Développement d’un formalisme Arbitraire Lagrangien Eulérien tridimensionnel en dynamique implicite. Application aux opérations de mise à forme.

Romain Boman – Ingénieur civil physicien

Le 6 mai 2010

Contents

Contents

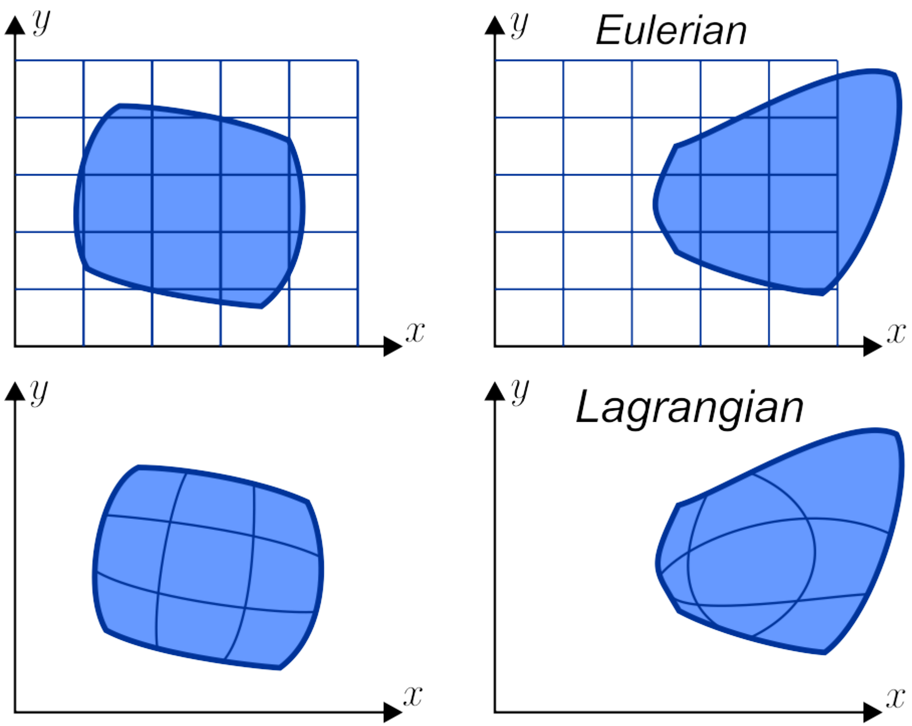
General framework & keywords

- **Continuum mechanics** – Nonlinear Solid Mechanics
  Highly nonlinear situation: large deformations, nonlinear materials, contact, thermomechanical coupling, inertia effects, etc.

- **Numerical simulation**
  - Finite Element Method (FEM)
  - Simulation of metal forming processes

- **Metafor** (home made software – http://metafor.ltas.ulg.ac.be/)

Kinematic description of the motion



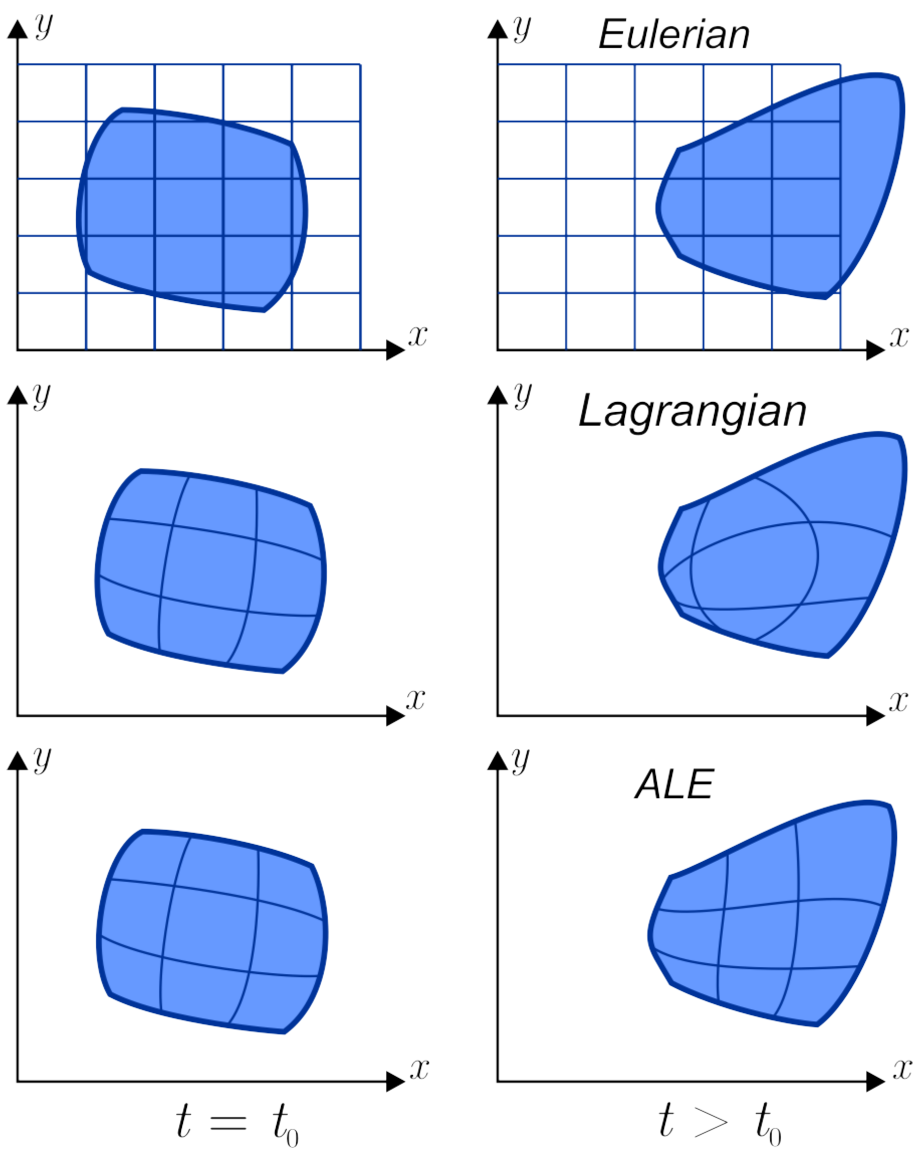
Eulerian formalism

- ✓ Undistorted mesh
- ✗ Free boundaries are difficult to track
- ✗ History-dependent materials are difficult to handle

Lagrangian formalism

- ✗ The mesh can be rapidly distorted
- ✓ Free boundaries are automatically computed
- ✓ History-dependent materials are easier to handle

Kinematic description of the motion



Arbitrary Lagrangian Eulerian (ALE) formalism

- Extension of both previous formalisms
- The mesh motion is uncoupled from material motion
- ALE can be crudely seen as a continuous remeshing procedure
- Mesh topology does not change
- Remapping of variables is faster than classical remeshing

Equations to be solved

Continuum mechanics (Lagrangian description)

- Mass

$$\left.\frac{\partial \rho}{\partial t}\right|_{\mathbf{X}} = -\rho \nabla \cdot \mathbf{v}$$

- Momentum

$$\rho \left.\frac{\partial \mathbf{v}}{\partial t}\right|_{\mathbf{X}} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b}$$

- Energy

$$\rho \left.\frac{\partial u}{\partial t}\right|_{\mathbf{X}} = \boldsymbol{\sigma} : \mathbf{D} + \rho r + \nabla \cdot \mathbf{q}$$

- Material

$$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\mathbf{X}} = \mathbf{H} : \mathbf{D} + \boldsymbol{\sigma}\mathbf{W}\boldsymbol{\sigma} - \mathbf{W}$$

coordinate systems

$\mathbf{X}$: material coordinates

$\mathbf{x}$: spatial coordinates

$\mathbf{v}$: material velocity

Equations to be solved

Continuum mechanics (ALE description)

- Mass
$$\left.\frac{\partial \rho}{\partial t}\right|_{\chi} + \mathbf{c}\cdot\nabla\rho = -\rho\nabla\cdot\mathbf{v}$$

- Momentum
$$\rho\left(\left.\frac{\partial \mathbf{v}}{\partial t}\right|_{\chi} + (\mathbf{c}\cdot\nabla)\mathbf{v}\right) = \nabla\cdot\boldsymbol{\sigma} + \rho\mathbf{b}$$

- Energy
$$\rho\left(\left.\frac{\partial u}{\partial t}\right|_{\chi} + \mathbf{c}\cdot\nabla u\right) = \boldsymbol{\sigma}:\mathbf{D} + \rho r + \nabla\cdot\mathbf{q}$$

- Material
$$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\chi} + (\mathbf{c}\boldsymbol{\sigma}\nabla) = \mathbf{H}:\mathbf{D} + \mathbf{W}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{W}$$

coordinate systems

$\mathbf{X}$: material coordinates

$\mathbf{x}$: spatial coordinates

χ : grid coordinates

$\mathbf{v}$: material velocity

$\mathbf{v}^*$: **arbitrary** grid velocity

with $\mathbf{c} = \mathbf{v} - \mathbf{v}^*$

(convective velocity)

ALE solution procedure

Fully coupled solution

- OK if mesh motion ($\mathbf{v}^*$) is known
- Otherwise, twice more mechanical unknowns ($\mathbf{v}$ and $\mathbf{v}^*$)

➔ too difficult and too slow for the simulation of forming processes

Operator split

ONE TIME INCREMENT

Lagrangian step: (classical)

Mesh sticks to the material ($\mathbf{v} = \mathbf{v}^*$, $\mathbf{c} = 0$)
Compute an equilibrated "Lagrangian configuration" at time $t + \Delta t$

Eulerian step: (when equilibrium is reached)

1. Define $\mathbf{c}$ (define a new mesh)

➔ see "mesh management" in this presentation

2. Data transfer from old mesh to the new one

➔ see "convective step" in this presentation

ALE Solution procedure

Operator split – Example: constitutive law

$$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\chi} + (\mathbf{c}\cdot\nabla)\boldsymbol{\sigma} = \mathbf{H}:\mathbf{D} + \mathbf{W}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{W}$$

unknowns

$\mathbf{x}(t)$, $\boldsymbol{\sigma}(t)$

Lagrangian step:

$$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\chi} + \underbrace{(\mathbf{c}\cdot\nabla)\boldsymbol{\sigma}}_{=0} = \mathbf{H}:\mathbf{D} + \mathbf{W}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{W}$$

$\mathbf{x}^L(t+\Delta t)$, $\boldsymbol{\sigma}^L(t+\Delta t)$

Lagrangian configuration

Eulerian step:

❶ *Define a new mesh (choose* $\mathbf{c}$*)*

$\mathbf{x}^E(t+\Delta t)$

❷ $$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\chi} + (\mathbf{c}\cdot\nabla)\boldsymbol{\sigma} = \underbrace{\mathbf{H}:\mathbf{D} + \mathbf{W}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{W}}_{=0}$$

$\boldsymbol{\sigma}^E(t+\Delta t)$

Eulerian configuration

Two families of ALE applications

1. Problems involving excessive mesh distortion

Benefits of ALE vs. Lagrangian models

- Helps to keep well-shaped elements despite large deformations
- Most often remeshing is completely avoided

Features of these ALE models

- ✓ Boundary nodes are (more or less) Lagrangian
- ✓ Small convective displacements are expected
- ✗ Complex smoothing methods for interior nodes

Example: axisymmetric forging

Lagrangian model *ALE model*

Two families of ALE applications

2. "Quasi-Eulerian" models

Example: rolling

ALE model (954 FEs)

Lagrangian model (1755 FEs)

Benefits of ALE vs. Lagrangian models

- The size of the model is decreased in the flow direction
- Loading is easier
- The element size may be optimised in the flow direction
- The contact regions do not change
- Less volume/contact elements → less CPU time

Two families of ALE applications

2. "Quasi-Eulerian" models

Features of these ALE models

- ✓ Mesh distortion is not a problem
- ✗ Material surface must be tracked
  - → Effective surface nodes management (2D and 3D)
- ✗ Material flows into/out of the meshed domain
  - → Special treatment of upstream/downstream boundary nodes
  - → Spurious fluxes should be as small as possible
- ✗ High convective displacements are expected
  - → A high order convection scheme is needed

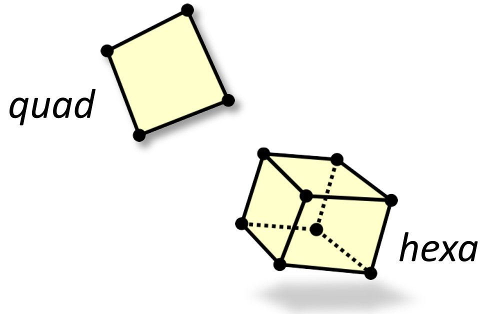
Main issues addressed in this work

Contents

Introduction

Mesh types

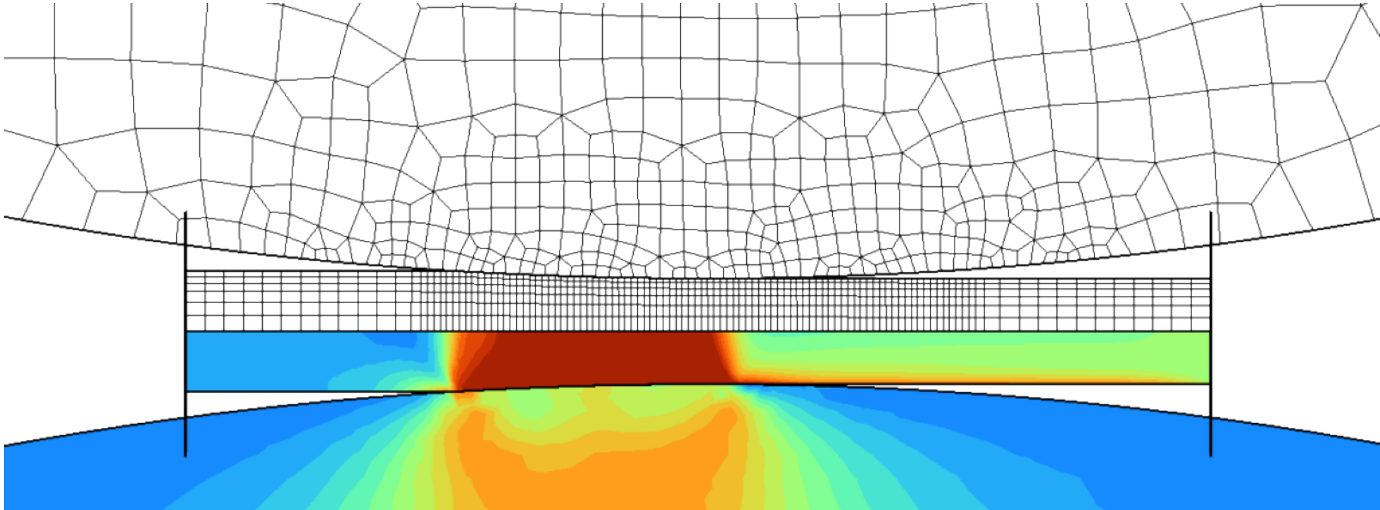
- Quadrangles or hexahedra only (Metafor)
- Sometimes unstructured
- Usually structured with local refinement for bending, contact ("graded mesh")



Example: modelling of cold rolling

Rolls:
unstructured

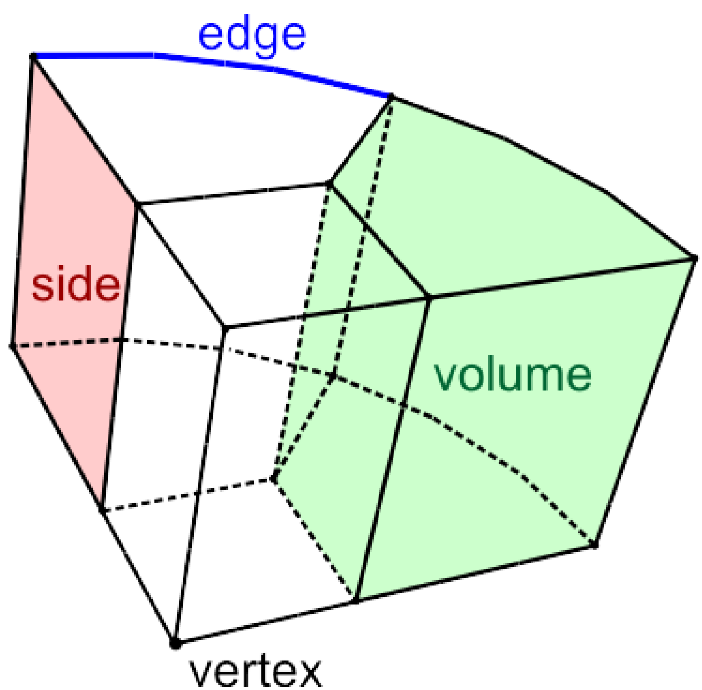
Strip:
graded & structured

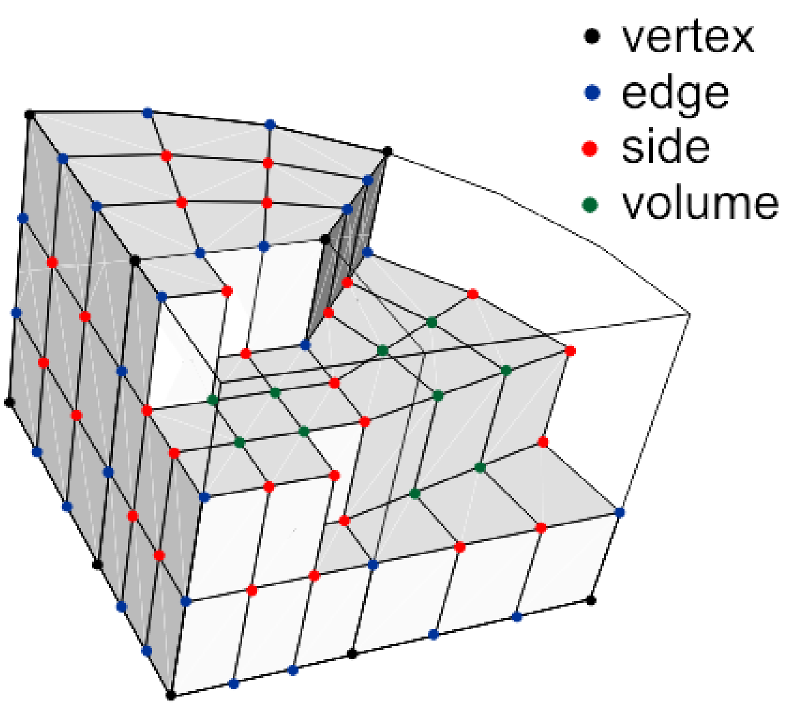


How to define mesh motion?

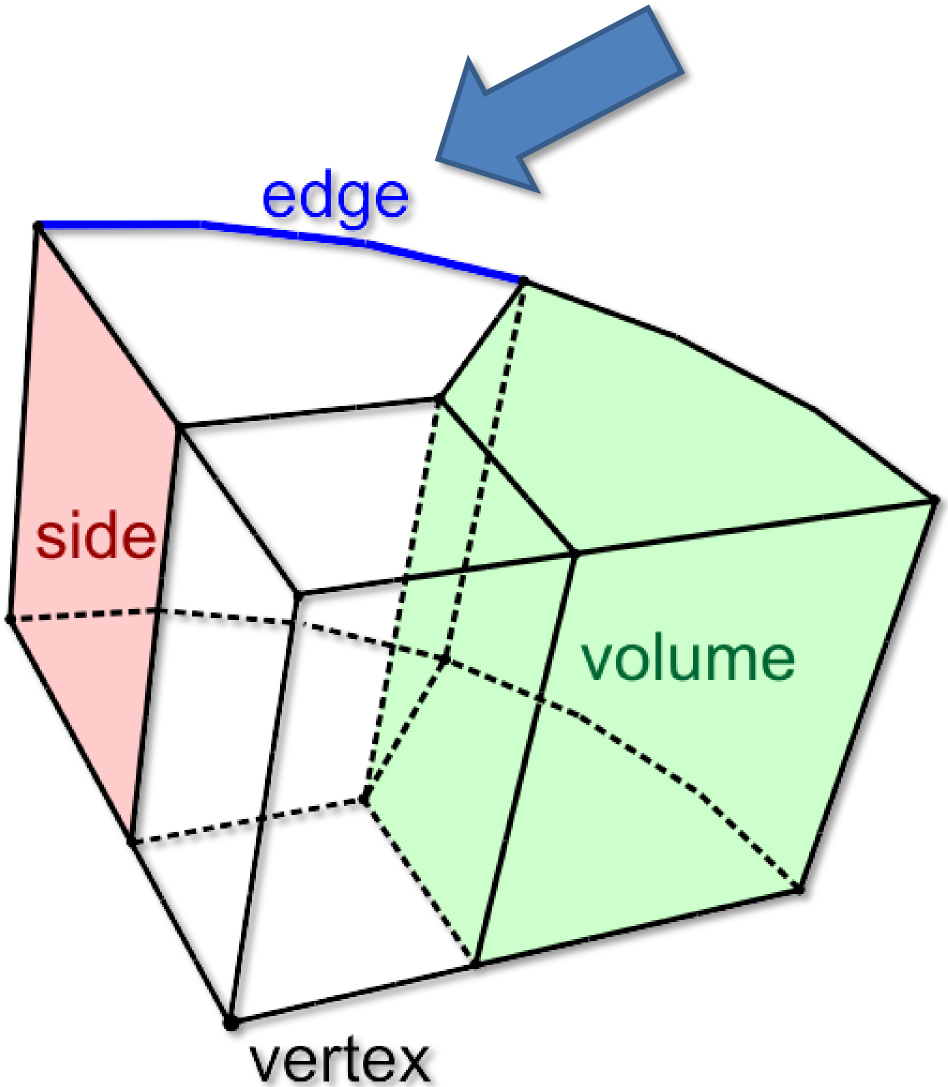
Manual procedure

- The methods highly depend on node position in the mesh.
  (vertex method ≠ edge method ≠ side method ≠ interior node method)
- They are applied to CAD entities (like loads and boundary conditions)





Nodes on sharp edges

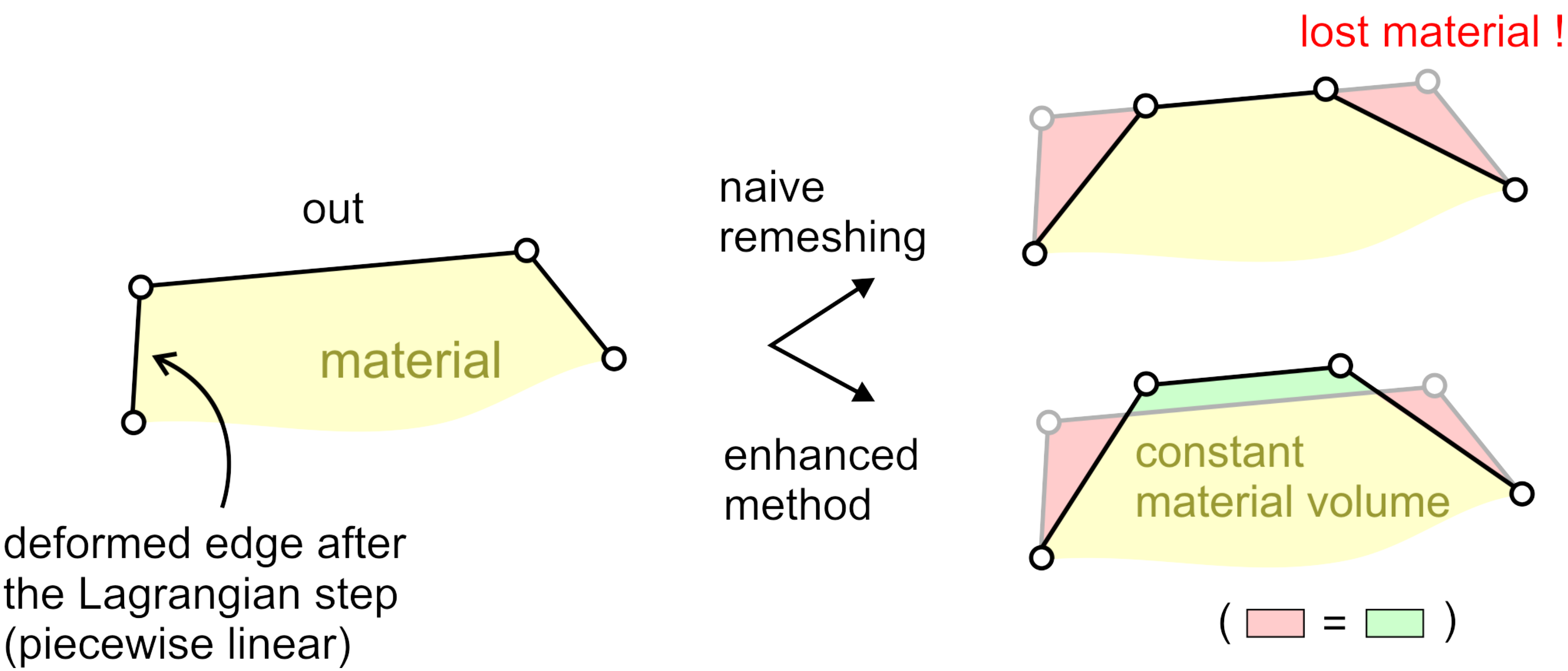


Nodes on sharp edges

Problem statement

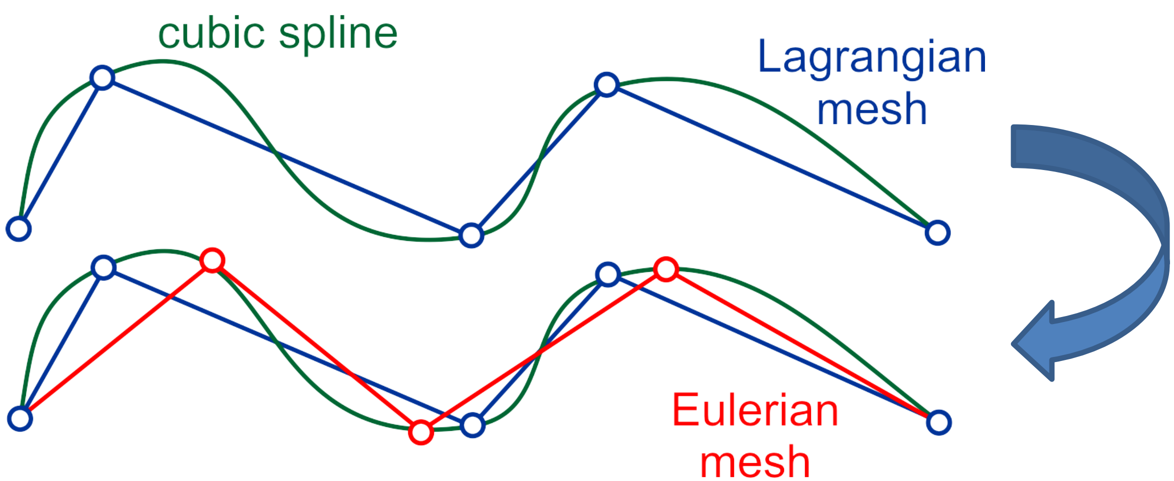
Remesh the deformed edge according to the initial node distribution

Difficulty: the edge is piecewise linear, material volume should not change



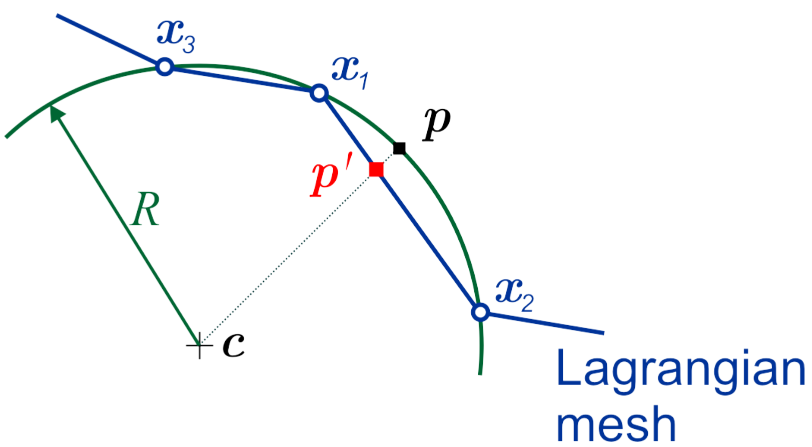
Nodes on sharp edges

Cubic spline method (Huétink)



Easy thanks to quasi-intrinsic parameterisation (Mc Conalogue)

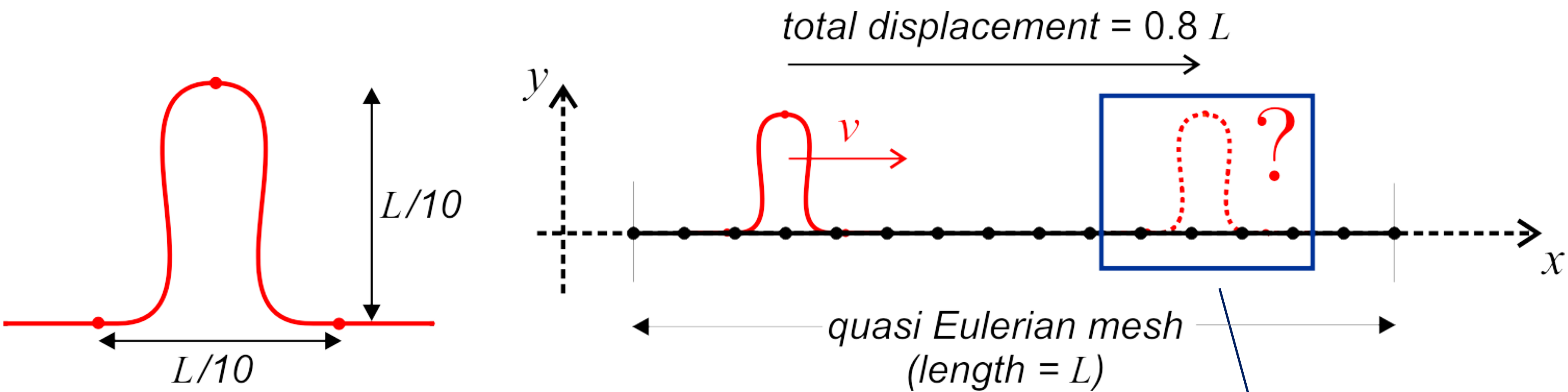
Arc method (Ponthot)



Same as the "naïve method"
except that
the node is projected on a circle built from the tree closest Lagrangian nodes

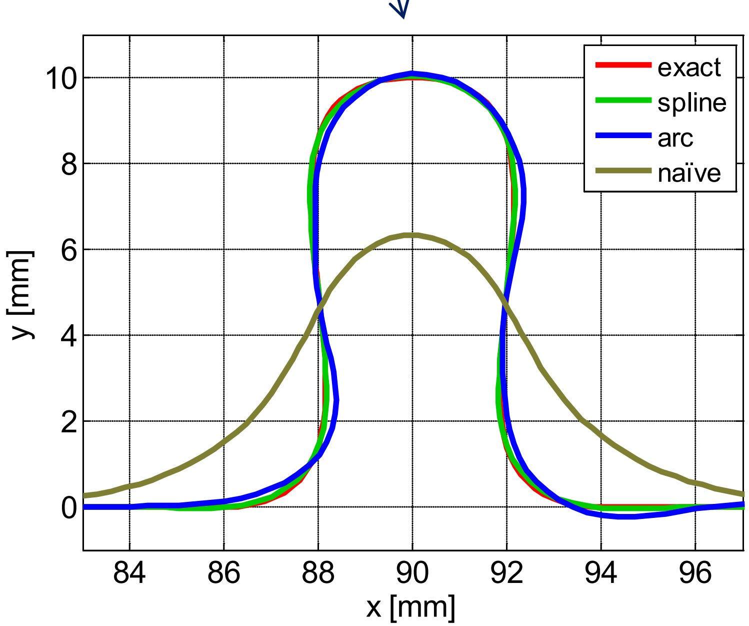
Nodes on sharp edges

Simple convection test



300 elements along x

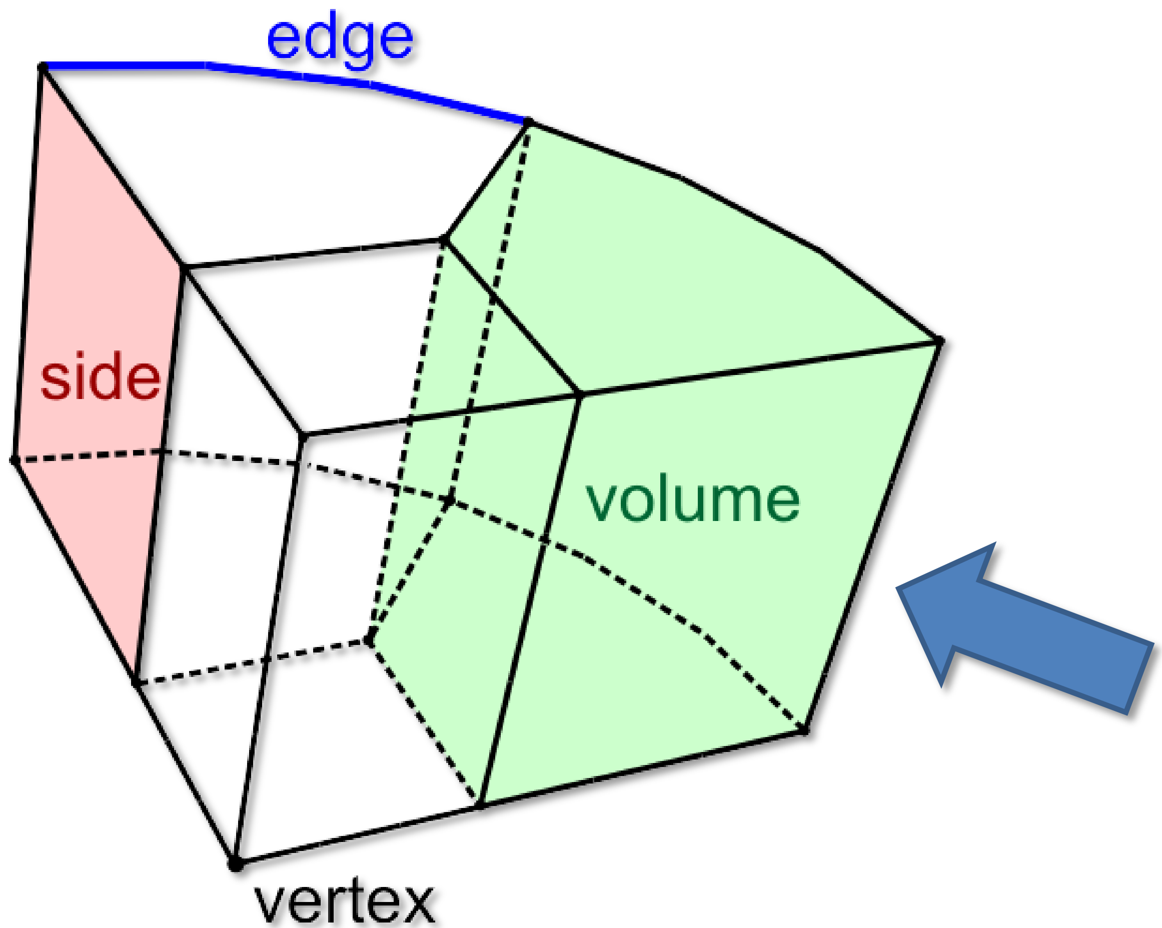
480 time steps (~1/2 element per step)



Observations:

- The "naïve method" should be avoided
- Arc and spline method are very close to the exact solution
- The spline method is slightly better

Interior (volume) nodes

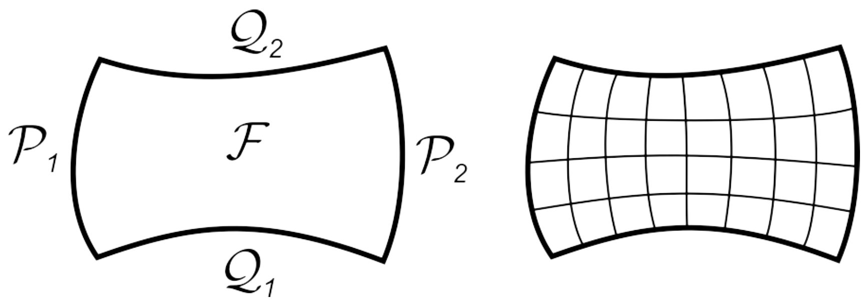


Interior nodes (2D/3D)

Direct methods

- Very fast but limited to structured meshes on quad-shaped domains

e.g.: Transfinite mapping (TM)



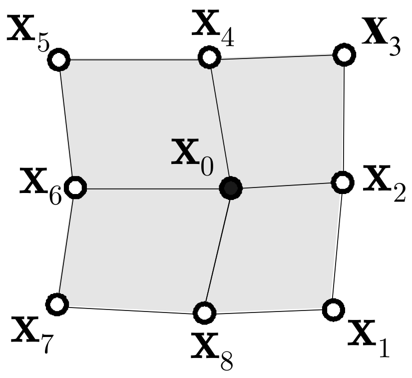
Iterative smoothing methods

- Slower than TM but more general
- Solver: Gauss Seidel + Successive OverRelaxation (SOR)

e.g.: weighted Laplacian

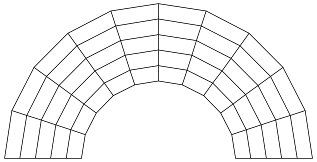
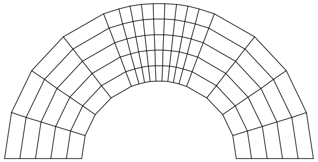
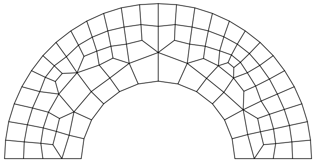
$$\mathbf{x}_0 = f(\mathbf{x}_i)$$

$$\mathbf{x}_0 = \frac{1}{\sum w_i} \sum w_i \, \mathbf{x}_i$$



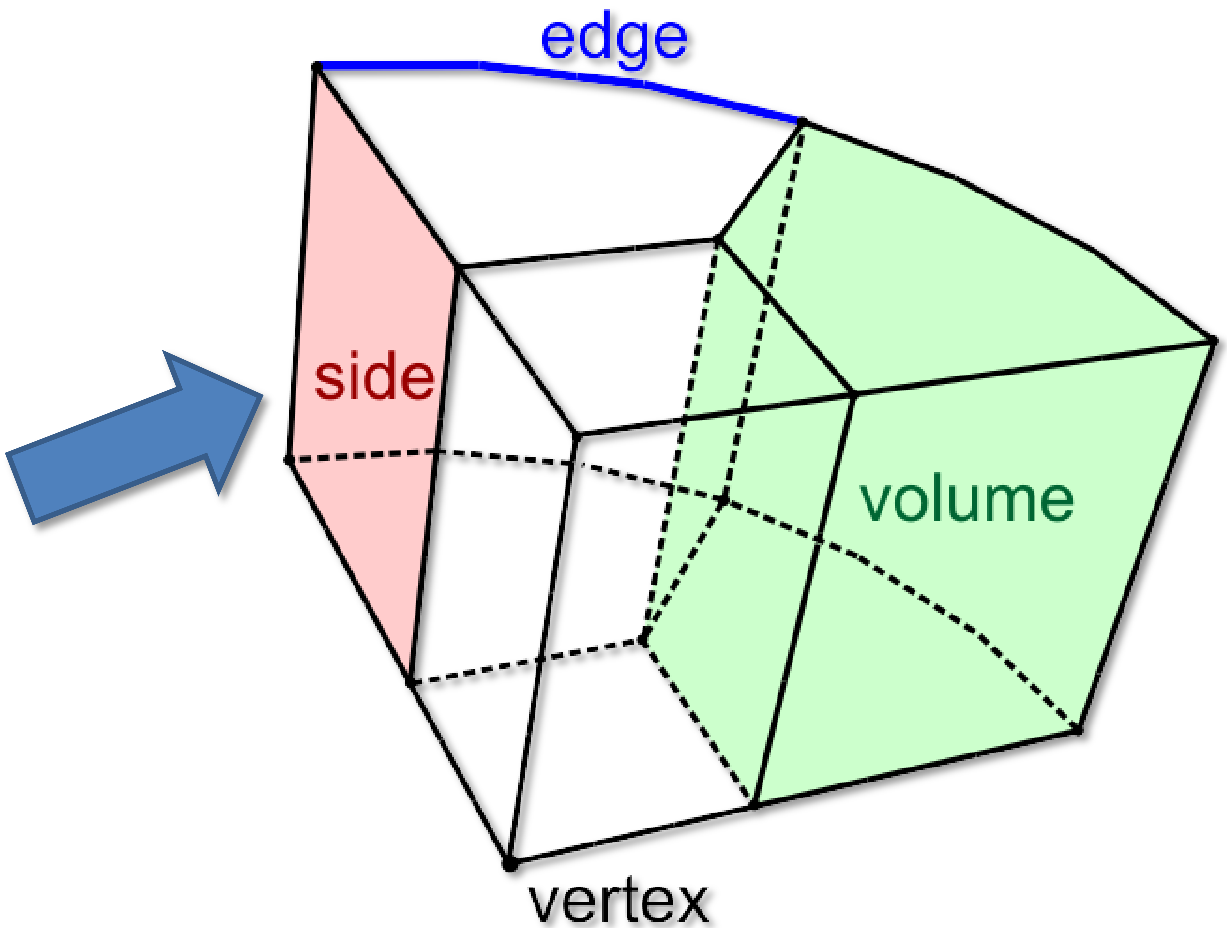
Interior nodes (2D/3D)

Available methods

	Structured constant	Structured graded	Unstructured
Transfinite Mapping	✓	✓	✗
Laplacian	✓	✗	✓
Weighted volumes	✓	✗	✓
Equipotential	✓	✗	✗
Isoparametric	✓	✓	✗
Giulinani	✓	✗	✓

Nodes on boundary surfaces in 3D



Nodes on boundary surfaces in 3D

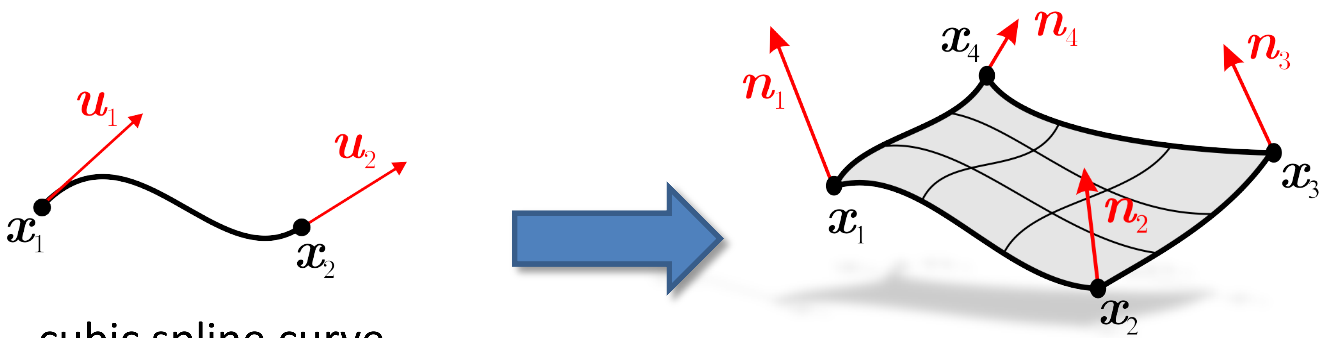
Problem statement

Remesh the deformed surface according to the initial sizes of the elements

Difficulty: the surface is piecewise bilinear, material volume should not change

Proposed solution:

Extend the previous method used for sharp edges

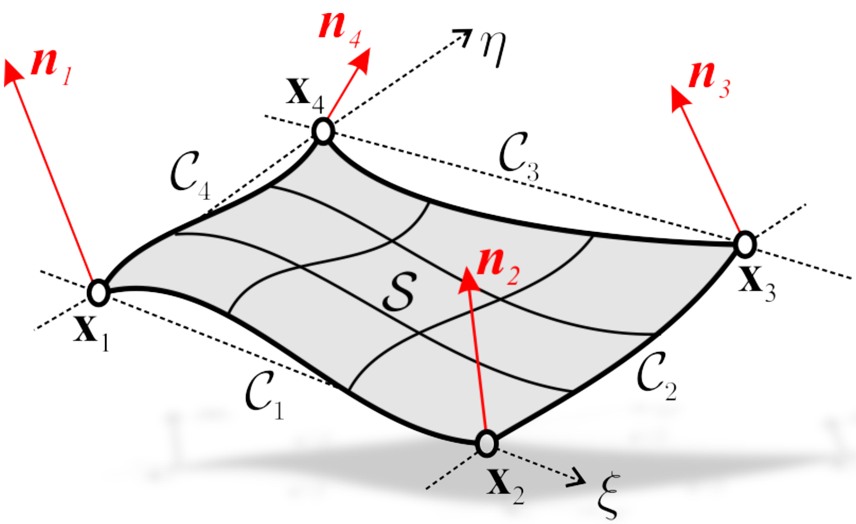


cubic spline curve

cubic spline surface

Nodes on boundary surfaces in 3D

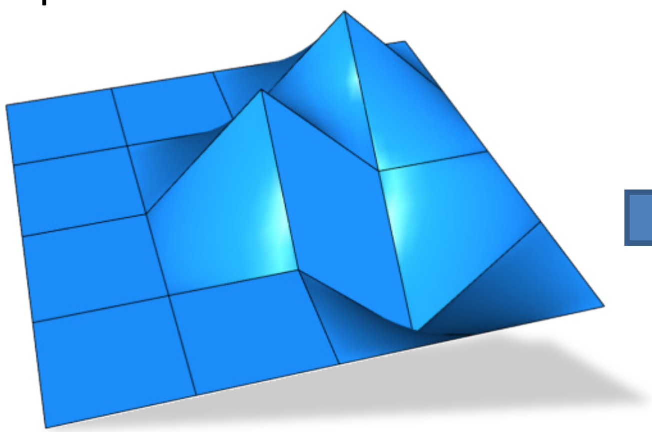
Cubic spline surface construction



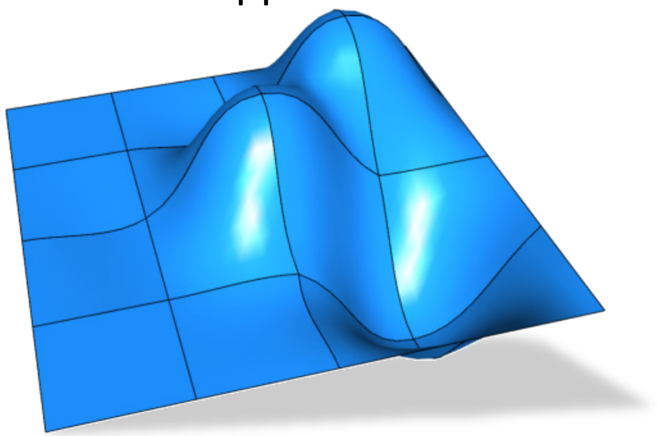
1. Normal continuity at nodes is obtained by averaging normals of neighbouring patches
2. Straight edges are converted to cubic segments
3. A Coons patch is built on the basis of these new edges

Example:

piecewise bilinear surface



cubic approximation



Nodes on boundary surfaces in 3D

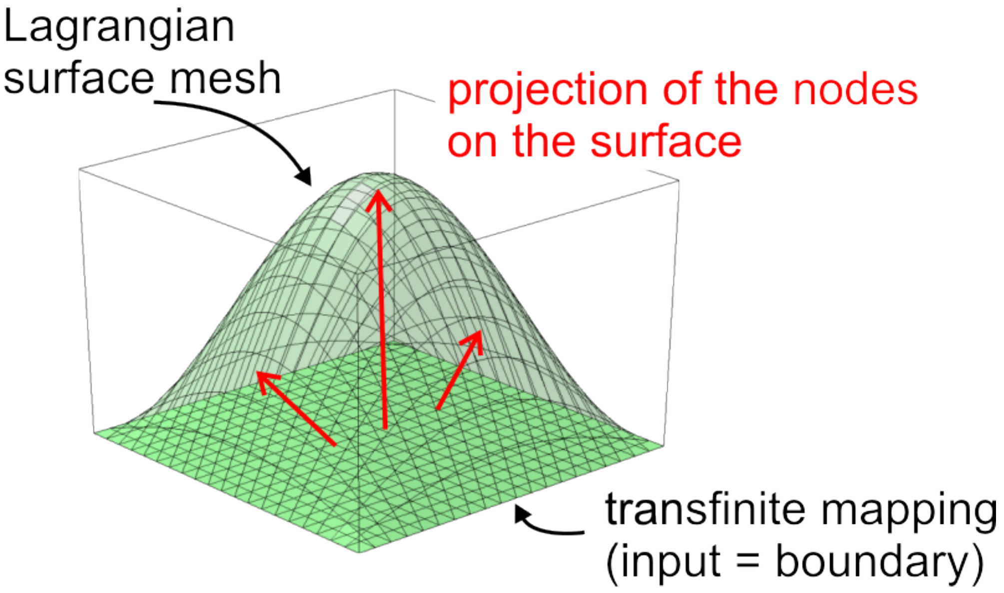
Direct method

Proposed method

- Build/Update the spline surface
- Generate a new mesh using the transfinite mapping method
- Project each node to the spline surface

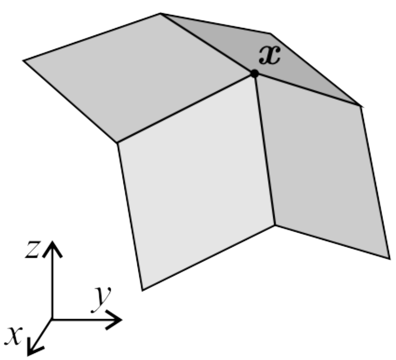
This method may fail if...

- the surface mesh is unstructured
- the deformed Lagrangian mesh is far from the bilinear interpolation of the boundaries of the surface

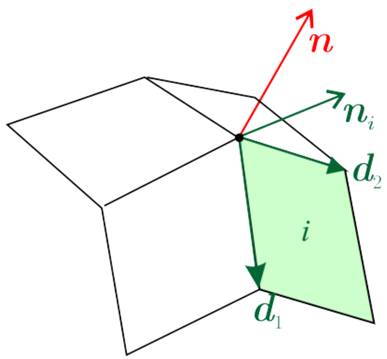


Nodes on boundary surfaces in 3D

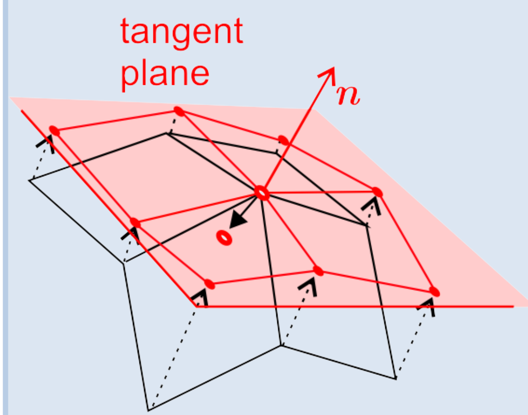
Iterative method



Lagrangian mesh

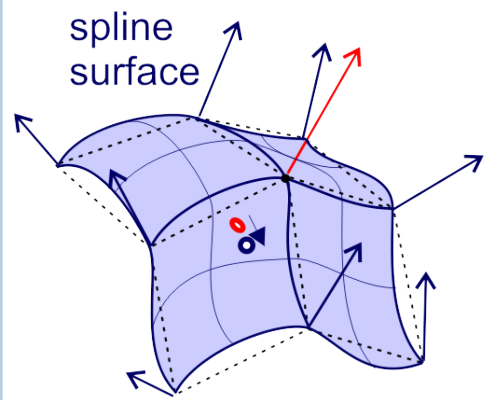


Normal averaging



Any 2D smoothing method in the tangent plane

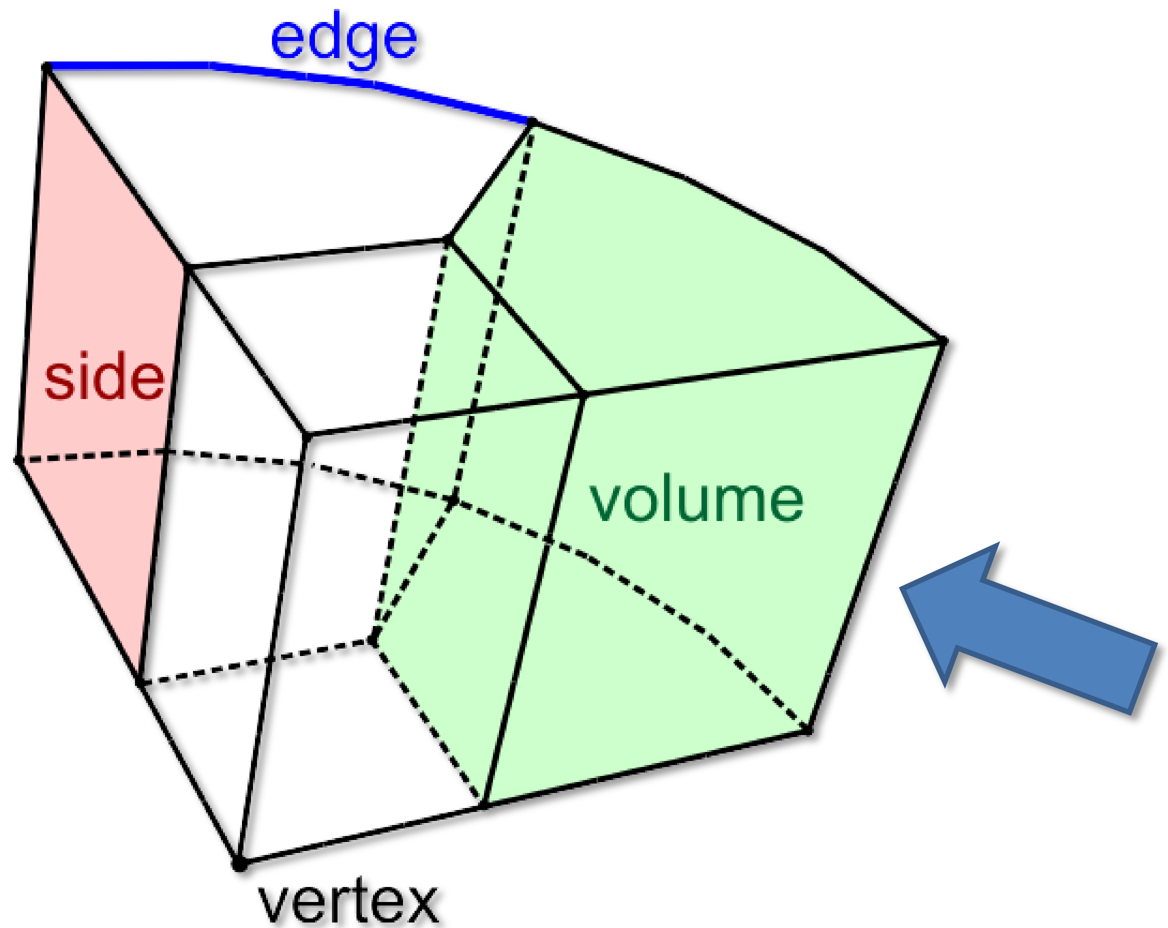
New



Robust and effective projection on the cubic approximation

Iterations

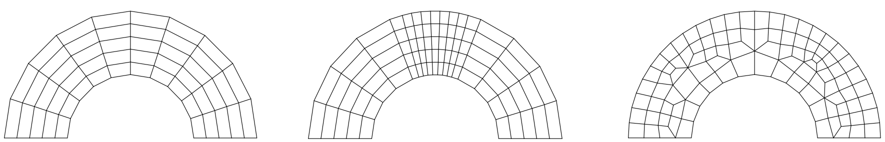
Back to interior nodes methods



Back to Interior nodes methods

Common issue in quasi Eulerian models

- Graded surface mesh
- Complex shape ➔ iterative method required



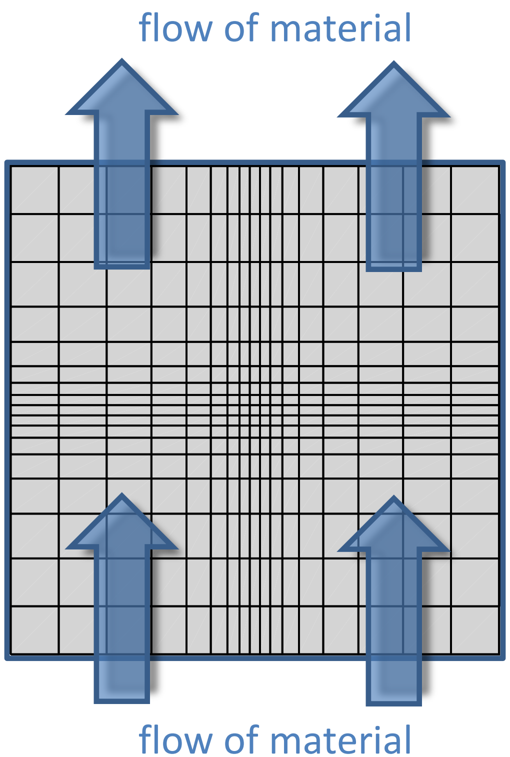
	Structured – constant	Structured – graded	Unstructured
~~Transfinite Mapping~~	~~✓~~	~~✓~~	~~✗~~
Laplacian	✓	✗	✓
Weighted volumes	✓	✗	✓
Equipotential	✓	✗	✗
Isoparametric	✓	✓	✗
Giulinani	✓	✗	✓

only one method available!
➔ "Isoparametric smoothing"

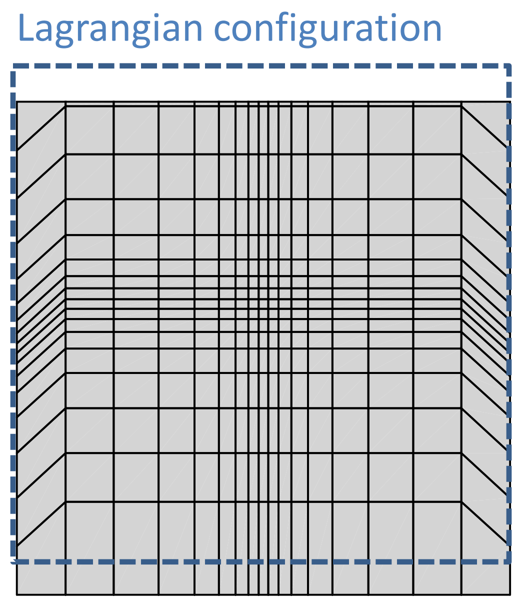
Back to Interior nodes methods

Isoparametric smoothing is VERY slow!

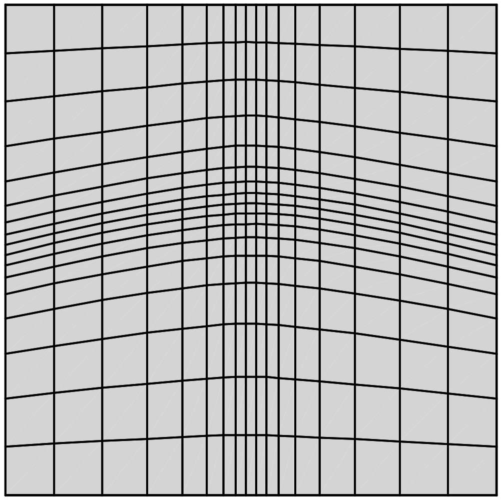
Simple 2D example



Reference configuration



Eulerian remeshing of boundaries

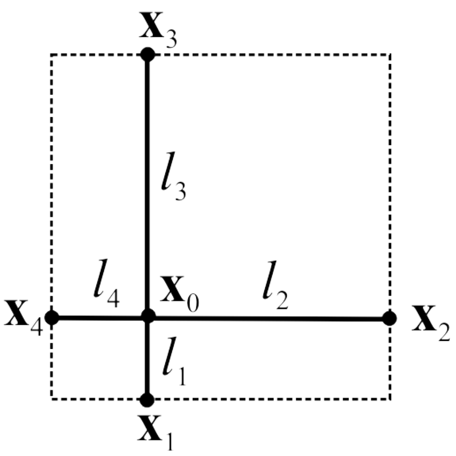


Isoparametric smoothing (after 200 iterations)

Back to Interior nodes methods

New smoothing method

Based on weighted Laplacian: $$\mathbf{x}_0 = \frac{1}{\sum w_i} \sum w_i \, \mathbf{x}_i$$

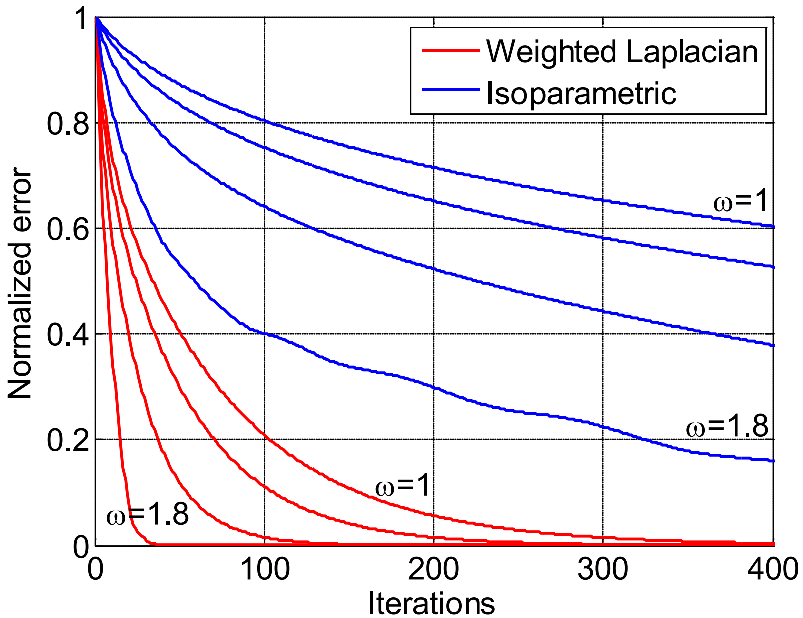


Find weights w_i that preserve the n edge length (l_i) ratios

$$\begin{cases} (l_1 + l_3)\,\mathbf{x}_0 = l_3\,\mathbf{x}_1 + l_1\,\mathbf{x}_3 \\ (l_2 + l_4)\,\mathbf{x}_0 = l_4\,\mathbf{x}_2 + l_2\,\mathbf{x}_4 \end{cases}$$

$$\Rightarrow \quad \boxed{w_i = \prod_{k=1}^{n} l_k}$$

Convergence speed on previous test

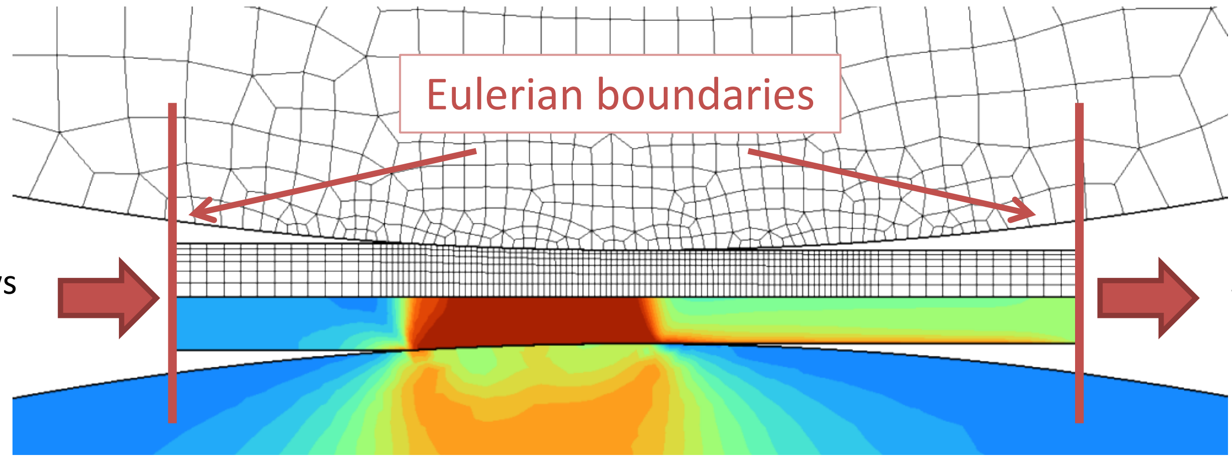


- Simple and efficient
- Faster than the isoparametric method
- Easier to implement

Eulerian boundaries

The mesh must remain inside the ALE domain

the material flows into the domain



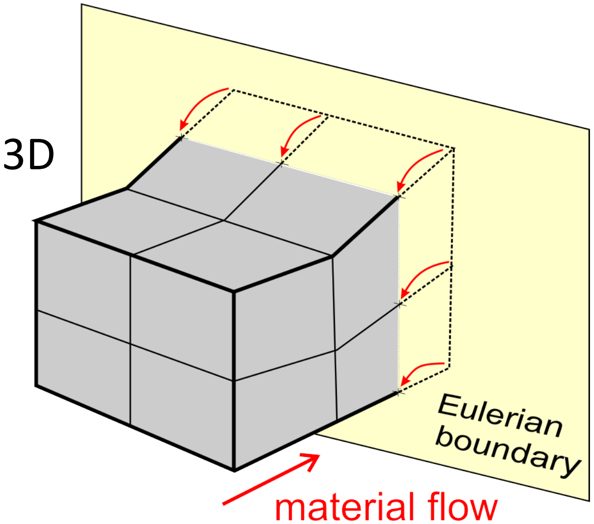
the material flows out of the domain

Orthogonal projection must be avoided!

Proposed solution

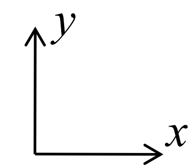
- The mesh is cut by a boundary surface (usually a plane)
- Additional smoothing may be added in order to improve the quality of the section mesh

3D

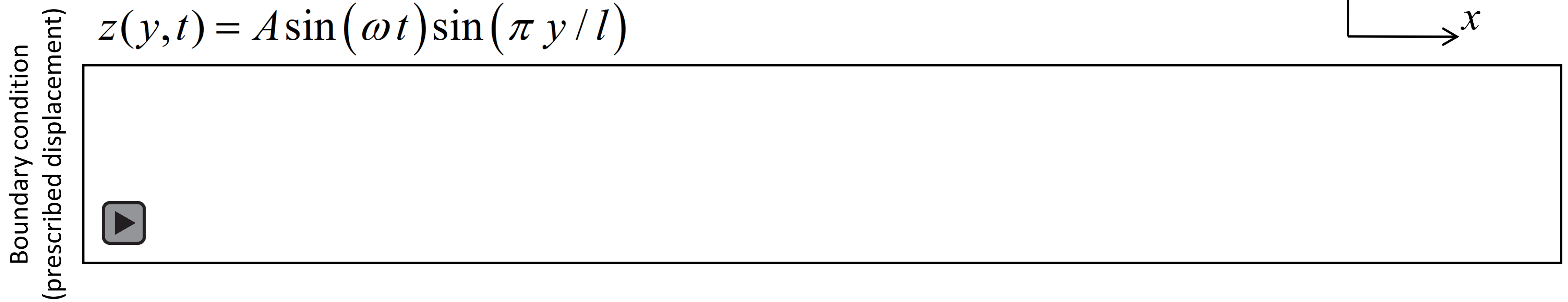


Numerical example

Sinusoïd convection on an unstructured quad mesh



$$z(y,t) = A\sin(\omega t)\sin(\pi y / l)$$

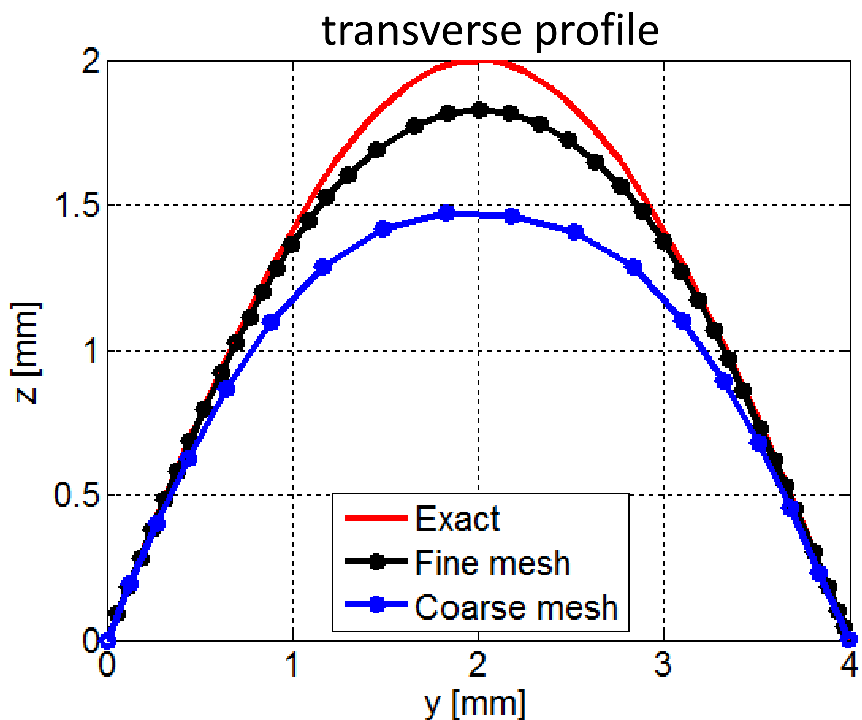


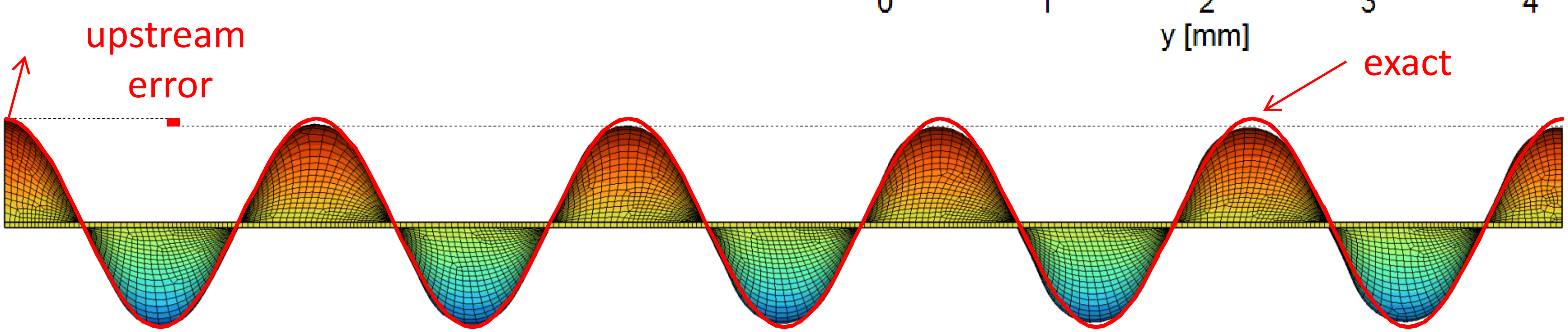
Numerical example

Sinusoïd convection on an unstructured quad mesh

Observations

- The shape is well preserved
- A better accuracy is obviously obtained with a finer mesh
- A small upstream error is due to a bad normal approximation on the upstream boundary





Numerical example

Sinusoïd convection on an unstructured quad mesh

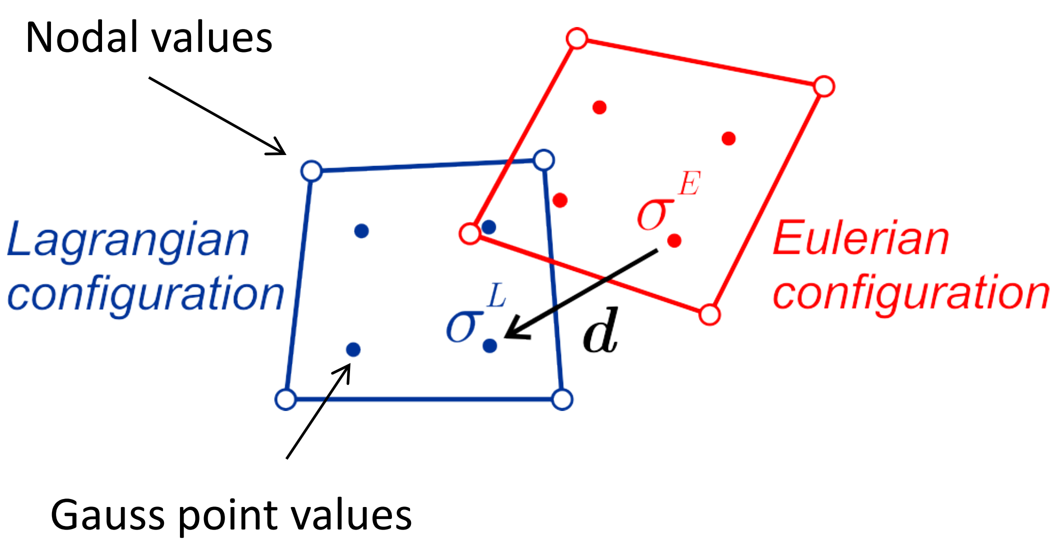
Downstream view of the Eulerian boundary

Contents

Introduction

Problem statement

- The Eulerian configuration (the “new” mesh) has been computed
- Data must be transferred from the Lagrangian to the Eulerian configuration



Two kinds of data
(always denoted by σ):

- Nodal (continuous)
- Gauss points (discrete)

$\mathbf{d}$: convective displacement
$\mathbf{c}$: convective velocity

$$\mathbf{d} = \mathbf{c}\,\Delta t$$

Fields to be transfered

From 6 to 38 scalar convection problems in 3D

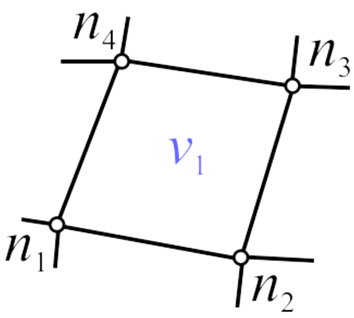
	Gauss Points (GP)		*Nodes*	
Constitutive law	Cauchy stresses $\boldsymbol{\sigma}$ ($\boldsymbol{s}$ and p) Equiv. plastic strain ($\bar{\varepsilon}^p$) Backstresses ($\boldsymbol{\alpha}$)	6 +1 +6	-	
Inertia effects	Density (ρ)	+1	Velocity ($\boldsymbol{v}$), Acceleration ($\boldsymbol{a}$)	+3 +3
Thermal effects	-		Temperature (T) and its derivative ($\dot{T}$)	+1 +1
EAS Elements	Additional EAS stresses ($\tilde{\boldsymbol{\sigma}}$)	+9	-	
Postprocessing	Deformation gradient tensor $\mathbf{F}$	+6	-	
TOTAL		***30***		***8***

Two kinds of methods

"finite elements" based methods

✓ Easier in the frame of a FEM code (same data structure)

✗ Artificial diffusion is difficult to handle → oscillations

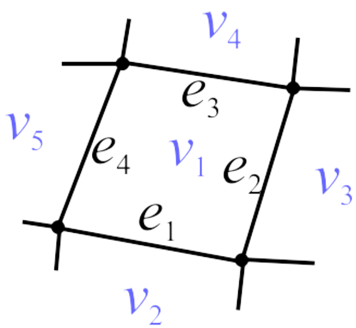


$v_1 = (n_1, n_2, n_3, n_4)$

"finite volumes" based methods ⇐ Our choice

✗ The finite volume mesh requires a different and separate data structure

✓ Efficient convection schemes used in Computational Fluid Dynamics (CFD)



$v_1 = (e_1, e_2, e_3, e_4)$

$e_1 = (v_1, v_2)$
$e_2 = (v_1, v_3)$
$e_3 = (v_1, v_4)$
$e_4 = (v_1, v_5)$

Introduction

Two mathematically equivalent points of view

Convection problem

$$\left.\frac{\partial \sigma}{\partial t}\right|_{\chi} + \mathbf{c} \cdot \nabla \sigma = 0$$

σ is **any** GP or nodal field (not necessarily a stress component)

Highlight on convective fluxes
➔Godunov's scheme

Interpolation problem

$$\left.\frac{\partial \sigma}{\partial t}\right|_{x} = 0$$

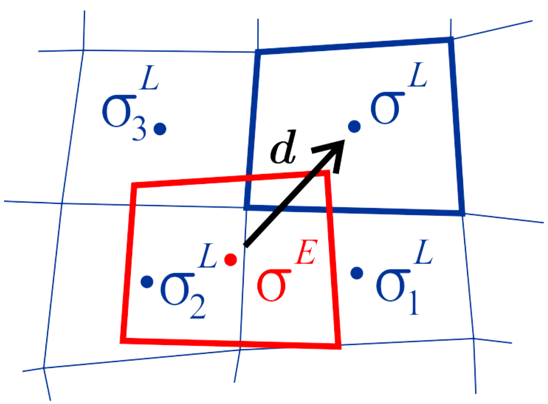
Highlight on field invariance
➔Projection scheme

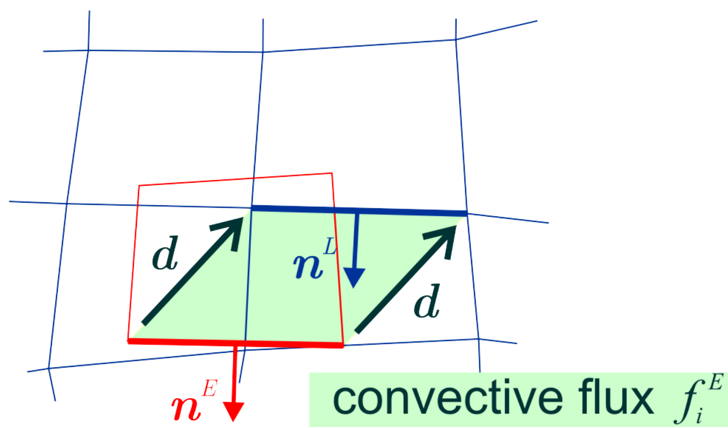
Godunov's scheme

Huerta, Casadei, Donéa 1992

Hypotheses

- One Gauss Point per finite element (explicit dynamics)
- One finite element ⇔ one finite volume (one cell)
- σ is constant over each cell ("constant reconstruction")





$$f_i = \int \mathbf{c}\cdot\mathbf{n}\, dt$$

$$f_i^E = \mathbf{d}\cdot\mathbf{n}^E$$

$$\left.\frac{\partial \sigma}{\partial t}\right|_\chi + \mathbf{c}\cdot\nabla\sigma = 0$$

$$\sigma^E = \sigma^L + \frac{\Delta t}{2V^E}\sum_{i=1}^{N} f_i^E \left(\sigma_i^L - \sigma^L\right)\left(1 - \alpha\,\mathrm{sign}\, f_i^E\right)$$

space discretisation: Finite Volume Method
time discretisation: Backward Euler explicit scheme

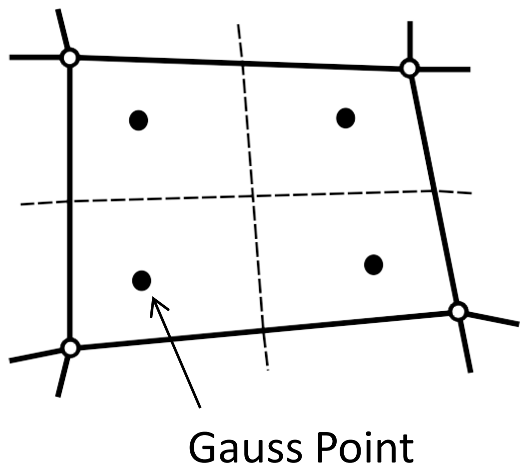
α: upwind factor ($\alpha \le 1$)

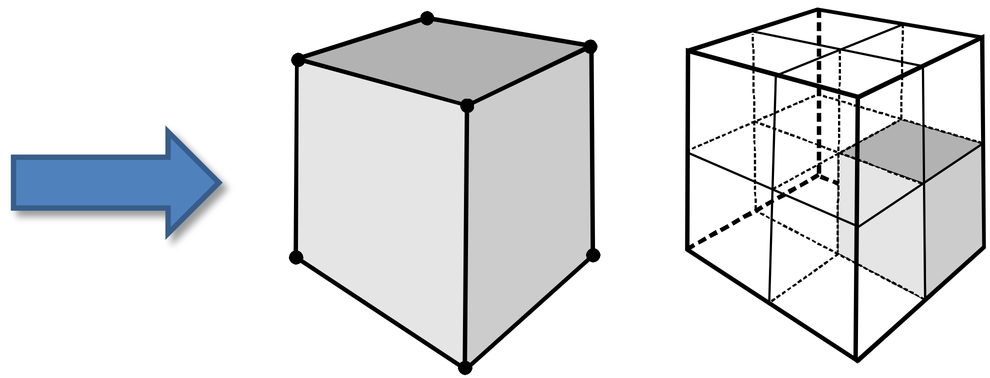
Godunov’s scheme

Extension to more than 1 Gauss Point per element

(for implicit problems)

in 2D: Huerta et al.	in 3D: this work





“Engineering-like approach”:
Each FE is split in 4 cells that surround each GP

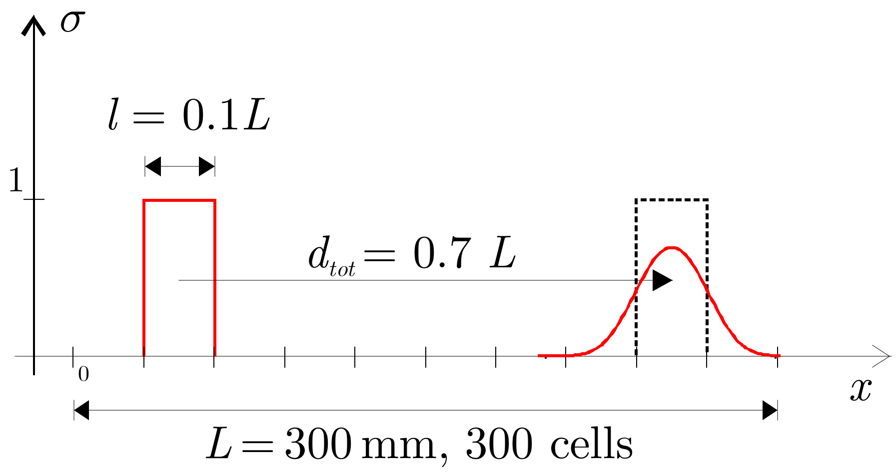
Main issues in 3D

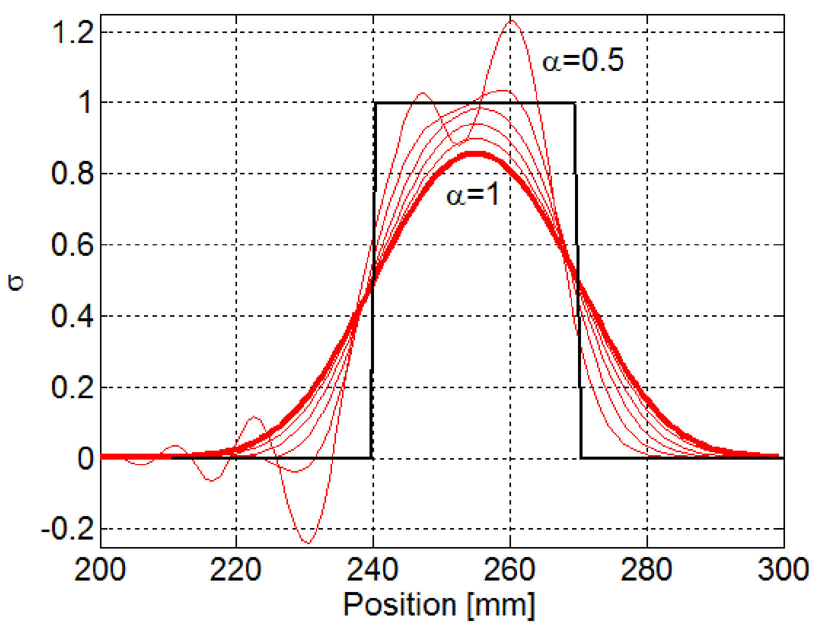
- Automatic generation of the auxiliary mesh from 3D unstructured FE meshes
- Data storage management

➔ Easily solved using Object Oriented Programming (OOP)

Godunov's scheme

Simple 1D convection test





This scheme is…

- conditionnaly stable : $0 \leq \alpha \leq C$ (C = Courant number) ➔ sub-stepping
- monotonicity preserving if $\alpha = 1$ ➔ always $\alpha = 1$
- first order accurate (too diffusive for quasi Eulerian problems)

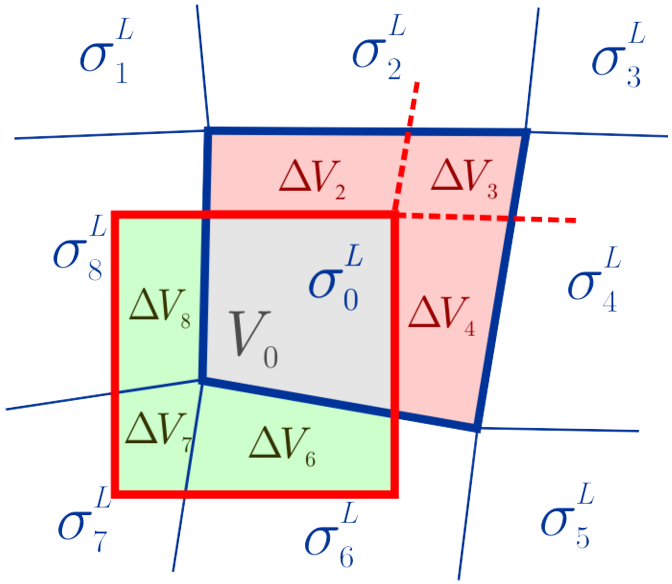
➔ A higher order scheme is required!

Projection scheme

Benson 1989

Hypotheses:

- Only one GP per finite element (explicit dynamics)
- 1 finite element = 1 cell



Error norm minimization:

$$\int_V \sigma^E \, dV = \int_V \sigma^L \, dV$$

Example: if σ is constant on each cell:

$$V^E \sigma_0^E = V_0 \sigma_0^L + \Delta V_6 \sigma_6^L + \Delta V_7 \sigma_7^L + \Delta V_8 \sigma_8^L$$

$$= V^L \sigma_0^L - \underbrace{(\Delta V_2 + \Delta V_3 + \Delta V_4)\sigma_0^L}_{\text{inward flux}} + \underbrace{\Delta V_6 \sigma_6^L + \Delta V_7 \sigma_7^L + \Delta V_8 \sigma_8^L}_{\text{outward flux}}$$

inward flux

outward flux

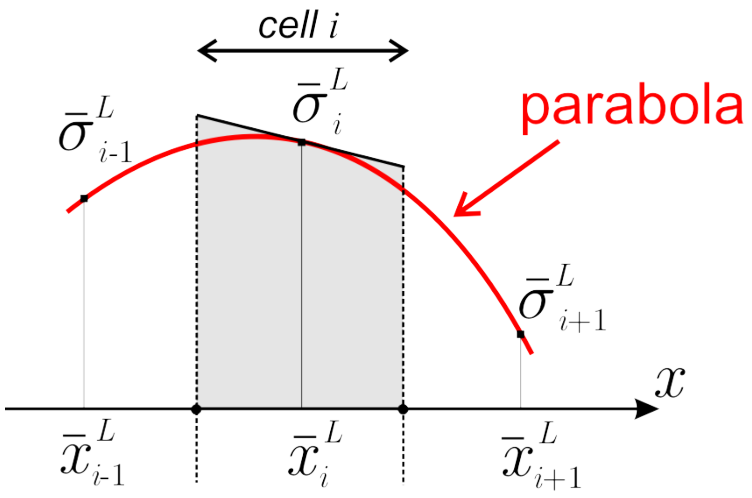
Projection scheme

Extension to second order accuracy : Linear reconstruction

σ is no more constant: $\sigma_i^L(\mathbf{x}) = \overline{\sigma}_i^L + \nabla\sigma_i^L \cdot (\mathbf{x} - \overline{\mathbf{x}})$

mean value (pointing to $\overline{\sigma}_i^L$); gravity centre (pointing to $\overline{\mathbf{x}}$)

The gradient $\nabla\sigma_i^L$ is computed from the values of neighbouring cells:

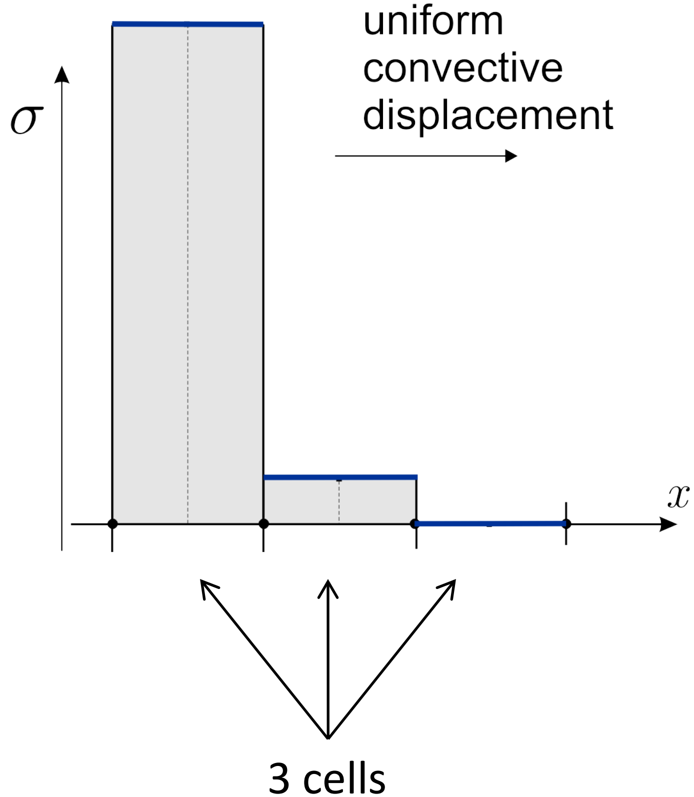


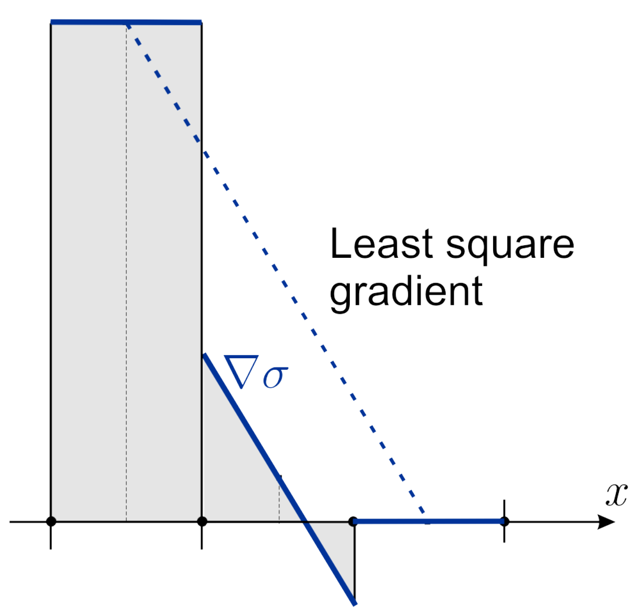
increasing complexity ↓

- Van Leer : MUSCL scheme (1D only)
- Benson (2D = 2x 1D problem)
- This work (2D-3D):
  CFD ➔ Least squares

Projection scheme

Monotonicity and flux limiters – 1D example





Linear reconstruction

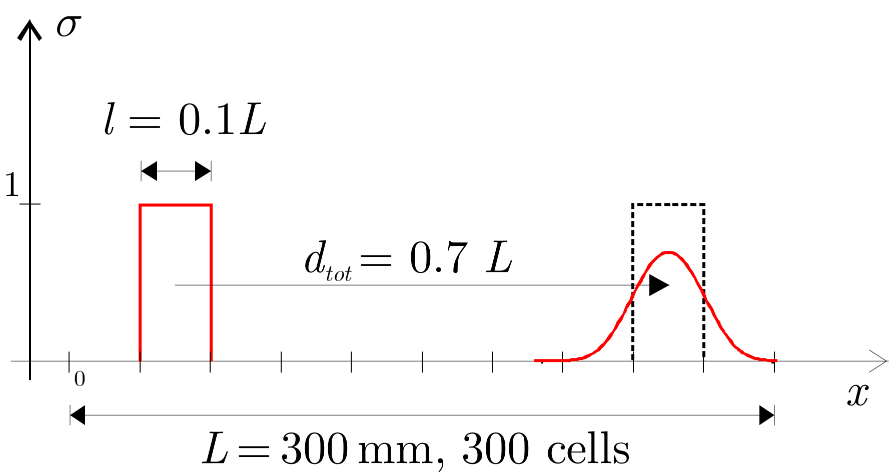
Projection scheme

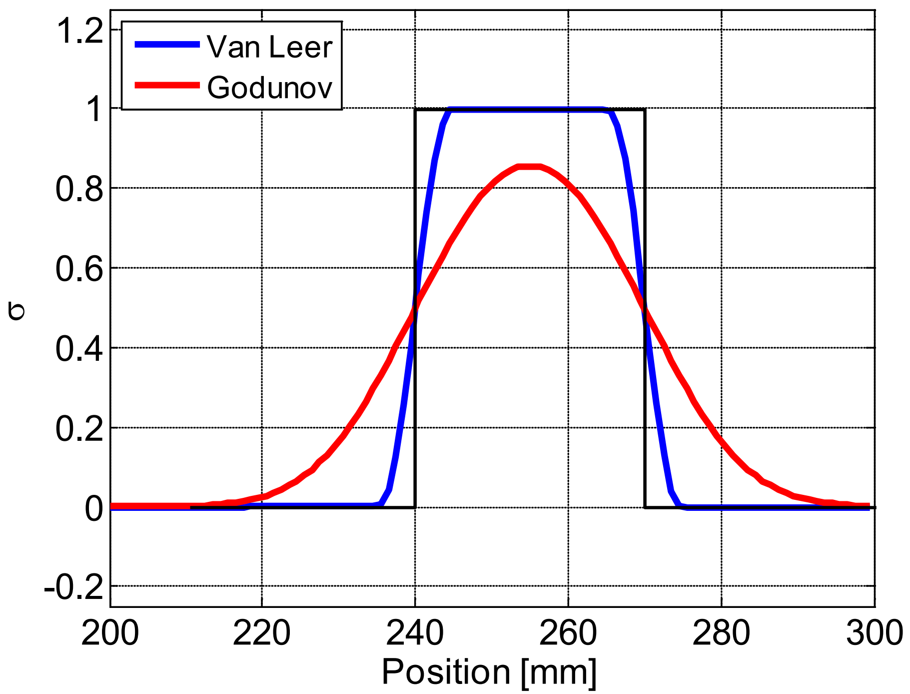
Monotonicity and flux limiters – 1D example

Second order accuracy is (locally) lost BUT oscillations are avoided

Projection scheme

Back to the simple 1D convection test





Steeper gradients are preserved without oscillations!

Projection scheme

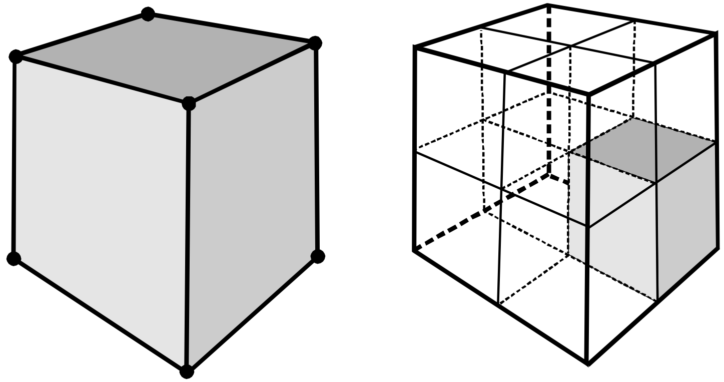
Extension to more than 1 Gauss Point per finite element

(for implicit problems)

Combine...

- Huerta's idea:
  "Split each finite element into cells surrounding each GP"
- Benson's idea:
  "Use a projection method with a linear reconstruction of σ"

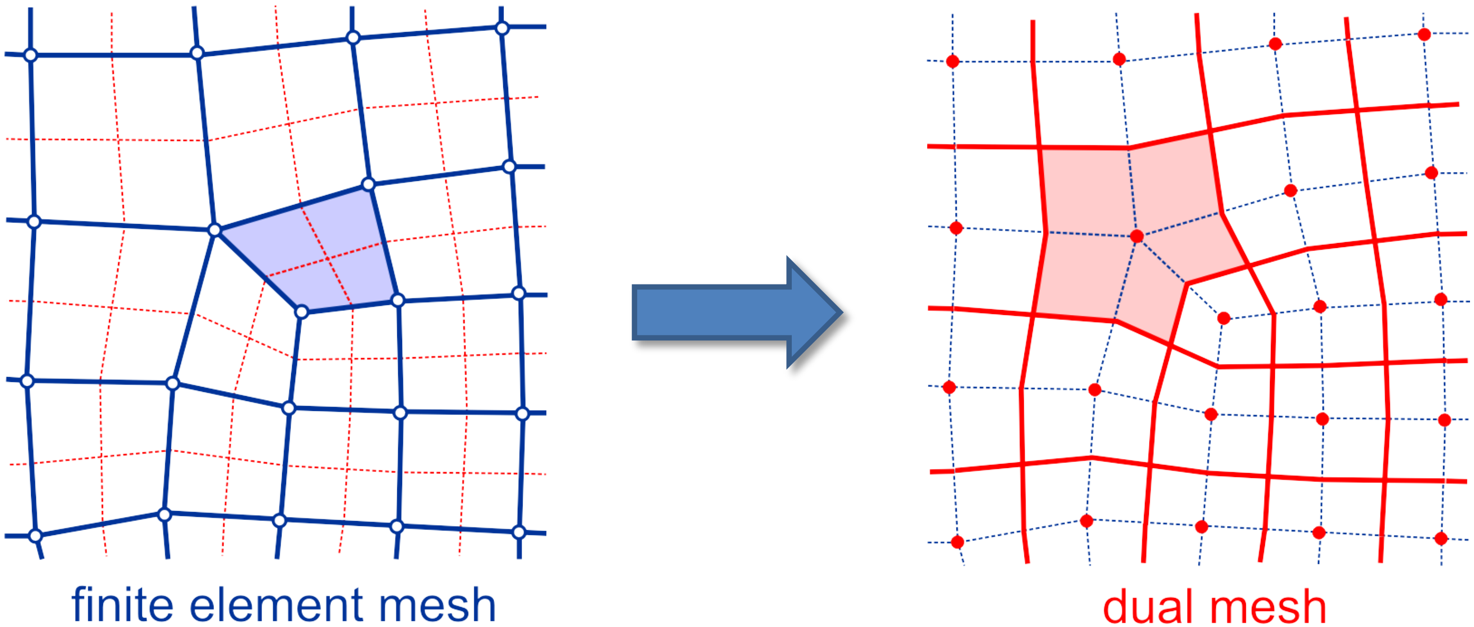
➔ Highly accurate ALE convection scheme
- for 2D/3D problems
- on structured/unstructured meshes
- using any kind of elements
  (including EAS – Enhanced Assumed Strain)



Nodal values

Convection scheme for the nodal values (temperatures, velocities, etc.)

The same methods (Godunov or projection) can be applied on a “dual mesh”



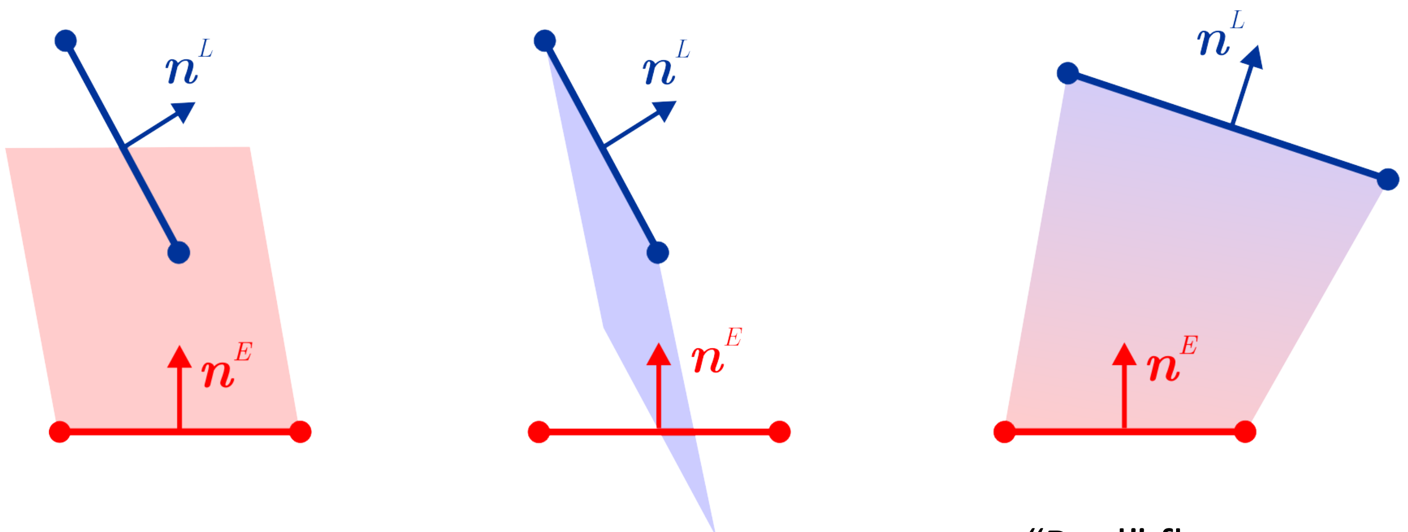
Main issues solved in this work

- Automatic construction of the dual mesh in 3D
- Unique set of routines for both problems (nodal and GP)

Flux computation

Three ways are commonly used in literature

The scheme accuracy can be highly reduced using a bad approximation of the fluxes



"Eulerian" flux

"Lagrangian" flux

"Real" flux
(swept area)

Eulerian and Lagrangian fluxes:

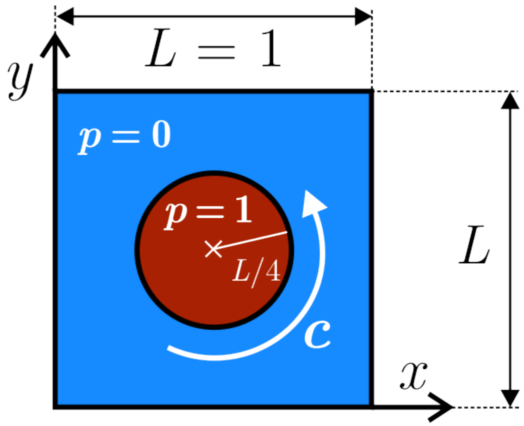
- Very fast
- Translations only!

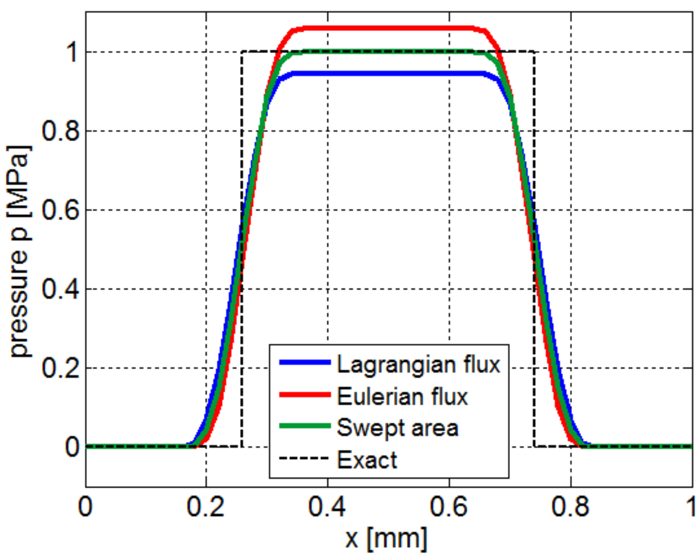
"Real" flux (swept area):

- Slower (in 3D)
- Translations/Rotations

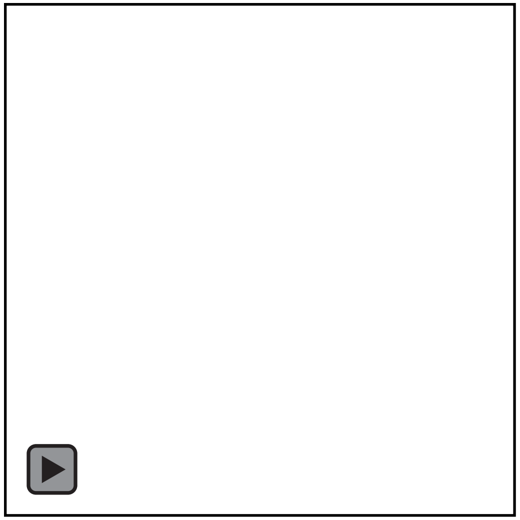
Flux computation

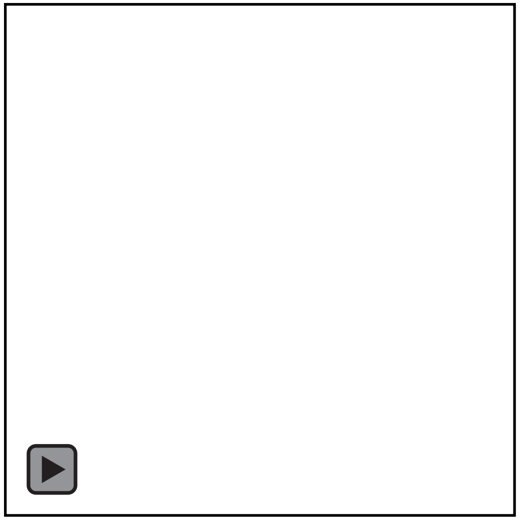
Numerical example





The mesh is fixed – material rotation: 1 revolution





Transverse fluxes are observed.

Transverse fluxes can be kept small using
- a linear reconstruction of p **AND**
- an exact flux computation

Boundary conditions

2. "Lagrangian" lines with "sliding nodes"

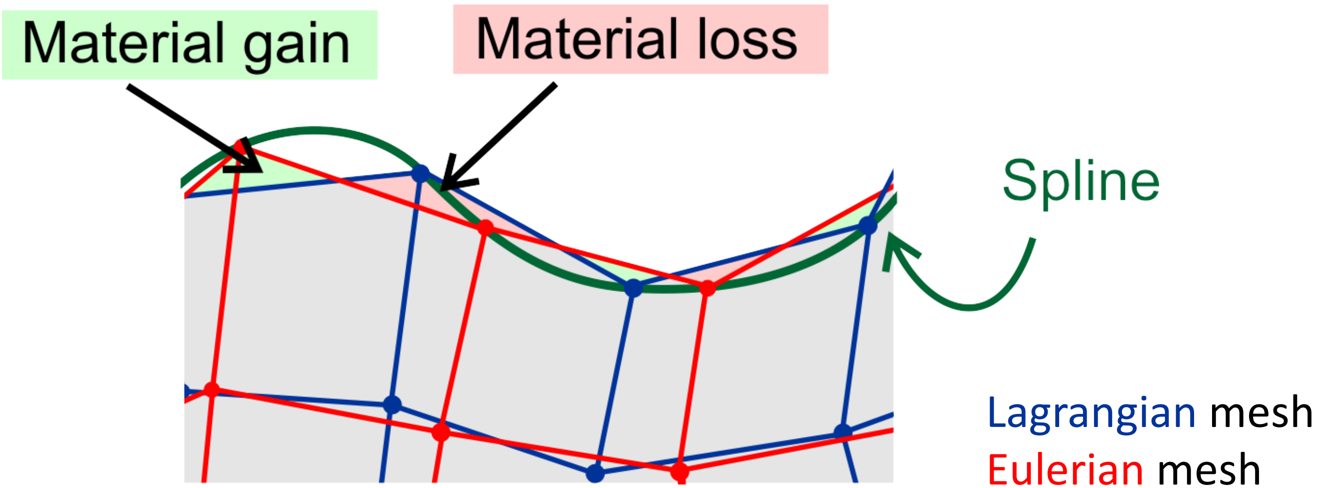
Difficulty

The material slides on the line
➔ The physical flux should be 0
Piecewise linear geometry + spline method
➔ The numerical flux is small but not 0

Two considered solutions

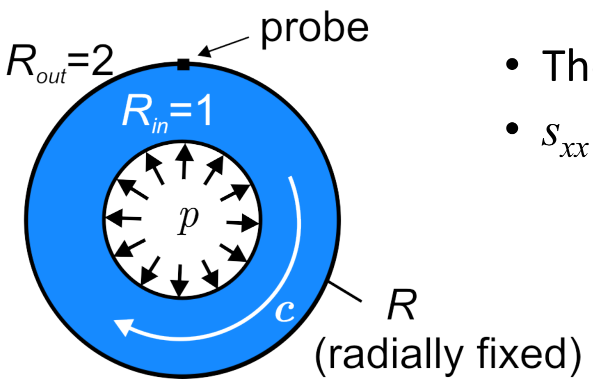
The flux is explicitly set to 0

The flux is computed (using an extrapolation of σ if needed)

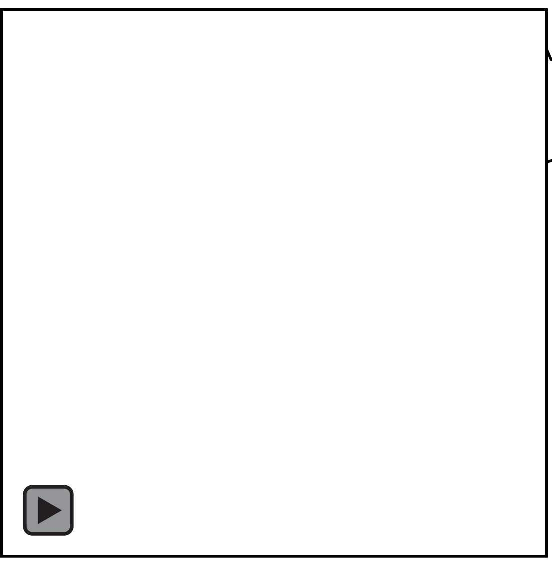


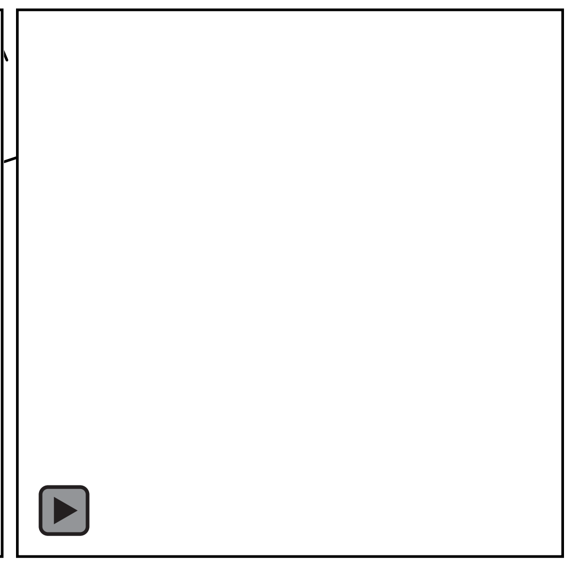
Boundary conditions

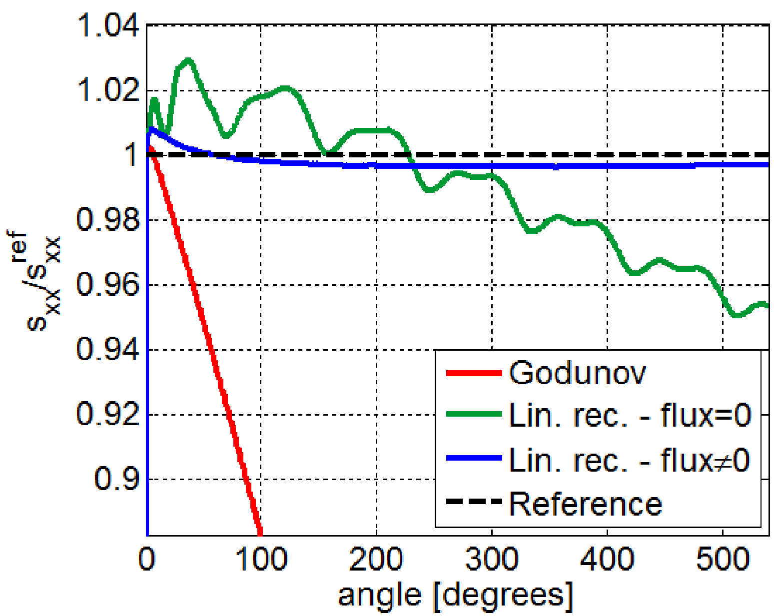
2. Lagrangian lines with "sliding nodes" – numerical example



- The mesh is fixed – material rotation : 1.5 revolution
- s_{xx} is observed and should remain constant due to symmetry


Boundary flux is set to 0

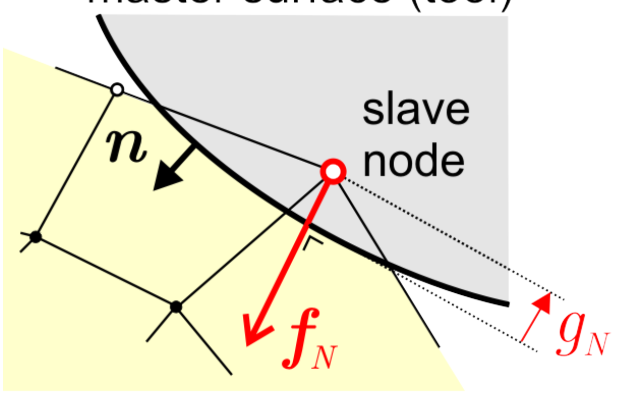

Boundary flux is computed



➔Boundary fluxes must be computed to get a steady (stationary) solution

Contact with friction

Contact (normal) force on the new mesh



Lagrangian step: (penalty method)

$$f_N^{ext} = -\alpha_N \, g_N$$

Similar to an elastic material
➔ no history variable

Eulerian step:

- Contact is ignored during the mesh management step
- The contact force is computed from the equilibrium

$$\underbrace{f_N^{ext} = f_N^{int} + f_N^{inert}}_{} \longrightarrow \quad g_N = -\frac{f_N^{ext}}{\alpha_N}$$

Computed using the values of the variables ($\boldsymbol{\sigma}^{\mathrm{E}}$, $\boldsymbol{a}^{\mathrm{E}}$, $\mathbf{M}^{\mathrm{E}}$) on the Eulerian mesh

The gap is NOT corrected to avoid spurious fluxes

Contact with friction

Friction (tangent) force on the new mesh

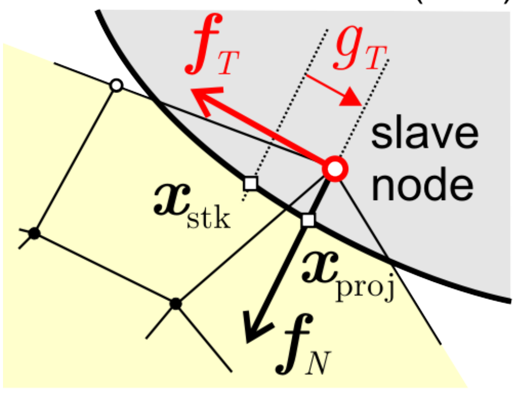
Lagrangian step: (penalty method)

$$f_T^* = \alpha_T \, g_T$$

$$\begin{cases} f_T^* > \mu \, f_N \rightarrow f_T = \mu \, f_N & \text{(slip)} \\ f_T^* \leq \mu \, f_N \rightarrow f_T = f_T^* & \text{(stick)} \end{cases}$$

Similar to an elastoplastic material

➔ history variable = $\boldsymbol{x}_{\text{stk}}$



Eulerian step:

$$f_T^{ext} = \underbrace{f_T^{int} + f_T^{inert}} \longrightarrow g_T = \frac{f_T^{ext}}{\alpha_T} \longrightarrow \boldsymbol{x}_{\text{stk}}$$

Computed using the values of the variables ($\boldsymbol{\sigma}^{\text{E}}$, $\boldsymbol{a}^{\text{E}}$, $\mathbf{M}^{\text{E}}$) on the Eulerian mesh

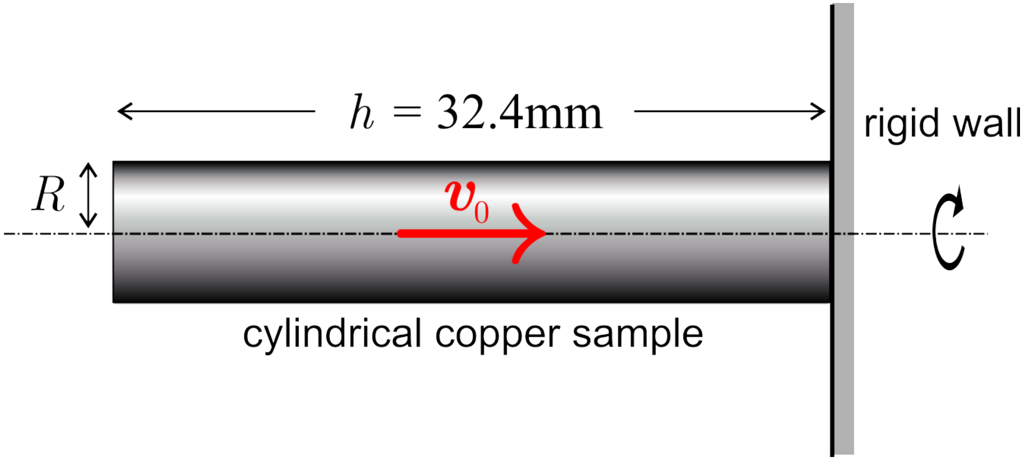
New gap for the next time step

Contents

Taylor's bar impact

Description of the test

- Impact of a cylindrical copper sample
- Used in practice to study the behaviour of materials at very high strain rates (v_0=227m/s)
- Classical benchmark of the ALE formalism

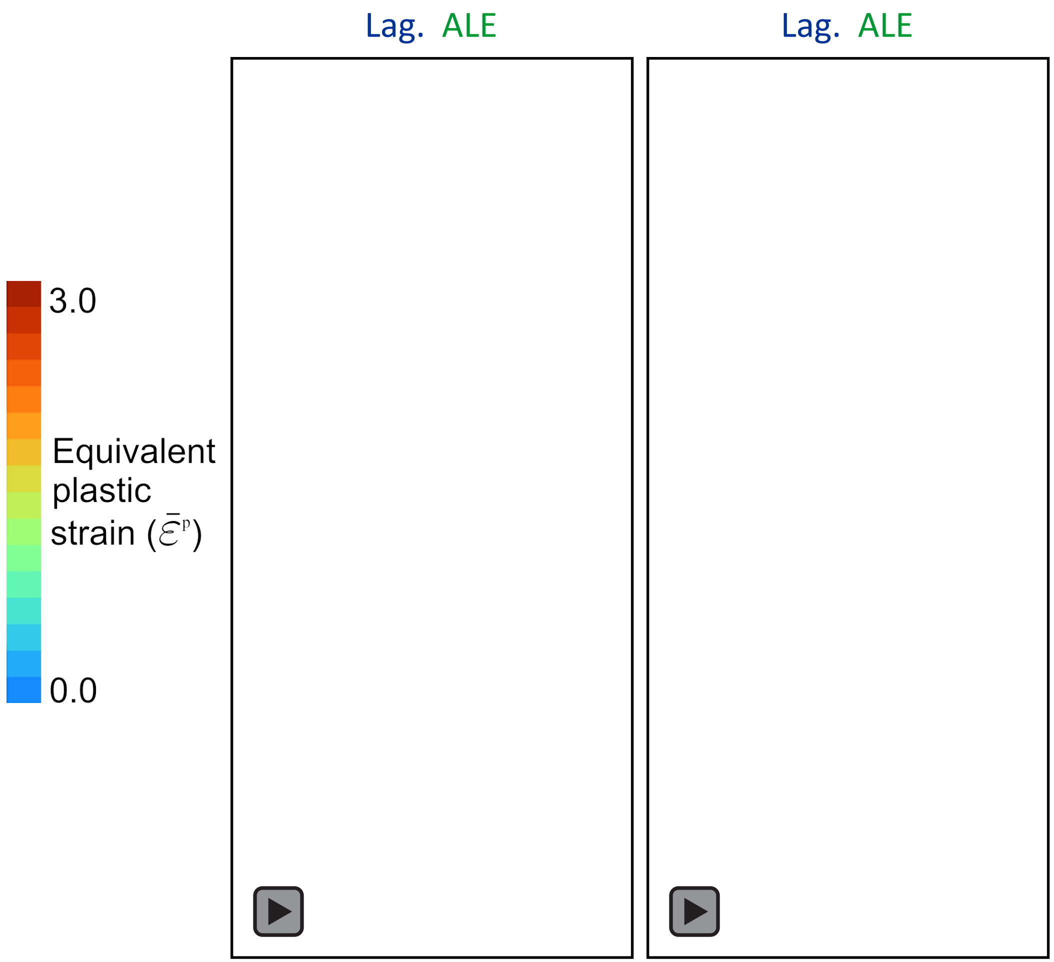


Why ALE?

- Explicit time integration
- The time step size (Δt) is proportional to the size of the smallest element of the mesh
- ALE helps to control this size (➔ $\Delta t_{ALE} > \Delta t_{Lag}$)
- ALE should be faster than Lagrangian models

Taylor's bar impact

No velocity field convection (common assumption in literature)

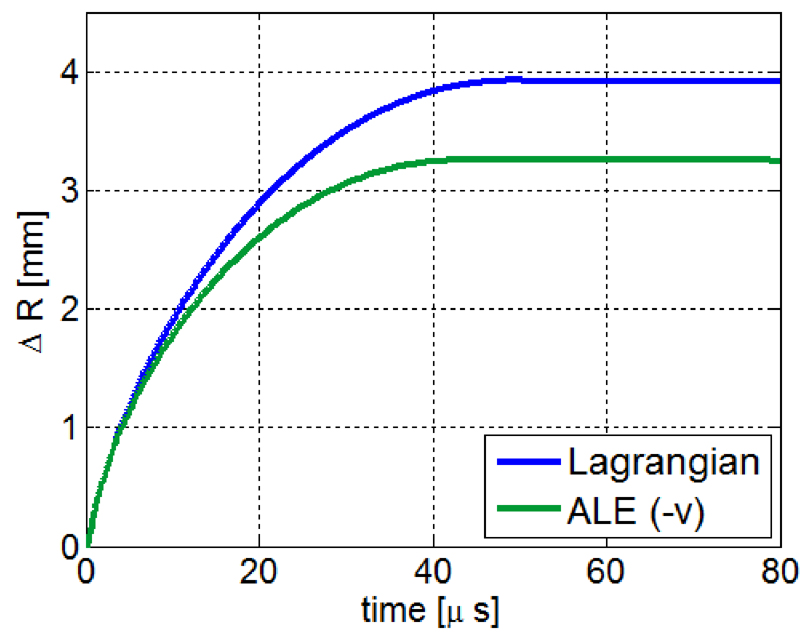


CPU: 1'24'' vs 12''

ΔR_{Lag} > ΔR_{ALE}

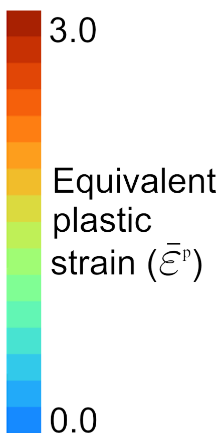
- Only GP values are convected
- Final height/radius are different
- In literature:

  "ALE solution is more accurate because of better-shaped elements"

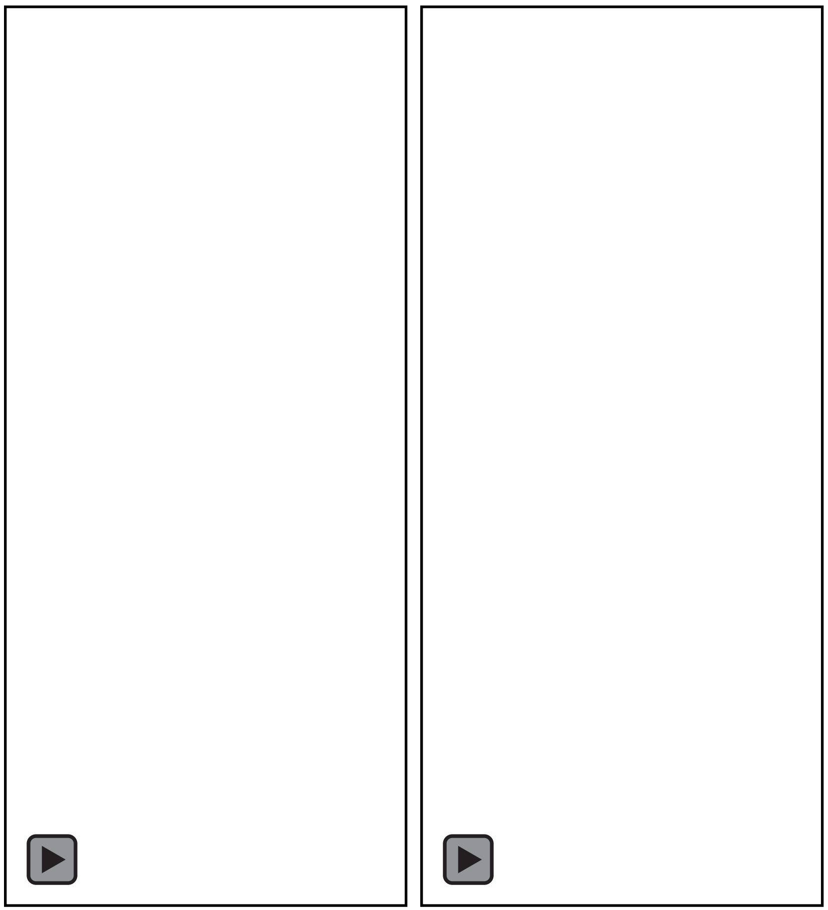


Taylor’s bar impact

Convection of the velocity field is added



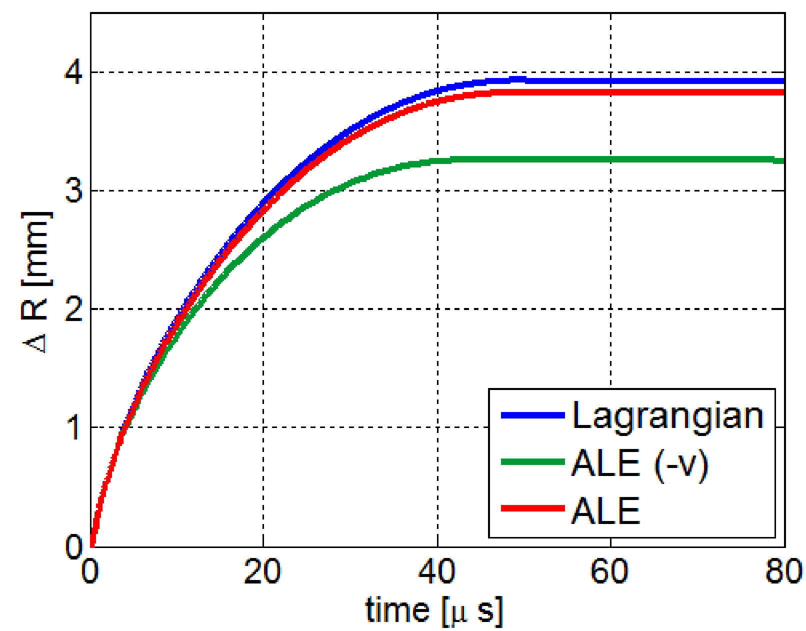
Lag. ALE   Lag. ALE



CPU: 1’24’’ vs 15’’

$\Delta R_{Lag} \cong \Delta R_{ALE}$

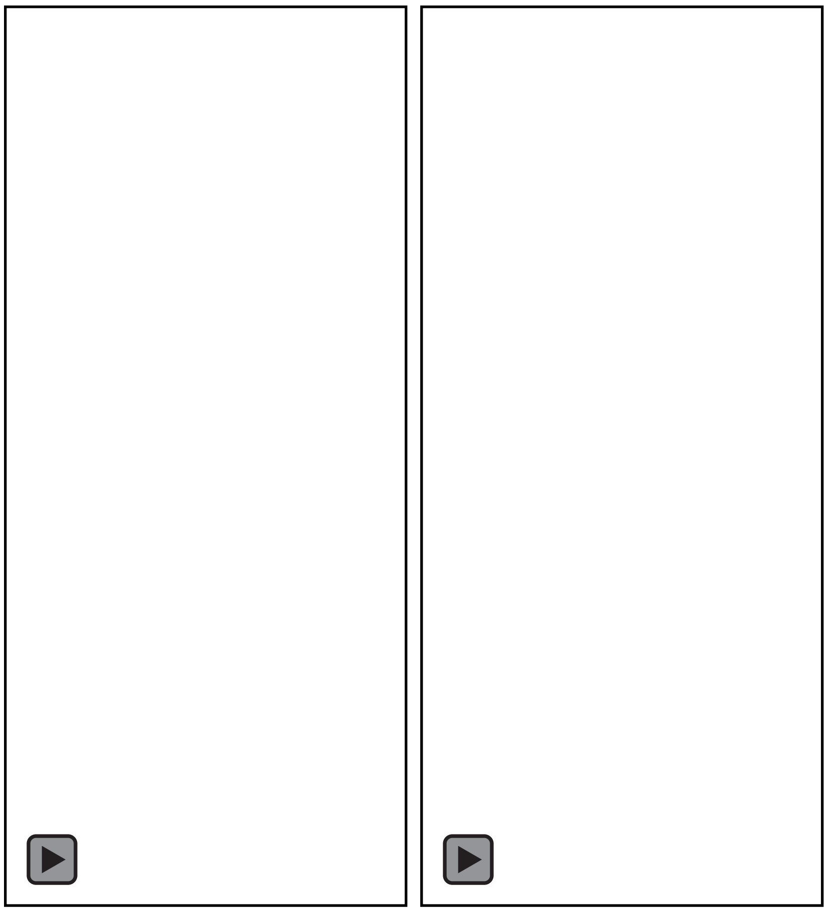
- Convection of GP and nodal values (required for kinetic energy conservation during the Eulerian step)
- Final height/radius are much closer but still different



Taylor's bar impact

The ALE mesh is optimised

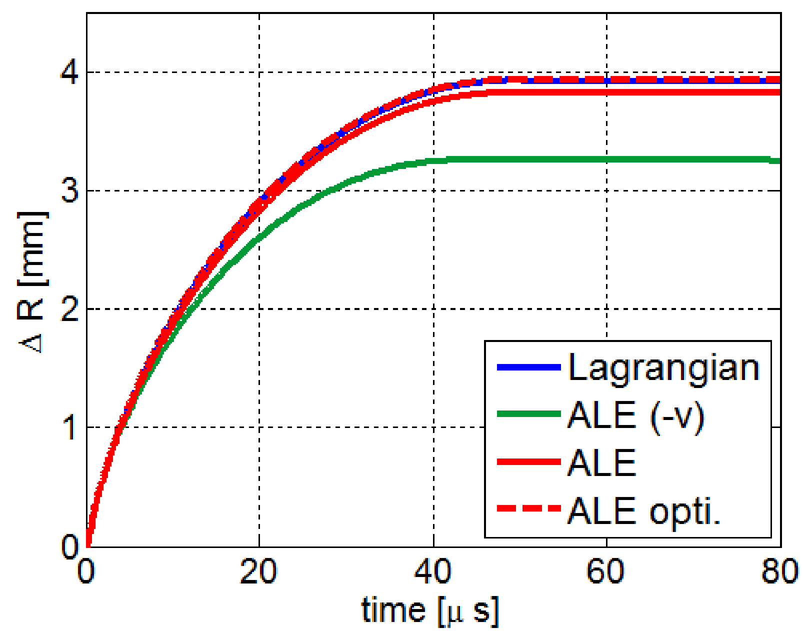
Lag. ALE | Lag. ALE



CPU: 1'24'' vs 31''

ΔR_{Lag} = ΔR_{ALE}

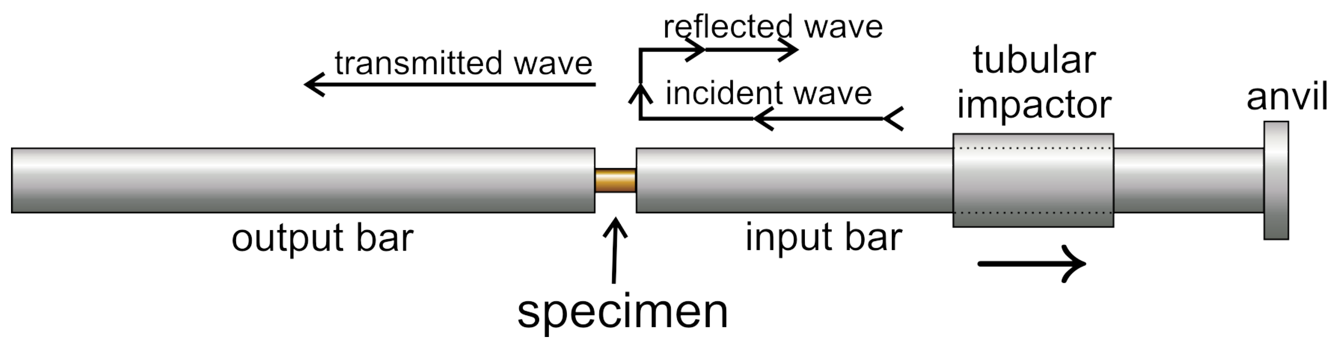
- Same total number of elements than before
- The mesh is densified near the rigid wall (the surface curvature of the bar is better modelled)



Hopkinson's test

Principle of the "Split Hopkinson Tensile Bar"

Characterisation of the behaviour of materials at high strain rates



Why ALE?

- Ductile fracture: the geometry of the specimen is badly modelled when necking occurs
- Accurate Lagrangian results require a very fine mesh
- CPU_{ALE} is expected to be smaller than CPU_{Lag} for a given accuracy in the results

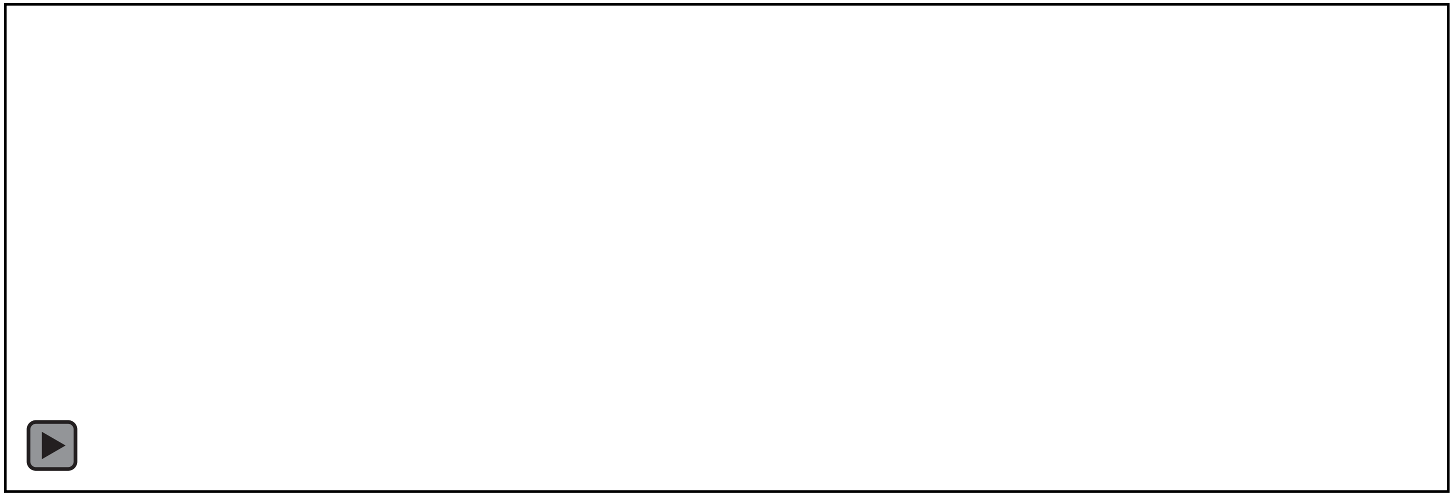
Parameters (Noble et al. 1999)

- Thermomechanical problem (temperature field should be convected)
- Staggered implicit dynamic time integration (Chung Hulbert)
- Elastoviscoplastic material (iron) modelled by a Zerilli-Armstrong law

Hopkinson's test

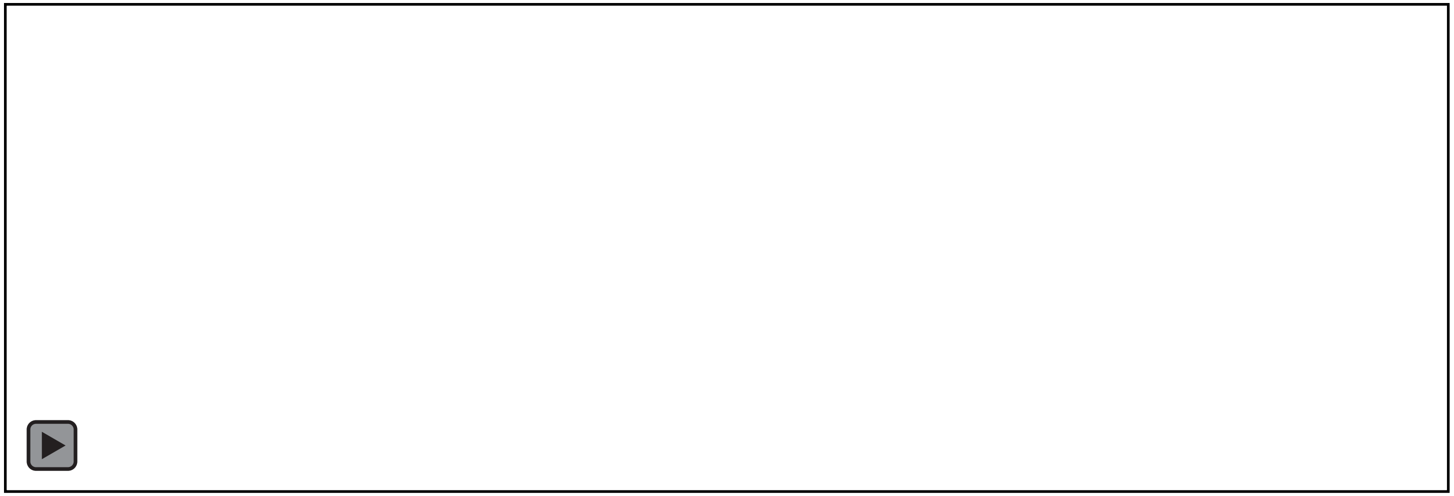
ALE vs. Lagrangian results (same mesh)

Lag.

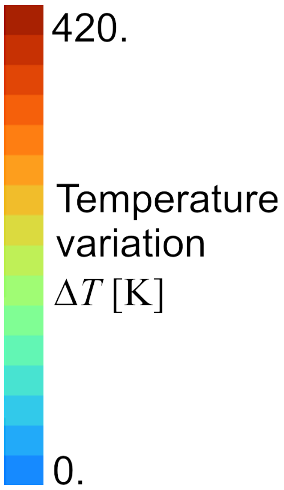


ALE

Lag.



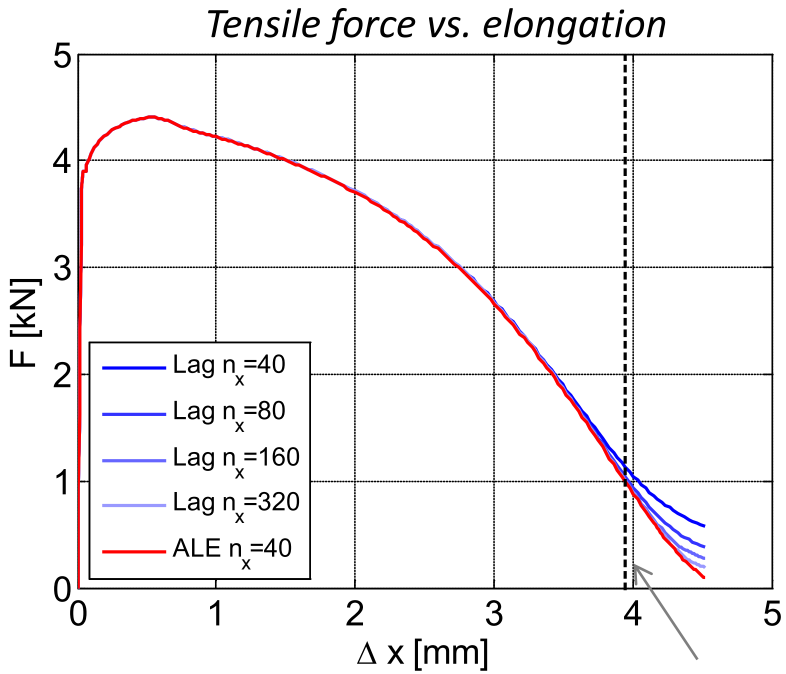
ALE

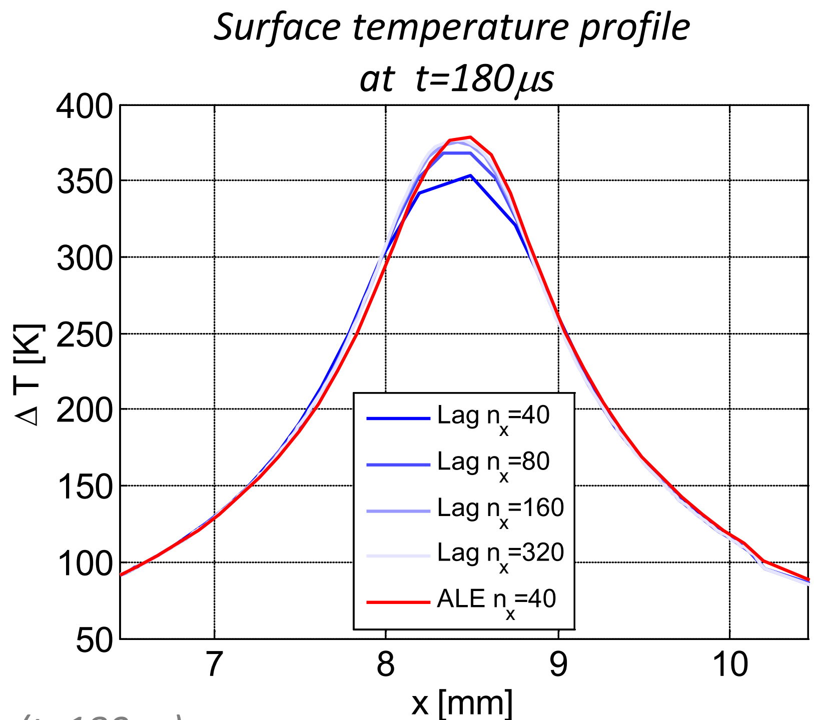


Hopkinson's test

ALE vs. Lagrangian results

(n_x = number of elements along the necking zone)



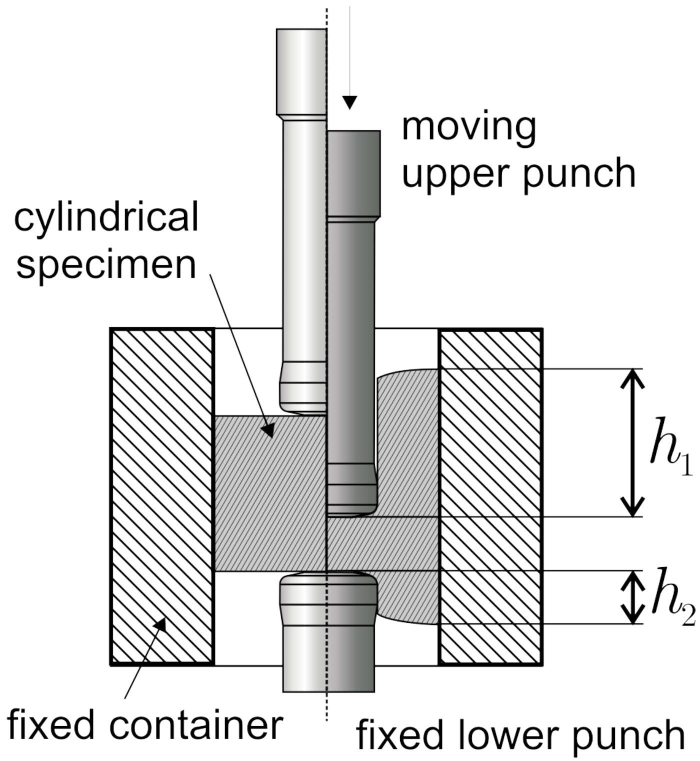


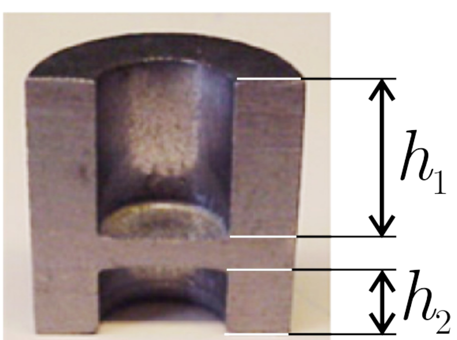
experimental fracture (t=180μs)

If the Lagrangian mesh is refined, the ALE results with the coarsest mesh are retrieved!

CPU_{ALE} = 1'07'' < CPU_{Lag} = 3'13''

Double cup extrusion test (DCET)





Process description

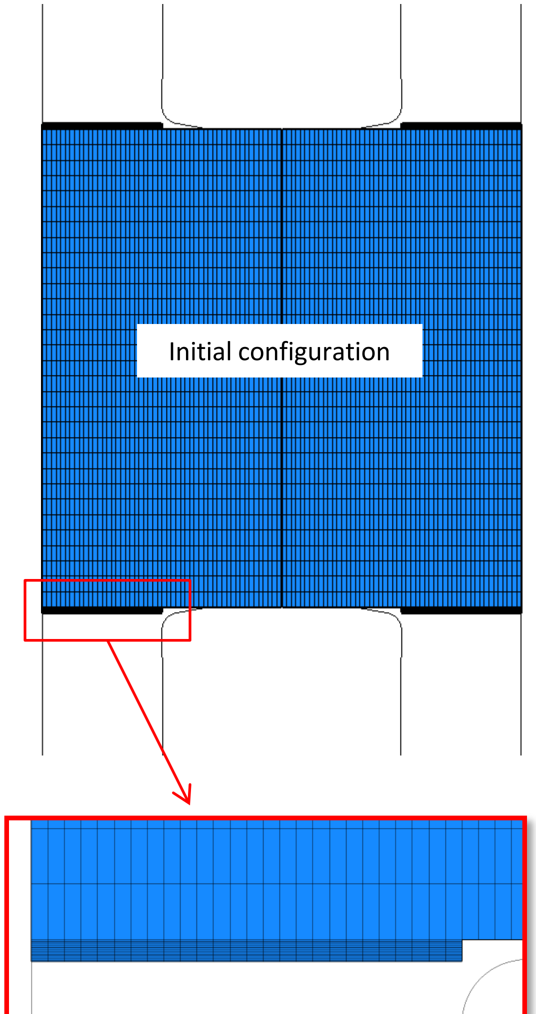
- Experimental friction test (Bushhausen /Altan - Ohio State University - 1992)
- Mean friction coefficient is deduced from the comparison of experimental and numerical (FEM) "cup height ratios" h_1/h_2 (=1 if frictionless)
- Frictional conditions close to the forging process
  - Interface pressure ~ 2500MPa
  - Surface temperature ~600°C
  - Surface enlargement ~3000%
- Typical numerical simulation needs remeshing

Why ALE?

- Complete remeshing is avoided

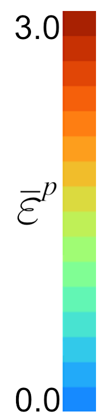
Double cup extrusion test (DCET)

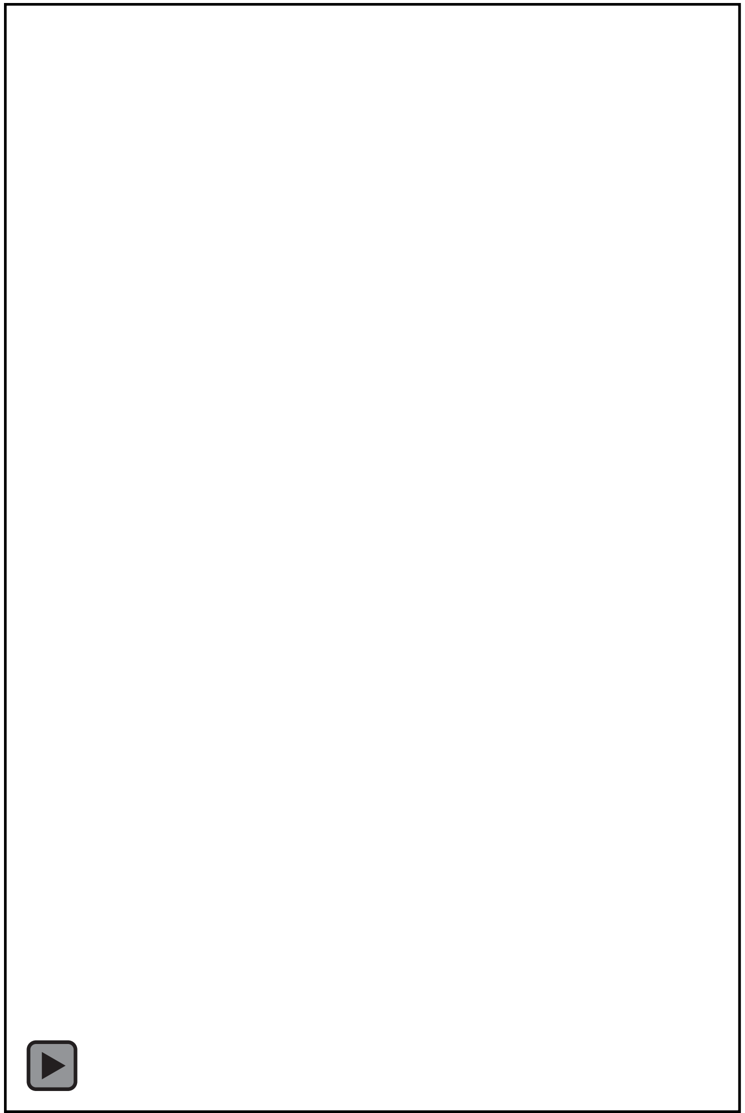
ALE mesh management



Observations

- the Godunov scheme is sufficient
- The volume remains constant
- CPU time: 5'45''





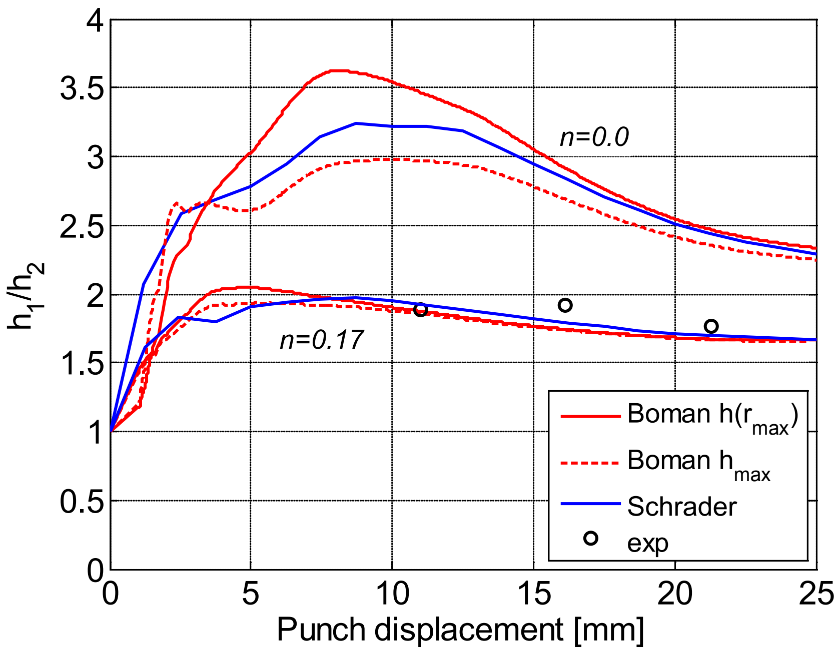
Numerical trick

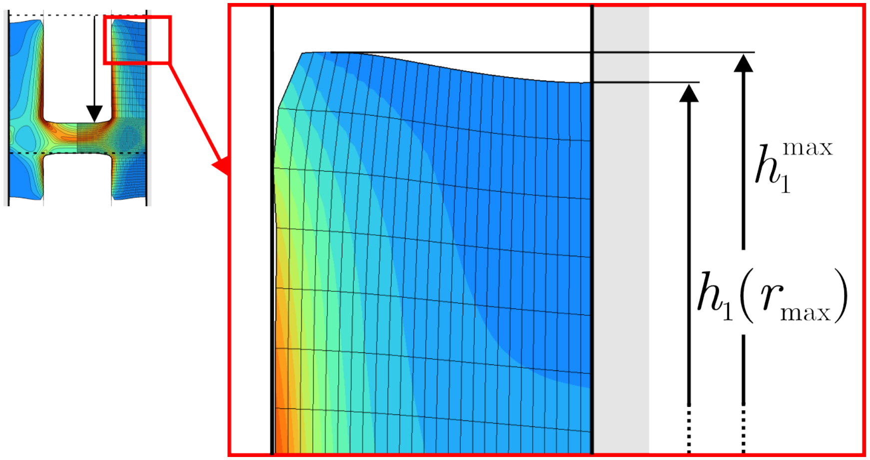
An initial thin squeezed mesh is added to the top of the billet (Atzema & Huétink 1992)

Double cup extrusion test (DCET)

Comparison with Schrader's results (2007)

- Material (AISI 1018 Steel) : $\overline{\sigma} = K\,\overline{\varepsilon}^{\,n} = 735\,\overline{\varepsilon}^{\,0.17}$
- Comparison with experimental results
- Comparison with FE results obtained using Deform2D (and remeshing)

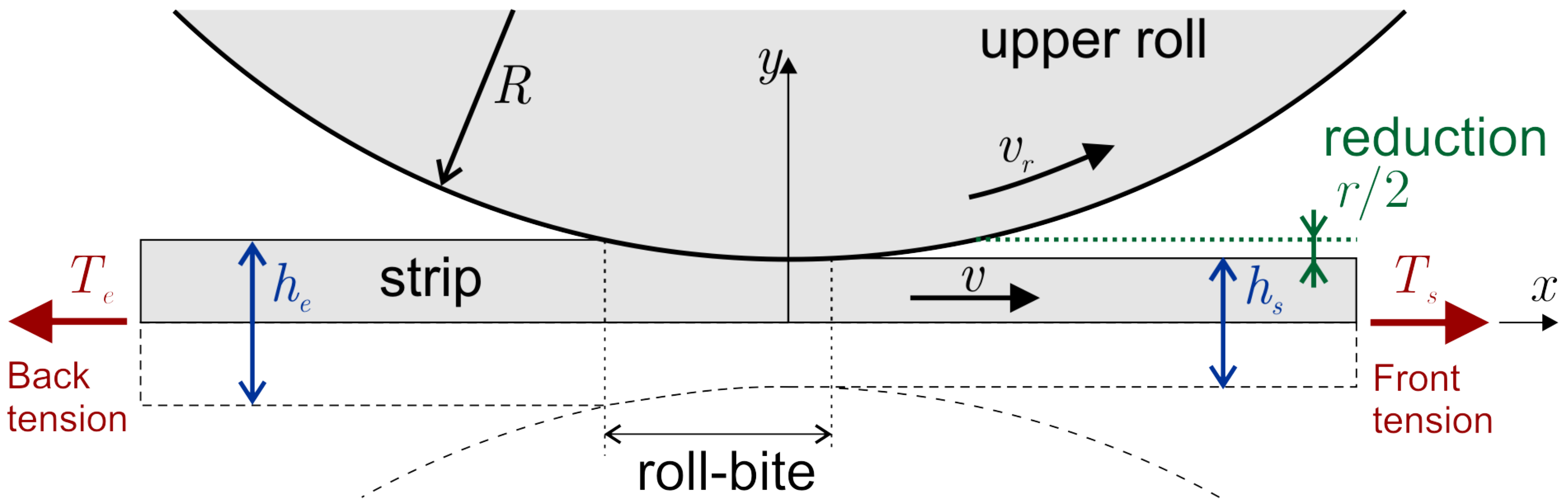




➔ The computation of frictional contact on arbitrary moving meshes is validated

Rolling

Process description



Why ALE?

ALE vs. Lagrangian codes:

- Shorter model, less finite elements → $CPU_{ALE} < CPU_{Lag}$ is expected

ALE vs. Eulerian codes:

- Eulerian codes are too specialised, too difficult to maintain
- Unsteady phenomena can also be studied using ALE (defects, vibrations)

Rolling

Convection – Constant or linear reconstruction?

Hot rolling (thick strip and rigid rolls)

ALE cst (158'')

Lag. (138'')

ALE lin. (502'')

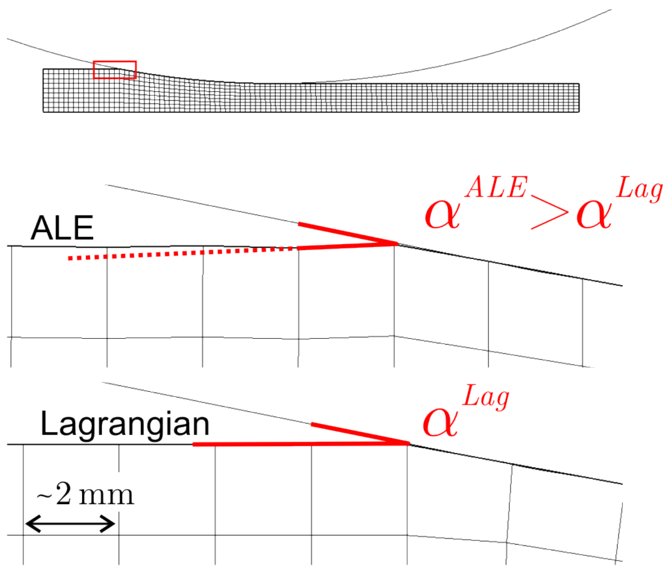
Lag. (138'')

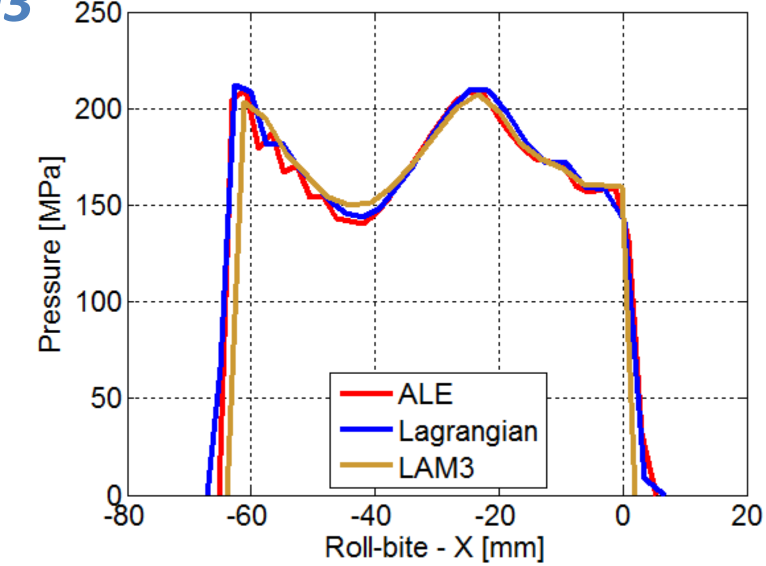
→ Constant reconstruction is sufficient if the transient state is not important!

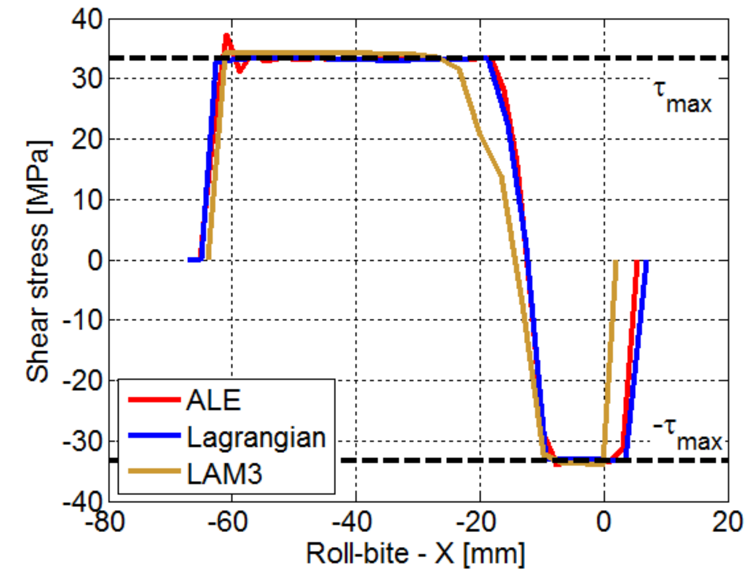
Rolling

Numerical results - Comparison with LAM3

Good match between results but
the ALE model shows spurious oscillations
near the inlet zone!







The spline method cannot model the sharp inlet angle α and oscillations propagate around it

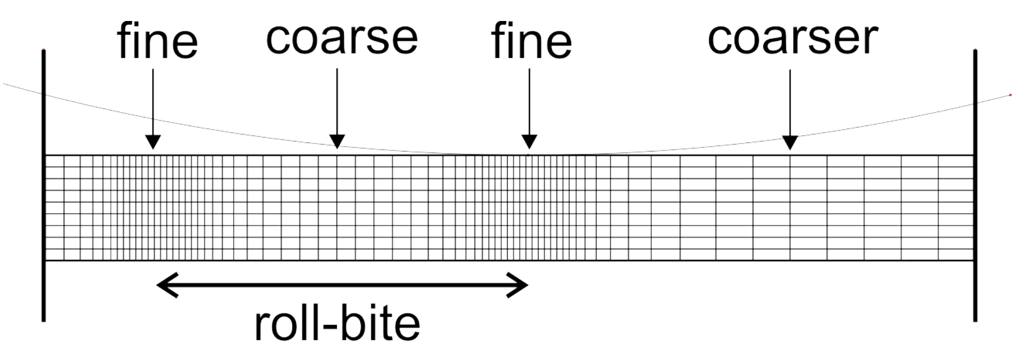
Rolling

Numerical results - mesh optimisation

Fair comparison

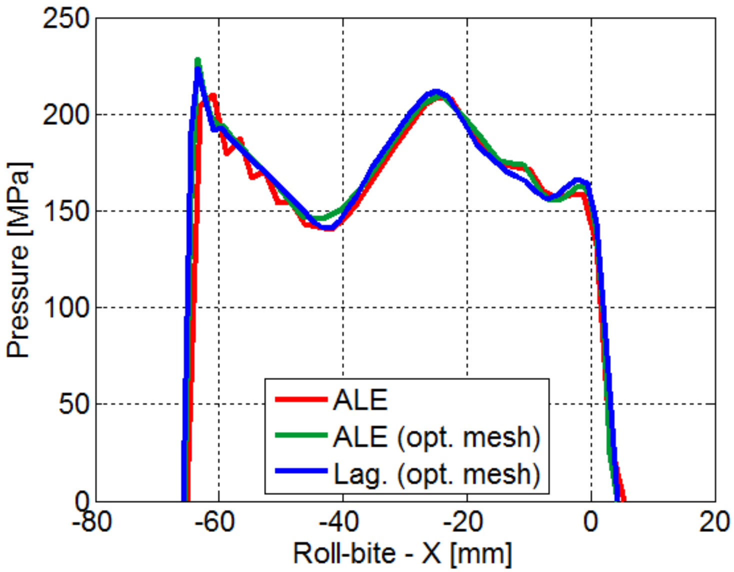
Both (Lagrangian/ALE) meshes are optimised

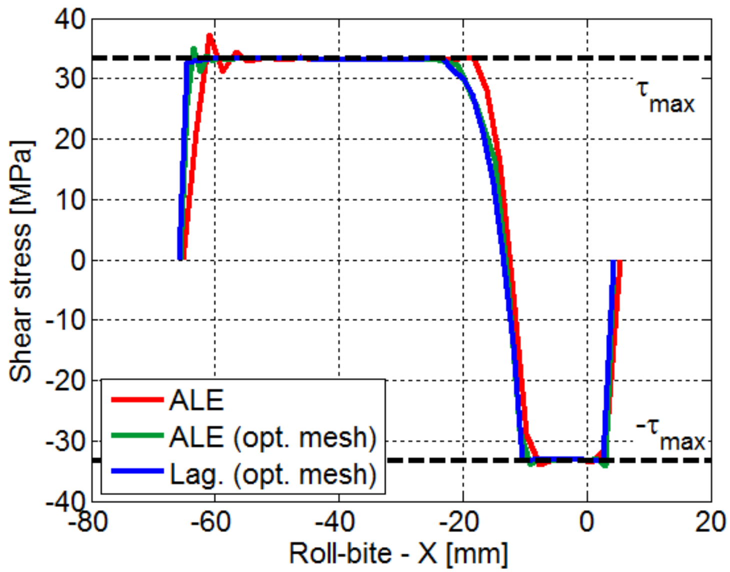
Optimised ALE mesh



Results

- CPU reduction (CPU_{ALE} = CPU_{Lag} = 52'')
- Inlet oscillations are smaller

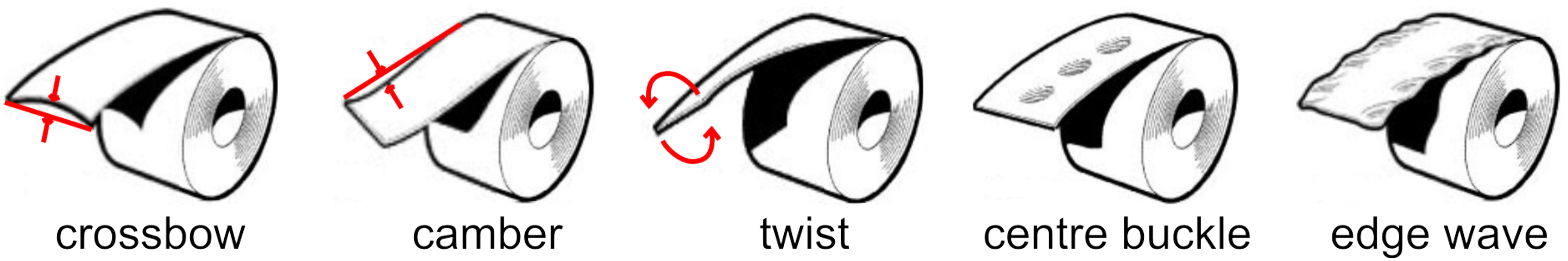




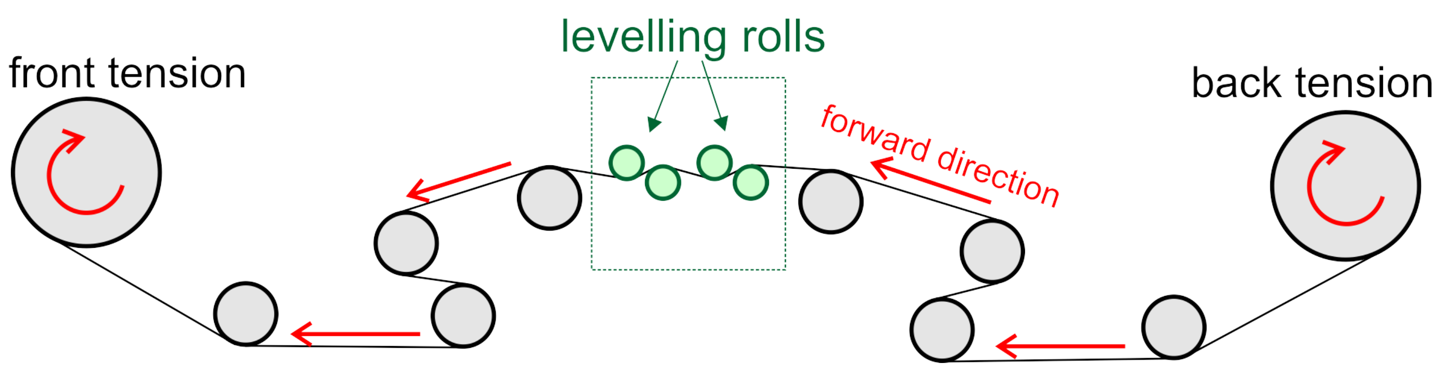
Tension levelling

Process description

Used at the end of the forming line to remove shape defects using plastic bending and stretching



Example: Pilot Mill of ArcelorMittal – Maizières – France



Tension levelling

Why ALE?

Quasi Eulerian model
➔ Less ALE elements than in the Lagrangian formulation.
➔ CPU_{ALE} < CPU_{Lag} is expected

Process parameters

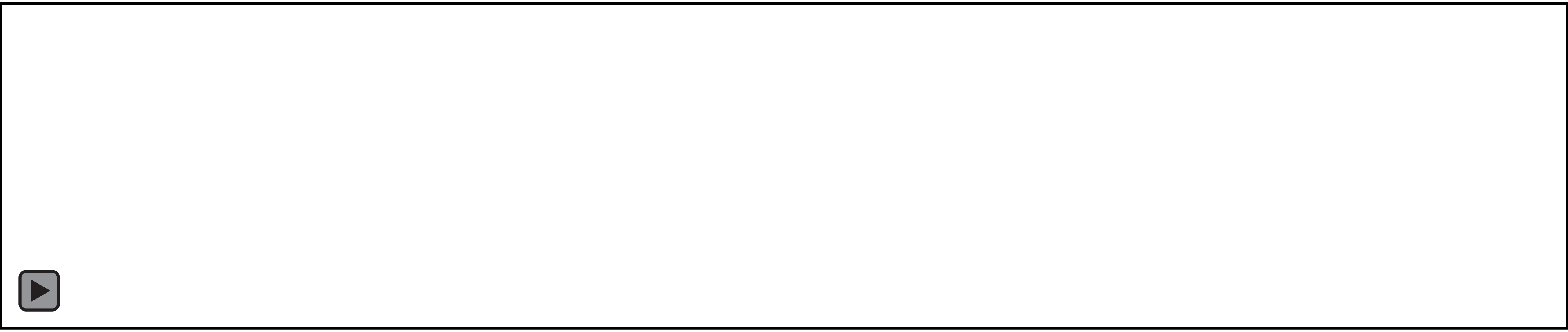
- Experimental results from the pilot line of ArcelorMittal are available.
- Material: Dual Phase DP600 Steel
- The strip has no initial defect ➔ the process generates camber (and crossbow in 3D)

Numerical parameters

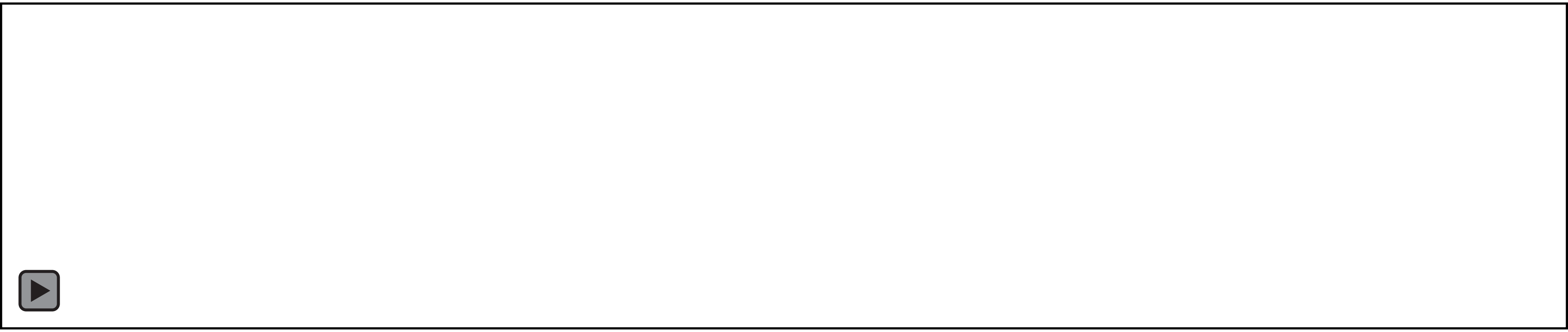
- EAS (Enhanced Assumed Strain) elements
- Chung-Hulbert implicit dynamic scheme (but $\boldsymbol{a}$ and $\boldsymbol{v}$ are not convected)
- Rolls are rigid and free to rotate

Tension levelling

Lagrangian model

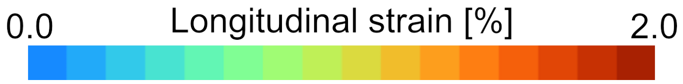


ALE model



Main results of the model

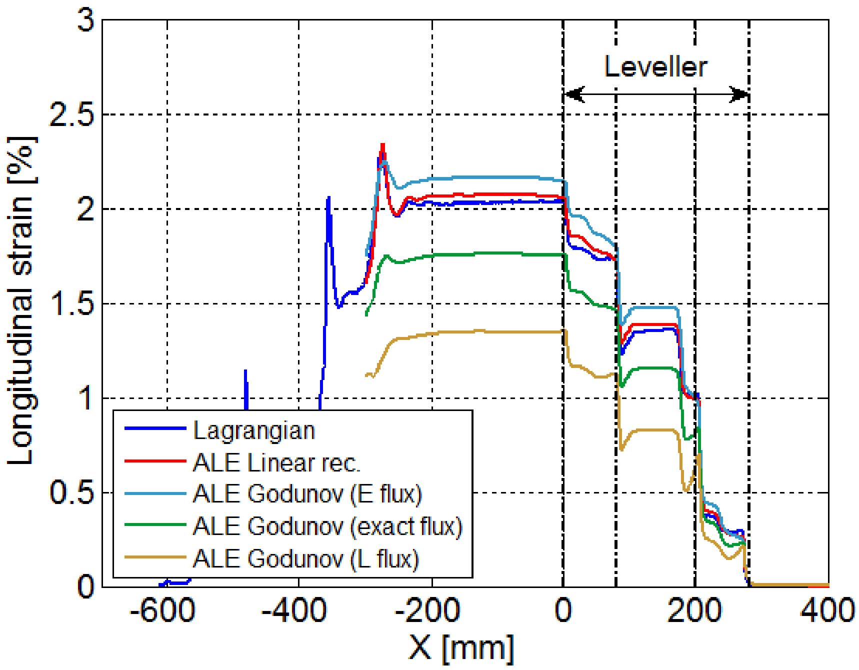
- Longitudinal strain (**F** tensor must be convected)
- Camber radius after springback



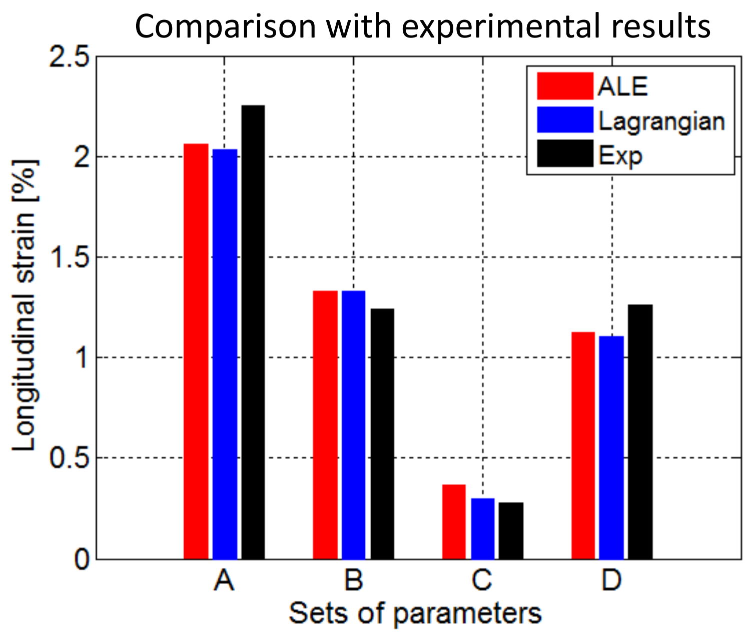
Tension levelling

Convection scheme

- The Godunov scheme is **NOT** sufficient for retrieving the Lagrangian longitudinal strain
- The flux computation method has a large influence on the ALE results



Linear Rec. Scheme

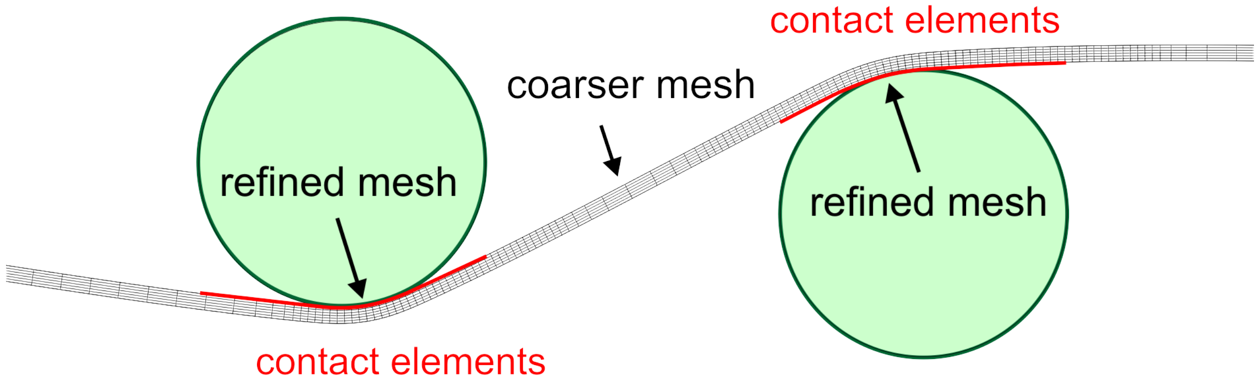


Tension levelling

ALE Mesh optimisation

- Linear reconstruction is very CPU expensive (+50% compared to Godunov)
- Both models are optimised

Optimised ALE model



CPU Times	Lagrangian	: (4566 FEs)	1h20’40’’	ALE is
	ALE	: (1818 FEs)	48’44’’	1.7x faster!

Conclusion

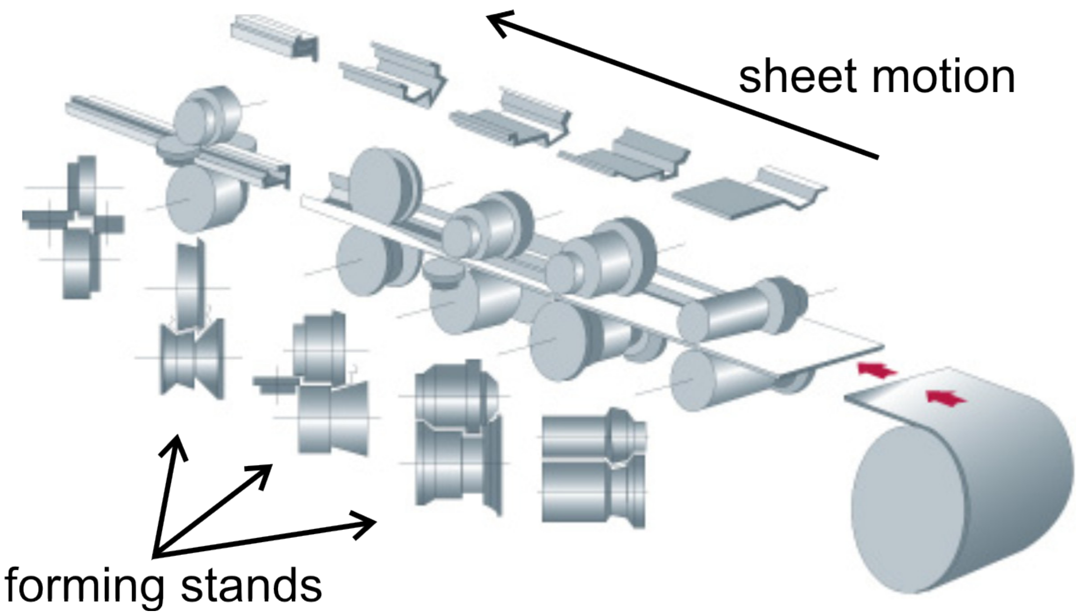
The ALE model is faster than the Lagrangian one (but requires more optimisation efforts)

Contents

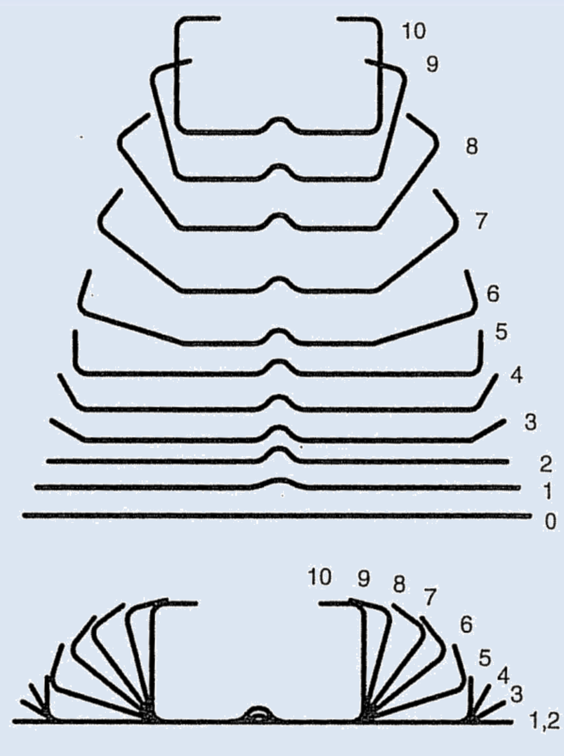
Roll forming

Process description

A metal strip is incrementally bent by sets of rolls (called "forming stands") until the desired cross section is obtained



"Flower diagram"



Why ALE?

Classical Lagrangian model

Features

- ✓ Can handle almost any kind of process (stationary or not) but…
- ✗ Large CPU times when sheet length increases
- ✗ Complex boundary conditions (friction, rigid body modes, etc)
- ✗ Severe contact conditions when entering/leaving a stand (impact, dynamic oscillations)
- ✗ Uniform mesh of small elements required along the main direction
- ✗ Transverse mesh refinement sometimes shifted due to lengthening

Why ALE?

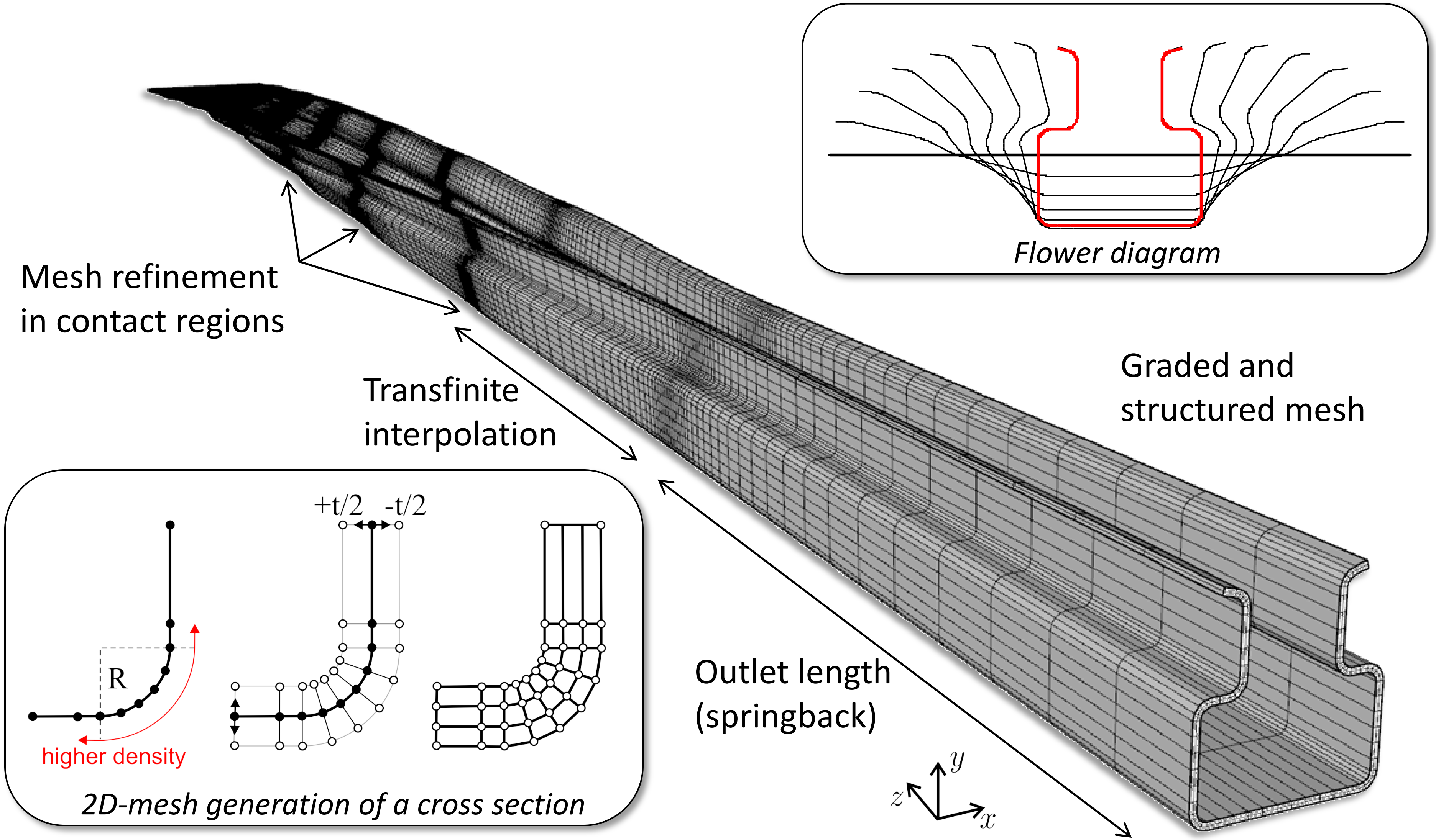
The proposed ALE model of continuous roll forming

Features

- ✓ Boundary conditions easier to handle (1-step simulation)
- ✓ Mesh can be optimised : less finite elements / contact elements
  - ➔ Longer forming line (many stands) can be modelled
  - ➔ More complex shapes are possible
  - ➔ Smaller CPU times are expected
- ✗ Continuous roll forming only
- ✗ Initial mesh?

Initial ALE mesh

Interpolation of the flower diagram



U-channel

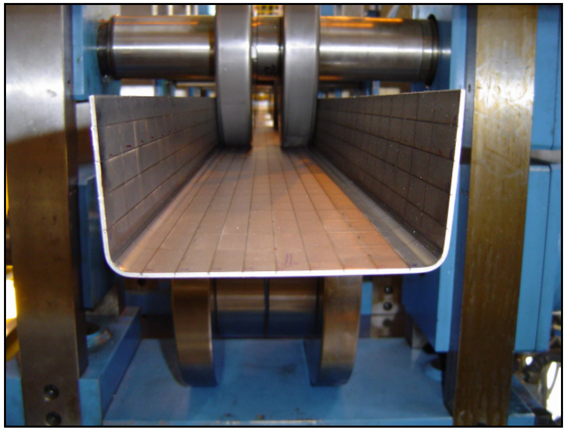
Forming of a symmetrical U-channel

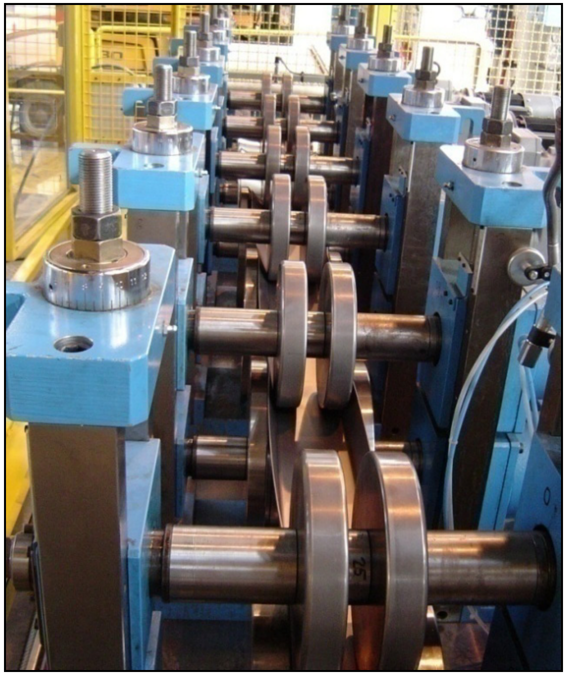
Process parameters

- Experimental mill (ArcelorMittal R&D, Montataire, France)
- 6 stands (15°, 32°, 50°, 68°, 80°, 90°)
- Final bending radii: 6 mm
- Inter-stand distance : 0.5 m
- Sheet : 2000 x 200 x 1.6 mm
- Sheet velocity: v = 200 mm/s
- Coulomb friction μ = 0.2
- DP980 steel (σ_{Y0} = 697.34 MPa)

Numerical parameters

- Symmetry
- Friction drives the sheet
- Two layers of EAS elements
- Dynamic implicit scheme (Chung-Hulbert)

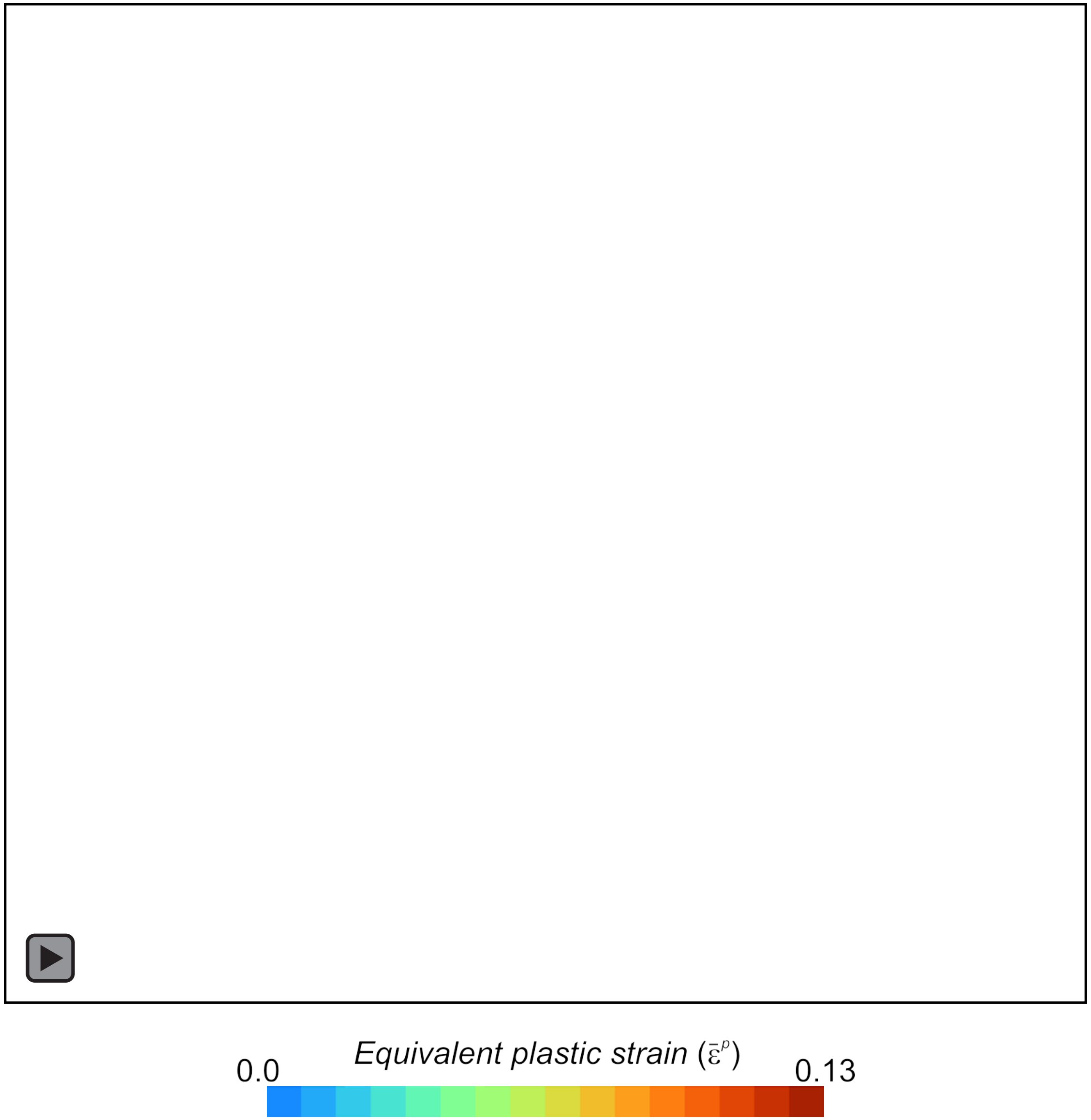




U-channel

ALE simulation results (global view from the exit)

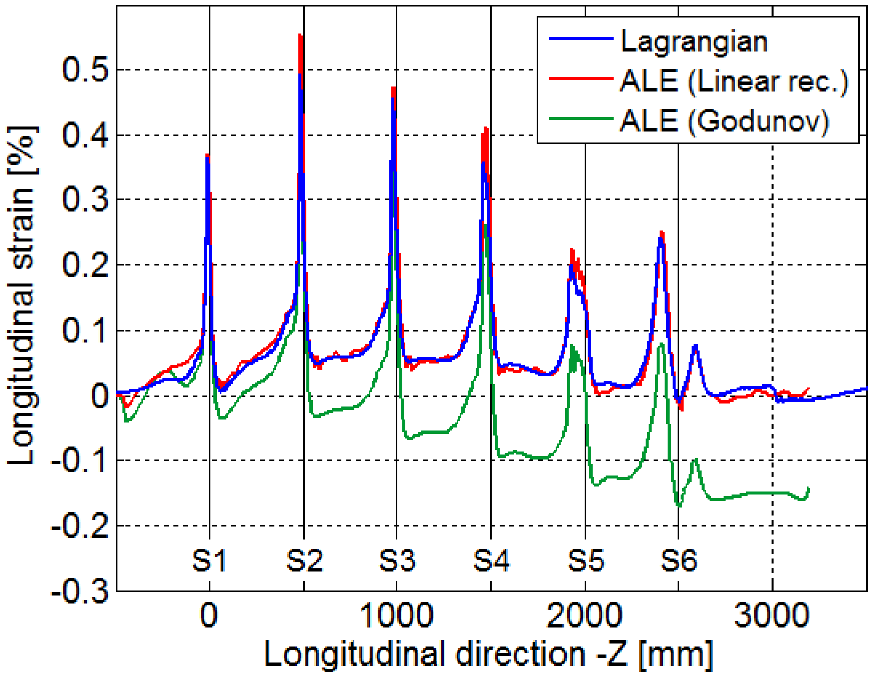
- Strains and stresses propagate through the quasi-Eulerian mesh
- The initial “perfect” U shape is modified and springback can be measured



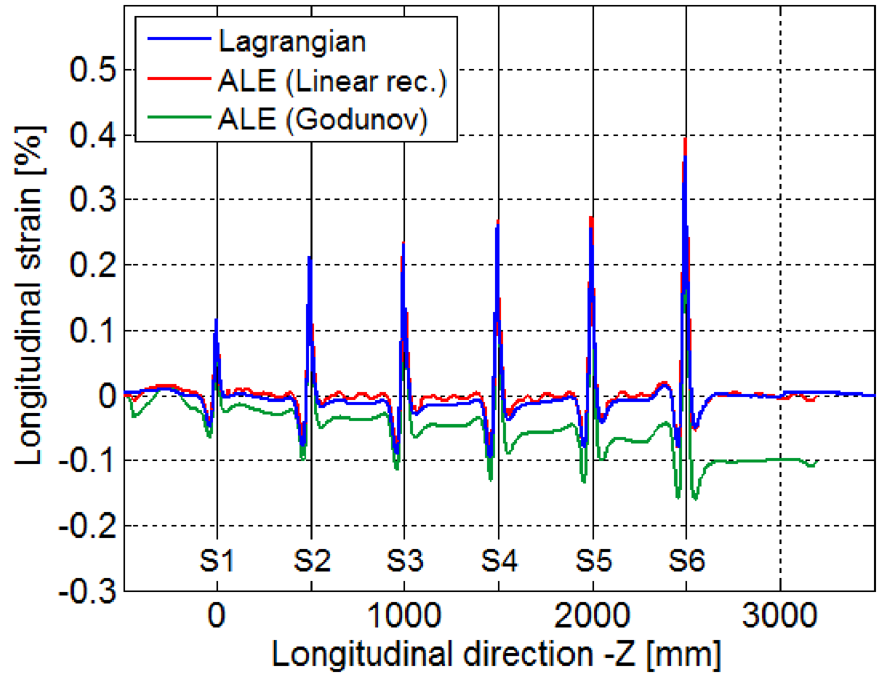
U-channel

Longitudinal membrane strain

…on the edges



…on the symmetry plane

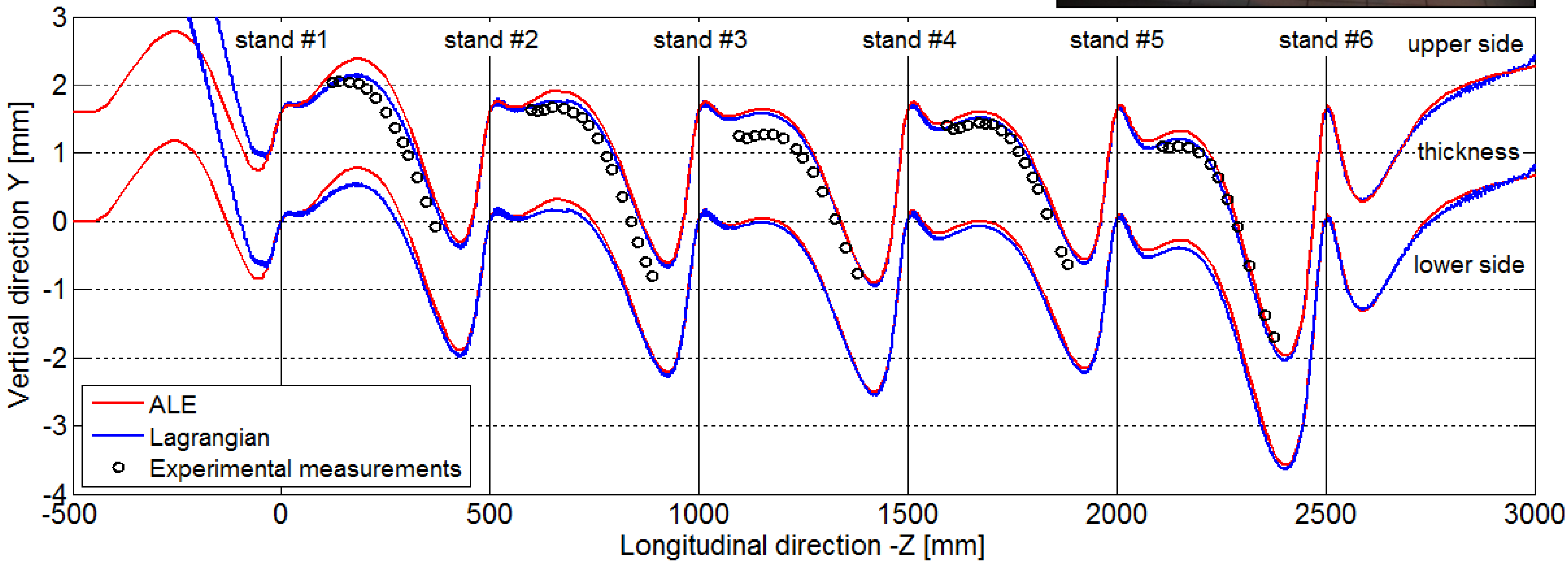


- Bad results are obtained using a constant reconstruction scheme (Godunov).
- Thanks to a linear reconstruction, ALE and Lagrangian longitudinal membrane strain curves are very similar

U-channel

Shape of the sheet inside the mill

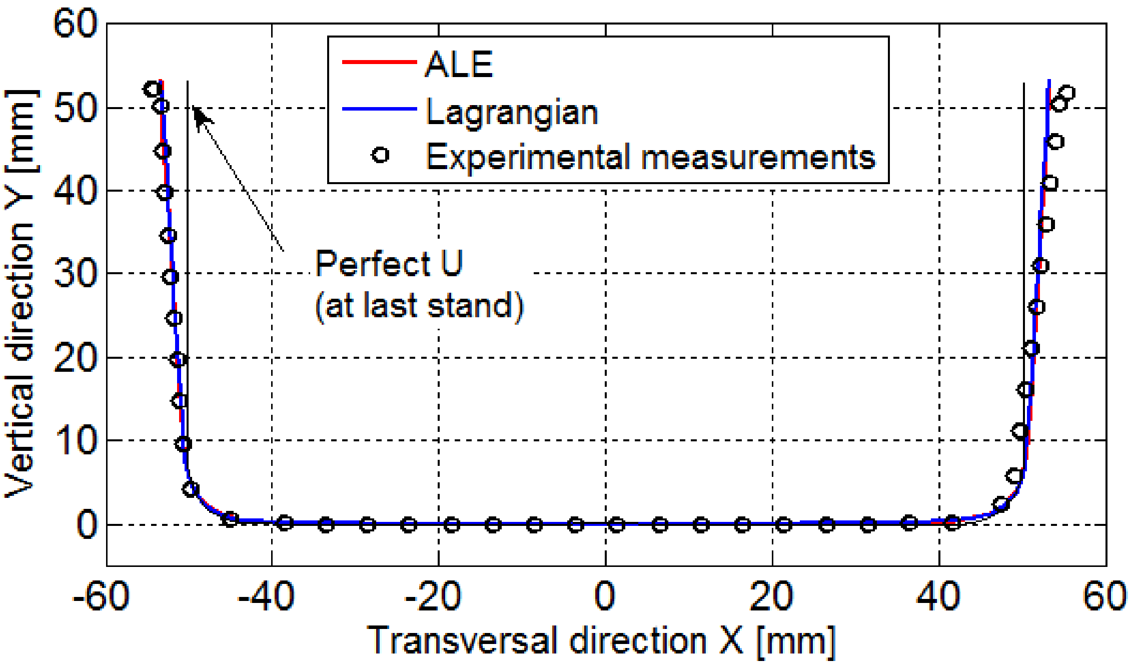
- Experimental data were collected with a portable measurement arm
- The particle trajectory of the Lagrangian simulation is very close to the final ALE mesh shape and nicely matches the experiments

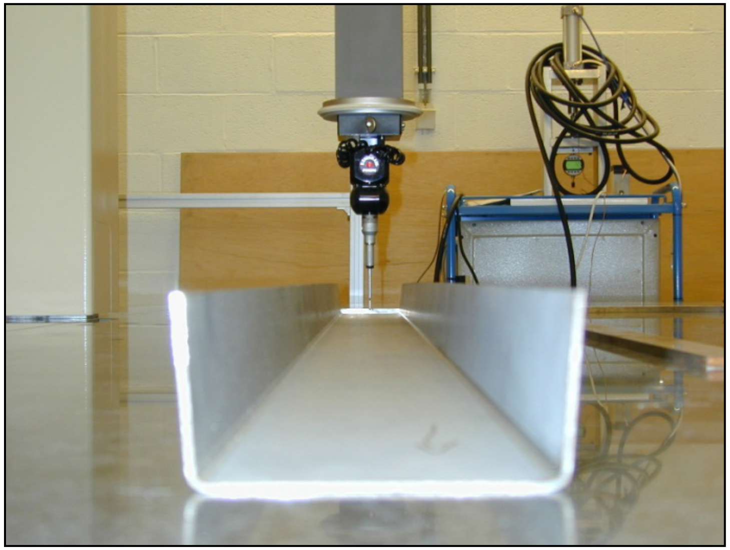


U-channel

Numerical vs. experimental springback

The final shape has been digitised using a high precision 3D measurement device and fits well both numerical curves (courtesy of ArcelorMittal)





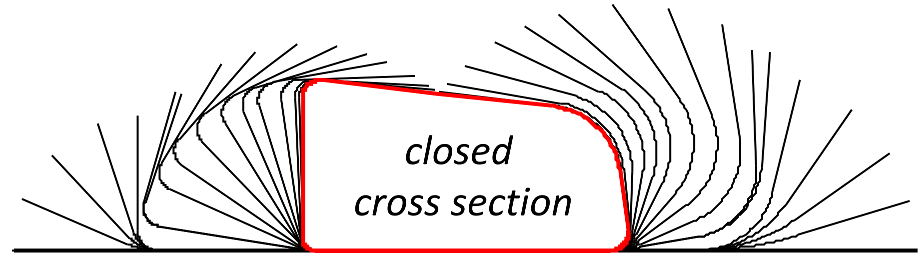
CPU Times:

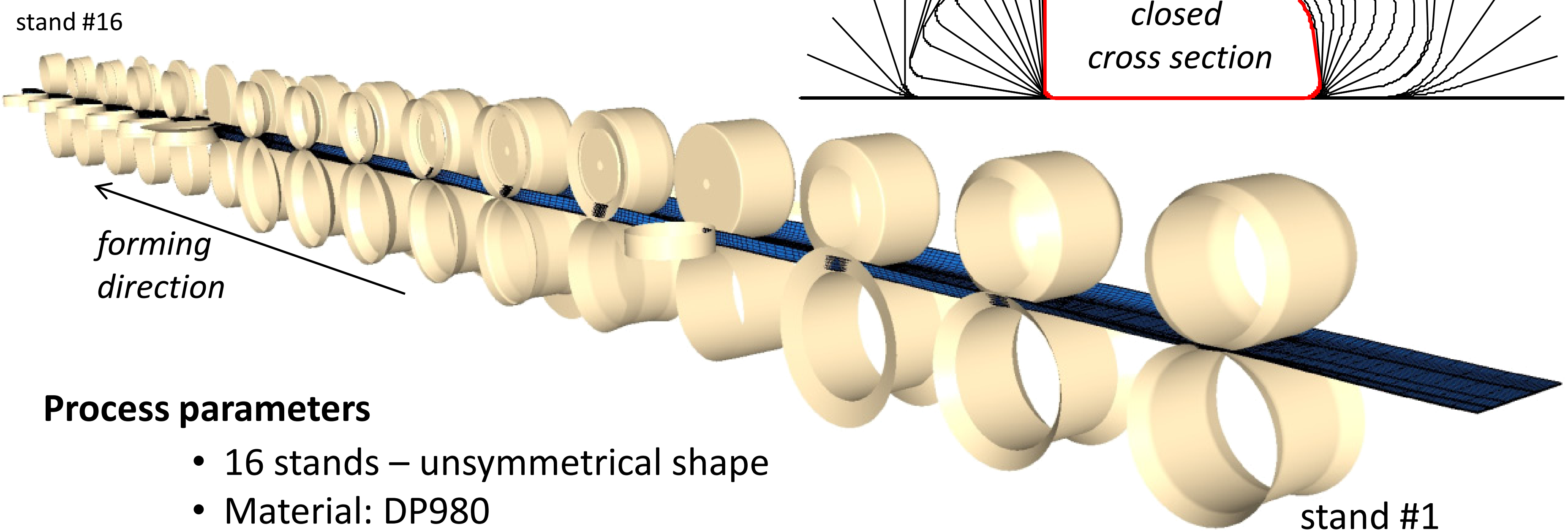
Lagrangian	: (21 320 FEs)	4d 12h 28’
ALE	: (12 768 FEs)	1d 17h 42’

ALE is 2.6x faster!

Forming of a rocker pannel

Simulation of an industrial line





Process parameters

- 16 stands – unsymmetrical shape
- Material: DP980
- Sheet: 5950 x 165 x 1.5 mm

Mesh

- 1 FE through the thickness
- FE length: from 3mm to 30mm
- 155 652 dofs

Forming of a rocker pannel

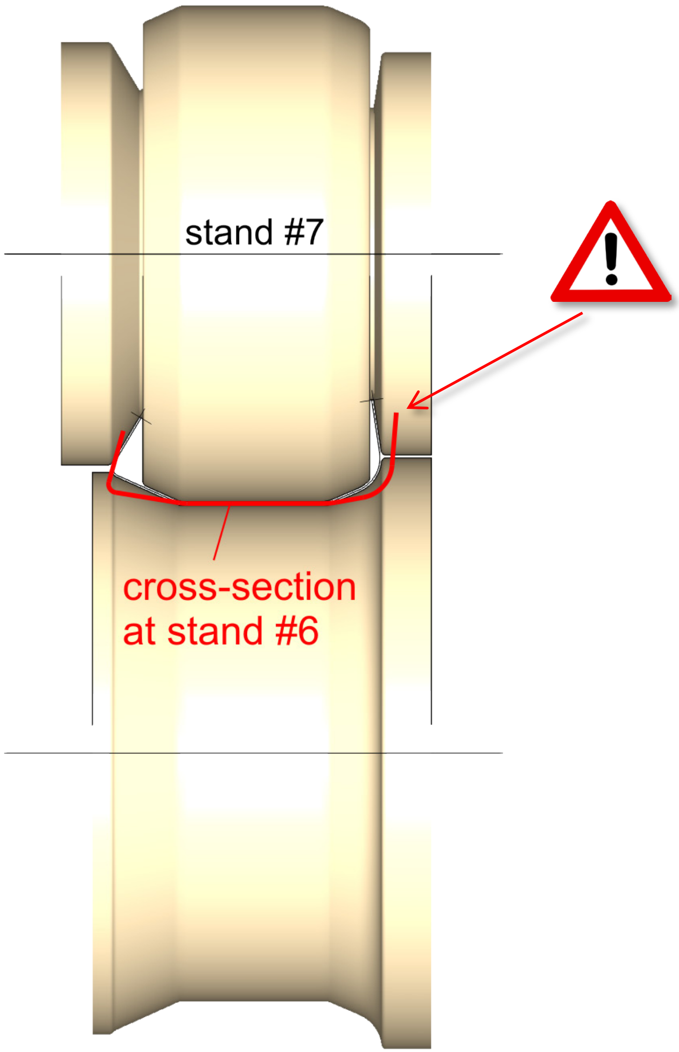
*Lagrangian simulation **FAILS**!*

Main reasons

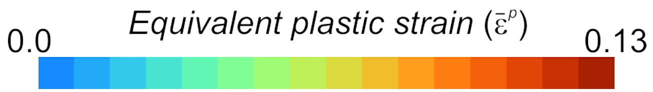
- The sheet cannot enter stand #7 by itself (in the industry the continuous process must be started using additional tools which are not modelled)
- Friction is needed to make the sheet advance through the mill but a uniform constant coefficient is very hard to guess

ALE *results*

- No problem encountered
- 2995 time increments
- CPU time: 3d 19h 01m



Forming of a rocker pannel



Contents

Conclusions

Main contributions of this work

Mesh management

- Efficient 3D surface mesh smoothing method based on a cubic spline surface
- Fast graded mesh smoothing on these surfaces
- Eulerian boundaries

Convection step

- Second order scheme on 3D unstructured meshes of finite elements using more than one Gauss points
- Simple but efficient management of friction forces

Applications

- ALE models of DCET, tension levelling and roll forming
- Systematic and fair comparison with Lagrangian results
- Comparison with experimental data when available

Conclusions

Future work

ALE formalism

ALE + remeshing (2 projects in progress at LTAS-MN^{2}L)

Applications

Unsteady phenomena using quasi Eulerian models (request from the industry)

CPU optimisation

Parallelisation (my current work)

Conclusions

Thank you for your attention