Geometric shape optimization for Dirichlet energy with physics informed and symplectic neural networks

Congrès d'Analyse Numérique 2024

A. Bélières–Frendo[1,2] E. Franck[1,2,3]
V. Michel-Dansac[1,2,3] Y. Privat[2,3,4]

IRMA, Université de Strasbourg[1] - NumPEx[2] - INRIA, MACARON team[3]
Institut Élie Cartan de Lorraine, Univ. Lorraine[4] - INRIA, SPHINX team[5]

May 28th 2024 - Île de Ré

Context

Lord Rayleigh, Kac

Links between geometry and eigenmodes

Pierre Henrot

Regularity of optimal shapes

Existence results for optimal Shapes

Allaire

Pironneau

And many other authors ...

Numerical Methods

Dassault System

Renault

Eiffage

Industrial Design

Very fundamental → Applied → Engineering

Objectives

Limitations of existing shape derivative methods

- Derivative calculation can be highly complex (multiphysics model)
- determination of an adjoint problem and a descent step: very costly in terms of computation time and memory allocation
- highly local approaches
- not "well-posed" for all physical models (turbulent Navier-Stokes equations)

Discover neural networks algorithms

- Automatic Differentiation (AD) avoids truncation errors
- Monte-Carlo integration for parametric problems and very complex topologies
- joint gradient descent on several mutually dependent networks
- GPU computation, highly parallelized/vectorized
- Asymptotic convexity

Objective: Proof of concept on a very simple model, the Dirichlet energy minimization

Underlying PDE

$$(\mathcal{P}) : \begin{cases} -\Delta u = f & \text{in } \Omega; \\ u = 0 & \text{on } \partial\Omega. \end{cases} \underset{\text{if } f \text{ is regular enough}}{\Leftrightarrow} (\mathcal{P}_{\mathcal{O}}) : \inf_{v \in H_0^1(\Omega)} \int_\Omega |\nabla v|^2 - fv$$

"Natural" energy:

$$\mathcal{E}(\Omega) = \int_\Omega |\nabla u|^2 - fu$$

Minimization problem:

$$(\mathcal{D}) : \quad \inf\{\mathcal{E}(\Omega),\, \Omega \text{ bounded open set of } \mathbb{R}^n,\ \text{such that } |\Omega| \leq V_0\}.$$

Mathematical results[1]

Existence and regularity

There exists an optimal shape. Studying the regularity of the solutions is not an easy issue, and not central in this work.

First order optimality condition

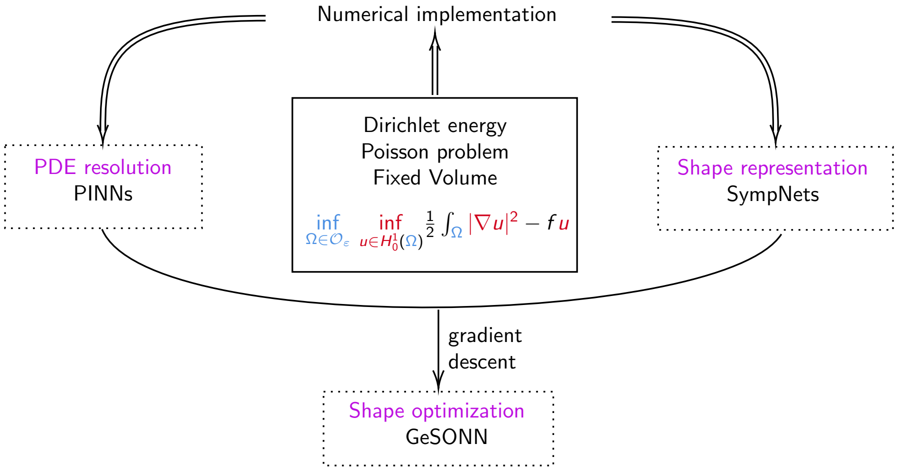
Let $f \in L^2_{loc}(\mathbb{R}^n)$. If Ω is a solution to $(\mathcal{D})$ with a C^2 boundary, then there exists $c > 0$ such that

$$|\nabla u_\Omega^f| = c \text{ on } \partial\Omega.$$

[1] A. Henrot and M. Pierre, *Variation et optimisation de formes: Une analyse géométrique*. Springer Berlin Heidelberg, 2005.

Overview

1. SympNets
2. PINNs
3. GeSONN



How to represent a shape?

Existing methods

- Boundary nodes of a mesh
- density and micro-structure
- level-set function
- porous materials
- **differentiable map**

Symplectic maps

- Symplectic maps are differentiable maps that preserves a symplectic structure
- algebraïc interpetation: $< ., J. >$ is preserved with $J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$
- physical idea: the flow of an Hamiltonian ODE is a symplectic map
- symplectic maps preserve the volume
- **the symplectic form is the volume form in** $\mathbb{R}^2$

Some usefull properties

Shear maps

One of the simplest families of symplectic transformations from $\mathbb{R}^{2d}$ into $\mathbb{R}^{2d}$ is called "shear maps", and is defined as follows

$$f_{\mathrm{up}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 + \nabla V_{\mathrm{up}}(x_2) \\ x_2 \end{pmatrix} \quad \text{and} \quad f_{\mathrm{down}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 + \nabla V_{\mathrm{down}}(x_1) \end{pmatrix}$$

where $V_{\mathrm{up/down}} \in C^1(\mathbb{R}^d, \mathbb{R})$, and $\nabla V : \mathbb{R}^d \to \mathbb{R}^d$ is the gradient of V.

Lemma

Any symplectic map can be approximated by the composition of several shear maps - the composition of several symplectic maps still remains symplectic.

SympNets[2]

Theorem

Let $q > 0$ be the depth of the NN. In practice, we set $q > 2d$. We define $\widehat{\sigma_{K,a,b}}$ the approximation of ∇V in terms of an activation function $\sigma : \mathbb{R} \to \mathbb{R}$, two vectors $a, b \in \mathbb{R}^q$, a matrix $K \in \mathcal{M}_{q,n}(\mathbb{R})$, and $\mathrm{diag}(a) = (a_i \delta_{ij})_{1 \leq i,j \leq q}$, as follows

$$\widehat{\sigma_{K,a,b}}(x) = K^t \mathrm{diag}(a) \sigma(Kx + b),$$

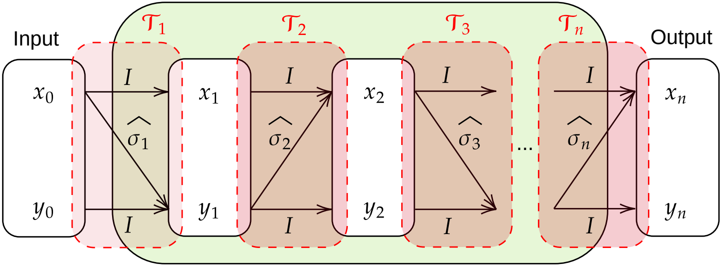
Then, gradient modules $\mathcal{G}_{\text{up}}$ and $\mathcal{G}_{\text{down}}$ are defined to approximate f_{up} and f_{down}, by

$$\mathcal{G}_{\text{up}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 + \widehat{\sigma_{K,a,b}}(x_2) \\ x_2 \end{pmatrix} \quad \text{and} \quad \mathcal{G}_{\text{down}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 + \widehat{\sigma_{K,a,b}}(x_1) \end{pmatrix}.$$

[2] P. Jin, Z. Zhang, A. Zhu, *et al.*, "SympNets: Intrinsic structure-preserving symplectic networks for identifying Hamiltonian systems," *Neural Networks*, vol. 132, pp. 166–179, 2020.

G-SympNets

A symplectic network



Loss function to learn a given symplectic map $\mathcal{T}$

$$\mathcal{J}_S(\omega; \{x_i\}_{i=1}^N) = \sum_{i=1}^{N} |T_\omega(x_i) - \mathcal{T}(x_i)|^2,$$

And for parametric problems?

$\widehat{\sigma_{K,a,b}}(x) = K^t \mathrm{diag}(a)\sigma(Kx + b)$ becomes:

$$\widetilde{\sigma}_{K,K_\mu,a,b}(x; \mu) = K^t \sigma(Kx + b + K_\mu \mu).$$

- n_μ number of parameters
- $\mu \in \mathbb{M} \subset \mathbb{R}^{n_\mu}$
- $K_\mu \in \mathcal{M}_{q,n_\mu}(\mathbb{R})$

One can show that $\widetilde{\sigma}_{K,K_\mu,a,b}$ is a gradient module, and that the whole network remains symplectic with respect to $x \in \mathbb{R}^{2d}$, for each parameter $\mu \in \mathbb{M}$.

Learning a simply connected parameterized shape with a SympNet

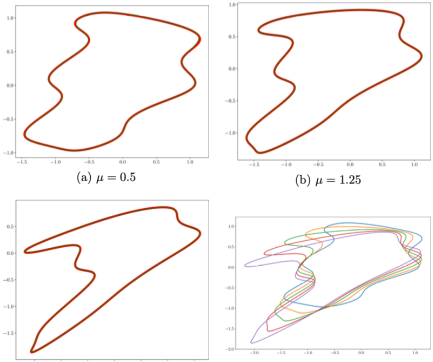
We introduce the symplectic map $\mathcal{T}_\mu = \mathcal{S}^1_\mu \circ \mathcal{S}^2_\mu$, with

$$\begin{cases} \mathcal{S}^1_\mu : (x_1, x_2; \mu) \mapsto (x_1 - \mu x_2^2 + 0.3 \sin(\frac{x_2}{\mu}) - 0.2 \sin(8x_2), \quad x_2), \\ \mathcal{S}^2_\mu : (x_1, x_2; \mu) \mapsto (x_1, \quad x_2 + 0.2\mu x_1 + 0.12 \cos(x_1)). \end{cases}$$

Numerical simulation parameters

- Number of networks: 4;
- width of networks: 10;
- learning rate: 10^{-2};
- collocation points: 10^4;
- epochs: 10^3;
- $\mu \in (0.5, 2)$.

Learning a parametric family of symplectic maps



(a) $\mu = 0.5$ (b) $\mu = 1.25$ (c) $\mu = 2$ (d) $\mu \in \{0.5, 0.875, 1.25, 1.625, 2\}$

Figure: Learning the symplectic map $\mathcal{T}_\mu = \mathcal{S}^1_\mu \circ \mathcal{S}^2_\mu$, with

$$\mathcal{S}^1_\mu : (x_1, x_2; \mu) \mapsto \left(x_1 - \mu x_2^2 + 0.3\sin\left(\frac{x_2}{\mu}\right) - 0.2\sin(8x_2), \quad x_2\right)$$

$$\mathcal{S}^2_\mu : (x_1, x_2; \mu) \mapsto \left(x_1, \quad x_2 + 0.2\mu x_1 + 0.12\cos(x_1)\right).$$

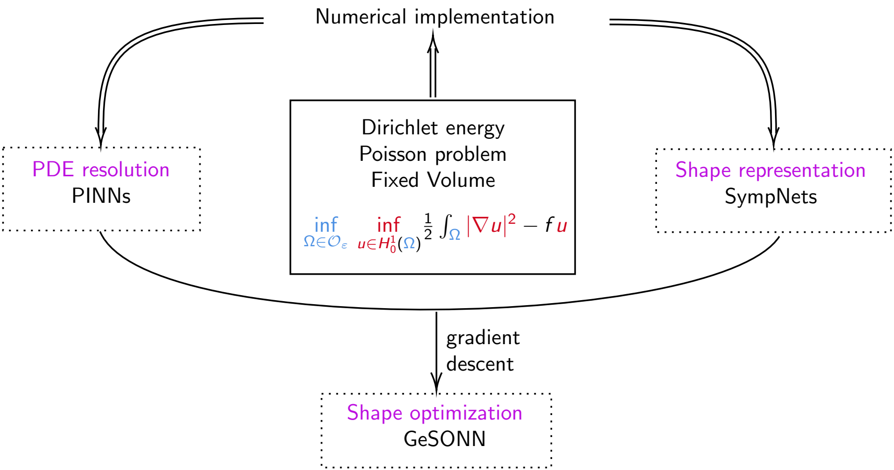
Hausdorff distance is computed
randomly sample 1000 points in $\mathbb{M} = [0.5, 2]$

Table: Statistics on the Hausdorff distance between the reference shape and the learned shape, on the parameter set $\mathbb{M} = [0.5, 2]$.

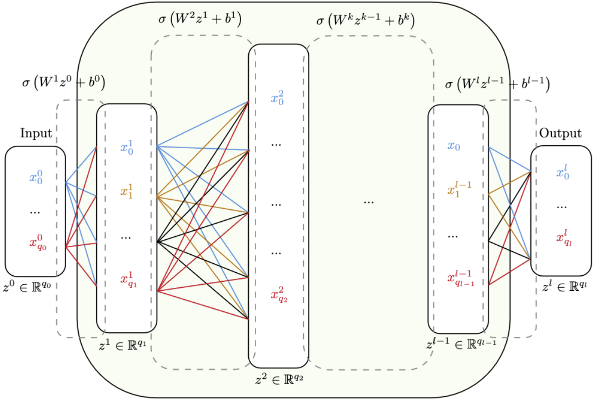
Mean value	Maximal value	Minimal value	Variance
9.45×10^{-3}	2.10×10^{-2}	6.05×10^{-3}	4.50×10^{-6}

Overview

1. SympNets
2. PINNs
3. GeSONN



Principle of the algorithm[3]



DeepRitz: $\mathcal{J}(u) = \dfrac{1}{2}\int_\Omega |\nabla u|^2 - fu$ or PINNs: $\mathcal{R}(u) = \|\Delta u + f\|^2_{L^2(\Omega)}$

[3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019

Network tuning

Boundary conditions

- Network: u_θ
- solution: $v_\theta = u_\theta \beta + \alpha$
- β s.t. $\gamma^0_{\partial\Omega}\beta = 0$
- α s.t. $\gamma^0_{\partial\Omega}\alpha$ is equal to the boundary condition of $(\mathcal{P})$

Example for $(\mathcal{P})$ in the unit disk: $\alpha(x, y) = 0$ and $\beta(x, y) = 1 - x^2 - y^2$.

Loss function

$$\mathcal{J}_N\left(\theta; \{x_i\}_{i=1}^N\right) = \frac{V_0}{N}\sum_{i=1}^N \left\{\frac{1}{2}|\nabla v_\theta(x_i)|^2 - f(x_i)v_\theta(x_i)\right\},$$

where $\{x_i\}_{i=1}^N \in \Omega^N$ are N collocation points.

Remark: $\mathcal{J}_N\left(\theta; \{x_i\}_{i=1}^N\right) = \mathcal{J}(u_\theta, \Omega) + \mathcal{O}(N^{-1/2})$.

Resolution of a parametric family of PDEs

Setting of the parametric problem

- n_μ: number of parameters μ in the problem
- $\mathbb{M} \subset \mathbb{R}^{n_\mu}$: the space of parameters
- $u_\theta^p : \Omega \times \mathbb{M} \to \mathbb{R}$ the parametric neural network
- $f^p : \Omega \times \mathbb{M} \to \mathbb{R}$

$$(\mathcal{P}^p) : \begin{cases} -\Delta u^p(x;\mu) = f^p(x;\mu), & \text{for } (x;\mu) \in \Omega \times \mathbb{M}; \\ u^p(x;\mu) = 0, & \text{for } (x;\mu) \in \partial\Omega \times \mathbb{M}. \end{cases}$$

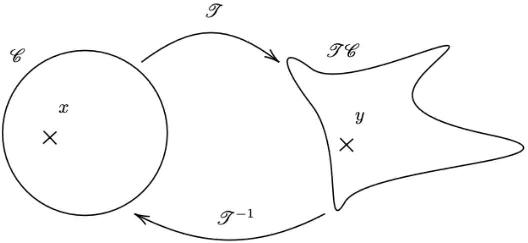
Example: $\mathbb{M} = \mathbb{R} \times \mathbb{R}$, $f^p : (x = (x_1, x_2); \mu = (\mu_1, \mu_2)) \mapsto \exp(1 - \left(\frac{x_1}{\mu_1}\right)^2 - \left(\frac{x_2}{\mu_2}\right)^2)$.

Loss function

$$\mathcal{J}_N^p \left(\theta; \{x_i, \mu_i\}_{i=1}^N\right) = \frac{V_0}{N} \sum_{i=1}^N \left\{ \frac{1}{2} |\nabla v_\theta^p(x_i;\mu_i)|^2 - f^p(x_i;\mu_i) v_\theta^p(x_i;\mu_i) \right\}.$$

How to manage the boundary conditions in complex geometries?

Domain generated by a symplectic transformation of the unit disk of $\mathbb{R}^2$



$$\begin{aligned} -\Delta u(x) &= f(y), \quad x \in \mathscr{C} \\ u(x) &= 0, \quad\quad x \in \partial\mathscr{C} \end{aligned}$$

$$\begin{aligned} -\Delta u_{\mathscr{T}}(y) &= f(y), \quad y \in \mathscr{T}\mathscr{C} \\ u_{\mathscr{T}}(y) &= 0, \quad\quad y \in \partial\mathscr{T}\mathscr{C} \end{aligned}$$

we introduce $w : \mathcal{C} \to \mathbb{R}$, defined for a.e. $x \in \Omega$ by

$$w(x) = (u_{\mathcal{T}} \circ \mathcal{T})(x)$$

New equation[4]

PDE

$$\begin{cases} -\mathrm{div}\big(A\nabla w\big) = f\circ\mathcal{T}, & \text{in } \mathcal{C}; \\ w = 0, & \text{on } \mathcal{C}, \end{cases}$$

with $A : \mathcal{C} \to \mathbb{R}$ a uniformly elliptic metric tensor, defined by

$$A = J_{\mathcal{T}}^{-1} \cdot J_{\mathcal{T}}^{-t},$$

Optimization problem

$$\inf \left\{ \frac{1}{2}\int_{\mathcal{C}} A\nabla w \cdot \nabla w - \int_{\mathcal{C}} \widetilde{f} v, \quad \exists u_{\mathcal{T}} \in H_0^1(\mathcal{T}\mathcal{C}), \quad w = u_{\mathcal{T}} \circ \mathcal{T} \in H_0^1(\mathcal{C}) \right\}$$

[4] A. Belieres Frendo, E. Franck, V. Michel Dansac, *et al.*, "Learning-based shape optimisation," *To be published soon...*,

Numerical experiments

Equation to solve

Poisson problem, with a parametric source term f, given by

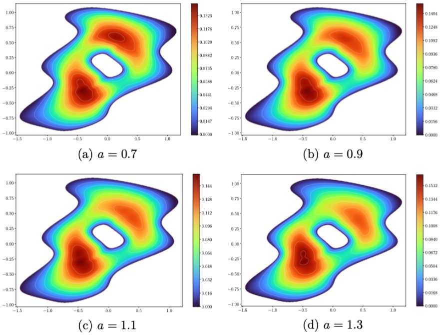
$$f(x, y; a) = \exp\left(1 - \left(\frac{x}{a}\right)^2 - (ay)^2\right), \quad a \in (0.5, 1.5), \tag{1}$$

$\mathcal{T}_{\mu=0.5}(\mathcal{B}(1,0)\backslash\mathcal{B}(0.2,0))$ is the computational domain.

Network parameters

- layer sizes: 3, 20, 40, 40, 20, 1
- learning rate: 2×10^{-3}
- activation: tanh
- collocation points: 20000
- epochs: 1500

Poisson equation in a patatoïd with parametric source term

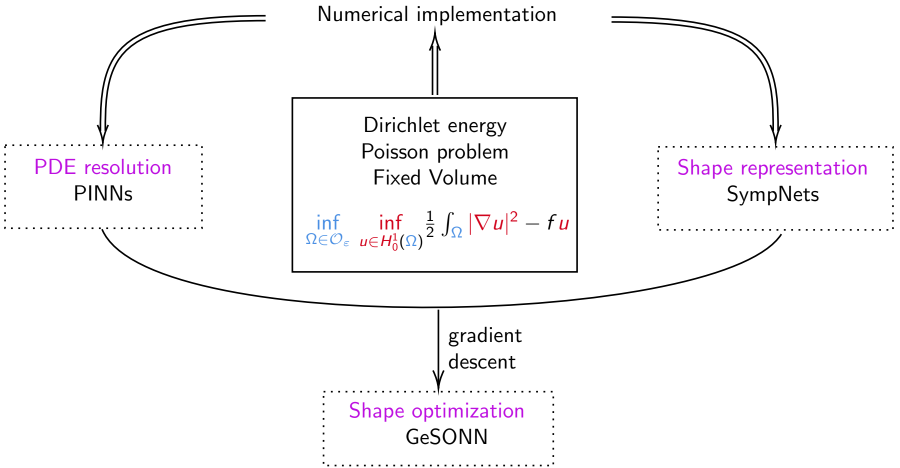


(a) $a = 0.7$ (b) $a = 0.9$ (c) $a = 1.1$ (d) $a = 1.3$

Table: Statistics on $\left| \int_\Omega A \nabla u \nabla \varphi - f \varphi \right|$ for 10^3 quasi-random test functions φ

Mean value	Maximal value	Minimal value	Variance
8.43×10^{-3}	4.53×10^{-2}	1.10×10^{-5}	5.01×10^{-5}

Overview



Numerical implementation
PDE resolution
PINNs
Dirichlet energy
Poisson problem
Fixed Volume
$\inf_{\Omega \in \mathcal{O}_\varepsilon} \inf_{u \in H_0^1(\Omega)} \frac{1}{2} \int_\Omega |\nabla u|^2 - f u$
Shape representation
SympNets
gradient
descent
Shape optimization
GeSONN

Two neural networks, but a single loss function[5]

The Dirichlet energy as a loss function

$$\mathcal{J}_{P/S}\left(\theta, \omega; \{x_i\}^N\right) = \frac{V_0}{N} \sum_{i=1}^{N} \left\{ \frac{1}{2} \left| A_\omega \nabla v_{\theta,\omega} \cdot \nabla v_{\theta,\omega} \right|^2 - \widetilde{f}_\omega v_{\theta,\omega} \right\} (x_i)$$

- θ the trainable weights of the PINN, ω the trainable weights of the SympNet;
- $v_{\theta,\omega} : \mathcal{C} \to \mathbb{R}; x \mapsto \beta(x) u_\theta(T_\omega x) + \alpha(x)$ the solution of the Poisson problem set in $T_\omega \mathcal{C}$;
- $T_\omega : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ the SympNet;
- $u_\theta : T_\omega \mathcal{C} \to \mathbb{R}$ the PINN;

[5] A. Belieres Frendo, E. Franck, V. Michel Dansac, *et al.*, "Learning-based shape optimisation," *To be published soon...*,

Numerical experiment on an analytical solution, $f = 1$

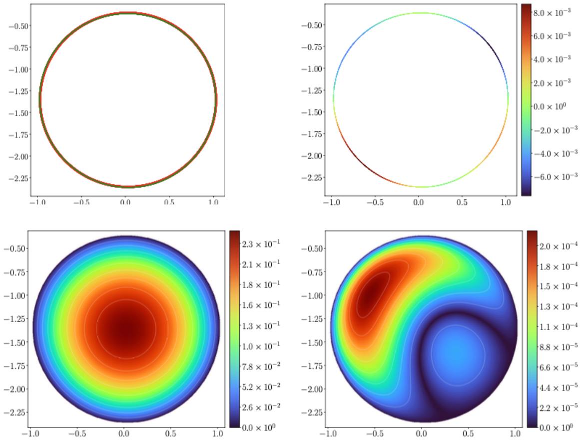


Figure: Shape optimization, $f = 1$. Top left: learned shape (green line) and reference shape (red line). Top right: learned optimality condition. Bottom left: learned PDE solution. Bottom right: pointwise error.

Numerical experiment on an analytical solution, $f = 1$

Relevant metrics related to the shape optimization of the Poisson equation with $f = 1$.

Hausdorff distance	variance of the optimality condition	L^2 error
3.09×10^{-2}	1.88×10^{-5}	9.15×10^{-5}

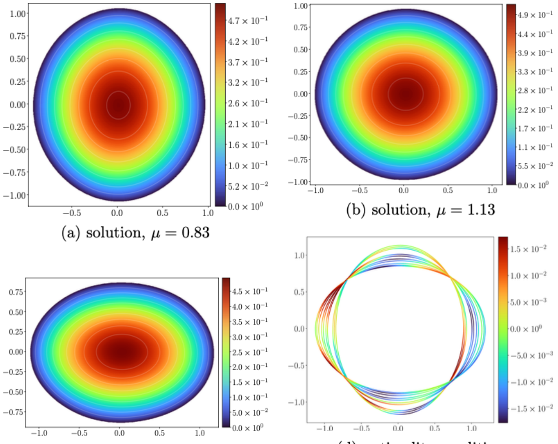
Source terms for numerical simulations

Exponential of level-set functions

We define $\phi(x, y)$ the level set function of a given shape. In the next slides we will use a source term under the form:

$$f(x, y) = \exp(\phi(x, y))$$

$\phi(x, y; a) = 1 - \left(\frac{x}{a}\right)^2 - (ay)^2, \quad a \in (0.5, 1.5)$ (1/2)



(a) solution, $\mu = 0.83$

(b) solution, $\mu = 1.13$

(c) solution, $\mu = 1.61$

(d) optimality condition

Figure: Approximate solution (top left, top right, and bottom left), optimality conditions (bottom right).

$\phi(x, y; a) = 1 - \left(\frac{x}{a}\right)^2 - (ay)^2, \quad a \in (0.5, 1.5)$ (2/2)

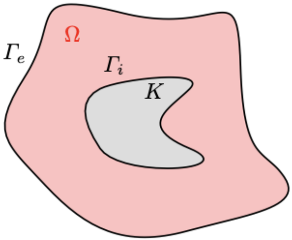
Statistics of relevant metrics in the case of a parametric source term given by (1), obtained by computing each metric for 10^3 values of μ.

Metric	Mean	Max	Min	Variance
optim. cond.	1.88×10^{-4}	4.97×10^{-4}	6.91×10^{-5}	7.12×10^{-9}
var. form.	5.89×10^{-2}	1.58×10^{-1}	3.21×10^{-3}	8.37×10^{-4}

The Bernoulli overdetermined PDE

Problem statement

$$\begin{cases} -\Delta u_\Omega^K = 0 & \text{in } \Omega, \\ u_\Omega^K = 1 & \text{on } \Gamma_i, \\ u_\Omega^K = 0 & \text{on } \Gamma_e, \\ \nabla u \cdot n = c & \text{on } \partial\Gamma_e. \end{cases}$$



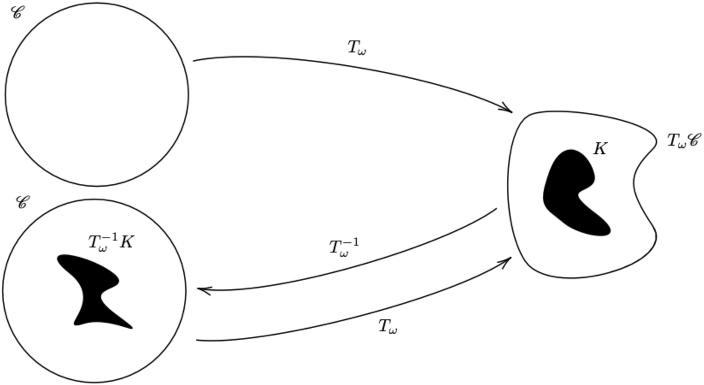
The Dirichlet energy has to be optimized!

The solution u_Ω^K of Bernoulli problem remains a minimum of the Dirichlet energy

$$\mathcal{E}_K(\Omega) = \frac{1}{2}\int_\Omega |\nabla u_\Omega^K|^2$$

Numerical strategy for the Bernoulli problem[6]

How to sample points?



[6] A. Belieres Frendo, E. Franck, V. Michel Dansac, *et al.*, "Learning-based shape optimisation," *To be published soon...*,

Bernoulli overdetermined PDE resolution

K is an ellipse of parameters $a = 6/10$ and $b = 10/6$

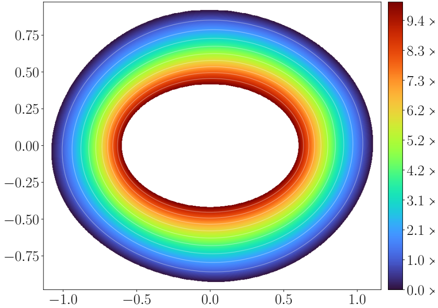


Figure: PDE

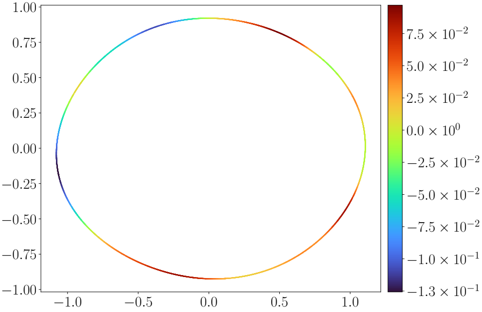
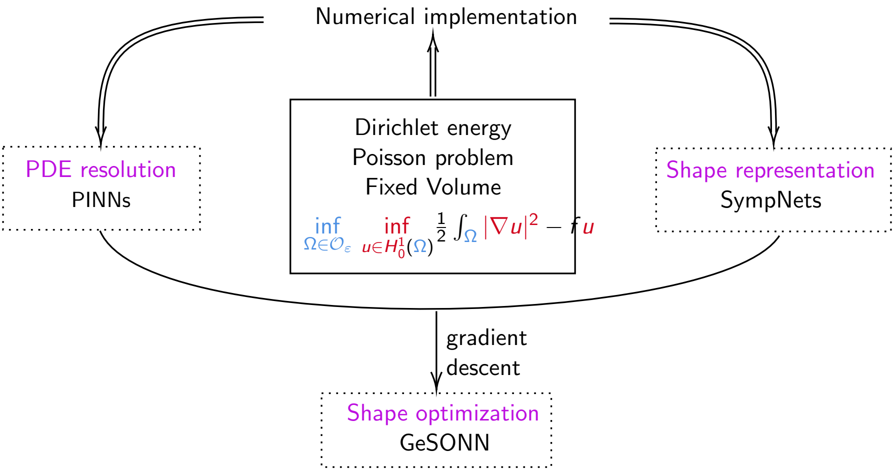


Figure: Optimality condition, variance: 10^{-3}

Conclusion


Numerical implementation
Dirichlet energy
Poisson problem
Fixed Volume
$\inf_{\Omega \in \mathcal{O}_\varepsilon} \inf_{u \in H^1_0(\Omega)} \frac{1}{2} \int_\Omega |\nabla u|^2 - f u$
PDE resolution
PINNs
Shape representation
SympNets
gradient
descent
Shape optimization
GeSONN

What is going next ?

With SympNets

- Minimization of the first eigen value of the Laplacian
- 3D
- Stokes equations
- compliance

Topological optimization

- Porous materials.

Thank you for your attention!

Bibliography

[1] A. Henrot and M. Pierre, *Variation et optimisation de formes: Une analyse géométrique*. Springer Berlin Heidelberg, 2005.

[2] P. Jin, Z. Zhang, A. Zhu, Y. Tang, and G. E. Karniadakis, "SympNets: Intrinsic structure-preserving symplectic networks for identifying Hamiltonian systems," *Neural Networks*, vol. 132, pp. 166–179, 2020.

[3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019.

[4] A. Belieres Frendo, E. Franck, V. Michel Dansac, and Y. Privat, "Learning-based shape optimisation," *To be published soon...*,