K \in (\mathbb{P}_1)^3, \forall K \in \mathscr{T}_h\}, \\ \mathbf{V}_{\mathbf{g},h} &= \{\mathbf{v} \in \mathbf{V}_h, \mathbf{v}(Q) = \mathbf{g}(Q), \ \forall Q \text{ vertex of } \mathscr{T}_h \text{ belonging to } \Gamma\}, \\ \mathbf{Q}_h &= \{\mathbf{q} \in (L^\infty(\Omega))^{3\times 2}, \mathbf{q}|_K \in \mathbb{R}^{3\times 2}, \forall K \in \mathscr{T}_h\}, \end{aligned}$$

where $\mathbb{P}_1$ is the space of two-variable polynomials of degree ≤ 1. Note that the gradient of functions in $\mathbf{V}_h$ belongs to $\mathbf{Q}_h$. Next, we equip $\mathbf{V}_h$, and its sub-space $\mathbf{V}_{\mathbf{g},h}$, with a discrete inner product (based on classical Gauss quadrature formulas):

$$(\mathbf{v}, \mathbf{w})_{0h} = \sum_{K \in \mathscr{T}_h} \sum_{i=1}^{m_k} W_i \mathbf{v}(\zeta_\mathbf{i}) \cdot \mathbf{w}(\zeta_i), \quad \forall \mathbf{v}, \mathbf{w} \in \mathbf{V}_h,$$

where W_i, resp. ζ_i are the weights, resp. evaluation points, of a Gauss quadrature rule of order ≥ 2, and m_k is the number of quadrature points in the element K (supposed constant). The quadrature formulas we used are implemented in the library `libmesh` [16]. The corresponding norm is $||\mathbf{v}||_{0h} := \sqrt{(\mathbf{v}, \mathbf{v})_{0h}}$, for all $\mathbf{v} \in \mathbf{V}_h$. In a similar fashion, we equip the space $\mathbf{Q}_h$ with the inner product and norm respectively defined by:

$$((\mathbf{p}, \mathbf{q}))_{0h} = \sum_{K \in \mathscr{T}_h} |K| \, \mathbf{p}|_K : \mathbf{q}|_K$$

and $|||\mathbf{q}|||_{0h} = \sqrt{((\mathbf{q}, \mathbf{q}))_{0h}}$ (with $|K|$ = area of K). The discrete version of the numerical algorithm uses the same steps as the continuous version presented in Section 3. However, let us sketch the main discrete milestones in the sequel.

4.2. **Discretization of the flow problem.** The discretized variational formulation of the initial value problem (3.14) reads as follows: Find $\mathbf{p}_h(t) \in \mathbf{Q}_h$ for a.e. $t \in (0,+\infty)$ satisfying

$$\begin{aligned}
&\int_\Omega \frac{\partial \mathbf{p}_h(t)}{\partial t} : \mathbf{q}_h d\mathbf{x} + \int_\Omega \mathbf{p}_h(t) : \mathbf{q}_h d\mathbf{x} + C\int_\Omega (\mathbf{B}\mathbf{p}_h(t) + \mathbf{u}_{\mathbf{g},h} - \mathbf{f}) \cdot \mathbf{B}\mathbf{q}_h d\mathbf{x} \\
&+ \langle \partial I_{\nabla_h}(\mathbf{p}_h(t)), \mathbf{q}_h \rangle \\
&+ \frac{1}{\varepsilon_2}\int_\Omega \Big[(|p_{1,h}(t)|^2 - 1) p_{1,h}(t) \cdot q_{1,h} + (|p_{2,h}(t)|^2 - 1) p_{2,h}(t) \cdot q_{2,h} \\
&+ \frac{1}{2} p_{1,h}(t) \cdot p_{2,h}(t) (p_{2,h}(t) \cdot q_{1,h} + p_{1,h}(t) \cdot q_{2,h}) \Big] d\mathbf{x} = 0, \qquad \forall \mathbf{q}_h \in \mathbf{Q}_h,
\end{aligned} \tag{4.1}$$

together with the initial condition $\mathbf{p}_h(0) = \mathbf{p}_{0,h}$ given. Here, we define

(i) the function $\mathbf{u}_{\mathbf{g},h} \in \mathbf{V}_{\mathbf{g},h}$ as the unique element of $\mathbf{V}_{\mathbf{g},h}$ verifying

$$\int_\Omega \nabla \mathbf{u}_{\mathbf{g},h} : \nabla \mathbf{v}_h d\mathbf{x} = 0, \quad \forall \mathbf{v}_h \in \mathbf{V}_{\mathbf{0},h}.$$

(ii) the function $\mathbf{B}\mathbf{q}_h$ as the unique element of $\mathbf{V}_{\mathbf{0},h}$ verifying

$$\int_\Omega \nabla \mathbf{B}\mathbf{q}_h : \nabla \mathbf{v}_h d\mathbf{x} = \int_\Omega (\nabla \cdot \mathbf{q}_h) \cdot \mathbf{v}_h d\mathbf{x}, \quad \forall \mathbf{v}_h \in \mathbf{V}_{\mathbf{0},h}.$$

(iii) the functional I_{∇_h} by

$$I_{\nabla_h}(\mathbf{q}_h) = \begin{cases} \dfrac{\varepsilon_1}{2}(\theta_h(\mathbf{q}_h), \theta_h(\mathbf{q}_h))_{0h} & \text{if } \mathbf{q}_h \in \nabla \mathbf{V}_{\mathbf{g},h}, \\ +\infty & \text{if } \mathbf{q}_h \in \mathbf{Q}_h \backslash \nabla \mathbf{V}_{\mathbf{g},h}. \end{cases}$$

where $\theta_h(\mathbf{q}_h)$ is uniquely defined from $\mathbf{q}_h$ by

$$\begin{cases} \theta_h(\mathbf{q}_h) \in \mathbf{V}_{\mathbf{0},h}, \\ (\theta_h(\mathbf{q}_h), \varphi)_{0h} = ((\mathbf{q}_h, \nabla\varphi))_{0h}, \quad \forall \varphi \in \mathbf{V}_{\mathbf{0},h}. \end{cases}$$

(iv) The initial condition $\mathbf{p}_{0,h} \in \mathbf{Q}_h$ is obtained as follows: first we calculate $\mathbf{u}_{0,h} \in \mathbf{V}_{\mathbf{g},h}$ verifying

$$\int_\Omega \nabla \mathbf{u}_{0,h} : \nabla \mathbf{v}_h d\mathbf{x} = \delta, \qquad \forall \in \mathbf{v}_h \in \mathbf{V}_{\mathbf{0},h},$$

where $\delta = (0.1, 0.1, 0.1)^T$; then, $\mathbf{p}_{0,h}$ is calculated as the gradient $\nabla \mathbf{u}_{0,h}$ that is piecewise constant on each element $K \in \mathscr{T}_h$.

We apply the operator-splitting strategy (3.15) (3.16) to solve (4.1), and we denote by $\mathbf{p}_h^n$ the related approximations of $\mathbf{p}_h(t^n)$. Starting from the initial condition $\mathbf{p}_h^0 = \mathbf{p}_{0,h}$, we compute successively $\mathbf{p}_h^{n+1/2}$ and $\mathbf{p}_h^{n+1}$ via the two following intermediate steps:

(A) Prediction step (local optimization problem): Find $\mathbf{p}_h^{n+1/2} \in \mathbf{Q}_h$ satisfying

$$\begin{aligned}
&\int_\Omega \frac{\mathbf{p}_h^{n+1/2}-\mathbf{p}_h^n}{\Delta t} : \mathbf{q}_h d\mathbf{x} + \int_\Omega \mathbf{p}_h^{n+1/2} : \mathbf{q}_h d\mathbf{x} \\
&+\frac{1}{\varepsilon_2}\int_\Omega \Big[(|p_{1,h}^{n+1/2}|^2-1)p_{1,h}^{n+1/2}\cdot q_{1,h} + (|p_{2,h}^{n+1/2}|^2-1)p_{2,h}^{n+1/2}\cdot q_{2,h} \\
&+\frac{1}{2}p_{1,h}^{n+1/2}\cdot p_{2,h}^{n+1/2}(p_{2,h}^{n+1/2}\cdot q_{1,h} + p_{1,h}^{n+1/2}\cdot q_{2,h})\Big] d\mathbf{x} = 0,
\end{aligned} \tag{4.2}$$

for all $\mathbf{q}_h \in \mathbf{Q}_h$.

(B) Correction step (variational problem): Find $\mathbf{p}_h^{n+1} \in \mathbf{Q}_h$ satisfying

$$\int_\Omega \frac{\mathbf{p}_h^{n+1}-\mathbf{p}_h^{n+1/2}}{\Delta t} : \mathbf{q}_h d\mathbf{x} + C\int_\Omega (\mathbf{B}\mathbf{p}_h^{n+1} + \mathbf{u}_{\mathbf{g},h} - \mathbf{f})\cdot \mathbf{B}\mathbf{q}_h \, d\mathbf{x} + \langle \partial I_{\nabla_h}(\mathbf{p}_h^{n+1}), \mathbf{q}_h\rangle = 0, \tag{4.3}$$

for all $\mathbf{q}_h \in \mathbf{Q}_h$.

4.3. **Solution of the discrete local optimization problems.** The finite dimensional nonlinear problem (4.2) can be solved *triangle-wise*; Indeed, if $\mathbf{p}_h^{n+1/2} := \{\mathbf{p}_{K,h}^{n+1/2}\}_{K\in\mathscr{T}_h}$, one can rewrite (4.2) as follows: For each triangle $K \in \mathscr{T}_h$, solve

$$\begin{aligned}
(2+\varepsilon_2)\alpha_{K,h}^{n+1/2} + \left(\mu_{K,h}^{n+1/2}-1\right)\alpha_{K,h}^{n+1/2} + \frac{\kappa_{K,h}^{n+1/2}\beta_{K,h}^{n+1/2}}{2} &= 2\alpha_{K,h}^n, \\
(2+\varepsilon_2)\beta_{K,h}^{n+1/2} + \left(\lambda_{K,h}^{n+1/2}-1\right)\beta_{K,h}^{n+1/2} + \frac{\kappa_{K,h}^{n+1/2}\alpha_{K,h}^{n+1/2}}{2} &= 2\beta_{K,h}^n,
\end{aligned} \tag{4.4}$$

where

$$\begin{aligned}
\mu_{K,h}^{n+1/2} &:= \left(\left(p_{K,h,11}^{n+1/2}\right)^2 + \left(p_{K,h,21}^{n+1/2}\right)^2 + \left(p_{K,h,31}^{n+1/2}\right)^2\right), \\
\lambda_{K,h}^{n+1/2} &:= \left(\left(p_{K,h,12}^{n+1/2}\right)^2 + \left(p_{K,h,22}^{n+1/2}\right)^2 + \left(p_{K,h,32}^{n+1/2}\right)^2\right), \\
\kappa_{K,h}^{n+1/2} &:= \left(p_{K,h,11}^{n+1/2}p_{K,h,12}^{n+1/2} + p_{K,h,21}^{n+1/2}p_{K,h,22}^{n+1/2} + p_{K,h,31}^{n+1/2}p_{K,h,32}^{n+1/2}\right), \\
\alpha_{K,h}^{n+1/2} &:= [p_{K,h,11}^{n+1/2}, p_{K,h,21}^{n+1/2}, p_{K,h,31}^{n+1/2}]^T, \\
\beta_{K,h}^{n+1/2} &:= [p_{K,h,12}^{n+1/2}, p_{K,h,22}^{n+1/2}, p_{K,h,32}^{n+1/2}]^T.
\end{aligned}$$

System (4.4) is similar to (3.18) and can be solved by Newton's techniques, taking $\mathbf{p}_{K,h}^n$ as an initial guess. When applied to the solution of problem (4.4), the Newton method always converged and never required more than 10 iterations for the test problems considered in Section 5.

4.4. **Solution of the discrete linear variational problems.** Taking $\nabla \mathbf{u}_h^{n+1} := \mathbf{p}_h^{n+1}$, (4.3) is equivalent to: Find $(\mathbf{u}_h^{n+1}, \mathbf{w}_h^{n+1}) \in \mathbf{V}_{\mathbf{g},h} \times \mathbf{V}_{\mathbf{0},h}$ such that

$$\begin{cases} \varepsilon_1 \Delta t((\nabla \mathbf{w}_h^{n+1}, \nabla \mathbf{v}_h))_{0h} + ((\nabla \mathbf{u}_h^{n+1}, \nabla \mathbf{v}_h))_{0h} + C\Delta t(\mathbf{u}_h^{n+1}, \mathbf{v}_h)_{0h} \\ = C\Delta t(\mathbf{f}, \mathbf{v}_h)_{0h} + ((\mathbf{p}_h^{n+1/2}, \nabla \mathbf{v}_h))_{0h}, \\ ((\nabla \mathbf{u}_h^{n+1}, \nabla \mathbf{q}_h))_{0h} - (\mathbf{w}_h^{n+1}, \mathbf{q}_h)_{0h} = 0, \end{cases} \tag{4.5}$$

for all $(\mathbf{v}_h, \mathbf{q}_h) \in \mathbf{V}_{\mathbf{0},h} \times \mathbf{V}_{\mathbf{0},h}$. This system is solved with a monolithic finite element solver.

5. Numerical experiments

We present here two numerical experiments for simple cases extracted from [11] to discuss the robustness and efficiency of our methodology. No adaptive mesh refinement strategy, such as the one introduced in [6], is adopted here.

The computational domain (i.e. the paper sheet to be folded) is chosen as the unit square $\Omega = (-1,1)^2$. All the experiments have been performed on a desktop computer with Intel Xeon E5-1650 (3.50 GHz $\times$ 12) and 64 GB memory. For all the numerical experiments we consider $\mathbf{f} = (0,0)^T$, $\varepsilon_2 = 10^{-15}$, $\Delta t = \varepsilon_2/2$, $C = 0$, and $\varepsilon_1 = \frac{h^2}{5\Delta t}$ (unless stated otherwise). The stopping criterion we use to decide on the flow stationarity is either if $n = 1500$, or if $||\mathbf{p}^{n+1} - \mathbf{p}^n||_{L^2(\Omega)} \leq 5 \cdot 10^{-4}$. The choice of a stringent stopping criterion ensures that the algorithm converges to a steady-state solution, and stationarity is verified numerically for constant time steps. The penalization constant ε_2 is chosen in order to guarantee that the orthogonality conditions are satisfied accurately. To do so, its value can actually vary between 10^{-9} and 10^{-15}. For all the experiments we use a structured asymmetric mesh when cutting small regular squares in two triangles along the first diagonal. The choice of ε_1 allows to have a regularization term in (3.19) of the order h^2. The value $\varepsilon_1 = \frac{h^2}{5\Delta t}$ allows to facilitate the convergence of the algorithm, while the value $\varepsilon_1 = 0$ allows to ensure sharp interfaces. Any value of the parameter ε_1 is a trade-off between these two objectives.

5.1. **90-degree folding.**

In the first experiment, we consider the exact solution that consists of folding once a paper at a 90-degree angle, namely

$$\mathbf{u}(x_1, x_2) = \begin{pmatrix} x_1 \\ x_2/\sqrt{2} \\ |x_2|/\sqrt{2} \end{pmatrix}, \quad \forall (x_1, x_2) \in \Omega. \tag{5.1}$$

In this case, the solution is piecewise linear with one folding line. Figure 2 (top left, top right, and bottom left) shows the representation of the three components of the solution $\mathbf{u}$, while Figure 2 (bottom right) shows the 3D representation of the folded paper which correspond to $\mathbf{u}(\Omega)$. It shows that the solution is appropriately recovered, but that there is an approximation error along the folding line, which is not perfectly straight. Table 1 illustrates the numerical results for several discretization parameters' sizes. It shows that all orthogonality constraints are very well respected, but that the convergence of the error in the L^2 norm is not reached. The algorithm here converges to a slightly different solution that is smoother, despite the fact that $\varepsilon_1 = 0$. Unlike for orthogonal maps [5], the convergence of the algorithm is difficult to guarantee.

We also observe that the smoothing term improves the convergence of the algorithm (reduces the number of iterations) without sacrificing the orthogonality constraints. But it has an effect on the magnitude of the L^2-error.

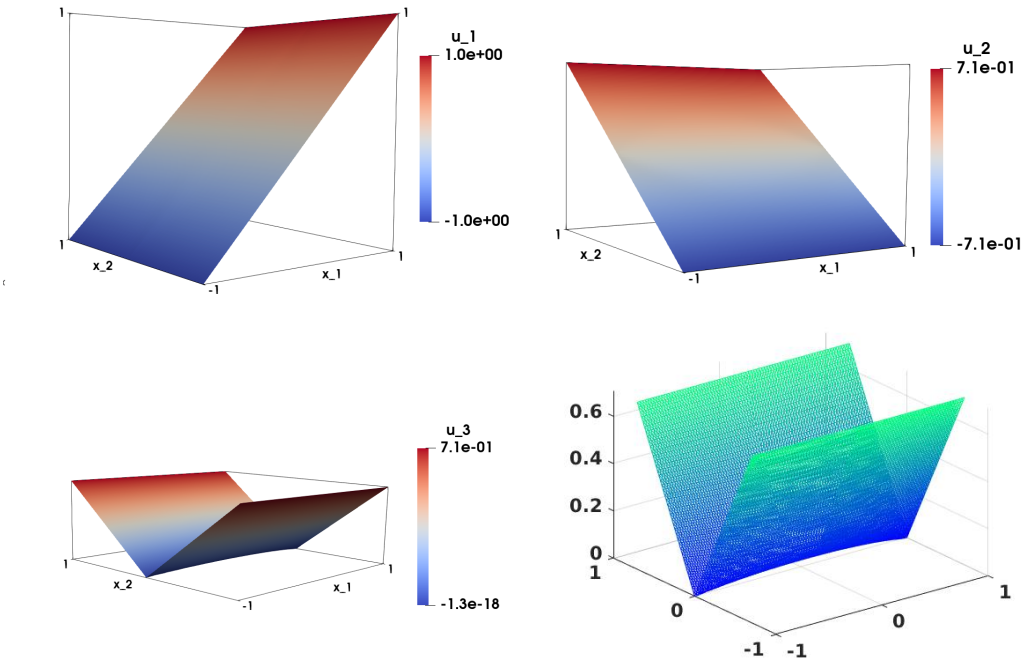

FIGURE 2. 90-degree folding. Snapshots of the approximation of three components of the solution $\mathbf{u}$ for $h = 0.025$, $\varepsilon_1 = 0$ and 1500 timesteps. Top left: component u_1, top right: component u_2, bottom left: component u_3, bottom right: visualization of the domain $\mathbf{u}(\Omega)$.

5.2. **Non piecewise linear example.** In the second experiment, we consider the exact solution that consists of folding once a paper while curving it, namely

$$\mathbf{u}(x_1,x_2) = \begin{pmatrix} x_1 \\ \text{sign}(x_2)(1-\cos(x_2)) \\ \sin(|x_2|) \end{pmatrix}, \quad \forall (x_1,x_2) \in \Omega.$$

In this case, the solution is not piecewise linear anymore, but piecewise smooth with one folding line. The gradient of the solution is still orthonormal. Figure 3 (bottom right) shows the 3D representation of the folded paper which correspond to the image $\mathbf{u}(\Omega)$. Figure 3 (top left, top right, and bottom left) shows the representation of the three components of the solution $\mathbf{u}$. In particular, it shows that the component u_2 introduces the bending of the paper. Table 2 compares well with Table 1, and shows that the behavior of the algorithm is similar when the solution is not piecewise linear. It shows that all constraints are well respected, but that the L^2 norm is stagnating again. The smoothing term has the same effect on the convergence of the algorithm (and on the number of iterations).

TABLE 1. 90-degree folding solution. (i) Variations with respect to h of the approximate orthogonality conditions (columns 2,3 and 4). (ii) Variations with respect to h of the $L^2(\Omega)$ norm of the computed approximation error $\mathbf{u}-\mathbf{u}_h$ (column 5). (iii) Variations with respect to h of the number of time steps necessary to achieve convergence (column 6). ($\Omega=(-1,1)^2$, structured asymmetric meshes).

Regularization term: $\varepsilon_1 = 0.2h^2/\Delta t$					
h	$\int_\Omega \mu_h^{n+1/2} d\mathbf{x}$	$\int_\Omega \lambda_h^{n+1/2} d\mathbf{x}$	$\int_\Omega \kappa_h^{n+1/2} d\mathbf{x}$	$\|\|(\mathbf{u}-\mathbf{u}_h)\|\|_{L^2}$	it
0.1	1.0093	0.9590	-9.48e-04	6.74e-02	163
0.05	1.0097	0.9880	-3.61e-04	7.18e-02	397
0.025	1.0084	0.9969	-1.28e-04	6.79e-02	880
0.0125	1.0060	0.9993	-4.37e-05	5.83e-02	1398
Regularization term: $\varepsilon_1 = 0.0$					
h	$\int_\Omega \mu_h^{n+1/2} d\mathbf{x}$	$\int_\Omega \lambda_h^{n+1/2} d\mathbf{x}$	$\int_\Omega \kappa_h^{n+1/2} d\mathbf{x}$	$\|\|(\mathbf{u}-\mathbf{u}_h)\|\|_{L^2}$	it
0.1	1.0025	1.0004	-2.64e-04	4.11e-02	1500
0.05	1.0028	1.0003	-1.07e-04	4.28e-02	1500
0.025	1.0028	1.0003	-3.74e-05	4.32e-02	1500
0.0125	1.0029	1.0002	-1.32e-05	4.33e-02	1500

TABLE 2. Non piecewise linear example. (i) Variations with respect to h of the approximate orthogonality conditions (columns 2,3 and 4). (ii) Variations with respect to h of the $L^2(\Omega)$ norm of the computed approximation error $\mathbf{u}-\mathbf{u}_h$ (column 5). (iii) Variations with respect to h of the number of time steps necessary to achieve convergence (column 6). ($\Omega=(-1,1)^2$, structured asymmetric meshes).

Regularization term: $\varepsilon_1 = 0.2h^2/\Delta t$					
h	$\int_\Omega \mu^{n+1/2} d\mathbf{x}$	$\int_\Omega \lambda^{n+1/2} d\mathbf{x}$	$\int_\Omega \kappa^{n+1/2} d\mathbf{x}$	$\|\|(\mathbf{u}-\mathbf{u}_h)\|\|_{L^2}$	it
0.1	1.0590	0.9114	-2.69e-03	1.34e-01	149
0.05	1.0511	0.9506	-8.09e-04	1.35e-01	388
0.025	1.0242	0.9769	-2.36e-04	8.10e-02	1367
0.0125	1.0129	0.9931	-7.37e-05	3.86e-02	1472
Regularization term: $\varepsilon_1 = 0.0$					
h	$\int_\Omega \mu^{n+1/2} d\mathbf{x}$	$\int_\Omega \lambda^{n+1/2} d\mathbf{x}$	$\int_\Omega \kappa^{n+1/2} d\mathbf{x}$	$\|\|(\mathbf{u}-\mathbf{u}_h)\|\|_{L^2}$	it
0.1	1.0040	1.0	-2.89e-04	2.11e-02	1500
0.05	1.0042	1.0	-1.17e-04	2.18e-02	1500
0.025	1.0042	1.0	-6.77e-05	2.20e-02	1500
0.0125	1.0042	1.0	-3.81e-05	2.21e-02	1500

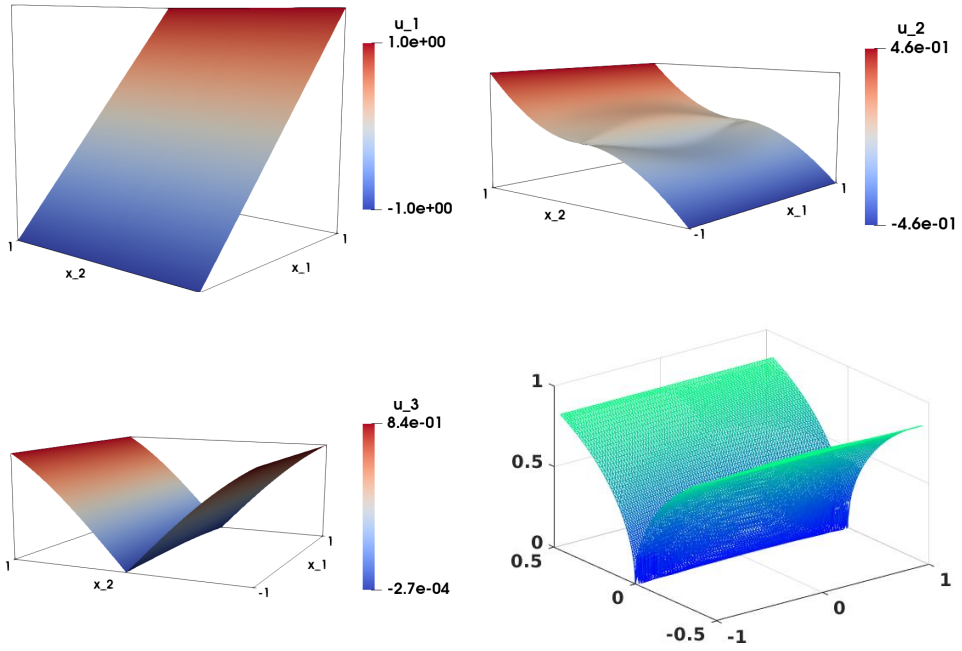

FIGURE 3. Non piecewise linear example. Snapshots of the approximation of three components of the solution $\mathbf{u}$ for $h = 0.025$, $\varepsilon_1 = 0$ and 1500 timesteps. Top left: component u_1, top right: component u_2, bottom left: component u_3, bottom right: visualization of the domain $\mathbf{u}(\Omega)$.

6. CONCLUSIONS

We discussed an operator-splitting/ finite element methodology for the numerical solution of the Dirichlet problem for so-called rigids maps. The approach is based on an extension of similar ideas for orthogonal maps [5], but now allows to consider functions with a range in $\mathbb{R}^3$, which are not piecewise linear but only piecewise smooth. This methodology is based on a variational principle, the introduction of the associated flow problem, and a time-stepping splitting algorithm. Numerical experiments in Section 5 demonstrate the robustness and the flexibility of this methodology. In particular the orthogonality constraints converge appropriately. However, the convergence of the error is not optimal, as the algorithm finds local optima. Adaptive mesh refinement will be needed in the future to accurately track the folding lines.

Acknowledgements
This work was partially funded by the Swiss National Science Foundation (Grant number 165785).

REFERENCES

[1] Z. Abel, E. D. Demaine, M. L. Demaine, D. Eppstein, A. Lubiw, R. Uehara, Flat foldings of plane graphs with prescribed angles and edge lengths, Journal of Computational Geometry 9 (2018) 74–93.
[2] B. M. Afkham, A. Bhatt, B. Haasdonk, J. S. Hesthaven, Symplectic model-reduction with a weighted inner product, arXiv preprint arXiv:1803.07799, 2018.

[3] A. Bonito, D. Guignard, R. H. Nochetto, S. Yang, LDG approximation of large deformations of prestrained plates, Journal of Computational Physics 448 (2022) 110719.
[4] A. Caboussat, R. Glowinski, A penalty-regularization-operator splitting method for the numerical solution of a scalar Eikonal equation, Chinese Annals of Mathematics Series B 36 (2015) 659–688.
[5] A. Caboussat, R. Glowinski, D. Gourzoulidis, M. Picasso, Numerical approximation of orthogonal maps, SIAM Journal on Scientific Computing, 41 (2019) B1341–B1367.
[6] A. Caboussat, D. Gourzoulidis, M. Picasso, An anisotropic adaptive method for the numerical approximation of orthogonal maps, Journal of Computational and Applied Mathematics 407 (2022) 113997.
[7] B. Dacorogna, P. Marcellini, Implicit Partial Differential Equations, Birkhaüser, Basel, 1999.
[8] B. Dacorogna, P. Marcellini, E. Paolini, An explicit solution to a system of implicit differential equations, Annales de l'Institut Henri Poincare (C) Non Linear Analysis, 25 (2008) 163–171.
[9] B. Dacorogna, P. Marcellini, E. Paolini, Lipschitz-continuous local isometric immersions: rigid maps and origami, Journal de Mathématiques Pures et Appliquées 90 (2008) 66–81.
[10] B. Dacorogna, P. Marcellini, E. Paolini, Functions with orthogonal Hessian, Differential and Integral Equations 23 (2010) 51–60.
[11] B. Dacorogna, P. Marcellini, E. Paolini, Origami and partial differential equations, Notices of the American Mathematical Society, 57 (2010) 598–606.
[12] B. Dacorogna, P. Marcellini, E. Paolini, On the n-dimensional Dirichlet problem for isometric maps, Journal of Functional Analysis 255 (2018) 3274–3280.
[13] E. D. Demaine, T. Tachi, Origamizer: A practical algorithm for folding any polyhedron, In: 33rd International Symposium on Computational Geometry, SoCG 2017, pp. 1–34, Brisbane, Australia, 2017.
[14] L. H. Dudte, E. Vouga, T. Tachi, L. Mahadevan, Programming curvature using origami tessellations, Nature Materials, 15 (2016) 583–588.
[15] R. Glowinski, Numerical Methods for Nonlinear Variational Problems, Springer-Verlag, New York, NY, second edition, 2008.
[16] B. S. Kirk, J. W. Peterson, R. H. Stogner, G. F. Carey, `libMesh`: A c++ library for parallel adaptive mesh refinement/coarsening simulations, Engineering with Computers 22 (2006) 237–254.