Morphogenèse et propagation de réseaux complexes de fissures induites par choc thermique
3 April, 2014 – LAMSID, EDF/CNRS/CEA

Corrado Maurini
Institut Jean Le Rond d’Alembert, Equipe Mises UPMC/CNRS (UMR 7190)

In collaboration with: B. Bourdin (LSU), J.-J. Marigo (LMS-X)
P. Sicsic (LMS-X/Lafarge), K. Pham (ENSTA), A.A. Léon Baldelli (LSU)
Introduction

Experimental results from Song et al. J. Am. Ceram. Soc. 2011

Modelling issues:

Distributed loading and cracks, Complex patterns, Crack nucleation, Morphogenesis

Giant’s Causeway
Discrete vs Continuous approaches to Fracture Mechanics

Brittle fracture à la Griffith

- Sharp crack
- Cracks as surfaces of sharp discontinuity of the displacement field

Regularized fracture (damage)

- Localized damage
- Smeared cracks: approximation of cracks by thin bands of localized damage

Issue with sharp crack models à la Griffith

- Complex crack topologies
- Crack nucleation
- Crack path in 3D

Our approach

- a regularized approach (gradient damage)
- a variational formulation
- a quasi-static model

The variational approach gives a mathematical meaning ($\Gamma$-convergence) to the approximation

Localized damage $\leftrightarrow$ Fracture

Ambrosio-Tortorelli (1990); Braides (1998)
Gradient damage model: the energy functional

**Total energy functional**
(linearized isotropic elasticity with isotropic gradient damage)

\[
\mathcal{E}_\ell(u, \alpha) = \frac{1}{2} \int_\Omega A(\alpha) \varepsilon(u) \cdot \varepsilon(u) \, dx + G_c c_w \int_\Omega \left( \frac{w(\alpha)}{\ell} + \ell \nabla \alpha \cdot \nabla \alpha \right) \, dx - (\text{Ext. Work})
\]

where \(1/c_w = 4 \int_0^1 \sqrt{w(\alpha)} d\alpha\) and

- \(\alpha\), a scalar field on \(\Omega\): an internal variable representing the damage field.
- \(A(\alpha) = a(\alpha) A_0\): the damaged elastic tensor (decreasing from \(A(0) = A_0\) to \(A(1) = 0\)).
- \(w(\alpha)\): dissipation for homogeneous damage (increasing from \(w(0) = 0\) to \(w(1) = 1\)).
- \(\ell\): internal length.

**Key requirements**

- stress softening \((w'/a^{−1} \text{ not increasing})\)
- finite elastic limit \((w'(0) > 0)\)
- finite dissipation at full damage \((c_w < \infty)\)

**Adopted model**

\[
w(\alpha) = \alpha, \quad a(\alpha) = (1 - \alpha)^2 \quad c_w = 3/8
\]

Pham, Marigo (CRAS 2010); Pham, Amor, Marigo, Maurini (IJDM 2011); Pham, Marigo, Maurini (JMPS 2011)
Gradient damage model: Variational quasi-static evolution principle

Let be:

- Admissible displacement: \( \mathcal{C}_t = \{ u : u = \bar{u}_t \text{ on } \partial u \Omega \} \)
- Admissible damage: \( \mathcal{D}(\alpha) = \{ \beta : \beta \geq \alpha \text{ on } \Omega \} \)

A quasi-static evolution \( t \rightarrow (u_t, \alpha_t) \) satisfies the following conditions:

- Irreversibility:

  \( \alpha_t \) is not decreasing with the time \( t \)

- Stability:

  At each time \( t \) the state \( (u_t, \alpha_t) \) must be a **unilateral local minimum** of the energy for all admissible variations \( v \in \mathcal{C}_0, \beta \in \mathcal{D}(0) \).

- Energy balance

  Theory of evolution of rate independent systems (Mielke 2005)
Gradient damage model: evolution problem

For evolutions smooth in space and time the three-items principle implies:

- **The equilibrium equations**
  \[ \text{div}\sigma_t = 0 \text{ in } \Omega, \quad \sigma_t n = f_t \text{ on } \partial_f \Omega, \text{ with } \sigma_t = a(\alpha_t) A_0 \varepsilon(u_t) \]  
  (1)

- **The damage criterion defined by the Kuhn-Tucker conditions**
  \[ \text{In } \Omega : \dot{\alpha}_t \geq 0, \quad D \geq 0, \quad D\dot{\alpha}_t = 0, \]  
  (2)
  \[ \text{On } \partial \Omega : \dot{\alpha}_t \geq 0, \quad \frac{\partial \alpha_t}{\partial n} \geq 0, \quad \frac{\partial \alpha_t}{\partial n}\dot{\alpha}_t = 0, \]  
  (3)
  where  
  \[ D(\varepsilon, \alpha, \Delta \alpha) = \frac{a'(\alpha)}{2} A_0 \varepsilon \cdot \varepsilon + c_w G_c \left( \frac{w'(\alpha)}{\ell} - w_1 \ell \Delta \alpha \right) \]

- **Elastic domain:**  
  \[ D(\varepsilon, 0, 0) \geq 0 \quad \Rightarrow \quad A_0^{-1} \sigma \cdot \sigma \leq -\frac{2 c_w G_c w'(0)}{\ell s'(0)}, \quad s(\alpha) = a^{-1}(\alpha) \]

- **Elastic limit in uniaxial traction:**  
  \[ \sigma_c = \sqrt{\frac{2 c_w G_c E w'(0)}{\ell s'(0)}} \quad \Rightarrow \quad \sigma_c = \sqrt{\frac{3 G_c E}{8 \ell}} \]
Example (Traction test)

Illustration: traction of a bar

Imposed displacement, linearized elasticity, brittle isotropic material

\[ U_t = tL \]

\[ \mathcal{E}_\ell (u, \alpha) = \frac{1}{2} \int_\Omega A(\alpha) \varepsilon(u) \cdot \varepsilon(u) \, dx + G_c \left( c_w \int_\Omega \left( \ell \nabla \alpha \cdot \nabla \alpha + \frac{w(\alpha)}{\ell} \right) \, dx \right) \]

Appr. elastic energy

Appr. crack area (dissipated energy)

with

\[ A(\alpha) = A_0 (1 - \alpha)^2, \quad w(\alpha) = \alpha, \quad c_w = 3/8 \]
Illustration: traction of a bar

\[ L = 1, \quad H = 0.1, \quad \ell = 0.05, \quad \text{Element size 0.005, 100 time steps} \]

- Elastic phase (homogeneous damage) \( \Rightarrow \) Loss of stability \( \Rightarrow \) localized solution (fracture)
- The dissipated energy in the localized solution is equal to the crack length \( (G_c = 1) \)
- How the critical load depends on the internal length \( \ell \)? How does the localized solution look like?
Homogeneous response in 1d traction

Stress-displacement diagram
(damage criterion and equilibrium)

\[ \frac{\sigma}{\sigma_M} \]

\[ \frac{U_t}{U_e} \]

Stability diagram
(constrained sign study of the second derivative of the energy)

\[ \frac{L}{\ell} \]

\[ \frac{U_t}{U_e} \]

\[ \sigma_c = \sigma_M = \sqrt{\frac{3G_c E_0}{8 \ell}}, \quad U_e = \sqrt{\frac{3G_c}{8E_0 \ell}} L. \]
Evolution and stability of damage models in the variational framework

Unilateral local minimality

We look for local minima of the damage models under the irreversibility condition.

- **First order condition optimality condition**: evolution problem
- **Second order condition optimality condition**: stability condition

Stability criterion

The state \((u, \alpha)\) is **stable** at time \(t\)

iff

\((u, \alpha)\) is a unilateral local minimum of the total energy \(E_\eta\) on \(C_t \times D\)

\[\exists h > 0 \text{ such that } E_\eta(u + hv, \alpha + h\beta) \geq E_\eta(u, \alpha), \forall (v, \beta) \in C_1 \times D\]

By Taylor expansion

\[E_\eta(u + hv, \alpha + h\beta) - E_\eta(u, \alpha) = hE'_\eta(u, \alpha)(v, \beta) + \frac{1}{2}h^2E''_\eta(u, \alpha)(v, \beta) + \ldots\]

Stability assessed by studying the sign of the second derivative
Localized solutions: fracture as localized damage

Solution with a single fully developed localization inside the bar for \( a(\alpha) = (1 - \alpha)^2 \), \( w(\alpha) = \alpha \)

Damage profile

\[
\alpha(x) = \left(1 - \frac{|x - x_0|}{2\ell}\right)^2
\]

for

\( x \in [x_0 - 2\ell, x_0 + 2\ell] \),

With the choice of \( c_w = \int_0^1 \sqrt{w(\alpha)} d\alpha \), the energy dissipated in this kind of solution is \( G_e \).
Gradient damage as a variational regularization of Griffith

The Griffith energy functional:

\[ E(u, K) = \int_{\Omega \setminus K} \frac{1}{2} A_0 \varepsilon(u) \cdot \varepsilon(u) \, dx + G_c \text{area}(K) \]

where \( K \) is the crack set and \( u \) may jump across cracks.

The energy of the gradient damage model:

\[ E_\ell(u, \alpha) = \frac{1}{2} \int_{\Omega} A(\alpha) \varepsilon(u) \cdot \varepsilon(u) \, dx + G_c \, c_w \int_{\Omega} \left( \ell \nabla \alpha \cdot \nabla \alpha + \frac{w(\alpha)}{\ell} \right) \, dx \]

may be interpreted as a variational regularization of the Griffith model

With suitable choices of the function \( w(\alpha) \) and \( A(\alpha) \) for \( \ell \to 0 \)

\[ E_\ell(u, \alpha) \overset{\Gamma}{\to} E(u, K) \]

(\( \Gamma \)-convergence results: convergence of global minima)

Summary of the key properties of the damage model

- The rate independent evolution is ruled by local minimality, energy balance, and irreversibility.

- There is a (finite) elastic limit stress:

$$\sigma_c = \sqrt{\frac{3GcE}{8\ell}}$$ (used to estimate $\ell$)

- The energy dissipated in a localised solution is $G_c$ (smeared representation of a cracks).

- It is a variational approximation of the Griffith fracture model (in the sense of $\Gamma$-convergence for $\ell \to 0$).

- It obtains (smeared) crack nucleation as loss of stability (local minimality) of elastic solutions.

- It is a phenomenological model (no micro-structural interpretation).
Thermal shock cracks

Experimental results from Song et al. J. Am. Ceram. Soc. 2011

- Ceramics slabs are heated and then quenched in a cold bath to induce a thermal shock on the exposed surfaces
- Periodic cracks are observed
- Crack density increases with the severity $\Delta T$ of the shock
- Crack density decreases with depth
Thermal loading and hypotheses

Hypotheses

- Uncoupled thermo-elasticity
- Thermal effects equivalent to inelastic deformations in the elastic energy
- Cracks do not modify the thermal problem
- Uniform initial temperature $T = T_0$ at $t = 0$
- Dirichlet condition on the temperature
  - $T = T_0 - \Delta T$ on the surface of the thermal shock

- Analytical solution for a semi-infinite slab
  \[ T - T_0 = \Delta T \text{ erfc} \left( \frac{y}{2\sqrt{t}} \right) \]

- For finite slabs we solve numerically the heat equation
  \[ \frac{\partial T}{\partial t} = \kappa \nabla^2 T + \text{B.C.} \]
Damage model with temperature loading

Quasi-static problem where the time is given by the temperature evolution (inertia neglected, evolution principle as before).

Energy functional with inelastic strains induced by the temperature loading

\[
\mathcal{E}_\ell(u, \alpha) = \int_\Omega \left( \frac{(1-\alpha)^2}{2} A_0 (\varepsilon(u) - \beta(T_t - T_0)I)^2 \right) dx + G_c \frac{3}{8} \int_\Omega \left( \frac{\alpha}{\ell} + \ell \nabla \alpha \cdot \nabla \alpha \right) dx
\]

Three characteristic lengths:
- Internal length \( \ell = \frac{3G_cE}{8\sigma_c^2} \rightarrow \text{material} \)
- Typical dimension of the slab \( L \rightarrow \text{geometry} \)
- Griffith length \( \ell_0 = \frac{G_c}{E(\beta \Delta T)^2} = \frac{G_c E}{\sigma_{th}^2} \rightarrow \text{loading} \)

Two dimensionless parameters (and poisson ratio \( \nu \)):
- Dimensions of the slab: \( \frac{L}{\ell} \).
- Mildness of the thermal shock: \( \frac{\ell_0}{\ell} = \frac{8}{3} \left( \frac{\sigma_c}{\sigma_{th}} \right)^2 \) (small for severe shocks).
Full scale simulation in 2d

Ceramic slab from Shao et al. 2011 with $\Delta T = 380 \, ^\circ C$

- Material: $E = 340 \, \text{GPa}$, $\nu = 0.22$, $G_c = 42.47 \, \text{Jm}^{-2}$, $\sigma_c = 342.2 \, \text{MPa}$, $\beta = 8 \times 10^{-6} \, \text{K}^{-1}$
- Dimensions: $5L \times L \times 0.1L$, with $L = 10 \, \text{mm}$

\[ \ell = \frac{3G_cE}{8\sigma_c^2} = 46 \, \mu\text{m}, \quad \ell_0 = \frac{G_c}{E\beta^2\Delta T^2} = 14 \, \mu\text{m} \quad \Rightarrow \quad L/\ell = 215 \quad \ell_0/\ell = 0.3 \]

5 millions of elements (mesh size $\sim \ell/5$), 100 time step.
Problem after time-discretization

\[
\min_{u, \alpha \geq \alpha_{i-1}} \left( E_\ell(u, \alpha) = \int_\Omega \left( \frac{((1 - \alpha)^2 + k_{\text{res}})}{2} A_0(\varepsilon(u) - \beta \Delta T_i I)^2 + G_c \frac{3}{8} \left( \frac{\alpha}{\ell} + \ell \nabla \alpha \cdot \nabla \alpha \right) dx \right) \right)
\]

Alternate Minimization

Key numerical parameters: \( \ell \), the mesh size, \( \Delta T \), \( k_{\text{res}} \), \( tol \)

---

0: \( (u_{\text{old}}, \alpha_{\text{old}}) = (u_{i-1}, \alpha_{i-1}) \) (initialization)

1: \( u_{\text{new}} = \arg \min_u E_\ell(u, \alpha_{\text{old}}) \) (linear solver)

2: \( \alpha_{\text{new}} = \arg \min_{\alpha \geq \alpha_{i-1}} E_\ell(u_{\text{new}}, \alpha) \) (bound constrained linear solver)

3: if \( ||\alpha_{\text{new}} - \alpha_{\text{old}}||_\infty > tol \) repeat 1-3 (convergence test)

4: \( (u_i, \alpha_i) = (u_{\text{new}}, \alpha_{\text{new}}) \) (update)

---

- The step (2) implies a \textit{bound-constrained} minimization of a quadratic functional of \( \alpha \).
- In the numerical work we perform \textbf{local minimization} (\( \Gamma \) convergence is for global minima!)
- The algorithm converges toward stationary points of the functional (Bourdin 2007)
Remarks on numerical implementation

Finite elements (unstructured, uniform meshes)

- Linear triangles/tetrahedra, $u$ and $\alpha$ as nodal dof.
- Structured non-uniform meshes may favour specific local minima and lead to anisotropic non-homogeneous fracture energies (Negri 1999).
- The mesh size $h$ must be smaller than $\ell$ (e.g. $h = \ell/4$).
- Large-scale computations!. Typically millions of dofs

Numerical implementation (homemade parallel finite-element code based open-source libraries).

- PETSc for parallel linear algebra and iterative solver/preconditioners.
- TAO (Toolkit for Advanced Optimization) for constrained quadratic optimization solvers.
- Two available implementations:
  - Fortran based used for massively parallel simulations
  - FEniCs/python based. Open-source demo codes available at https://bitbucket.org/cmaurini/varfrac_for_cism
Scaling law for crack spacing vs depth

Comparisons of experimental, numerical and semi-analytical results
No fitting parameters

(a) numerics
(b) experiments Shao et al.
Numerics
Semi-analytical scaling law from Bahr et al. 2010
(c)

Full scale computation. **Blue line**: universal scaling law from Griffith (Bahr et al. 2010)

\[
\ell = 46 \, \mu m, \ \ell_0 = 14 \, \mu m, \ G_c = 42 \, J/m^2, \ E = 340 \, GPa, \ \Delta T = 380 \, ^\circ C
\]
Homogenous damage $\Rightarrow$ Damage localisation $\Rightarrow$ Crack propagation with period doubling
For a semi-infinite slab the solution is with uniaxial stress and uniaxial strain

\[ \sigma_{11}(e_1 \otimes e_1) \]
\[ \varepsilon_{22}(e_2 \otimes e_2) \]
\[ \partial \alpha / \partial x_1 = 0 \]
\[ \text{No damage for} \]

\[ \ell_0 = \frac{8 \sigma_c^2}{3 \sigma_{th}^2} > \frac{8}{3} \]

- The semi-analytical results obtained by assuming dependence only w.r.t the depth \((x_2)\).
- The damage is maximum at the surface, never reaching \(\alpha = 1\).
Bifurcation analysis: wavelength, critical time and damage penetration
Comparison of semi-analytical and numerical results for a semi-infinite slab

- Semi-analytical expression of the fundamental solution
- Sign analysis of the second derivative of the energy after a partial Fourier transform
- For \( \ell_0 \ll \ell \), \( \lambda^* \sim \sqrt{\ell \ell_0} \), \( D^* \sim \ell \), \( t^* \sim \ell_0 \ell / k_c \).
Numerical results: 2D/3D transition

Results of 3D simulations where only fully damaged areas ($\alpha > 0.9$) are shown. Slab 5 mm $\times$ 1 mm $\times$ 1 mm with increasing temperature contrasts

(a) $380^\circ$C ($\ell_0 = 0.27\ell$); (b) $480^\circ$C ($\ell_0 = 0.17\ell$); (c) $580^\circ$C ($\ell_0 = 0.12\ell$); (d) $680^\circ$C ($\ell_0 = 0.08\ell$)
Numerical results: an overview of 3D results
Bourdin, Marigo, Maurini, Sicsic PRL 2014

Brick - Thermal shock on the bottom face, free on the boundary

Only fractures ($\alpha > 0.9$) are reported. Colors represent the distance from the bottom surface.
24 millions of elements, 2400 cpu, wall time 6h
Numerical results: an overview of 3D results

Average size $d/\ell_0$ of the cells as a function of the depth $a/\ell_0$

Comparison with the two–dimensional scaling law from Bahr et al. 2010.

Series of 3D simulations on cubic domain of different size $L$ with $\ell_0 = 14 \, \mu m$ and $\ell = L/40$. 

The thermal shock problem

November 14, 2013 25 / 50
Hexagonal cracks in thin-film systems
A.A. Leon-Baldelli's Ph.D. Thesis

2D model of a thin film on a substrate with possible debonding (shaded surfaces)

Numerical results with loading given by a uniform inelastic strain (e.g. temperature)
Numerical implementation: alternate minimization
Algorithm for solving the regularized minimization problem at the loading step $i$

Problem after time-discretization

$$\min_{u, \alpha \geq \alpha_{i-1}} \left( \mathcal{E}_\ell(u, \alpha) = \int_{\Omega} \left( \frac{((1 - \alpha)^2 + k_{\text{res}})}{2} \right) A_0(\varepsilon(u)) \cdot (\varepsilon(u)) + \frac{3}{8} G_c \left( \frac{\alpha}{\ell} + \ell \nabla \alpha \cdot \nabla \alpha \right) \, dx \right)$$

Alternate Minimization
Key numerical parameters: $\ell$, the mesh size, $\Delta T$, $k_{\text{res}}$, $tol$

0: $(u_{old}, \alpha_{old}) = (u_{i-1}, \alpha_{i-1})$ (initialization)

1: $u_{new} = \text{argmin}_u \mathcal{E}_\ell(u, \alpha_{old})$ (linear solver)

2: $\alpha_{new} = \text{argmin}_{\alpha \geq \alpha_{i-1}} \mathcal{E}_\ell(u_{new}, \alpha)$ (bound constrained linear solver)

3: if $||\alpha_{new} - \alpha_{old}||_\infty > tol$ repeat 1-3 (convergence test)

4: $(u_i, \alpha_i) = (u_{new}, \alpha_{new})$ (update)
Remarks on numerical implementation

Finite elements (unstructured, uniform meshes)

- Linear triangles/tetrahedra, $u$ and $\alpha$ as nodal dof.
- Structured non-uniform meshes may favour specific local minima and lead to anisotropic non-homogeneous fracture energies (Negri 1999).
- The mesh size $h$ must be smaller than $\ell$ (e.g. $h = \ell/4$).
- Large-scale computations!. Typically millions of dofs

Numerical implementation (homemade parallel finite-element code based open-source libraries).

- PETSc for parallel linear algebra and iterative solver/preconditioners.
- TAO (Toolkit for Advanced Optimization) for constrained quadratic optimization solvers.
- Two available implementations:
  - Fortran based used for massively parallel simulations
  - FEniCs/python based. Open-source demo codes available at https://bitbucket.org/cmaurini/varfrac_for_cism
The FEniCS Project is a collection of free software with an extensive list of features for automated, efficient solution of differential equations.

- Automated solution of variational problems
- Automated C++ code generation from simple python scripts
- High performance linear algebra (PETSc)
- Computational meshes (Mesh generation, refinement, partitioning using SCOTCH, ParMETIS, CGAL)
- Basic built-in post processing, (parallel) input/output in standard format (pvd, hdf5). Paraview or Visit may be used for advanced post-processing.
FEniCs: automatic code generation and JIT

User program (Python)

Form (UFL)

Matrix

Object code

DOLFIN (assembler)

Python interface

Form compiler (FFC/SFC)

C++ code (UFC)

C++ compiler (GCC)

SWIG
Examples

(Available online at https://bitbucket.org/cmaurini/varfrac_for_cism)

- Linear elasticity
- Variational inequality
- Static damage model
- Quasi-static damage model
Linear Elasticity (\(u\)-problem): variational formulation

The potential energy is

\[
\mathcal{E}(u) = \int_{\Omega} \left( \frac{1}{2} \sigma(\varepsilon(u)) \cdot \varepsilon(u) - f \cdot u \right) \, dx
\]

with \(u \in C_U = \{v \in H^1\Omega : v = U \text{ on } \partial_u\Omega\}\) and

\[
\varepsilon(u) = \frac{1}{2} (\nabla u + \nabla^T u), \quad \sigma(\varepsilon) = E_0 \left( \frac{1}{1 + \nu} \varepsilon + \frac{\nu}{1 - \nu^2} \text{tr}(\varepsilon) I_2 \right)
\]

Problem

Linear Elasticity Find \(u \in C_U\) such that

\[
\mathcal{E}'(u)(v) = \int_{\Omega} (\sigma(\varepsilon(u)) \cdot \varepsilon(v) - f \cdot v) \, dx = 0, \quad \forall v \in C_0
\]
V_u = VectorFunctionSpace(mesh, "CG", 1)  # Linear Lagrange Elements on mesh

u = Function(V_u)

def eps(v):
    return sym(grad(v))

def sigma(v):
    mu = E/(2.0*(1.0 + nu))
    lmbda = E*nu/(1.0 - nu)**2
    return 2.0*mu*(eps(v)) + lmbda*tr(eps(v))*Identity(ndim)

f = Constant((0.,0.))

Energy = 0.5*inner(sigma(u),eps(u))*dx-dot(f,u)*dx

du = TrialFunction(V_u)
v = TestFunction(V_u)

Energy_u = derivative(Energy,u,v)
Energy_du = replace(Energy_u,{u:du})

problem_u = LinearVariationalProblem(lhs(Energy_du), rhs(Energy_du), u, bc_u)
solver_u = LinearVariationalSolver(problem_u)
solver_u.solve()
Linear Elasticity (\(u\)-problem): numerical solution

Mesh

Displacement field (norm) for a traction problem
Given the domain $\Omega = [0, L] \times [0, H]$ and the space

$$
\mathcal{D} = \{ \beta : 0 \leq \beta \leq 1 \text{ in } \Omega, \quad \beta = 0 \text{ on } x_1 = 0, \quad \beta = 1 \text{ on } x_1 = L \}
$$

let us define

$$
\mathcal{E}(\alpha) = \int_{\Omega} \left( \frac{w(\alpha)}{\ell} + \frac{\ell \alpha'^2}{2} \right) dx.
$$

Consider the variational problem consisting in finding $\alpha^* \in \mathcal{D}$ such that

$$
\mathcal{E}'(\alpha^*)(\beta - \alpha^*) \geq 0, \quad \forall \beta \in \mathcal{D}
$$
Variational inequality (α-problem): FEniCs python code

```python
V = FunctionSpace(mesh, "CG", 1)  # Scalar Linear Lagrange Elements
alpha = Function(V)
dalpha = TrialFunction(V)
beta = TestFunction(V)

def w(alpha):
    return alpha

F = (ell/2.*inner(grad(alpha), grad(alpha)) + w(alpha)/ell)*dx
dF = derivative(F, alpha, beta)
ddF = derivative(dF, alpha, dalpha)

ub = interpolate(Constant(1.), V)
lb = interpolate(Constant(0.), V)

problem_nl = NonlinearVariationalProblem(dF, alpha, bc, ddF)
solver_nl = NonlinearVariationalSolver(problem_nl)

snes_solver_parameters_bounds = {
    "nonlinear_solver": "snes", "linear_solver": "lu",
    "snes_solver": {
        "maximum_iterations": 100, "method": "virs",
        "absolute_tolerance": 1e-6}
}

solver_nl.parameters.update(snes_solver_parameters_bounds)
solver_nl.solve(lb.vector(), ub.vector())
```
Variational inequality ($\alpha$-problem): numerical solution

Numerical solution for a bar of length $L = 1$ with $w(\alpha) = \alpha$, $\ell = 0.4/\sqrt{2}$

The analytical result for the length of zone with $\alpha > 0$ is $\sqrt{2}\ell$
Problem (Static problem - first order optimality conditions.)

Given $\alpha_0 \in D_0$ and $t \geq 0$, find $u \in C_t$, $\alpha \in D(\alpha_0)$ such that

\[
\mathcal{E}_t'(u, \alpha)(v, 0) = 0, \quad \forall v \in C_0
\]
\[
\mathcal{E}_t'(u, \alpha)(0, \beta - \alpha) \geq 0, \quad \forall \beta \in D(\alpha_0)
\]

Numerics: Alternate minimization

1. Initialization $\alpha^0 \leftarrow \bar{\alpha}$, $k \leftarrow 0$, $\text{err} = 1$. Set the tolerance $\text{tol}$ and the maximum number of iterations $\text{maxiter}$

2. while ($\text{err} < \text{tol}$) or ($k > \text{maxiter}$)

   1. Find $u^k$ by solving the $u$-problem at fixed $\alpha = \alpha^{k-1}$.
   
   2. Find $\alpha^k$ by solving the $\alpha$-problem at $u = u^k$.
   
   3. Calculate the error $\text{err} \leftarrow \|\alpha^k - \alpha^{k-1}\|_{\infty} > \text{tol}$
   
   4. Updating $k \leftarrow k + 1$, $\alpha^k \leftarrow \alpha^{k-1}$
Part I: define the variational problems

\begin{verbatim}
V_u = VectorFunctionSpace(mesh, "CG", 1)
V_alpha = FunctionSpace(mesh, "CG", 1)

u, du, v = Function(V_u), TrialFunction(V_u), TestFunction(V_u)
alpha, dalpha, beta = Function(V_alpha), TrialFunction(V_alpha), TestFunction(V_alpha)
alpha_0 = Function(V_u), Function(V_alpha)

#Energy and derivatives
elastic_energy = 0.5*inner(sigma(u,alpha),eps(u))*dx
dissipated_energy = Gc/float(c_w)*(w(alpha)/ell + 0.5*ell*dot(grad(alpha),grad(alpha)))*dx
total_energy = elastic_energy + dissipated_energy
E_u = derivative(total_energy,u,v)
E_alpha = derivative(total_energy,alpha,beta)
E_alpha_alpha = derivative(E_alpha,alpha,dalpha)

# Writing tangent u-problem in term of test and trial functions for matrix assembly
E_du = replace(E_u,{u:du})

# Variational problem for the displacement
problem_u = LinearVariationalProblem(lhs(E_du), rhs(E_du), u, bc_u)

# Variational problem for the damage (nonlinear)
problem_alpha_nl = NonlinearVariationalProblem(E_alpha, alpha, bc_alpha, E_alpha_alpha)
lb = Function(interpolate(Constant(0.0),V_alpha)) # lower bound, initialized at 0
ub = Function(interpolate(Constant(1.0),V_alpha)) # upper bound, set to 1
\end{verbatim}
Part II : Solvers and alternate minimization

```python
# Set up the solvers
solver_u = LinearVariationalSolver(problem_u)
solver_alpha = NonlinearVariationalSolver(problem_alpha_nl)
solver_alpha.parameters.update(snes_solver_parameters_bounds)
# Alternate minimization
# Initialization
iter=1; err_alpha=1
alpha_0 = interpolate(Expression("0.0"), V_alpha)  # initial (known) alpha
# Iterations
while err_alpha > toll and iter < maxiter:
    # solve elastic problem
    solver_u.solve()
    # solve damage problem
    solver_alpha.solve(lb.vector(), ub.vector())
    # test error
    alphadiff = alpha.vector().array() - alpha_0.vector().array()
    err_alpha = np.linalg.norm(alphadiff, ord=np.Inf)
    # update iteration
    alpha_0.assign(alpha)
    iter = iter+1
# plot the damage field
plot(alpha)
```
Static damage: numerical solution

Damage field for $\ell = 0.2/\sqrt{2}$, $L = 1$, mesh size = 0.01
Errors and maximal damage value during alternate minimization iterations
Influence of the mesh on the surface energy estimation

Red line: \[ 1 + \frac{3}{8} \frac{h}{\ell} \]
Problem (Time-discrete quasi-static evolution)

Given \( \{t_0 = 0, \ldots, t_i, \ldots, t_n = T\} \), find \( (u_i, \alpha_i) \in C_i \times D(\alpha_i) \) where

\[
C_i = \{v : v = t_i U \text{ on } \partial_D \Omega\}, \quad D(\alpha) = \{\beta : \alpha \leq \beta \leq 1 \text{ dans } \Omega\}
\]

such that

\[
\begin{align*}
\forall (v, \beta) \in C_{i+1} \times D(\alpha_{i-1}), & \quad \exists \bar{h} > 0, \quad \forall h \in [0, \bar{h}], \\
\mathcal{E}_i(u_i + h(v - u_i), \alpha_i + h(\beta - \alpha_i)) & \geq \mathcal{E}_i(u_i, \alpha_i).
\end{align*}
\]

Numerically, we impose at time \( t_{i+1} \) the following first order optimality conditions:

\[
\begin{align*}
\mathcal{E}'_{t_{i+1}}(u, \alpha_i)(v, 0) & = 0, \quad \forall v \in C_0 \\
\mathcal{E}'_{t_{i+1}}(u_i, \alpha)(0, \beta - \alpha_i) & \geq 0, \quad \forall \beta \in D(\alpha_i)
\end{align*}
\]

(and use alternate minimization)
Example: \( w(\alpha) = a, \ a(\alpha) = (1 - \alpha)^2 \)

Analytical results: homogenous response and localised solution

Example (AT1, \( w(\alpha) = a, \ a(\alpha) = (1 - \alpha)^2 \))

\[
\begin{align*}
\sigma_c &= \sqrt{\frac{3G_cE_0}{4\sqrt{2}l}} , \\
t_c &= \frac{\sigma_c}{E_0}L , \\
D_0 &= \sqrt{2}l , \\
\alpha(x) &= \left(1 - \frac{|x - x_i|}{D_0}\right)^2
\end{align*}
\]

\[U_t/\sigma_M\] vs \( U_t/U_e \)

\[0 \quad x_0 - \sqrt{2}l \quad x_0 \quad x_0 + \sqrt{2}l \quad L\]
Quasi-static evolution: numerical solution

\[ \ell = 0.2/\sqrt{2}, \quad L = 1, \quad \text{mesh size} = 0.01 \]

\[ \sigma_c = c \sqrt{\frac{E_0 G_c}{\ell}} \]

\[ t_c = \frac{\sigma_c}{E_0} L \]
Critical load for localization versus internal length

\[ \sigma_c = c \sqrt{\frac{E_0 G_c}{\ell}} \]

Red dashed line: analytical elastic limit. Blue continuous line: numerical results.
Demo codes and further infos

Demo codes available here (for FEniCs 1.3 and dev):
https://bitbucket.org/cmaurini/varfrac_for_cism

We develop a more advanced application on the top of FEniCs
(varfrac_fenics)
Objectives: solve for a generic gradient damage model, include contact and plasticity in the crack lines, include
dynamics, optimised preconditioners and solvers, mixed formulations, thin film problems, etc.
(not yet open access, email to corrado.maurini@upmc.fr)

To download FEniCs: http://fenicsproject.org/download/