# An introduction to the Level Set method

Charles Dapogny

CNRS & Laboratoire Jean Kuntzmann, Université Grenoble-Alpes, Grenoble, France

Fall, 2023

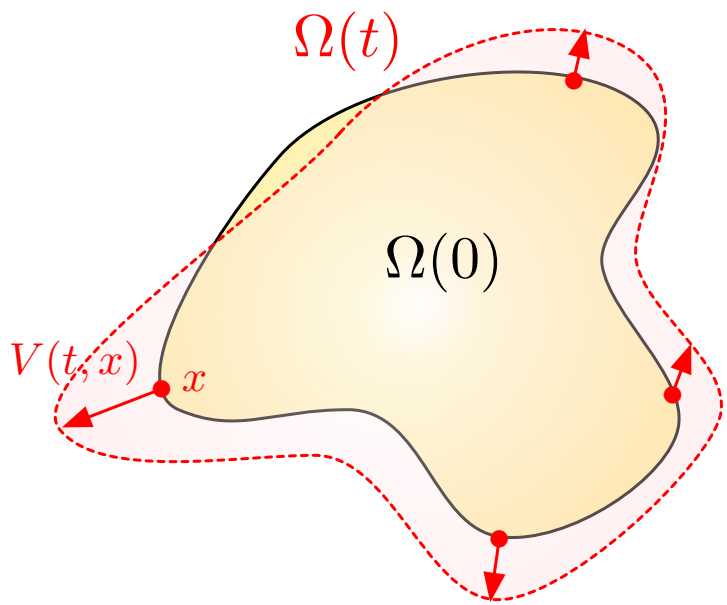
# Part I

# Introduction: evolving domains

# Motion of a domain

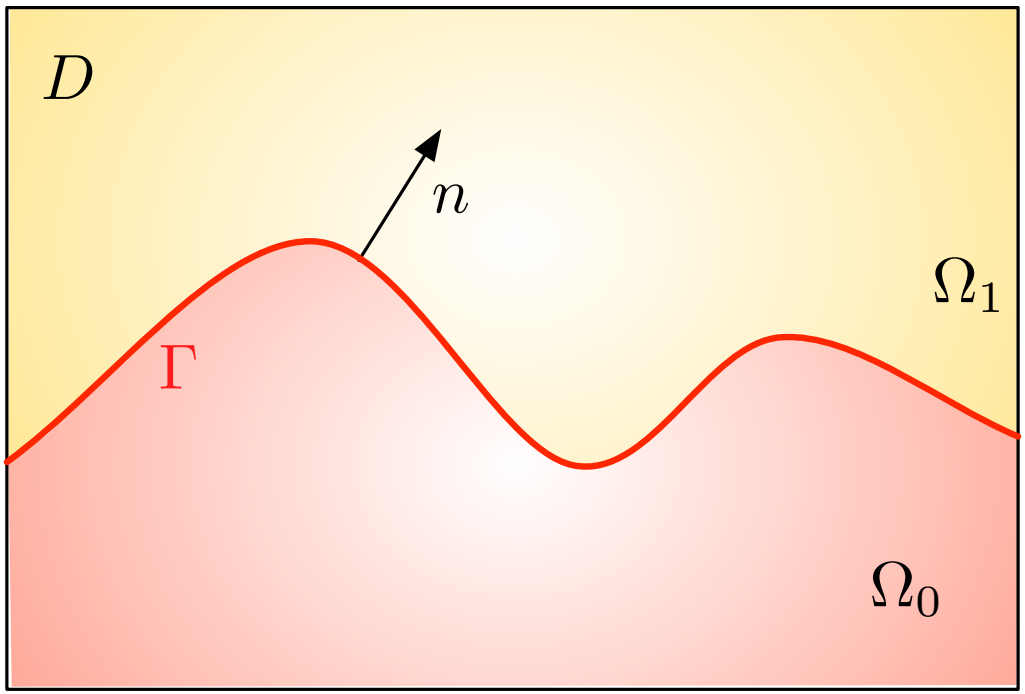
- This course is devoted to the study of the motion of a domain (or a phase) $\Omega(t)$, with boundary $\Gamma(t) := \partial\Omega(t)$, along a velocity field $V(t, x)$.
- Since its inception by Osher and Sethian [OSeth], the level set method has been one convenient framework from both theoretical and numerical viewpoints.
- It allows to describe very large evolutions (including topological changes) in a robust way.

- Other theoretical or numerical methods for the study of moving domains include:
  - Arbitrary Lagrangian-Eulerian (ALE) methods,
  - Phase-field methods,
  - Volume of Fluid (VOF) methods,
  - etc.



# A guiding example: bifluid flows (I)

A domain $D \subset \mathbb{R}^d$ is filled with two immiscible fluids, occupying complementary phases $\Omega_0, \Omega_1$, with different densities $\rho_0, \rho_1$ and dynamic viscosities $\nu_0, \nu_1$.




*Model situation for a bifluid problem.*

# A guiding example: bifluid flows (II)

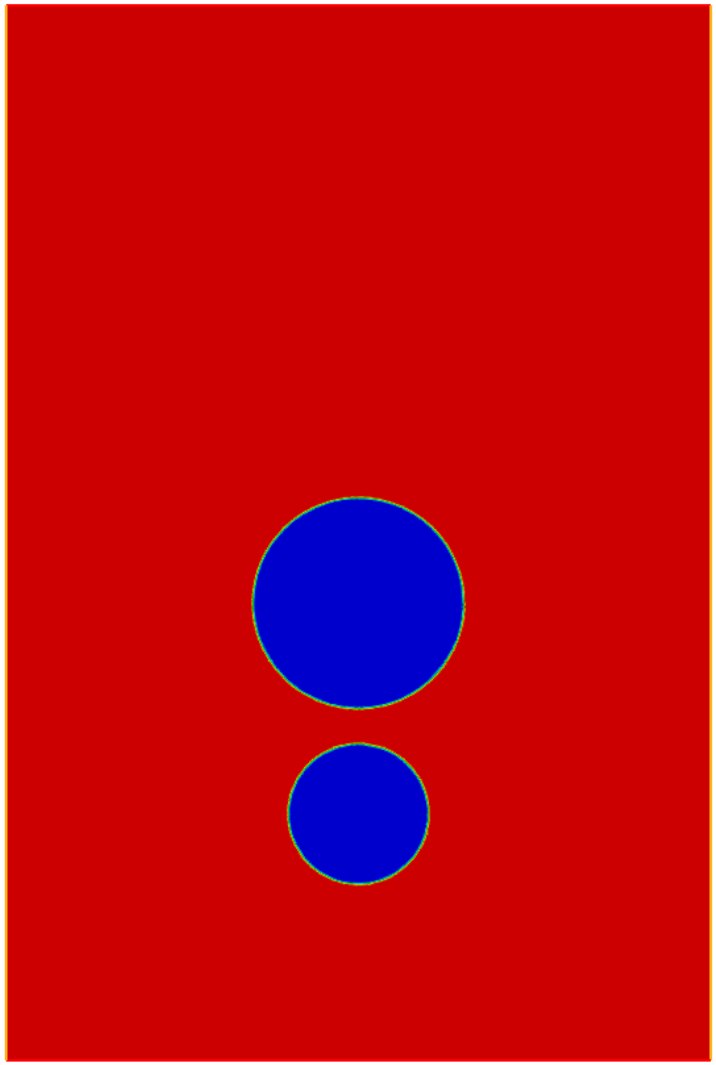
- The velocity $u(t,x)$ and pressure $p(t,x)$ of the mixture solve the Navier-Stokes equations:

$$\begin{cases} \rho_i \left(\frac{\partial u}{\partial t} + u \cdot \nabla u\right) - \nu_i \Delta u + \nabla p = f_i & \text{for } (t,x) \in (0,T) \times \Omega_i(t), \\ \operatorname{div}(u_i) = 0 & \text{for } (t,x) \in (0,T) \times \Omega_i(t), \\ u_i(t,x) = 0 & \text{for } (t,x) \in (0,T) \times \partial D, \\ u_0(t,\cdot) = u_1(t,\cdot) & \text{on } \Gamma(t), \\ (\sigma_0 - \sigma_1) \cdot n_t = -\gamma \kappa_t n_t & \text{on } \Gamma(t), \\ u_i(t=0,\cdot) = u_{i,0}(\cdot) & \text{on } \Omega_i(0). \end{cases}$$

- The interface $\Gamma(t)$ between both fluids moves according to the velocity of the fluid:

$$u_0(t,x) = u_1(t,x), \quad t \geq 0, \ x \in \Gamma(t).$$

# A guiding example: bifluid flows (III)

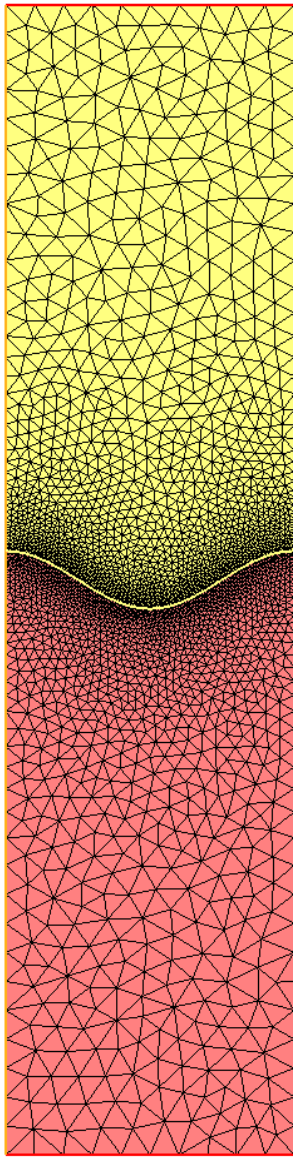


*Evolution of two fluid bubbles immersed in a fluid with larger density*

# A guiding example: bifluid flows (IV)

- "Naive" numerical methods (e.g. acting by mesh deformation) are typically not robust enough to handle large shape deformations.

- This calls for more sophisticated descriptions of shapes and their evolutions.



# Three classes of domain motions

In general, the motion of $\Omega(t)$ may be classified into three categories depending on the nature of the velocity field $V(t,x)$.

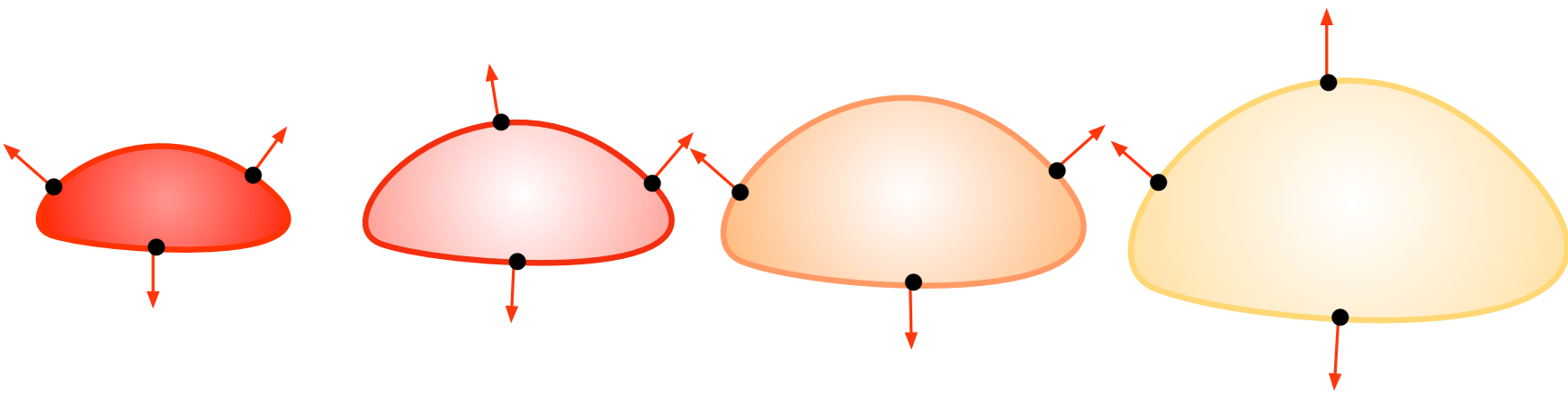
1. $\Omega(t)$ is passively transported by $V(t,x)$: $V(t,x)$ is externally prescribed, i.e. it does not depend on $\Omega(t)$.

2. The velocity $V(t,x)$ depends on local features of $\Omega(t)$ or $\Gamma(t)$, such as:
   - the normal vector $n_t(x)$ at $x \in \Gamma(t)$;
   - the mean curvature $\kappa_t(x)$ of $\Gamma(t)$.

# Three classes of domain motions

**Example** 1 *The flame propagation model*: $\Omega(t)$ represents a burnt region, whose front expands with constant, normal velocity $c$:

$$V(t,x) = c\, n_t(x), \text{ where } c > 0 \text{ is a constant.}$$

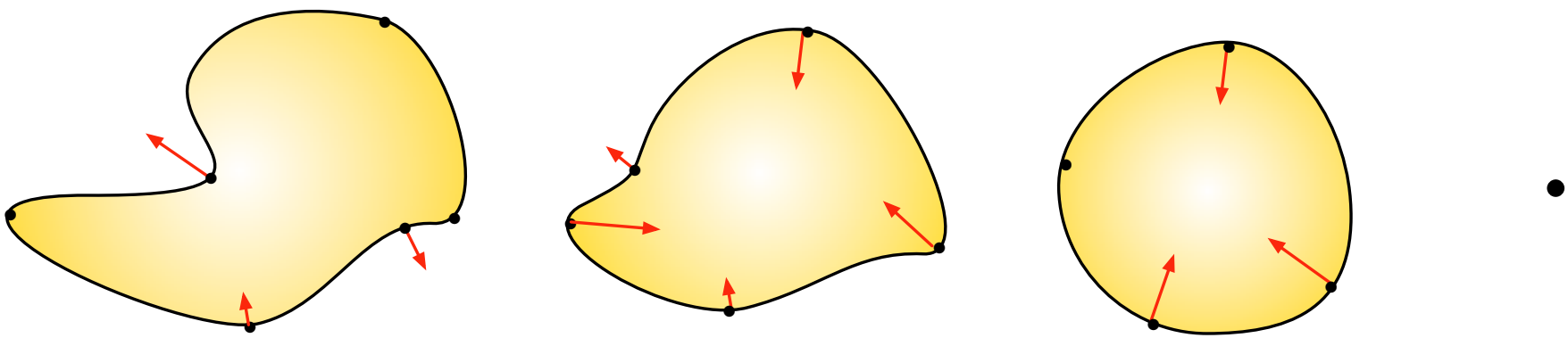


*An example of the dynamics in the flame propagation model.*

**Example** 2 *The mean curvature flow*:

$$V(t,x) = -\kappa_t(x)\, n_t(x),$$

that is, $\Omega(t)$ evolves by "resorption of its bumps", and "filling of its creases".



*An example of the dynamics of the Mean Curvature Flow: Grayson's result [Grayson].*

# Three classes of domain motions

3. The field $V(t,x)$ depends on global features of the domain $\Omega(t)$, e.g. it is obtained by solving one (or several) PDE posed on $\Omega(t)$.

   - We have seen that when $\Omega(t)$ accounts for a fluid domain, $V(t,x)$ is the solution to the Stokes, or the Navier-Stokes equations, whose physical coefficients depend on $\Omega(t)$.

   - We shall see several other examples of motions of this category; for instance, in structural optimization, $V(t,x)$ involves the solution to one, or several linear elasticity equations posed on $\Omega(t)$.

   - Motions of this category are by far the most interesting ones in practice, ... and the most difficult to analyze. They are usually approximated by a series of motions of one of the first two kinds.


## Disclaimer

- This course is an introduction, and is only devoted to the basic features of the Level Set method.
- It is oriented towards applications, and difficult mathematical details are only hinted at, see the monograph [Giga] around these points.

# Part II

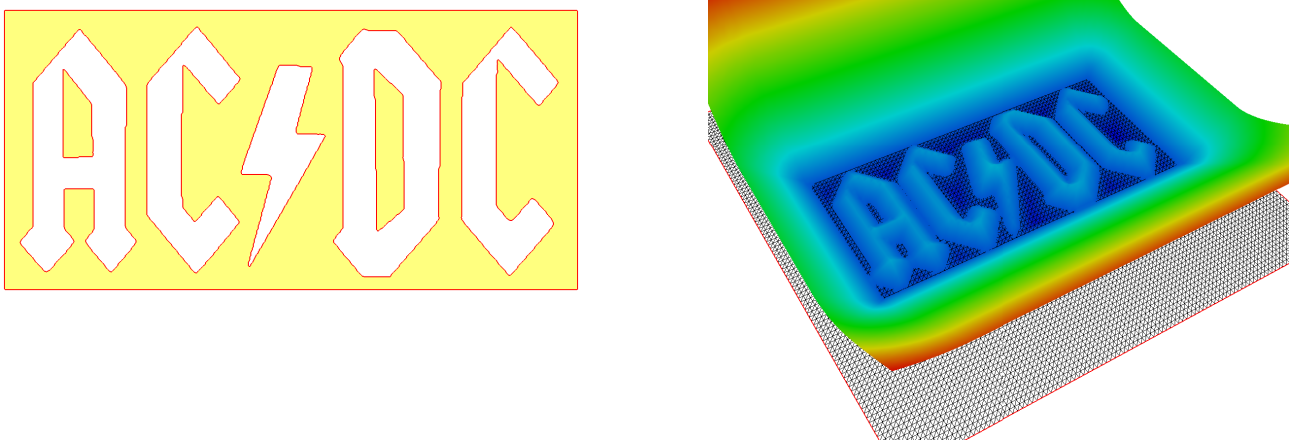
# The mathematical framework of the level set method

# Implicit geometries (I)

**A paradigm:** *the motion of an evolving domain is best described in an implicit way.*

A domain $\Omega \subset \mathbb{R}^d$ is equivalently defined by a function $\phi : \mathbb{R}^d \to \mathbb{R}$ such that:

$$\phi(x) < 0 \quad \text{if } x \in \Omega \quad ; \quad \phi(x) = 0 \quad \text{if } x \in \Gamma \quad ; \quad \phi(x) > 0 \quad \text{if } x \in {}^c\overline{\Omega}$$



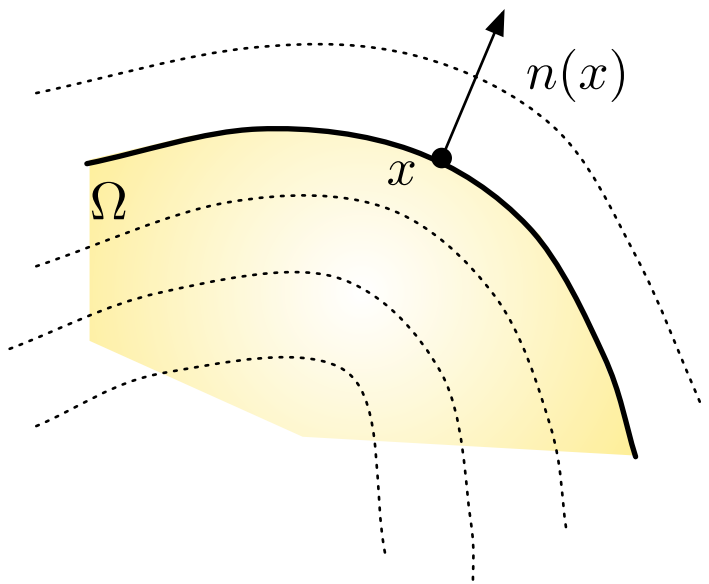
*(Left) a domain $\Omega \subset \mathbb{R}^2$; (right) the graph of an associated level set function.*

# Implicit geometries (II)

Let $\Omega \subset \mathbb{R}^d$ be a domain, $\phi : \mathbb{R}^d \to \mathbb{R}$ be a level set function of class $\mathcal{C}^2$ for $\Omega$, such that $\nabla\phi(x) \neq 0$ on a neighborhood of $\Gamma$.

- The normal vector $n$ to $\Gamma$ pointing outward $\Omega$ reads:

$$\forall x \in \Gamma, \quad n(x) = \frac{\nabla\phi(x)}{|\nabla\phi(x)|}.$$




*Normal vector to a domain $\Omega$; some isolines of the function $\phi$ are dotted.*

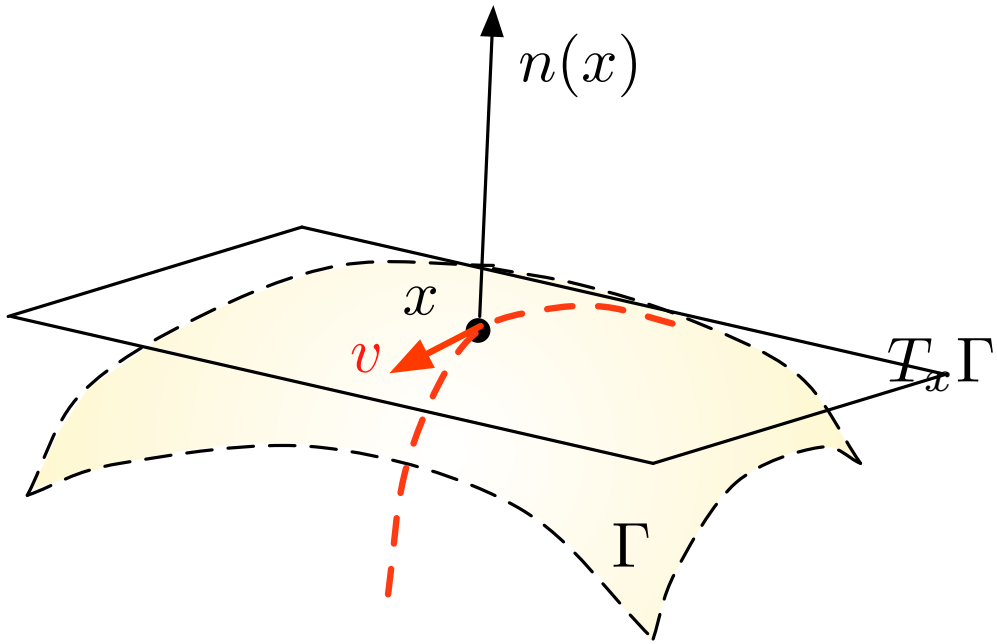
# Implicit geometries (III)

- The second fundamental form $\mathrm{II}$ of $\Gamma$ is:

$$\forall x \in \Gamma,\ \mathrm{II}(x) = \nabla\left(\frac{\nabla\phi(x)}{|\nabla\phi(x)|}\right).$$

- The mean curvature $\kappa$ of $\Gamma$ is:

$$\forall x \in \Gamma,\ \kappa(x) = \mathrm{div}\left(\frac{\nabla\phi(x)}{|\nabla\phi(x)|}\right).$$




*$\mathrm{II}_x(v,v)$ is the curvature of a curve drawn on $\Gamma$ with tangent vector $v$ at $x$.*

# Part II

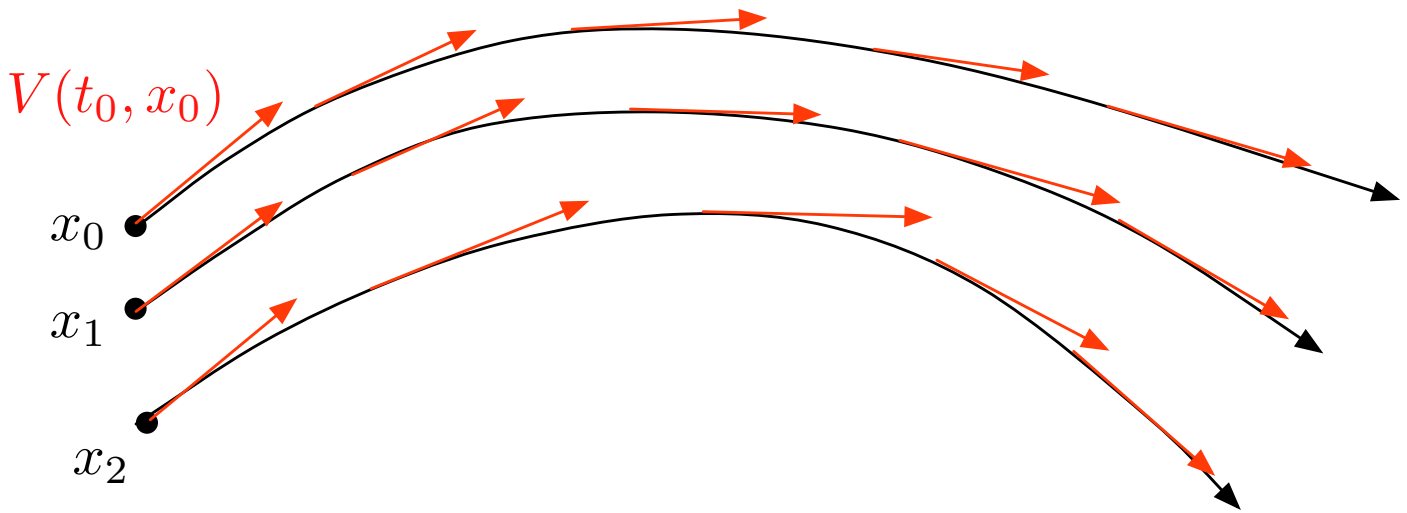
# The mathematical framework of the level set method

# Evolving domains (I)

## Definition 1.

*Let $V(t,x)$ be a smooth velocity field. The characteristic curve emerging from a point $x \in \mathbb{R}^d$ at time $t = t_0$ is the curve $t \mapsto \chi(x,t,t_0)$ defined by the ODE:*

$$\begin{cases} \frac{d}{dt}(\chi(x,t,t_0)) = V(t,\chi(x,t,t_0)), & \text{for } t \in (0,T) \\ \chi(x,t_0,t_0) = x. \end{cases}$$



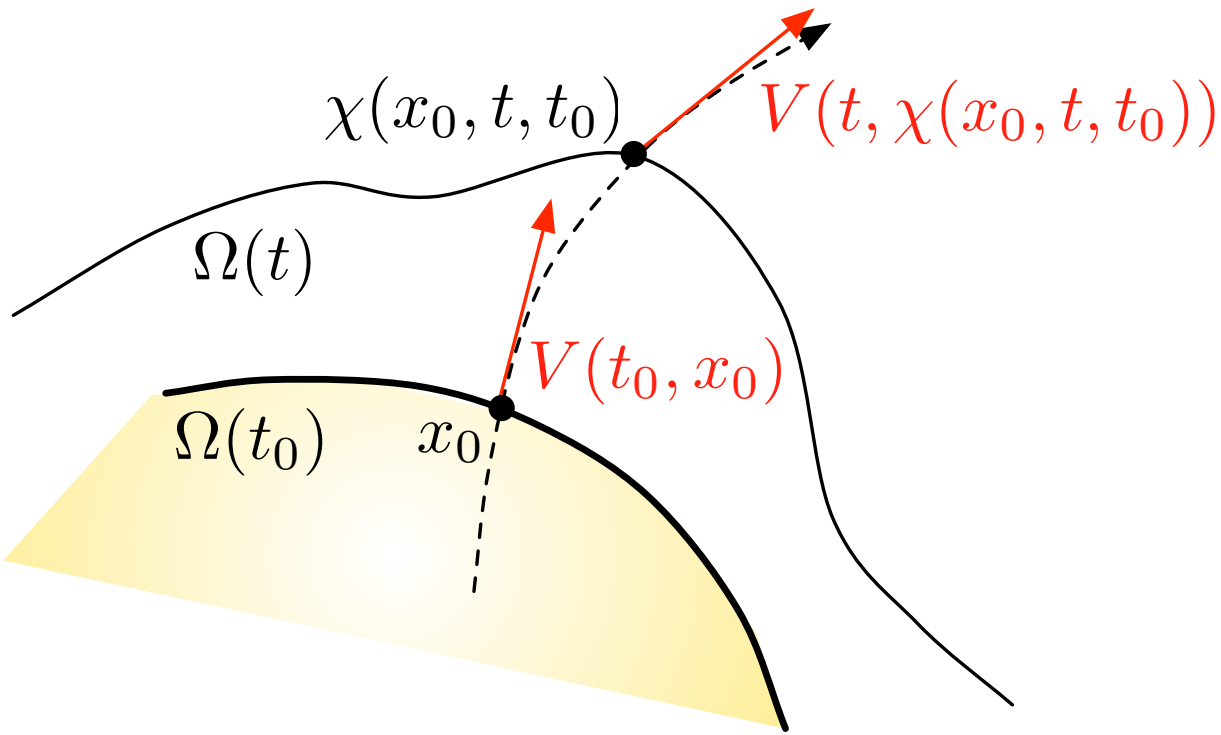

*Three characteristic curves of the velocity field $V$ issued at $t = t_0$ from different points $x_0, x_1, x_2$.*

# Evolving domains (II)

## "Intuitive" notion of an evolving domain

A domain $\Omega(t)$ evolves according to a velocity field $V(t,x)$ from an initial position $\Omega(t_0)$ if it is obtained by transporting its points along $V$:

$$\Omega(t) = \{\chi(x_0, t, t_0),\ x_0 \in \Omega(t_0)\}\,.$$



# Evolving domains (III)

- Let $\Omega(t)$ be a (smooth) domain, moving over $(0, T)$ along the (smooth) velocity field $V(t, x)$. Let $\phi(t, x)$ be a smooth level set function, i.e:

$$\forall t \in (0, T),\ x \in \mathbb{R}^d,\ \begin{cases} \phi(t, x) < 0 & \text{if } x \in \Omega(t), \\ \phi(t, x) = 0 & \text{if } x \in \Gamma(t), \\ \phi(t, x) > 0 & \text{if } x \in {}^c\overline{\Omega(t)}. \end{cases}$$

- Let $x_0 \in \Gamma(0)$ be fixed. By the intuitive definition of an evolving domain, it comes:

$$\forall t \in (0, T),\ \ \phi(t, \chi(x_0, t, 0)) = 0.$$

- Differentiating and using the chain rule, we obtain:

$$\frac{\partial \phi}{\partial t}(t, \chi(x_0, t, 0)) + \frac{d}{dt}(\chi(x_0, t, 0)) \cdot \nabla \phi(t, \chi(x_0, t, 0)) = 0.$$

# Evolving domains (IV)

- Since this holds for any point $x_0 \in \Gamma(0)$, we obtain the level set advection equation ($\neq$ "classical" advection equation):

$$\boxed{\forall t \in (0,T),\ \forall x \in \mathbb{R}^d,\ \ \frac{\partial \phi}{\partial t} + V(t,x) \cdot \nabla \phi = 0.}$$

- If, in addition, the velocity is consistently oriented along the normal vector $n_t(x)$ to $\Omega(t)$, that is:

$$V(t,x) = v(t,x)\frac{\nabla \phi(t,x)}{|\nabla \phi(t,x)|}, \text{ for some scalar field } v(t,x),$$

the equation rewrites as the Level Set Hamilton-Jacobi equation ($\neq$ "classical" Hamilton-Jacobi equation):

$$\boxed{\forall t \in (0,T),\ \forall x \in \mathbb{R}^d,\ \ \frac{\partial \phi}{\partial t} + v(t,x)|\nabla \phi| = 0.}$$

# Evolving domains: comments (I)

- Strictly speaking, both equations only hold for pairs $(t, x)$ with $x \in \Gamma(t)$. However, the previous analysis can be applied to any level set of $\phi$:

$$\Gamma_c(t) := \left\{ x \in \mathbb{R}^d, \ \ \phi(t, x) = c \right\}.$$

  Hence, the equation:

$$\forall t \in (0, T), \ \forall x \in \mathbb{R}^d, \ \ \frac{\partial \phi}{\partial t}(t, x) + V(t, x) \cdot \nabla \phi(t, x) = 0$$

  actually accounts for the fact that all the level sets of $\phi$ (and not only its 0 level set) move according to $V(t, x)$.

- In many applications, the velocity field $V(t, x)$ makes sense only for $x \in \Gamma(t)$. In the above derivation, it is implicitly assumed that $V(t, x)$ has been extended to the whole space $\mathbb{R}^d$.

# Evolving domains: comments (II)

- More fundamentally, this derivation rests upon the assumption that $\Omega(t)$, $V(t,x)$, $\phi(t,x)$ stay "smooth" over $(0,T)$.

## Question

How is it possible to account for the evolution of $\Omega(t)$ when either the domain $\Omega(t)$, or the velocity field $V(t,x)$ has developed a singularity?

This problem is not a pure mathematicality: even in the simplest models, $\Omega(t)$ and $V(t,x)$ (thus $\phi(t,x)$) become singular in finite time.

# Part II

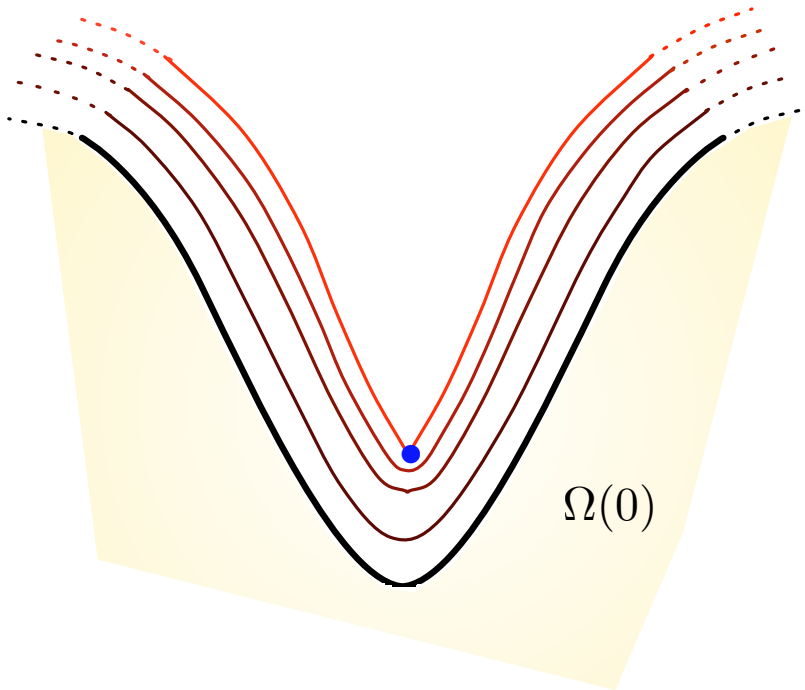
# The mathematical framework of the level set method

# Emergence of singularities (I)

- In the flame propagation model, a domain $\Omega(t)$, whose initial boundary $\Gamma(0)$ is (locally) described by the curve:

$$\gamma(s) = \left(1 - s, \frac{1 + \cos(2\pi s)}{2}\right), \quad s \in [0, 1],$$

evolves according to the velocity field $V(t, x) = n_t(x)$.

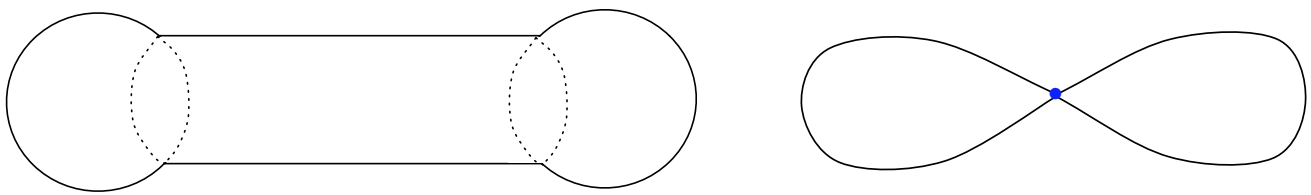



*Some positions of the interface $\Gamma(t)$; at a critical time $t = t_c$, $\Gamma(t)$ develops a singularity where $n_t(x)$ (thus $V(t, x)$) is not defined (blue dot).*

# Development of singularities (II)

- In the context of the mean curvature flow, consider a "dumbbell"-shaped initial domain $\Omega(0)$, evolving according to the velocity field

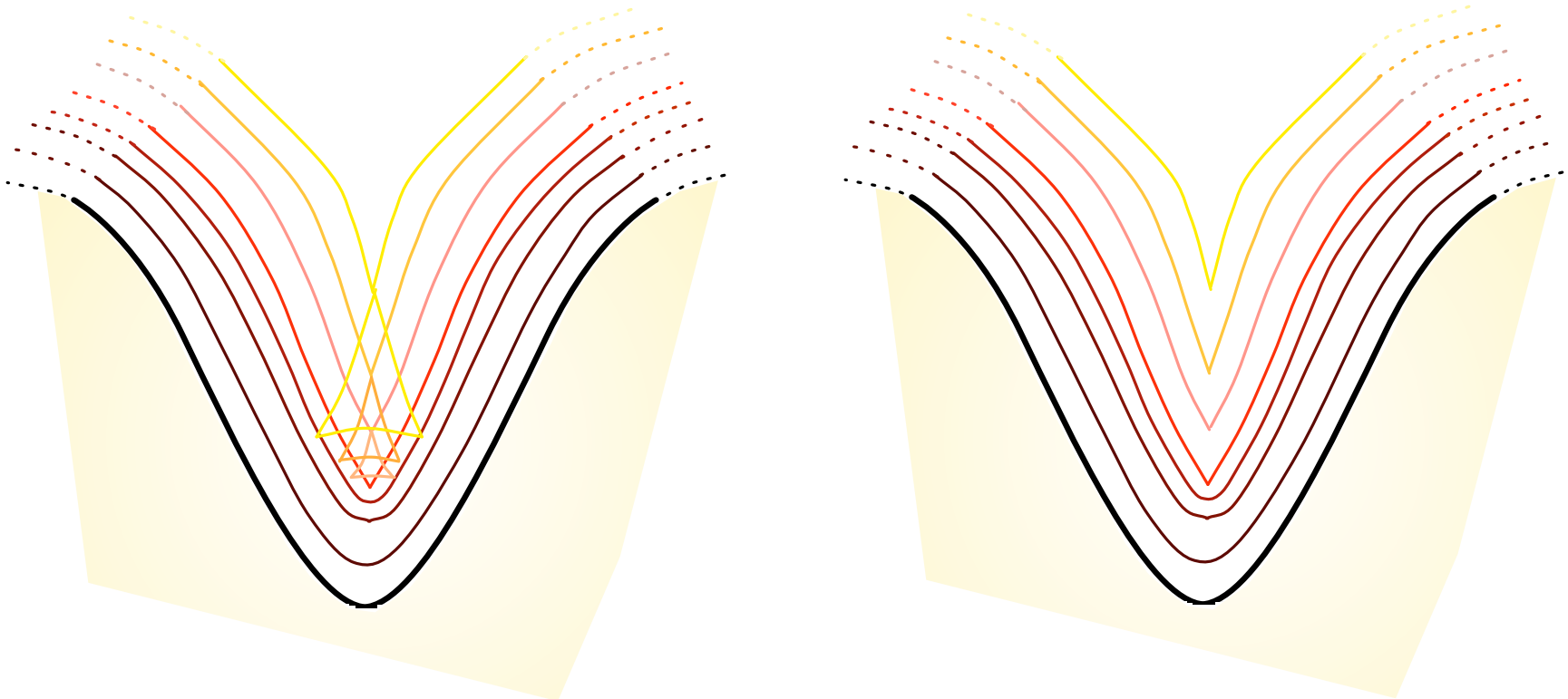
$$V(t,x) = -\kappa_t(x) n_t(x).$$



*Evolution of a three-dimensional dumbbell under the mean curvature flow. The central part of the bar ends up pinching.*

# Development of singularities (III)

In the flame propagation example (and in general), there are several ways of giving a sense to the evolution of the front once a singularity has appeared.



*(Left) Evolution of $\Omega(t)$ obtained by "pursuing the motion" of all the points of $\Gamma(t)$ where the normal vector is defined; (right) Evolution of $\Omega(t)$ obtained by imposing an "entropy criterion": "a burnt point stays burnt".*

# Development of singularities (IV)

- Singularities are inevitable, even in the case of a "very smooth" motion.
- What happens after the onset of singularities is actually a matter of defining the motion of a possibly non smooth domain, under a possibly non smooth velocity.
- The level set equations have "too many" solutions.

$\Rightarrow$ Need to devise a "good", generalized notion of solutions, which selects the correct "physical" behavior.

## Mathematical definition of an evolving domain

1. Start from any level set function $\phi_0(x)$ for the initial domain $\Omega(0)$.
2. Solve the level set evolution equation

$$\begin{cases} \frac{\partial \phi}{\partial t} + V(t,x) \cdot \nabla \phi = 0 & \text{for } t \in (0,T),\ x \in \mathbb{R}^d, \\ \phi(0,x) = \phi_0(x) & \text{for } x \in \mathbb{R}^d. \end{cases}$$

3. Define the domain $\Omega(t)$ by $\Omega(t) = \left\{ x \in \mathbb{R}^d,\ \ \phi(t,x) < 0 \right\}$.

## Definition 2.

*Let $U \subset \mathbb{R}^d$ be open, and $H : \mathbb{R}^d_x \times \mathbb{R}_u \times \mathbb{R}^d_p \times \mathcal{S}_d(\mathbb{R})$ be a continuous function (the Hamiltonian). Consider the second-order Hamilton-Jacobi equation:*

$$\frac{\partial u}{\partial t}(t,x) + H(x, u, \nabla u, \nabla^2 u)(t,x) = 0, \quad \text{on } (0,T) \times U. \tag{HJ}$$

- *A function $u$ is a viscosity subsolution of (HJ) if:*
  1. *it is upper semicontinuous on $(0,T) \times U$,*
  2. *for any function $\varphi$ of class $\mathcal{C}^2$ on $U$ such that $u - \varphi$ reaches a local maximum (say, 0) at $(t,x)$,*

$$\frac{\partial \varphi}{\partial t}(t,x) + H(x, u(t,x), \nabla \varphi(t,x), \nabla^2 \varphi(t,x)) \leq 0.$$

## Definition 2.

- *A function $u$ is a viscosity supersolution of (HJ) if:*
  1. *It is lower semicontinuous on $(0,T)\times U$,*
  2. *for any function $\varphi$ of class $\mathcal{C}^2$ on $(0,T)\times U$ such that $u-\varphi$ reaches a local minimum (say, 0) at $(t,x)$,*

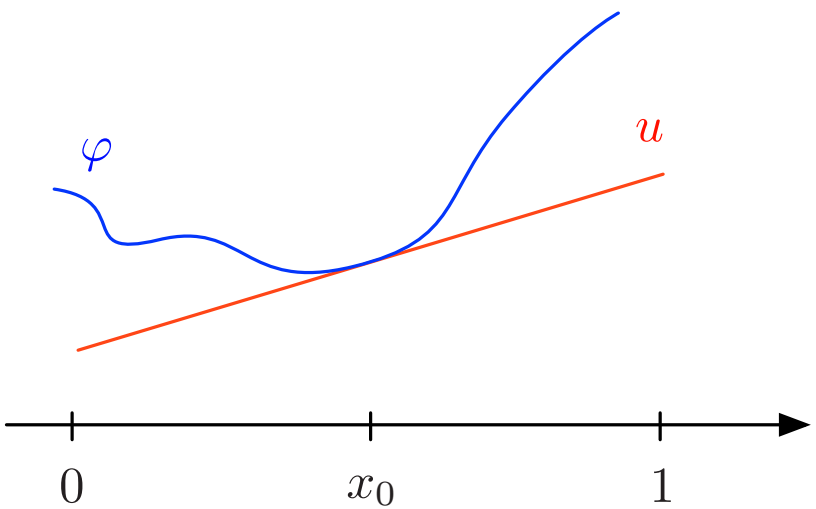
$$\frac{\partial\varphi}{\partial t}(t,x)+H(x,u(t,x),\nabla\varphi(t,x),\nabla^2\varphi(t,x))\geq 0.$$

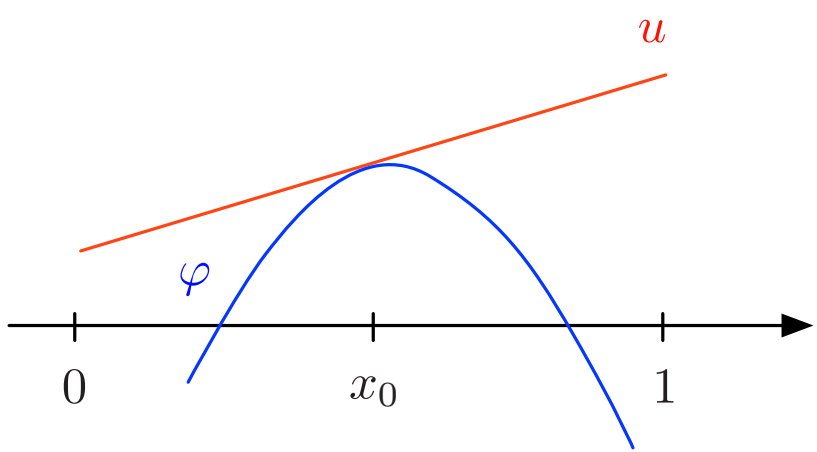
- *A function $u$ is a viscosity solution of (HJ) if:*
  1. *(It is continuous on $(0,T)\times U$)*
  2. *it is both a viscosity subsolution and a viscosity supersolution.*

# Viscosity solutions to Hamilton-Jacobi equations (III)

**Example** Affine functions of the form $u(x) = ax + b$ are viscosity solutions to the equation $-u'' = 0$ in $(0, 1)$:

- *Subsolution inequality:* If $u - \varphi$ has a local maximum at $x_0$ (situation on the left), $\varphi$ is locally above $u$ around $x_0$, and
$$-\varphi''(x_0) \leq 0.$$
- *Supersolution inequality:* If $u - \varphi$ has a local minimum at $x_0$ (situation on the right), $\varphi$ is locally below $u$ around $x_0$, and
$$-\varphi''(x_0) \geq 0.$$

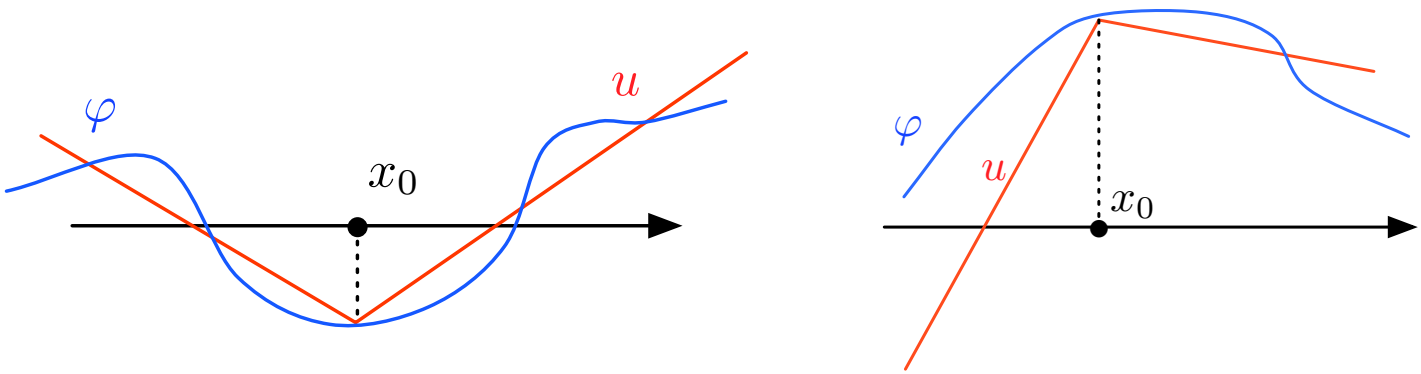


# Viscosity solutions to Hamilton-Jacobi equations (IV)

*Motivations for this definition:*

- It leaves the room for solutions $u$ which are only continuous, but not differentiable: the gradient and the Hessian matrix of $u$ in (HJ) are replaced by those of *any* smooth function which locally "behaves like" $u$.
- The two comparison criteria take into account important monotonicity properties of Hamilton-Jacobi equations.
- Viscosity solutions enjoy many "physical" properties...




*(Left) $u - \varphi$ has a local minimum at $x_0$; (right) $u - \varphi$ has a local maximum at $x_0$.*

# Properties of viscosity solutions (I)

Under "reasonable assumptions" on the Hamiltonian function $H$,

- *Existence and uniqueness.* For a given initial data $u_0$, the viscosity solution $u$ of (HJ) exists and is unique.

- *Generalization of classical solutions.* If the viscosity solution $u$ of (HJ) is of class $\mathcal{C}^2$, then it is also a solution of this equation in the classical sense.

- *Vanishing viscosity limit of solutions to "regular" equations.* For small $\varepsilon > 0$, let $u_\varepsilon(t,x)$ be the (smooth) solution to the equation:

$$\begin{cases} \frac{\partial u_\varepsilon}{\partial t}(t,x) - \varepsilon \Delta u_\varepsilon(t,x) + H(x, u_\varepsilon, \nabla u_\varepsilon, \nabla^2 u_\varepsilon)(t,x) = 0, \\ u_\varepsilon(t=0,\cdot) = u(t=0,\cdot) \end{cases},$$

  obtained by adding to (HJ) the regularizing viscosity term $-\varepsilon \Delta u_\varepsilon$.

  Then, $u_\varepsilon \overset{\varepsilon \to 0}{\longrightarrow} u$, uniformly on every compact subset of $[0,T] \times \mathbb{R}^d$.

# Properties of viscosity solutions (II)

- *Independence from the initial level set function.* Let $u_0$, $v_0$ be two level set functions for an initial domain $\Omega_0$, i.e.

$$\Omega_0 = \left\{ x \in \mathbb{R}^d, \ u_0(x) < 0 \right\} = \left\{ x \in \mathbb{R}^d, \ v_0(x) < 0 \right\},$$

and $u(t,\cdot)$, $v(t,\cdot)$ be the corresponding solutions of (HJ). Then, $u$ and $v$ define the same domain:

$$\forall t \in (0,T), \ \left\{ x \in \mathbb{R}^d, \ u(t,x) < 0 \right\} = \left\{ x \in \mathbb{R}^d, \ v(t,x) < 0 \right\}.$$

- *Monotonicity.* Let $\Omega_0 \subset \widetilde{\Omega_0}$ be domains in $\mathbb{R}^d$, $u_0$ and $\widetilde{u_0}$ be corresponding level set functions. Define:

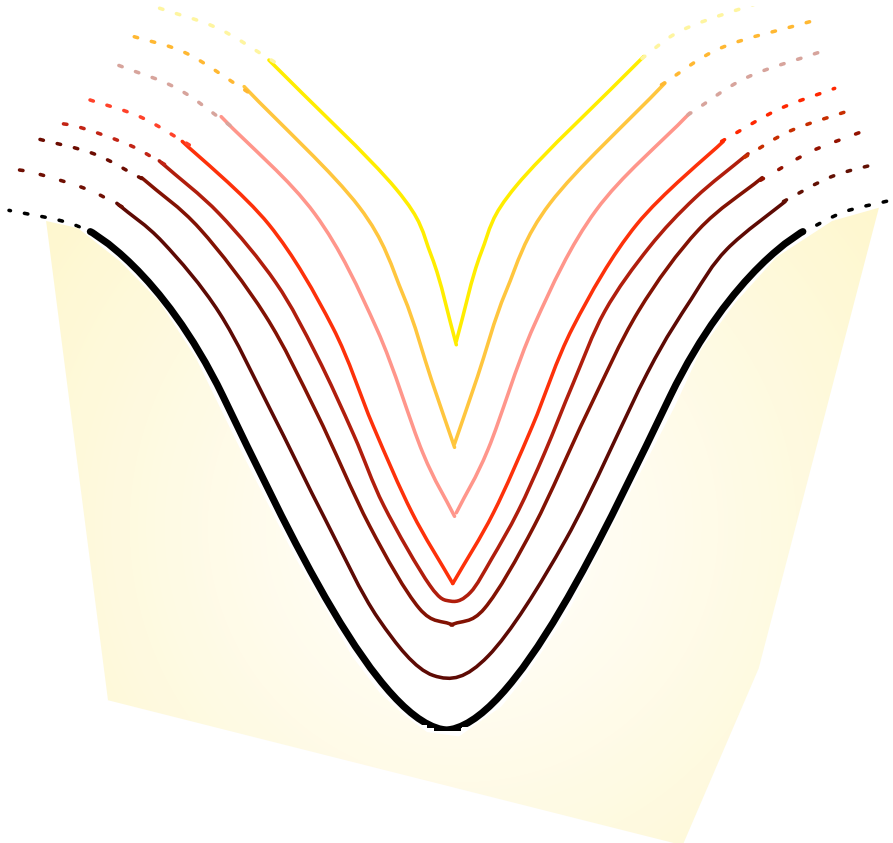
$$\Omega(t) = \left\{ x \in \mathbb{R}^d, \ u(t,x) < 0 \right\},$$

where $u(t,\cdot)$ solves (HJ) with initial data $u_0$, and likewise for $\widetilde{\Omega}(t)$.

Then, $\Omega(t) \subset \widetilde{\Omega}(t)$.

# Properties of viscosity solutions (III)

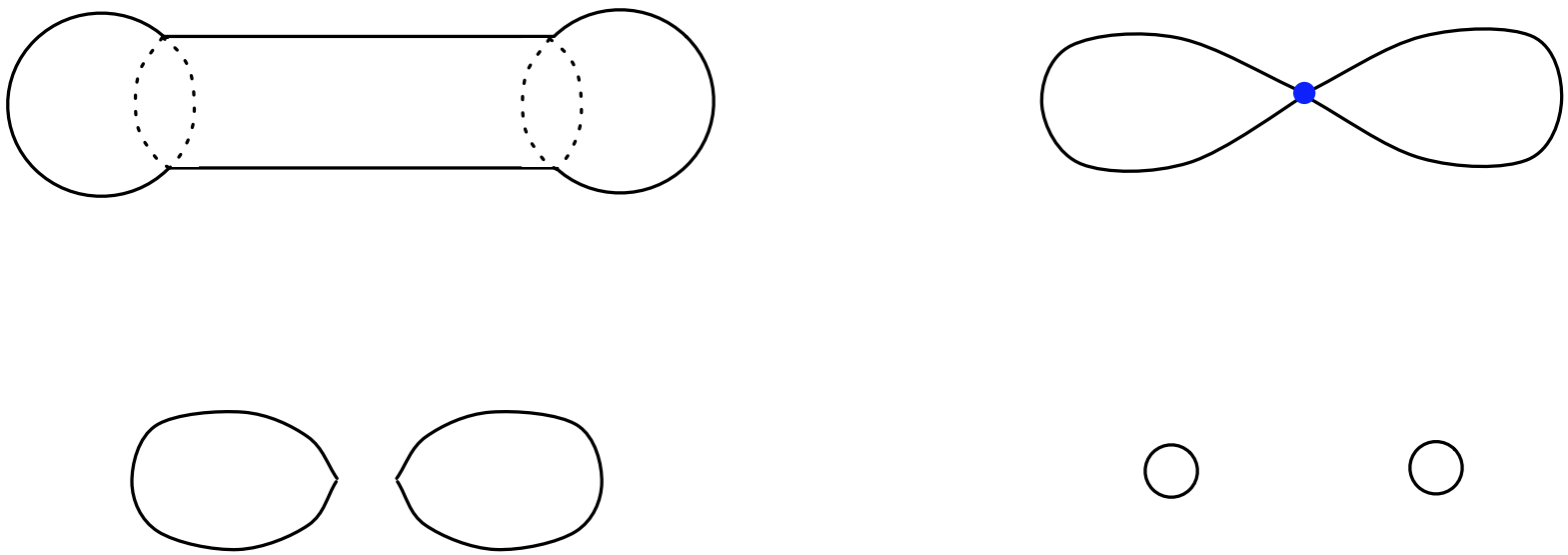
**Example** 1 In the flame propagation model, the evolution of $\Omega(t)$ selected by this process coincides with that obtained by imposing the "entropy criterion".



*Domain* $\Omega(t) := \{x \in \mathbb{R}^d,\ \phi(t,x) < 0\}$, *where* $\phi$ *is the viscosity solution to the Hamilton-Jacobi equation* $\frac{\partial \phi}{\partial t} + |\nabla \phi| = 0$.

# Properties of viscosity solutions (IV)

**Example** 2 When the mean curvature flow, initialized by the "dumbbell", the evolution of $\Omega(t)$ selected by (an adaptation of) the notion of viscosity solutions looks as follows:



*Domain* $\Omega(t) := \{x \in \mathbb{R}^d,\ \phi(t,x) < 0\}$, *where* $\phi$ *is the "viscosity solution" to the mean curvature flow* $\frac{\partial \phi}{\partial t} - \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)|\nabla \phi| = 0$ *(Time is increasing from left to right, top to bottom).*

# Part II

# The mathematical framework of the level set method

# Initial value problems

- An equivalent, convenient reformulation of the level set equations is available when the domain $\Omega(t)$ expands (resp. retracts) along its normal vector,

$$V(t,x) = c(x)n_t(x), \text{ where } c(x) > 0 \text{ (resp. } c(x) < 0).$$

- A stationary PDE can be derived in terms of the time function $T(x)$:

$$T(x) = \inf\{t \geq 0,\ x \in \Omega(t)\}.$$

- The derivation of this PDE follows the same trail as that of the level set equations:
  1. At first, it is rigorously established in the regions of space where $\Omega(t)$, $V(t,x)$ and $T$ are smooth,
  2. Then, a generalized notion of solutions is introduced for this PDE to impose a "physical" behavior where they are not smooth.

# A stationary PDE for initial value problems (I)

- We rely again on the intuitive notion of an evolving domain.

  Let $x_0 \in \Gamma(0)$, and $t \mapsto x(t)$ be the characteristic curve of $V(t,x)$, emerging from $x_0$ at $t = 0$:

$$x(0) = x_0, \text{ and } x'(t) = c(x(t))n_t(x(t)).$$

- By definition of the time function, it holds:

$$\Omega(t) = \left\{ x \in \mathbb{R}^d, \;\; T(x) < t \right\}, \text{ and } \Gamma(t) = \left\{ x \in \mathbb{R}^d, \;\; T(x) = t \right\}.$$

- In particular, $\phi(x) := T(x) - t$ is one level set function for $\Omega(t)$. Hence,

$$\forall t \geq 0, \; \forall x \in \Gamma(t), \;\; n_t(x) = \frac{\nabla T(x)}{|\nabla T(x)|}.$$

# A stationary PDE for initial value problems (II)

- On the other hand, differentiating the relation $T(x(t)) = t$, we obtain:

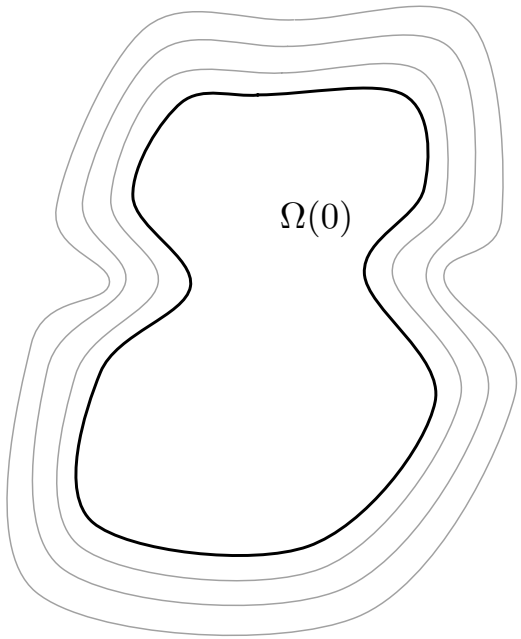
$$\forall t > 0, \quad x'(t) \cdot \nabla T(x(t)) = 1.$$

- Inserting

$$x'(t) = c(x(t)) \frac{\nabla T(x(t))}{|\nabla T(x(t))|},$$

it follows that $T$ is solution to the Eikonal equation:

$$\begin{cases} c(x)|\nabla T(x)| = 1 & \text{for } x \in \mathbb{R}^d \setminus \overline{\Omega(0)}, \\ T(x) = 0 & \text{for } x \in \Gamma(0). \end{cases}$$




*Some isolines of the time function $T$ in the particular case where $c \equiv 1$.*

# A stationary PDE for initial value problems (III)

- A similar analysis holds in the case where $\Omega(t)$ constantly retracts in the normal direction:

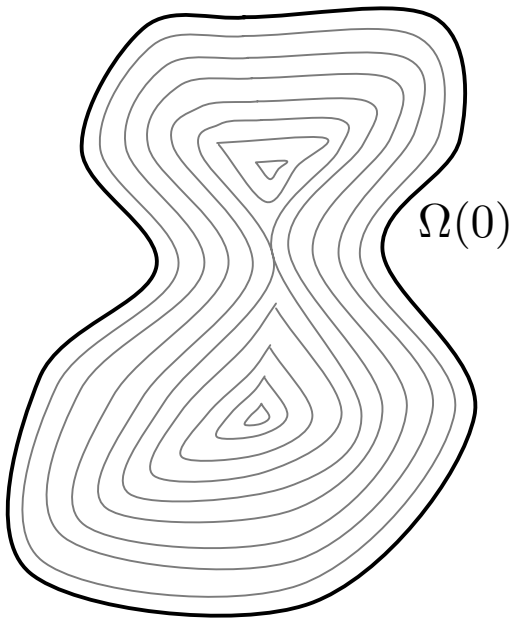
$$V(t,x) = -c(x)n_t(x), \text{ where } c(x) > 0.$$

- The time function $T : \Omega(0) \to \mathbb{R}$ is then defined by:

$$T(x) = \inf\left\{ t \geq 0,\ x \in \mathbb{R}^d \setminus \overline{\Omega(t)} \right\}.$$

- It turns out that $T$ is solution to the Eikonal equation:

$$\begin{cases} c(x)|\nabla T(x)| = 1 & \text{for } x \in \Omega(0), \\ T(x) = 0 & \text{for } x \in \Gamma(0). \end{cases}$$




*Some isolines of the time function $T$ in the particular case where $c \equiv 1$.*

## Definition 3.

*Let $H : \mathbb{R}^d_x \times \mathbb{R}^d_p \to \mathbb{R}$ be a continuous Hamiltonian function; consider the stationary Hamilton-Jacobi equation:*

$$\begin{cases} H(x, \nabla u(x)) = 0 & \text{in } \Omega, \\ u(x) = 0 & \text{on } \Gamma \end{cases} . \tag{S-HJ}$$

*A continuous function $u$ on $\overline{\Omega}$ is a viscosity solution to (S-HJ) if:*

- *Subsolution inequality: For any point $x_0 \in \Omega$, and any function $\varphi$ of class $\mathcal{C}^2$ such that $(u - \varphi)$ has a local maximum (say, 0) at $x_0$:*

$$H(x_0, \nabla\varphi(x_0)) \leq 0.$$

- *Supersolution inequality: For any point $x_0 \in \Omega$, and any function $\varphi$ of class $\mathcal{C}^2$ such that $(u - \varphi)$ has a local minimum (say, 0) at $x_0$:*

$$H(x_0, \nabla\varphi(x_0)) \geq 0.$$

# Viscosity solutions for the Eikonal equations (II)

## Theorem 1.

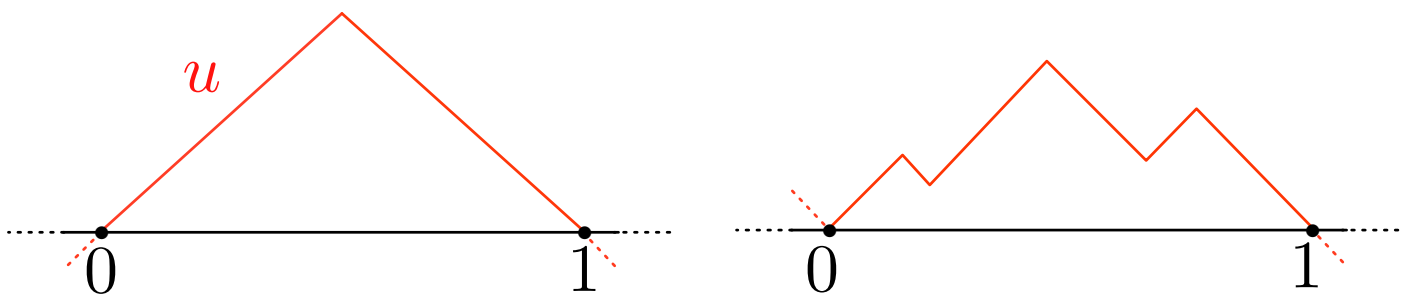
*Assume that* $c(x) > 0$ *is continuous; the Eikonal equation*

$$\begin{cases} c(x)|\nabla u(x)| = 1 & \text{in } \Omega, \\ u(x) = 0 & \text{on } \Gamma \end{cases}.$$

*has a unique viscosity solution* $u \in \mathcal{C}(\overline{\Omega})$.

*In the particular case* $c(x) \equiv 1$, $u$ *is the Euclidean distance function:*

$$u(x) = d(x, \Gamma) = \inf_{y \in \Gamma} d(x, y).$$

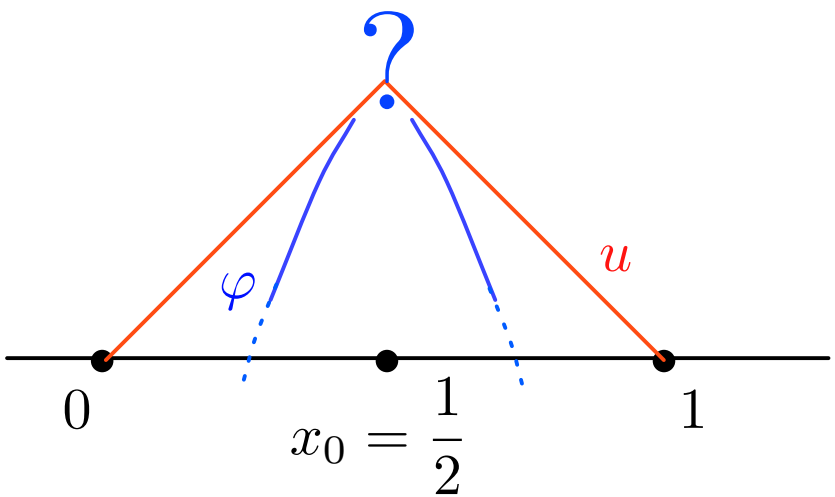



*(Left) graph of the distance function* $u = d(\cdot, \Gamma)$, *(right) graph of a function satisfying* $|u'(x)| = 1$ *a.e. which is not a viscosity solution of the equation* $|u'| = 1$.

# Viscosity solutions for the Eikonal equation (III)

*Intuitive idea of the proof (in the case $\Omega = (0,1) \subset \mathbb{R}$):*

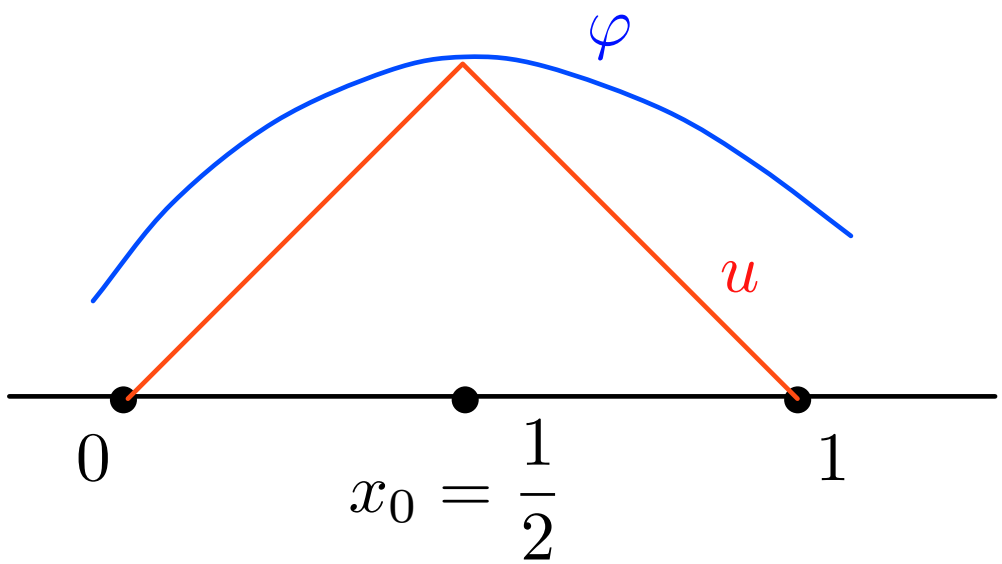
- We only prove that $u = d(x, \Gamma)$ is *one* viscosity solution (the uniqueness is admitted).
- At any point $x_0 \neq \frac{1}{2}$, $u$ is differentiable, with derivative $|u'(x)| = 1$.
- *Supersolution inequality:* there exists no function $\varphi$ of class $\mathcal{C}^2$ such that $(u - \varphi)$ has a local minimum (say 0) at $x_0$.




*No smooth function $\varphi$ has its graph under that of $u$ around $x_0$.*

- *Subsolution inequality:* let $\varphi$ be a function of class $\mathcal{C}^2$ such that $(u - \varphi)$ has a local maximum (say 0) at $x_0$.

  Then it is easily seen that $|\varphi'(x_0)| \leq 1$.




*Graph of a smooth function $\varphi$ such that $(u - \varphi)$ has local maximum 0 at $x_0$.*

# Part III

# Numerical resolution of the level set equation

# Solving the Level Set equations (I)

**Problem** The exact level set equation

$$\begin{cases} \frac{\partial \phi}{\partial t}(t,x) + V(t,x) \cdot \nabla \phi(t,x) = 0 & \text{for } (t,x) \in (0,T) \times \mathbb{R}^d, \\ \phi(t=0,x) = \phi_0(x) & \text{for } x \in \mathbb{R}^d. \end{cases} \tag{LS}$$

is too complicated for a general velocity $V(t,x)$ (depending on $\phi(t,x)$).

**Remedy** The time interval $(0,T)$ is split into a series of subintervals

$$(t^n, t^{n+1}), \text{ where } 0 = t^0 < t^1 < \ldots < t^N = T,$$

and $V(t,x)$ is approximated on each $(t^n, t^{n+1})$.

*Two such approximations are possible:*

1. The whole velocity field $V(t,x)$ is frozen over $(t^n, t^{n+1})$:

$$\forall t \in (t^n, t^{n+1}), \quad V(t,x) \approx V^n(x) := V(t^n, x),$$

and over each interval, a standard advection equation is solved:

$$\begin{cases} \frac{\partial \phi}{\partial t}(t,x) + V^n(x) \cdot \nabla \phi(t,x) = 0 & \text{on } (t^n, t^{n+1}) \times \mathbb{R}^d, \\ \phi(t=t^n,x) \text{ given} & \text{for } x \in \mathbb{R}^d. \end{cases} \tag{ADV}$$

# Solving the Level Set equations (II)

2. Only the normal component of $V(t,x) = v(t,x)n_t(x)$ is frozen:

$$\forall t \in (t^n, t^{n+1}),\ \ V(t,x) \approx v^n(x)n_t(x),\ \text{ where } v^n(x) = v(t^n,x).$$

Over each interval, a "classical" Hamilton-Jacobi equation is solved:

$$\begin{cases} \frac{\partial \phi}{\partial t}(t,x) + v^n(x)|\nabla \phi(t,x)| = 0 & \text{on } (t^n, t^{n+1}) \times \mathbb{R}^d, \\ \phi(t = t^n, x) \text{ given} & \text{for } x \in \mathbb{R}^d. \end{cases} \tag{HJ}$$

**Remarks:**

- Advection equations of the form (ADV) are quite well-known, and efficient numerical schemes exist for their resolution.
- The Hamilton-Jacobi formulation (HJ) preserves the information that the velocity field is consistently oriented along the normal vector $n_t(x)$ to $\Omega(t)$, and is thus appealing in many cases.

# Solving the Level Set equations (III)

- We focus on the resolution of the level set Hamilton-Jacobi equation over a generic time period $(0, T)$ ($=$ any of the $(t^n, t^{n+1})$ in the previous context):

$$\begin{cases} \dfrac{\partial \phi}{\partial t} + v(x)|\nabla \phi| = 0 & \text{on } (0,T) \times \mathbb{R}^d, \\ \phi(0,.) = \phi_0 & \text{on } \mathbb{R}^d, \end{cases}$$

  for given normal velocity field $v(x)$, and initial function $\phi_0$.

- The device of efficient algorithms for solving this equation relies on the theory of numerical schemes for first order Hamilton-Jacobi equations:

$$\begin{cases} \dfrac{\partial \phi}{\partial t} + H(x, \nabla \phi) = 0 & \text{on } (0,T) \times \mathbb{R}^d, \\ \phi(0,.) = \phi_0 & \text{on } \mathbb{R}^d, \end{cases} \tag{HJ}$$

  in the particular case where $H(x,p) = v(x)|p|$.

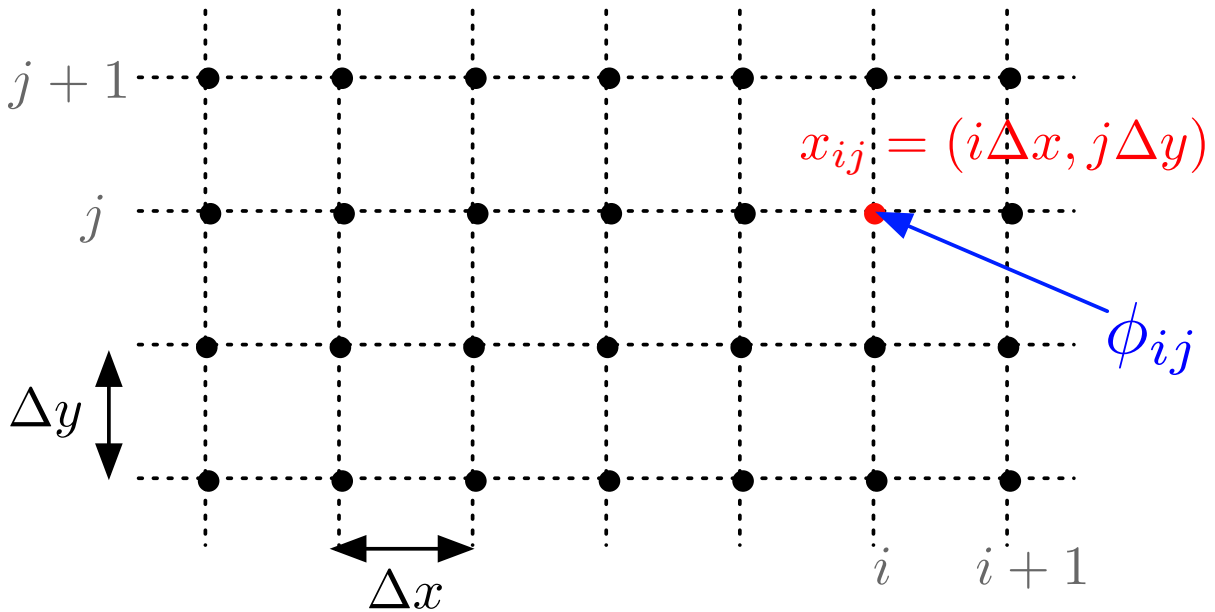
# Numerical setting

- We focus on the $2d$ situation.
- The time interval $(0, T)$ is split into $N = T/\Delta t$ subintervals:

$$(t^n, t^{n+1}), \text{ where } t^n = n\Delta t, \quad n = 0, ..., N,$$

and $\Delta t$ is a time step.

- The space is discretized by a Cartesian grid with steps $\Delta x, \Delta y$.



- For $i, j \in \mathbb{Z}$, we denote the finite difference quantities:

$$D_{ij}^{+x}\phi = \frac{\phi_{i+1j} - \phi_{ij}}{\Delta x} \quad ; \quad D_{ij}^{-x}\phi = \frac{\phi_{ij} - \phi_{i-1j}}{\Delta x},$$

and:

$$D_{ij}^{+y}\phi = \frac{\phi_{ij+1} - \phi_{ij}}{\Delta y} \quad ; \quad D_{ij}^{-y}\phi = \frac{\phi_{ij} - \phi_{ij-1}}{\Delta y}.$$

- An explicit, first-order scheme for the Hamilton-Jacobi equation reads:

$$\begin{cases} \forall i, j \in \mathbb{Z}, & \phi_{ij}^0 = \phi_0(i\Delta x, j\Delta y), \\ \forall n \in \mathbb{N}, i, j \in \mathbb{Z}, & \phi_{ij}^{n+1} = \phi_{ij}^n - \Delta t\, \mathcal{H}\left(x_{ij}, D_{ij}^{-x}\phi^n, D_{ij}^{+x}\phi^n, D_{ij}^{-y}\phi^n, D_{ij}^{+y}\phi^n\right), \end{cases}$$

where the numerical Hamiltonian

$$\mathcal{H}\left(x_{ij}, D_{ij}^{-x}\phi^n, D_{ij}^{+x}\phi^n, D_{ij}^{-y}\phi^n, D_{ij}^{+y}\phi^n\right)$$

is intended as an approximation of $H(x_{ij}, \nabla\phi(x_{ij}))$.

## Definition 4.

*A numerical scheme of the above form is said to be:*

- *consistent if, for any* $x \in \mathbb{R}^2$ *and* $p \in \mathbb{R}^2$, $\mathcal{H}(x, p_x, p_x, p_y, p_y) = H(x, p)$.
- *monotone if, for any* $x \in \mathbb{R}^2$, *and any* $i, j \in \mathbb{Z}$, *the update function*

$$\{\phi_{kl}\}_{k,l\in\mathbb{Z}} \longmapsto \phi_{ij} - \Delta t\, \mathcal{H}\left(x, D_{ij}^{-x}\phi, D_{ij}^{+x}\phi, D_{ij}^{-y}\phi, D_{ij}^{+y}\phi\right)$$

*is increasing with respect to each of its arguments.*

## Theorem 2.

*Under mild, technical hypotheses on* $H$ *and* $\phi_0$, *first-order consistent and monotone numerical schemes converge to the viscosity solution to* (*HJ*).

# Sethian's first-order scheme (I)

In the particular case of interest, $H(x,p) = v(x)|p|$ and (HJ) reads:

$$\begin{cases} \dfrac{\partial \phi}{\partial t} + v(x)|\nabla \phi| = 0 & \text{on } (0,T)\times \mathbb{R}^d, \\ \phi(0,.) = \phi_0 & \text{on } \mathbb{R}^d. \end{cases}$$

We introduce the numerical scheme:

$$\begin{cases} \forall n \in \mathbb{N}, i,j \in \mathbb{Z}, & \phi_{ij}^{n+1} = \phi_{ij}^n - \Delta t \left( \max(v_{ij},0)\nabla_{ij}^+ \phi^n + \min(v_{ij},0)\nabla_{ij}^- \phi^n \right), \\ \forall i,j \in \mathbb{Z}, & \phi_{ij}^0 = \phi_0(i\Delta x, j\Delta y), \end{cases}$$

with the discretizations $\nabla_{ij}^+ \phi$ and $\nabla_{ij}^- \phi$ of the gradient norm $|\nabla \phi|$ defined by:

$$\nabla_{ij}^+ \phi = \left( \begin{array}{c} \max(\max(D_{ij}^{-x}\phi,0), -\min(D_{ij}^{+x}\phi,0))^2 \\ +\max(\max(D_{ij}^{-y}\phi,0), -\min(D_{ij}^{+y}\phi,0))^2 \end{array} \right)^{\frac{1}{2}},$$

and

$$\nabla_{ij}^- \phi = \left( \begin{array}{c} \max(\max(D_{ij}^{+x}\phi,0), -\min(D_{ij}^{-x}\phi,0))^2 \\ +\max(\max(D_{ij}^{+y}\phi,0), -\min(D_{ij}^{-y}\phi,0))^2 \end{array} \right)^{\frac{1}{2}}.$$

# Sethian's first-order scheme (II)

- The quantity $\nabla_{ij}^{+}\phi$ (resp. $\nabla_{ij}^{-}\phi$) is upwind (resp. downwind): it is a finite difference approximation of $|\nabla\phi|$ at $x_{ij}$ based only on the values among $\{\phi_{i-1j}, \phi_{i+1j}, \phi_{ij-1}, \phi_{ij+1}\}$ which are smaller (resp. larger) than $\phi_{ij}$.

- The discretization of the (exact) Hamiltonian $H(x,p) = v(x)|p|$ by the numerical counterpart:

$$H(x_{ij}, \nabla\phi(x_{ij})) \approx \mathcal{H}_{ij}(\{\phi_{kl}^{n}\}_{k,l\in\mathbb{Z}}) := \max(v_{ij}, 0)\nabla_{ij}^{+}\phi^{n} + \min(v_{ij}, 0)\nabla_{ij}^{-}\phi^{n}$$

  is upwind: for given $i, j, n$, the update $\phi_{ij}^{n} \to \phi_{ij}^{n+1}$ is only carried out using information coming from

  - smaller values than $\phi_{ij}^{n}$ if $v_{ij}$ is positive,
  - larger values than $\phi_{ij}^{n}$ if it is negative.

# Sethian's first-order scheme (III)

- Sethian's first-order scheme is consistent:

$$\forall x \in \mathbb{R}^d, \forall p = (p_x, p_y) \in \mathbb{R}^2, \ \mathcal{H}(x, p_x, p_x, p_y, p_y) = v(x)|p|.$$

- It is monotone, provided the following CFL-like condition is fulfilled:

$$\left(\sup_{i,j} v_{ij}\right) \frac{\Delta t}{\min(\Delta x, \Delta y)} \leq 1, \text{ i.e.}$$

*"The information cannot travel more than one cell during one time step".*

- It is therefore convergent (under the CFL condition).

- In addition, the following error estimate can be proved between the numerical result $\{\phi_{ij}\}$ of Sethian's scheme, and the exact viscosity solution $\phi(t, x)$:

$$\forall i, j \in \mathbb{Z}, \ \forall n \leq N, \ |\phi_{ij}^n - \phi(t^n, x_{ij})| \leq C\sqrt{\Delta t}$$

The time accuracy can be increased thanks to the Runge-Kutta methodology, applied below, *for simplicity*, to the device of a second-order in time scheme.

1. An attempt step $\phi_{ij}^n \to \widetilde{\phi_{ij}^{n+1}}$ is performed for the value of $\phi$ at time $t^{n+1}$, using the previous first-order scheme:

$$\widetilde{\phi_{ij}^{n+1}} = \phi_{ij}^n - \Delta t \left( \max(v_{ij}, 0) \nabla_{ij}^+ \phi^n + \min(v_{ij}, 0) \nabla_{ij}^- \phi^n \right).$$

2. Another attempt step $\widetilde{\phi_{ij}^{n+1}} \to \widetilde{\phi_{ij}^{n+2}}$ is performed for an approximation of the value of $\phi$ at $t^{n+2}$:

$$\widetilde{\phi_{ij}^{n+2}} = \widetilde{\phi_{ij}^{n+1}} - \Delta t \left( \max(v_{ij}, 0) \nabla_{ij}^+ \widetilde{\phi_{ij}^{n+1}} + \min(v_{ij}, 0) \nabla_{ij}^- \widetilde{\phi_{ij}^{n+1}} \right).$$

3. The actual update $\phi_{ij}^n \to \phi_{ij}^{n+1}$ is obtained by averaging:

$$\phi_{ij}^{n+1} = \frac{1}{2} \phi_{ij}^n + \frac{1}{2} \widetilde{\phi_{ij}^{n+2}}.$$

# Towards increased space accuracy

- The space accuracy can be enhanced by using a higher order discretization of the derivatives of $\phi$ instead of the previous first-order formulae

$$D_{ij}^{+x}\phi = \frac{\phi_{i+1j} - \phi_{ij}}{\Delta x} \quad ; \quad D_{ij}^{-x}\phi = \frac{\phi_{ij} - \phi_{i-1j}}{\Delta x},$$

  and $D_{ij}^{-y}\phi$, $D_{ij}^{+y}\phi$.

- This discretization should take great care of the fact that $\phi$ may be singular in some regions of space.

- To achieve this, the idea of Essentially Non Oscillatory (ENO) finite differences consists in:

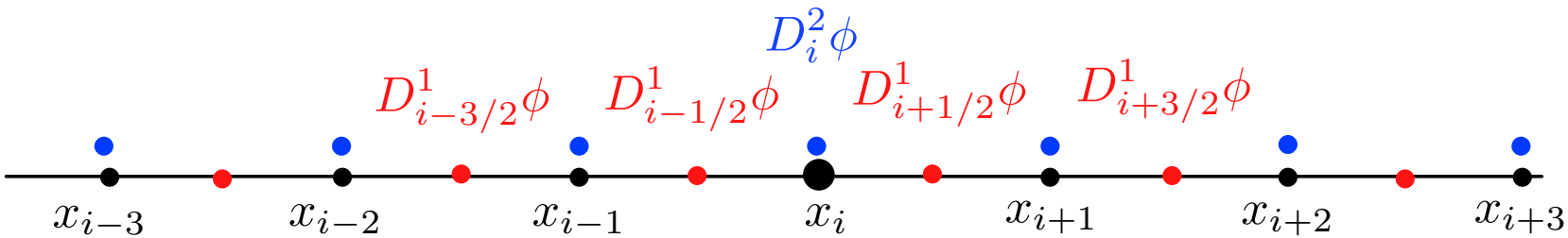
  1. Constructing a (second-, third-order) polynomial approximation $P$ of $\phi$ around the considered node $x_{ij}$, by using only information from the nodes around $x_{ij}$ where $\phi$ is "smooth enough".
  2. Calculating $D_{ij}^{\pm x}\phi$, $D_{ij}^{\pm y}\phi$ as the derivatives of $P$.

# ENO reconstruction in one dimension (I)

- *Setting:* The real line $\mathbb{R}$ is subdivided with a set of nodes $x_i = i\Delta x$, $i \in \mathbb{Z}$, and a numerical quantity $\{\phi_i\}_{i\in\mathbb{Z}}$ is defined at these nodes.
- Information about the derivatives of $\phi$ is approximated at several nodes around that of interest; for instance:

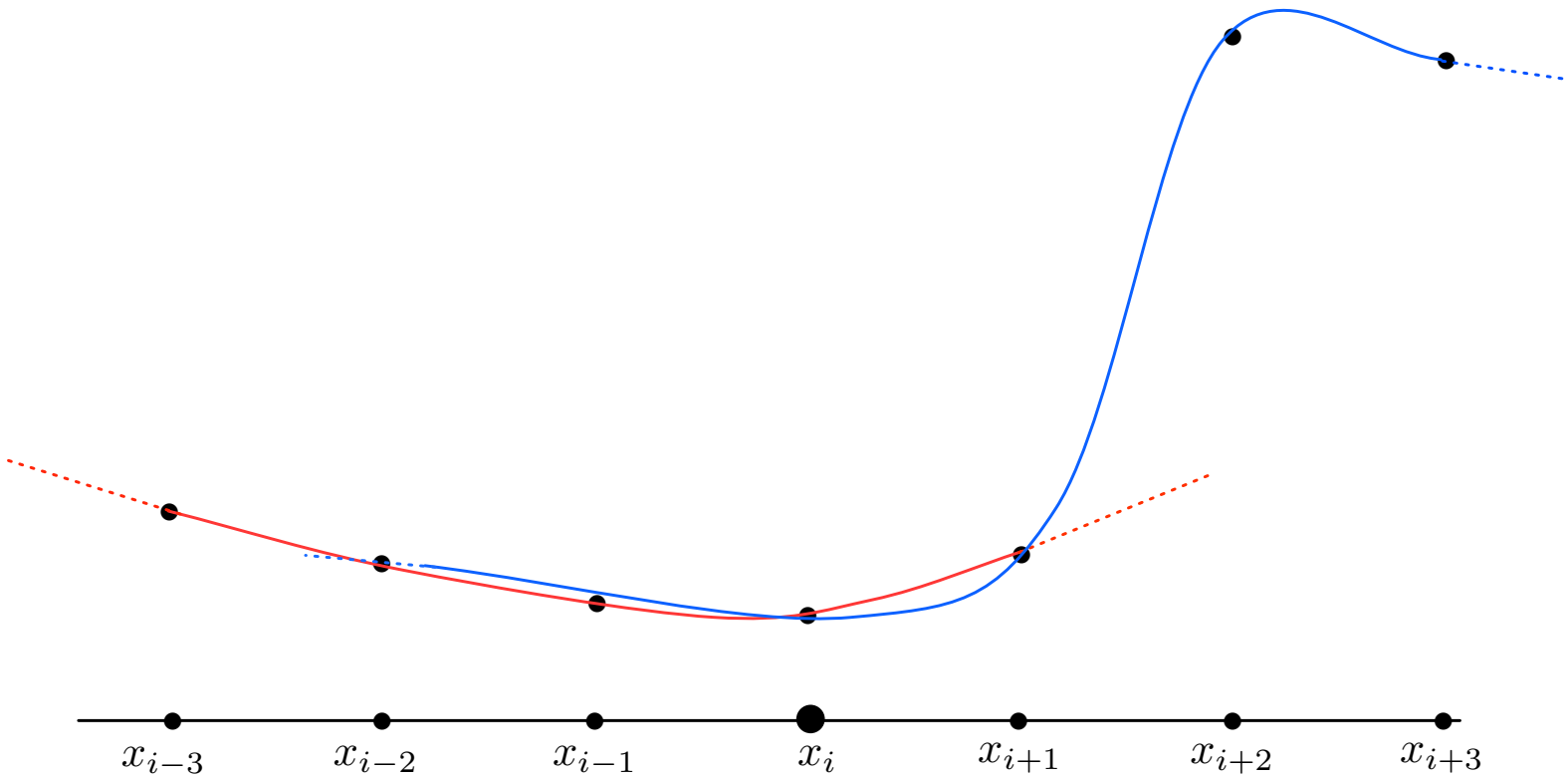
$1^{\text{st}}$ *divided differences:* $D^1_{i-1/2}\phi = \dfrac{\phi_i - \phi_{i-1}}{\Delta x}$, $D^1_{i+1/2}\phi = \dfrac{\phi_{i+1} - \phi_i}{\Delta x}$,

$2^{\text{nd}}$ *divided differences:* $D^2_i\phi = \dfrac{D^1_{i+1/2}\phi - D^1_{i-1/2}\phi}{2\Delta x}$.



# ENO reconstruction in one dimension (II)

- A polynomial $P(x)$ of degree e.g. 3 or 4 is fitted to the data by selecting some of these derivatives, so that $P(x)$ does not present too steep variations (which could account for a region of discontinuity of $\phi$).




*Different reconstructions of the data $\phi$ (black dots) using different stencils; the blue reconstruction is polluted by the presence of a shock.*

# ENO reconstruction in one dimension (III)

- Taking the derivative of $P$ at $x_i$ results in an explicit, high order formula for the derivative of the numerical quantity $\{\phi_i\}_{i\in\mathbb{Z}}$, with adaptive stencil.

- That the stencil may change from one evaluation of the derivatives of $\phi$ to another is undesirable:
  - The convergence analysis of ENO schemes is difficult,
  - This "lack of smoothness" in the stencil selection procedure causes trouble in applications to hyperbolic PDE,
  - In practice, the stencil could change just because of round-off errors.

  As a remedy, Weighted ENO schemes (WENO) feature a convex combination of several reconstruction formulae of the previous form (with different stencils), the weights of each particular reconstruction depending on the local smoothness of $\phi$.

# Other numerical methods

- Different techniques are needed when the computational support is a triangulation instead of a Cartesian grid, e.g.:
  - A generalization of the above concepts of consistency and monotonicity, paving the way for new rules for devising convergent schemes - see [Abgrall],
  - Stabilized (Petrov-Galerkin) finite element formulations for the Hamilton-Jacobi equations, where some quadratic terms are added to their variational formulation to penalize oscillations; see [Barth].
- Semi-Lagrangian schemes (see [Strain]) use the direction along which the information is conveyed by Hamilton-Jacobi equations, grossly speaking by backtracking the corresponding characteristic curves of the equation.

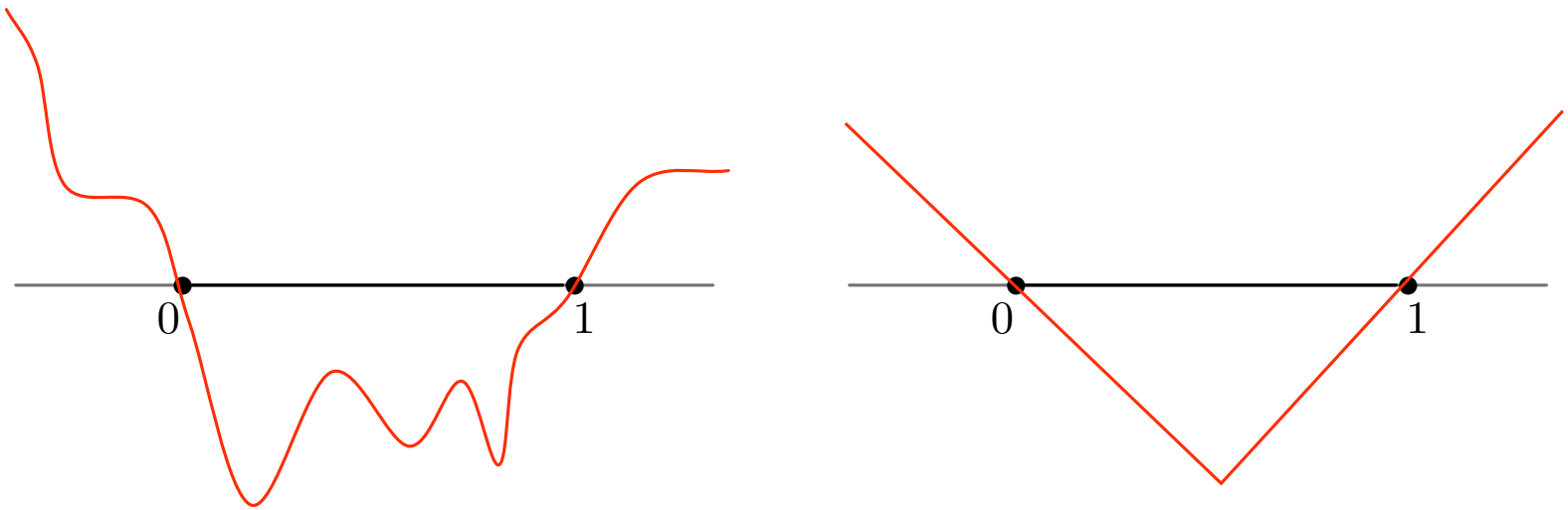
  This idea can be worked out whatever the computational support.

# Part IV

# Initialization of a level set function

# Initializing level set functions: the signed distance function (I)

- Let $\Omega \subset \mathbb{R}^d$ be a (smooth, bounded) domain. We seek to construct an associated Level Set function $\phi : \mathbb{R}^d \to \mathbb{R}$.
- There are "a lot" of level set functions associated to a given domain $\Omega$.




*Two level set functions for the domain* $\Omega = (0, 1) \subset \mathbb{R}$.

- The theoretical framework of the level set method is independent of which particular level set function is used.

## Definition 5.

*Let $\Omega \subset \mathbb{R}^d$ be a domain. The signed distance function $d_\Omega$ to $\Omega$ is defined by:*

$$d_\Omega(x) = \begin{cases} -d(x,\Gamma) & \text{if } x \in \Omega, \\ 0 & \text{if } x \in \Gamma, \\ d(x,\Gamma) & \text{if } x \in {}^c\overline{\Omega}. \end{cases},$$

*where $d(\cdot,\Gamma)$ is the usual Euclidean distance function to $\Gamma$:*

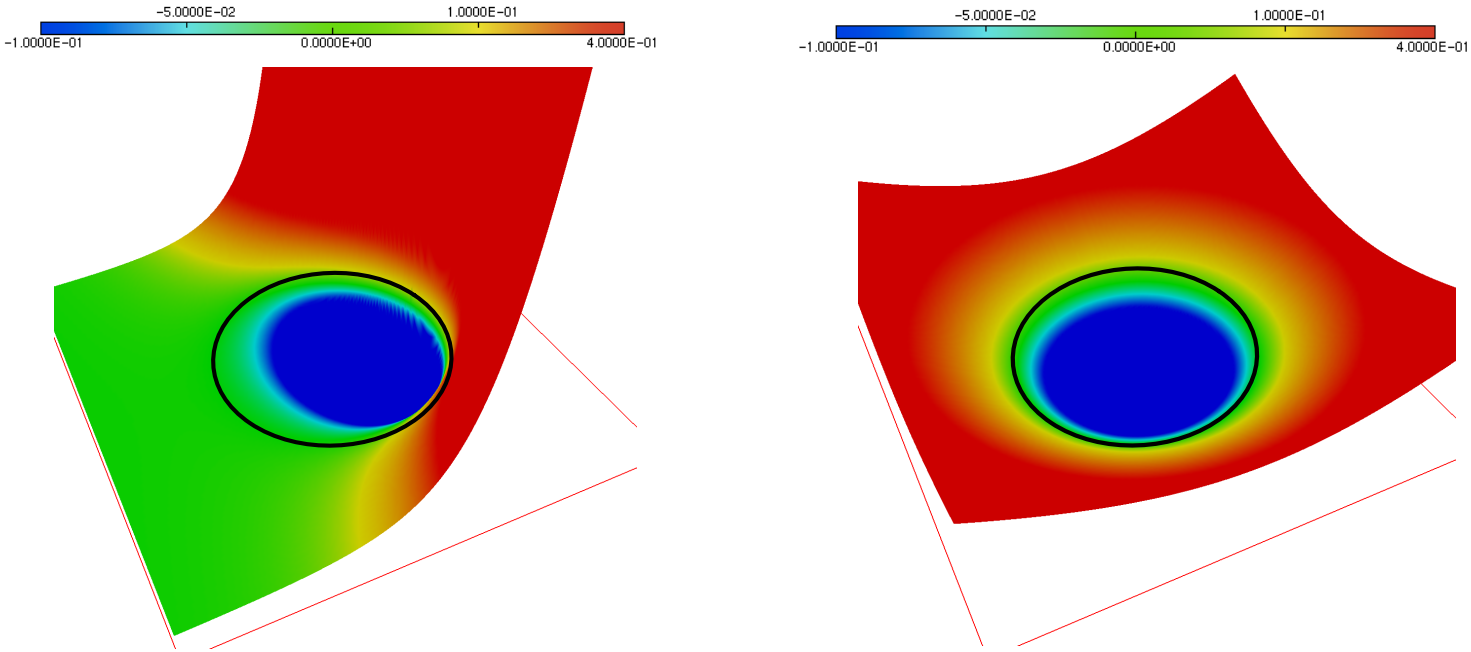
$$d(x,\Gamma) = \inf_{y\in\Gamma} |x-y|.$$

**Remarks**

- $d_\Omega$ is Lipschitz continuous (*exercise*).
- From Rademacher's theorem, it is almost everywhere differentiable.
- Wherever it makes sense, its gradient has unit norm:

$$|\nabla d_\Omega(x)| = 1 \text{ a.e. on } \mathbb{R}^d.$$

# Initializing level set functions: the signed distance function (III)




*Graphs of (left) one very steep level set function associated to a disk, (right) the signed distance function to the disk.*

# The fast marching algorithm

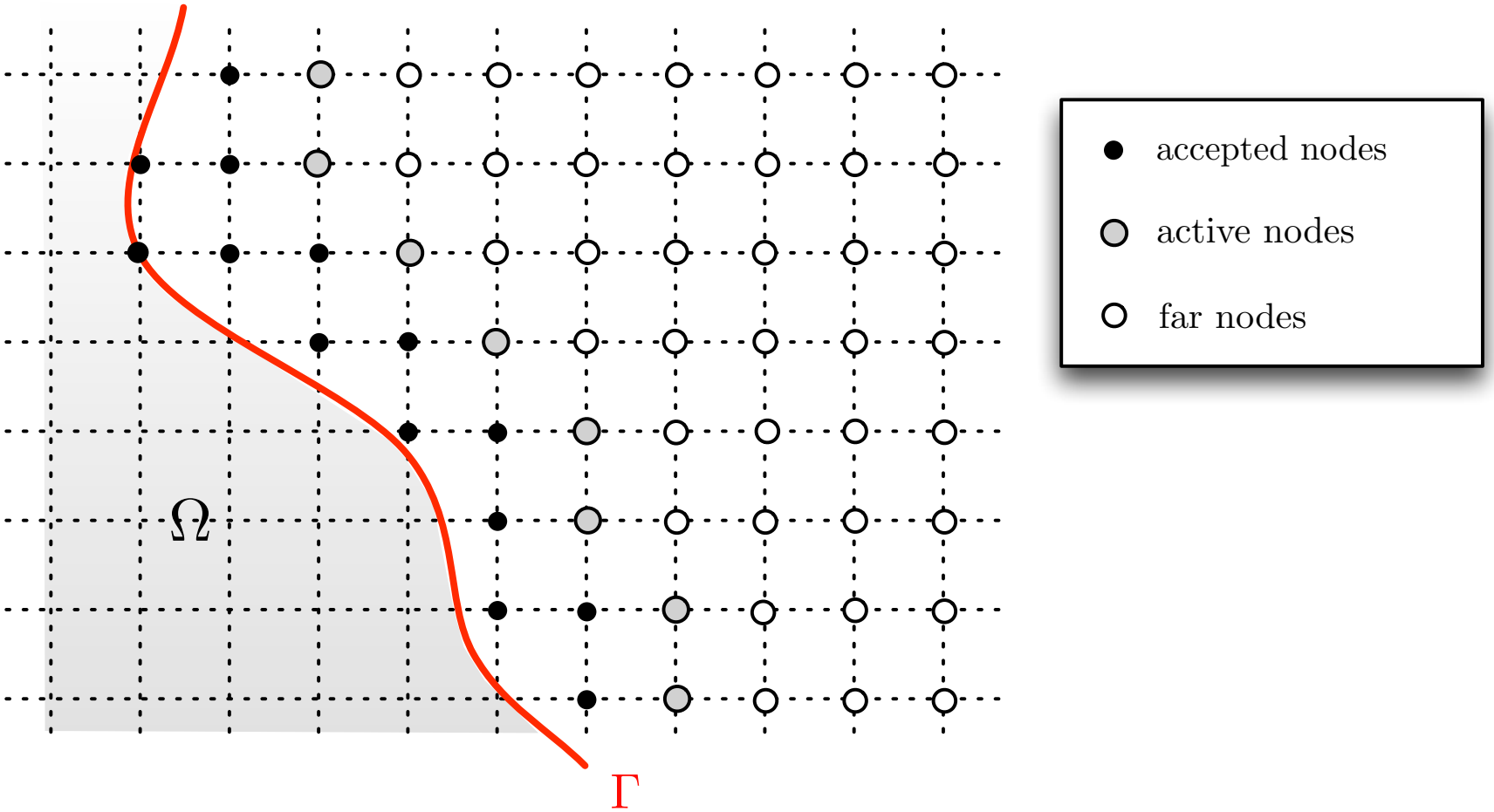
- The most celebrated method to calculate (signed) distance functions is the fast marching method, introduced by Sethian in [SethianFMM].
- *Setting:*
  - $\Omega$ is a $2d$ domain, and the (unsigned) distance function $d(\cdot, \Gamma)$ is calculated on the outer domain $\mathbb{R}^d \setminus \overline{\Omega}$.
  - The plane is again equipped with a Cartesian grid, whose nodes are denoted $x_{ij} = (i\Delta x, j\Delta y)$, for $i, j \in \mathbb{Z}$.
  - The fast marching method produces, at each iteration $n = 0, \ldots$ a numerical quantity $\left\{ T_{ij}^n \right\}_{i,j \in \mathbb{Z}}$, intended as an increasingly accurate approximation of $d(\cdot, \Gamma)$.
- The fast marching method is a combination of two ingredients:
  - A numerical discretization of the Eikonal equation $|\nabla T| = 1$, used to update the values $T_{ij}^n \mapsto T_{ij}^{n+1}$ from the values $T_{kl}^n$ at neighbors $x_{kl}$ of $x_{ij}$.
  - A marching procedure, giving an order for accepting values.

# The marching procedure (I)

The fast marching method mimicks the propagation of a front.

- The nodes $x_{ij}$ of the grid are consistently classified into 3 categories:
  - The accepted nodes $x_{ij}$ are those "where the front has already passed". The value $T_{ij}^n$ is assumed to have converged and is no longer updated.
  - The active nodes $x_{ij}$ are those "on the front". One of their 4 neighbors $x_{i-1j}, x_{i+1j}, x_{ij-1}$ or $x_{ij+1}$ is accepted, and a first trial value $T_{ij}^n$ has been computed, but may still be subject to updates.
  - The far nodes are those $x_{ij}$ for which no *trial value* is available: $T_{ij}^n = \infty$.
- At each iteration $n \to n+1$, the algorithm
  1. *accepts* one active node, that with the smallest trial value,
  2. redefines the set of active nodes (i.e. tags *active* those who are neighbor to the newly accepted node),
  3. calculates new trial values where need be, using the update procedure.

# The marching procedure (II)




*Setting of the fast marching method*

# The local update

- At an iteration $n \to n+1$, a temporary value $\widetilde{T}_{ij}^n$ is calculated at each active node $x_{ij}$, thanks to a discretization of the Eikonal equation:

$$|\nabla T(x)| = 1.$$

- The discretization is:

$$\sqrt{\begin{array}{l} \max\left(\max\left(\frac{\widetilde{T}_{ij}^n - T_{i-\mathbf{1}j}^n}{\Delta x}, 0\right), -\min\left(\frac{T_{i+\mathbf{1}j}^n - \widetilde{T}_{ij}^n}{\Delta x}, 0\right)\right)^2 \\ + \max\left(\max\left(\frac{\widetilde{T}_{ij}^n - T_{ij-\mathbf{1}}^n}{\Delta y}, 0\right), -\min\left(\frac{T_{ij+\mathbf{1}}^n - \widetilde{T}_{ij}^n}{\Delta y}, 0\right)\right)^2 \end{array}} = 1.$$

- The calculation of $\widetilde{T}_{ij}^n$ from the $T_{kl}^n$ is upwind:
  - Only the accepted values within the set $\left\{T_{i-1j}^n, T_{i+1j}^n, T_{ij-1}^n, T_{ij+1}^n\right\}$ are used in the above formula.
  - Only solutions $\widetilde{T}_{ij}^n$ larger than these accepted values are retained.

- In the end, the new trial value $T_{ij}^{n+1}$ is obtained as:

$$T_{ij}^{n+1} = \min\left(\widetilde{T}_{ij}^n, T_{ij}^n\right).$$

# The fast marching algorithm

- **Initialization:**
  1. Compute the exact distance function at the nodes of the cells which intersect $\Gamma$, and mark them as accepted.
  2. Use the local update procedure to compute a trial value at the neighbor of the accepted points which are not accepted, and mark them as active.
  3. Mark all the remaining nodes as far, and assign them the value $\infty$.

- **Loop (while the set of active nodes is non empty):**
  1. Travel the set of active nodes, and identify the one with minimum trial value. This node becomes accepted.
  2. Identify the new set of active nodes, and compute a new trial value for each one of them, using the local update solver for the Eikonal equation.

# Comments

- The method extends straightforwardly to general Eikonal equations:

$$c(x)|\nabla T(x)| = 1, \text{ where } c(x) > 0.$$

- *Computational cost:* The fast marching method requires $\mathcal{O}(M \log(M))$ operations, where $M$ is the number of nodes in the grid:
  - During every iteration, one value is accepted.
  - The only costly operation within one iteration consists in searching in the list of trial values which is the smallest.
  - In practice, a heapsort algorithm is used to make this search effficient - in $\mathcal{O}(\log(\widetilde{M}))$, where $\widetilde{M}$ is the number of trial values.

- Under mild hypotheses, one proves that the fast marching algorithm converges to the solution to the Eikonal equation.

# Extensions

- The fast marching method extends fairly straightforwardly to the case of a Cartesian grid in $3d$.

- It can also be extended (with some adjustments) to the cases of:
  - A triangular mesh of the computational domain in $\mathbb{R}^2$,
  - A triangulated surface embedded in $\mathbb{R}^3$,
  - A tetrahedral mesh in $\mathbb{R}^3$.

- Other algorithms are available to calculate (signed) distance functions, e.g. the fast sweeping method [Zhao].

# Part V

# Miscellanies

# Miscellanies

- We have seen how well-adapted the level set framework is when it comes to describing the evolution of a domain $\Omega(t)$, however dramatic (even if it involves topological changes).

- On the other hand, several operations to be performed on $\Omega(t)$ may be difficult to carry out in this implicit framework, since $\Omega(t)$ is not explicitely discretized.

  $\Rightarrow$ Need for numerical tricks to perform these operations.

- In addition, several complementary features make it possible to substantially improve the performance of the level set method.

# Operations within the level set framework (I)

1. *Evaluation of the normal vector, or the curvature of a domain.*

Let $\Omega \subset \mathbb{R}^d$ be a domain, $\phi$ be an associated level set function.

- The normal vector $n(x)$ to $\Gamma$, pointing outward $\Omega$, is approximated as:

$$n(x) \approx \frac{\nabla\phi(x)}{\sqrt{|\nabla\phi(x)|^2+\varepsilon^2}}, \text{ for some } \varepsilon \ll 1.$$

  This formula is discretized depending on the computational support, e.g.:
  - by using standard first-order finite differences, or a higher-order ENO approximation on a Cartesian grid,
  - by using $\mathbb{P}^1$ interpolation on a triangular mesh.

- The mean curvature $\kappa$ of $\Gamma$ is approximated as:

$$\kappa(x) \approx \operatorname{div}\left(\frac{\nabla\phi(x)}{\sqrt{|\nabla\phi(x)|^2+\varepsilon^2}}\right).$$

# Operations within the level set framework (II)

❷ *Evaluation of integrals on $\Omega$ or $\Gamma$.*

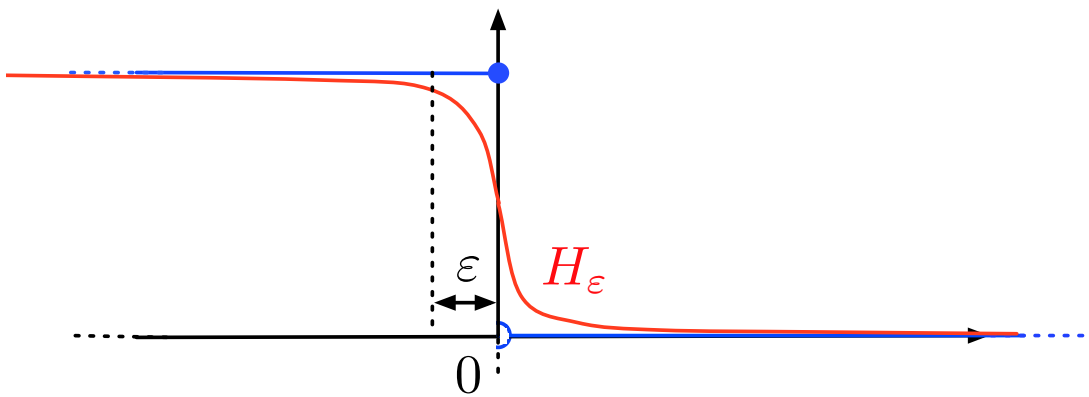
Let $f : \mathbb{R}^d \to \mathbb{R}$ be a smooth function; we aim to calculate $I = \int_\Omega f(x)\,\mathrm{d}x$.

- We first devise an approximate characteristic function of $\Omega$:

$$\forall x \in \mathbb{R}^d,\ \ \chi_\Omega(x) \approx H_\varepsilon(\phi(x)),\ \text{ where } H_\varepsilon(t) := \frac{1}{2}\left(1 - \frac{t}{\sqrt{t^2+\varepsilon^2}}\right).$$

- The resulting approximation of $I$ reads:

$$I \approx \int_{\mathbb{R}^d} f(x) H_\varepsilon(\phi(x))\,\mathrm{d}x.$$




*Approximation $H_\varepsilon$ (in red) of the characteristic function of $(-\infty, 0)$ (in blue).*

# Operations within the level set framework (III)

❷ *Evaluation of integrals on $\Omega$ or $\Gamma$ (continued).*

Let $g : \mathbb{R}^d \to \mathbb{R}$ be a smooth function; we aim to calculate $J = \int_\Gamma g(x)\,\mathrm{d}s$.

- We rely on an approximation of the surface measure distribution $\delta_\Gamma$ on $\Gamma$:

$$\forall \varphi \in \mathcal{C}_c^\infty(\mathbb{R}^d),\ \langle \delta_\Gamma, \varphi \rangle = \int_\Gamma \varphi\,\mathrm{d}s.$$

- A use of Green's formula reveals that, in the sense of distributions:

$$\delta_\Gamma = -\frac{\partial \chi_\Omega}{\partial n} \approx -\frac{\partial}{\partial n}(H_\varepsilon(\phi)).$$

- The resulting formula for the calculation of $J$ is:

$$J \approx -\int_{\mathbb{R}^d} \frac{\partial}{\partial n}(H_\varepsilon(\phi(x)))\, g(x)\,\mathrm{d}x.$$

# Operations within the level set framework (IV)

❸ *Algebraic operations over sets.*

Let $\Omega, \Omega_1, \Omega_2 \subset \mathbb{R}^d$ be domains, and $\phi, \phi_1, \phi_2$ be associated Level Set functions.

- One level set function $\phi_c$ for the complement ${}^c\overline{\Omega}$ of $\Omega$ is:

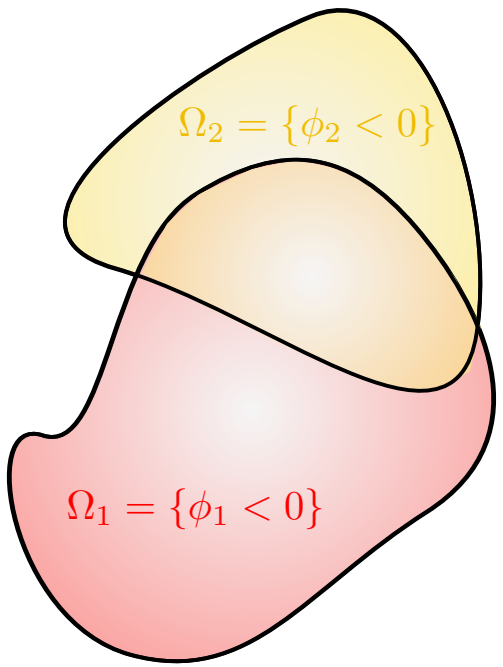
$$\phi_c = -\phi.$$

- One level set function $\phi_u$ for the reunion $\Omega_1 \cup \Omega_2$ is:

$$\phi_u = \min(\phi_1, \phi_2).$$

- One level set function $\phi_i$ for the intersection $\Omega_1 \cap \Omega_2$ is:

$$\phi_u = \max(\phi_1, \phi_2).$$



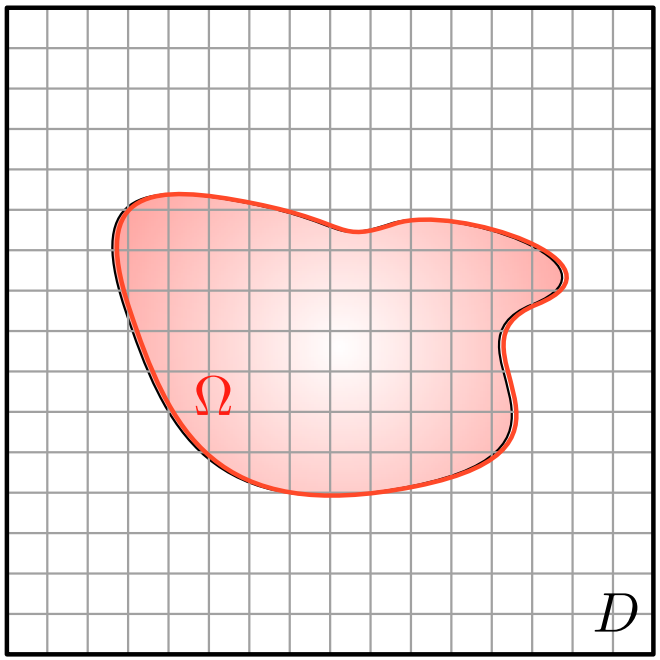
# Operations within the Level Set framework (V)

4. *Solving PDE on the domain $\Omega$.*

Consider the frequent situation:

- A domain $\Omega$ lies in a computational box $D$, equipped with a mesh $\mathcal{T}$.
- $\Omega$ is solely known via a level set function $\phi$ defined on $\mathcal{T}$.
- We aim to solve the PDE:

$$\begin{cases} -\mathrm{div}(a\nabla u) + u = f & \text{in } \Omega, \\ \frac{\partial u}{\partial n} = 0 & \text{on } \Gamma. \end{cases}$$




*A domain $\Omega$, included in the computational box $D$, equipped with a Cartesian grid. No mesh of $\Omega$ is available.*

# Operations within the level set framework (VI)

4. *Solving PDE on the domain $\Omega$ (continued).*

- **Idea:** Approximate $u$ with a function $u_\varepsilon$, solution to a PDE posed on $D$.
- For instance, let $u_\varepsilon \in H^1(D)$ be the solution to the system:

$$\begin{cases} -\mathrm{div}(c_\varepsilon a \nabla u) + c_\varepsilon u = c_\varepsilon f & \text{in } D, \\ \frac{\partial u}{\partial n} = 0 & \text{on } \partial D \end{cases}, \text{ where } c_\varepsilon(x) = \begin{cases} 1 & \text{for } x \in \Omega, \\ \varepsilon & \text{for } x \in D \setminus \Omega \end{cases};$$

In other words,

*The void $D \setminus \overline{\Omega}$ is filled with a material of very small conductivity $\varepsilon \ll 1$.*

- It is possible to prove that:

$$||u - u_\varepsilon||_{H^1(\Omega)} \overset{\varepsilon \to 0}{\longrightarrow} 0.$$

- The function $u_\varepsilon$ can be calculated as an approximation of $u$ by solving the corresponding PDE on $D$.

# Level set redistancing (I)

- We have highlighted the importance, in practice, that the level set function $\phi(t, \cdot)$ of $\Omega(t)$ stay "close" to a signed distance function for $t \geq 0$.
- Unfortunately, even if the initial level set function $\phi_0$ is a signed distance function, $\phi(t, \cdot)$ is bound not to stay so.
- In practice, it is a very important feature to restore periodically $\phi$ to a signed distance function.
- One could simply generate the signed distance function, e.g. by using the fast marching method.
- However, the situation is pretty different from that of the initialization: we have one level set function at hand; it would be a pity not to exploit this fact.

# Level set redistancing (II)

- Let $\Omega \subset \mathbb{R}^d$ be a domain, $\phi_0$ be an associated level set function (with possibly very steep, or flat variations).

- $\phi_0$ is used as the initial state of the redistancing equation:

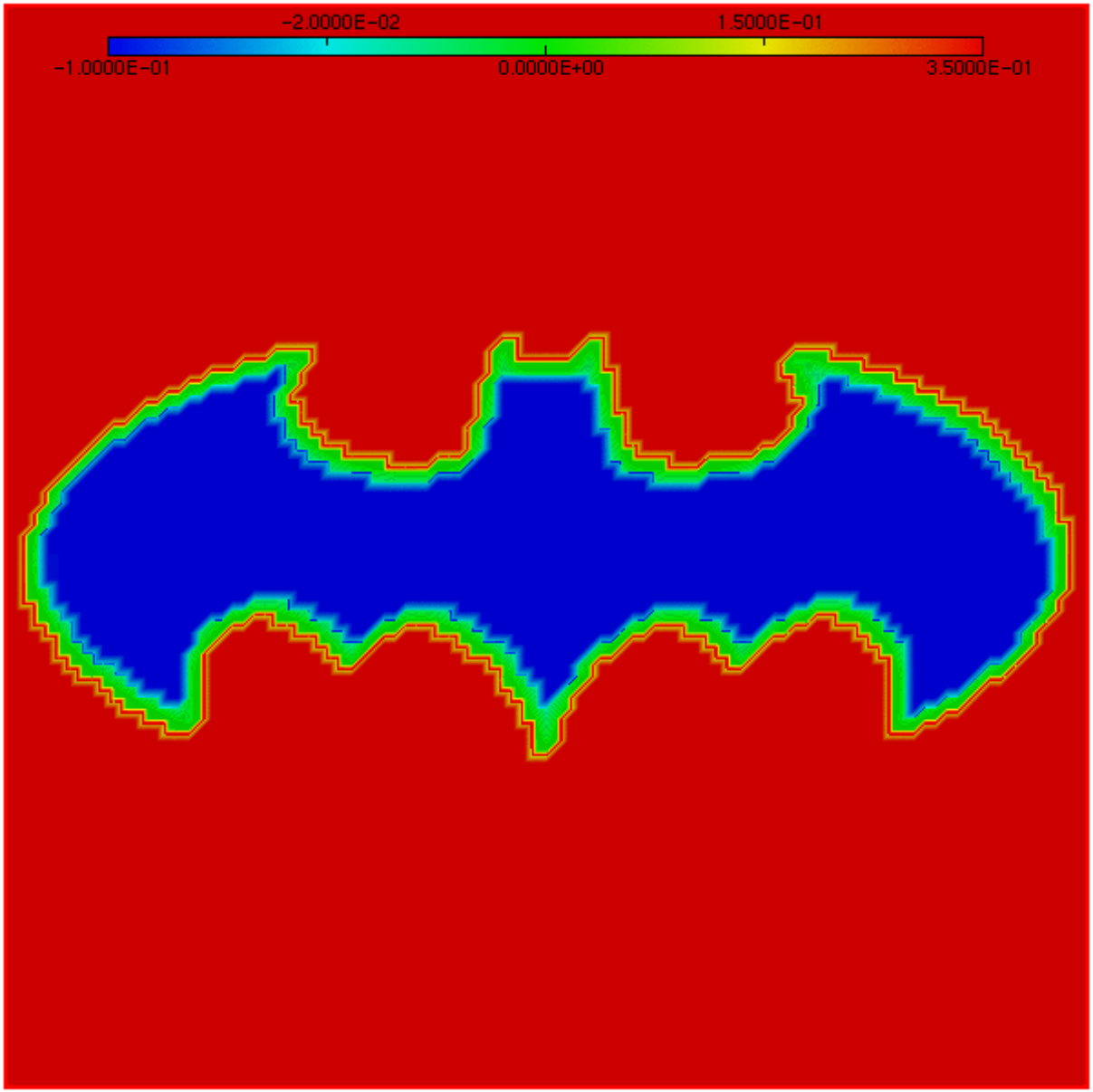
$$\begin{cases} \frac{\partial \psi}{\partial t}(t,x) + \mathrm{sgn}(\phi_0(x))\,(|\nabla \psi| - 1) = 0 & \text{for } (t,x) \in (0,\infty) \times \mathbb{R}^d \\ \psi(0,x) = \phi_0(x) & \text{for } x \in \mathbb{R}^d \end{cases}.$$

- Formally, the steady state $\widetilde{\psi}$ of this equation satisfies

$$|\nabla \widetilde{\psi}| - 1 = 0, \text{ and } \widetilde{\psi}(x) = 0 \text{ on } \Gamma.$$

- A study of this equation reveals that $\phi_0$ is steadily "regularized" into the signed distance function $d_\Omega$, starting from $\Gamma$, to the region far from $\Gamma$.

# Level set redistancing (III)



# Level set redistancing (IV)

*Practical use:*

- In the numerical resolution of the level set equation

$$\begin{cases} \frac{\partial \phi}{\partial t} + v|\nabla \phi| = 0 & \text{on } (0, T) \times \mathbb{R}^d, \\ \phi(t = 0, \cdot) = \phi_0 & \text{on } \mathbb{R}^d, \end{cases}$$

  periodically interrupt the process (say, every 4-5 iterations).

- At a corresponding time $t^n$, solve the redistancing equation:

$$\begin{cases} \frac{\partial \psi}{\partial t}(t, x) + \text{sgn}(\phi(t^n, x)\ (|\nabla \psi| - 1) = 0 & \text{for } (t, x) \in (0, \infty) \times \mathbb{R}^d \\ \psi(0, x) = \phi(t^n, x) & \text{for } x \in \mathbb{R}^d \end{cases}$$

  over a short period of time $(0, t^*)$, using a numerical scheme in the spirit of those presented above.

- Trade $\phi(t^n, \cdot)$ for $\psi(t^*, \cdot)$, and resume the resolution of the level set equation.

# Velocity extension

- The level set method requires $V(t,x)$ to be defined on the whole ambient space $\mathbb{R}^d$ (actually, a narrow band around $\Gamma(t)$ is enough).
- Unfortunately, in many applications, $V(t,x)$ only makes sense for $x \in \Gamma(t)$, e.g. when it involves the normal vector $n_t(x)$, the mean curvature $\kappa_t(x)$, etc.
- Actually, even when $V(t,x)$ can be "naturally" extended outside $\Gamma(t)$, this extension is often ill-suited, e.g. it anticipates the stretching of the level set function $\phi(t,x)$.
- On the contrary, there is a great latitude on how to extend $V(t,x)$ for $x \notin \Gamma(t)$; the only strong requirement is that it should coincide with $V(t,x)$ on $\Gamma(t)$.
- We present two possibilities to achieve this velocity extension, with competing assets.

# Normal extension of the velocity field (I)

- This extension "alleviates" the need for redistancing the level set function.
- *Heuristic motivation:* Assume that, for all $t \geq 0$, the solution $\phi(t, \cdot)$ to the equation with a (everywhere defined) normal velocity $v$

$$\frac{\partial \phi}{\partial t}(t, x) + v(t, x)|\nabla \phi(t, x)| = 0$$

is the signed distance function to $\Omega(t)$; a formal calculation yields:

$$0 = \frac{\partial}{\partial t}\left(|\nabla \phi|^2\right) = -2|\nabla \phi| \nabla \phi \cdot \nabla v - 2v \nabla \phi \cdot \nabla \left(|\nabla \phi|\right) = -2 \nabla \phi \cdot \nabla v \quad \text{on } \mathbb{R}^d.$$

- Hence, a necessary condition for $\phi(t, \cdot)$ to stay a signed distance function when the extension $v_{\text{ext}}$ of a field $v$ defined only for $x \in \Gamma(t)$ is used reads:

$$\nabla v_{\text{ext}}(t, x) \cdot \nabla \phi(t, x) = 0,$$

i.e. at any $t \geq 0$, $v_{\text{ext}}(t, \cdot)$ is constant along the (extended) normal $n_t$.

# Normal extension of the velocity field (II)

- *Numerical setting:*
  1. The time interval $(0, T)$ is divided into subintervals $(t^n, t^{n+1})$, where $t^n = n\Delta t$,
  2. The level set equation is solved on each interval $(t^n, t^{n+1})$ with initial data $\phi^n := \phi(t^n, \cdot)$, and velocity field $v^n := v(t^n, \cdot)$.

- One method to stick with the previous observations is the following, at every stage $t^n$ of the level set process:
  1. Calculate the signed distance function $d_{\Omega^n}$ to $\Omega^n$.
  2. Calculate $v^n_{\text{ext}}$ as the solution to:

$$\begin{cases} \nabla v^n_{\text{ext}} \cdot \nabla d^n_\Omega = 0 & \text{in } \mathbb{R}^d \setminus \Gamma(t^n) \\ v^n_{\text{ext}} = v^n & \text{on } \Gamma(t^n). \end{cases}$$

# PDE-based extension of the velocity field

- Let $\Omega$ be a domain, $v : \Gamma \to \mathbb{R}$ be a (scalar) velocity field, to be extended into $v_{\text{ext}}$, defined on a larger computational domain $D$.

- *One possibility:* Search for the solution $v_{\text{ext}} \in H^1(D)$ to the equation:

$$\begin{cases} -\alpha \Delta v_{\text{ext}} + v_{\text{ext}} = 0 & \text{in } D, \\ v_{\text{ext}} = v & \text{on } \partial\Omega \end{cases} .$$

- $\alpha$ is a "small" diffusion parameter, controlling the degree of smoothing in $v_{\text{ext}}$ which is intrinsically 'regular' ($v_{\text{ext}} \in H^1(D)$).

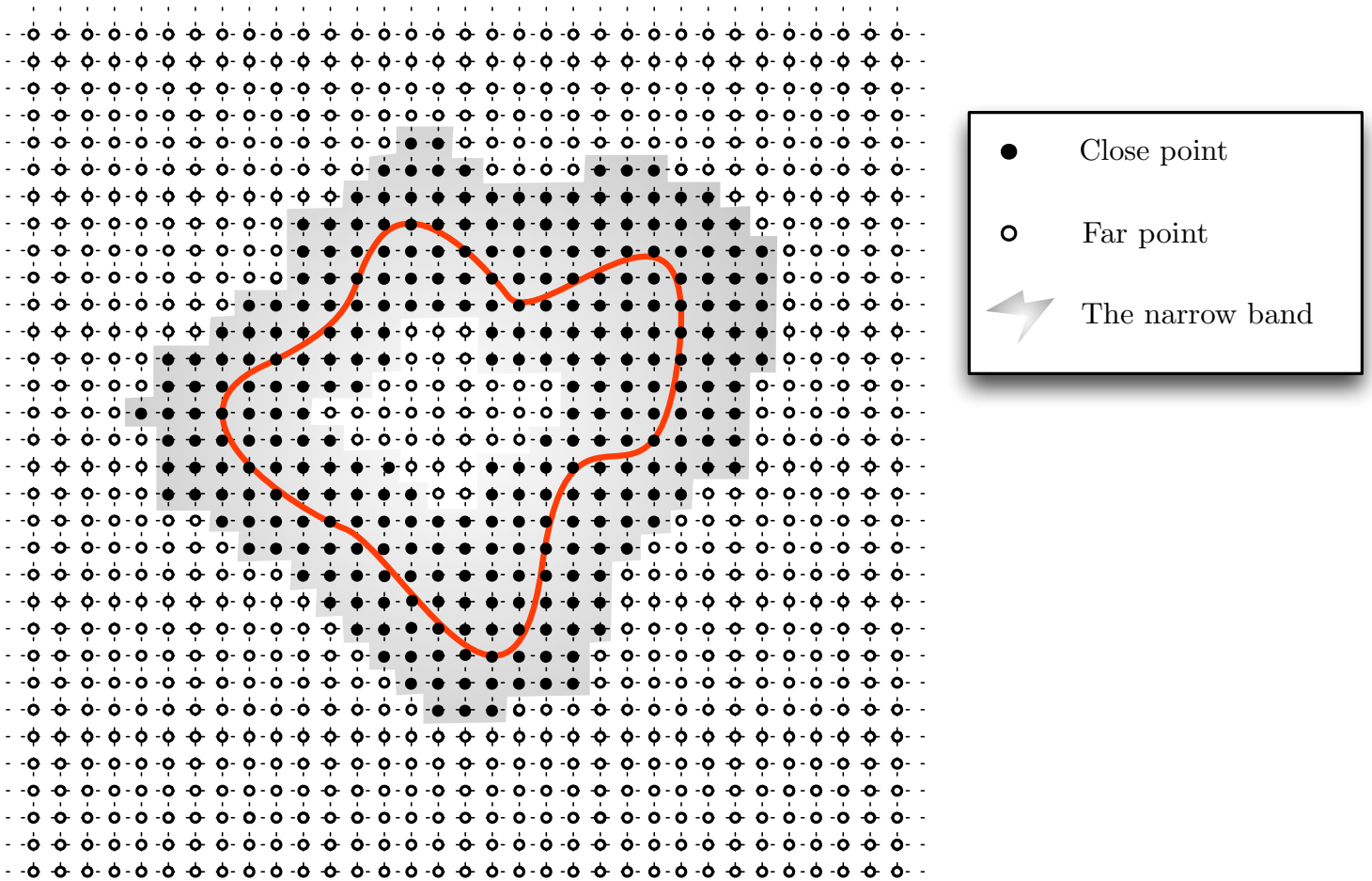
- *Other possibilities:* the constraint $v_{\text{ext}} = v$ on $\Gamma$ may be dropped. This could be unacceptable in some situations, but prove very useful in others.

# The narrow band paradigm (I)

- In capturing the evolution of a domain $\Omega(t)$ by that of an associated level set function $\phi(t, x)$, only the region of space which is near the 0 level set $\Gamma(t)$ is of interest.
- Hence, all the operations associated to the practice of the level set method, namely:
  - The initialization of a level set function for the domain $\Omega(0)$,
  - The occasional redistancing of the Level Set function $\phi(t, \cdot)$ when it has become too "steep",
  - The resolution of the Level Set evolution equation,
  - The velocity extension procedure,

  can be restricted to a narrow band around $\Gamma(t)$.
- This allows to substentially decrease the CPU time of the process.

# The narrow band paradigm (II)




*The narrow band setting: a tube of "close" points is maintained around* $\Gamma(t)$.

# The narrow band paradigm (III)

In practice,

- A narrow band $\mathcal{B}$ of close points is initialized around the boundary $\Gamma(0)$, e.g. as a tube of $k$ elements around $\Gamma(0)$.

- At every iteration $n \to n+1$, an attempt step is carried out:

$$\forall i,j \in \mathbb{Z} \text{ s.t. } x_{ij} \in \mathcal{B}, \quad \left\{\phi_{ij}^{n}\right\} \mapsto \left\{\phi_{ij,\text{temp}}^{n+1}\right\},$$

  i.e. the level set evolution equation is solved only at the close points, (special attention must be paid to the calculation of derivatives at the points near the border of $\mathcal{B}$).

- If the new front $\Gamma(t^{n+1})$ is still inside $\mathcal{B}$, accept the iteration; this can be checked from the signs of the $\phi_{ij}$, for nodes $x_{ij}$ near the border of $\mathcal{B}$.

- Else, return to step $n$, reinitialize a narrow band $\mathcal{B}$ around $\Gamma(t^n)$, and retry the iteration $n \to n+1$.

# Part VI

# Applications of the level set method

# Example: image segmentation (I)

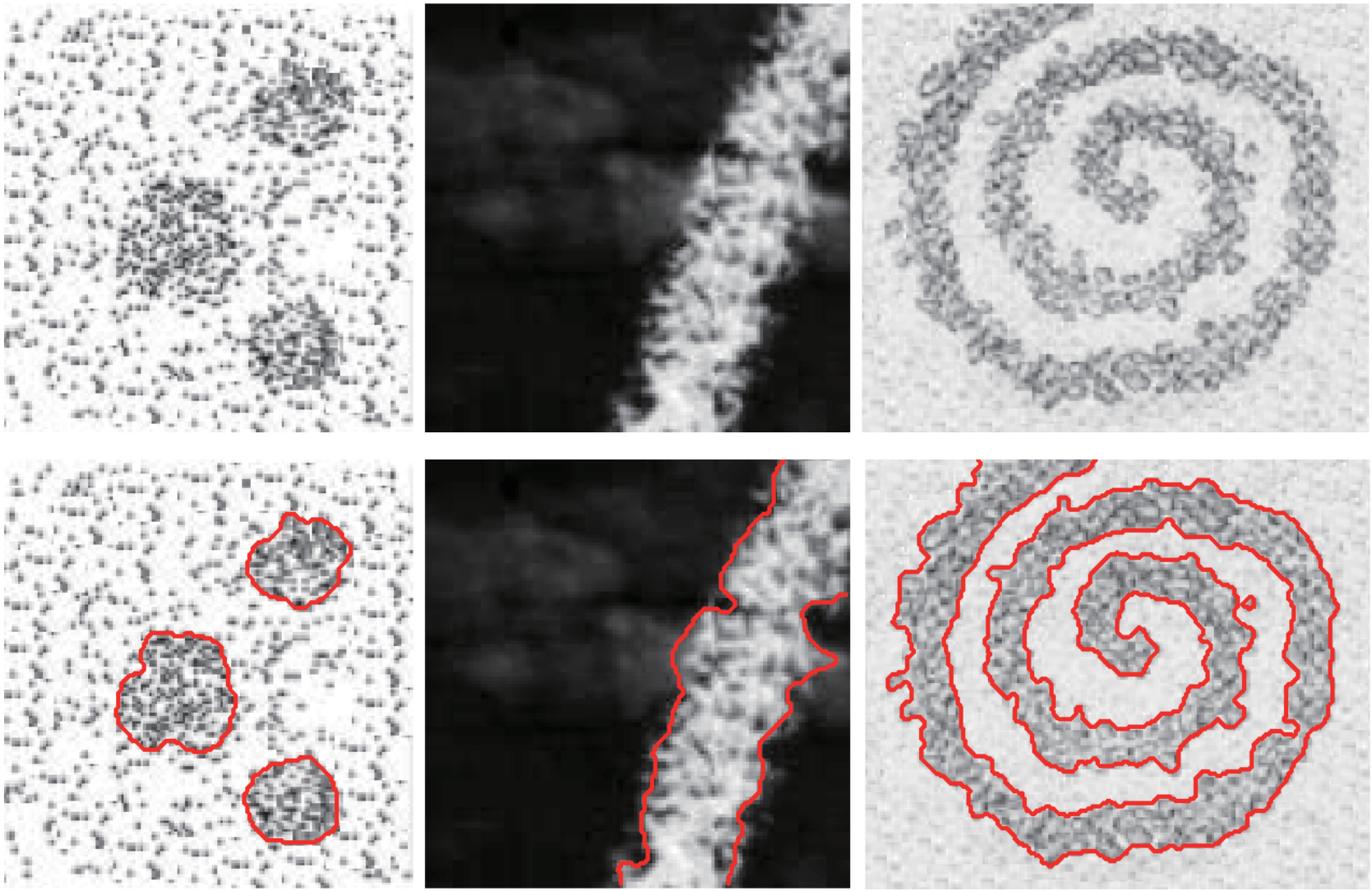
- A greyscale image is described by an intensity function $I : [0,1]^2 \to \mathbb{R}$.
- The image is composed of several objects, i.e. regions with different values of $I$. One of them, $\Omega_T$, identified by the intensity $I_T$ is to be accurately separated from the others.
- The idea of active contour methods is to track the evolution of a domain $\Omega(t)$, starting from an arbitrary "initial guess" $\Omega^0$, according to a velocity field of the form:

$$V(t,x) = (f_0(x) + f_1(\kappa_t(x)))n_t(x),$$

where $f_0$ and $f_1$ are two scalar functions:

  - $f_0$ "attracts" the domain $\Omega(t)$ towards $\Omega_T$,
  - $f_1(\kappa_t(x))$ compels $\Omega(t)$ to stay "smooth enough".

# Example: image segmentation (II)



*Three examples of segmentation in biomedial imaging [Credits: [WHL]].*

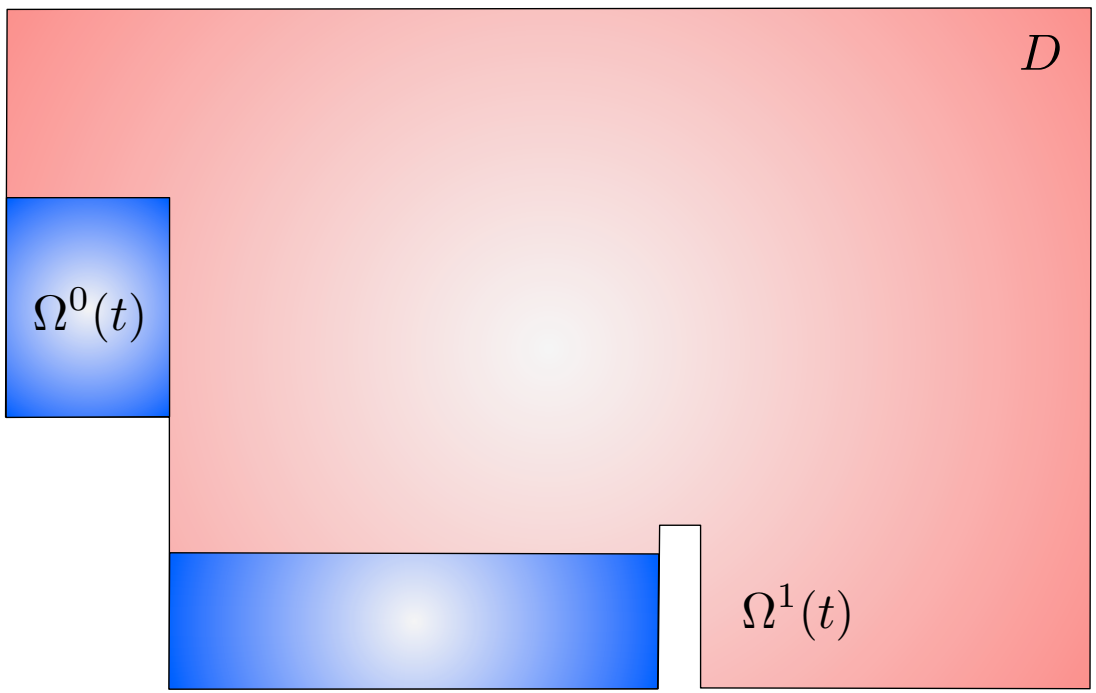
# Part VI

# Applications of the level set method

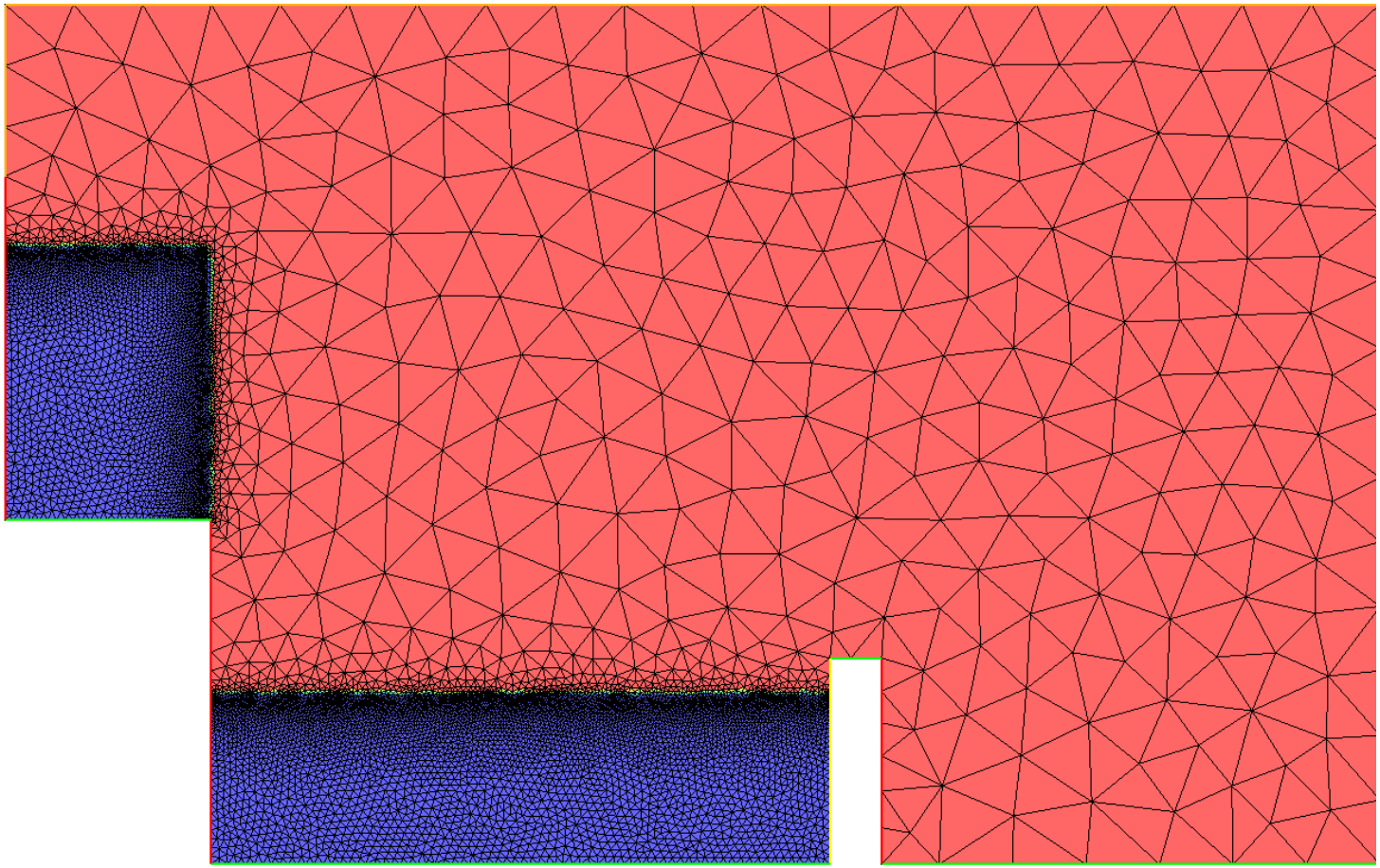
# Example: bifluid flows (I)

As a result of the rupture of a dam, a water column discharges into a lower basin.

- The problem involves two complementary fluid phases $\Omega^0(t)$, $\Omega^1(t) \subset D$.
- $\Omega^0(t)$ is filled with water, $\Omega^1(t)$ is made of air.
- The velocity $V(t,x)$ of the motion is the solution to the two-phase Navier-Stokes equations.



# Example: bifluid flows (II)



*Evolution of a collapsing water column*

# Part VI

# Applications of the level set method

# Example: shape optimization (I)

- Shape optimization aims at improving the performance of the initial design $\Omega^0$ of a mechanical structure (e.g. a beam, a mechanical actuator,...) or a fluid duct, with respect to a physical criterion.
- The problem arises under the form:

$$\min_{\Omega \in \mathcal{U}_{\mathrm{ad}}} J(\Omega),$$

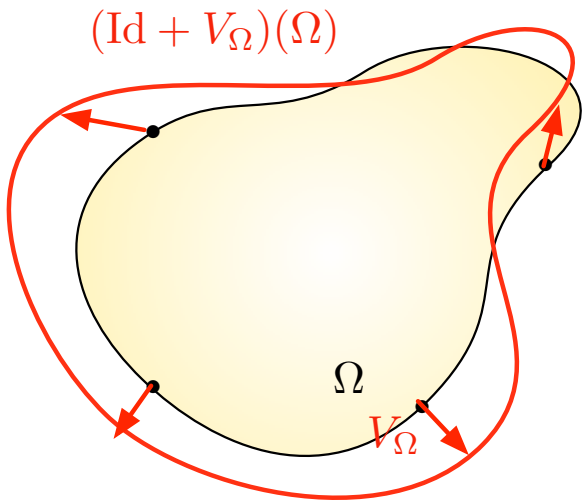
where

  - $J(\Omega)$ is a cost functional, depending on $\Omega$ in a possibly very complicated way (via the solution to a PDE posed on $\Omega$). For instance,
    - When $\Omega$ is a structure, $J(\Omega)$ may be the work of external forces on $\Omega$, a vibration frequency, etc.
    - When $\Omega$ is a fluid duct, $J(\Omega)$ may account for the work of viscous forces inside $\Omega$.
  - $\mathcal{U}_{\mathrm{ad}}$ is a set of admissible designs, which encompasses, e.g. volume, or manufacturability constraints.

# Example: structural optimization (II)

- Techniques from shape optimization make it is possible to calculate a shape gradient at a shape $\Omega$, i.e. a vector field $V_\Omega : \mathbb{R}^d \to \mathbb{R}^d$ such that:

$$J((\mathrm{Id} + \tau V_\Omega)(\Omega)) < J(\Omega), \quad \text{for } \tau > 0 \text{ small enough.}$$




- Starting from an initial design $\Omega^0$, the sequence of shapes

$$\Omega^{n+1} := (\mathrm{Id} + \tau^n V_{\Omega^n})(\Omega^n), \quad \text{where } \tau^n \text{ is a pseudo-time step,}$$

evolves by decreasing the criterion $J(\Omega)$.

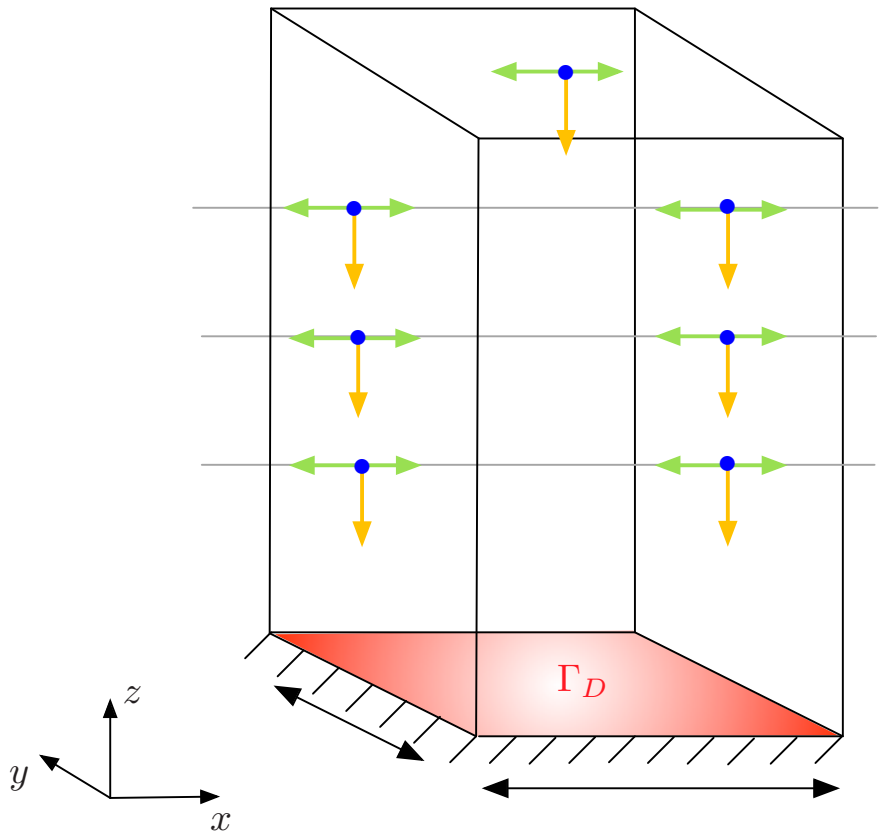
# Example: structural optimization (II)

We consider the optimization of an electric pylon $\Omega \subset \mathbb{R}^3$.

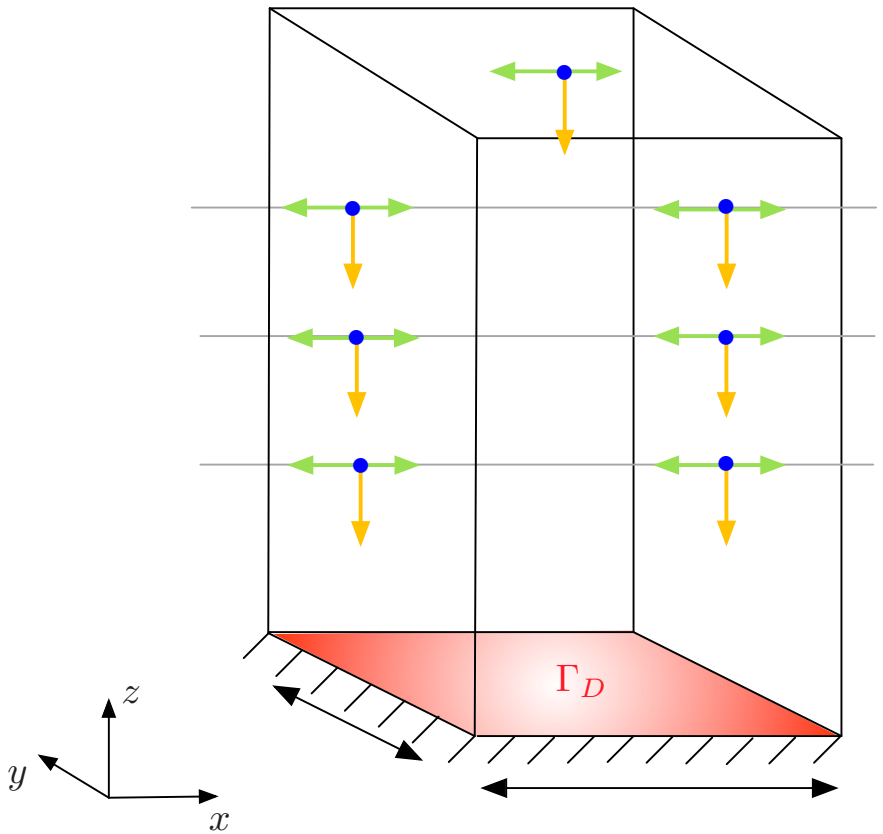
- At its basis $\Gamma_D$, the pylon is fixed to the ground.
- It is submitted to the weight of the cables attached to its arms, and to wind loads.
- The displacement $u_\Omega : \Omega \to \mathbb{R}^3$ is the solution to the linear elasticity system.
- The compliance of the structure,

$$J(\Omega) = \int_\Omega Ae(u_\Omega) : e(u_\Omega)\, dx$$

is minimized under a volume constraint.



# Example: structural optimization (III)

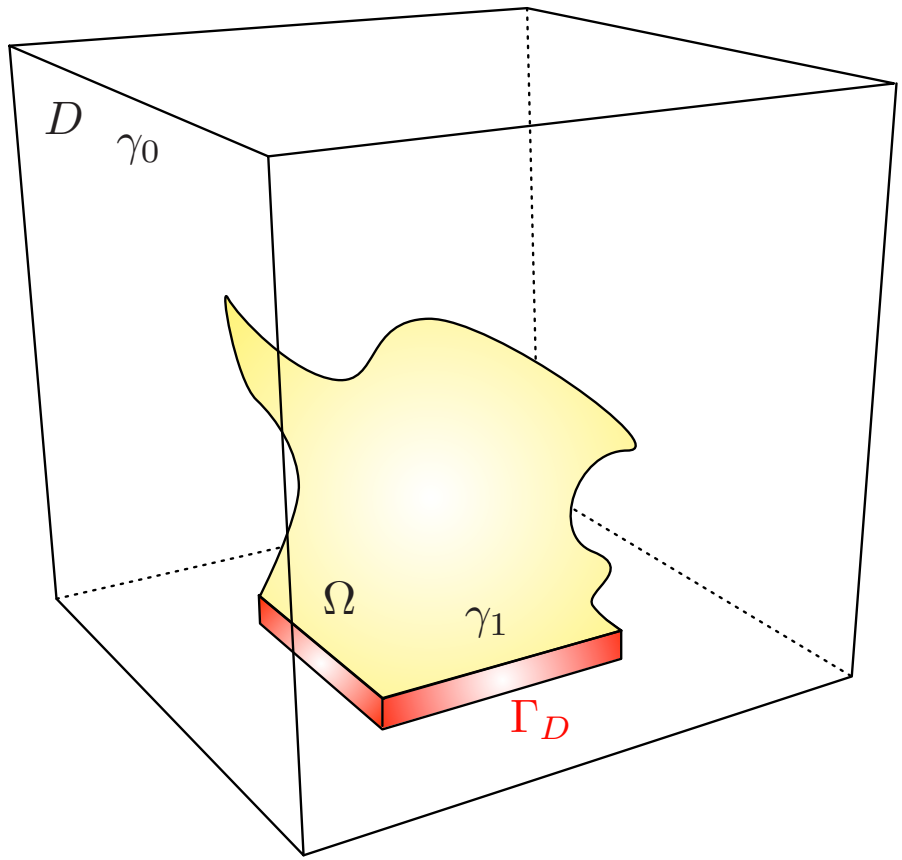


# Optimization of the shape of a heat diffuser (I)

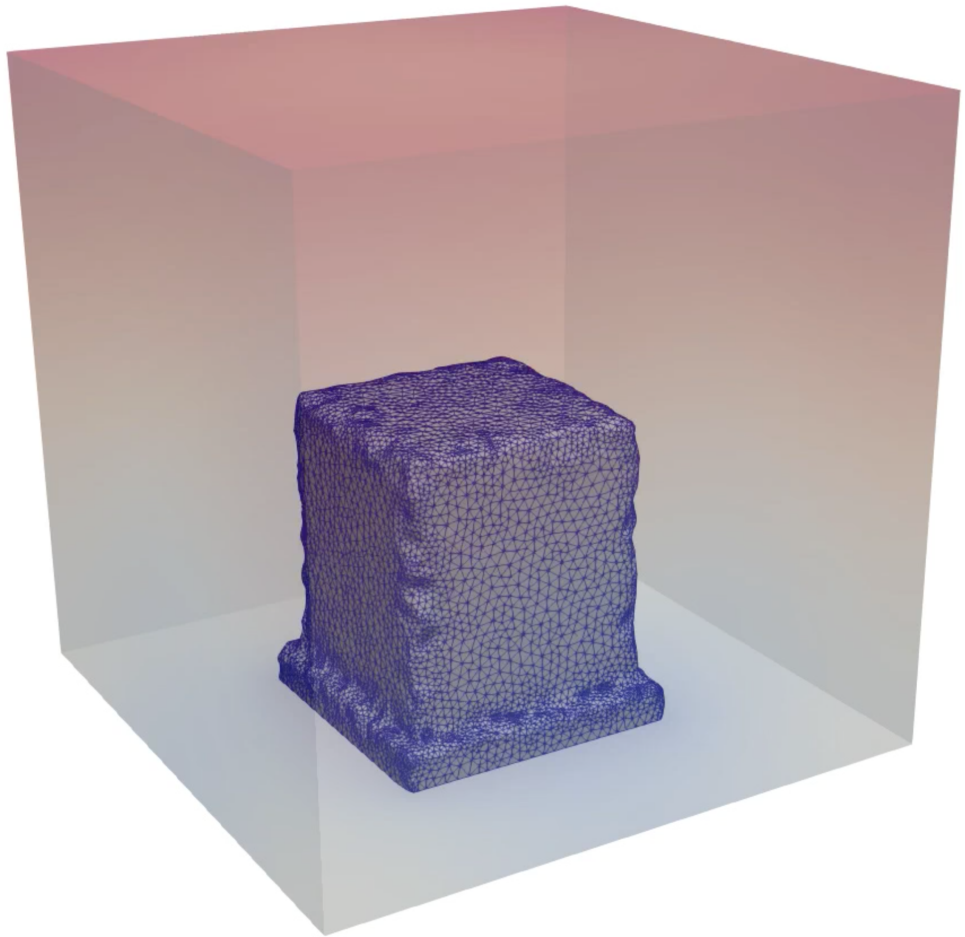
- A thermal chamber $D$ is divided into
  - A phase $\Omega$ with high conductivity $\gamma_1$
  - A phase $D \setminus \overline{\Omega}$ with low conductivity $\gamma_0$.
- A temperature $T_0 = 0$ is imposed on $\Gamma_D$ and the remaining boundary $\partial D \setminus \overline{\Gamma_D}$ is insulated from the outside.
- A heat source is acting inside $D$.
- The temperature $u_\Omega$ inside $D$ is solution to the two-phase Laplace equation.
- The average temperature inside $D$,

$$J(\Omega) = \frac{1}{|D|} \int_D u_\Omega \, \mathrm{d}x$$

is minimized under a volume constraint.



# Optimization of the shape of a heat diffuser (II)



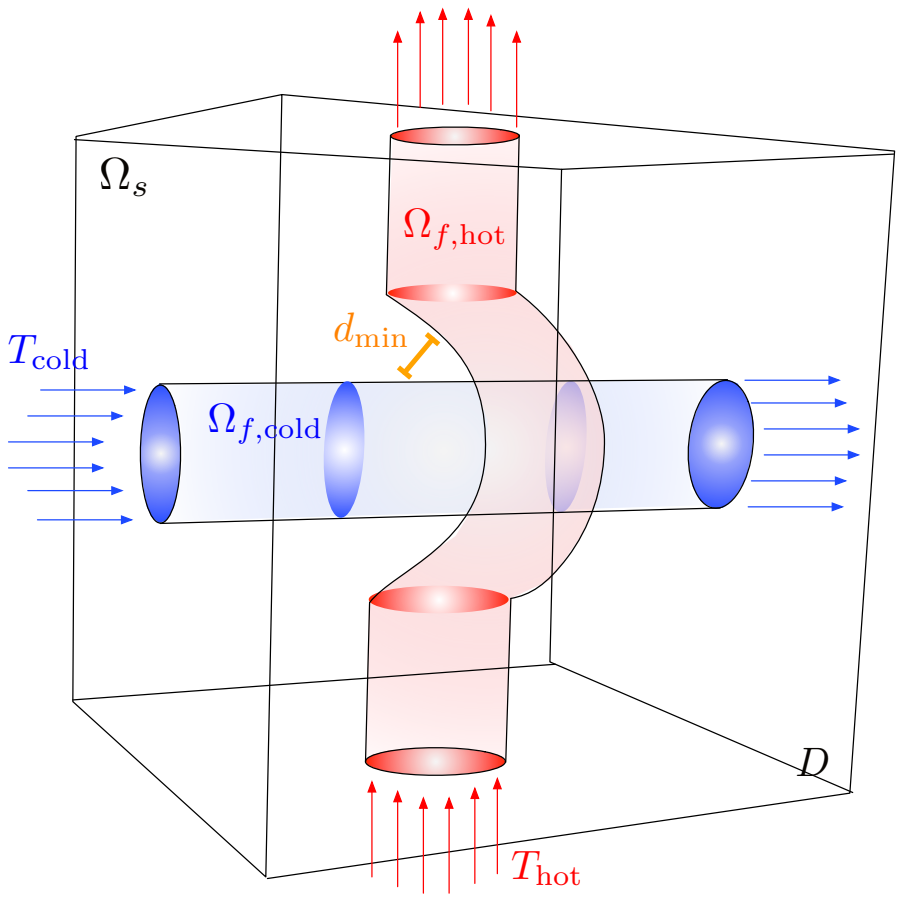
*Optimization of the shape of a heat diffuser.*

# Optimization of the shape of a heat exchanger (I)

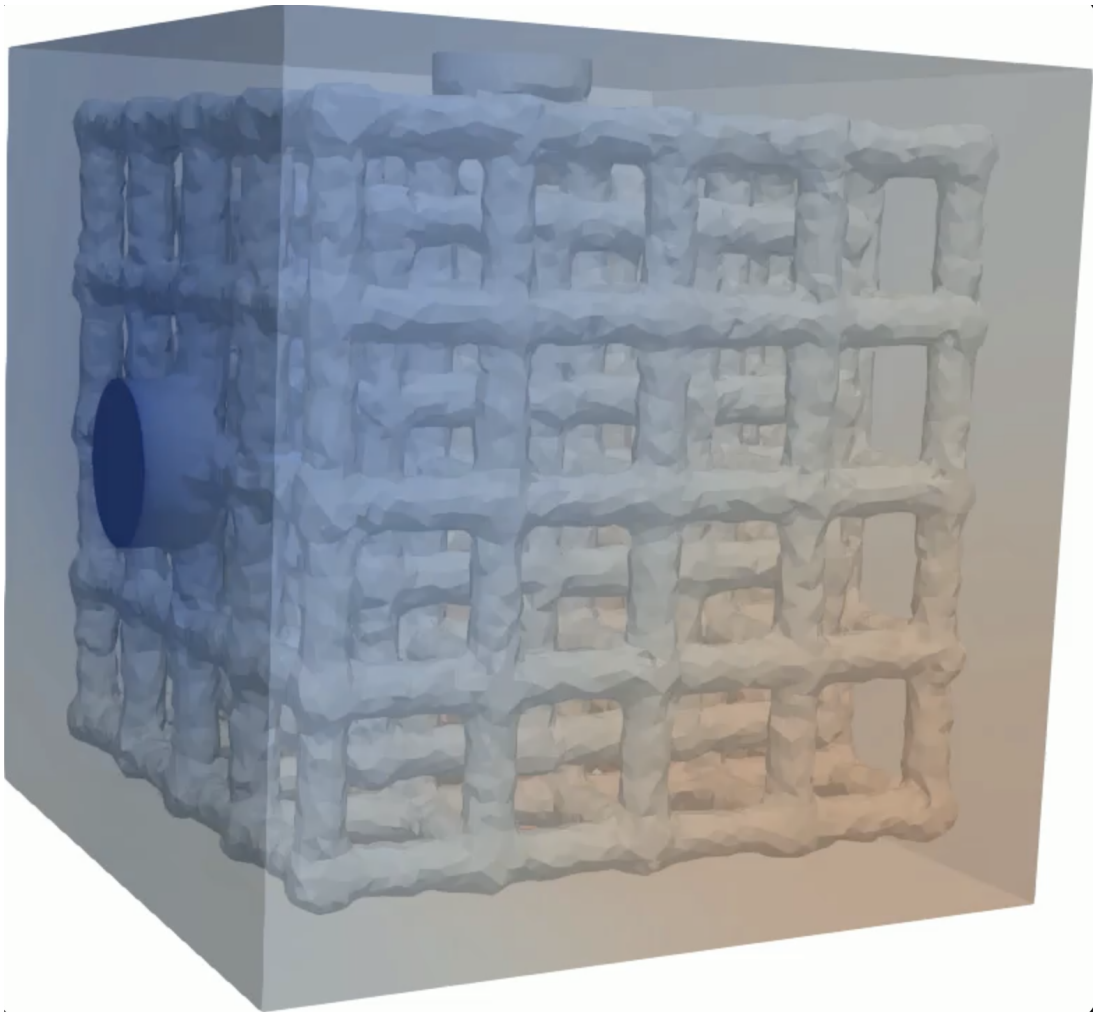
- A thermal chamber $D$ is divided into
  - A phase $\Omega_{f,\text{hot}}$ conveying a hot fluid;
  - A phase $\Omega_{f,\text{cold}}$ conveying a cold fluid;
  - A solid phase $\Omega_s$.
- The Navier-Stokes equations are satisfied in $\Omega_{f,\text{hot}}$, $\Omega_{f,\text{cold}}$.
- The stationary heat equation accounts for the temperature diffusion within $D$.
- The heat transferred from $\Omega_{f,\text{hot}}$ to $\Omega_{f,\text{cold}}$ is maximized.
- A constraint is imposed on the minimal distance between $\Omega_{f,\text{hot}}$ and $\Omega_{f,\text{cold}}$:

$$d(\Omega_{f,\text{hot}}, \Omega_{f,\text{cold}}) \geq d_{\text{min}}.$$

- Volume and pressure drop constraints are added on $\Omega_{f,\text{hot}}$, $\Omega_{f,\text{cold}}$.



# Optimization of the shape of a heat exchanger (II)



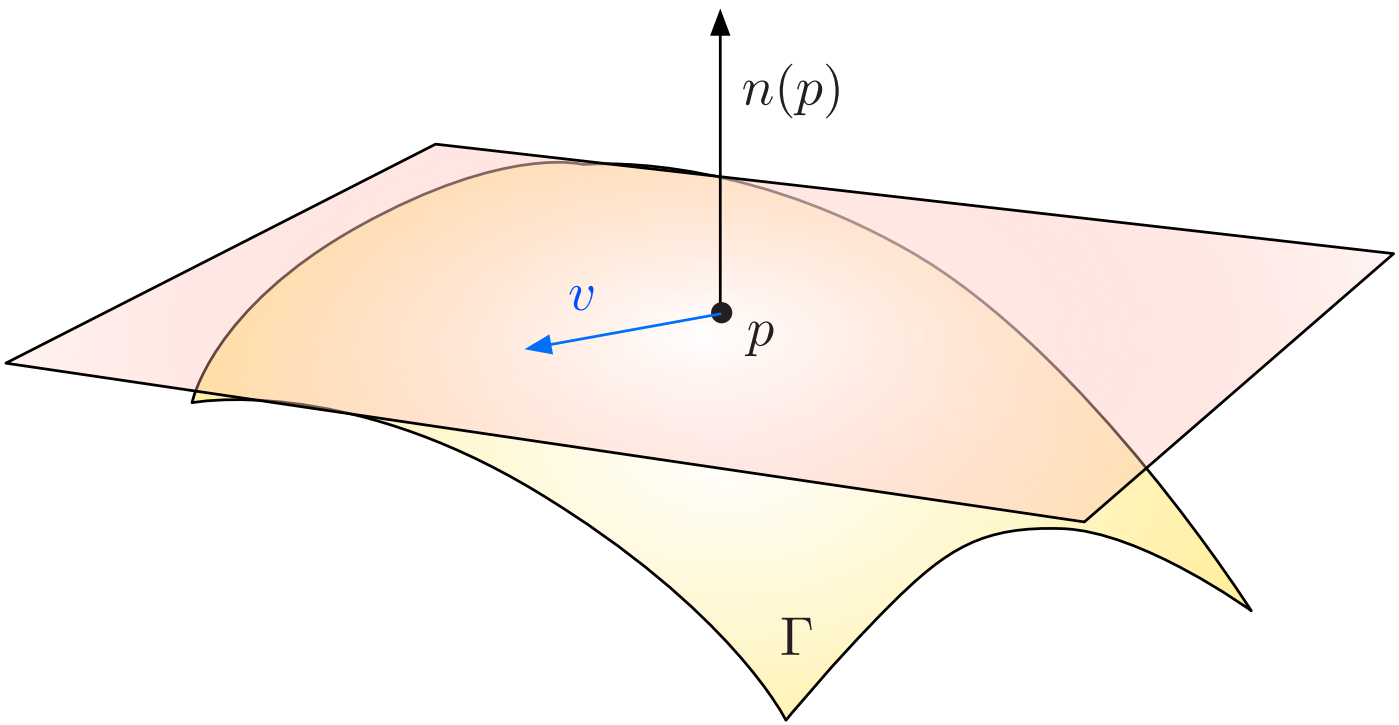
*Optimization of the shape of a heat exchanger.*

# Technical appendix

# Surfaces and curvature (I)

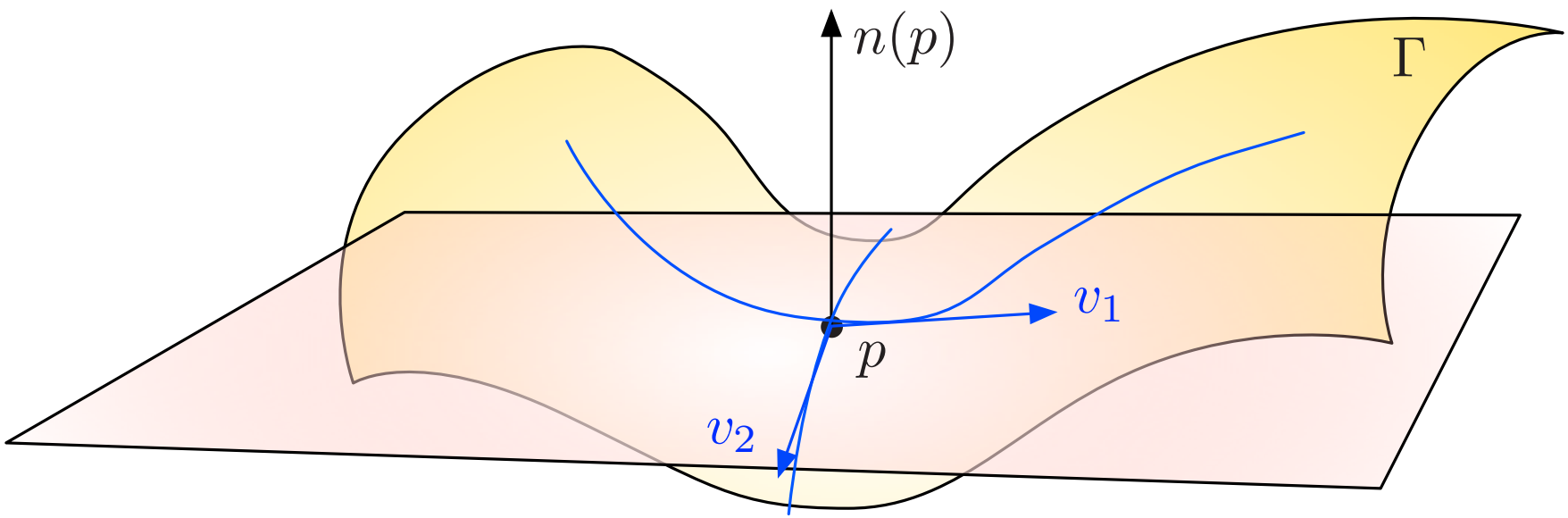
At first order, in the neighborhood of a point $p \in \Gamma$, a surface $\Gamma$ behaves like a plane, the tangent plane,

- With normal vector $n(p)$,
- Which contains the tangential directions to $\Gamma$.



# Surfaces and curvature (II)

- At second order in the neighborhood of $p \in \Gamma$, the surface $\Gamma$ has one curvature in each tangential direction.
- The principal directions at $p$ are those tangential directions $v_1(p)$ et $v_2(p)$ associated to the lower and larger curvatures $\kappa_1(p)$ et $\kappa_2(p)$.
- The mean curvature $\kappa(p)$ is the sum $\kappa(p) = \kappa_1(p) + \kappa_2(p)$.

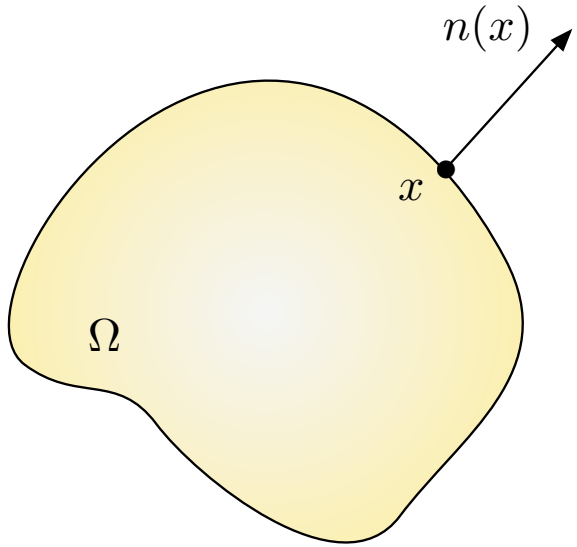


# The Green's formula

Green's formula is a generalization of integration by parts for functions defined on a smooth bounded domain $\Omega \subset \mathbb{R}^d$.

In such a context,

- $n = (n_1, \ldots, n_d)$ is the unit normal vector to $\partial\Omega$, pointing outward $\Omega$;
- $\mathrm{d}s$ is the integration measure on the oriented hypersurface $\partial\Omega$.




## Proposition 3.

*In the above setting, let $u : \mathbb{R}^d \to \mathbb{R}$ be a function of class $\mathcal{C}^1$; then*

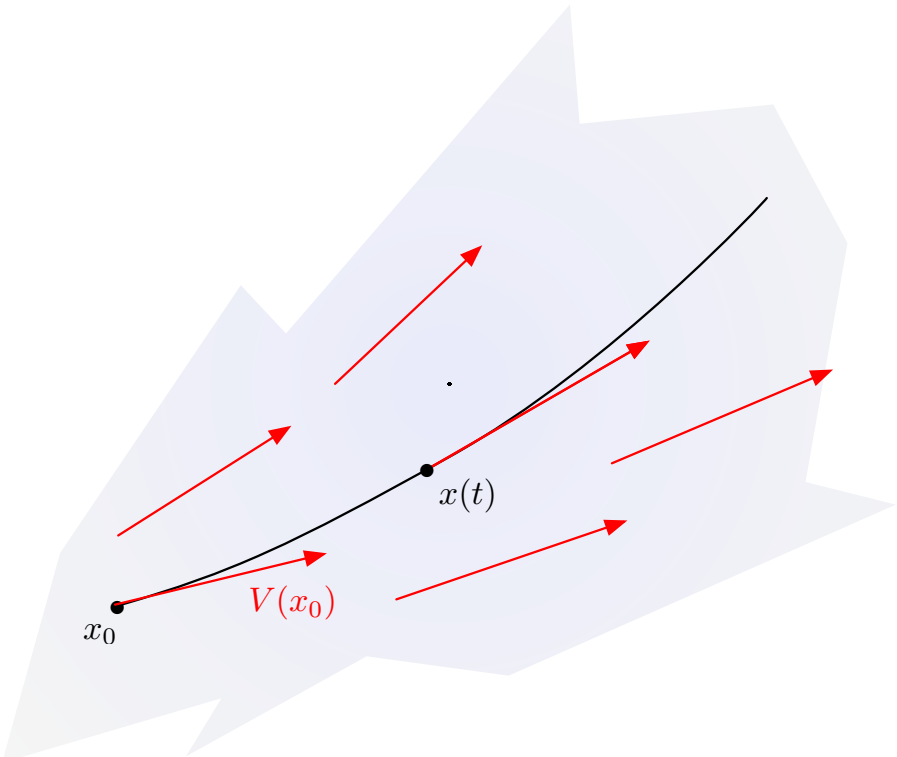
$$\int_\Omega \frac{\partial u}{\partial x_i}(x)\,\mathrm{d}x = \int_{\partial\Omega} u n_i(x)\,\mathrm{d}s(x).$$

# Runge-Kutta integration of dynamical systems (I)

Let $V : \mathbb{R}^d \to \mathbb{R}^d$ be a (smooth) vector field; we consider the dynamical system

$$\begin{cases} x'(t) = V(x(t)) & \text{for } t \in (0, T), \\ x(0) = x_0, \end{cases}$$

for the trajectory $t \mapsto x(t)$ of a particle with velocity $V$.

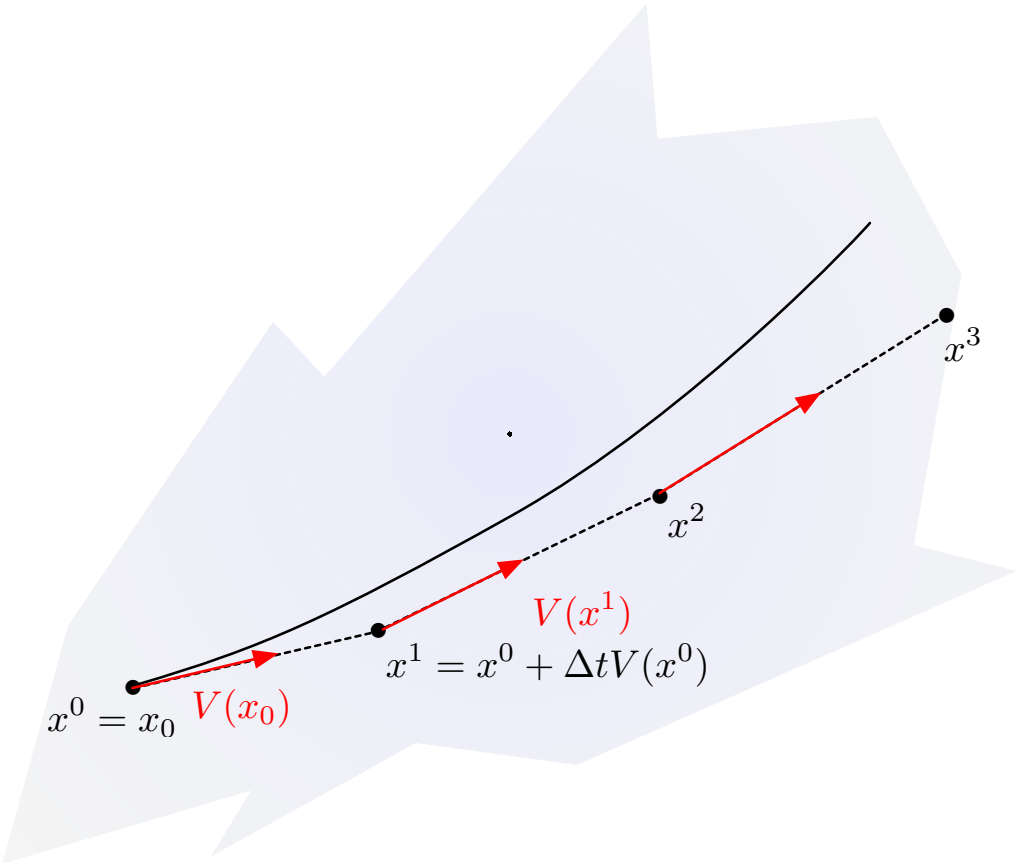



Introducing a subdivision $t^n = n\Delta t$ of $(0, T)$, $n = 0, \ldots, N := T/\Delta t$, we aim to calculate an approximation $x^n$ of $x(t^n)$.

# Runge-Kutta integration of dynamical systems (II)

The first-order, explicit Euler approximation of this dynamical system reads:

$$\begin{cases} x^{n+1} = x^n + \Delta t V(x^n) & \text{for } n = 0, \ldots, N-1, \\ x^0 = x_0. \end{cases}$$



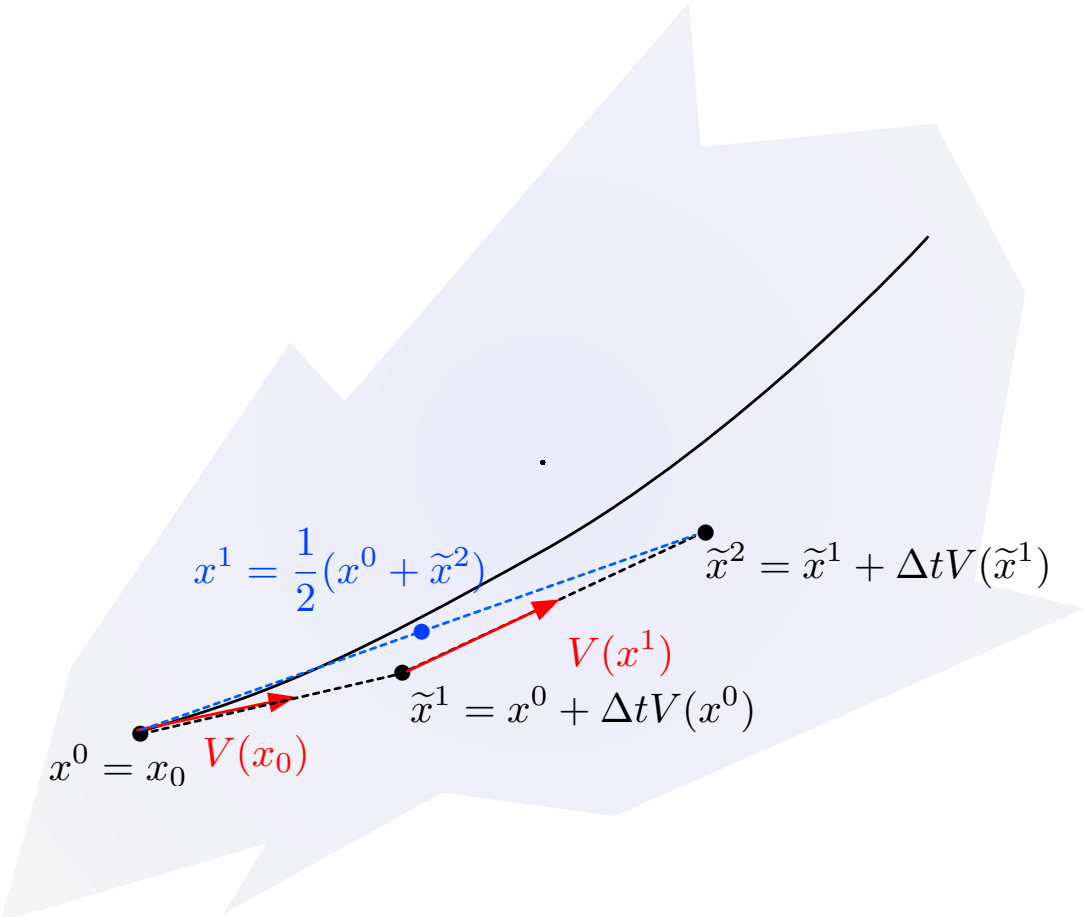

This method is only first-order accurate as $\Delta t \to 0$:

$$\forall n \in 0, \ldots, N, \quad |x(t^n) - x^n| \leq C \Delta t \text{ for some constant } C > 0.$$

# Runge-Kutta integration of dynamical systems (III)

According to the Runge-Kutta 2 method, the iterate $x^{n+1}$ is obtained from $x^n$ by:

1. An attempt step is performed with the 1st-order Euler method:
$$\widetilde{x}^{n+1} := x^n + \Delta t V(x^n).$$
2. Another attempt step is performed from $\widetilde{x}^{n+1}$:
$$\widetilde{x}^{n+2} := \widetilde{x}^{n+1} + \Delta t V(\widetilde{x}^{n+1}).$$
3. The point $x^{n+1}$ is obtained by averaging:
$$x^{n+1} = \frac{1}{2}(x^n + \widetilde{x}^{n+2}).$$




This method is second-order accurate:

$$\forall n = 0, \ldots, N, \quad |x(t^n) - x^n| \leq C\Delta t^2 \text{ for some constant } C > 0.$$

# References

# Mathematical references I



[Barles] G. Barles, *Remarks on a flame propagation model*, INRIA: Technical Report, 451 (1985).


[CIL] M.G. Crandall, H.Ishii and P.L. Lions, *User's guide to viscosity solutions of second order partial differential equations*, Bulletin of the American Mathematical Society, 27 (1992), pp. 1–67.


[Giga] Y. Giga, *Surface Evolution Equations, a Level Set Approach*, Monographs in Mathematics, 99. Birkhäuser, Basel-Boston-Berlin, (2006).


[Grayson] M.A. Grayson, *The heat equation shrinks embedded plane curves to round points*, J. Differential Geometry, 26, (1987), pp. 285–314.

# Numerical references I


[Abgrall] R. Abgrall, *Numerical Discretization of the First-Order Hamilton-Jacobi Equation on Triangular Meshes*, Com. Pure Applied Math. XLIX, (1996) pp. 1339–1373.

[Barth] T.J. Barth and J.A. Sethian, *Numerical Schemes for the Hamilton–Jacobi and Level Set Equations on Triangulated Domains*, J. Comput. Phys., 145 (1998) pp. 1–40.

[OFed] S.J. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer Verlag, (2003).

[OSeth] S.J. Osher and J.A. Sethian, *Fronts propagating with curvature-dependent speed : Algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys., 79 (1988), pp. 12–49.

[SethianFMM] J.A. Sethian, *A fast marching level set method for monotonically advancing fronts*, Proc. Natl. Acad. Sci. USA Vol. 93, (1996), pp. 1591–1595.

# Numerical references II


[Sethian] J.A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, (1999).

[Shu] C.-W. Shu, *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*, in Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, A. Quarteroni, Editor, Lecture Notes in Mathematics, CIME subseries, Springer-Verlag, ICASE Report 97–65, (1997).

[Strain] J. Strain, *Semi-Lagrangian Methods for Level Set Equations*, J. Comput. Phys., 151, (1999), pp. 498–533.

[WHL] W. Wen, C. He and M. Li, *Transition region-based active contour model for image segmentation*, J. Electron. Imaging. 22(1), (2013).

[Zhao] H.-K. Zhao, *A Fast Sweeping Method for Eikonal Equations*, Math. Comp., 74 (2005), pp. 603–627.