Sandia National Laboratories

Exceptional service in the national interest

Working towards a modular, fully-coupled phase field fracture model integrating elasticity, plasticity, and damage

Chiraag Nataraj
Andrew Stershic

Sandia National Laboratories

Sandia National Laboratories
U.S. DEPARTMENT OF ENERGY
NNSA

ASCE EMI 2023, June 6–9, Georgia Tech

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Secu DE-NA0003525.

NNSA

Outline

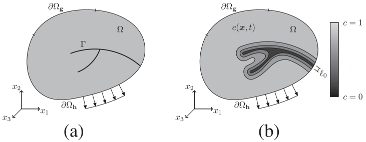
Background

Overview

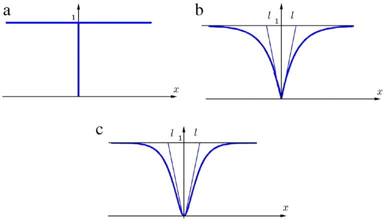
- Phase field methods are a widely used set of techniques used to capture several different types of phenomena
- Initially developed for tracking physical phase changes
- Modified to use a continuous damage variable to track crack propagation in continuum models

Origins of phase field fracture

- Derived from Griffith brittle fracture
- Represent discrete cracks as "smeared" continuum damage models
- Length scale converts infinitesimal crack into finite-width region
- Fracture problem reformulated as coupled PDE system



Borden *et al.* (2014)



Amiri *et al.* (2016)

Phase field fracture models

- Many different phase field fracture models exist
- Different choices for:
  - elastic model
  - inclusion of plasticity (and choice of plastic model)
  - rate-dependence
  - how damage impacts elastic and plastic responses (degradation functions)
  - fracture potential
  - tension-compression split
- Each one has to be derived from scratch based on the assumptions made in that work

Existing formulation within SIERRA

- Existing phase field formulation incorporating plasticity in SIERRA (`PhaseFieldFeFp`) hard-codes many things:
  - elastic, plastic, and fracture models
  - constraints
  - flow rule
  - number of internal variables and phase variables
  - degradation function
  - tension-compression split
- What if we could *generalize* the existing formulation?

Formulation

Guiding Principles

- Potential-based formulation — elastic, plastic, and damage models are all derived from potentials (same for "viscous" parts and kinetic potentials)
- As modular as possible — almost every part has multiple choices (elastic potential, plastic potential, damage potential, degradation function, tension-compression split) and new ones can be implemented as long as they provide the required "bits" (usually: value, jacobian, hessian)
- Generalized — solution method is as general as possible (Sequential Quadratic Programming with Active-Set constraint selection) and almost nothing is assumed about the form of the constraints (can be nonlinear functions of multiple variables)

Derivation

- Start with the usual assumption: $\boldsymbol{F} = \boldsymbol{F}^e \boldsymbol{F}^p$
- Define potentials $\psi^e\left(\boldsymbol{F}\boldsymbol{F}^{p^{-1}}, \boldsymbol{\phi}\right)$, $\psi^p\left(\boldsymbol{F}^p, \boldsymbol{\phi}, \boldsymbol{Q}\right)$, and $\psi^f\left(\boldsymbol{\phi}, \boldsymbol{\nabla}\boldsymbol{\phi}\right)$
- Further, split the $\boldsymbol{\phi}$ dependence of ψ^e and ψ^p into a multiplicative component:

$$\psi^e\left(\boldsymbol{F}\boldsymbol{F}^{p^{-1}}, \boldsymbol{\phi}\right) = g^e\left(\boldsymbol{\phi}\right)\tilde{\psi}^e\left(\boldsymbol{F}\boldsymbol{F}^{p^{-1}}\right)$$

$$\psi^p\left(\boldsymbol{F}^p, \boldsymbol{\phi}, \boldsymbol{Q}\right) = g^p\left(\boldsymbol{\phi}\right)\tilde{\psi}^p\left(\boldsymbol{F}^p, \boldsymbol{Q}\right)$$

- Define kinetic potentials ψ^*_P, ψ^*_Q, and ψ^*_ϕ that define the viscous response of $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\boldsymbol{\phi}$ respectively
- Split the $\boldsymbol{\phi}$ dependence of ψ^*_P and ψ^*_Q as above

Derivation

- Define generalized flow rule as $\dot{\boldsymbol{F}}^p = \dot{\boldsymbol{Q}}\boldsymbol{M}\left(\boldsymbol{K}\right)\boldsymbol{F}^p$
  - This introduces a *vector* of internal variables $\boldsymbol{Q}$ and a *third-order flow tensor* $\boldsymbol{M}$
    - If $\boldsymbol{Q}$ is size n (n internal variables), then $\boldsymbol{M}$ is $n \times 3 \times 3$
  - Reduces to the standard case when there is only one internal variable (outer dimension of $\boldsymbol{M}$ is 1)
- From now on, we deal only with $\boldsymbol{M}$
  - In the case that $\boldsymbol{M}$ is fully defined (i.e. doesn't need to be solved for), can introduce constraints that fix the components of $\boldsymbol{M}$

Derivation

- At each step, we are given $\{\boldsymbol{F}_n, \phi_n, \boldsymbol{F}^p_n, \boldsymbol{Q}_n\}$ as well as $\boldsymbol{F}_{n+1}$
- We need to find $\left\{\boldsymbol{M}_{n+1}, \boldsymbol{Q}_{n+1}, \boldsymbol{F}^p_{n+1}\right\}$
  - We *also* need to solve for ϕ_{n+1}, but that is done separately due to limitations within SIERRA (alternating minimization approach)
- We have to minimize

$$\Delta A\left(\boldsymbol{F}, \phi, \boldsymbol{\nabla}\phi, \boldsymbol{F}^p, \boldsymbol{Q}\right) + \Delta t g^p\left(\phi_{n+\alpha}\right) \tilde{\psi}^*_Q\left(\frac{\boldsymbol{Q}_{n+1} - \boldsymbol{Q}_n}{\Delta t}; \boldsymbol{Q}_{n+\alpha}\right)$$

with respect to $\boldsymbol{Q}_{n+1}$ and $\boldsymbol{M}_{n+1}$, subject to a system of (possibly non-linear) constraints

Derivation

$$\mathcal{L} = g^e\left(\boldsymbol{\phi}_{n+1}\right)\tilde{\psi}^e\left(\boldsymbol{F}^e_{n+1}\right) + g^p\left(\boldsymbol{\phi}_{n+1}\right)\tilde{\psi}^p\left(\boldsymbol{F}^p_{n+1}, \boldsymbol{Q}_{n+1}\right)$$
$$+ \Delta t g^p\left(\boldsymbol{\phi}_{n+\alpha}\right)\tilde{\psi}^*_p\left(\frac{\boldsymbol{Q}_{n+1} - \boldsymbol{Q}_n}{\Delta t}; \boldsymbol{Q}_{n+\alpha}\right) - \lambda_i c_i$$

- Minimize to solve for updated $\boldsymbol{Q}_{n+1}$ and $\boldsymbol{M}_{n+1}$
- Easy to solve if all constraints are equalities (Newton-Raphson or such would work nicely)
- Active-Set methods can be utilized for constraint selection

Derivation

$$\begin{pmatrix} \mathcal{L}_{,\boldsymbol{xx}} & -\boldsymbol{c}_{,\boldsymbol{x}} \\ \boldsymbol{c}_{,\boldsymbol{x}} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{p}_k \\ \boldsymbol{\lambda}_{n+1} \end{pmatrix} = \begin{pmatrix} -F_{,\boldsymbol{x}} \\ -\boldsymbol{c} \end{pmatrix}$$

- We convert it to a quadratic problem and utilize SQP with Active-Set constraint selection
- Derivation is involved but fairly straight-forward
- To use this method, we need the Hessian of the Lagrangian as well as derivatives of the constraint equations

Examples

- Pure elasticity
  - Elastic potential turned on, all other potentials and kinetic potentials disabled, degradation functions turned off
  - $\mathcal{L} = \tilde{\psi}^e \left(\boldsymbol{F}_{n+1} \right)$
- Elasto-plasticity, isotropic plasticity
  - Elastic and plastic potentials turned on, all other potentials and kinetic potentials disabled, degradation functions turned off
  - $\mathcal{L} = \tilde{\psi}^e \left(\boldsymbol{F}_{n+1} \boldsymbol{F}^{p^{-1}}_{n+1} \right) + \tilde{\psi}^p \left(\boldsymbol{F}^p_{n+1}, Q_{n+1} \right) - \lambda_i c_i$, with $M_{ii} = 0$, $M_{ij} M_{ij} = \frac{3}{2}$, $M_{ij} = M_{ji}$, and $Q_{n+1} - Q_n \geq 0$
- Elasto-plasticity, isotropic + kinematic + rate-dependent plasticity with damage
  - All potentials enabled, plastic kinetic potential enabled, all other kinetic potentials disabled, degradation functions turned on
  - $\mathcal{L} = g^e \left(\boldsymbol{\phi}_{n+1} \right) \tilde{\psi}^e \left(\boldsymbol{F}^e_{n+1} \right) + g^p \left(\boldsymbol{\phi}_{n+1} \right) \tilde{\psi}^p \left(\boldsymbol{F}^p_{n+1}, Q_{n+1} \right) + \Delta t g^p \left(\boldsymbol{\phi}_{n+\alpha} \right) \tilde{\psi}^*_p \left(\frac{\Delta Q}{\Delta t}; Q_{n+\alpha} \right) - \lambda_i c_i$ with the same constraints as before

Challenges

- No generalized n-dimensional tensor library available in SIERRA...so created one!
- Finding relevant derivatives of the log strain tensor was highly non-trivial
- Implementing the tension-compression split in a generic way, especially the spectral tension-compression split, was non-trivial
- Numerical sensitivities
- Efficiency concerns

Dealing with the TCS

$$\frac{\mathrm{d}\tilde{\psi}^{e\pm}}{\mathrm{d}F^e_{kl}} = \frac{\partial\tilde{\psi}^{e\pm}}{\partial E^{e\pm}_{uv}} \frac{\partial E^{e\pm}_{uv}}{\partial E^e_{mn}} \frac{\partial E^e_{mn}}{\partial C^e_{pq}} \frac{\partial C^e_{pq}}{\partial F^e_{kl}}$$

- $\frac{\partial E^{e\pm}_{uv}}{\partial E^e_{mn}}$ will be different for each tension-compression split
- Fairly straightforward for the deviatoric-volumetric tension-compression split
- A bit more involved for the spectral tension-compression split

Example problem

- Uses the developed *framework* (`Potential` classes, TCS handling, etc) within the earlier (specialized) formulation
- Does *not* use generalized solver (yet)

Costs and Benefits

Costs

- SQP can be slower than more specialized methods (e.g. return-mapping)
- May need stronger guarantees on potentials or other components than with more specialized solvers (most components need at least C^2)
- Derivations can be tricky

Benefits

- Can implement material models whose constraints are not independent
- Can investigate alternative flow rules
- Converts the existing formulation into a *framework* rather than one specific model
    - As long as requirements for material models are met, can be used within this framework

Conclusions

Current State

- Preliminary implementation of `GeneralizedPhaseFieldFeFp`
- Phase field solver has also been mostly modularized
- Designing correctness and regression tests
- Further optimization of n-dimensional tensor library may be needed
- Comparison of old and new formulations to verify correctness of behavior

Next Steps

- New elastic, plastic, and damage models can be developed, explored, and compared against experimental data to better model materials of interest
- Inclusion of void mechanics and void nucleation, potentially through development of a surrogate model
- Exploration of ways to model crack initiation while being consistent with our variational formulation

Questions?