Isogeometric cohesive interface elements for 2D/3D delamination analysis

Vinh Phu NGUYEN, Pierre KERFRIDEN, Stéphane P.A. BORDAS

Institute of Mechanics and Advanced Materials
Cardiff University, Wales, UK

18 June 2013
Failure of composite laminates
Three levels of observation for composite laminates

- Mesolevel
- Unidirectional ply: orthotropic materials

Failure modes of composite laminates

**delamination**
(interlaminar cracking)

**matrix failure**
(intralaminar cracking)
Computational modeling of delamination

Finite Element Method (FEM)

VCCT (Virtual Crack Closure Technique)
- Linear Elastic Fracture Mechanics
- Only for delamination growth
- Not computationally intensive

Cohesive interface elements
- Cohesive zone models
- Delamination initiation/growth
- Computationally intensive/robustness issue
- Decohesion elements, cohesive zones, cohesive elements...
Cohesive cracks weak form

\[ \bar{\sigma} = \bar{D} \bar{\epsilon} \]

Unknown field is the displacement \( u \)

\[ \int_{\Omega} \delta u \cdot b d\Omega + \int_{\Gamma_t} \delta u \cdot \bar{t} d\Gamma_t = \int_{\Omega} \delta \epsilon : \sigma(u) d\Omega + \int_{\Gamma_d} \delta [u] \cdot t^c([u]) d\Gamma_d \]
Delamination is a problem in which crack path is known in advance.

\[
\int_{\Omega} \delta \mathbf{u} \cdot \mathbf{b} d\Omega + \int_{\Gamma_t} \delta \mathbf{u} \cdot \mathbf{t} d\Gamma_t = \int_{\Omega} \delta \mathbf{\epsilon} : \mathbf{\sigma}(\mathbf{u}) d\Omega + \int_{\Gamma_d} \delta \mathbf{[u]} \cdot \mathbf{t}^c(\mathbf{[u]}) d\Gamma_d
\]

\[
[u]_i = u^+_i - u^-_i
\]

\[
u^+_i = N_1(\xi)u_{i3} + N_2(\xi)u_{i4}
\]

\[
u^-_i = N_1(\xi)u_{i1} + N_2(\xi)u_{i2}
\]
Interface elements: internal force vectors

\[
\begin{align*}
\mathbf{f}^{\text{ext}} &= \mathbf{f}^{\text{int}} + \mathbf{f}^{\text{coh}} \\
\mathbf{f}^{\text{int}} &= \int_\Omega \mathbf{B}^T \mathbf{\sigma} \, d\Omega \\
\mathbf{f}^{\text{ext}} &= \int_{\Gamma_t} \mathbf{N}^T \mathbf{t} \, d\Gamma \\
\mathbf{f}^{\text{coh}}_{ie,+} &= \int_{\Gamma} \mathbf{N}^T_{\text{int}} \mathbf{t}^{c} \, d\Gamma \\
\mathbf{f}^{\text{coh}}_{ie,-} &= -\int_{\Gamma} \mathbf{N}^T_{\text{int}} \mathbf{t}^{c} \, d\Gamma
\end{align*}
\]
Interface elements: tangent matrix

\[ K_e^{\text{int}} = \left[ \int_\Gamma N^TQ^TTQNd\Gamma \right] - \left[ \int_\Gamma N^TQ^TTQNd\Gamma \right] 
\]

Numerical integration
Common interface elements

2D

3D
What is wrong with standard interface elements?

- time consuming pre-processing (generation of interface el.)
- no link to CAD data: not ideal for design-analysis cycles
- standard low order Lagrange elements: poor derivative fields such as stresses => very fine mesh in front of the crack tip
- geometry: not exactly represented
Isogeometric interface elements

- fast pre-processing: automatic generation of interface el.
- link to CAD data: ideal for design-analysis cycles
- high order NURBS elements: highly accurate derivative fields
- less expensive than low order Lagrange elements
- geometry: exactly represented

There are no free lunch. However let talk about the good news first.
Isogeometric analysis
Approximate the unknown fields with the basis functions used to generate the CAD model.

CAD basis functions: B-splines, **NURBS**, T-splines, subdivision surfaces...

- Exact geometry
- High order continuity
- \textit{hpk}-refinement

MA Scott et al, CMAME 2013.
B-splines basis functions

\[ \Xi = \{ \xi_1, \xi_2, \ldots, \xi_{n+p+1} \} \]

knot vector

\[
N_{i,0}(\xi) = \begin{cases} 
1 & \text{if } \xi_i \leq \xi < \xi_{i+1} \\
0 & \text{otherwise}
\end{cases}
\]

\[
N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)
\]

Properties
- Partition of Unity
- Linear independence
- Non-negativity
- \(C^p\) continuity
- Not interpolants

\(p = 2\)

\[ \Xi = \{0, 0, 0, 1, 2, 3, 4, 4, 5, 5, 5\} \]
B-splines

\[ \Xi = \{ \xi_1, \xi_2, \ldots, \xi_{n+p+1} \} \]

\[ \Xi^1 = \{ \xi_1, \xi_2, \ldots, \xi_{n+p+1} \} \]

\[ \Xi^2 = \{ \eta_1, \eta_2, \ldots, \eta_{m+q+1} \} \]

\[ C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) B_i \]

\[ S(\xi, \eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta) B_{ij} \]
Enriching B-splines

Knot insertion (h-refinement) + order elevation (p-refinement) does not change B-splines geometrically/parametrically
Knot insertion to create discontinuities

\[ \Xi = \{0, 0, 0, 1, 1, 1\} \]

\[ \Xi' = \{0, 0, 0, 0.5, 0.5, 0.5, 1, 1, 1\} \]

\[ p = 2 \]

does not change B-splines geometrically/parametrically

knot insertion: stable algorithms, available implementations
Isogeometric cohesive elements

2 quadratic int. elements
[1,2,3,5,6,7]
[2,3,4,6,7,8]

automatically generated using knot insertion.
Isogeometric cohesive elements: advantages

- 2D Mixed mode bending test (MMB)
- 2 x 70 quartic-linear B-spline elements
- run time on a laptop 4GB of RAM: 6 s
- energy arc-length control


3. V.P. Nguyen, P. Kerfriden, S. Bordas. Isogeometric cohesive elements for two and three dimensional composite delamination analysis, 2013, Arxiv.
Examples
**MIGFEM**

- open source Matlab Isogeometric (X)FEM
- 2D/3D solid mechanics with geometry nonlinearities
- 2D XIGA for LEFM and material interfaces
- Structural mechanics: beam, plate, shells (large deformation)

**jem-jive** *(Linux, Mac OS, Windows)*

- commercial C++ toolkit for PDEs
- not a general purpose FE package
- tailor made applications, suitable for researchers
- apps: XFEM, dG, IGA, DEM, FVM etc.
- support parallel computing
- implements useful concepts available in other programming languages--Java, Fortran 90, Matlab and C#
- tensor class: useful to evaluating complex constitutive models
jem-jive: some typical examples

- XFEM
- Non-local damage model
- NURBS/T-splines
- DEM
Isogeometric cohesive elements: 2D example

- 12 layers (1.59 mm)
- Initial notches
- Interface elements
- 2 layers (0.265 mm)
- Length l = 120 mm
- a1 = 40 mm
- a2 = 20 mm
- Thickness 20 mm
Isogeometric cohesive elements: 2D example

Isogeometric cohesive elements: 2D example

- exact geometry by NURBS
- It is straightforward to vary
  (1) number of plies and
  (2) # of interface elements:
- Suitable for parameter studies/design
- Cohesive law: bilinear law of Turon et al. 2006
Isogeometric cohesive elements: 2D example


Isogeometric cohesive elements: 3D example with shells

- Rotation free B-splines shell elements (Kiendl et al. CMAME)
- Two shells, one for each lamina
- Bivariate B-splines cohesive interface elements in between
Isogeometric cohesive elements: 3D examples

- cohesive elements for 3D meshes the same as 2D
- large deformations
- suitable: delamination buckling analysis
Isogeometric cohesive elements

- singly curved thick-wall laminates
- geometry/displacements: NURBS
- trivariate NURBS from NURBS surface
- cohesive surface interface elements
- compression test
Curve offsetting

Curve offsetting

\[ E(B) = \frac{1}{2} \sum_{i=1}^{m} k_s u_i (B)^2 \rightarrow \min \]

\[ B^{(1)} = B^{(0)} - \gamma^{(0)} \nabla E^{(0)} \]

\[ \gamma^{(0)} = \arg \min_{\gamma^{(0)}} E(B^{(0)} - \gamma^{(0)} \nabla E^{(0)}) \]
Multi patch NURBS

- Tensor-product: 4-sided shape
- Complex geom: multi-patch
- Each patch: its own parametrisation
- Joining patches: not trivial if there is no match at the interface

Nitsche’s method
Concluding remarks

For composite laminates modeling, NURBS IGA offers

- fast pre-processing: automatic generation of interface el.
- link to CAD data: ideal for design-analysis cycles
- high order NURBS elements: highly accurate derivative fields
- less expensive than low order Lagrange elements
- geometry: exactly represented

- Tensor-product: no local refinement

- T-splines: complex algorithms
- Hierarchical B-splines
- Discontinuous Galerkin methods (NURBS)
On-going and future work

- Multi model coupling: plate (macrolevel) and refined 3D continuum models (mesolevel)
- Macrolevel: through-thickness homogenisation can be used
- Conforming coupling: coupling via an interface
- Non-conforming coupling: 3D model placed anywhere on a
Multi model coupling with Nitsche’s method

\[
\int_{\Omega} \delta \varepsilon^T \sigma \, d\Omega - \int_{\Gamma_T^*} [\delta u]^T n \{ \sigma \} \, d\Gamma - \int_{\Gamma_T^*} \{ \delta \sigma \}^T n^T u \, d\Gamma + \int_{\Gamma_T^*} \alpha [\delta u]^T [u] \, d\Gamma = \int_{\Gamma_T^*} \delta u^T \bar{t} \, d\Gamma
\]
Thank You!