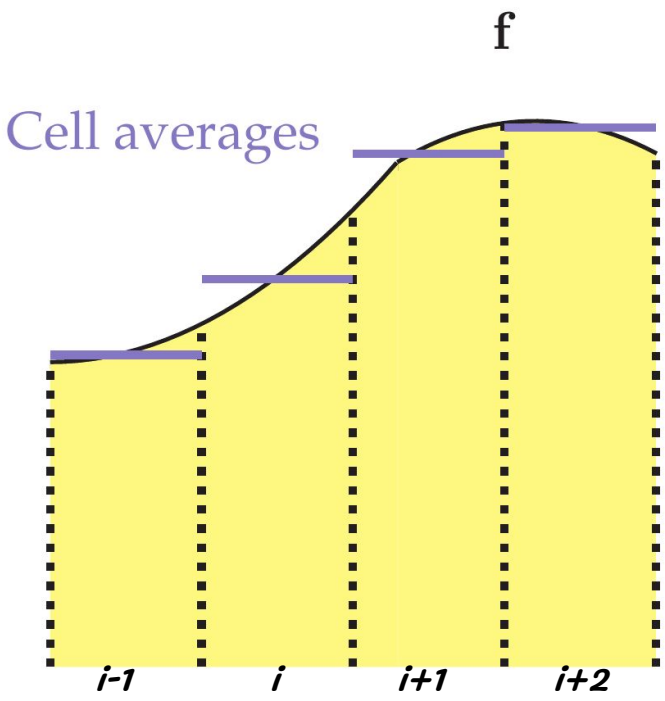
# Edge adaptive schemes and machine learning for image super-resolution

**ICIAM**
2023

Somacal Agustin
In collaboration with A. Cohen, F. Arandiga,
D. Matthieu, O. Mula.

# (1) Reconstruction problem

# Function reconstruction 1D



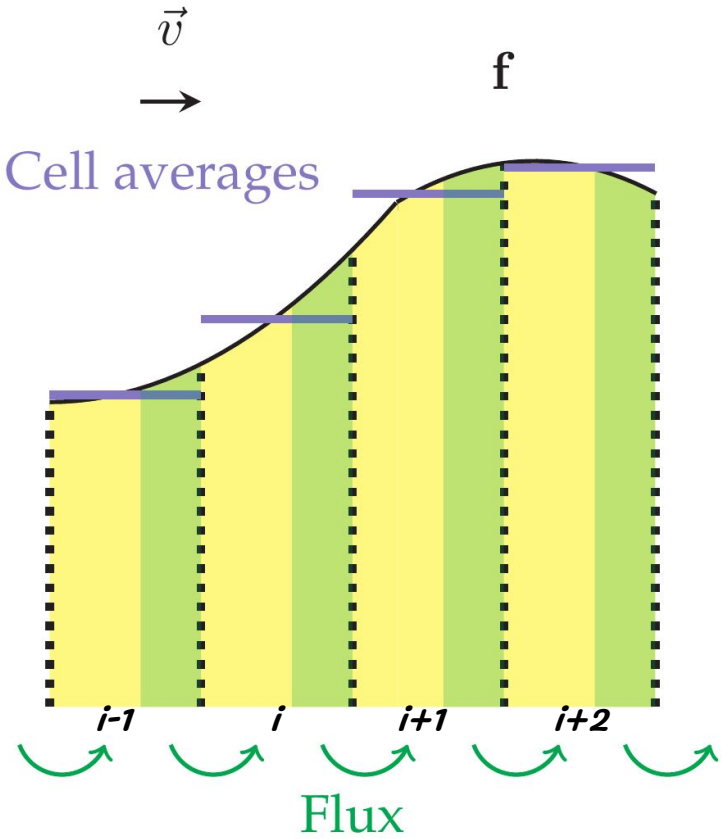

<u>Objective</u>: **approximate** a function ***f*** from cell averages ***a*** of regular grids.

$$a_i := \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} f(x)dx$$

Motivation:

- Image processing: ***a*** being a low resolution image, we would like to output reconstructions of arbitrary resolution.

# Finite volume scheme 1D




<u>Objective</u>: **approximate** a function ***f*** from cell averages ***a*** of regular grids.

$$a_i := \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} f(x)dx$$

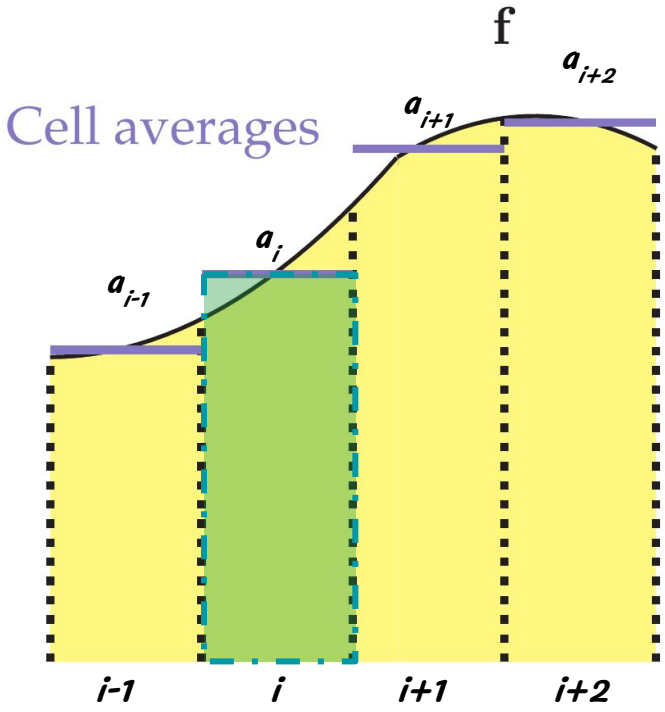
Motivation:

- Image processing: ***a*** being a low resolution image, we would like to output reconstructions of arbitrary resolution.
- Finite volume schemes: ***a*** being the initial condition of a PDE,

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0$$

  we would iteratively calculate the numerical fluxes ***F*** to solve it.

# Polynomial reconstruction

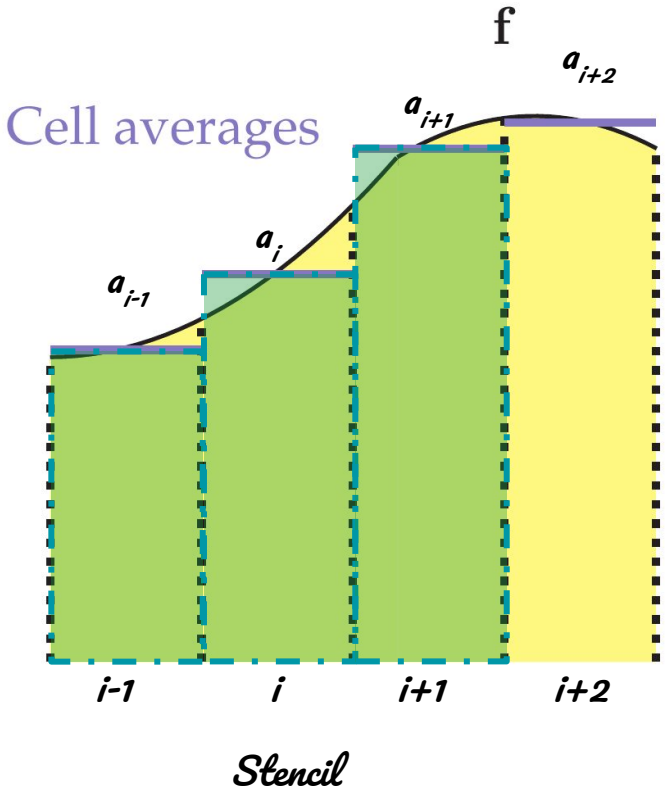



If ***f*** is regular enough (Continuous) reconstruct by fitting average values with polynomials ***p***.

$$\int_{i-\frac{1}{2}}^{i+\frac{1}{2}} f(x)dx = a_i = \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} p(x)dx,$$

$$= \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} c_n x^n + \ldots c_0 dx$$

# Polynomial reconstruction




If ***f*** is regular enough (Continuous) reconstruct by fitting average values with polynomials ***p***.

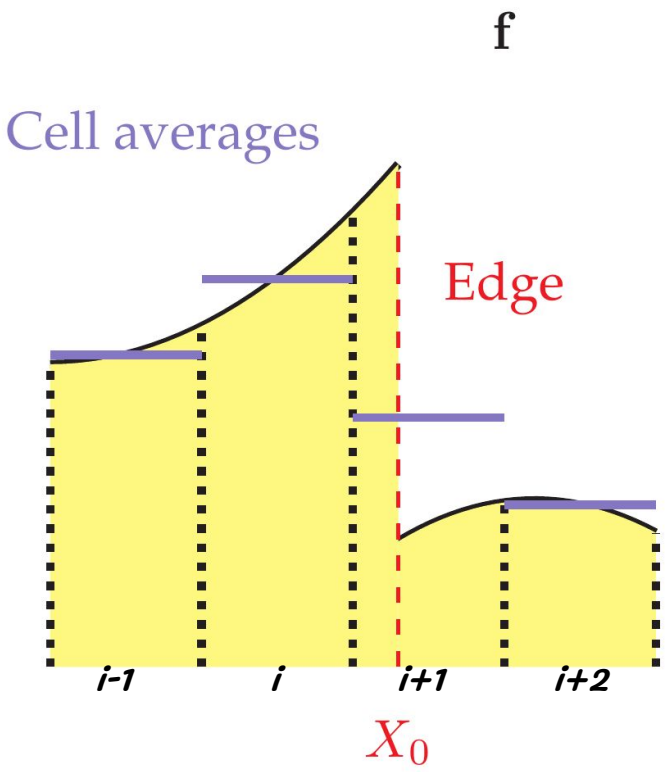
$$\int_{i-\frac{1}{2}}^{i+\frac{1}{2}} f(x)dx = a_i = \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} p(x)dx,$$

$$= \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} c_n x^n + \dots c_0 dx$$

$$\begin{bmatrix} \dots \\ a_{i+1} \\ a_i \\ a_{i-1} \\ \dots \end{bmatrix} = \begin{bmatrix} \text{Monomials integrals matrix} \end{bmatrix} \begin{bmatrix} c_n \\ c_{n-1} \\ \dots \\ c_0 \end{bmatrix}$$

- This can be done for any dimension

# Reconstruction with discontinuities 1D




If ***f*** has sharp discontinuities:

- Find the singular cells:
  - finite differences threshold
  - Avoided cells **[A. Harten, 1987: ENO Schemes with subcell resolution]**
- Identify the position of the discontinuity:
  - If ***f*** is piecewise constant the average value gives the proportion of each side.
  - If ***f*** is a piecewise polynomial one can extend the fitted polynomials by the sides and solve the resulting polynomial equation.

# Reconstruction 2D (or more)

**<u>Difficulty</u>**: find a parametrized curve $\boldsymbol{C}(\boldsymbol{\theta})$ instead of a point.

- Parametrized curves
  - Lines ($\#\boldsymbol{\theta}$=2)
  - Quadratics ($\#\boldsymbol{\theta}$=3)
  - Polynomials
  - Circles ($\#\boldsymbol{\theta}$=3)

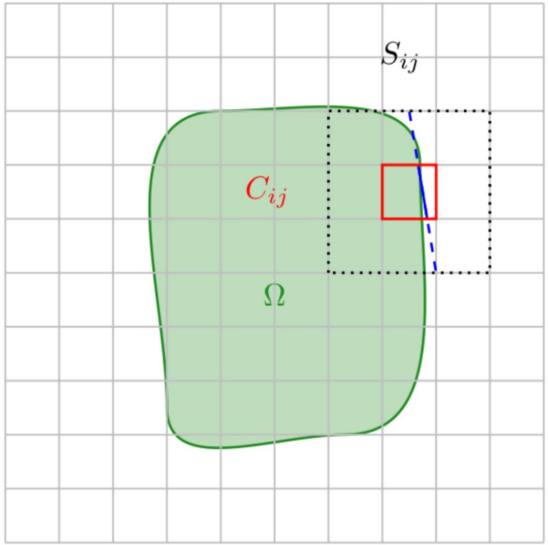
As before, use the average values of the surrounding cells to find the parameters $\boldsymbol{\theta}$ that best approximate the averages.

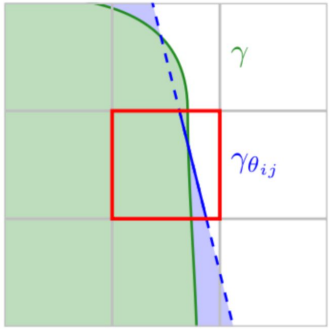
- $\#\boldsymbol{\theta}$ must be equal or inferior to the stencil width (ex, for a 3x3 stencil $\#\boldsymbol{\theta}$ can be at maximum 3)

We will work with $\boldsymbol{f}$ <u>piecewise constant</u>.

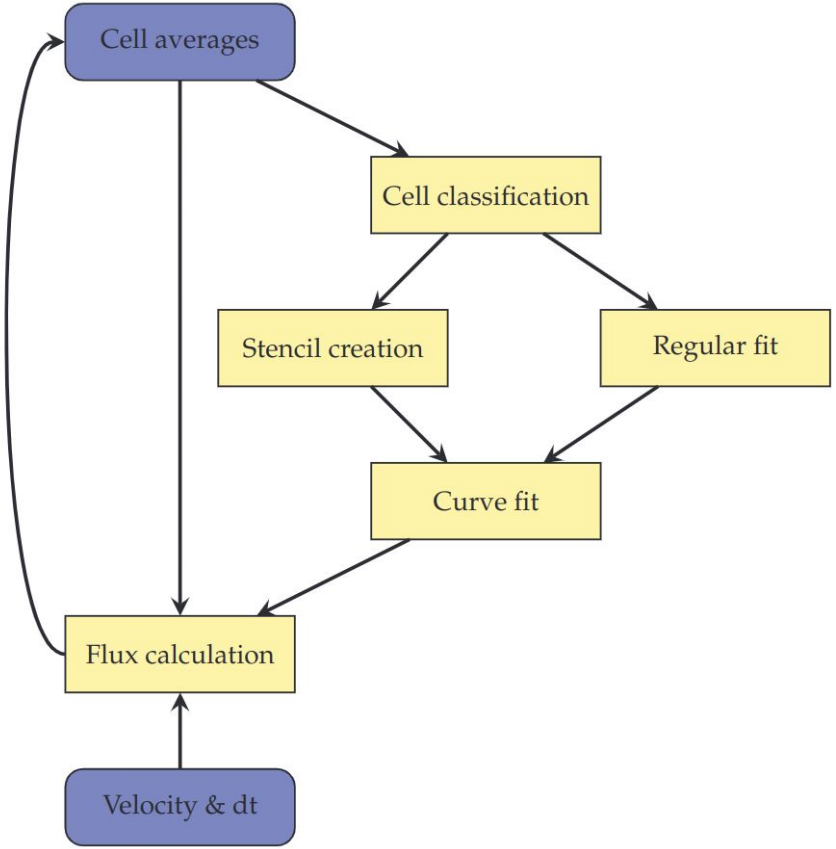
$$a_{ij} := \int\int_{i-\frac{1}{2},j-\frac{1}{2}}^{i+\frac{1}{2},j+\frac{1}{2}} f(x,y)dxdy$$

$h$



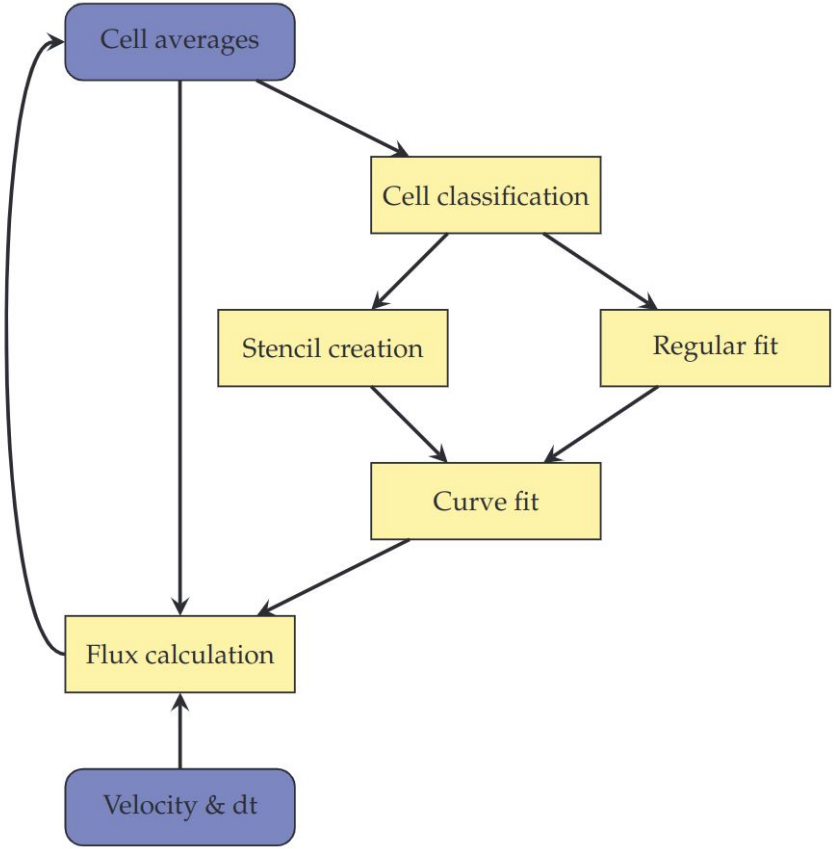
# Image reconstruction pipeline future work




**Curve reconstruction**:

- Fitting lines
  - **ELVIRA** [Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]
  - **ENO-EA** [Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]
- Fitting other curves
- Treating vertices
  - **ML** for 90º vertices [Despres, Jourdren, Machine Learning design of Volume of Fluid schemes for compressible flows, 2020]

# Image reconstruction pipeline future work




**Curve reconstruction**:

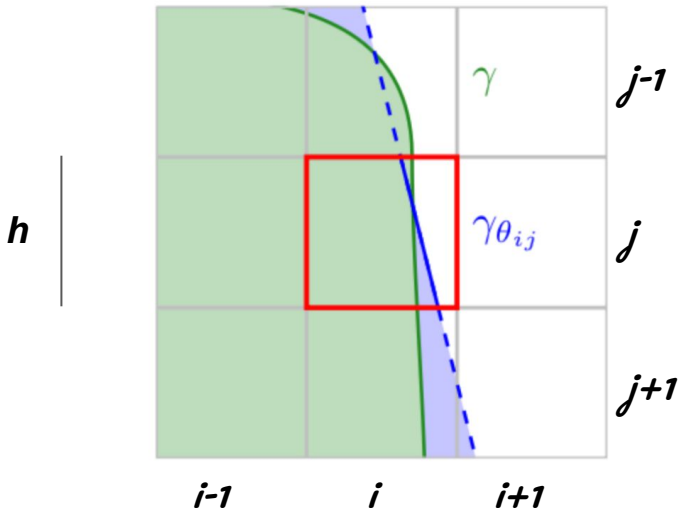
- Fitting lines
  - **ELVIRA** [Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]
  - **ENO-EA** [Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]
  - **ML**
- Fitting polynomials
  - **AERO**
- Fitting other curves
  - **Algebraic closed form**? **Generally not feasible**
- Treating vertices
  - **ML** for 90º vertices [Despres, Jourdren, Machine Learning design of Volume of Fluid schemes for compressible flows, 2020]
  - **AERO**
  - **TEM**

# (2) Linear interface reconstruction

# a) LVIRA: optimization perspective

# LVIRA

LVIRA: Least squares Volume of fluids Reconstruction Algorithm. [E.G. Puckett, A volume-of-fluid interface tracking algorithm with applications to computing shock wave refraction, 1991]




- Take a <u>3x3 stencil</u> around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

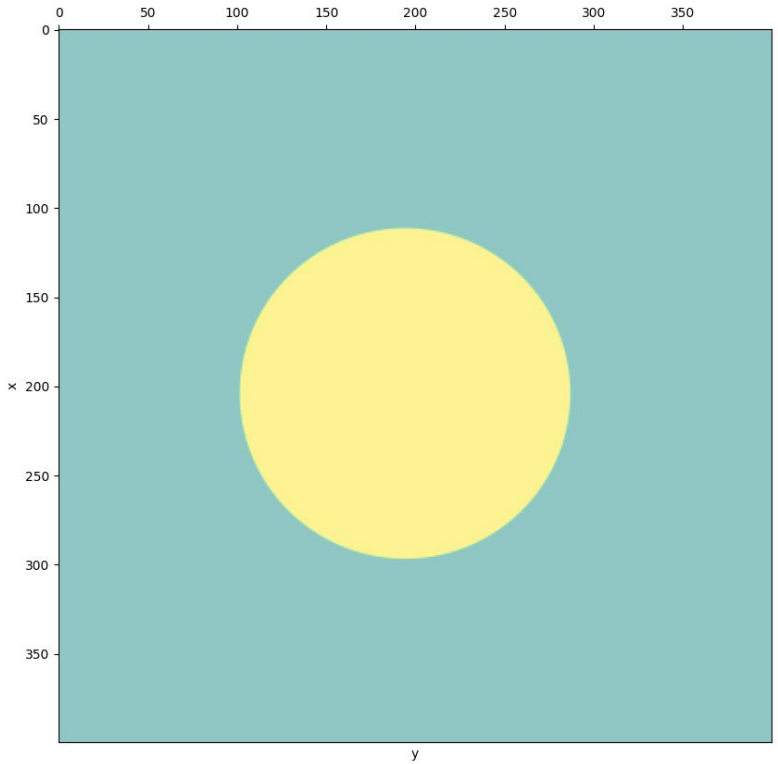
$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

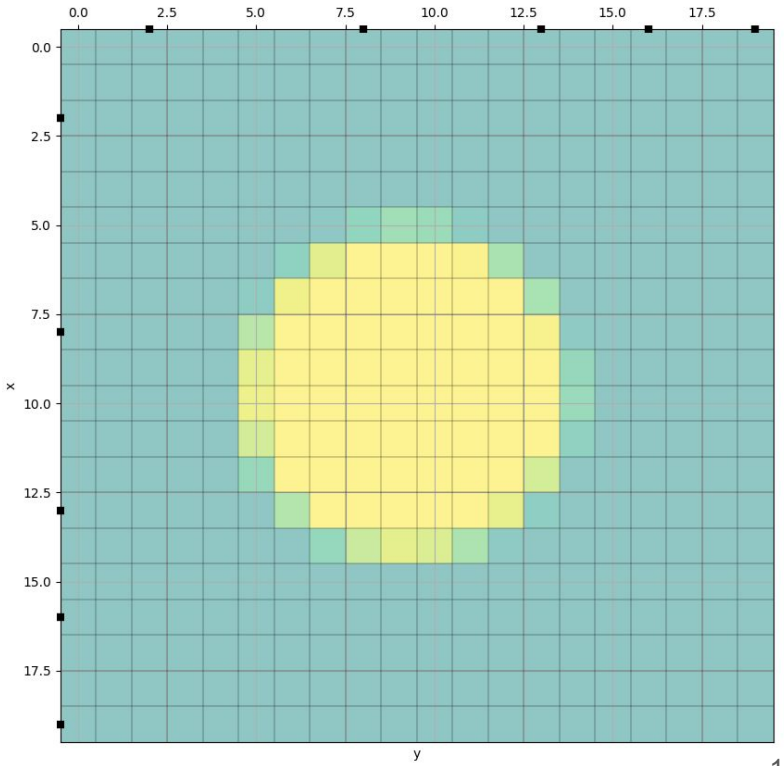
[Cohen, Dolbeault, Mula, S; Non linear approximation spaces for inverse problems; 2023]

<u>Linear methods</u> can't achieve better than **O(*h*$^1$)**

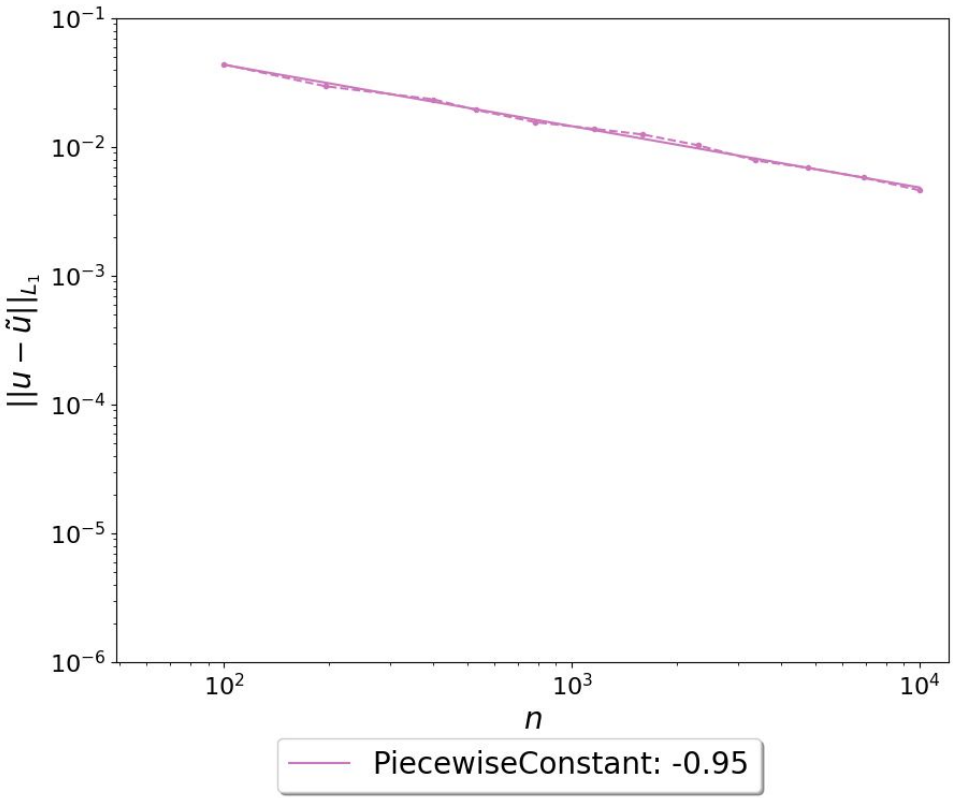
<u>Non-Linear methods</u>, Linear interfaces: **O(*h*$^2$)**

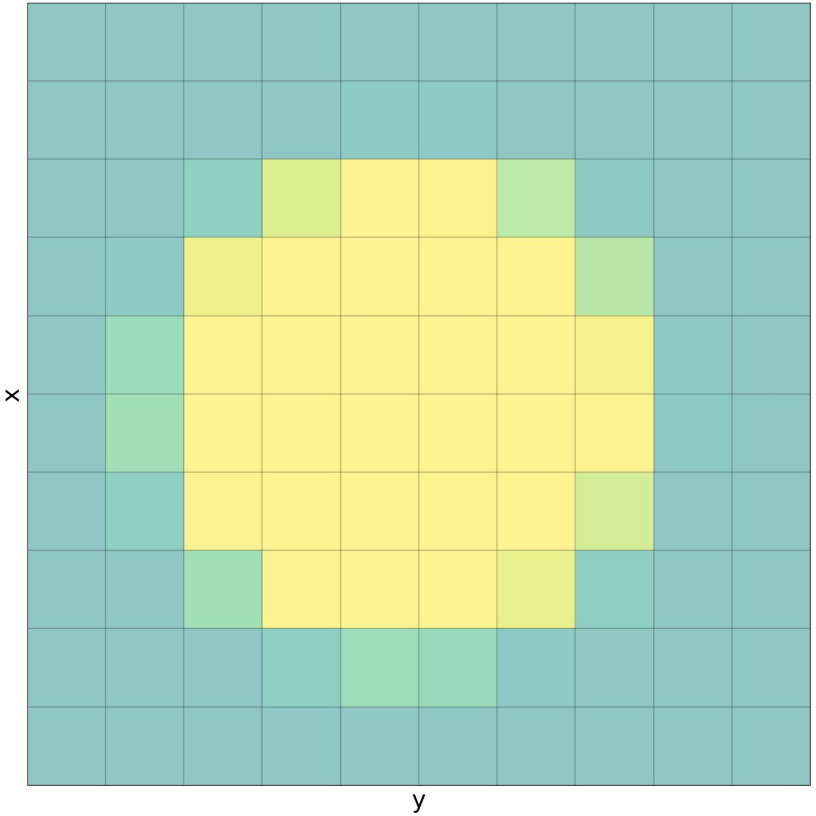
# Test case: the circle



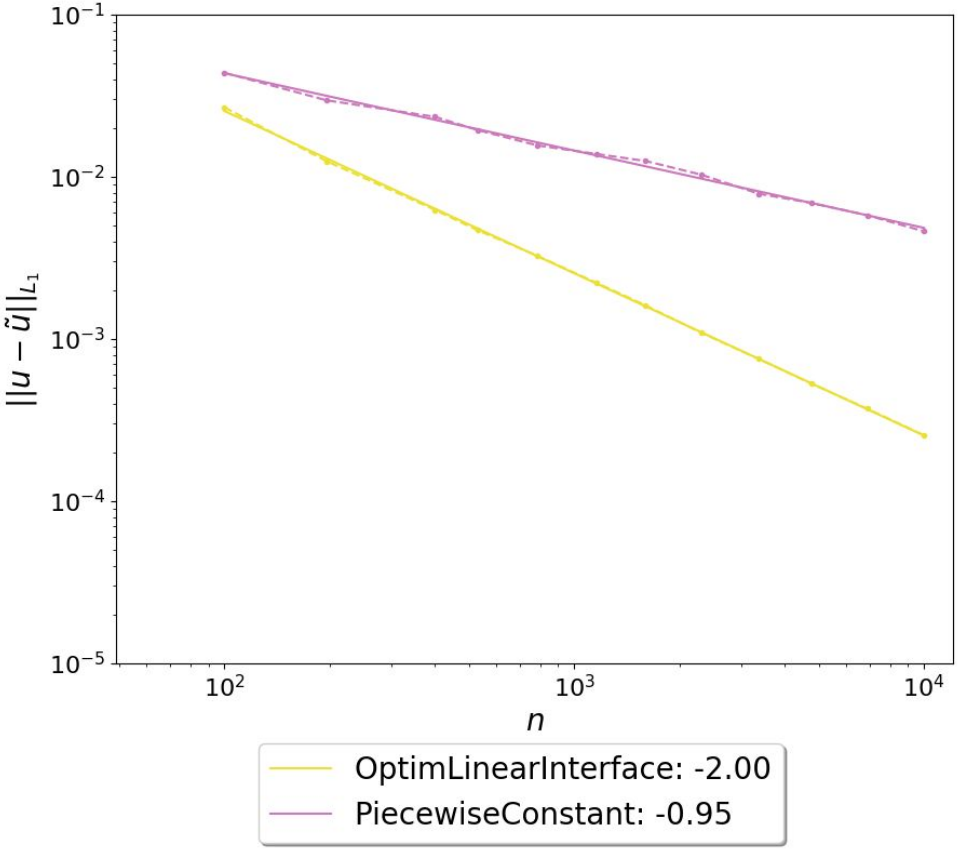



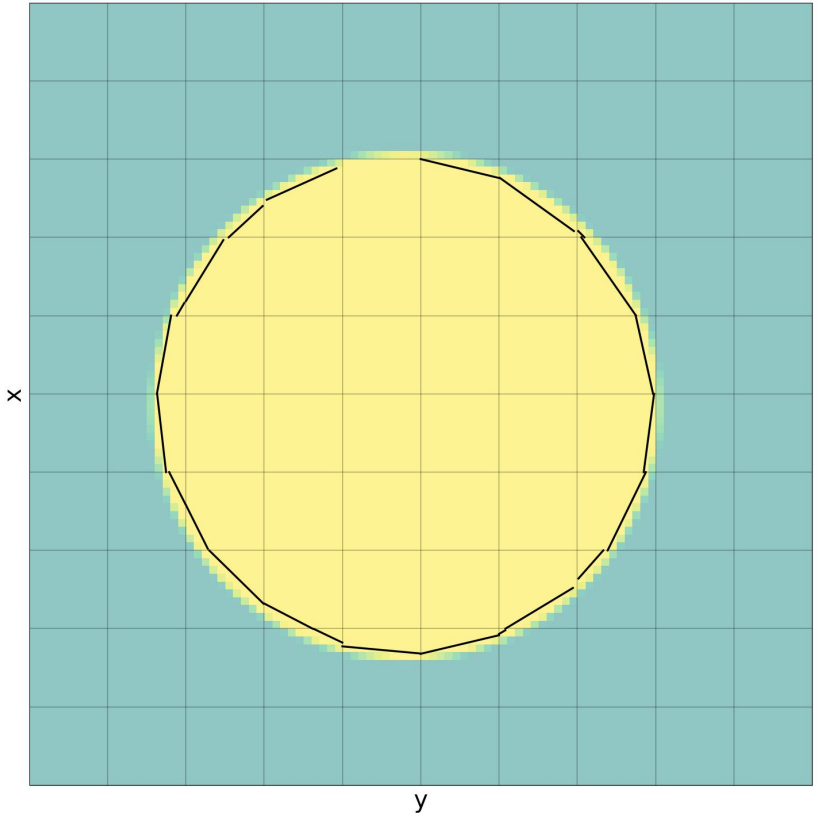
# Piecewise constant: convergence and reconstruction



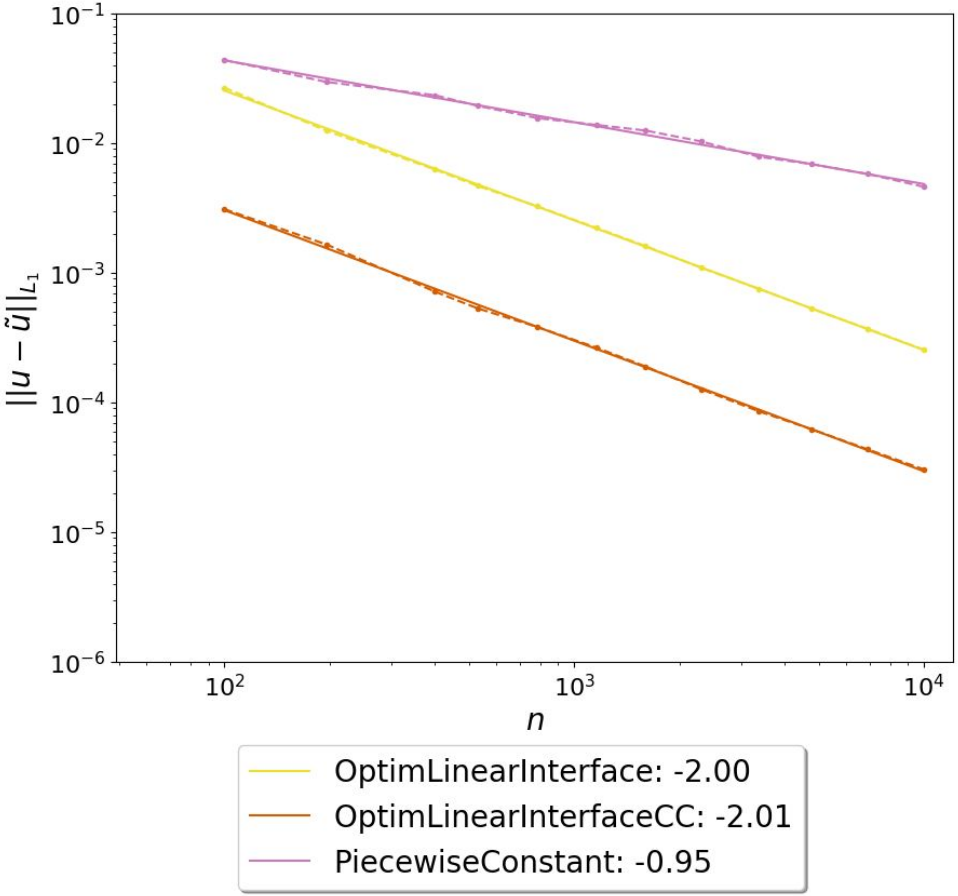



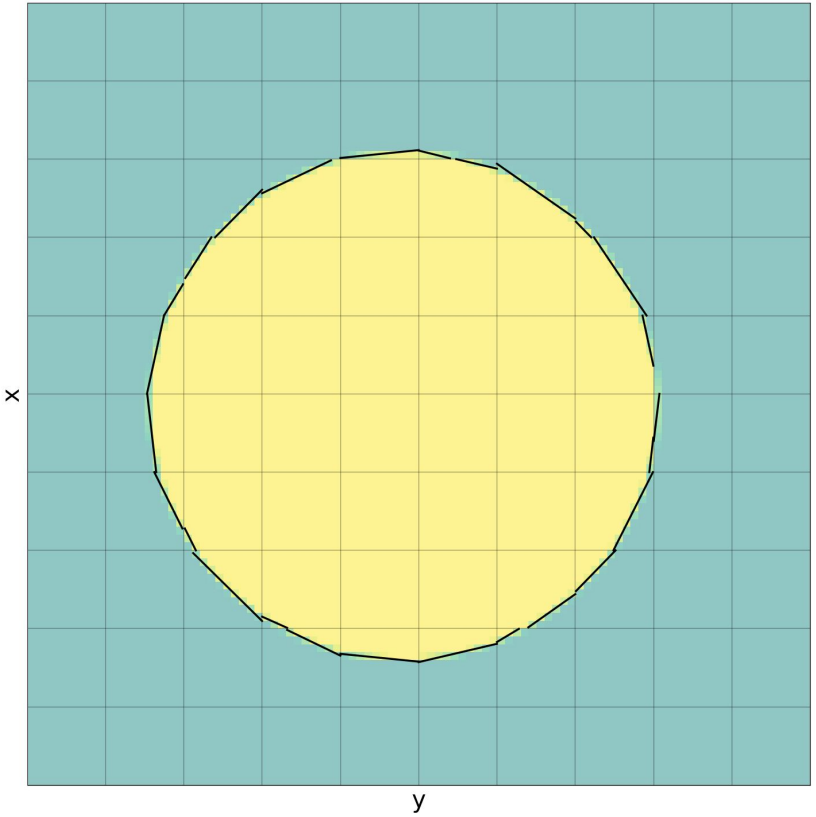
# Linear optimized: convergence and reconstruction



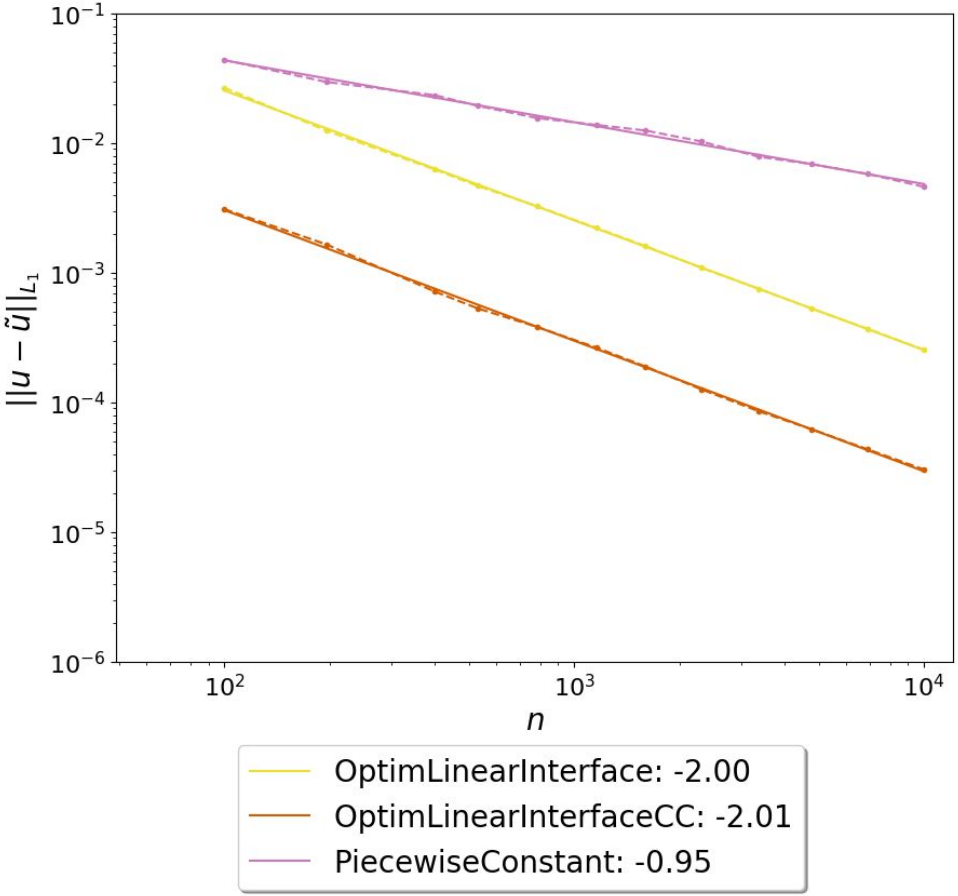



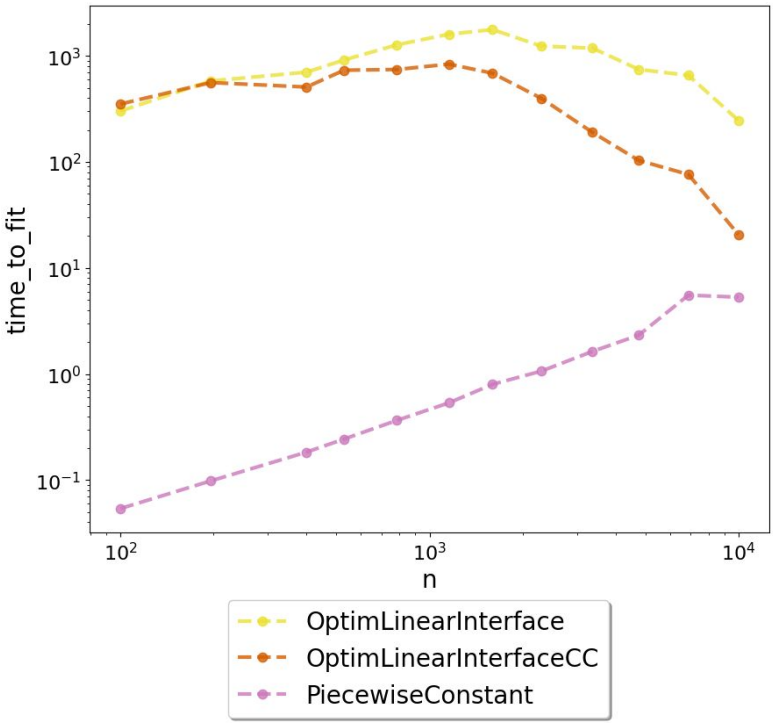
# Linear optimized Central Cell: convergence and reconstruction



# Linear optimized Central Cell: convergence and time

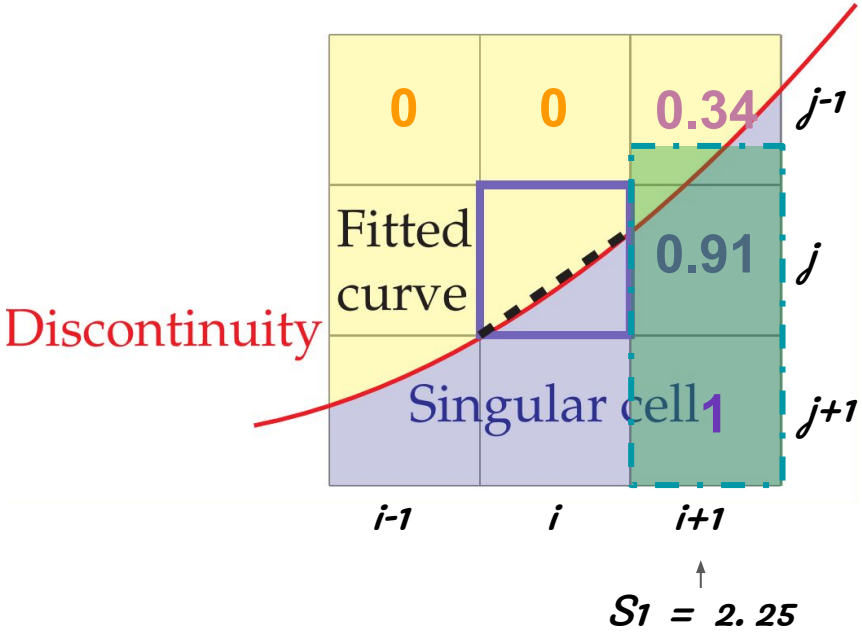


# b) ELVIRA: avoiding optimization

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluids Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a 3x3 stencil around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by finite differences, three for each orientation.

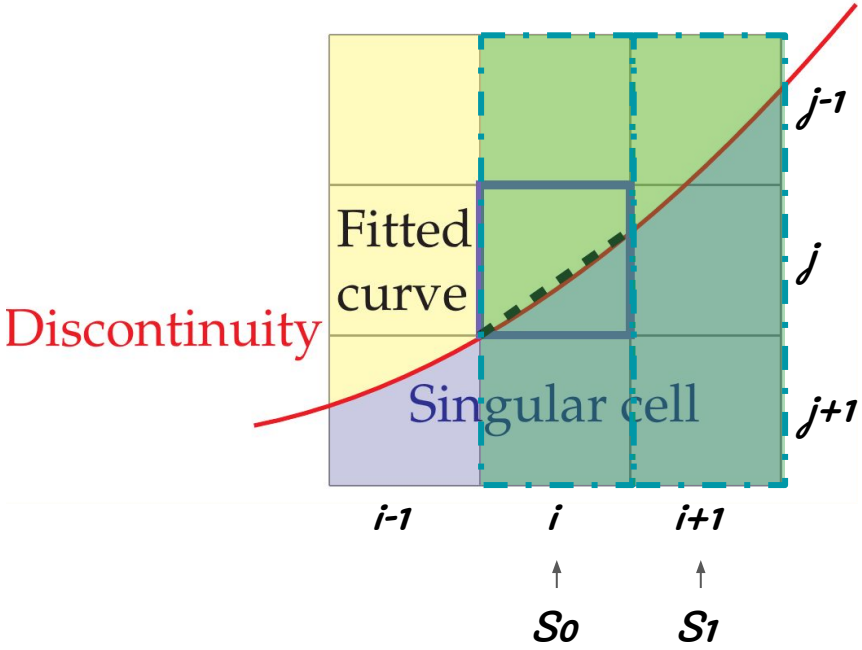
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluids Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a 3x3 stencil around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by finite differences, three for each orientation.

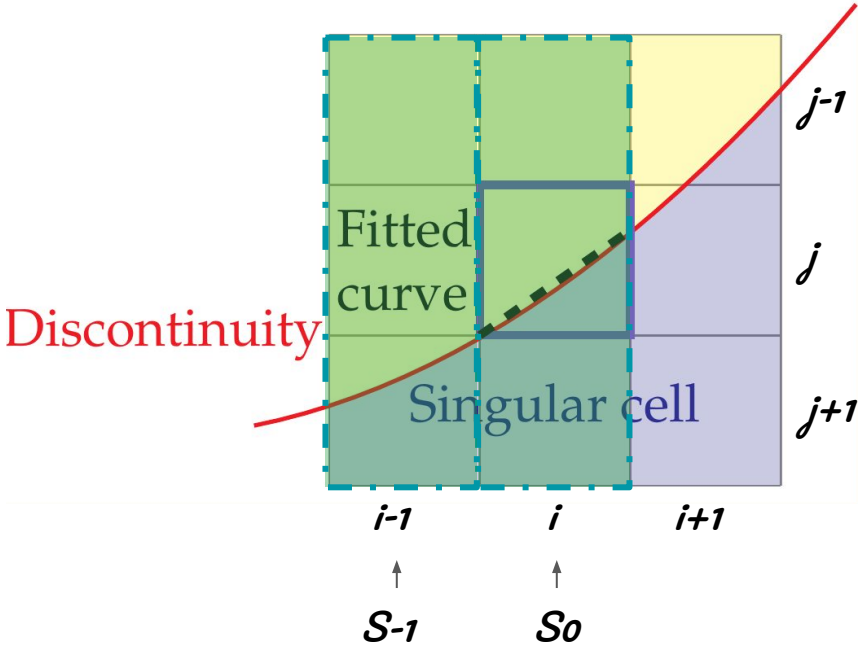
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluIds Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a 3x3 stencil around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by finite differences, three for each orientation.

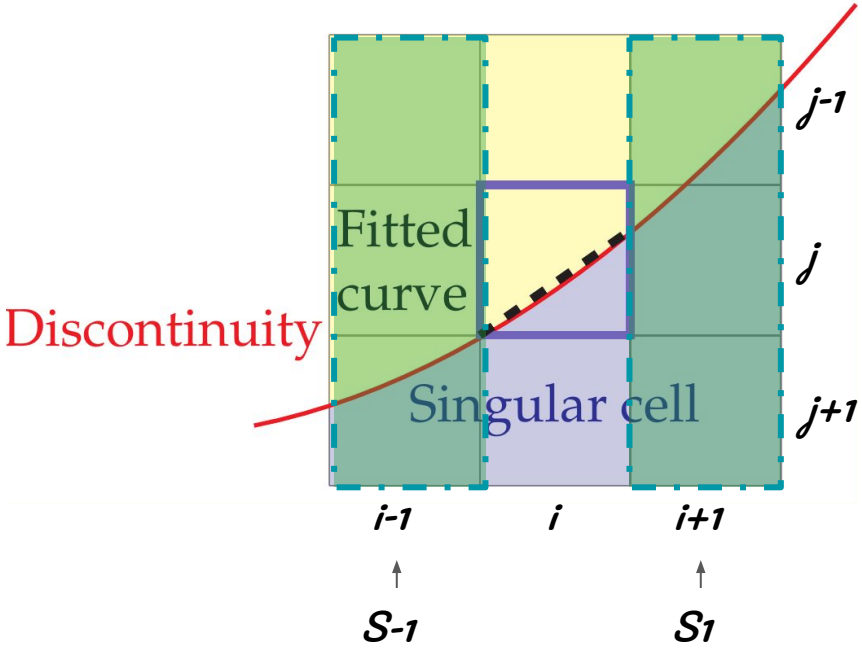
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluids Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a 3x3 stencil around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by finite differences, three for each orientation.

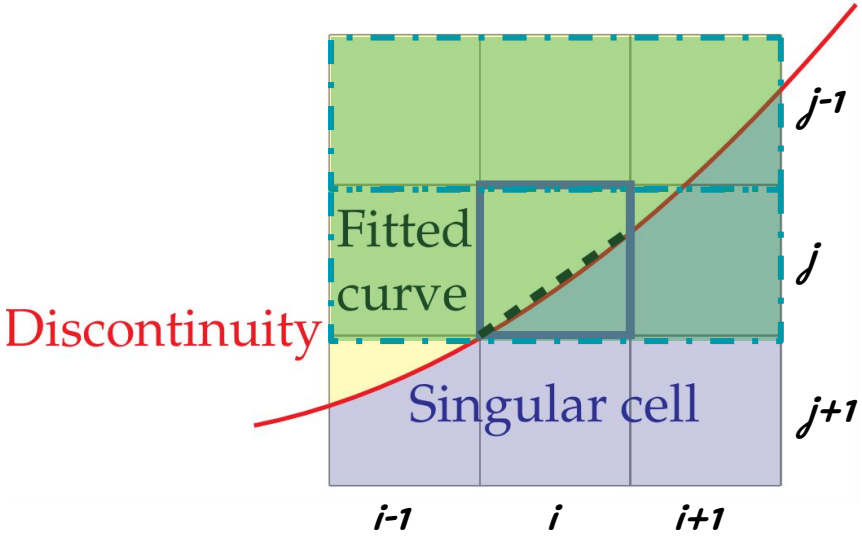
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluIds Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a <u>3x3 stencil</u> around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by <u>finite differences</u>, three for each orientation.

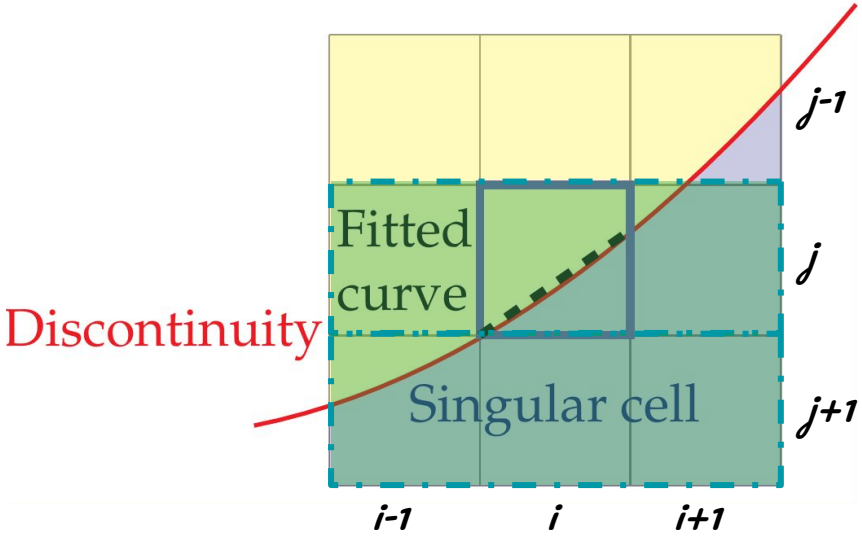
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluids Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a <u>3x3 stencil</u> around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

- Calculate only **6** slopes ***m*** by <u>finite differences</u>, three for each orientation.

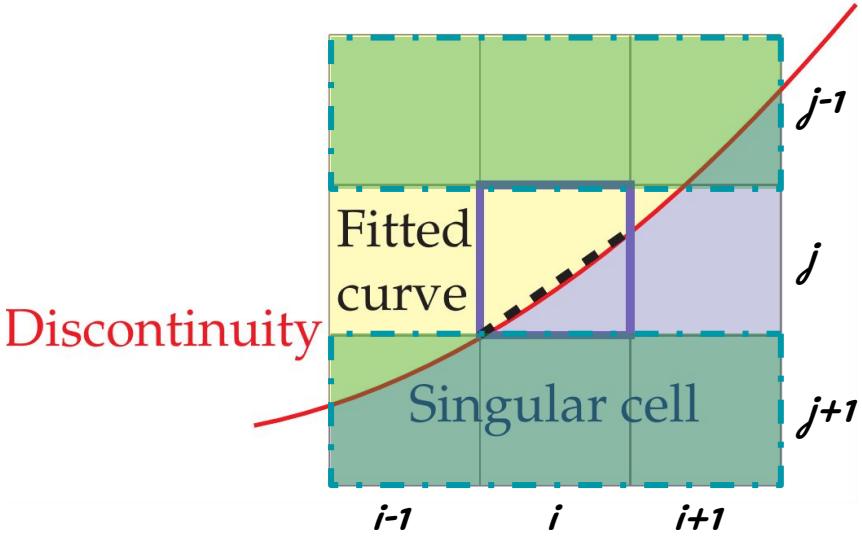
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA

ELVIRA: Efficient Least squares Volume of fluids Reconstruction Algorithm.[Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]




- Take a 3x3 stencil around the singular cell.
- Find the parameters ***θ = (m, b)*** of the line ***mx+b*** that minimize the ***l2*** error between the true cell averages ***a*** and the reconstructed ones

$$\theta^* = \underset{\theta}{argmin} \sum_{\alpha,\beta=(-1,-1)}^{(1,1)} \left(a_{i+\alpha,j+\beta} - \tilde{a}(\theta)_{i+\alpha,j+\beta}\right)^2$$

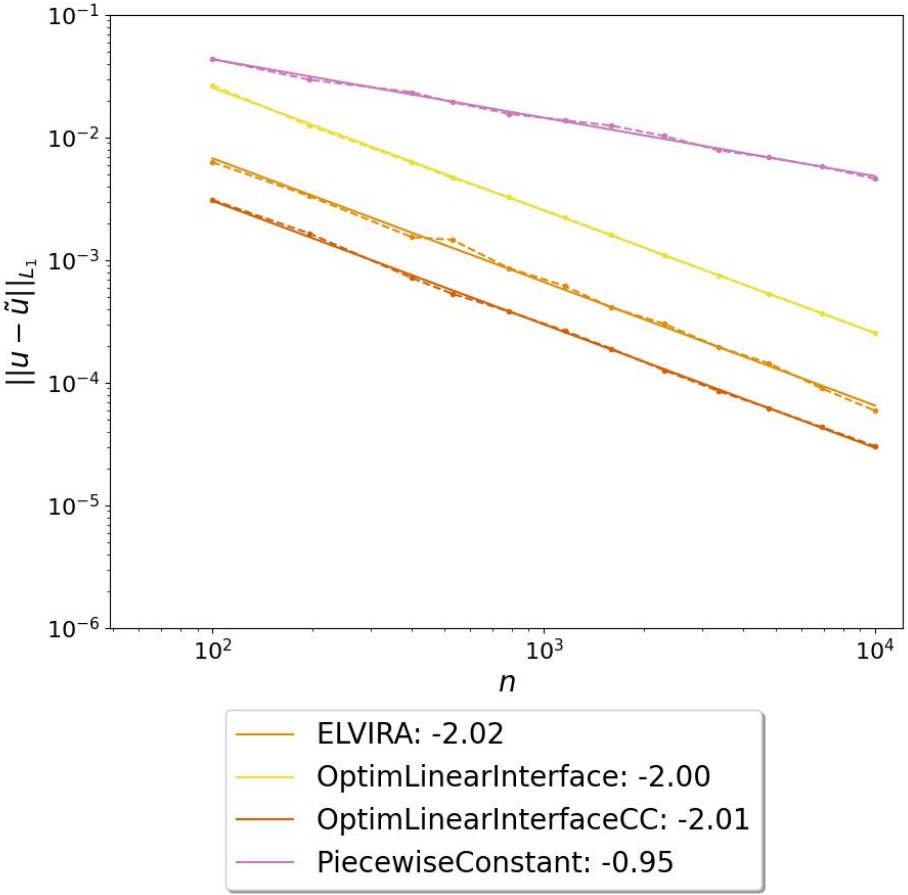
- Calculate only **6** slopes ***m*** by finite differences, three for each orientation.

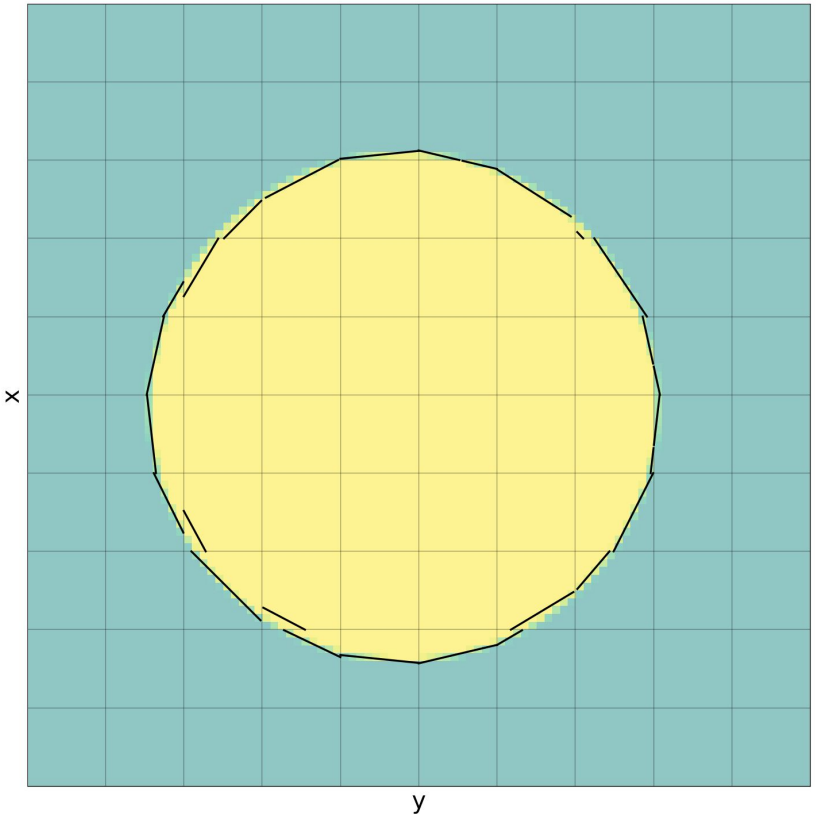
$$S_{i+\alpha} = \sum_{\beta=-1}^{1} a_{i+\alpha,j+\beta}$$

$$m \approx S_1 - S_0, \quad m \approx S_0 - S_{-1}, \quad m \approx \frac{S_1 - S_{-1}}{2}$$

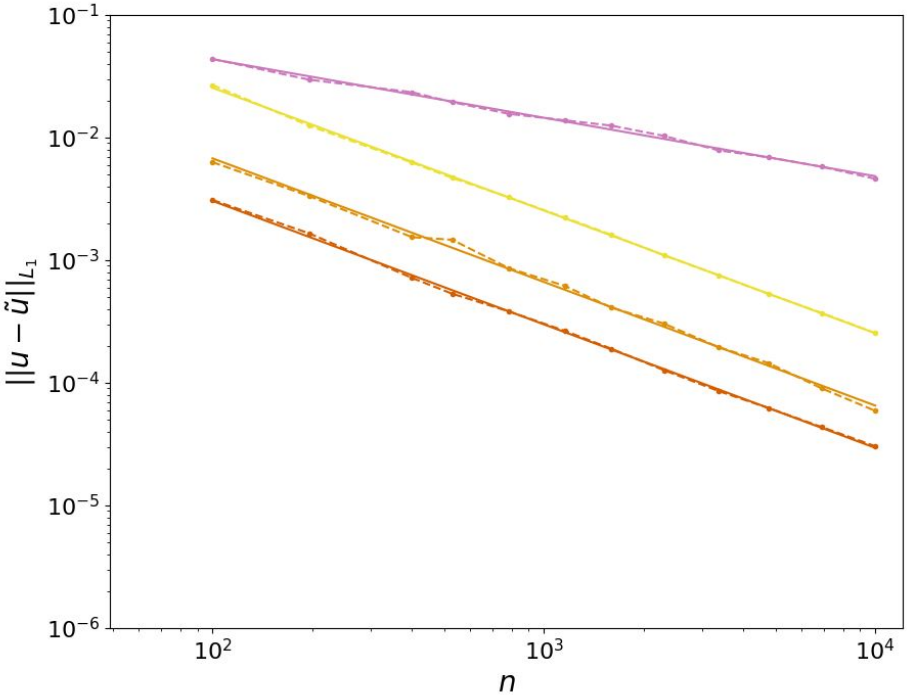
If it the true discontinuity is a line, one of the 6 correspond to the exact parameters.

# ELVIRA: convergence and reconstruction

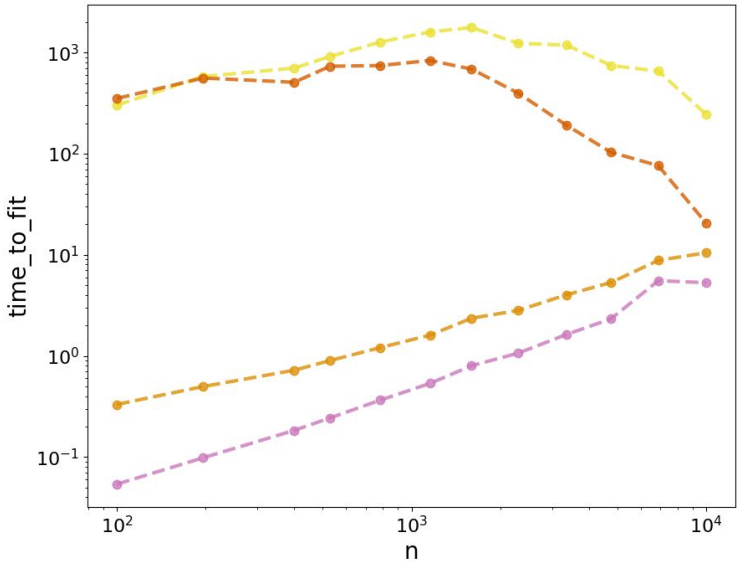


# ELVIRA: convergence and time



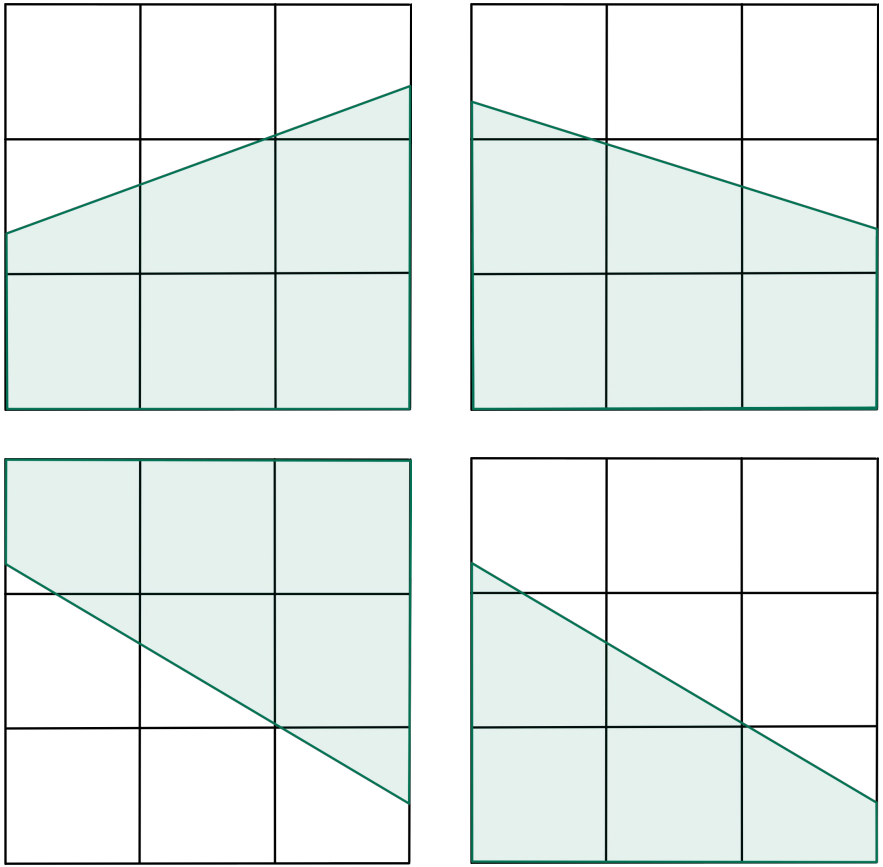

ELVIRA: -2.02
OptimLinearInterface: -2.00
OptimLinearInterfaceCC: -2.01
PiecewiseConstant: -0.95




ELVIRA
OptimLinearInterface
OptimLinearInterfaceCC
PiecewiseConstant

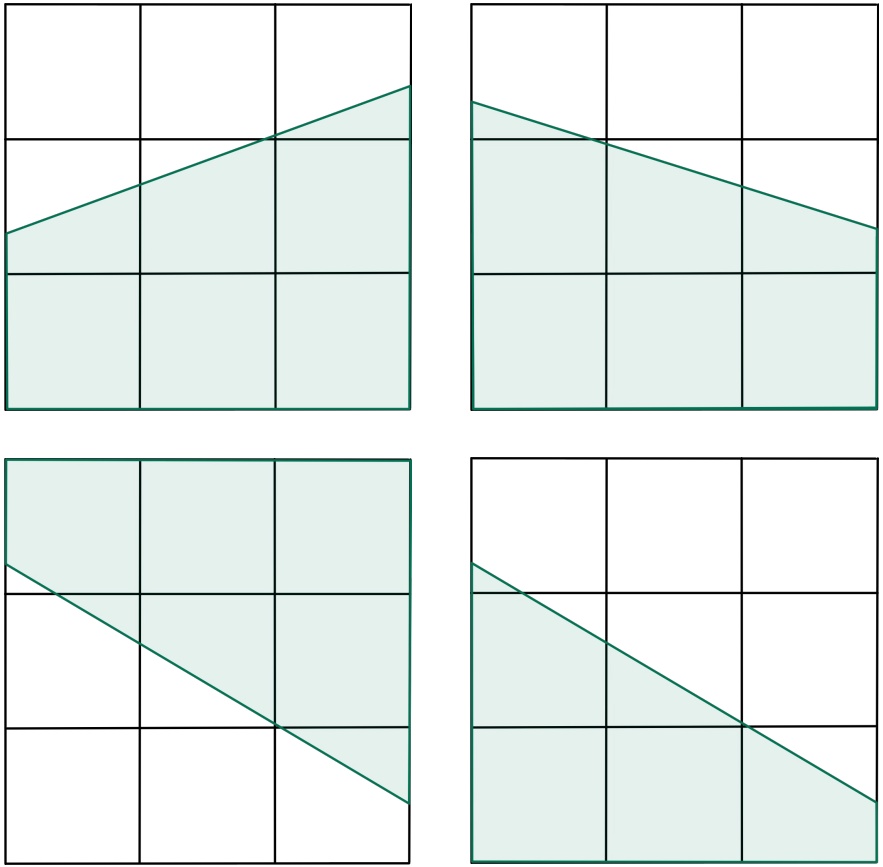
# (3) Machine learning interface reconstruction

# Building the database for learning lines



- 3x3 stencil
- Line parameters $\boldsymbol{\theta}$=(m,b)
  - $m \sim U[m_{min}, m_{max}]$
  - $b \sim U[-1, 1]$
- Calculate cell averages **a**
- $\mathbf{N_{train} = 10^6}$ pairs of (a, (m, b)) training examples.

# Building the database for learning lines

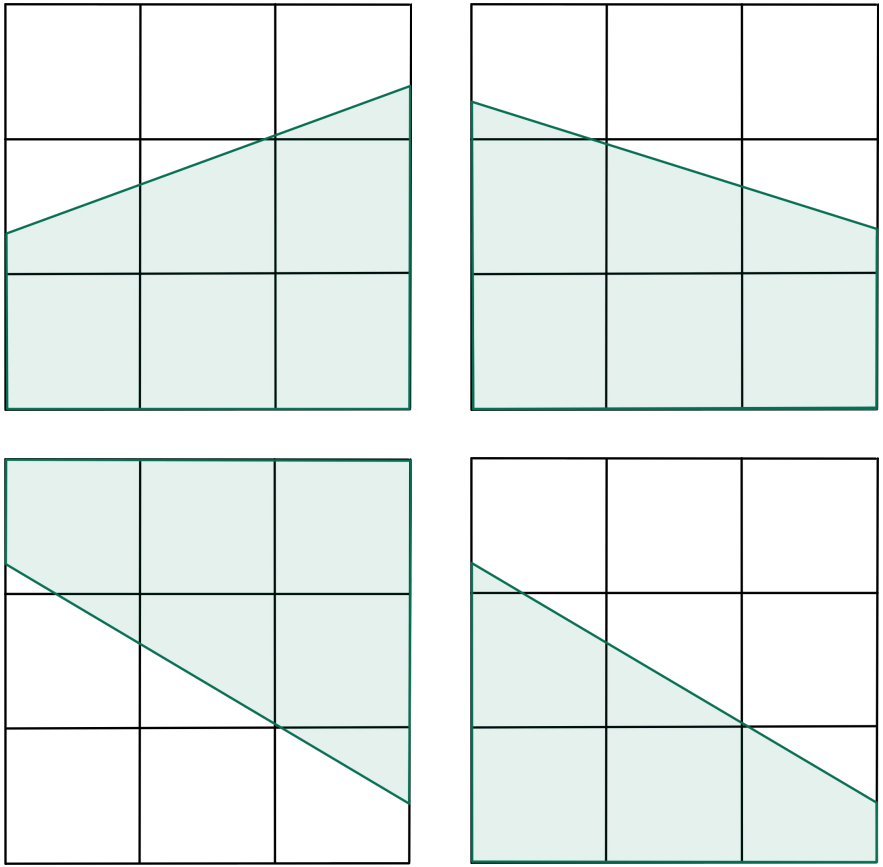


- 3x3 stencil
- Line parameters $\boldsymbol{\theta}$=(m,b)
  - m ~ U[$m_{min}$, $m_{max}$]
  - b ~ U[-1, 1]
- Calculate cell averages **a**
- $\mathbf{N_{train} = 10^6}$ pairs of (a, (m, b)) training examples.

Changing how to parametrize for learning:

- (m, b) -> unbounded
- **<u>(b, angle)</u>** -> not periodic
- (b, cos(angle), sin(angle))

# Building the database for learning lines



- 3x3 stencil
- Line parameters $\boldsymbol{\theta}$=(m,b)
  - m ~ U[$m_{min}$, $m_{max}$]
  - b ~ U[-1, 1]
- Calculate cell averages **a**
- $\mathbf{N_{train} = 10^6}$ pairs of (a, (m, b)) training examples.

Changing how to parametrize for learning:

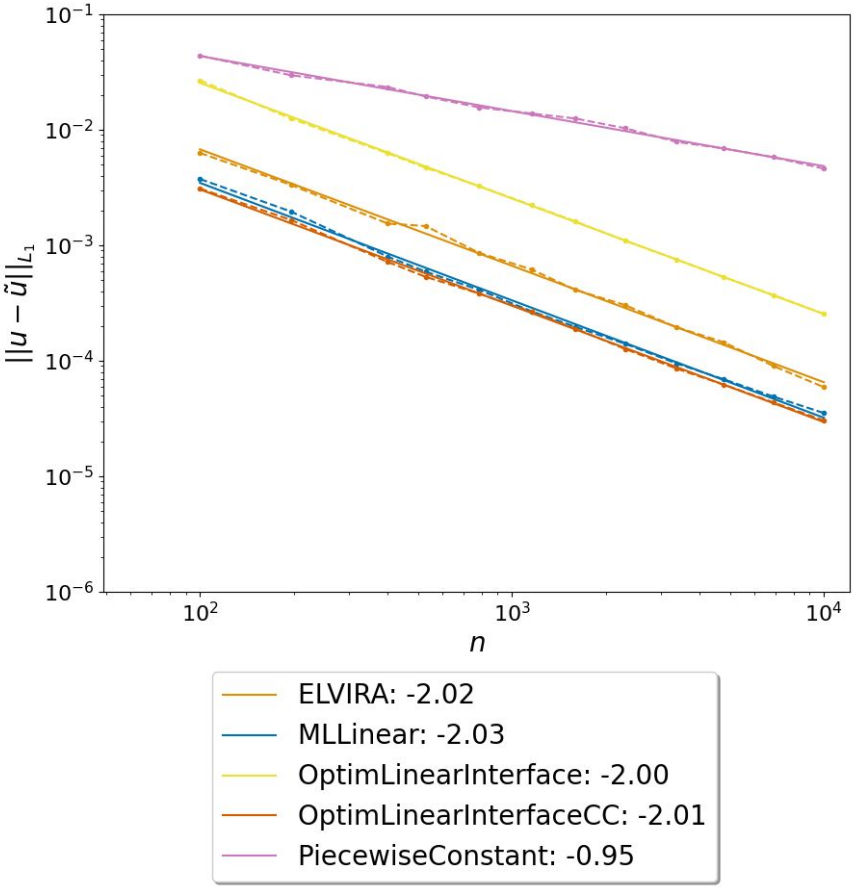
- (m, b) -> unbounded
- **(b, angle)** -> not periodic
- (b, cos(angle), sin(angle))

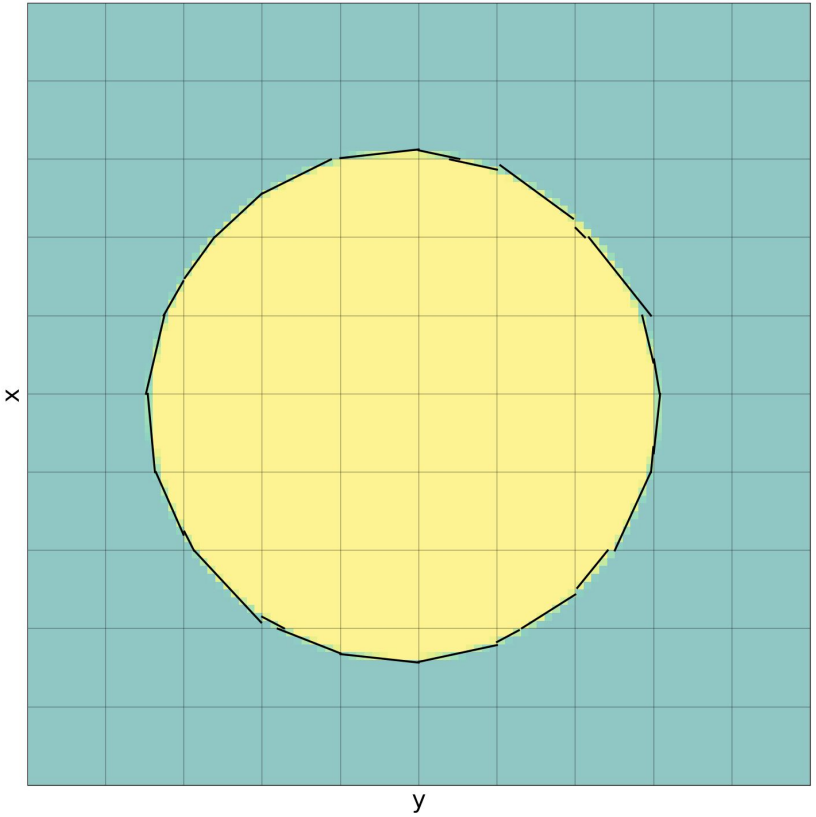
Learning methods:

- Decision Tree (scikit-learn)
- Neural network 20x20 ReLu (scikit-learn)
- Neural network 20x20 ReLu (pytorch)
- **Neural network 20x20 ReLu (tensorflow)**

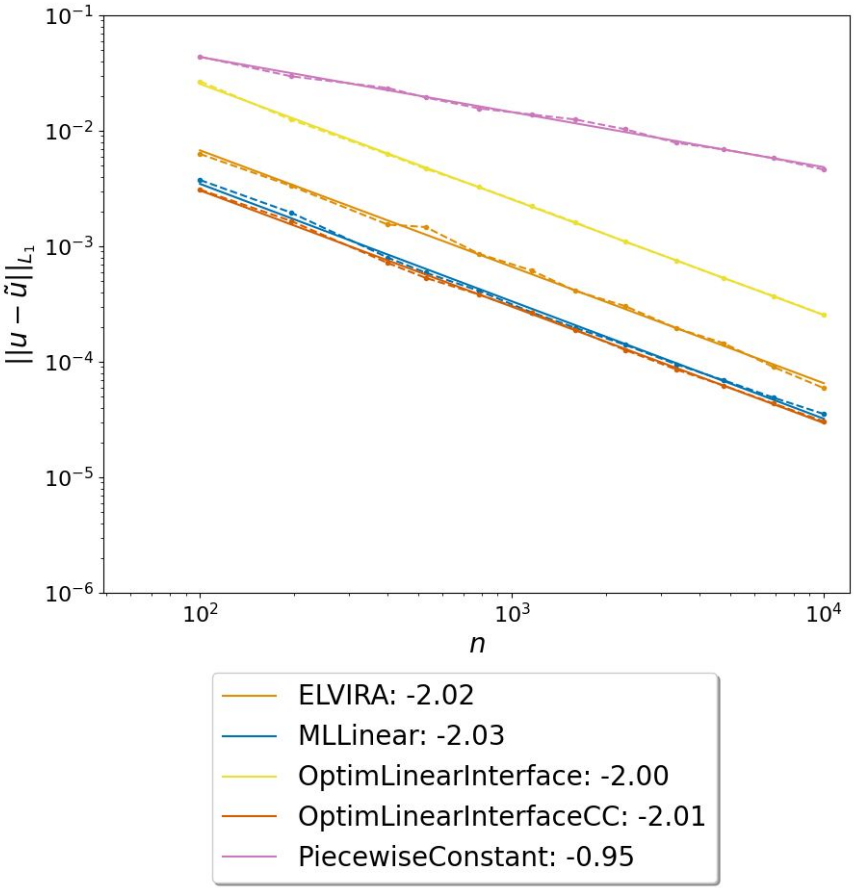
Add noise in the input to increase generalization capabilities of the learning method.

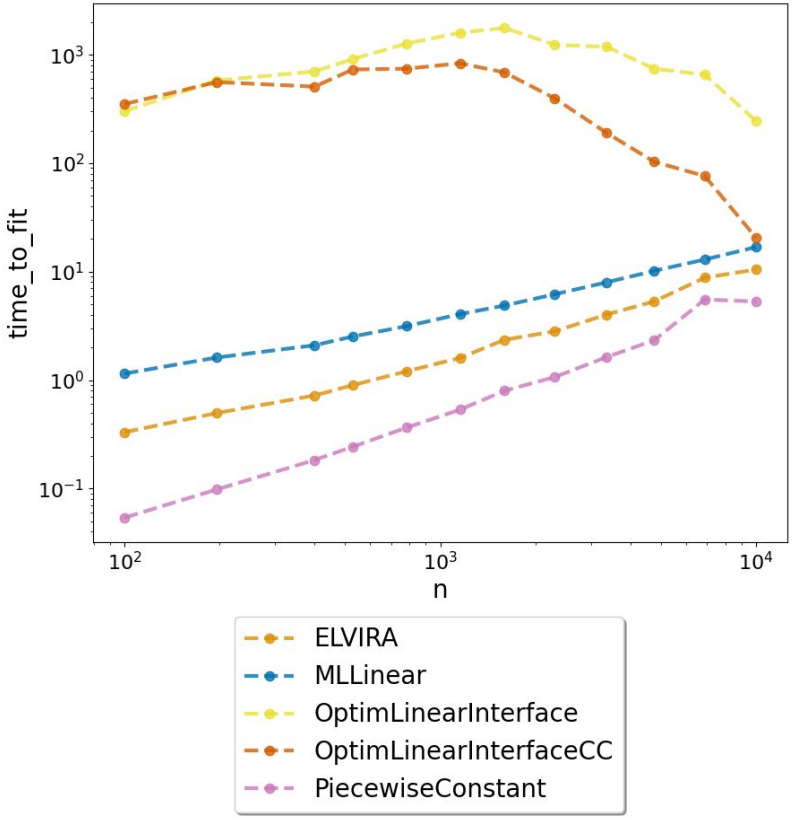
# ML Linear: convergence and reconstruction



# ML Linear: convergence and time



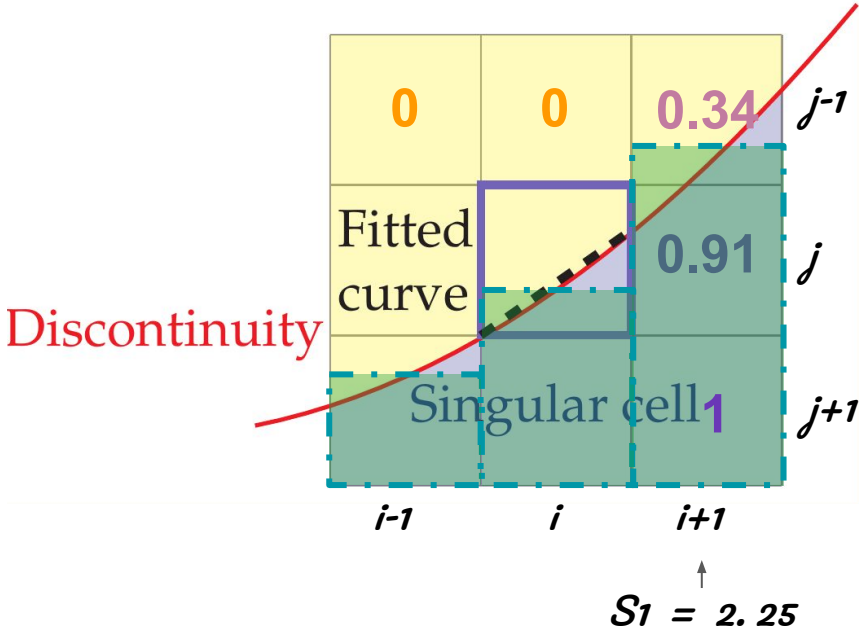



# (4) Polynomial interface reconstruction

AERO: Algorithms for Edge Reconstruction using
Oriented stencils

# AERO

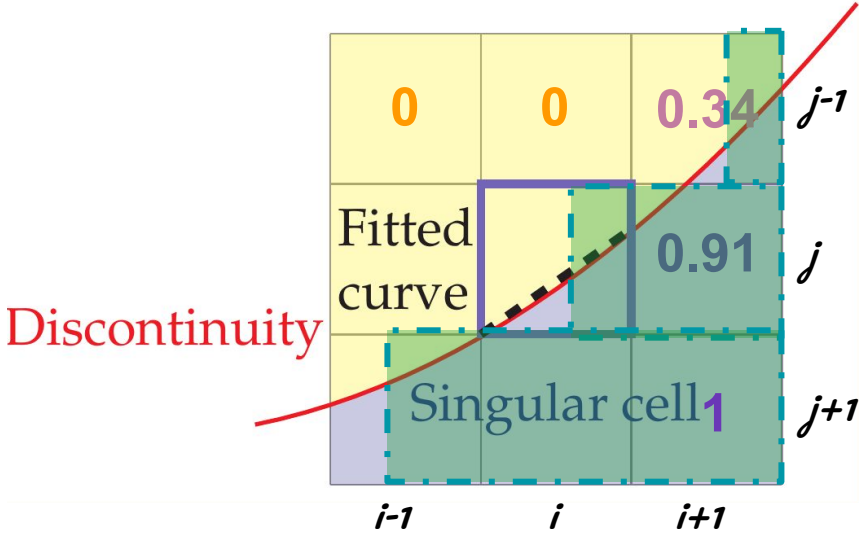
[Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]



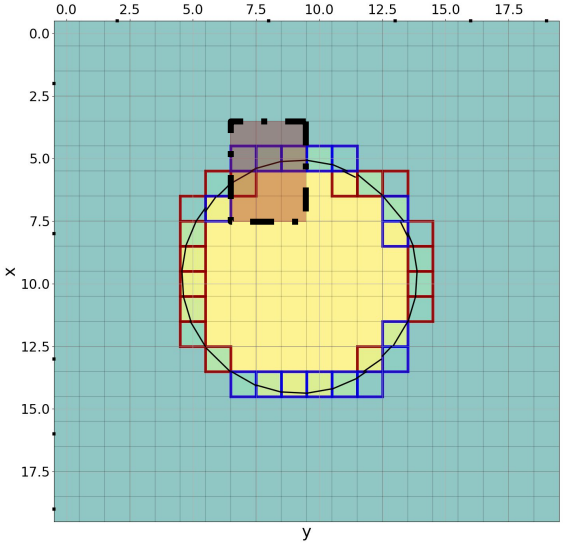

- Choose orientation (H or V)
- Choose appropriate stencil

# AERO

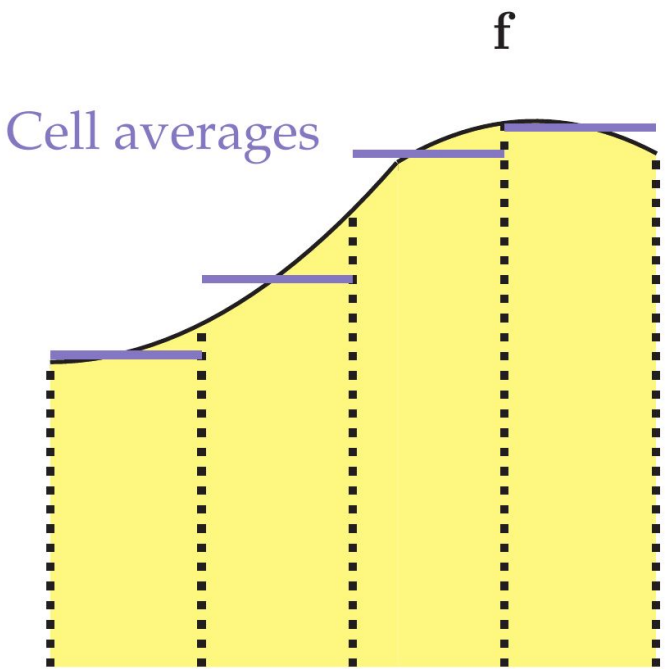
[Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]




- Choose orientation (H or V)
- Choose appropriate stencil

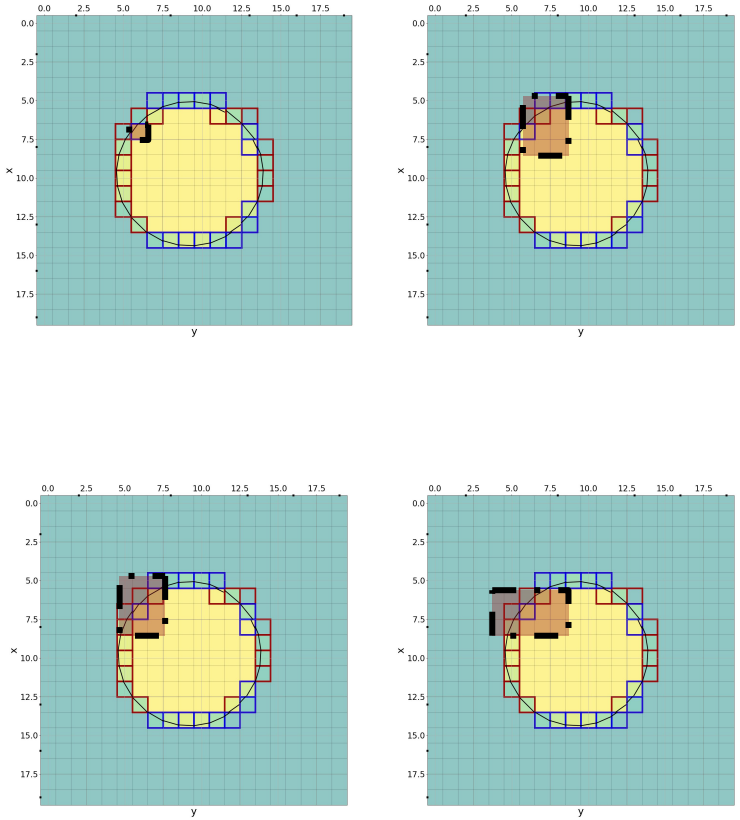


# AERO - Polynomial




- Choose orientation (H or V)
- Choose appropriate stencil
- Fit a polynomial like in the regular 1D case.

# AERO orientation

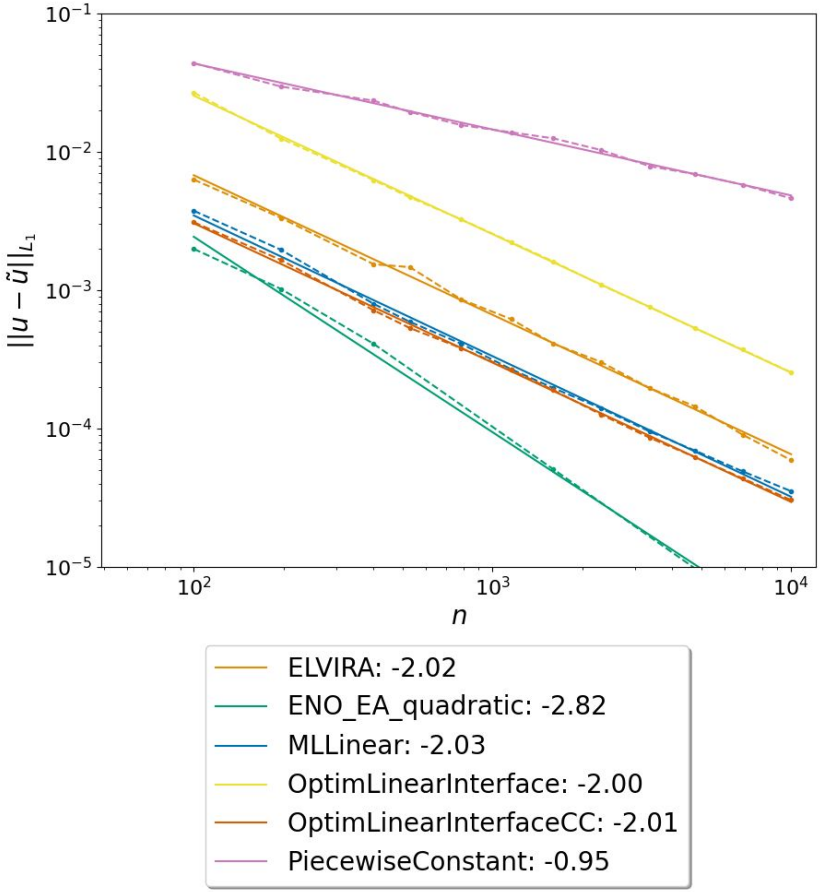


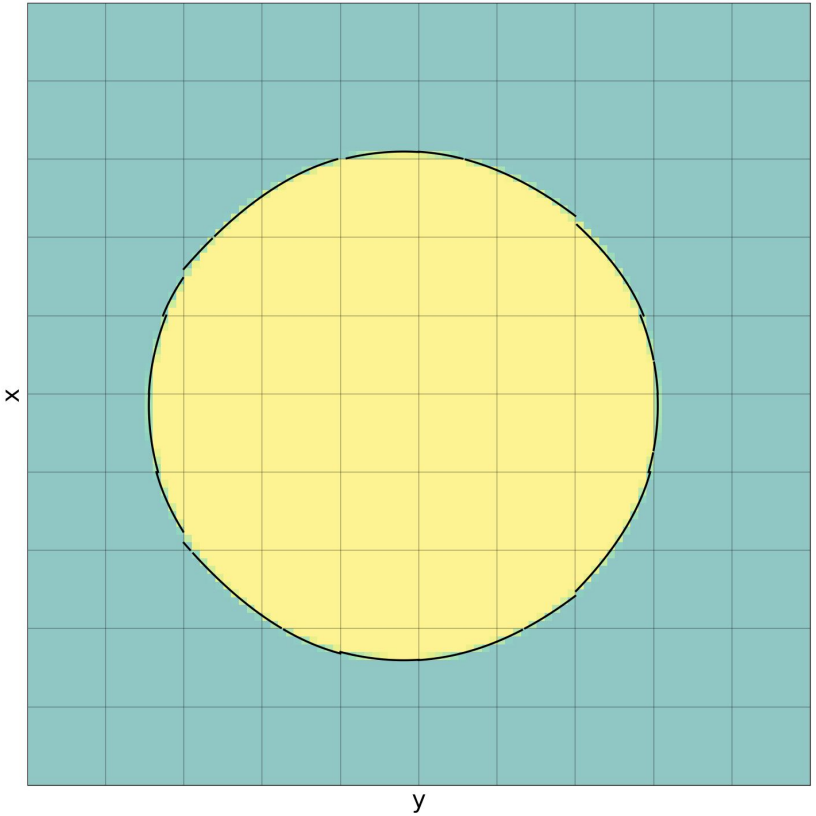

- Choose orientation (H or V)
- Choose appropriate stencil
- Fit a polynomial like in the regular 1D case.

**Problem**: in borders where there is change of orientation.

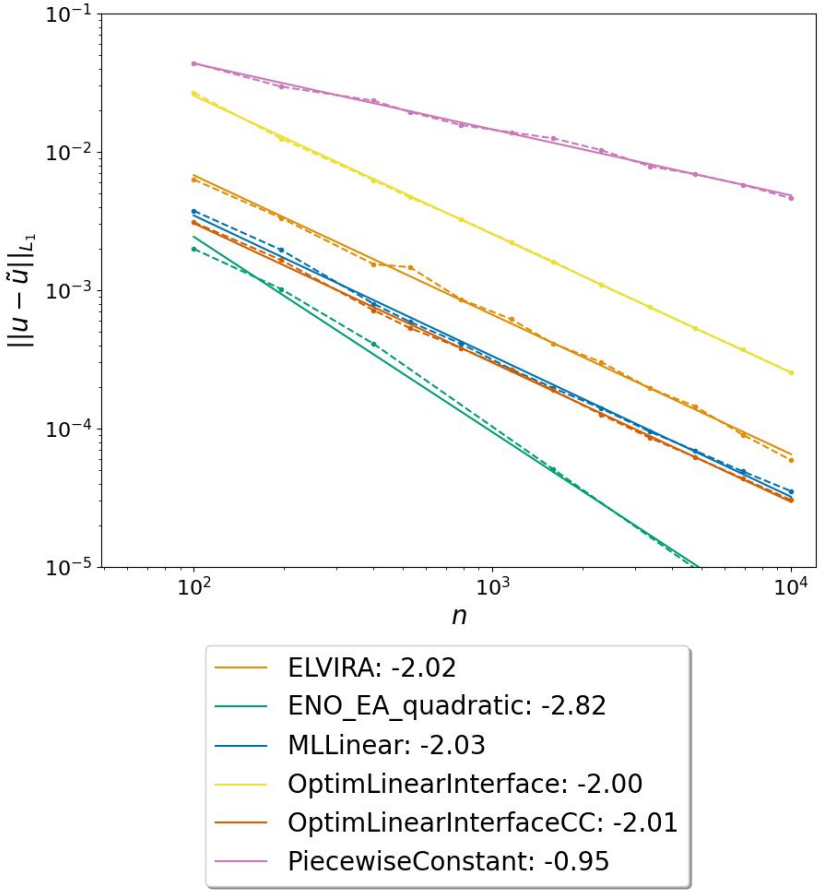
a) <u>Adaptive stencil</u> (like in ENO)
b) Following the idea of ELVIRA of minimizing $I_2$ to decide, if it is possible to fit a polynomial for each orientation and choose the <u>orientation a posteriori</u> by minimization.

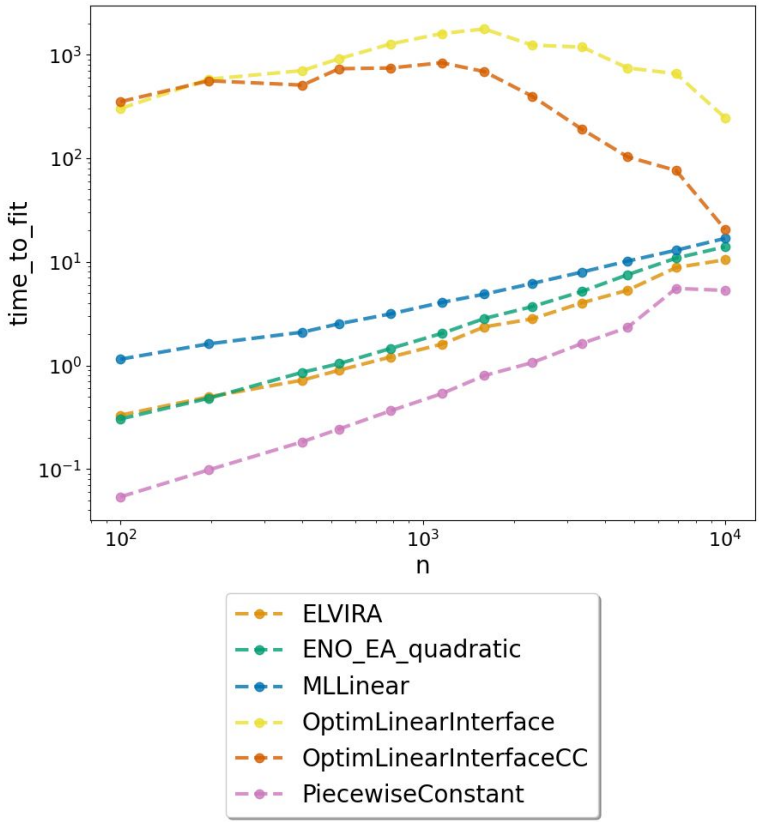
# Polynomial AERO convergence and reconstruction



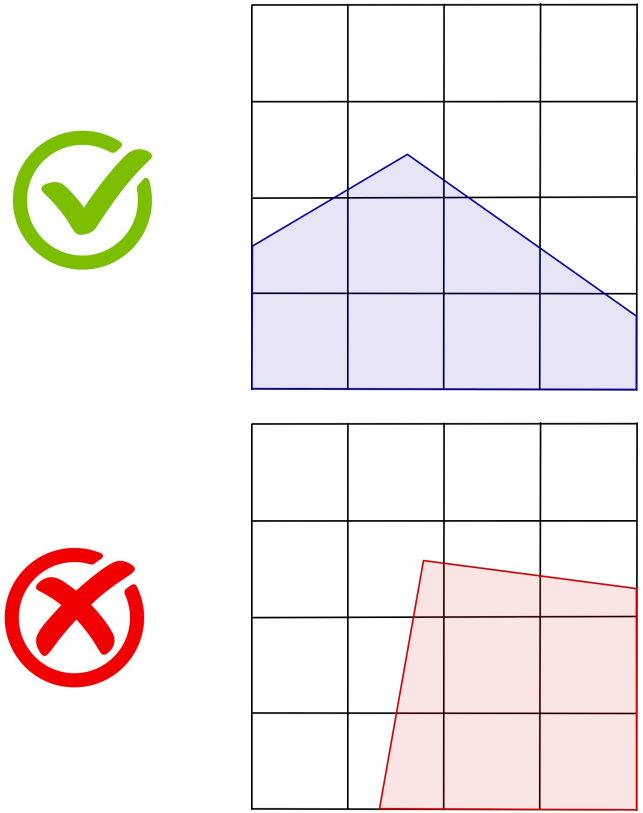
# Polynomial AERO convergence and time



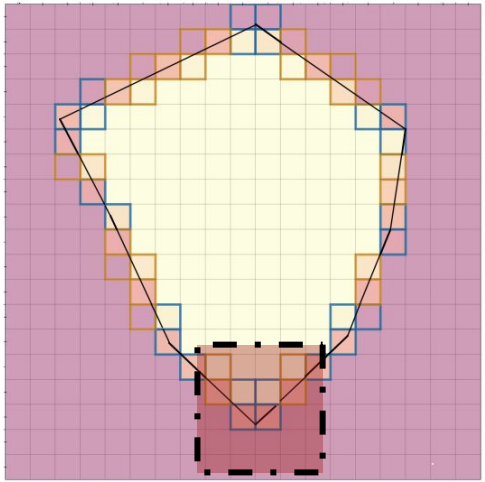



# (6) Vertices

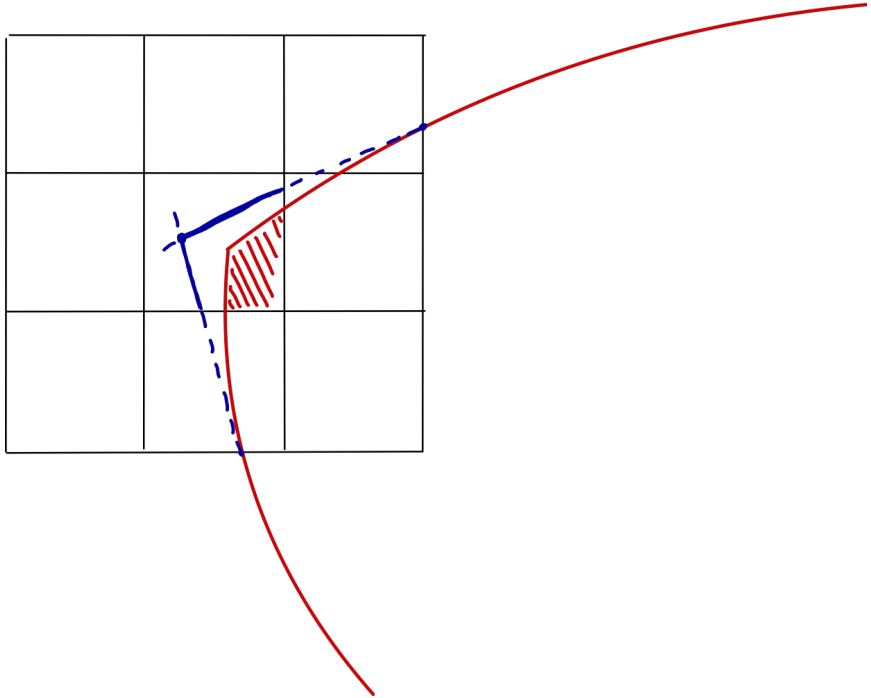
# AERO - Vertex

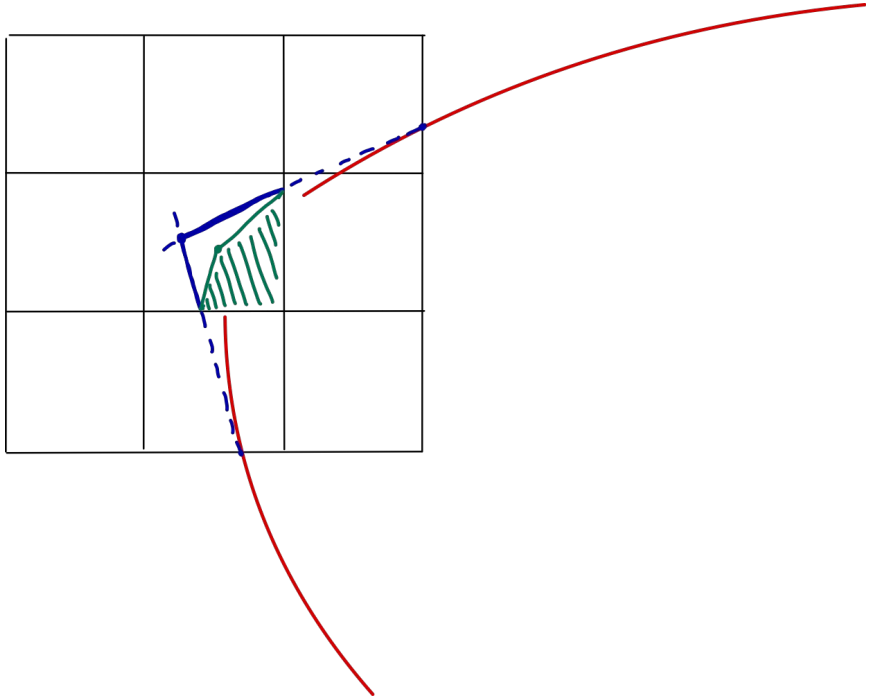


- Choose orientation (H or V)
- Choose appropriate stencil so the interface crosses the sides.

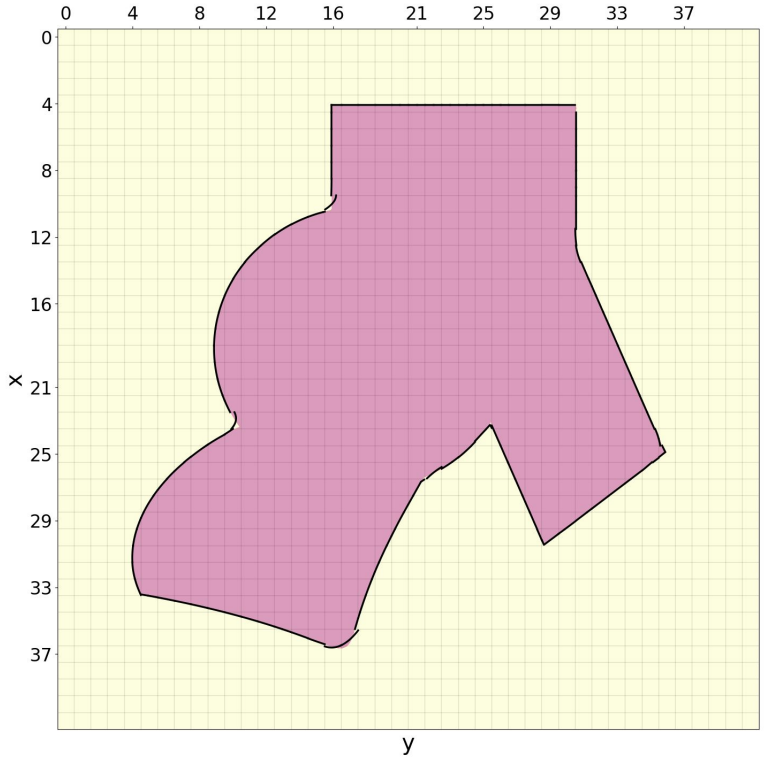


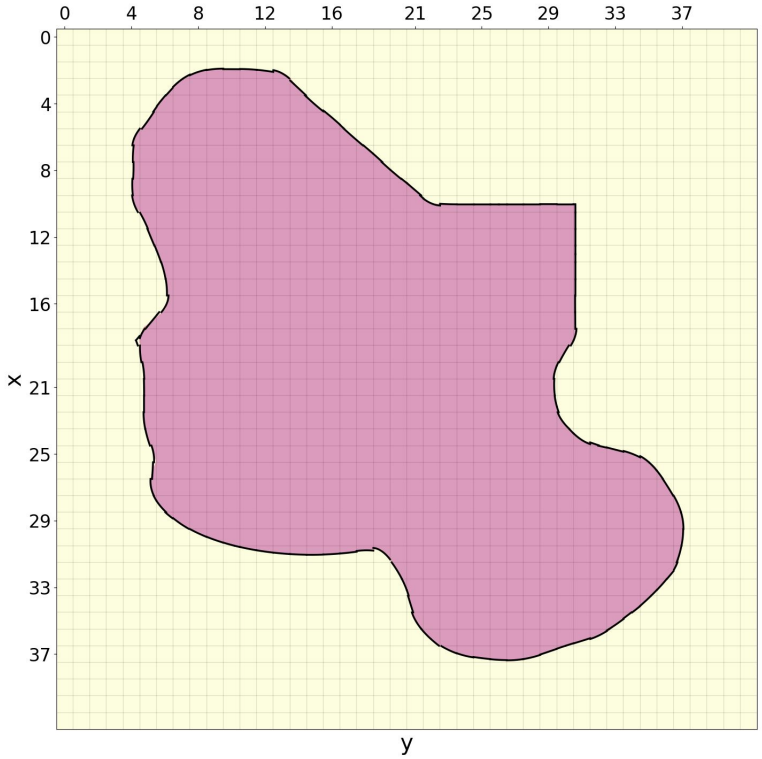
# TEM: Tangent extension method for vertex reconstruction





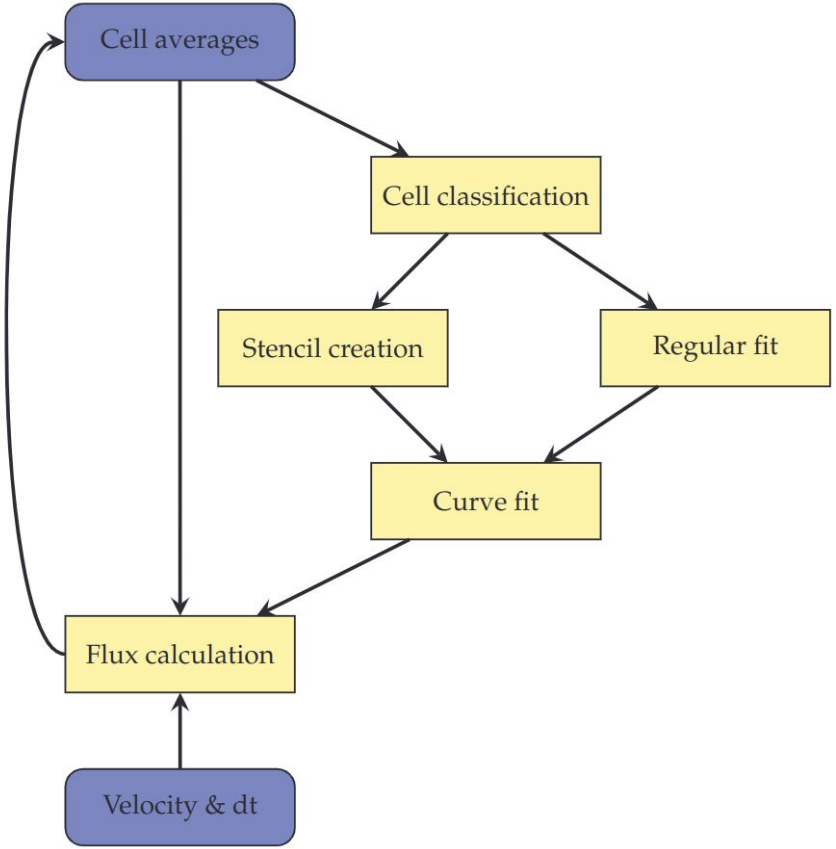
# AERO and Vertices in action



# (5) Summary and conclusions

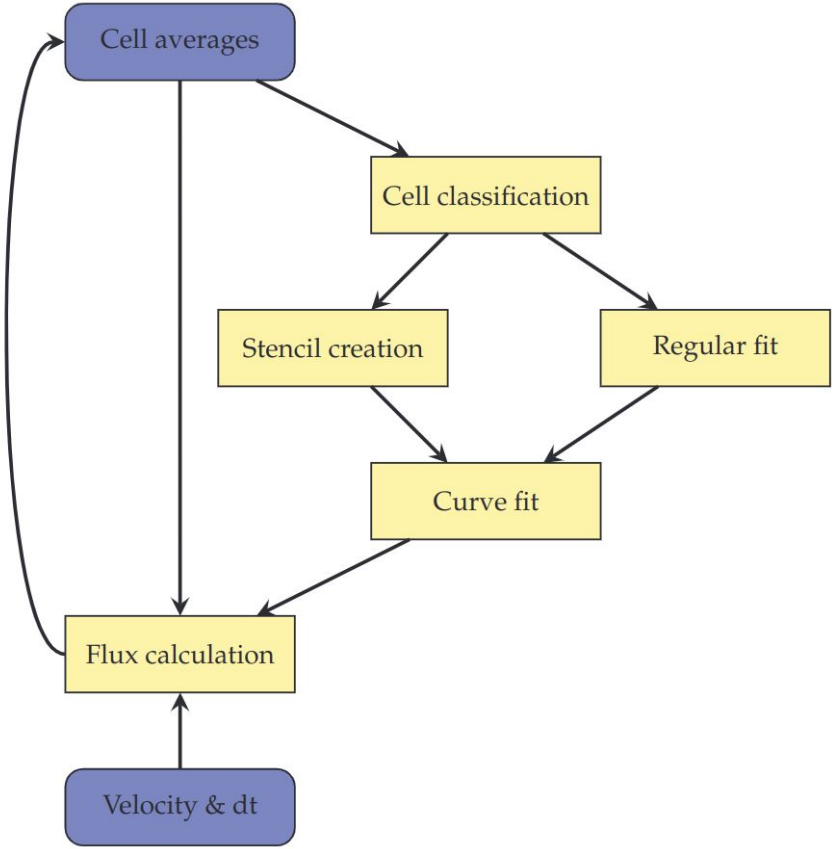
# Image reconstruction pipeline future work




**Curve reconstruction**:

- Fitting lines
  - **ELVIRA** [Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]
  - **ENO-EA** [Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]
  - **ML**
- Fitting polynomials
  - **AERO**
- Fitting other curves
  - **Algebraic closed form**? **Not feasible**
- Treating vertices
  - **ML** for 90º vertices [Despres, Jourdren, Machine Learning design of Volume of Fluid schemes for compressible flows, 2020]
  - **AERO**
  - **TEM**

# Image reconstruction pipeline future work




**Curve reconstruction**:

- Fitting lines
  - **ELVIRA** [Pilliod, Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, 2003]
  - **ENO-EA** [Arandiga, Cohen, et all, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, 2008]
  - **ML**
- Fitting polynomials
  - **AERO**
  - **ML**?
- Fitting other curves
  - **Algebraic closed form**? **Not feasible**
  - **ML**?
- Treating vertices
  - **ML** for 90º vertices [Despres, Jourdren, Machine Learning design of Volume of Fluid schemes for compressible flows, 2020]
  - **AERO**
  - **TEM**
- Direct flux calculation
  - **ML** (linear interfaces)
- Piecewise regular functions

# Thanks!

# Reconstruction 2D (or more)

**<u>Difficulty</u>**: find a parametrized curve $\boldsymbol{C(\mu)}$ instead of a point.

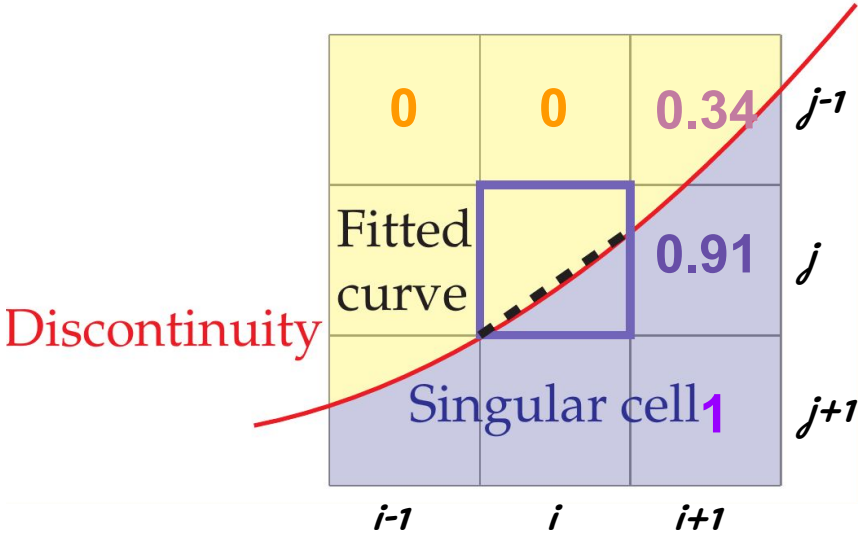
- Parametrized curves
  - Lines (**#**$\mu$=2)
  - Quadratics (**#**$\mu$=3)
  - Polynomials
  - Circles (**#**$\mu$=3)

As before, use the average values of the surrounding cells to find the parameters $\mu$ that best approximate the averages.

- **#**$\mu$ must be equal or inferior to the stencil width (ex, for a 3x3 stencil **#**$\mu$ can be at maximum 3)

We will work with ***f*** <u>piecewise constant</u>.

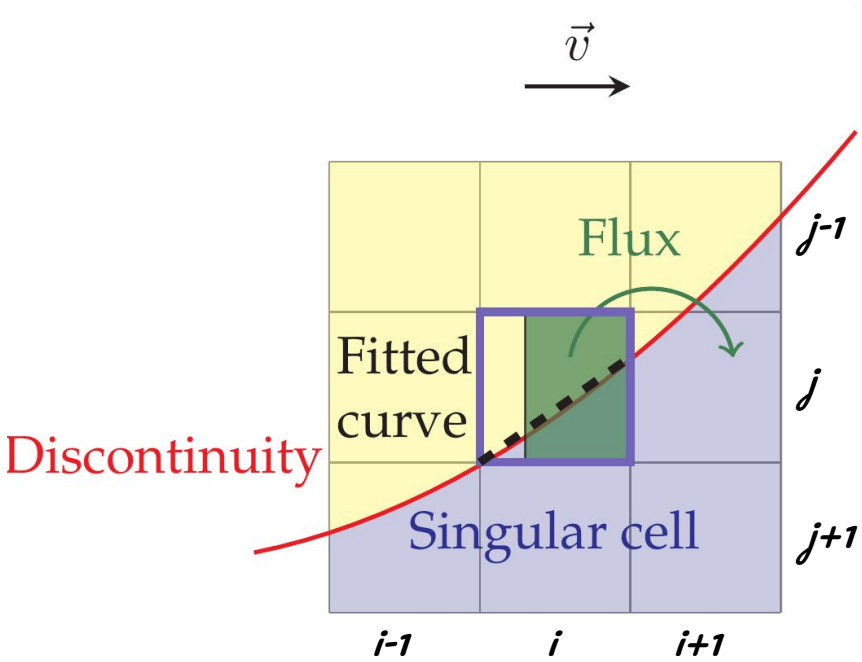
$$a_{ij} := \int\int_{i-\frac{1}{2},j-\frac{1}{2}}^{i+\frac{1}{2},j+\frac{1}{2}} f(x,y)dxdy$$



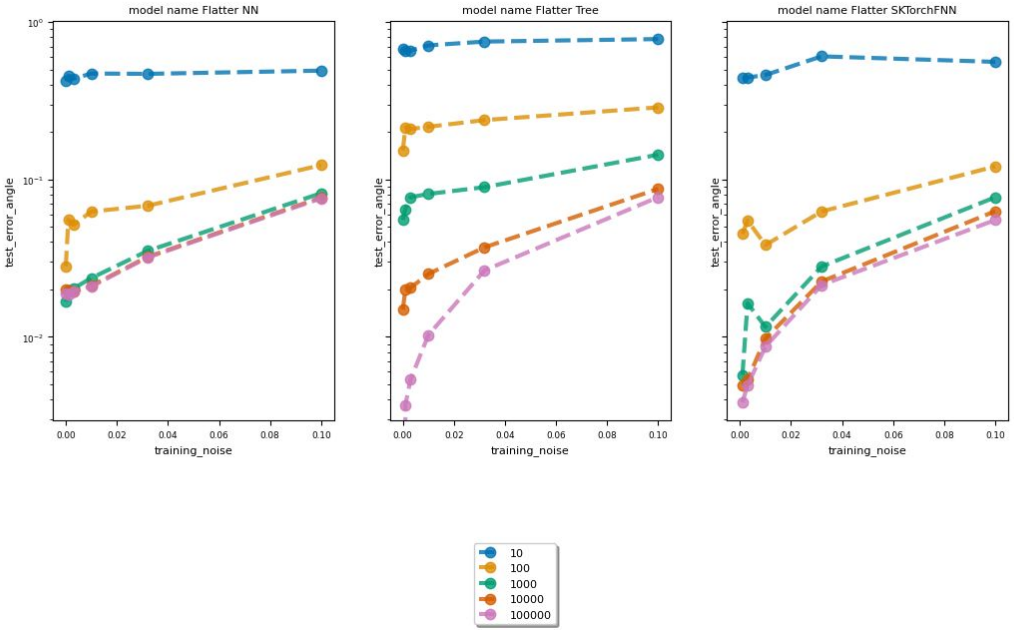
# Finite volume scheme 2D (or more)

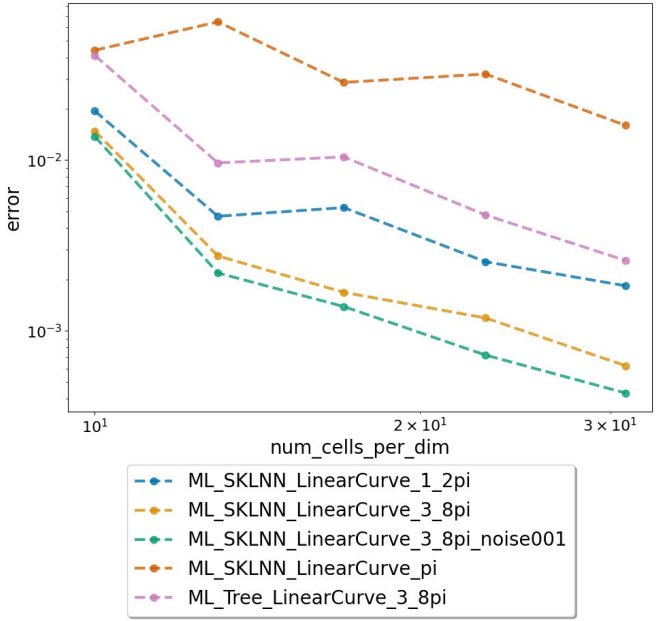
**<u>Difficulty</u>**: given the velocity and cell averages, calculate the numerical flux.

- By fitting a parametrized curve $\boldsymbol{C}(\mu)$ as a surrogate for calculating the fluxes.
- Directly learn the map cell averages to flux.

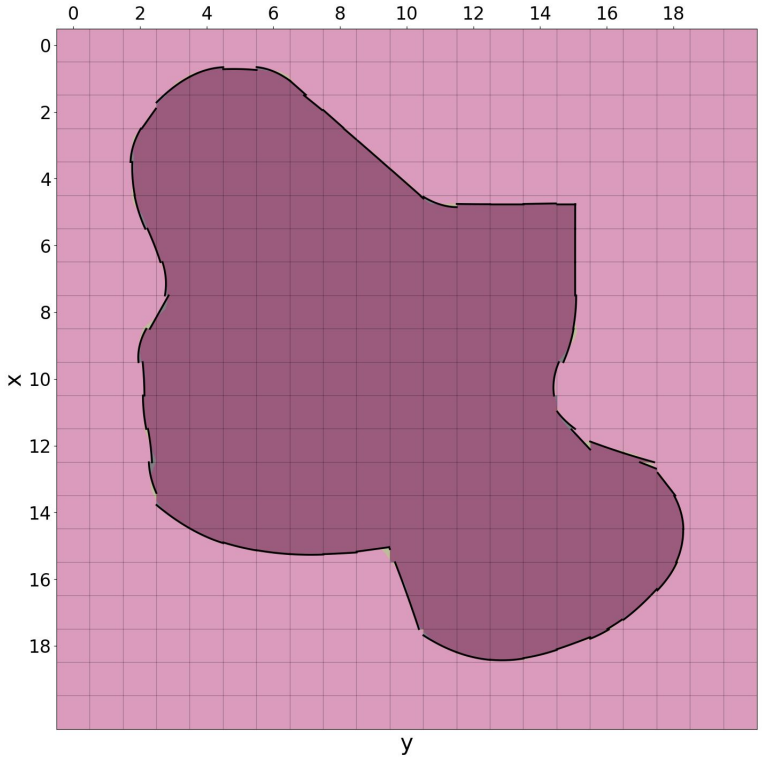


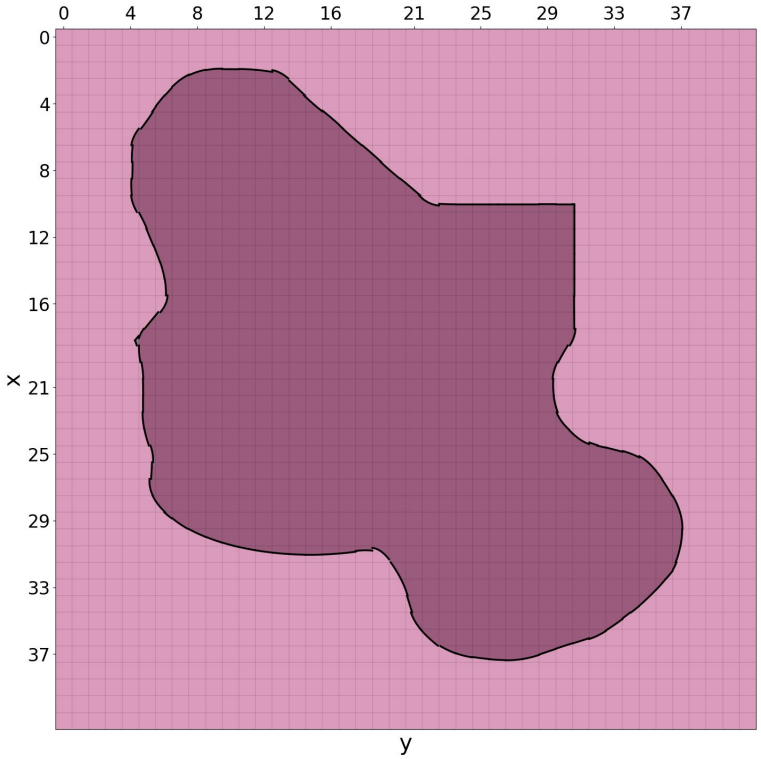
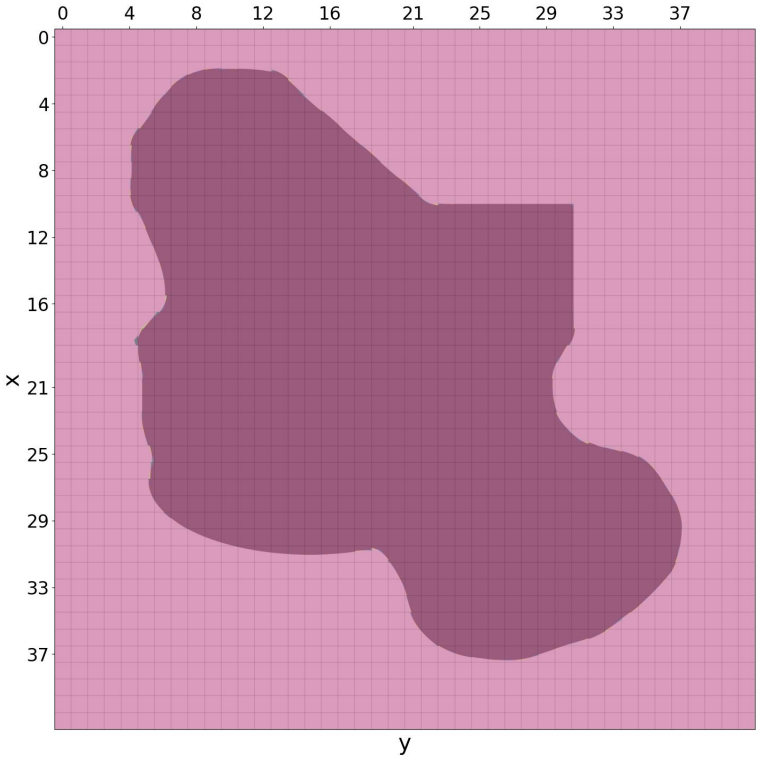
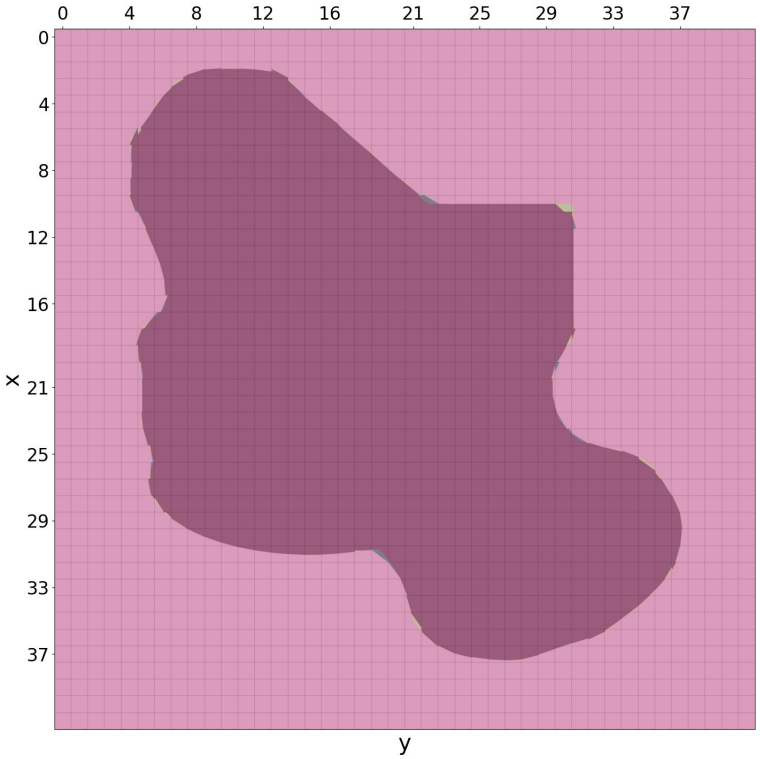
# Training with noise



# Vertex ENO-EA






# Vertex ENO-EA