University of Liège
Aerospace & Mechanical Engineering

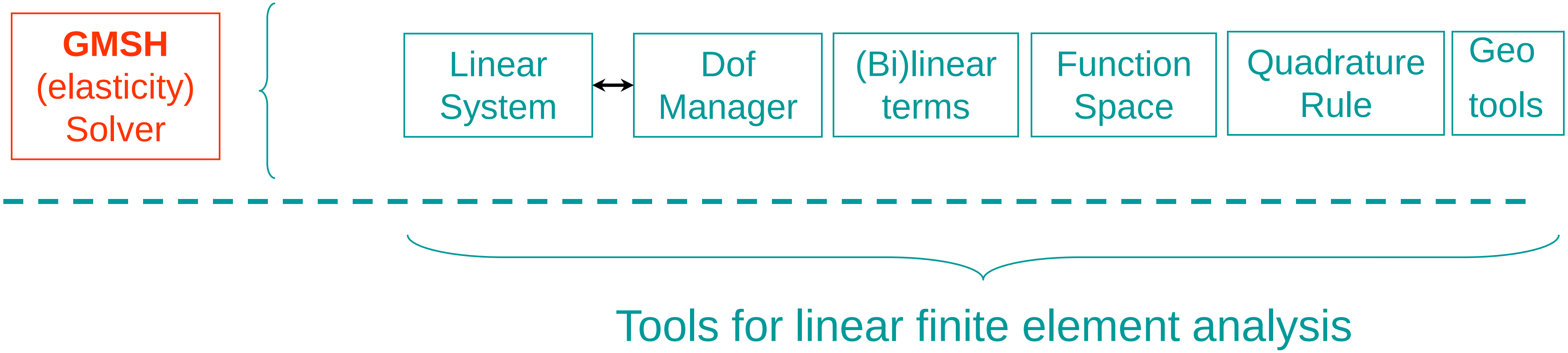
1st Cenaero Workshop
Projects in Fracture Simulations

Authors: G. Becker
L. Wu
V.-D. Nguyen
X. Wang
L. Noels

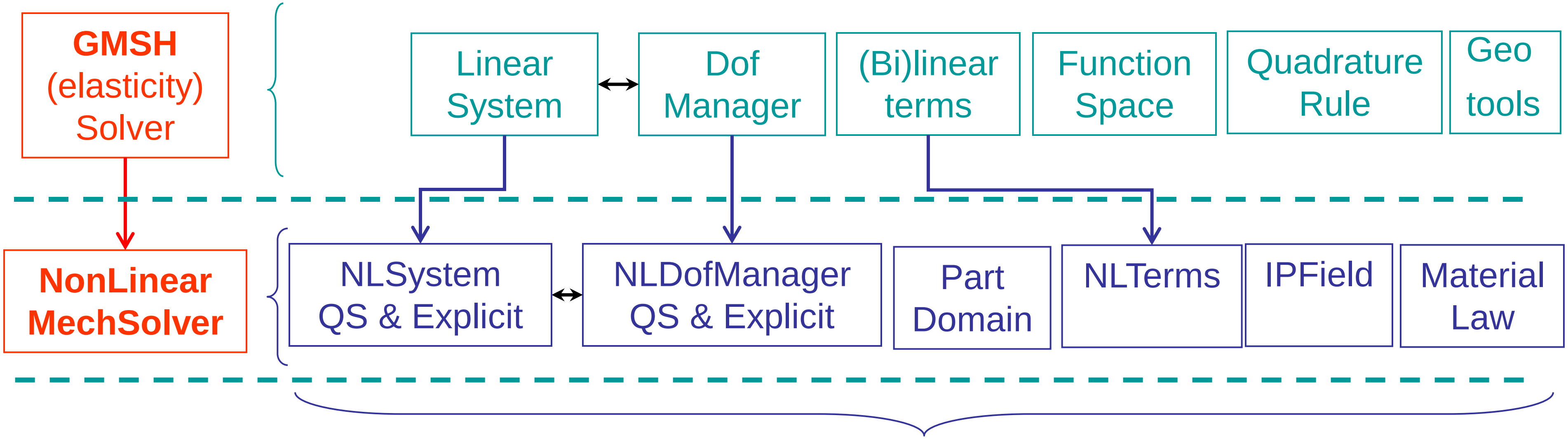
Université de Liège

- Different projects going on
  - Fracture of composites
    - ERA-NET
    - CENAERO, e-Xstream, IMDEA, Tudor
  - Mean-field homogenization with damage
    - ERA-NET
    - CENAERO, e-Xstream, IMDEA, Tudor
  - Fracture or MEMS
    - UCL
  - Fracture of thin structures
    - MS3, GDTech
  - Multiscale
    - ARC
- Different methods
  - Need of common computational tools
  - Need of maximum flexibility

Flowchart


GMSH
(elasticity)
Solver
Linear System
Dof Manager
(Bi)linear terms
Function Space
Quadrature Rule
Geo tools
Tools for linear finite element analysis

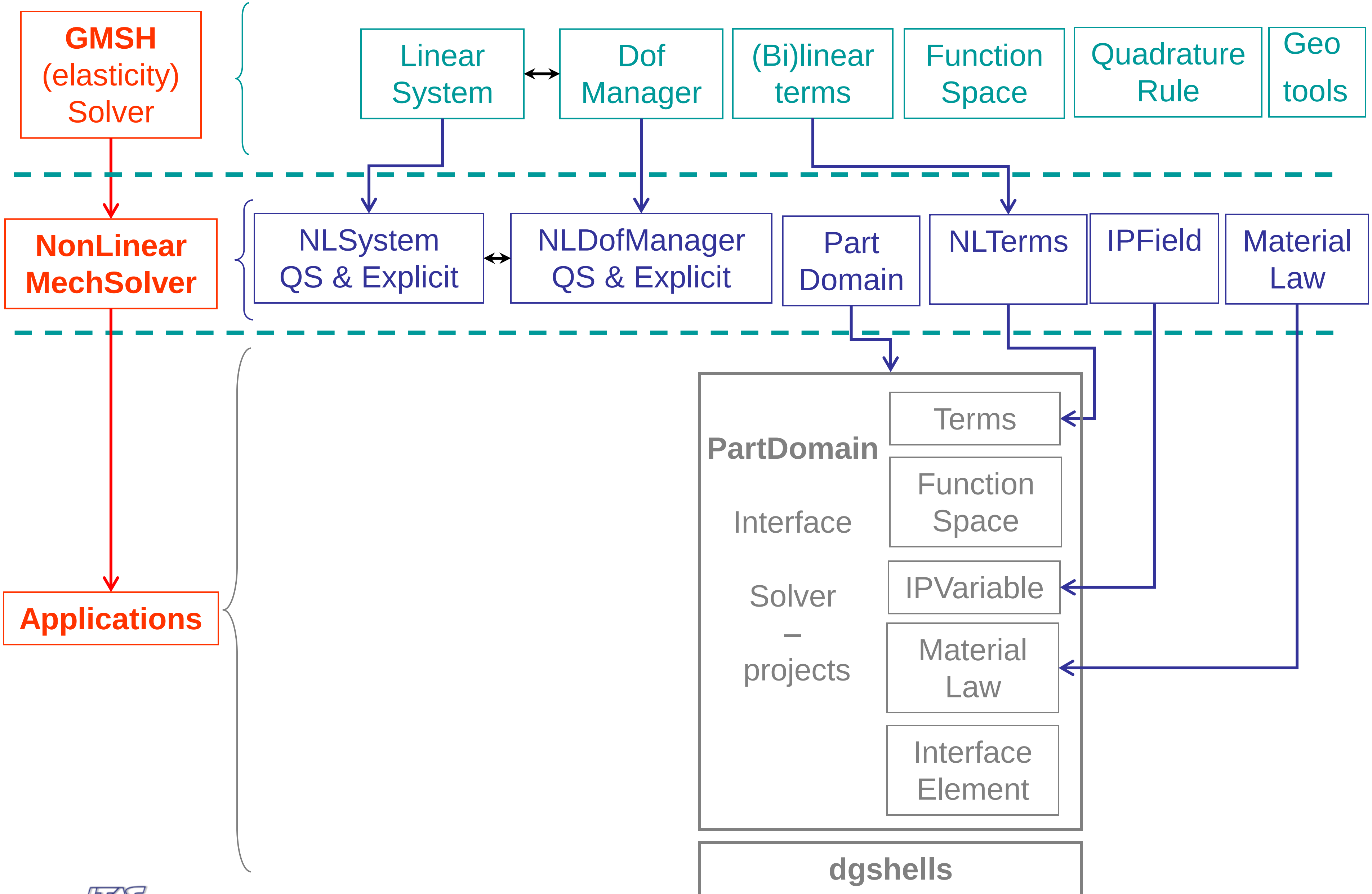
Flowchart



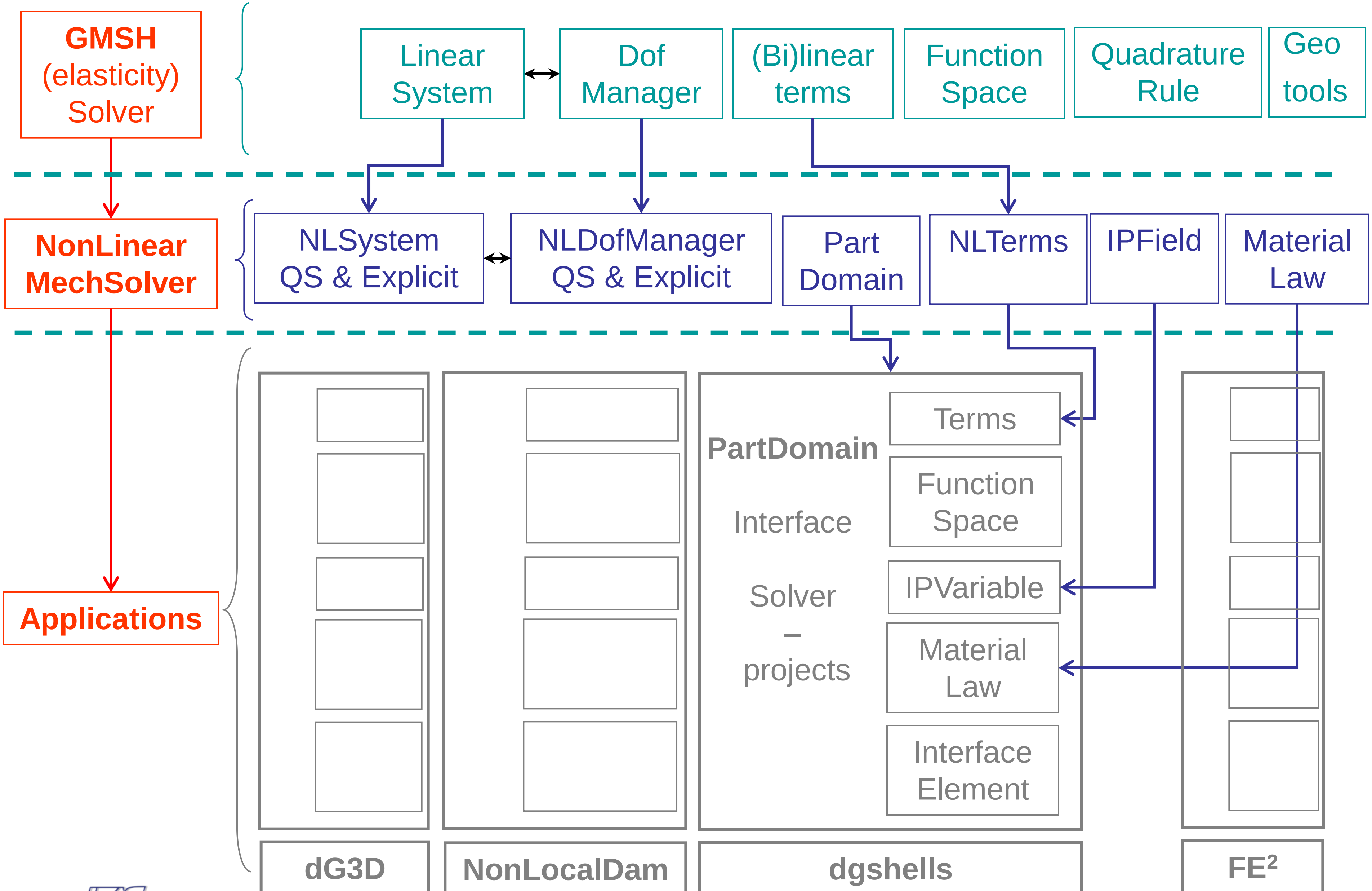
Structure for non-linear finite element analysis

- Pure virtual classes
- Definition of classical material laws
- Time integration
- Parallel implementation
- …

Flowchart


GMSH
(elasticity)
Solver
Linear System
Dof Manager
(Bi)linear terms
Function Space
Quadrature Rule
Geo tools
NonLinear MechSolver
NLSystem QS & Explicit
NLDofManager QS & Explicit
Part Domain
NLTerms
IPField
Material Law
Applications
PartDomain
Interface
Solver
–
projects
Terms
Function Space
IPVariable
Material Law
Interface Element
dgshells

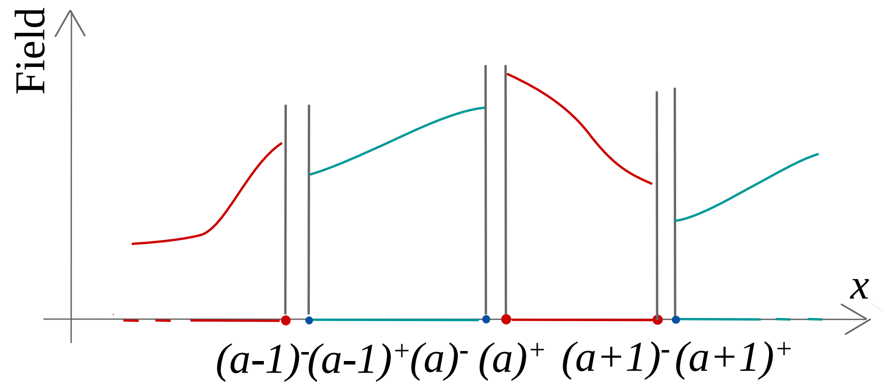
Flowchart



GMSH (elasticity) Solver
Linear System
Dof Manager
(Bi)linear terms
Function Space
Quadrature Rule
Geo tools
NonLinear MechSolver
NLSystem QS & Explicit
NLDofManager QS & Explicit
Part Domain
NLTerms
IPField
Material Law
Applications
PartDomain
Interface
Solver – projects
Terms
Function Space
IPVariable
Material Law
Interface Element
dG3D
NonLocalDam
dgshells
FE²

Interface for dG3D

- Discontinuous Galerkin formulation
  - Finite-element discretization
  - Same **discontinuous** polynomial approximations for the
    - **Test** functions φ_h and
    - **Trial** functions $\delta\varphi$

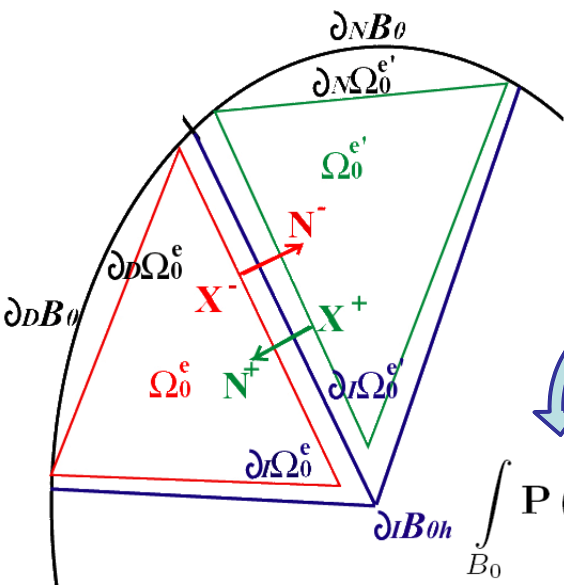


  - Definition of operators on the interface trace:
    - **Jump** operator: $[\![\bullet]\!] = \bullet^+ - \bullet^-$
    - **Mean** operator: $\langle\bullet\rangle = \frac{\bullet^+ + \bullet^-}{2}$
  - Continuity is weakly enforced, such that the method
    - Is consistent
    - Is stable
    - Has the optimal convergence rate

- Discontinuous Galerkin formulation
  - // & fracture
  - Formulation in terms of the first Piola stress tensor $\mathbf{P}$

$$\nabla_0 \cdot \mathbf{P}^T = 0 \text{ in } \Omega \quad \& \quad \begin{cases} \mathbf{P} \cdot \boldsymbol{N} = \bar{\boldsymbol{T}} \text{ on } \partial_N \Omega \\ \varphi_h = \bar{\varphi}_h \text{ on } \partial_D B \end{cases}$$

  - Weak formulation obtained by integration by parts **on each element** Ω^e



$$\sum_e \int_{\Omega_0^e} \nabla_0 \cdot \mathbf{P}^T(\varphi_h) \cdot \delta\varphi \, dB = 0$$

$$\sum_e \int_{\Omega_0^e} -\mathbf{P}(\varphi_h) : \nabla_0 \delta\varphi \, dB + \sum_e \int_{\partial\Omega_0^e} \delta\varphi \cdot \mathbf{P}(\varphi_h) \cdot \boldsymbol{N} \, d\partial B = 0$$

$$\int_{B_0} \mathbf{P}(\varphi_h) : \nabla_0 \delta\varphi \, dB + \int_{\partial_I B_0} [\![\delta\varphi \cdot \mathbf{P}(\varphi_h)]\!] \cdot \boldsymbol{N}^- \, d\partial B = \int_{\partial_N B_0} \bar{\boldsymbol{T}} \cdot \delta\varphi \, d\partial B$$

New interface terms

- Interface term rewritten as the sum of 3 terms
  - Introduction of the numerical flux $\boldsymbol{h}$

$$\int_{\partial_I B_0} [\![\delta\boldsymbol{\varphi} \cdot \mathbf{P}(\boldsymbol{\varphi}_h)]\!] \cdot \boldsymbol{N}^- \, d\partial B \rightarrow \int_{\partial_I B_0} [\![\delta\boldsymbol{\varphi}]\!] \cdot \boldsymbol{h}\left(\mathbf{P}^+, \mathbf{P}^-, \boldsymbol{N}^-\right) \, d\partial B$$

    - Has to be consistent:
$$\begin{cases} \boldsymbol{h}\left(\mathbf{P}^+, \mathbf{P}^-, \boldsymbol{N}^-\right) = -\boldsymbol{h}\left(\mathbf{P}^-, \mathbf{P}^+, \boldsymbol{N}^+\right) \\ \boldsymbol{h}\left(\mathbf{P}_{\text{exact}}, \mathbf{P}_{\text{exact}}, \boldsymbol{N}^-\right) = \mathbf{P}_{\text{exact}} \cdot \boldsymbol{N}^- \end{cases}$$
    - One possible choice:
$$\boldsymbol{h}\left(\mathbf{P}^+, \mathbf{P}^-, \boldsymbol{N}^-\right) = \langle \mathbf{P} \rangle \cdot \boldsymbol{N}^-$$
  - Weak enforcement of the compatibility

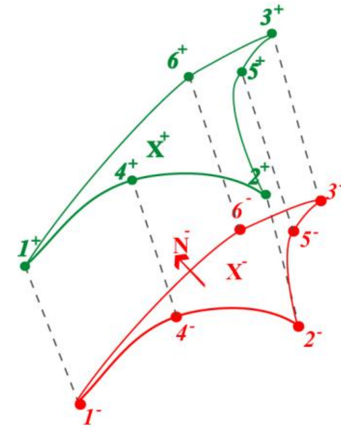
$$\int_{\partial_I B_0} [\![\boldsymbol{\varphi}_h]\!] \cdot \left\langle \frac{\partial \mathbf{P}}{\partial \mathbf{F}} : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi} \right\rangle \cdot \boldsymbol{N}^- \, d\partial B$$

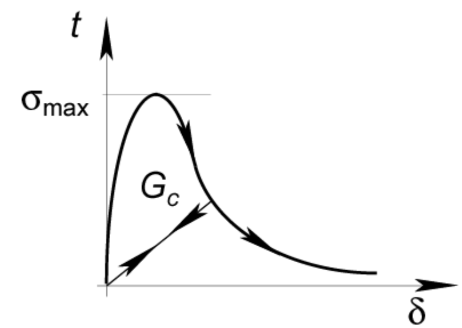
  - Stabilization controlled by parameter β, for all mesh sizes h^s

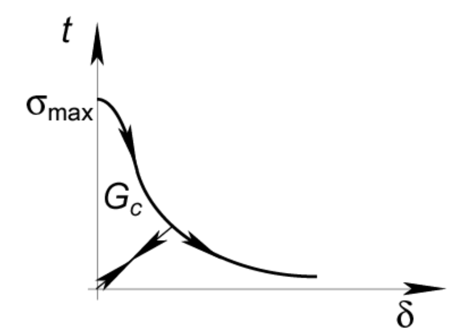
$$\int_{\partial_I B_0} [\![\boldsymbol{\varphi}_h]\!] \otimes \boldsymbol{N}^- : \left\langle \frac{\beta}{h^s} \frac{\partial \mathbf{P}}{\partial \mathbf{F}} \right\rangle : [\![\delta\boldsymbol{\varphi}]\!] \otimes \boldsymbol{N}^- \, d\partial B$$

  - Those terms can also be explicitly derived from a variational formulation (Hu-Washizu-de Veubeke functional) [Noels & Radovitzky, IJNME 2006 & JAM 2006]

- Cohesive Zone Method for fracture
  - Based on the use of cohesive elements
    - Inserted between bulk elements
  - Intrinsic Law
    - Cohesive elements inserted from the beginning
    - Drawbacks:
      - Efficient if a priori knowledge of the crack path
      - Mesh dependency [Xu & Needelman, 1994]
      - Initial slope modifies the effective elastic modulus
      - This slope should tend to infinity [Klein et al. 2001]:
        - » Alteration of a wave propagation
        - » Critical time step is reduced
  - Extrinsic Law
    - Cohesive elements inserted on the fly when failure criterion is verified [Ortiz & Pandolfi 1999]
    - Drawback
      - Complex implementation in 3D (parallelization)

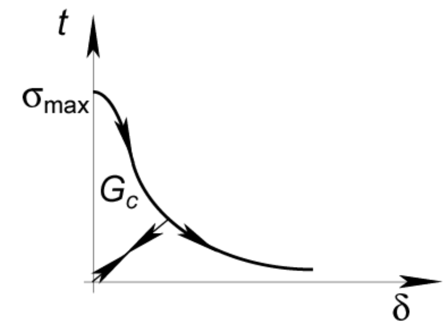


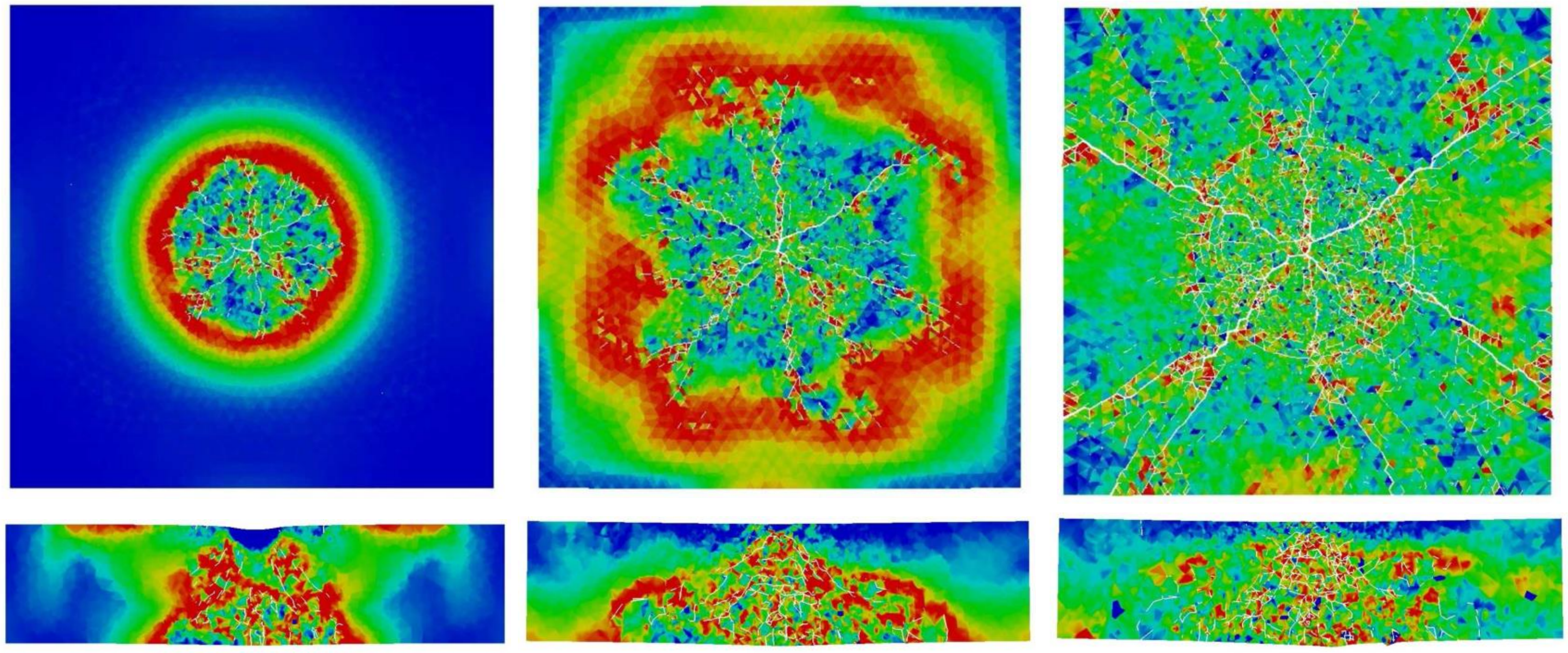




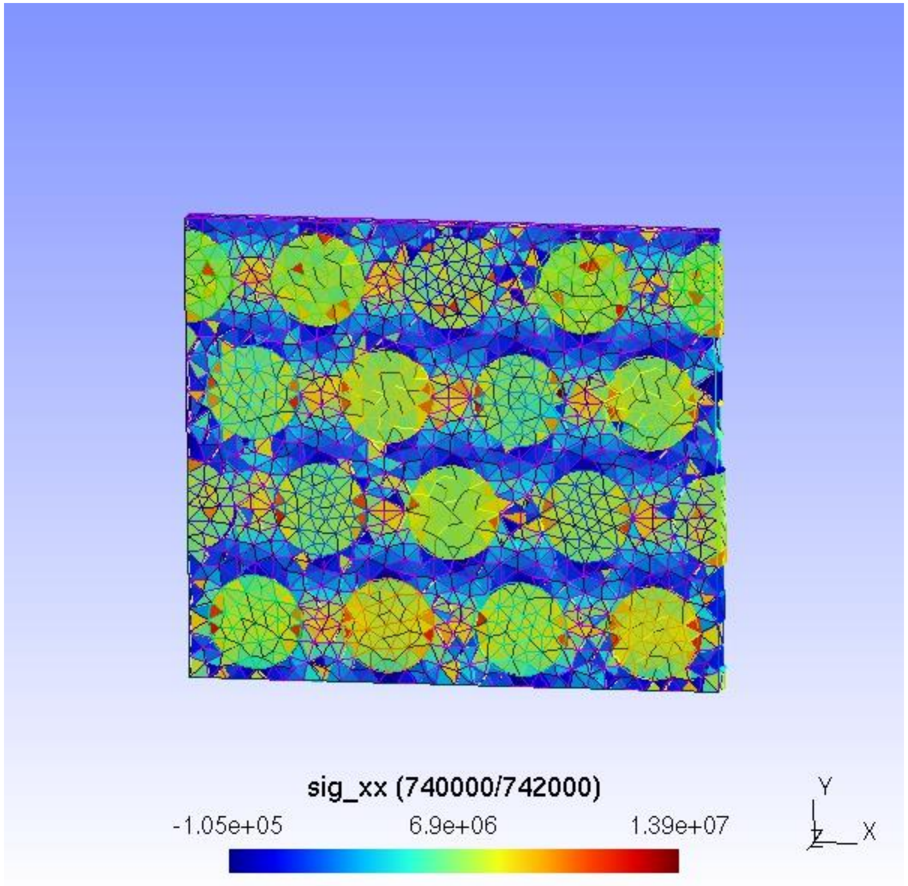
Interface for dG3D

- New DG/extrinsic method [Radovitzky, Seagraves, Tupek, Noels, CMAME 2011]
  - Interface elements inserted from the beginning
  - Interface law initially the DG interface forces
  - Impact of alumina plate





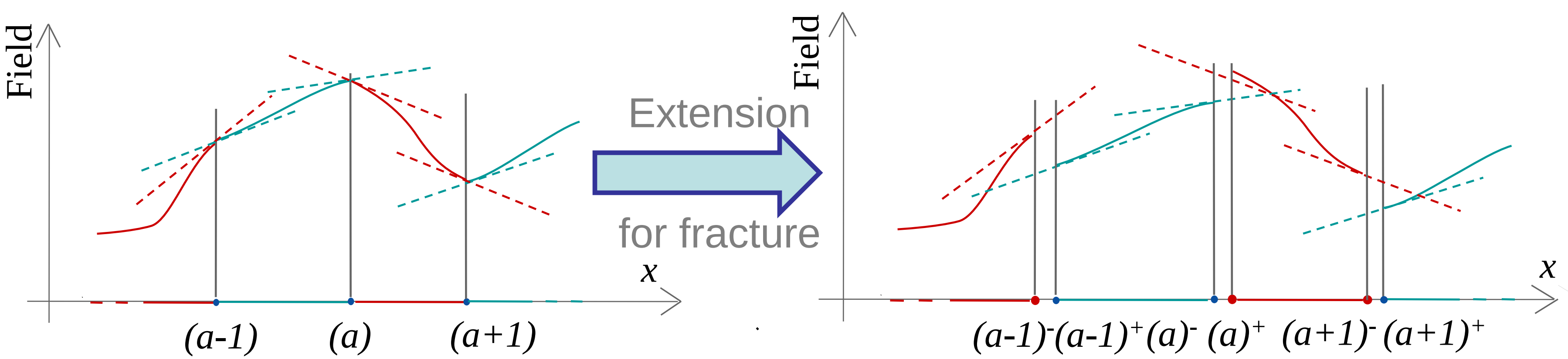
- MATERA project: SIMUCOMP
  - CENAERO, e-Xstream, IMDEA Materials, Tudor, ULg
  - Application to composite
  - Representative nature?
  - First results



- Thin bodies
  - FRIA (MS3, GDTech)
  - C^1 continuity required
  - Test functions

C^0/DG formulation
[Noels & Radovitzky, CMAME 2008]

DG formulation
[Becker & Noels, IJNME 2011]



- New DG interface terms
  - Consistency
  - Compatibility
  - Stability

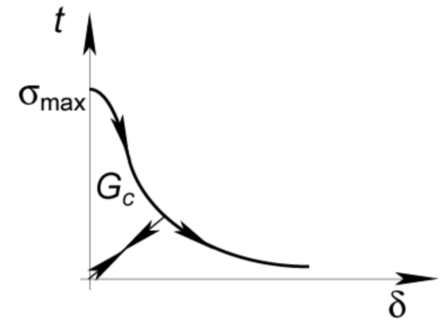
Interface for shells

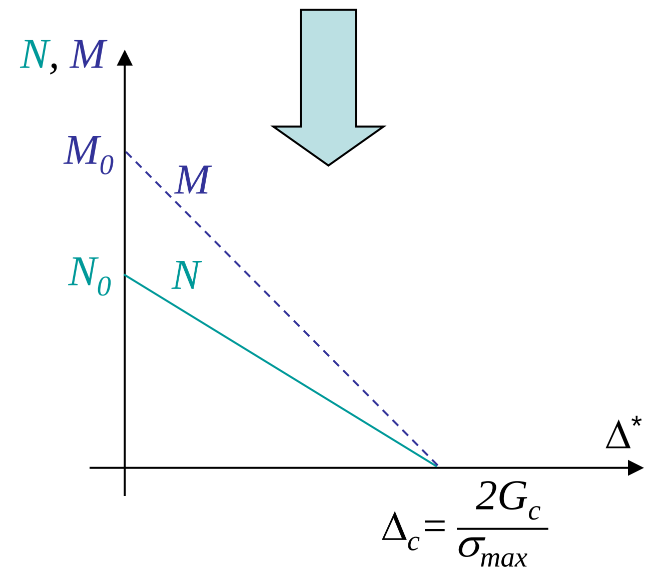
- New cohesive law for thin bodies
  - Should take into account a through the thickness fracture
    - Problem : no element on the thickness
    - Very difficult to separate fractured and not fractured parts
  - Solution:
    - Application of cohesive law on
      - Resultant stress

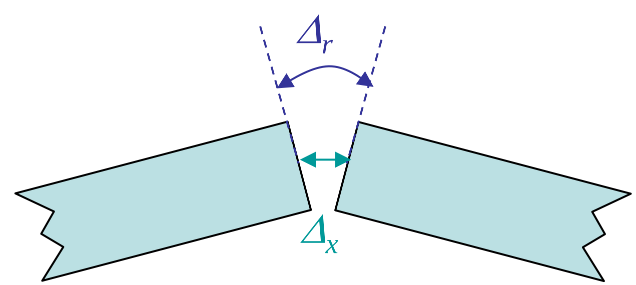
        $n^{11} \Rightarrow N(\Delta^*)$
      - Resultant bending stress

        $\tilde{m}^{11} \Rightarrow M(\Delta^*)$
    - In terms of a resultant opening Δ^*

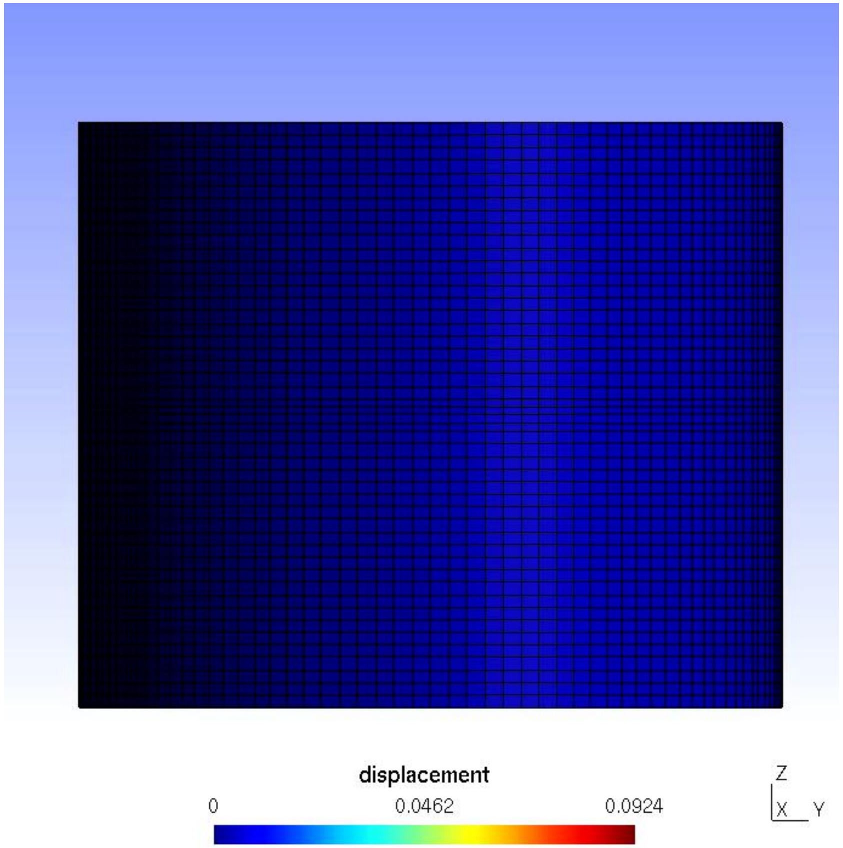
$$\Delta^* = (1-\beta)\Delta_x + \beta\frac{h}{6}\Delta_r$$

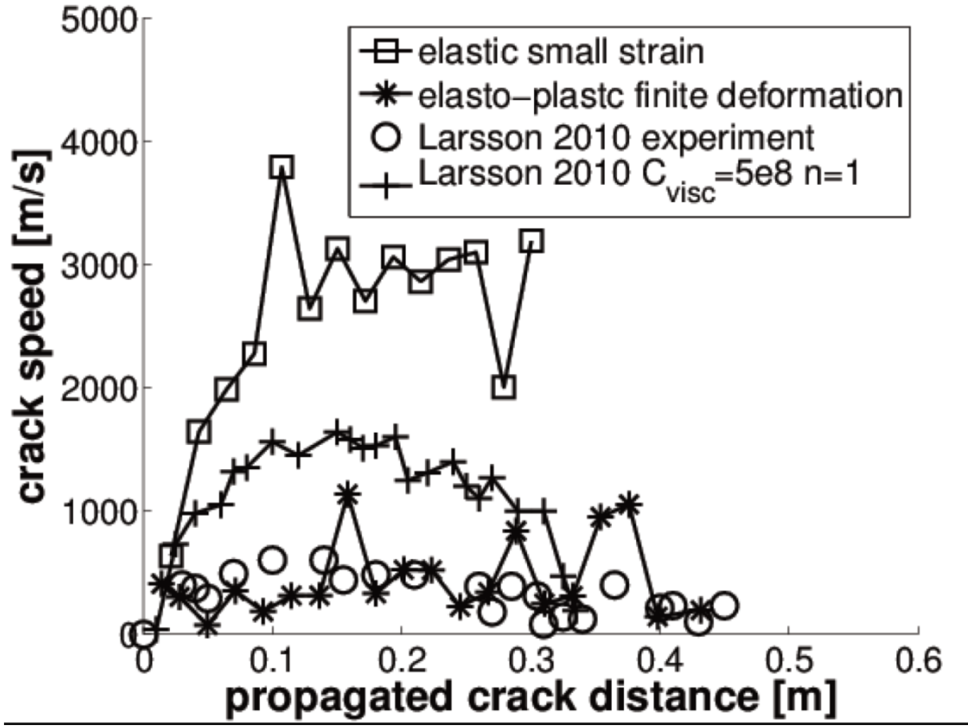






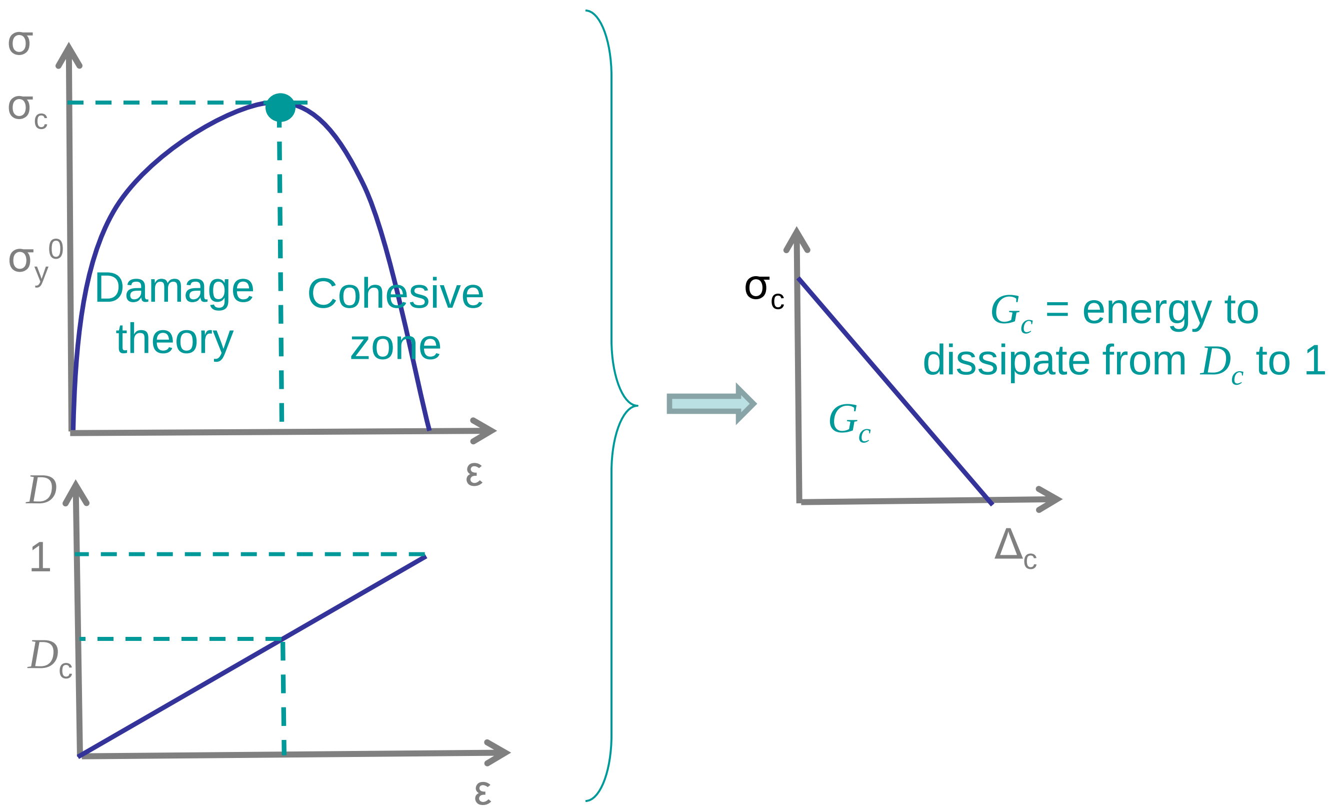
- Application
  - Notched elasto-plastic cylinder submitted to blast



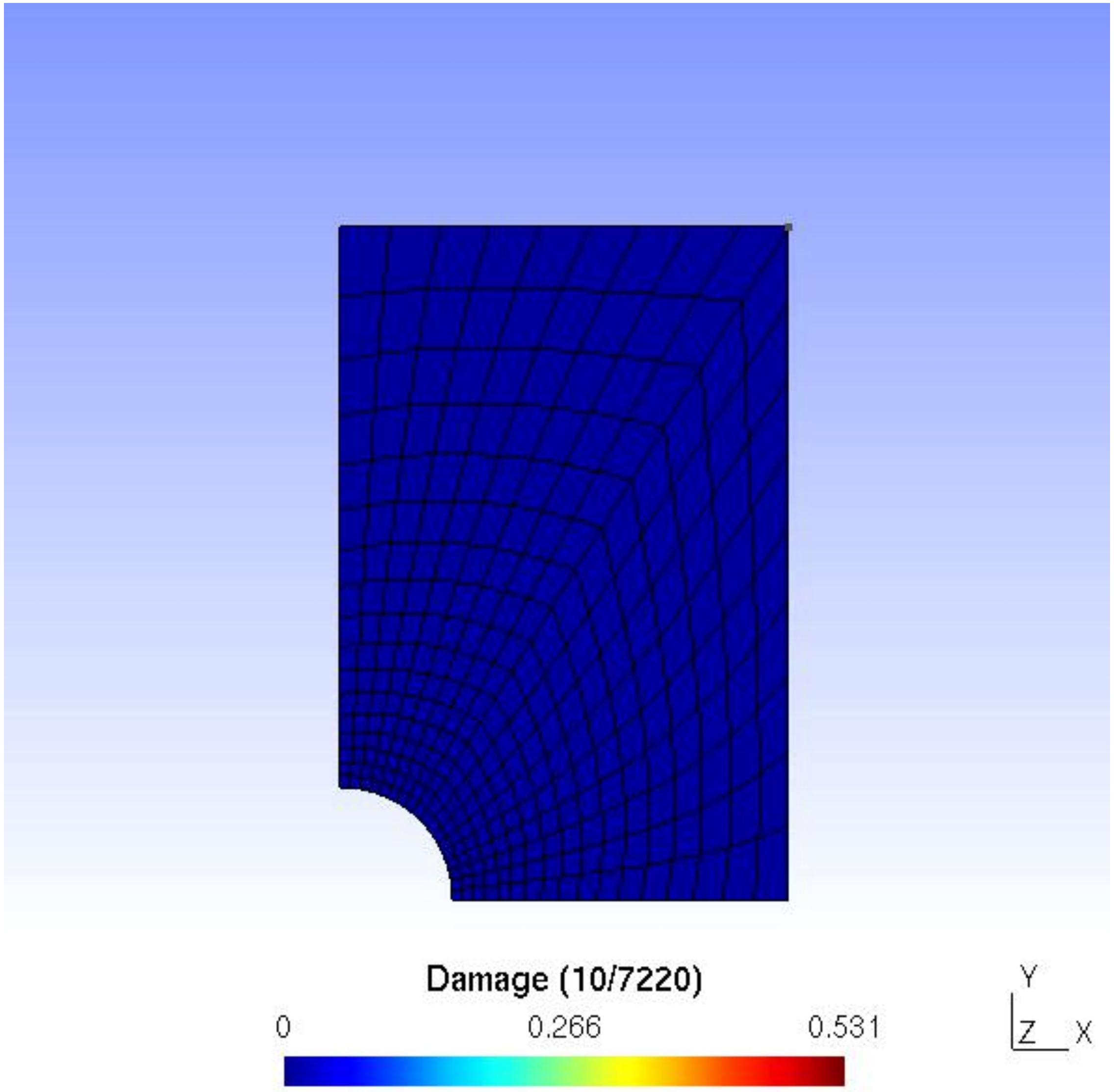


  - Future application: Rupture of MEMS
    - UCL

- Extension to damage
  - Transition from damage to crack
    - At lost of ellipticity or at lost of convergence [Huespe 2009]
    - Fracture energy = Damage energy remaining



- Extension to damage
  - First results
    - Elastic damage
    - Loading too fast!!!



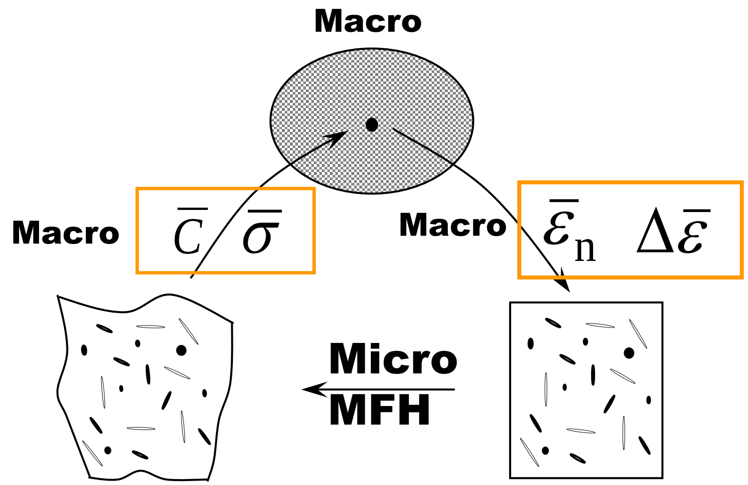
Interface for Non Local Damage

- MATERA project: SIMUCOMP
  - CENAERO, e-Xstream, IMDEA Materials, Tudor, ULg
- Mean Field Homogenization
  - 2-phase composite

$$\langle \boldsymbol{\sigma} \rangle = v_0 \langle \boldsymbol{\sigma} \rangle_{\omega_0} + v_1 \langle \boldsymbol{\sigma} \rangle_{\omega_1}$$

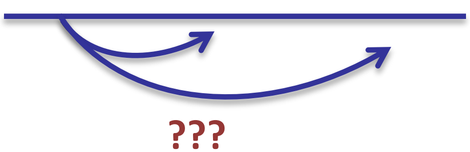
$$\langle \boldsymbol{\sigma} \rangle_{\omega_1} = \bar{\boldsymbol{C}}_1 : \langle \boldsymbol{\varepsilon} \rangle_{\omega_1}$$

$$\langle \boldsymbol{\sigma} \rangle_{\omega_0} = \bar{\boldsymbol{C}}_0 : \langle \boldsymbol{\varepsilon} \rangle_{\omega_0}$$



  - Mori-Tanaka assumption

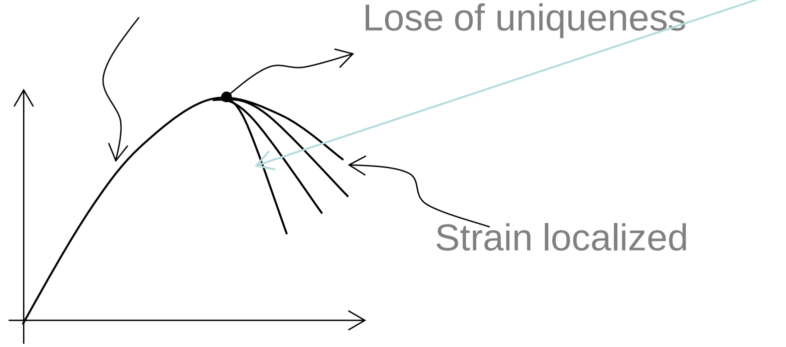
$$\langle \boldsymbol{\varepsilon} \rangle = v_0 \langle \boldsymbol{\varepsilon} \rangle_{\omega_0} + v_1 \langle \boldsymbol{\varepsilon} \rangle_{\omega_1}$$



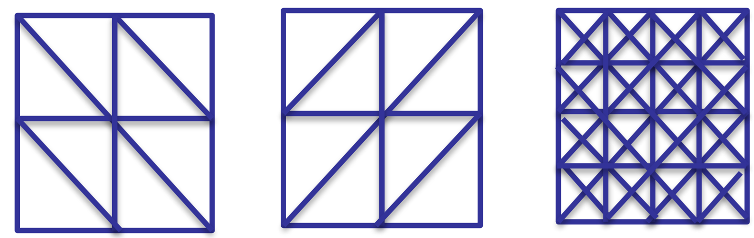
- Extension to damage?

- Damage

Homogenous unique solution



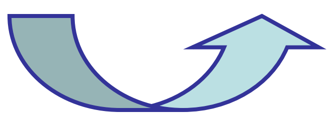
The numerical results change with the size of mesh and direction of mesh



The numerical results change without convergence

- Implicit non-local approach
  - New equation on an internal variable

$$\bar{a} = \frac{1}{V_c}\int_{V_c} a w dV \qquad \bar{a} - c\nabla^2 \bar{a} = a$$



Green function as weight functions w

[Peerlings et al., 1996]

- Non-local damage
  - Lemaitre-chaboche

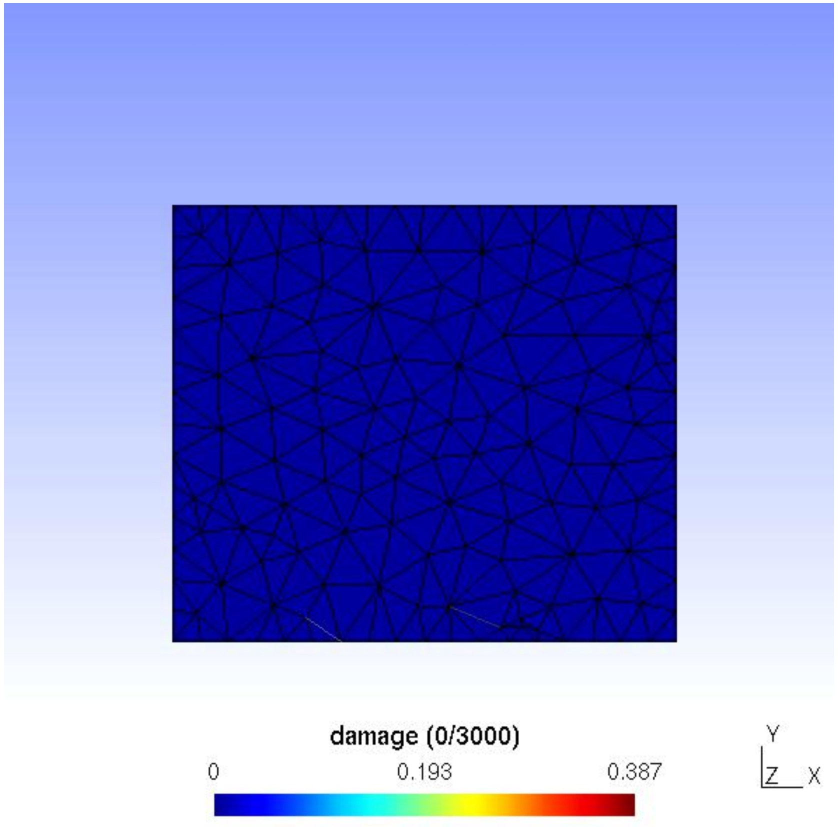
$$\dot{D} = (\frac{Y}{S_0})^n (\dot{p} + c_1 \nabla^2 \dot{p} + c_2 \nabla^4 \dot{p} + \dots)$$
$$= (\frac{Y}{S_0})^n \dot{\bar{p}}$$

- S_0 and n are the material parameters
- Y is the strain energy release rate
- p is the accumulated plastic strain

  - New equation in the system

$$\bar{p} - c\nabla^2 \bar{p} = p$$

$$\Rightarrow \begin{bmatrix} \boldsymbol{K}_{uu} & \boldsymbol{K}_{u\bar{p}} \\ \boldsymbol{K}_{\bar{p}u} & \boldsymbol{K}_{\bar{p}\bar{p}} \end{bmatrix} \begin{bmatrix} d\boldsymbol{u} \\ d\bar{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\text{ext}} - \boldsymbol{F}_{\text{int}} \\ \boldsymbol{F}_{p} - \boldsymbol{F}_{\bar{p}} \end{bmatrix}$$



- MFH with Non-local damage
  - Based on Linear Composite Comparison

$$\delta\boldsymbol{\sigma} = \upsilon_1 \delta\boldsymbol{\sigma}_1 + \upsilon_0 \delta\boldsymbol{\sigma}_0$$

$$\delta\boldsymbol{\sigma}_0 = (1-D)\boldsymbol{C}_0^{\mathrm{alg}} : \delta\boldsymbol{\varepsilon}_0 - \hat{\boldsymbol{\sigma}}_0 \delta D \quad \& \quad \hat{\boldsymbol{\sigma}}_0 = \boldsymbol{\sigma}_0 / (1-D)$$

$$\delta\boldsymbol{\sigma} = \upsilon_1 \boldsymbol{C}_{\mathrm{I}}^{\mathrm{alg}} \delta\boldsymbol{\varepsilon}_{\mathrm{I}} + \upsilon_0 (1-D)\boldsymbol{C}_0^{\mathrm{alg}} : \delta\boldsymbol{\varepsilon}_0 - \upsilon_0 \hat{\boldsymbol{\sigma}}_0 \delta D$$

Mori-Tanaka

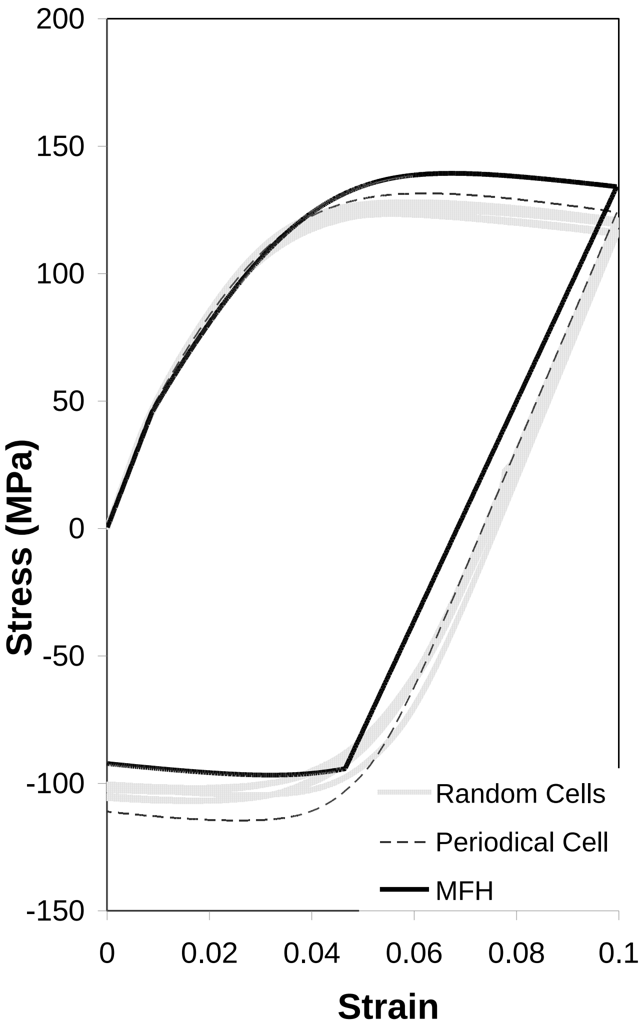
$$\delta\boldsymbol{\sigma} = \overline{\boldsymbol{C}}^{\,\mathrm{alg}\,D} : \delta\boldsymbol{\varepsilon} - \upsilon_0 \hat{\boldsymbol{\sigma}}_0 \frac{\partial D}{\partial \bar{p}} \delta\bar{p}$$

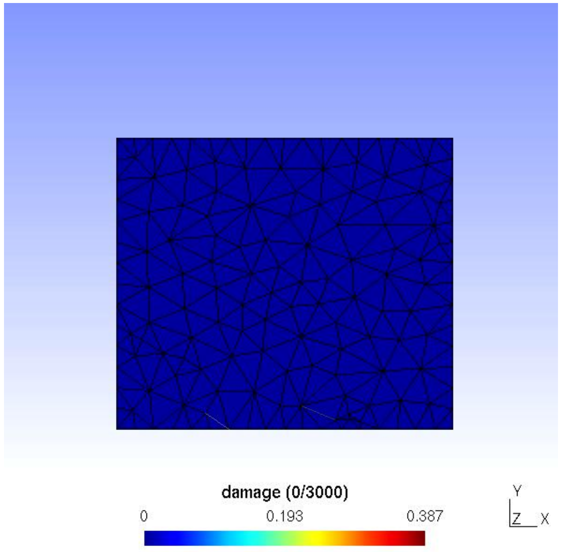
  - Finite elements with 4dofs/node

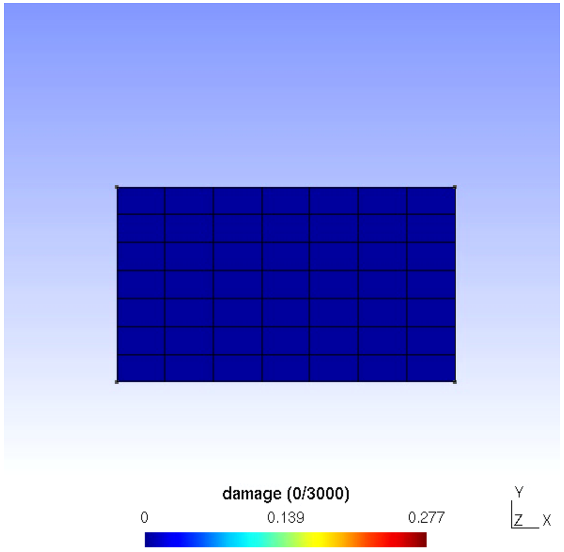
$$\begin{cases} \nabla\boldsymbol{\sigma} + \boldsymbol{f} = \boldsymbol{0} & \text{for homogenized material} \\ \bar{p} - l^2 \nabla^2 \bar{p} = p & \text{related to matrix only} \end{cases}$$

$$\Rightarrow \begin{bmatrix} \boldsymbol{K}_{uu} & \boldsymbol{K}_{u\bar{p}} \\ \boldsymbol{K}_{\bar{p}u} & \boldsymbol{K}_{\bar{p}\bar{p}} \end{bmatrix} \begin{bmatrix} d\boldsymbol{u} \\ d\bar{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\mathrm{ext}} - \boldsymbol{F}_{\mathrm{int}} \\ \boldsymbol{F}_{p} - \boldsymbol{F}_{\bar{p}} \end{bmatrix}$$

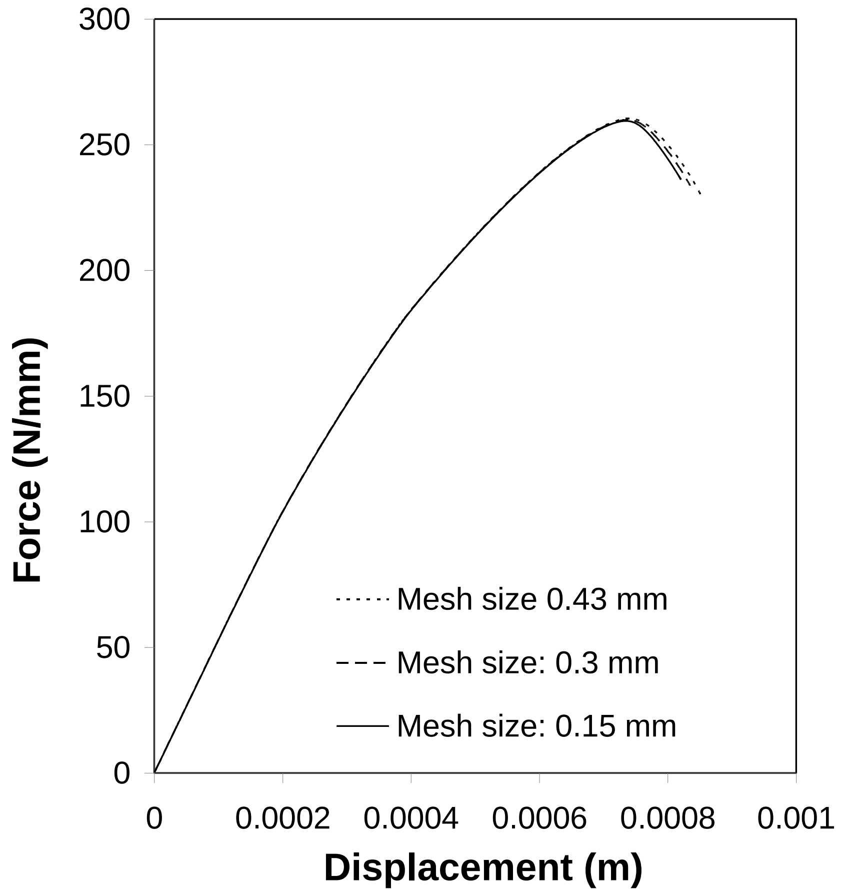
- MFH with Non-local damage
  - Epoxy-CF (30%)

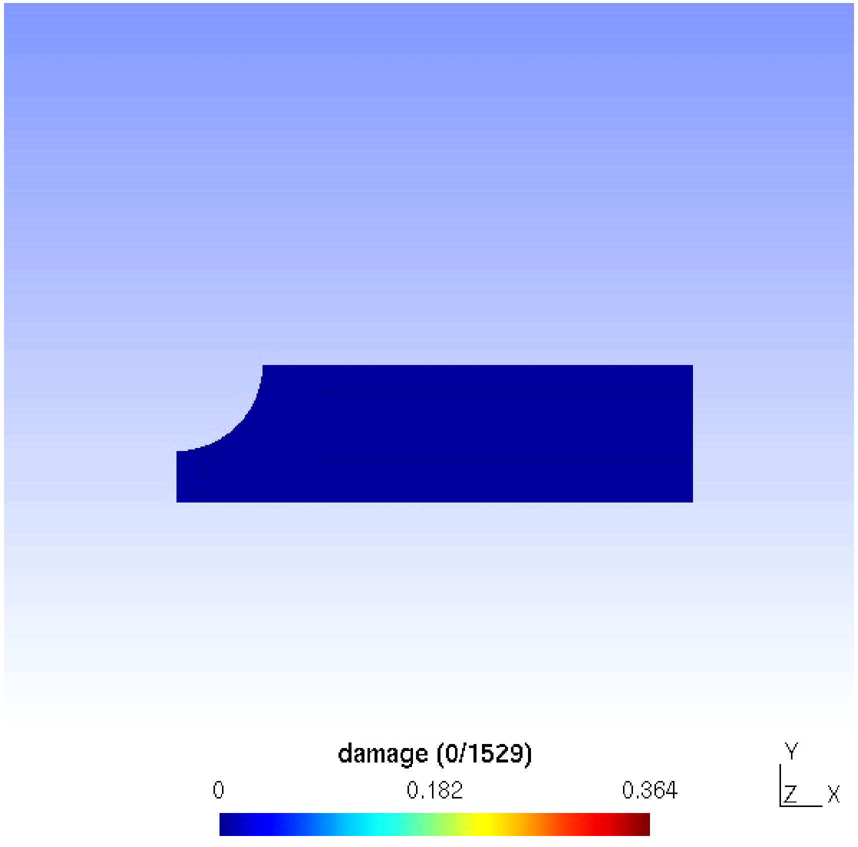




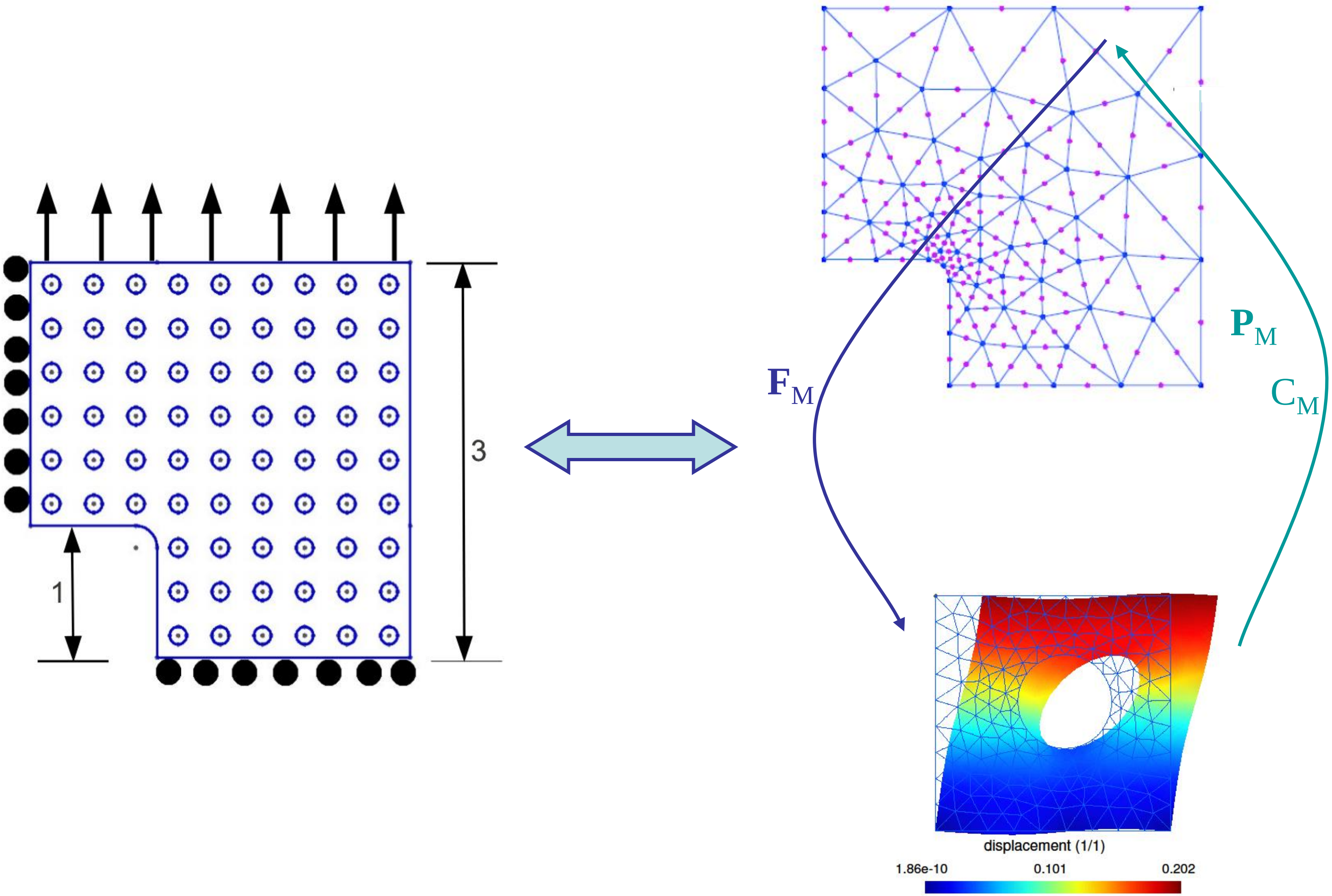


- Application
  - Epoxy-CF (30%)
  - Transverse loading
  - Mesh independent

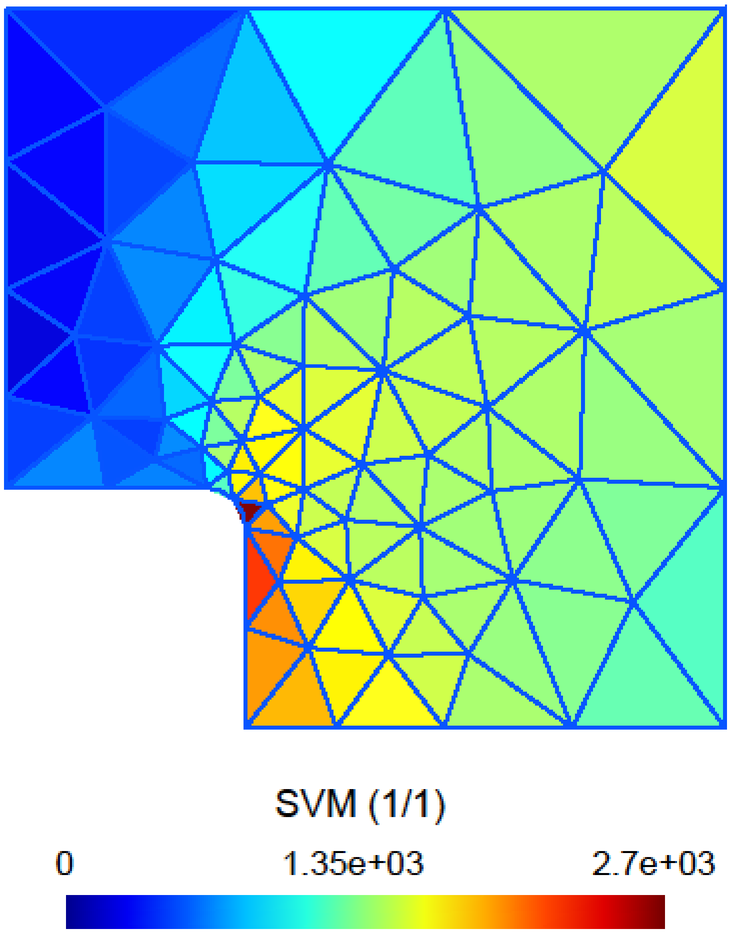


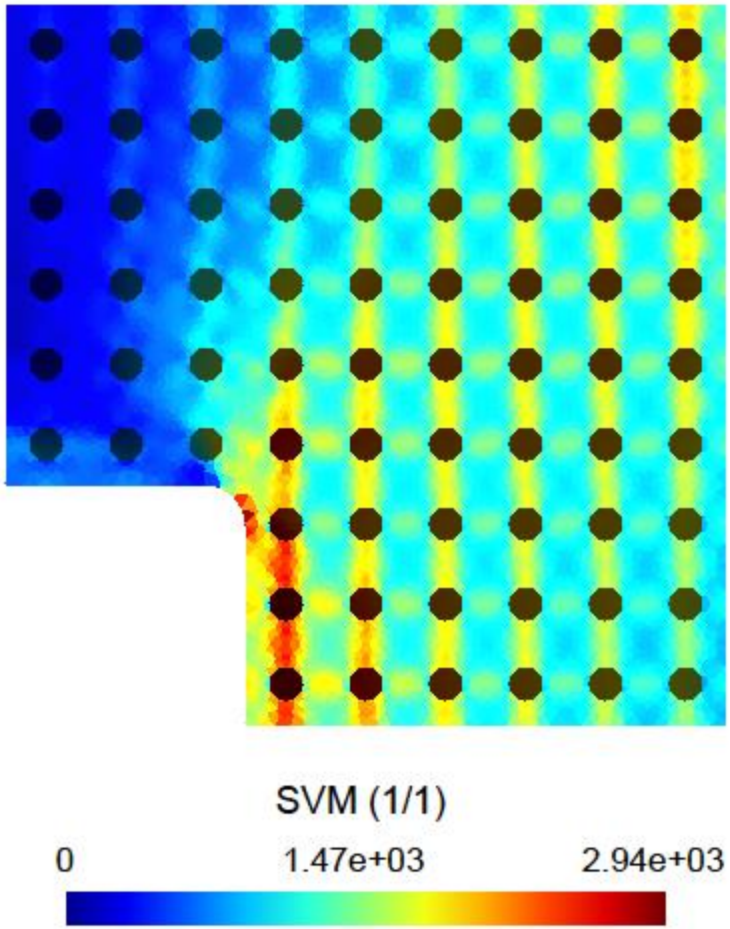


- Computational Multiscale



- Comparison





Conclusions

- NonLinearMechSolver
  - Generic tool to solve mechanic problems
  - // implementation based on DG

- Applications
  - Different projects which include the solver
    - Projects are independent
  - First results
  - More work coming …

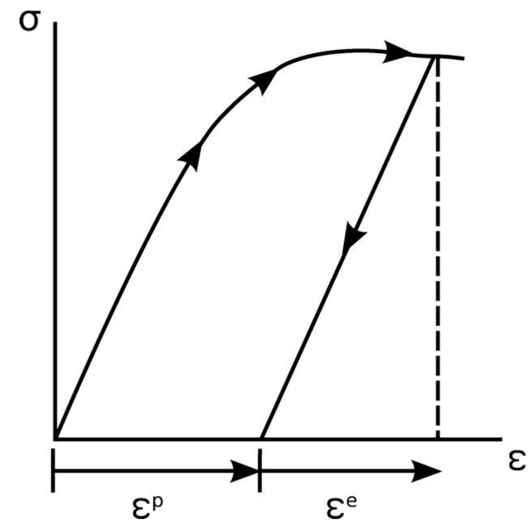
materialLaw

- Material library
  - Mechanics: get stresses from deformations
  - Generic law are defined in nonLinearMechSolver
    - Can be derived in applications (specificities)
  - Basic class

```
class materialLaw
{
 public:
  enum matname{…}
 protected :
  int _num; // law number (must be unique !)
  bool _initialized; // to initialize law
  double _timeStep; // for law which works on increment. (same for all)
  double _currentTime; // time of simulation (same for all)
 public:
  // constructors, destructor, set & get data functions
  virtual void createIPState(IPStateBase* &ips, const bool* state_=NULL,
                                        const MElement* ele=NULL,  const int nbFF_=0) const=0;
};
```

  - IPStateBase allows to save data at integration points

- Non linear laws have to store history
  - Elasto-plastic law
    - Plastic deformation
    - …..



- IPStateBase regroups IPVariables for same point but at different times
  - The contain depends on the law
    - For each law there is a specific IPVariable (inheritance tree are the same)

```
class IPStateBase{
 public:
  // constructor & destructor
  enum whichState{initial, previous, current};
  virtual IPVariable* getState( const whichState
                wst=IPStateBase::current) const=0;
};
```

```
class IPVariable{
 public :
  // constructor & destructor
  virtual double get(const int i) const{
                                        return 0.;}
};
```

Resolution

- Same than (elasticity)Solver
  - dofManager assembles linear and bilinear terms by
    - Linking a Dof to a unique system position
  - The systems are implemented in different formats
    - Taucs, PETSc,Gmm for quasi-statics
    - Blas and PETSc for dynamics (only vector operations)
- In parallel
  - Based on DG method