AN IMPLEMENTATION OF THE REGULARIZED EXTENDED FINITE ELEMENT

METHOD IN ABAQUS

by

YU-JUI LIANG

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2019

To my wife and family

ACKNOWLEDGEMENTS

ABSTRACT

AN IMPLEMENTATION OF THE REGULARIZED EXTENDED FINITE ELEMENT

METHOD IN ABAQUS

Yu-Jui Liang, PhD

The University of Texas at Arlington, 2019

Supervising Professor: Endel V. Iarve

The Regularized eXtended Finite Element Method (Rx-FEM) methodology is a discrete damage modeling (DDM) technique, which represents an approach to the progressive damage analysis (PDA) in laminated composites when multiple damage events such as matrix cracks and delamination are introduced into the model via the displacement discontinuities. In the Rx-FEM, the Heaviside step function that is typically used to introduce a displacement discontinuity across a crack surface in the eXtended Finite Element Method (x-FEM) is replaced by a continuous function approximated by using the finite element (FE) shape functions. This regularization offers unique possibility for the implementation of the Rx-FEM methodology in commercial finite element software which has many advantages including the powerful solver, built-in capabilities, post-processing, and visualization. The proposed implementation of the Rx-FEM methodology in Abaqus consists of the mesh-independent

cracking (MIC) technique to modeling the transverse matrix cracks in each ply of the composite laminate, and modeling the delamination between plies by using the mixed-mode cohesive formulation which is also used to describe the MIC matrix crack propagation characteristics, and the interaction between matrix crack and delamination. The validity of the proposed implementation is tested by several verification models of MIC modeling in composite laminates, and delamination between plies. The proposed implementation not only demonstrates accurate predictions, but also allows to take advantages of various built-in capabilities from the host software Abaqus such as geometrically nonlinear solution, contact interactions, post-processing, and visualization. More importantly, it opens up a new venue for the Rx-FEM implementation across different element platforms including multiphysics applications.

TABLE OF CONTENTS

LIST OF FIGURES

x

LIST OF TABLES

CHAPTER I

INTRODUCTION

The usage of composite materials has increased rapidly from defense industries to civil aviation and other automotive applications. Composite materials offer high specific strength, corrosion resistance and manufacturability. However, the cost associated with certification and qualification of such advanced structures remains high. Detailed modeling of the deformation and fracture of composite materials is widely considered a vital step toward a new design and qualification paradigm where significant amounts of experimental testing are replaced by computational modeling and simulation. This brings the need for the development of high-fidelity methodologies and analysis tools with detailed modeling of damage accumulation and evolution in composite materials. The fracture mechanism, especially in laminated composites, is a progressive evolution of discrete damage events, such as fiber/matrix debonding, matrix cracking within a ply, delamination between plies and fiber failure. These damage modes evolve in various combinations, and they can be isolated in some specific situations. However, even in these cases, the specific failure modes can be isolated only at certain load levels, i.e. transverse cracking will lead to delamination and delamination propagation in unidirectional composites is accompanied by fiber bridging. Free-edge delamination in most cases are precipitated and/or closely followed by matrix cracking. An idealized damage progression sequence in a laminated composite plate subjected to tensile loading is shown in Figure 1 [1]. Thus, one of the objectives of the Advanced Composites Project (ACP) program which was

recently launched by NASA is development and validation of physics-based analysis tools in order to predict performance and durability of structural elements made from composite materials.
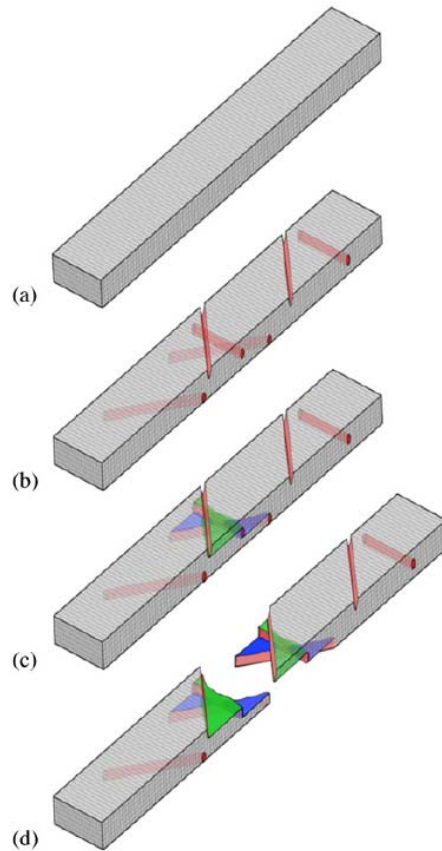
Figure 1. Damage progression in a composite laminate under tensile loading. (a) Pristine state; (b) Matrix cracking stage; (c) Delamination stage, linking up matrix cracks in various plies and (d) failure [1].

1-1 Continuum Damage Modeling

To date, significant progress has been achieved in developing progressive damage modeling methodologies for the damage evolution and interaction phenomena of laminated composite materials. The continuum damage modeling (CDM) methodology is the most common and widely used approach for progressive damage modeling in laminated composites [2]–[4]. Intra-ply damage modes have been studied primarily while delamination between plies has been investigated extensively using the interface fracture modeling technique such as cohesive zone models [5]–[7] and virtual crack closure techniques [8], [9]. A critical distinction between the CDM and interface fracture modeling exists in the approach where a displacement discontinuity is provided. The CDM represents all forms of damage as the local volumetric stiffness degradation which describes matrix and fiber damage by developing local constitutive material models for directionally reducing the stiffness properties of the affected elements, whereas the interface fracture modeling techniques directly include the kinematics of the displacement jump.

The ease of implementation for the CDM methodology in conventional FE framework has greatly proliferated its application [10]–[12]. However, it has significant difficulties in predicting local effects of interactions between various damage modes and local effects of stress redistribution in the damage area without any special meshing techniques such as fiber aligned meshing [13]–[15]. This issue has been investigated in detail [16]–[18] where the CDM is inaccurate to modeling longitudinal splits in plies with fiber aligned with the tensile loading direction as shown in *Figure 2* [14], or the fracture is characterized by strong coupling between transverse matrix crack and delamination [19], [20].

3

(a) Radial Mesh   (b) Aligned Mesh
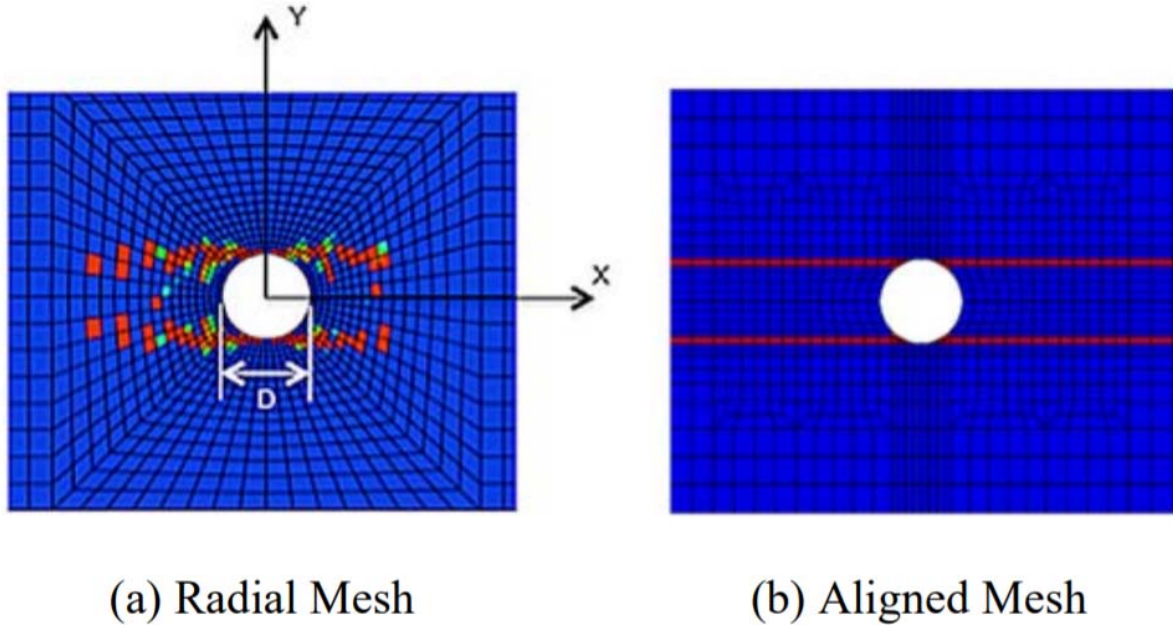
Figure 2. Splitting damage predicted using a radial mesh and an aligned mesh [14].

1-2 Discrete Damage Modeling

Discrete damage modeling (DDM) represents an approach to progressive damage analysis when multiple individual damage events, such as matrix cracks in plies, delamination between plies and their interaction, are explicitly introduced into model through the displacement discontinuities. The DDM approach to modeling crack-induced displacement discontinuities employs the mesh-independent crack (MIC) technique for crack growth without remeshing, such as the eXtended Finite Element Method (x-FEM) [21] which has been developed over the past decade. In the x-FEM formulation, additional degrees of freedom are added to the element along the crack surface to describe the displacement discontinuities. This crack surface partitions the element into two parts in which the integration schema must be customized. Although the x-FEM formulation is not straightforward and much more

complicate than CDM, it can solve some of the most significant problems associated with the CDM methodology.
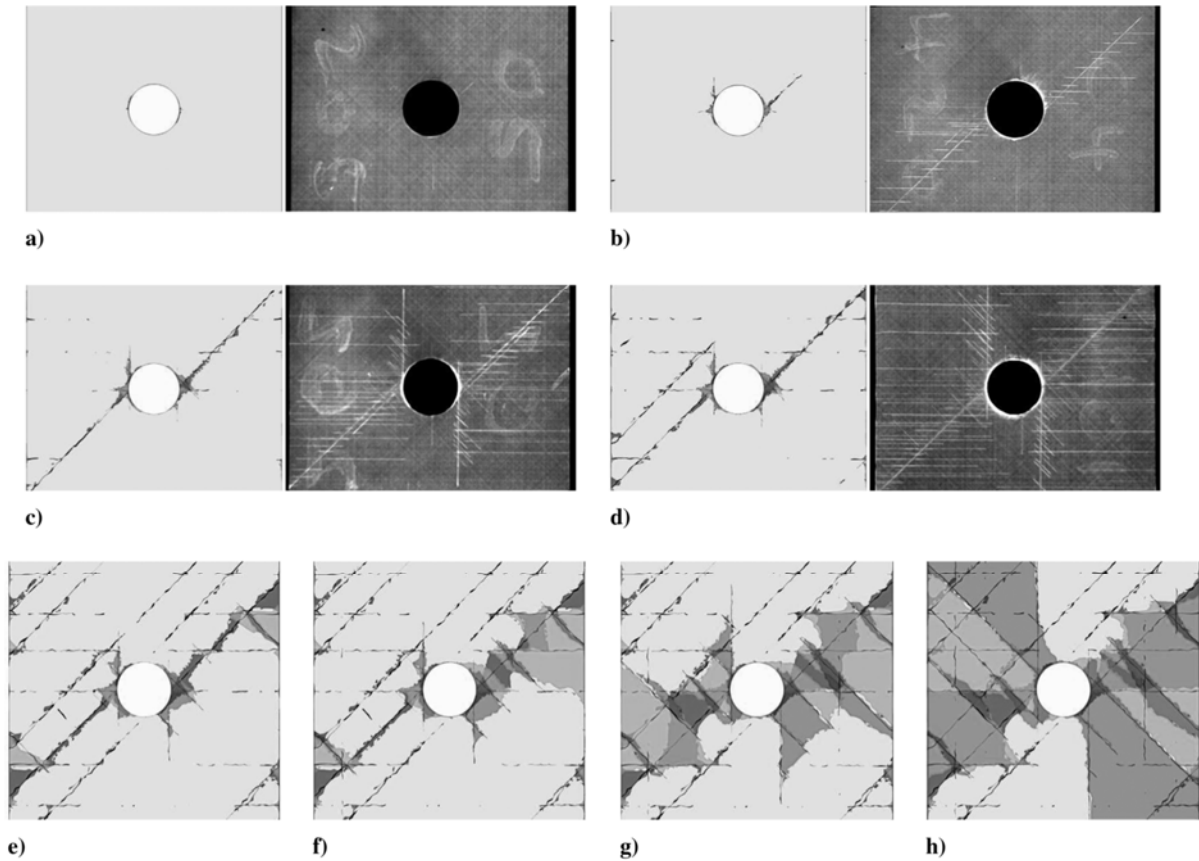
Figure 3. Damage progression of open hole composites for different load levels [22].

Another direction of the DDM is the Regularized eXtended Finite Element Method (Rx-FEM) [1], [22]–[26], where the Heaviside step function used in x-FEM to represent the crack surface is replaced by a continuous function changing from 0 to 1 over a narrow volume called crack region or gradient zone. The narrow volume integrals in the gradient zone to the surface integrals in the limit of mesh refinement was discussed in [1]. The Rx-FEM

formulation [23] uses the displacement approximation shape functions to approximate the Heaviside step function and maintain the Gauss integration schema for computing the stiffness matrix without regard to crack orientation. This is a robust numerical approach for modeling complex networks of matrix cracks and delamination as shown in Figure 3. This methodology is implemented in an in-house custom software program BSAM, and extensively applied to static and fatigue analysis in laminated composite structures [27]–[31]. To date, several stand-alone implementations of the DDM have been reported, and applied to failure prediction in laminated composite coupons [1], [32], [33] and structures of increasing complexity [27], [28], [34].

1-3 Implementation in Commercial Finite Element Software

Commercial finite element software has several advantages including established markets, reputation, very powerful solver, and many built-in capabilities which are impractical to implement again in a research code. The implementation of the DDM methodologies in commercial software adds significant mathematical complexity compared to the implementation of the CDM methodologies, where the damage is represented as a change of element stiffness properties. To date, several implementations of DDM methodologies into the commercial FE software and Abaqus [35] in particular have been reported [36]–[39]. What is common in these implementations is that a user defined element (UEL) was created encapsulating the kinematics of the x-FEM element pair, which is produced after the degrees of freedom are enriched. This type of implementation has wide freedom to allow customizing the integration scheme of the cracked elements and introducing either phantom nodal degrees of freedom (Augmented-FEM [38], [39]) or degrees of freedom representing displacements at

the crack surface and element edge intersection (Floating node method [37]), and traditional x-FEM [36]. However, these implementations only allow to take advantages of a very small subset of capabilities in the commercial FE software, and this approach can take advantage of the solver portion of the code, while even visualization represents challenges, not to speak of more essential features such as contact, etc.
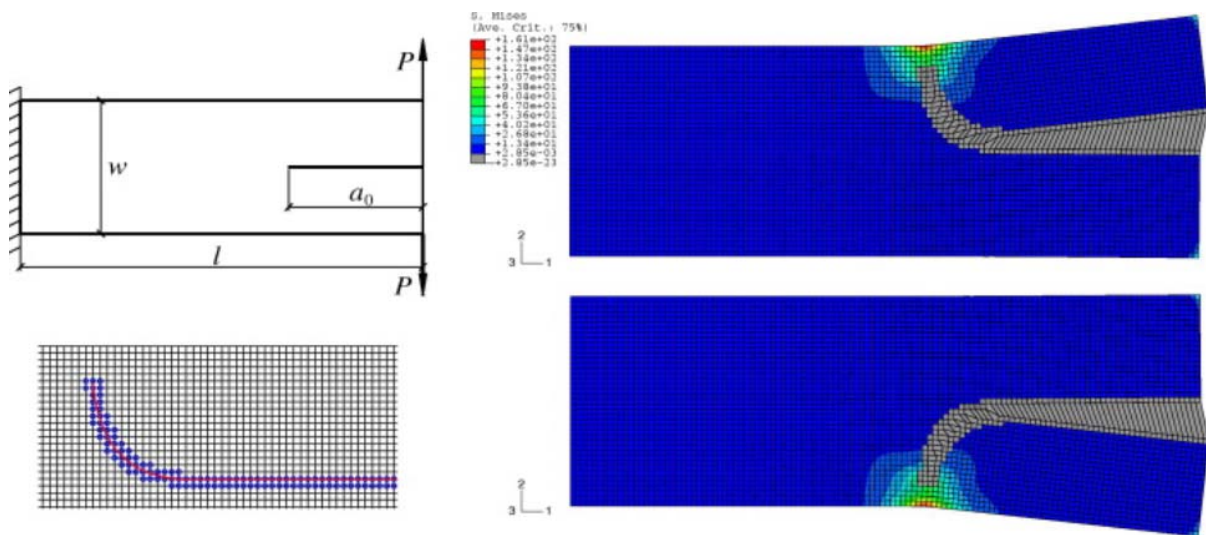
Figure 4. Crack growth of a cantilever beam with an eccentric crack [36].

CHAPTER II

RESEARCH OBJECTIVES

Within the Rx-FEM, the matrix crack surface is described by a continuous function, which is approximated by using the same shape functions for representing the displacement discontinuities. The elements along the crack surface are not partitioned because the crack surface is not encapsulated in an element but spread over several neighboring elements. This regularization of the Heaviside step function offers unique possibility to implement the Rx-FEM in commercial FE software. The implementation capitalizes on utilization of original Gauss integration schema of the native Abaqus element in Rx-FEM even after the enrichment is introduced to accommodate a MIC. Therefore, the first research objective of the present research is to provide a novel approach of the Rx-FEM implementation in commercial FE software Abaqus [40], [41], which is based on native element superposition.

To date, there are several implementations [36]–[39] of DDM methodologies into the commercial FE software and Abaqus [35] in particular. What is common in these implementations is that a user defined element (UEL) was created encapsulating the kinematics of the x-FEM element pair, which is produced after the degrees of freedom are enriched. However, these implementations only allow to take advantages of a very small subset of capabilities in the commercial FE software, and this approach can take advantage of the solver portion of the code, while even visualization represents challenges, not to speak of more essential features such as contact, etc. Therefore, the second objective of the present research

aims to utilization of core capabilities such as geometrically nonlinear capability, contact feature, and visualization from the host FE software without modification of the Rx-FEM implementation.

Last, the third objective of the research is to apply the proposed implementation to solution of a number of problems and evaluate the accuracy and robustness of the developed methodology.

In the following chapters, the chapter of the Rx-FEM theory is introduced first, and then followed by the chapter of the Abaqus implementation which includes the implementation of the MIC technique to modeling mesh-independent matrix cracks in each ply of the composite laminate, and the implementation for modeling delamination between plies at the ply interface. Last, several numerical models devoted to verification and application of the proposed method in laminated composites are considered to verify the transverse matrix crack insertion and propagation correctly, and the accuracy of delamination for mode I fracture and mode II fracture and the interaction between initial matrix crack and delamination by comparing with the benchmark solutions.

The proposed implementation of the Rx-FEM not only demonstrates accurate predictions for progressive damage analysis in laminated composites, but also allows to take advantage of various built-in capabilities from the host finite element software Abaqus such as geometrically nonlinear solution, contact, post-processing, and visualization.

CHAPTER III

REGULARIZED EXTENDED FINITE ELEMENT METHOD

In this chapter, the Regularized eXtended Finite Element Method (Rx-FEM) methodology which consists of modeling transverse matrix cracks in each ply of composite laminate by the mesh-independent cracking (MIC) technique and modeling delamination between plies is presented in detail. First, a brief overview of the standard finite element method (FEM) is provided for context, and to point out certain aspects of element formulation. The traditional eXtended Finite Element Method (x-FEM) is then discussed for introducing the basic concept of displacement discontinuity by partitioning element.

Next, the Rx-FEM methodology which is a variant of the x-FEM is then presented. First, the mixed-mode cohesive zone model used for delamination and matrix crack propagation is described. This is followed by development of the MIC technique for modeling transverse matrix cracks within a single ply of composite laminate. The usage of the surface-based cohesive zone model to describe the MIC crack propagation characteristics is then discussed. Finally, the discussion of modeling the interaction between transverse matrix cracks and delaminations in a composite laminate is presented.

3-1 Standard Finite Element Method

In the standard finite element method, the displacement field is described by defining the values at nodes which are connected by elements. Both the physical location of the element

and the displacement field within the element are described by using the same parametric equations/shape functions:

$$x(\xi, \eta, \zeta) = \sum_{i=1}^{N_{nodes}} N_i(\xi, \eta, \zeta) x_i$$

$$u(\xi, \eta, \zeta) = \sum_{i=1}^{N_{nodes}} N_i(\xi, \eta, \zeta) u_i$$

$$(1)$$

where the x_i and u_i are the nodal values of the coordinates and displacements, N_{nodes} is number of nodes, and N_i is the shape functions in natural coordinates (ξ, η, ζ). These continuous shape functions keep continuities of the physical location and the displacement field within the element. The local strain field $\boldsymbol{\varepsilon}$ is then computed from the derivatives of the displacement field in natural coordinates as follows:

$$\boldsymbol{\varepsilon}(\xi, \eta, \zeta) = \mathbf{B}(\xi, \eta, \zeta) \cdot \mathbf{u} \qquad (2)$$

where $\mathbf{B}$ is the strain-displacement matrix which contains the derivatives of the shape functions. The local stress filed $\boldsymbol{\sigma}$ is then calculated by multiplying the local strain field by the material modulus matrix $\mathbf{D}$ as follows:

$$\boldsymbol{\sigma}(\xi, \eta, \zeta) = \mathbf{D} \cdot \boldsymbol{\varepsilon}(\xi, \eta, \zeta) \qquad (3)$$

The strain energy of an element is be computed by integration of the strain energy density over the whole element as follows:

$$W^e = \frac{1}{2} \int_{\Omega_e} \mathbf{u}^T \mathbf{B}^T \mathbf{D} \mathbf{B} \mathbf{u} d\Omega \tag{4}$$

The strain energy of the volume is obtained by the summation of the strain energy for all the elements. Next, the element stiffness $\mathbf{k^e}$ which is the relationship between the nodal forces and displacements is calculated as follows:

$$\mathbf{k^e} = \int_{\Omega_e} \mathbf{B}^T \mathbf{D} \mathbf{B} d\Omega \tag{5}$$

In the numerical computation, the integral in element formulation is typically using a Gauss quadrature rule where the volume integration is approximated by the summation as follows:

$$\mathbf{k^e} \approx \sum_{l=1}^{N_{int}} (\mathbf{B}^T \mathbf{D} \mathbf{B} |\mathbf{J}|)_{\xi_l} \cdot W_l \tag{6}$$

where the parameters ξ_l is the location of the integration points, N_{int} is the number of integration point, W_l is the weight functions which depend on the order of the quadrature, and J is the Jacobian matrix which maps the derivatives with respect to the parametric variables ξ to the spatial derivatives. A brief overview of the element formulation in standard FEM was presented above. Multiple combinations of elements in standard FEM comes from the choices of the element type (tetrahedral, wedge, and hexahedral), shape functions (first order and second order), and integration schemes (full, reduced, and selective).

A significant challenge of the standard FEM is, however, representation of a discontinuity such as a crack within the finite element model, since the continuity of the shape functions within the element is one of the requirements for converging the solution. Although the easiest way to introduce a discontinuity is along the element boundaries, many restrictions must be taken into account such as either the crack location must be considered when generating the initial mesh of the structure before the analysis start, or the structure is remeshed for crack insertion and propagation during the analysis. Both approaches to describe discontinuity have serious limitations. First, it is not always possible to know where the cracks will appear before the analysis start except for certain situations, and the idea of this way is contrary to the goal of predicting damage in structure. Second, remeshing technique used for laminated composites is questionable because multiple meshing constraints occurs at the ply interface, where the cracks with different orientations in adjoining plies must be accommodated.

3-2 The eXtended Finite Element Method

The eXtended Finite Element Method (x-FEM) methodology was proposed to introduce the displacement discontinuities for crack growth without any remeshing technique [21]. The idea of the x-FEM is that this methodology enriches the displacement field within a domain with an arbitrary crack by introducing additional degrees of freedom to accommodate for discontinuity, and then divides the domain into two regions where the additional degrees of freedom act. The domain boundaries, which represents the discontinuity, may cross the element boundaries as shown in Figure 5 [21] which shows a schema of an arbitrary crack on a structured and unstructured mesh..
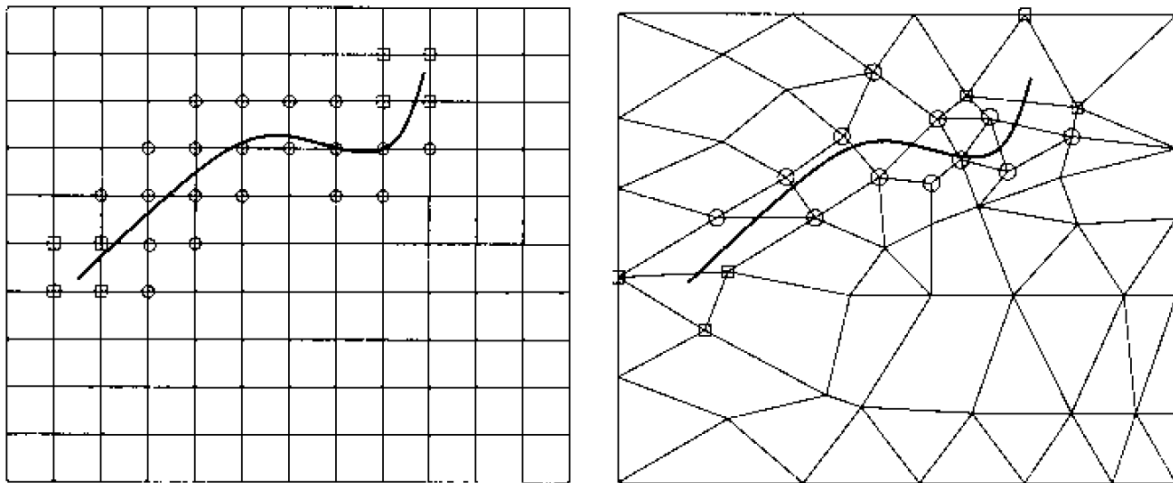
Figure 5. Crack on a structured mesh (left) and an unstructured mesh (right). The circled nodes are enriched by the Heaviside step function whereas the squared nodes are enriched by the crack tip function [21].

The crack surface Γ_α, where the displacement function is experiencing a discontinuity, can be described by the signed surface distance function $f_\alpha(\mathbf{x})$ as follows:

$$f_\alpha(\mathbf{x}) = \text{sign}\big(\mathbf{n}(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})\big) \min_{\bar{\mathbf{x}} \in \Gamma_\alpha} \|\mathbf{x} - \bar{\mathbf{x}}\| \tag{7}$$

where $\mathbf{n}(\bar{\mathbf{x}})$ is the normal to the crack surface Γ_α at the point $\bar{\mathbf{x}}$. The signed surface distance function is defined at an arbitrary point $\mathbf{x}$ of the domain and is equal to the distance from that point to the crack surface, and is positive if the point is above the crack surface and negative on the other side of the crack surface as shown in Figure 6.
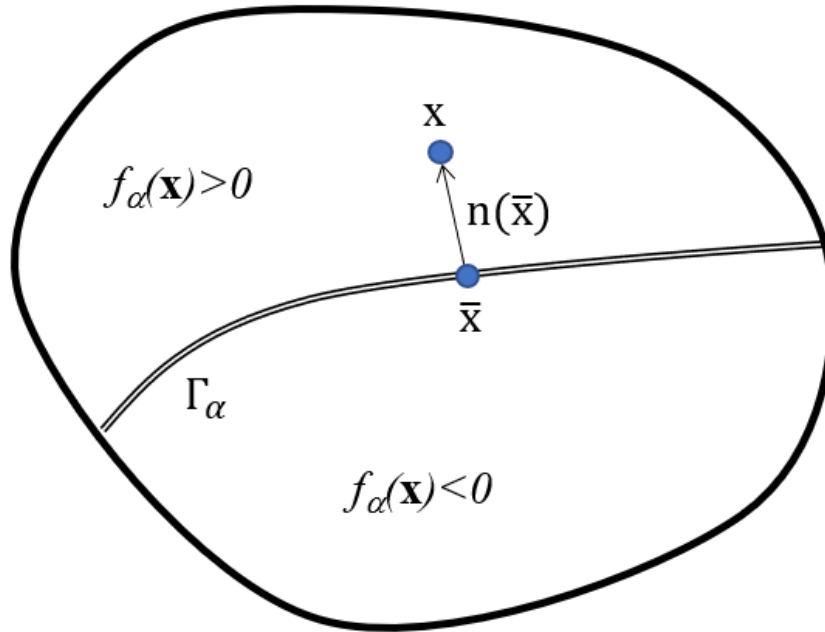
Figure 6. Subdivision of the domain by the signed distance function.

The Heaviside step function *H(x)* as follows:

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$

(8)

where the Heaviside step function is used to describe the displacement jump at the crack surface Γ_α. Assuming $\mathbf{u}^{(1)}$ is the displacement field within the region below the crack surface ($f_\alpha(\mathbf{x}) < 0$) and $\mathbf{u}^{(2)}$ is the displacement field within the region above the crack surface ($f_\alpha(\mathbf{x}) > 0$). Thus, the entire displacement field of the domain can be described as follows:

$$\mathbf{u} = (1 - H(f_\alpha(\mathbf{x}))) \cdot \mathbf{u}^{(1)} + H(f_\alpha(\mathbf{x})) \cdot \mathbf{u}^{(2)}$$

(9)

The derivative of the Heaviside step function is the Dirac delta function δ, where the value within the domain is zero except for the origin. At the origin where the Heaviside step function is discontinuous, the Dirac delta function is undefined. The Dirac delta function is not a function in the usual sense, and it is more correctly referred to as Dirac delta distribution. It is also has the property for an arbitrary function $g(\mathbf{x})$ of x, then:

$$\int_{-\infty}^{\infty} \delta(\mathbf{x}) g(\mathbf{x}) dx = g(\mathbf{x_0})$$

(10)

16

Different forms of approximation of the fields $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ have been proposed over the years such as an approach by Hansbo and Hansbo [42] in which the displacements $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ are approximated by the original shape functions and result in phantom nodes in the enriched element. Common to all such approaches is the fact that the integration schema for the domain 1 and domain 2 which correspond to the sub-volumes $f_\alpha(\mathbf{x}) < 0$ and $f_\alpha(\mathbf{x}) > 0$ are different as shown in Figure 7.
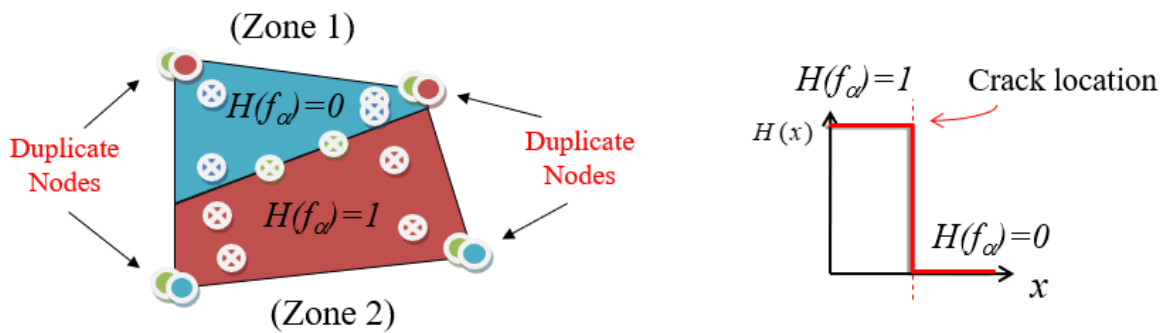
Figure 7. The schema of the x-FEM formulation.

Therefore, a custom integration schema in each sub-volume is needed. Such a construction is possible but is quite tedious and involves many possible combinations of crack locations with respect to the element boundaries. For instance, in two-dimension a quadrilateral element in this case may divide to two quadrilateral elements or two triangle elements or one triangle element and one five-sided region that will need to be further subdivided. There are even more complicate combinations to consider in three-dimensional case. Moreover, at the

interface between two plies with different fiber orientation in laminated composites, a double partition of the element surface must be considered in order to represent the delamination propagation from the crack intersections. Thus, just as it is advantageous to preserve the mesh structure, another representation of the crack that does not involve the recalculation of integration point locations is desirable.

3-3 The Regularized eXtended Finite Element Method

Generally, any smoothing or filtering operation could be thought of as a regularization process when the "regularized" form is used in place of the original. Hence, the term "Regularized" is used in this methodology that a function ill-suited for analytical purposes is replaced by a smoother approximation which captures the essential qualities of the original function.

The overview of Regularized eXtended Finite Element Method (Rx-FEM) could be in order the following sections describe the formulation of the Rx-FEM methodology. First, the cohesive zone model used for the delamination and transverse matrix crack propagation is described followed by the development of the regularized mesh-independent cracking (MIC) modeling technique for transverse matrix crack within a single ply of the composite laminate. Next, the usage of the mixed-mode surface-based cohesive zone model to describe the MIC crack propagation is provided. Finally, the discussion of modeling the interaction between transverse matrix cracks and delamination in a composite laminate is presented.

3-3-1 Cohesive Zone Model

The mixed-mode cohesive zone interface fracture model proposed by Turon [7] is discussed in the present section. A brief overview is provided for completeness of the Rx-FEM formulation and the reader is referred to reference [7] for full details. An arbitrary point at the crack interface with a normal vector $\mathbf{n}$ and a displacement jump vector $\Delta\mathbf{u}$. The goal is to describe the fracture energy of separation of an arbitrary point at the crack interface which will be used in deriving the system of equations from the minimum potential energy principle. This discussion of the cohesive model directly pertains to the delamination propagation between plies and will be used for the MIC crack propagation as well.

In this model, the cohesive traction on the crack surface is related to the displacement jump $\Delta\mathbf{u}$ in the following:

$$\boldsymbol{\tau} = (1 - d)K\Delta\mathbf{u} + dK\langle -\Delta u_n \rangle \mathbf{n} \tag{11}$$

where the K is a high initial penalty stiffness of the surface bond and d is the damage variable. The first term in Eq. (11) is responsible for the crack cohesive force, and the second term is presented to prevent interpenetration of the crack surfaces. The brackets $\langle x \rangle = \frac{1}{2}(x + |x|)$ represent the McAuley operator. The component of the displacement jump normal to the crack surface is as follows:

$$\Delta u_n = (\Delta\mathbf{u} \cdot \mathbf{n}) \tag{12}$$

where Δu_n represents the mode I component of the total separation. According to the methodology proposed by Turon [7], the cohesive energy can be written in the invariant form as a function of the absolute value of the displacement jump $\lambda = |\Delta \mathbf{u}|$, and a mode mixity parameter B defined as follows:

$$B = 1 - \frac{\langle \Delta u_n \rangle^2}{\lambda^2} \tag{13}$$

The parameter B is equal to 0 for representing pure mode I fracture and is equal to 1 for pure mode II and III fracture.
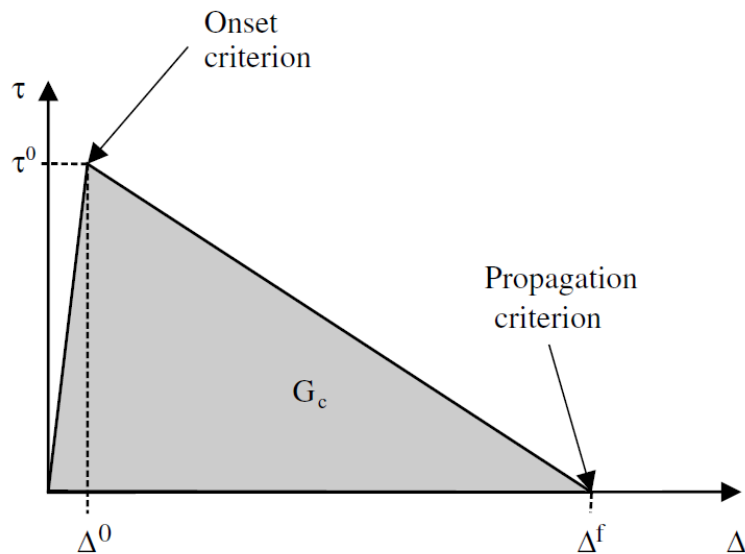
Figure 8. Traction-Separation and Damage-Separation relationships [7]

As one can see in Figure 8 [7], a bi-linear traction-separation law is assumed if B is fixed to a constant. Once the maximum traction τ_0 is reached at the separation Δ_0, the cohesive traction starts to linearly reduce until separation reaches the final separation Δ_f where the cohesive traction is zero. The traction-separation equation is shown as follows:

$$\tau(\lambda) = \begin{cases} K\lambda, & \lambda < \Delta_0 \\ \dfrac{\Delta_f - \lambda}{\Delta_f - \Delta_0}\tau_0, & \Delta_0 < \lambda < \Delta_f \\ 0, & \lambda > \Delta_f \end{cases} \tag{14}$$

where τ_0 is the cohesive strength. In Eq. (14), the initial value of the displacement jump beyond which the beginning of the interface failure takes place is defined as follows:

$$\Delta_0 = \frac{\tau_0}{K} \tag{15}$$

The damage variable d is determined by substituting Eq. (14) into Eq. (11) as shown below:

$$d(\lambda) = \begin{cases} 0, & \lambda < \Delta_0 \\ \left(\dfrac{\lambda - \Delta_0}{\Delta_f - \Delta_0}\right)\dfrac{\Delta_f}{\lambda}, & \Delta_0 < \lambda < \Delta_f \\ 1, & \lambda > \Delta_f \end{cases} \qquad (16)$$

And the damage rate law can be found from the derivative:

$$\frac{\partial d}{\partial \lambda} = \begin{cases} 0, & \lambda < \Delta_0 \\ \dfrac{\Delta_0 \Delta_f}{\lambda^2 (\Delta_f - \Delta_0)}, & \Delta_0 < \lambda < \Delta_f \\ 0, & \lambda > \Delta_f \end{cases} \qquad (17)$$

Both the cohesive strength and the stiffness are reduced by prior damage. If the cohesive zone is unloaded, damage d does not accumulate again until the new maximum traction associated with the prior damage is reached.

The cohesive strength τ_0 also depends on the mode mixity parameter B as follows:

$$(\tau_0)^2 = Y^2 + (S^2 - Y^2)B^\eta \qquad (18)$$

where Y and S are the interfacial normal and shear strength, and η is an experimentally determined influence parameter. All cohesive parameters in the analysis can be measured by

using standard test methods such as fracture toughness and cohesive strength. The fracture energy density $g(\lambda, B)$ required to create a displacement jump λ at the given point of the fracture surface is the area under the $\tau(\lambda)$ curve as follows:

$$g(\lambda, B) = \int_{q=0}^{\lambda} \tau(q, B)\, dq \tag{19}$$

To ensure the correct crack propagation response, the final value of the displacement jump Δ_f is defined so that the following condition is satisfied:

$$g(\Delta_f, B) = G_c(B) \tag{20}$$

where the critical energy release rate (ERR), G_c, or fracture toughness, is assumed to be a function of the mode mixity as follows:

$$G_c = G_{Ic} + (G_{IIc} - G_{Ic})B^{\eta} \tag{21}$$

and G_{Ic} and G_{IIc} are experimentally measured fracture toughness values. In the case of a bi-linear relationship cohesive law, the final value of the displacement jump is determined by the initial value of the displacement jump and the fracture toughness as follows:

23

$$\Delta_f = \frac{2G_c}{K\Delta_0} \tag{22}$$

The fracture energy associated with a delamination between plies is computed by integrating the fracture energy over the interface between n and $n+1$, designated by the horizontal crack surface $z = z_n$, and is given by:

$$\Phi_n = \int\int_{z=z_n} g(\lambda, B) \, dS \tag{23}$$

In the Rx-FEM methodology, the displacement jump at the ply interface surface $z = z_n$ is computed by using the enriched displacement approximation in the adjacent plies n and $n+1$, provided in the next section in Eq. (39).

3-3-2 Mesh-Independent Crack modeling within a ply

The goal of the MIC technique is modeling mesh-independent matrix cracks in each ply of the laminate. The regularized MIC formulation is derived from the traditional x-FEM proposed by Moes [21], where the cracked element is enriched by adding additional degrees of freedom to ensure the displacement jump across the crack surface. These additional degrees of freedom are associated with the shape functions, which are partitioned along the crack surface, and represented by the Heaviside step function. In the Rx-FEM formulation [1], [22], the Heaviside step function used in the x-FEM for representing the matrix crack surface is

replaced by a continuous function changing from 0 to 1 over a narrow area or volume called gradient zone as shown in Figure 9.
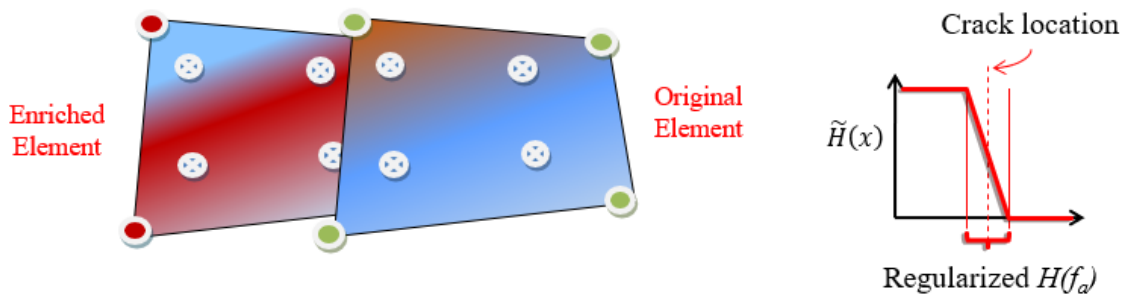
Figure 9. The schema of the Rx-FEM formulation.

This continuous function is approximated by the same shape functions as the displacement discontinuities. An advantage of the regularized MIC formulation is in maintaining a fixed Gauss integration schema throughout the analysis without regard to location and direction of the crack created during the analysis. In this case, the Gauss integration schema of the initial approximation may be used for integration of the enriched functions. This also simplifies the interaction between plies with different crack orientations because they maintain a common integration schema. The elements, as opposed to x-FEM are not partitioned since the crack surface is not inside an element, but rather spread over several neighbor elements which become an area or volume [43].

The displacement field is represented by defining its value at discrete nodes x_i, and i is from 1 to the total number of nodes of a given element. Both the physical domain and the

displacement field of an isoparametric element are described by using the same shape functions. Denote a set of all nodes of a given element e as Ω_e and conversely a set of elements sharing a node i as $\mho_i$. We will assume that the nodes are numbered in global numeration and write for each element as follows:

$$x(\xi^e) = \sum_{i \in \Omega_e} N_i^e(\xi^e) x_i$$

$$\mathbf{u}(\xi^e) = \sum_{i \in \Omega_e} N_i^e(\xi^e) \mathbf{u}_i$$

(24)

where ξ is the parametric coordinates. Considering a matrix crack appearing in this volume with a crack surface Γ_α, where the displacement exhibits a discontinuity, define a signed distance function f_α, which is the same as Eq. (7) as follows:

$$f_\alpha(\mathbf{x}) = \text{sign}\big(\mathbf{n}(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})\big) \min_{\bar{\mathbf{x}} \in \Gamma_\alpha} \|\mathbf{x} - \bar{\mathbf{x}}\| \qquad (25)$$

where $\mathbf{n}(\bar{\mathbf{x}})$ is the normal to the crack surface Γ_α at the point $\bar{\mathbf{x}}$. In the traditional x-FEM, the displacement discontinuity over the crack surface is obtained by multiplying the shape functions by the Heaviside step function and redefining the integration domains on each side of the discontinuity. The Heaviside step function in the Rx-FEM formulation, however, is replaced by a continuous function $\tilde{H}(\xi^e)$ as follows:

$$\tilde{H}(\xi^e) = \sum_{i \in \Omega_e} N_i^e(\xi^e) h_i \tag{26}$$

where N_i^e are the same shape functions as in Eq. (24). The continuity of the regularized step function in Eq. (26) through the domain is assured by single field of nodal values computed for node i, by summing over all elements sharing that node as following:

$$h_i = \frac{1}{2}\left\{1 + \left(\int_V \sum_{e \in \mho_i} N_i^e f_\alpha dV\right) \Big/ \left(\int_V \sum_{e \in \mho_i} N_i^e |f_\alpha| dV\right)\right\} \tag{27}$$

where we have omitted the coordinate dependence for clarity. All integrals are taken over element domains in parametric coordinates. This definition involves only continuous functions and the integrals can be calculated by using standard Gauss quadratures. The coefficients h_i are equal to 0 or 1 if the signed distance function f_α does not change sign within the domain of the shape functions N_i^e. If an arbitrary crack crosses the domain of the shape functions, then the displacement field within the crack region can be defined in the following form:

$$\mathbf{u} = \tilde{H} \cdot \mathbf{u}^{(1)} + \left(1 - \tilde{H}\right) \cdot \mathbf{u}^{(2)}$$

$$\mathbf{u}^{(1)} = \sum_{i \in \Omega_e} N_i^e \mathbf{u}_i^{(1)} \tag{28}$$

$$\mathbf{u}^{(2)} = \sum_{i \in \Omega_e} N_i^e \mathbf{u}_i^{(2)}$$

where the spatial argument has been omitted for conciseness. This displacement approximation in Eq. (28) contains enriched displacement fields in the crack region via displacement fields $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$, which are approximated by each original shape functions. If the fields $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ coincide then there is no displacement discontinuity and the crack begins to open otherwise. Since the independence of the displacement fields $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$, the strain and stress fields in each zone are also independent. The strain field is calculated from each displacement field in zone 1 and zone 2 using original strain-displacement relation matrix $[\boldsymbol{B}]$ as follows:

$$\boldsymbol{\varepsilon} = \widetilde{H} \cdot \boldsymbol{\varepsilon}^{(1)} + \left(1 - \widetilde{H}\right) \cdot \boldsymbol{\varepsilon}^{(2)}$$

$$\boldsymbol{\varepsilon}^{(1)} = [\boldsymbol{B}]\mathbf{u}^{(1)} \tag{29}$$

$$\boldsymbol{\varepsilon}^{(2)} = [\boldsymbol{B}]\mathbf{u}^{(2)}$$

And the stress field is computed from the strain as follows:

$$\boldsymbol{\sigma} = \widetilde{H} \cdot \boldsymbol{\sigma}^{(1)} + \left(1 - \widetilde{H}\right) \cdot \boldsymbol{\sigma}^{(2)}$$

$$\boldsymbol{\sigma}^{(1)} = [\boldsymbol{D}]\boldsymbol{\sigma}^{(1)} \tag{30}$$

$$\boldsymbol{\sigma}^{(2)} = [\boldsymbol{D}]\boldsymbol{\sigma}^{(2)}$$

where $[\boldsymbol{D}]$ is the stress-strain constitutive matrix. Note that in the Rx-FEM formulation the integration schema for the displacement zone 1 and zone 2 coincides with each original element integration schema without subdivision of the element as opposed to the x-FEM which

partition of the element. The strain energy W in the volume V computed using Eq. (28) to Eq. (30) is written as follows:

$$W = W^{(1)} + W^{(2)}$$

$$W^{(1)} = \frac{1}{2} \int_V \widetilde{H}(\boldsymbol{\varepsilon}^{(1)})^T \boldsymbol{\sigma}^{(1)} dV \tag{31}$$

$$W^{(2)} = \frac{1}{2} \int_V (1 - \widetilde{H})(\boldsymbol{\varepsilon}^{(2)})^T \boldsymbol{\sigma}^{(2)} dV$$

The strain energy given in Eq. (31) does not include the energy associated with the MIC propagation which is introduced in the next section. Denote a set of all integration point l of a given element as Π. The stiffness matrices $[K]$ associated with the displacement fields $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ are computed using the Gauss integration points as follows:

$$[\boldsymbol{K}] = [\boldsymbol{K}]^{(1)} + [\boldsymbol{K}]^{(2)}$$

$$\boldsymbol{K}^{(1)} \approx \sum_{l \in \Pi} (\boldsymbol{B}^T \boldsymbol{D} \boldsymbol{B}|\boldsymbol{J}|)|_{\xi_l} \cdot \widetilde{H} \cdot W_l \tag{32}$$

$$\boldsymbol{K}^{(2)} \approx \sum_{l \in \Pi} (\boldsymbol{B}^T \boldsymbol{D} \boldsymbol{B}|\boldsymbol{J}|)|_{\xi_l} \cdot (1 - \widetilde{H}) \cdot W_l$$

where $\boldsymbol{J}$ is the Jacobian matrix, and W_l are the Gauss weights. The load-displacement relationship for the element becomes as follows:

$$\begin{Bmatrix} \boldsymbol{F}^{(1)} \\ \boldsymbol{F}^{(2)} \end{Bmatrix} = \begin{bmatrix} \boldsymbol{K}^{(1)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{K}^{(2)} \end{bmatrix} \begin{Bmatrix} \mathbf{u}^{(1)} \\ \mathbf{u}^{(2)} \end{Bmatrix} \tag{33}$$

The off-diagonal $\boldsymbol{0}$ matrices represent stiffness associated with the separation of the two fields. When a cohesive zone model is used to simulate tractions at the crack surfaces, non-zero stiffness coefficients will appear in these positions.

3-3-3 Propagation of the Mesh-Independent Crack

In the standard x-FEM the fracture energy of the crack would be calculated over crack surface by integrating over each facet of the crack surface within each traversed element. In the regularized MIC formulation of the Rx-FEM methodology, the MIC propagation is governed by the cohesive constitutive relationship developed by Turon [7] for interface fracture which is directly inserted into the gradient zone of the MIC. A volume integration (3D) or area integral (2D) over gradient zone encompassing crack surface is performed instead. This narrow volume surrounding the crack surface is called gradient zone because in this region $|\nabla\widetilde{H}| = \sqrt{\nabla\widetilde{H} \cdot \nabla\widetilde{H}} > 0$. The idea is that the true crack surface area is equal to the effective area of the crack surface described by a continuous step function, and the fracture energy balance has to be represented through volume integration. Consider an arbitrary volume v of a ply

which contains a crack surface Γ_α defined by a signed distance function f_α. The surface area of this crack S_v in the volume v can be calculated as:

$$S_v = \int_v \delta_D(f_\alpha)\, \mathrm{d}V \tag{34}$$

where $\delta_D(f_\alpha)$ is the Dirac's delta function of the signed distance function. Assuming an arbitrary continuous function $g(\mathbf{x})$ defined in volume v, an equivalence of the surface integral over the crack surface Γ_α ($\Gamma_\alpha \epsilon V$) and a volume integral can be described by applying Eq. (34) as:

$$\iint_{\Gamma_\alpha \in v} g(\mathbf{x})\, \mathrm{d}S = \int_V g(\mathbf{x})\delta_D(f_\alpha)\, \mathrm{d}V \tag{35}$$

In the case of the Rx-FEM formulation, the Dirac's delta function of the signed distance function $\delta_D(f_\alpha)$ of the right term in Eq. (35) is replaced by the gradient of the continuous step function $|\nabla \widetilde{H}|$. Now the continuous function $g(\mathbf{x})$, which is defined over the volume v, is replaced by the point-wise fracture energy of crack opening which depends implicitly upon the spatial coordinates as a function the separation λ and the mode mixity B. Thus, the fracture energy required for crack surface opening within the arbitrary volume V is equivalent in the regularized and conventional crack surface formulations, and can be calculated as:

$$\int_V g(\lambda, B)\delta_D(f_\alpha)\, dV \approx \int_V g(\lambda, B)|\nabla \tilde{H}|\, dV \tag{36}$$

Note that the volume V must be sufficiently large with respect to the mesh size for the crack surface area computed by using the gradient of the step function to be equal to the real crack surface area.

In traditional interface cohesive zone formulation, the fracture energy is a function of the crack opening and dependent on the ratio of the mode I fracture (normal to the crack surface) and mode II fracture (tangential to the crack surface). In the regularized MIC formulation, the displacement jump is calculated, and the normal vector is defined at all points of the gradient zone as follows:

$$\Delta \mathbf{u} = \mathbf{u}^{(1)} - \mathbf{u}^{(2)}$$
$$\mathbf{n} = \nabla f_\alpha \tag{37}$$

where $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ are defined by using the enriched displacement fields in Eq. (28). Considering the Eq. (36), an equivalent crack surface fracture energy is calculated by using the following volume integral as follows:

$$M = \sum_{e=1}^{N_{elem}} \int_{v_e} g(\lambda, B)\,|\nabla \tilde{H}|\, dV \tag{38}$$

32

where the M is the cohesive energy associated with the crack, v_e is the element domain, $g(\lambda, B)$ is the point-wise cohesive fracture energy corresponding to the displacement jump in the gradient zone, and N_{elem} is the total number of elements. Note that only the elements in the gradient zone will be contributing to the cohesive energy.

3-3-4 Delaminations and their Interaction with Mesh-Independent Cracks

The same cohesive zone model mentioned above [7] is used to modeling the delamination between two plies in composites laminate. Consider the ply interface between n and $n + 1$, where the enriched displacement fields are created by the MIC technique for both plies. The displacement jump between two plies is computed as follows:

$$\Delta \mathbf{u} = \widetilde{H}_{n+1} \cdot \mathbf{u}_{n+1}^{(1)} + \left(1 - \widetilde{H}_{n+1}\right) \cdot \mathbf{u}_{n+1}^{(2)} - \widetilde{H}_n \cdot \mathbf{u}_n^{(1)} - \left(1 - \widetilde{H}_n\right) \cdot \mathbf{u}_n^{(2)} \qquad (39)$$

where the subscript denotes the ply number. The delamination fracture energy Φ involves the interaction of displacements between two neighboring plies and is computed by integrating the point-wise fracture energy over the area of the ply interface in Eq. (23).

The minimum potential energy principle for deriving the equilibrium equations of an N-ply laminate which contains matrix cracks in a ply and delamination between plies can be written as follows:

$$\delta\left(\sum_{n=1}^{N}\left(\tilde{H}W_n{}^{(1)} + \left(1 - \tilde{H}\right)W_n{}^{(2)} + \mathrm{M}_n\right) + \sum_{n=1}^{N-1}\Phi_n - A\right) = 0 \qquad (40)$$

where A is the work of external tractions applied at the laminate boundary. The first summation

for the MIC modeling includes the strain energy $W_n{}^{(1)}$ and $W_n{}^{(2)}$ on the two sides of the crack

surface, and the MIC fracture energy M_n of the nth ply. The delamination fracture energy Φ_n

for the second summation designates that it is computed for the interface between nth and

(n+1)th ply. Calculation of the surface energy associated with delamination is conceptually

straightforward and analogous to MIC which is based on Eq. (19), where the displacement

jump is calculated as the difference of the displacements on the interface between plies nth and

(n+1), which is given in each ply by Eq. (39). As a result, all quantities of the regularized

formulation in Eq. (40) is calculated by Gauss integration throughout the entire element

domain. The equilibrium system equations are computed without introducing any special

partitions or integration rules and without regard to the mesh orientation.

CHAPTER IV

IMPLEMENTATION OF THE REGULARIZED EXTENDED FINITE ELEMENT

METHOD IN ABAQUS

The implementation of the Rx-FEM within a stand-alone finite element code called BSAM has been already extensively applied to the static and fatigue analysis in laminated composites [27], [29]–[31]. However, commercial FE software, and Abaqus in particular, has its advantages including the established markets and trained users, powerful solver in explicit and implicit solution forms, and many built-in capabilities which are impractical to implement again in the custom code. The following sections describe a path of implementation of the Rx-FEM methodology which allows to take advantage of such capabilities directly since the kinematic portion of the formulation is based on native elements of the host solver. First, the implementation of the regularized MIC modeling technique for transverse matrix cracks within a single ply is presented. This is followed by the implementation for modeling delamination between plies by using the Turon's [7] surface-based cohesive model, which is also used to describe the MIC crack propagation characteristics. Finally, the solution algorithm of the Rx-FEM implementation in Abaqus is introduced and further detail about the mechanism of the MIC implementation is then discussed.

4-1 Implementation of the Mesh-Independent Cracking

The idea of implementation of the MIC modeling by the Rx-FEM approach is based on representing the enriched displacement field corresponding to $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ in Eq. (28) by superimposing two built-in native Abaqus elements. The cohesive force between the two, which corresponds to energy M_n in Eq. (40) is implemented by means of an ABAQUS user defined element (UEL). Figure 10 shows two native Abaqus elements (CPS4) with nodes 1 to 4 and the element with nodes 5 to 8, where the node 1 to 4 and node 5 to 8 are pairwise coincident. The Abaqus user defined material (UMAT) capability is employed to multiply the material elasticity matrix by $\widetilde{H}(x)$ and $\left(1 - \widetilde{H}(x)\right)$ which correspond to the first and second terms in Eq. (40), respectively. We shall call these superimposed elements as the "original element" and "duplicate element" and rewrite Eq. (28) for the purposes of implementation as follows:

$$\mathbf{u} = \widetilde{H} \cdot \mathbf{u}_{\text{original}} + \left(1 - \widetilde{H}\right) \cdot \mathbf{u}_{\text{duplicate}} \tag{41}$$

The original and duplicate element form identical but independent FE model. It introduces an overhead, which can be avoided in standalone implementation but not commercial software implementation. Note that this overhead is present in any x-FEM type implementation where the respective UEL contains additional (phantom or floating) degrees of freedom. In the present work the UEL implemented to account for the cohesive forces corresponding to energy M_n contains no independent or additional degrees of freedom and only ties together the physical degrees of freedom of the kinematic elements involved akin to a

typical cohesive UEL often build for interface failure modeling in standard element analysis. Since there are two different UEL usage in this article, we shall call the UEL described in the present section as the "MIC UEL" for the MIC implementation and another UEL as the "interface UEL" for delamination implementation which will be introduced in the later section. In the two-dimensional formulation, the MIC UEL contains nodes 1 to 8 and calculates the cohesive traction as a result of separation, which can be obtained as follows:

$$\Delta u(x, y) = \{N\}\{(u_5 - u_1)\ (u_6 - u_2)\ (u_7 - u_3)\ (u_8 - u_4)\}^T$$

$$\Delta v(x, y) = \{N\}\{(v_5 - v_1)\ (v_6 - v_2)\ (v_7 - v_3)\ (v_8 - v_4)\}^T$$

(42)

and

$$\{N\} = \{N_1\ N_2\ N_3\ N_4\}$$

$$N_1 = \frac{(1 - \xi)(1 - \eta)}{4}, N_2 = \frac{(1 + \xi)(1 - \eta)}{4}$$

$$N_3 = \frac{(1 + \xi)(1 + \eta)}{4}, N_4 = \frac{(1 - \xi)(1 + \eta)}{4}$$

(43)

where the shape function vector $\{N\}$ in the natural coordinates (ζ, η) is that of the original element or duplicate element which are identical, and u_1 to u_8 and v_1 to v_8 are the nodal displacements of normal and shear components in the global coordinates. In the MIC implementation, the matrix cracks within a single ply of the laminate are inserted parallel to

the fiber orientation and therefore the normal Δu_n and shear Δu_t components of the displacement separation vector can be written as:

$$\begin{Bmatrix} \Delta u_t \\ \Delta u_n \end{Bmatrix} = [R] \begin{Bmatrix} \Delta u(x, y) \\ \Delta v(x, y) \end{Bmatrix}$$

(44)

$$[R] = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

where $[R]$ is the global-local transformation matrix, and θ is the fiber orientation with respect to the X axis of the global coordinate system as shown in Figure 10. We shall express the displacement separation vector through nodal displacements in the two elements at the interface between plies. Introduce a nodal displacement vector containing 16 components corresponding to the nodes 1 to 8 with each x and y degrees of freedom in global coordinates. Adopting the following sequence of the degrees of freedom $\{U\}$ as shown below:

$$\{U\} = \{u_1 \ v_1 \ u_2 \ v_2 \ u_3 \ v_3 \ u_4 \ v_4 \ u_5 \ v_5 \ u_6 \ v_6 \ u_7 \ v_7 \ u_8 \ v_8\}$$

(45)

and substituting Eq. (42) (43) and (45) into Eq. (44), the relationship between the separation and the nodal displacements is derived as follows:

$$\begin{Bmatrix} \Delta u_t \\ \Delta u_n \end{Bmatrix} = [\mathbf{Y}]\{\mathbf{U}\}^T \tag{46}$$

and

$$[\mathbf{Y}] = \begin{bmatrix} -CN_1 & -SN_1 & -CN_2 & -SN_2 & -CN_3 & -SN_3 & -CN_4 & -SN_4 \\ SN_1 & -CN_1 & SN_2 & -CN_2 & SN_3 & -CN_3 & SN_4 & -CN_4 \\[4pt] CN_1 & SN_1 & CN_2 & SN_2 & CN_3 & SN_3 & CN_4 & SN_4 \\ -SN_1 & CN_1 & -SN_2 & CN_2 & -SN_3 & CN_3 & -SN_4 & CN_4 \end{bmatrix} \tag{47}$$

where $[\mathbf{Y}]$ is the global displacement-separation relation matrix, and C and S are cos θ and sin θ, respectively. A specific form of the traction separation law of the implemented cohesive zone model is then used to calculate the normal τ_n and shear τ_t components of a point-wise traction vector, which prior to damage initiation is defined by high penalty coefficients K_1 and K_2 as following:

$$\begin{Bmatrix} \tau_t \\ \tau_n \end{Bmatrix} = \begin{Bmatrix} K_1 \Delta u_t \\ K_2 \Delta u_n \end{Bmatrix} \tag{48}$$

The formulation of the MIC UEL requires a residual load vector often denoted as $\{\mathbf{RHS}\}$ in Abaqus which is obtained in the following expression:

$$\{\mathbf{RHS}\} = \int_{V_{elem}} |\nabla \tilde{H}| J [\mathbf{Y}]^T \begin{Bmatrix} \tau_t \\ \tau_n \end{Bmatrix} dV \tag{49}$$

where J is the Jacobian matrix. The expression for the tangent stiffness matrix can be obtained

by standard approach based on differentiation as in Turon [7]. In the present implementation

secant matrix is used instead for stable convergence in lieu of efficiency.
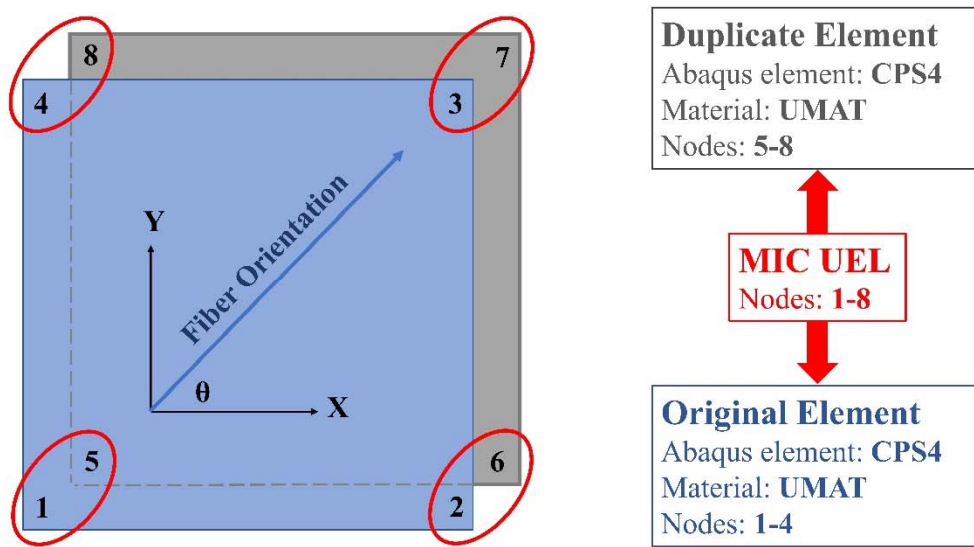
Figure 10. The MIC implementation by superimposing two native Abaqus elements connected by a MIC UEL.

4-2 Implementation of Delamination

In this section, another Abaqus UEL called "interface UEL" connecting two plies with

enriched displacement approximation in Eq. (41) is described. Figure 11 shows the schematics

of the interface UEL as well as the normal (Δ_2, Δ_4) and shear (Δ_1, Δ_3) components of the

displacement jump in the local coordinates in two-dimensional implementation. Suppose a matrix crack inserted by the MIC modeling exists in the top ply and bottom ply as well as the associated regularized Heaviside step function $\tilde{H}^{Top}$ and $\tilde{H}^{Bottom}$, respectively. We will define the values of the regularized Heaviside step function in the nodes of the original element in each ply first, and denote them as $\tilde{H}_1^{Bottom}$, $\tilde{H}_2^{Bottom}$ and $\tilde{H}_3^{Top}$, $\tilde{H}_4^{Top}$. The respective values of the regularized Heaviside step function in the nodes of the duplicate element are then computed as $(1 - \tilde{H}_1^{Bottom})$, $(1 - \tilde{H}_2^{Bottom})$ and $(1 - \tilde{H}_3^{Top})$, $(1 - \tilde{H}_4^{Top})$. We will further omit the superscript *Top* and *Bottom* in the following equations since we have unique nodal indices for each ply eliminating confusion as to which ply the regularized Heaviside step function belongs. Therefore, the displacement jump between top and bottom ply can be computed by taking into account Eq. (39) and Eq. (41), and the notations described above as following:

$$\Delta_1 = \left(\tilde{H}_4 u_{4l} + (1 - \tilde{H}_4) u_{8l} \right) - \left(\tilde{H}_1 u_{1l} + (1 - \tilde{H}_1) u_{5l} \right)$$

$$\Delta_2 = \left(\tilde{H}_4 v_{4l} + (1 - \tilde{H}_4) v_{8l} \right) - \left(\tilde{H}_1 v_{1l} + (1 - \tilde{H}_1) v_{5l} \right)$$

$$(50)$$

$$\Delta_3 = \left(\tilde{H}_3 u_{3l} + (1 - \tilde{H}_3) u_{7l} \right) - \left(\tilde{H}_2 u_{2l} + (1 - \tilde{H}_2) u_{6l} \right)$$

$$\Delta_4 = \left(\tilde{H}_3 v_{3l} + (1 - \tilde{H}_3) v_{7l} \right) - \left(\tilde{H}_2 v_{2l} + (1 - \tilde{H}_2) v_{6l} \right)$$

where u_{1l} to u_{8l} and v_{1l} to v_{8l} are the nodal displacements of normal and shear components in the local coordinates. On the basis of the Rx-FEM theory, if there is no matrix crack insertion in each ply the values of $\widetilde{H}^{Top}$ and $\widetilde{H}^{Bottom}$ in the nodes of the original element in each ply are equal to 1, and the respective values of $(1 - \widetilde{H}^{Top})$ and $(1 - \widetilde{H}^{Bottom})$ in the nodes of the duplicated element in each ply are computed as 0. Then the expressions of the interface UEL displacement jump in Eq. (50) reduces to the standard surface-based cohesive zone formulation without any matrix cracks. This interface UEL contains nodes 1 to 8, and calculates the cohesive traction as a result of separation. The separation or displacement jump can be computed as follows:

$$\begin{Bmatrix} \Delta u_t \\ \Delta u_n \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & N_2 & 0 \\ 0 & N_1 & 0 & N_2 \end{bmatrix} \{\Delta_1 \ \Delta_2 \ \Delta_3 \ \Delta_4\}^T$$

$$(51)$$

$$N_1 = \frac{1-\zeta}{2}, N_2 = \frac{1+\zeta}{2}$$

where the shape function matrix $[N]$ in the natural coordinates (ζ) is that of the original or duplicate element which are identical, and the Δu_n and Δu_t are the normal and shear components of the displacement separation vector along the delamination direction. Next, the point-wise cohesive traction $\{\tau\}$ related to the displacement separation can be obtained by following the Eq. (11) which uses the mixed-mode surface-based cohesive zone model by Turon [7]. The normal τ_n and shear τ_t components of the cohesive traction prior to the damage initiation can be calculated by using Eq. (48). The global X and Y degrees of freedom of the

interface UEL contain 16 components corresponding to 8 nodes in two-dimensional implementation, which for the node numbering on Figure 11 are numbered in the same sequence as in Eq. (45). The residual load vector {**RHS**} of the interface UEL, which is required to be defined in Abaqus, can then be written as following:

$$\{\mathbf{RHS}\} = \int_{V_{elem}} J[\mathbf{Y}]^T \begin{Bmatrix} \tau_t \\ \tau_n \end{Bmatrix} dV \tag{52}$$

and

$[\mathbf{Y}]$

$$= \begin{bmatrix} -\tilde{H}_1 CN_1 & -\tilde{H}_1 SN_1 & -\tilde{H}_2 CN_2 & -\tilde{H}_2 SN_2 & \tilde{H}_3 CN_2 & \tilde{H}_3 SN_2 & \tilde{H}_4 CN_1 & \tilde{H}_4 SN_1 \\ \tilde{H}_1 SN_1 & -\tilde{H}_1 CN_1 & \tilde{H}_2 SN_2 & -\tilde{H}_2 CN_2 & -\tilde{H}_3 SN_2 & \tilde{H}_3 CN_2 & -\tilde{H}_4 SN_1 & \tilde{H}_4 CN_1 \end{bmatrix}$$

$$\begin{matrix} (-1+\tilde{H}_1)CN_1 & (-1+\tilde{H}_1)SN_1 & (-1+\tilde{H}_2)CN_2 & (-1+\tilde{H}_2)SN_2 \\ (1-\tilde{H}_1)SN_1 & (-1+\tilde{H}_1)CN_1 & (1-\tilde{H}_2)SN_2 & (-1+\tilde{H}_2)CN_2 \end{matrix} \tag{53}$$

$$\begin{matrix} (1-\tilde{H}_3)CN_2 & (1-\tilde{H}_3)SN_2 & (1-\tilde{H}_4)CN_1 & (1-\tilde{H}_4)SN_1 \\ (-1+\tilde{H}_3)SN_2 & (1-\tilde{H}_3)CN_2 & (-1+\tilde{H}_4)SN_1 & (1-\tilde{H}_4)CN_1 \end{matrix}$$

where [**Y**] is the global displacement-separation relation matrix which can be obtained by following the same procedure as in the previous section, and the, however, in this case $C=\cos \varphi$ and $S=\sin \varphi$, where $(\sin \varphi, \cos \varphi)$ is the normal vector of the ply surface in the X and Y coordinates. When top and bottom plies start to move apart, the displacement jump between plies will be created, which generates the cohesive tractions and eventually softening and full separation.
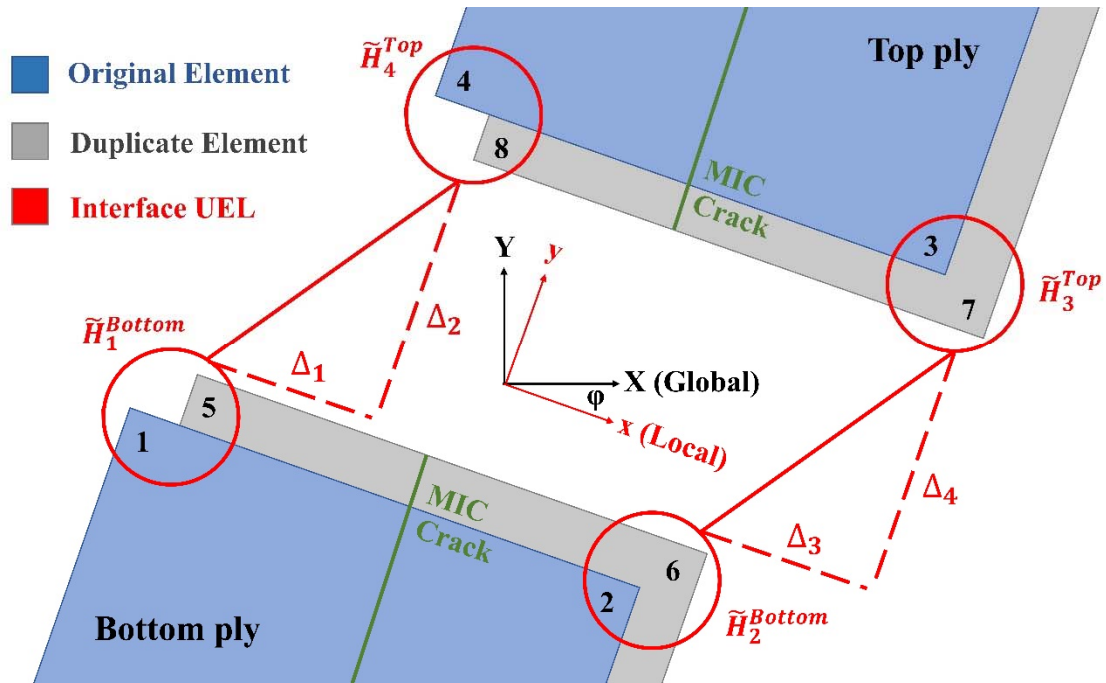
Figure 11. An interface UEL used to connect four native Abaqus elements in the top and bottom plies with a MIC crack in each ply.

4-3 Solution Algorithm

We begin by discussing the implementation of the Rx-FEM kinematic discontinuity and follow up with the overall implementation algorithm. The numerical simulation begins without any initial matrix cracks, i.e. displacement discontinuity built into the model. As the loading is increased, the matrix cracks oriented parallel to the fiber orientation are inserted according to a failure criterion LaRC04 failure criterion [44] evaluated at each integration point at the end of each loading. If the failure criterion is exceeded, a level set function of the signed distance function of the matrix crack surface is introduced, and a regularized Heaviside step function associated with this level set is calculated for elements involved. The step function in

original and duplicate elements as well as its gradient in the MIC UEL are shown below for a 9 elements example in Figure 12.
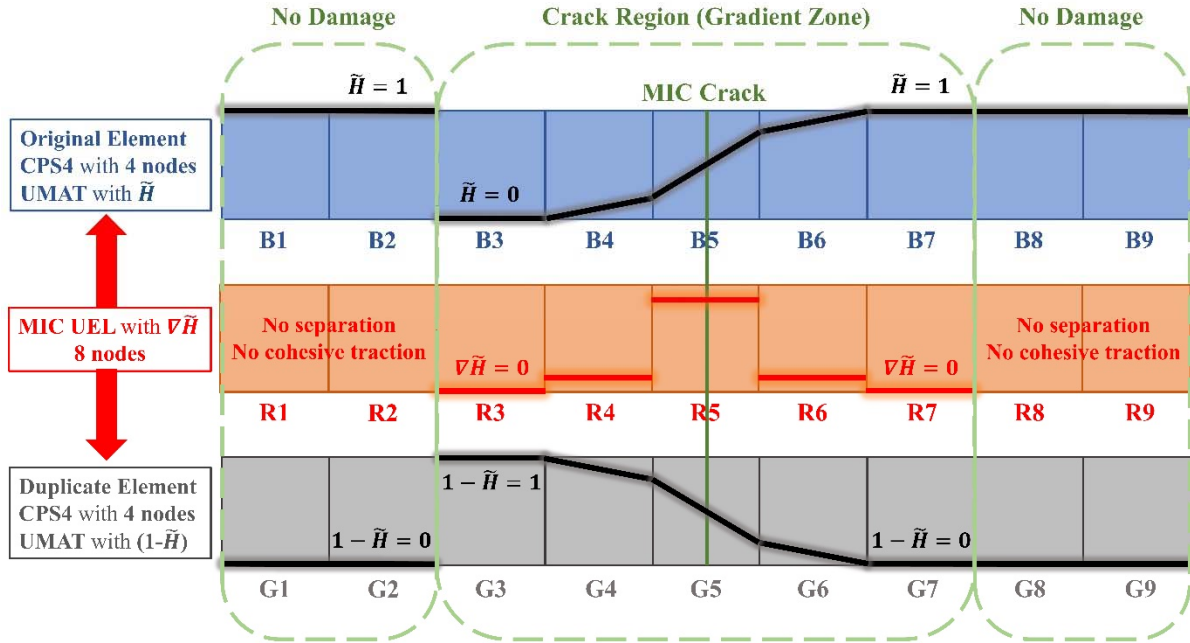
Figure 12. The schema of the regularized Heaviside step function and its gradient within a crack region.

Prior to crack insertion the step function in all elements was as in the "No Damage" region. The original element had an unmodified stiffness and the duplicate element had zero stiffness, which created no resistance nor separation in the nodes between the two models. Note that with minimal control of the stiffness matrix assembly in the host software the duplicate elements and the MIC UELs in the "No Damage" region can be excluded from assembly eliminating any overhead. Due to this limitation the MIC UELs in the "No Damage" region

are set to tie the original and duplicate models, which in this zone contains zero stiffness elements. When the crack is inserted and the "Crack Region" appears, the stiffness of both the original and duplicate element is nonzero and they start moving independently. Note that the crack region is one element larger than the gradient zone on each side, i.e. in the elements R3 and R7 ($\nabla \widetilde{H} = 0$). The presence of these elements in the "Crack Region" is required for modeling an arbitrary number of cracks in the model. While no manipulation of nodal connectivity is required nor allowed in commercial FE, these elements facilitate the load transfer in and out of the duplicate model. As one can see at the boundaries B1, G1 and B9 and G9 the elements of the duplicate model have zero stiffness. The applied external load, in this case axial tension is transferred from the left side through elements B1 and B2, however, the next element B3 has 0 stiffness thus not transferring any load to B4. The B3 on the other hand is tied to element G3 effectively transferring the loading into the duplicate model elements. The load from the right-hand side is transferred through the original element B4-B9. Thus, if the cohesive force in the gradient zone the MIC UELs R4-R6 has been overcome, then we have the left and right-hand side of the model independent of each other with active elements B1, B2, G3-G6 on one side and B4-B9 on the other. Such setup allows to replace any of the "No Damage" regions with another "Crack Region" thus modeling multiple cracking.

The solution algorithm of a typical simulation is shown in Figure 13. The left part of the Figure 13 is the flow chart of the execution inside the Abaqus/Standard, and the right part of the Figure 13 is the external code which communicates with Abaqus by using user subroutines. In the model preparation step, both original and duplicate models which are connected by the MIC UELs in each ply and the interface UELs between plies must be built into the full model first before the start of analysis. Once the analysis starts, the MIC database

is initiated at the first call of the UEXTERNALDB before the first increment and will be updated on each load increment. During the update we perform pointwise evaluation of the failure criterion [44]. If it is exceeded, then a signed distance function for the crack surface, regularized step function and its gradient are calculated and passed to the respective elements of the original and duplicate models according to schematics on Figure 12. After the load is incremented a solution on a new step is converged and the process repeated. Addition of the interface UEL and extension to multiple plies requires no modification of the algorithm on Figure 13.
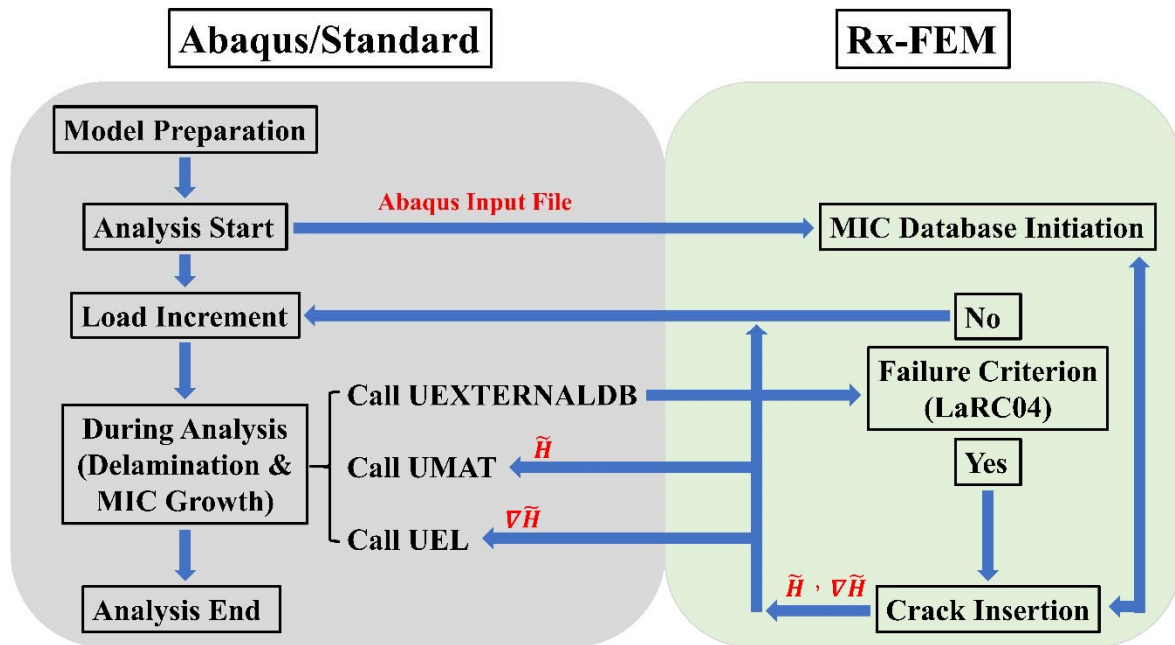
Figure 13. The framework for the proposed implementation of the Rx-FEM methodology in Abaqus.

CHAPTER V

VERIFICATION MODELS

In this chapter, several numerical models devoted to verification and application of the proposed implementation in laminated composites are presented below. Uniaxial tension example of MIC implementation illustrated previously in Figure 12 is examined first. Validation of the transverse matrix crack insertion and propagation is performed for off-axis pristine and open-hole tensile (OHT) unidirectional composites. Second, several benchmark examples with analytical solutions including a double cantilever beam (DCB) test, an end-notched flexure (ENF) test and a transverse crack tension (TCT) test are performed to verify the accuracy of the interface UEL for mode I, mode II and the interaction between an initial matrix crack and delamination, respectively. Finally, the bending test of a pristine composite beam with 0° and 90° plies is considered to examine all the capabilities of the proposed implementation for modeling the progressive damage analysis.

5-1 Uniaxial Tension of a 9 Elements Bar

In this example, a bar containing a row of 9 elements length with an initial MIC insertion in the middle under uniaxial traction is simulated as shown in Figure 14. The low initiation strength and high fracture toughness of the cohesive properties are considered for easy convergence. The stiffness properties are isotropic and plane stress conditions imposed via CPS4 elements. This model appears to have 9 elements but contains 27 elements. This is

the same configuration as in Figure 12 which includes 9 original elements, 9 duplicate elements, and 9 MIC UELs. After increasing load to completely separation of the bar, the final configuration appears to have 10 elements in Figure 14(b). Here and below the visualization is performed within Abaqus by using the state variable parameters (SDVs), which allow to display or hide elements which satisfy certain criterion defined in Abaqus/CAE. In the present and following models within the crack region, only the original and duplicate elements which have $\widetilde{H}(x) > 0.5$ at any integration point in the element are shown. Corresponding to the Figure 12, the 5 elements of left-hand side shown from left to right in Figure 14(b) are active elements B1, B2, G3, G4, G5, whereas active element G6 is hidden. Similarly, the right-hand side of the same figure displays active elements B5-B9, in this case element B4 is hidden. Note that the MIC UELs do not require visualization since all the kinematics are accommodated by Abaqus native elements.
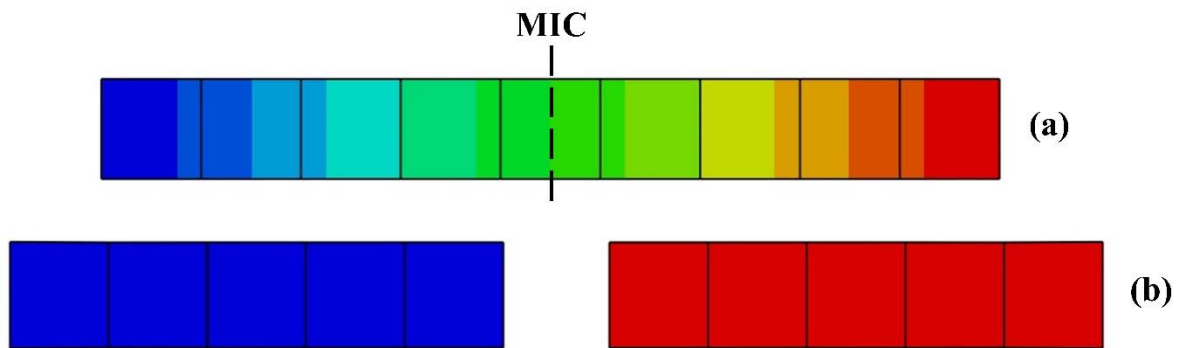
Figure 14. The displacement field of the uniaxial tensile test of a 9 elements bar for (a) pristine and (b) failure.

5-2 Off-Axis Test of Unnotched Composites

The geometry of unidirectional unnotched composites in this test is shown in Figure 15(a). The ratio of the unnotched length 125 mm to width 25 mm (L/W) is 5, and the structured mesh size is 1 mm. Ply-level properties of composites used for the analysis are shown in Table 1 including the stiffness, strength properties, as well as the fracture toughness required for the cohesive law. The matrix crack initiation is determined by checking the ply-level failure criterion, LaRC04 [44], at each integration point in the end of each increment. If the failure criterion is met or exceeded, one or more MICs are inserted at that location. The matrix crack is assumed to be planar, parallel to the fiber orientation, and extend to the edges of the model; however, it is introduced in the fully closed state.
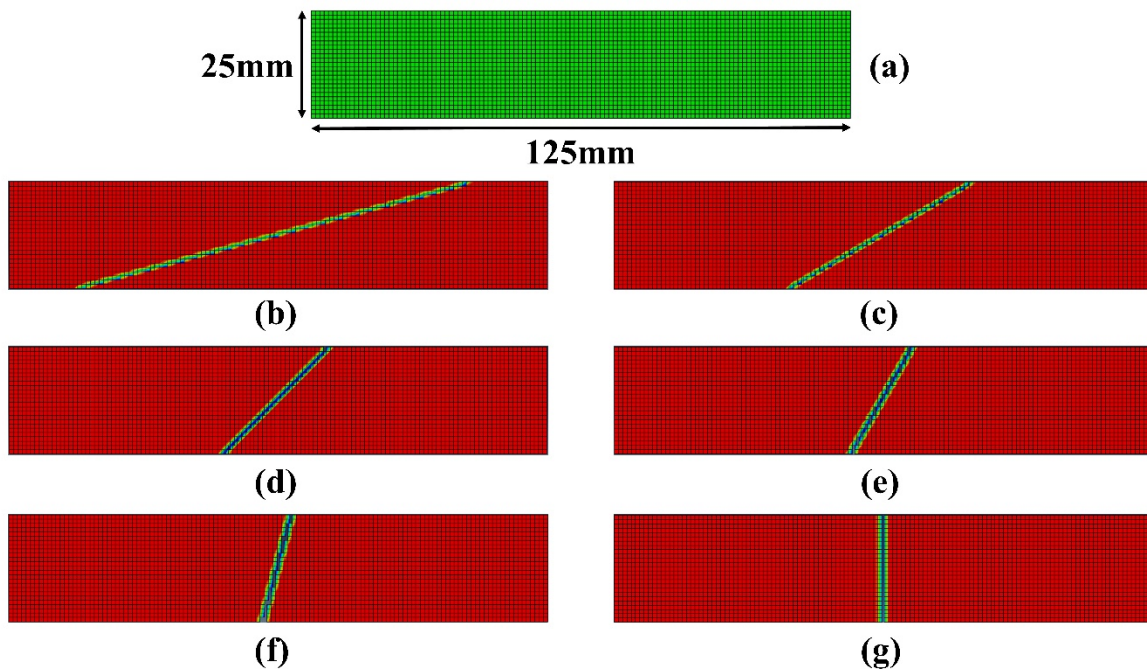
Figure 15. The off-axis test of unnotched composites. (a) Geometry, (b) 15°, (c) 30°, (d) 45°, (e) 60°, (f) 75°, (g) 90°.

50

Table 1. Unidirectional ply stiffness and strength properties.

Property	Value
E_{11} (MPa)	157600
E_{22} (MPa)	8977
G_{12} (MPa)	5000
ν_{12}	0.32
Y_t (MPa)	92
S (MPa)	89
G_{IC} (N/mm)	0.24
G_{IIC} (N/mm)	0.739

Figure 15(b)-(g) display the regularized Heaviside step function $\tilde{H}(x)$ distribution for the MIC inserted for six different fiber angles (15°, 30°, 45°, 60°, 75° and 90°) when reaching the failure criterion of the matrix crack initiation. The visualization is performed according to rules as described in previous section thus in the bulk of the specimen away from the crack $\tilde{H}(x) = 1$ and $0.5 < \tilde{H}(x) < 1$ in the gradient zone. As expected from physical considerations these specimens exhibit unstable two pieces failure along the crack surface immediately after crack insertion, which was observed in progressive failure analysis simulations as well. Figure 16 shows the predicted failure strength as well as the failure envelop corresponding to LaRC04 failure criterion [44], which coincide and thus validate the MIC insertion portion of the Rx-FEM implementation.
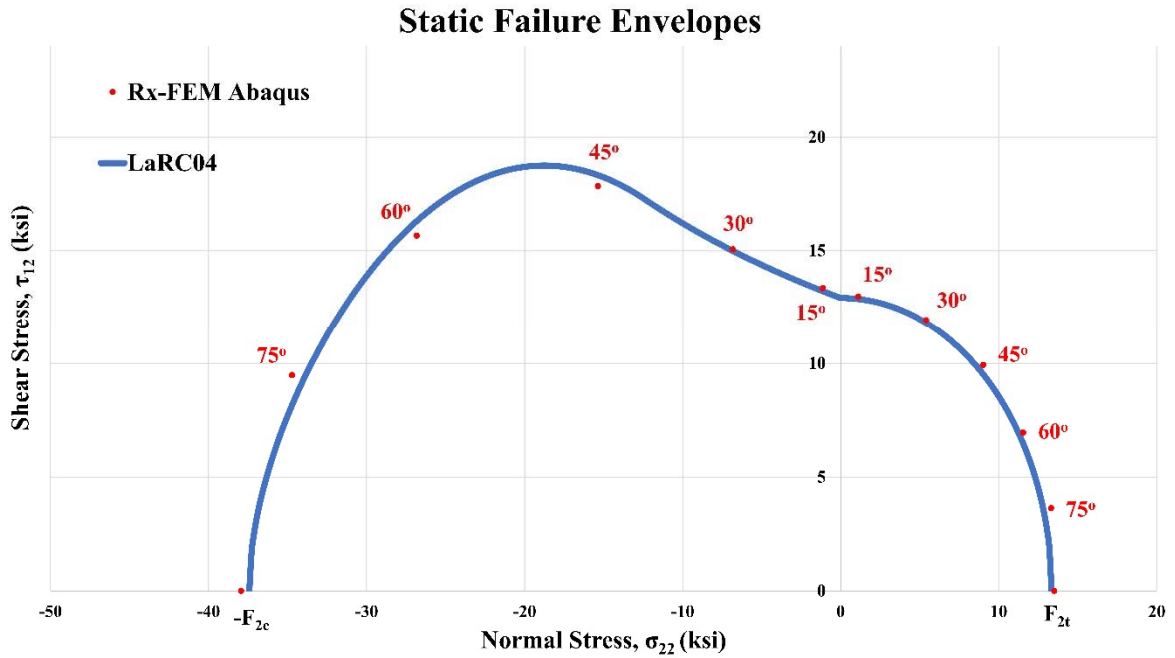
Figure 16. The static failure envelop simulation from the proposed implementation of the Rx-FEM in Abaqus.

5-3 Unidirectional Composites with Open Hole

A rectangular unidirectional composite plate with a central hole under uniaxial tension is considered in this next. The material properties of the OHT coupons used for the analysis are identical to the off-axis test as shown in Table 2. The diameter of the hole was 6.375 mm (1/4 inch) and the ratio of width to hole diameter (W/D) and length to diameter (L/D) are 6 and 18 respectively as shown in Figure 17. Three coupons with 90°, 45° and 0° fiber orientations are examined. The structured mesh size is 1 mm for 90° and 45° coupons and 0.5 mm for 0° coupon. The stress and displacement fields in the pristine and failed specimens are examined. In the cases of the 90° and 45° coupons, the normal transverse stress component σ_{22} and the u_x

displacement are shown in Figure 18 and Figure 19, whereas for the 0° coupon the fiber direction σ_{11} stress and u_x are shown in Figure 20.

108mm

Hole diameter: 6mm
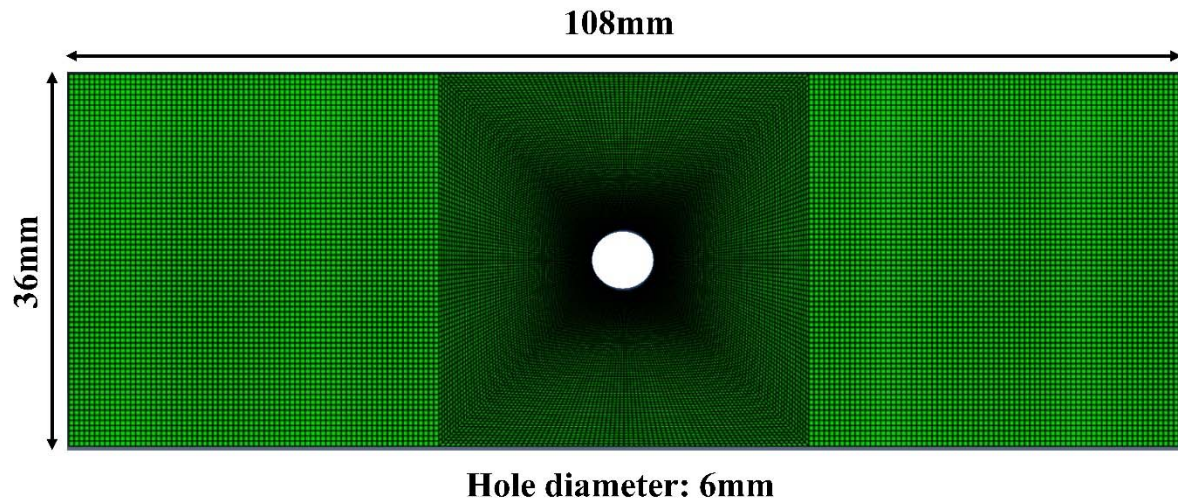
Figure 17. Geometry of the OHT unidirectional composites

In both cases of the 90° and 45° coupons, as one can see the crack insertion locations coincide with the transverse stress concentration location as shown on the right of the Figure 18 and Figure 19. One MIC on each side of the hole is inserted, and these coupons with two matrix cracks are separated immediately after increasing the load. The figure of stress distribution after separation is not shown here since the stress of specimen vanishes right after fully separation. The displacement distribution after separation indicates a complete two pieces failure as shown in Figure 18(c) and Figure 19(c). Note that the ability to readily display the displacement field, stress field and separation of the specimens by using standard Abaqus visualization options can only be achieved by using native Abaqus elements for simulation.

Table 2. Unidirectional ply stiffness and strength properties.

Property	Value
E_{11} (MPa)	157600
E_{22} (MPa)	8977
G_{12} (MPa)	5000
v_{12}	0.32
Y_t (MPa)	92
S (MPa)	89
G_{IC} (N/mm)	0.24
G_{IIC} (N/mm)	0.739
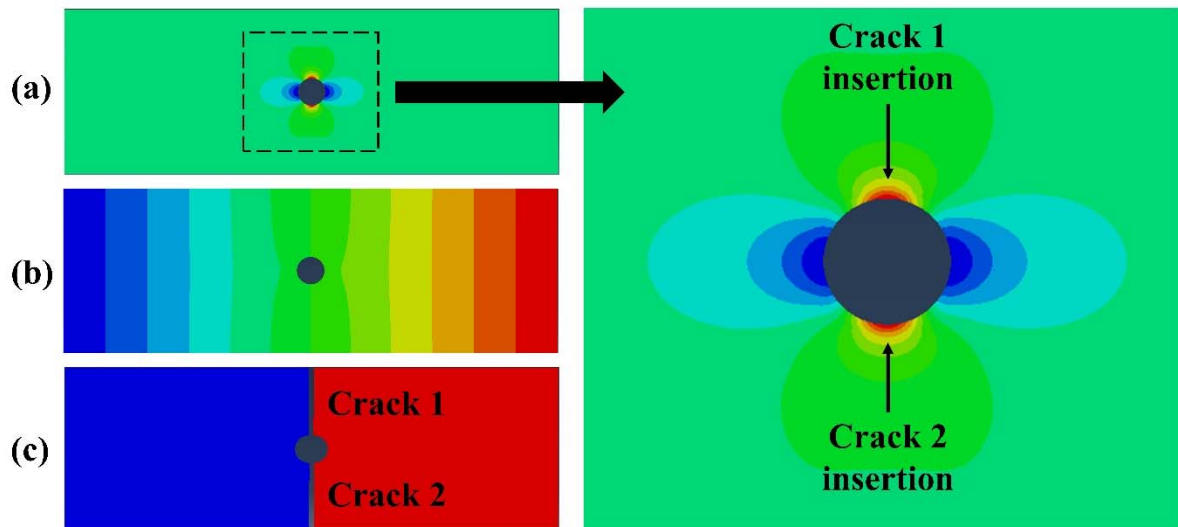
Figure 18. Numerical results of the OHT specimen with 90° fiber orientation. (a) Pristine σ_{22} stress field, (b) pristine u_x displacement field, and (c) failure u_x displacement field.
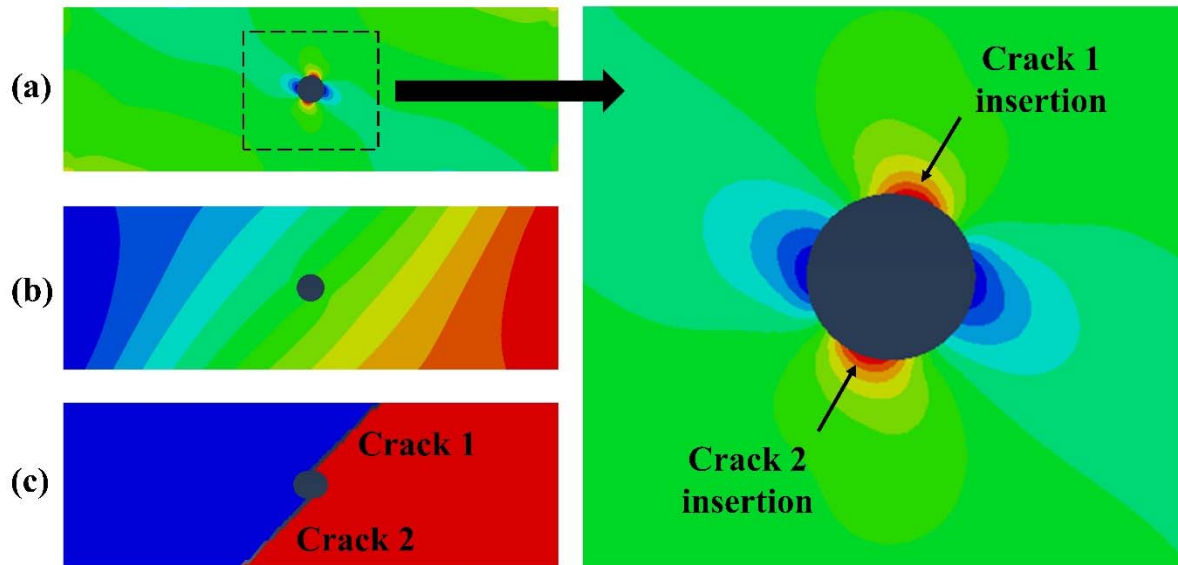
Figure 19. Numerical results of the OHT specimen with 45° fiber orientation. (a) Pristine σ_{22} stress field, (b) pristine u_x displacement field, and (c) failure u_x displacement field.

Next a 0° coupon is considered. In this case, 4 longitudinal cracks are inserted at the locations corresponding to the LaRC04 failure criteria [44] at approximately ±78° angle with respect to the loading direction at the edge of the hole. The axial displacement field u_x before the crack insertion and after separation are shown in Figure 20(a) and Figure 20(b). As one can see the specimen separates into 4 part which include the outer ligaments and two middle ligaments. Two middle ligaments are pulled out, whereas the outer ligaments are displaying linear displacement patterns as shown in Figure 20(b). The stress fields corresponding to such kinematics are represented in Figure 20(c) and Figure 20(d). One can observe a high fiber stress concentration at the ±90° locations at the edge of the hole before the coupon splitting as shown in Figure 20(c). After separation, as seen in Figure 20(d), the top and bottom ligaments

are uniformly loaded with no stress concentration, whereas the middle ligaments are completely unloaded. This splitting phenomena is very important for understanding various failure patterns and strength effects in laminated composites and needs to be accurately modeled on the ply level [45].
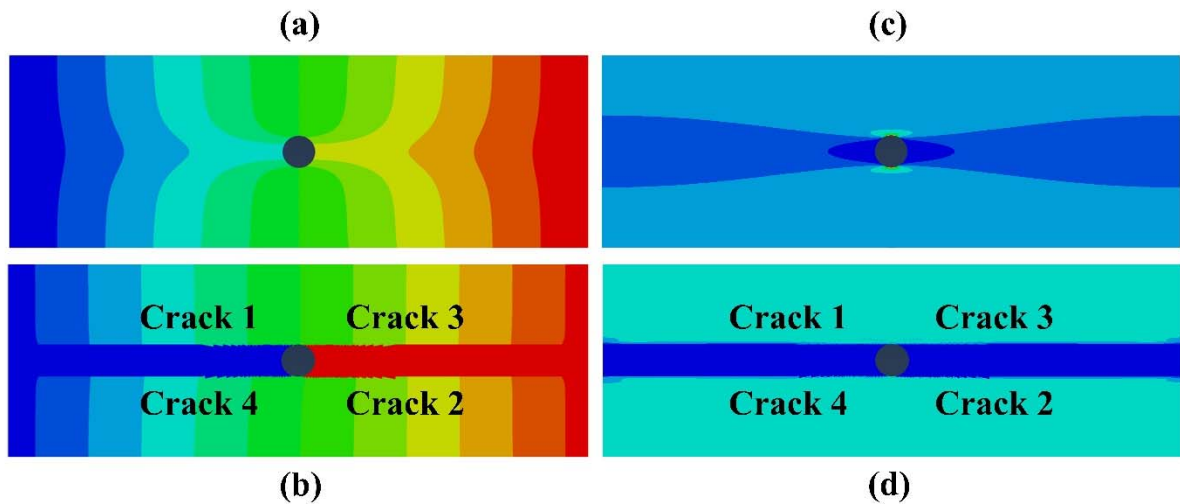
Figure 20. Numerical results of the OHT specimen with 0° fiber orientation. (a) Axial displacement field u_x before cracking and (b) after separation. (c)Axial stress field σ_{11} before cracking and after separation.

Figure 21 displays the average traction versus applied strain and normalized crack length versus applied strain which is also compared with the experiment data. When the applied strain is around 0.0016, four cracks are inserted and their opening leads to the stiffness reduction. When the applied strain is increasing, the crack 1 and 2 start propagating first followed by the crack 3 and 4 and the trend has a good agreement compared with the experiment data.

56
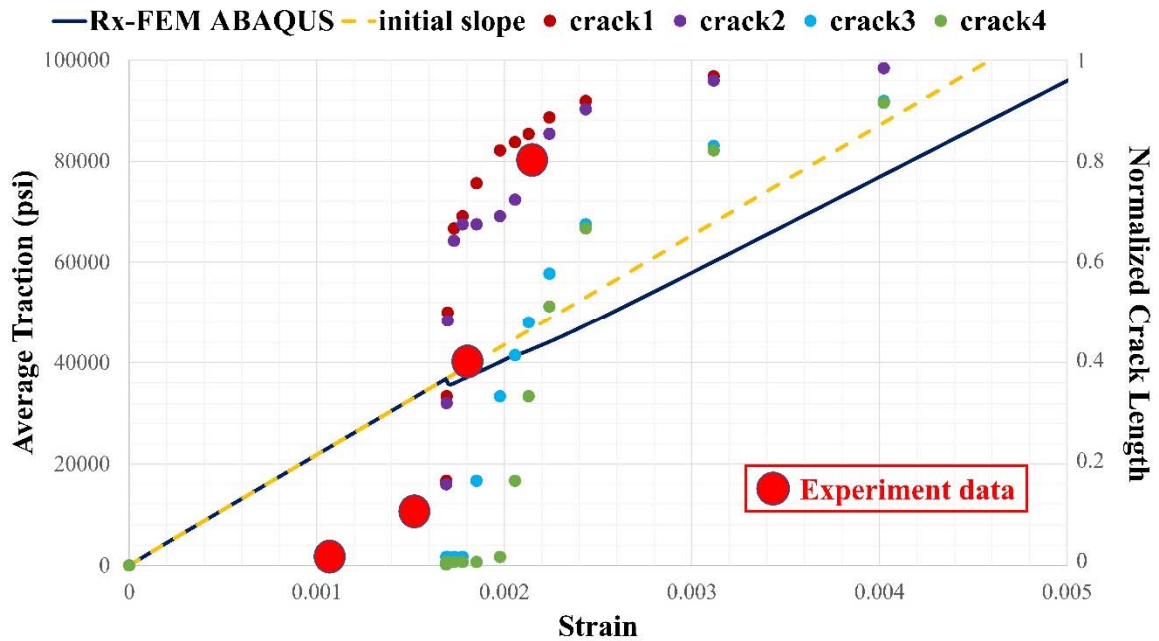
Figure 21. Numerical results of both average traction and normalized crack length of each crack versus applied strain, and comparison between numerical results and experiment data for normalized crack length of each crack versus applied strain.

5-4 Double Cantilever Beam Test

The DCB benchmark test [46] is used to verify the performance of the interface UEL in the proposed Abaqus implementation under pure mode I fracture. As shown in Figure 22, the dimensions of the DCB specimen are length L = 50 mm, thickness h = 3 mm. The width is 20 mm, and the right end of the DCB model is a fixed boundary condition. An initial crack length a = 30 mm is created, and the material properties are given in Table 3.
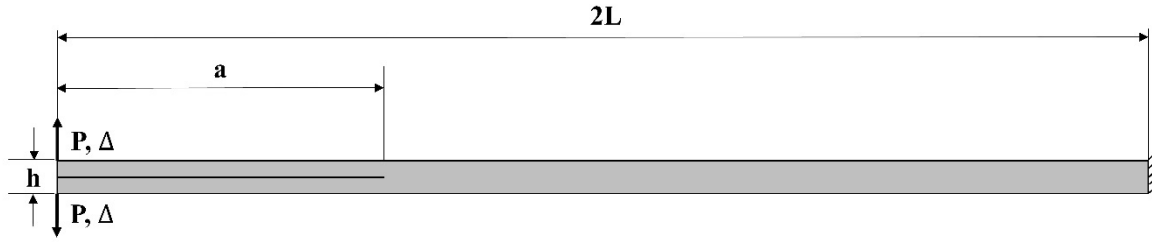
Figure 22. Geometry of the DCB specimen.

Table 3. Material properties for the DCB and ENF models.

Property	Value
E_{11} (MPa)	135300
E_{22} (MPa)	9000
G_{12} (MPa)	5200
v_{12}	0.32
Y_t (MPa)	57
S (MPa)	90
G_{IC} (N/mm)	0.28
G_{IIC} (N/mm)	0.63

The structured mesh size 0.2 mm along the interface and 0.3 mm through the thickness are sufficiently fine. The top beam of the DCB specimen, as well as the bottom beam, contains the native Abaqus plane strain elements CPE4 used for both original and duplicate models which are connected by the MIC UELs. The bottom surface of the top beam and the top surface

of the bottom beam are connected by using the interface UELs except for the initial crack surface.
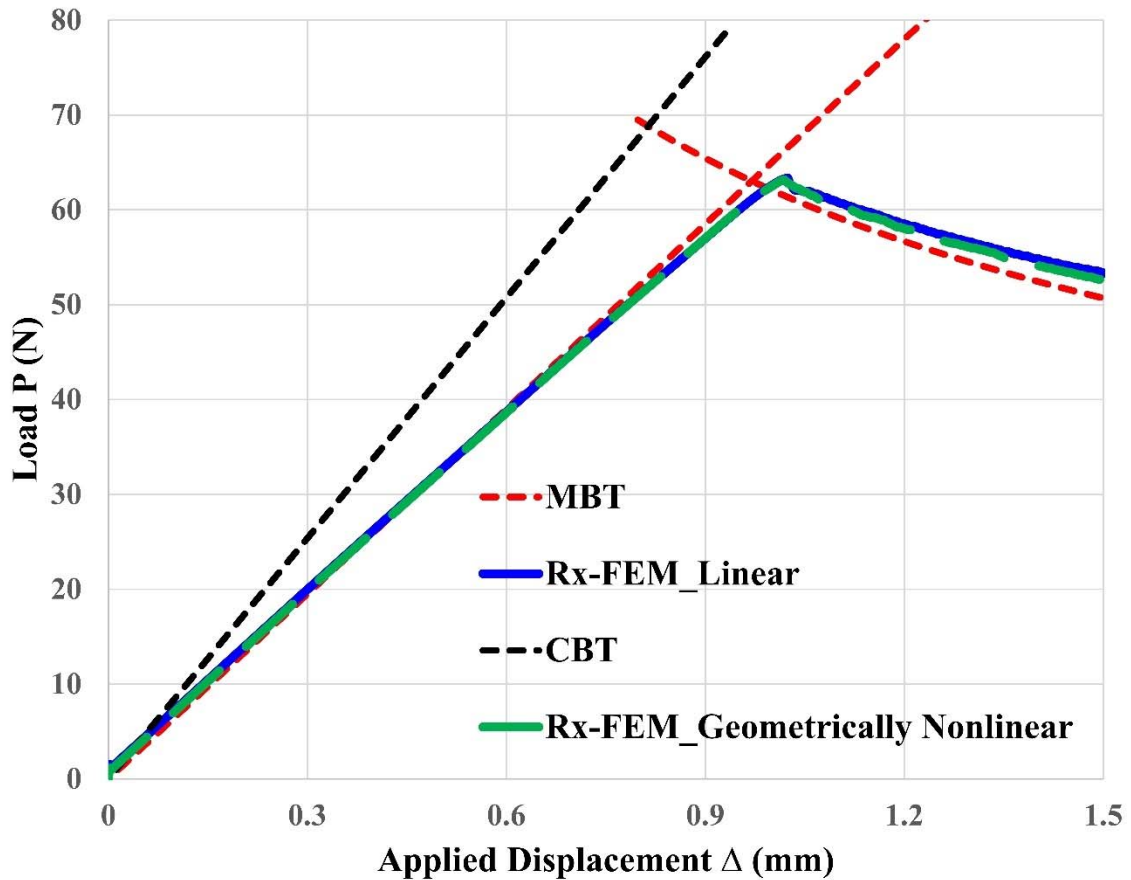
Figure 23. Load-displacement curve of the DCB test.

While not necessarily displaying the MIC, this example, as well as the following example, provides verification of the Mode I and Mode II delamination propagation by using the proposed interface UEL. For the pure mode I static failure analysis, the load-displacement

curve of the DCB test is compared with the analytical solutions which include classical beam theory (CBT) and modified beam theory (MBT) as shown in Figure 23. As a result, the numerical simulation has a good agreement with the MBT analytical solution in the DCB test. The proposed implementation which is based on utilization of the native Abaqus elements allows to take advantages of the features of the host software, e.g. geometrically nonlinear analysis. In this particular case, numerical solutions by using geometrically linear and nonlinear analysis were performed and show similar results as expected.

5-5 End-Notched Flexure Test

The ENF benchmark test [46] is used to verify the performance of the proposed interface UEL under pure mode II failure. As shown in Figure 24, the dimensions of the ENF specimen and the material properties are the same as the DCB specimen. The left end node and right end node of the ENF bottom surface are the pinned and roller boundary conditions applied, respectively. The Abaqus CPE4 elements with 0.2 mm length and 0.3 mm thickness mesh size are adopted to the ENF specimen which is identical to the DCB model configuration, where the original and duplicate elements are tied by the MIC UELs and the interface UELs are set to connect between the top beam and bottom beam along the potential crack propagation path excluding the initial crack surface.
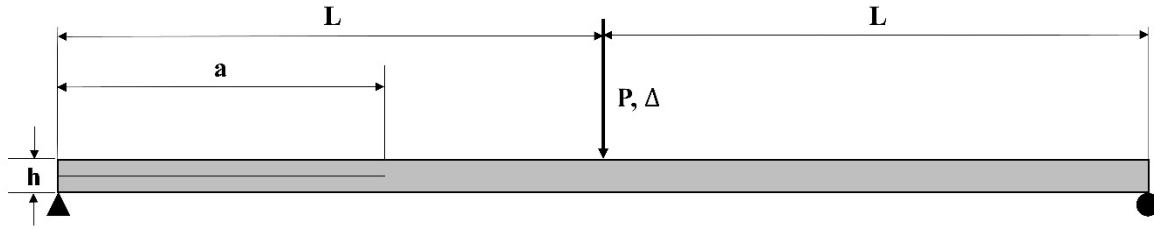
Figure 24. Geometry of the ENF specimen.

In the present case, another capability of the host software which is the Abaqus built-in contact interactions is added between the bottom surface of the top beam and the top surface of the bottom beam. Contact properties with "Frictionless" for tangential behavior and "Hard contact" for normal behavior are assigned to four different contact pairs including:

- The bottom surface of the original top beam and the top surface of the original bottom beam

- The bottom surface of the original top beam and the top surface of the duplicate bottom beam

- The bottom surface of the duplicate top beam and the top surface of the original bottom beam

- The bottom surface of the duplicate top beam and the top surface of the duplicate bottom beam

For the pure mode II static failure analysis, the load-displacement curve of the ENF test is compared with the analytical solution as shown in Figure 25. It can be seen that the numerical result has a good agreement with the analytical solution for this ENF analysis. The deformed shape of the ENF specimen is shown in Figure 26 displaying the displacement fields and

sliding contact between top and bottom beams thus proving the ability to use host software

contact capability within Rx-FEM implementation. Similarly, the DCB case the computations

were carried out in geometrically linear and nonlinear formulations.
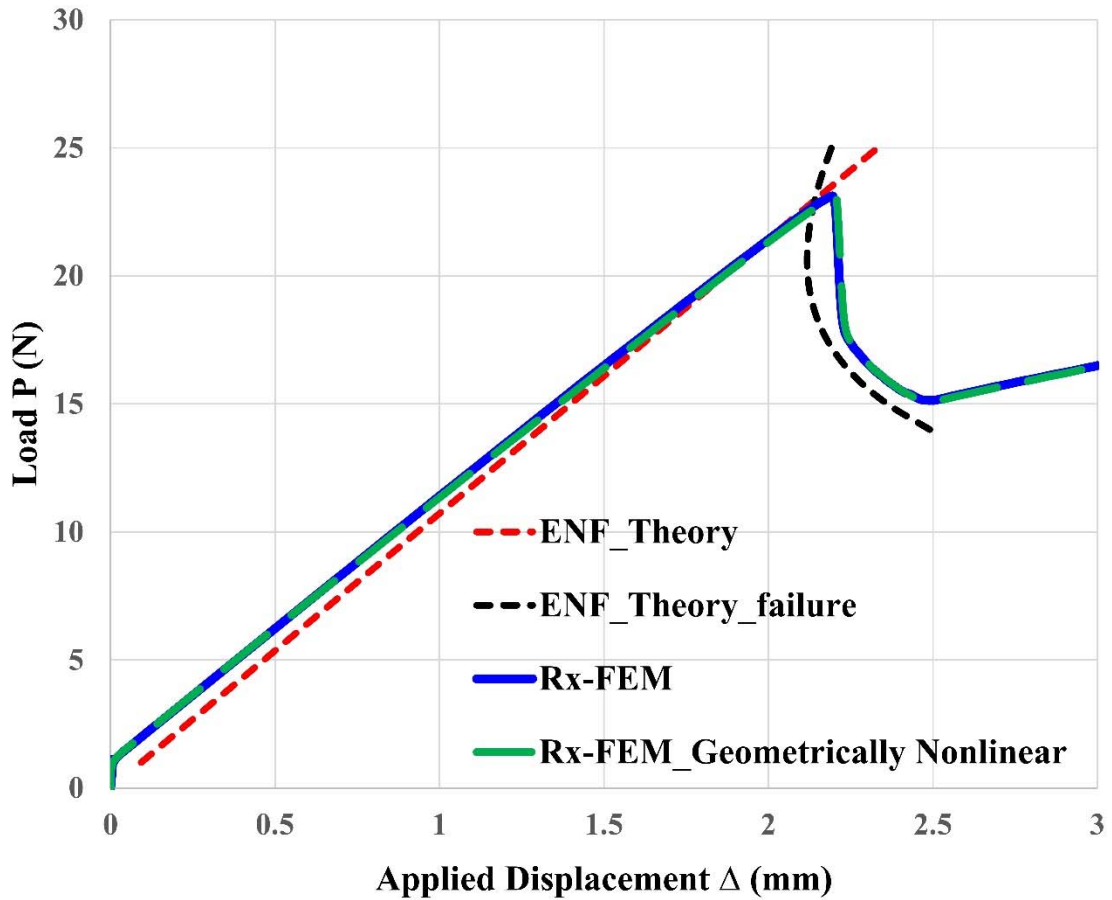
Figure 25. Load-displacement curve of the ENF test.
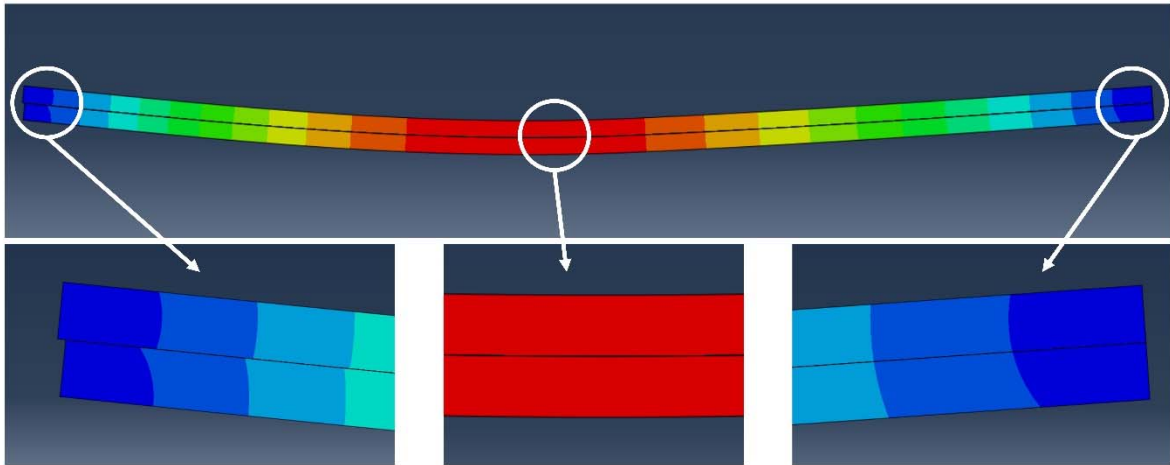
Figure 26. Deformed shape of the ENF specimen displaying the contact between delaminated plies.

5-6 Transverse Crack Tension Test

The TCT specimen, described in reference [47], is examined to evaluate the accuracy of the proposed implementation for predicting the interaction between an initial matrix crack and delamination. As shown in Figure 27, the TCT model consists of three 0° unidirectional plies with thickness h, 2h and h from top ply, middle ply to bottom ply. The dimensions of the TCT specimen are length L = 120 mm, h = 1 mm and width = 32 mm. An initial crack is cut through the width at the middle of the middle ply. The potential delamination propagation paths including the top-middle surface and middle-bottom surface are modeled by using the interface UEL previously described. The contact interaction pairs of the Abaqus built-in capability are also activated for those interfaces, where 4 contact pairs are set for the top-middle surface and the other 4 contact pairs for the middle-bottom surface. The TCT model is subjected to uniaxial tensile loading, and the material properties used in this test are

summarized in Table 4. The Abaqus plane strain elements (CPE4) for the original and duplicate elements are used with mesh size 0.5 mm length and 0.5 mm through the whole thickness which are sufficiently fine. Geometric linear analysis is applied in the Abaqus/Standard.
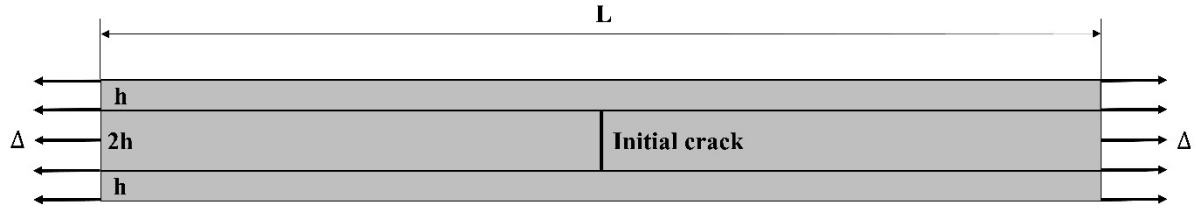
Figure 27. Geometry of the TCT specimen.

Table 4. Material properties for the TCT model

Property	Value
E_{11} (MPa)	139900
E_{22} (MPa)	10100
G_{12} (MPa)	4600
ν_{12}	0.3
Y_t (MPa)	80
S (MPa)	100
G_{IC} (N/mm)	0.12
G_{IIC} (N/mm)	0.5

The applied stress versus applied displacement curve compared with the analytical analysis is shown in Figure 28.
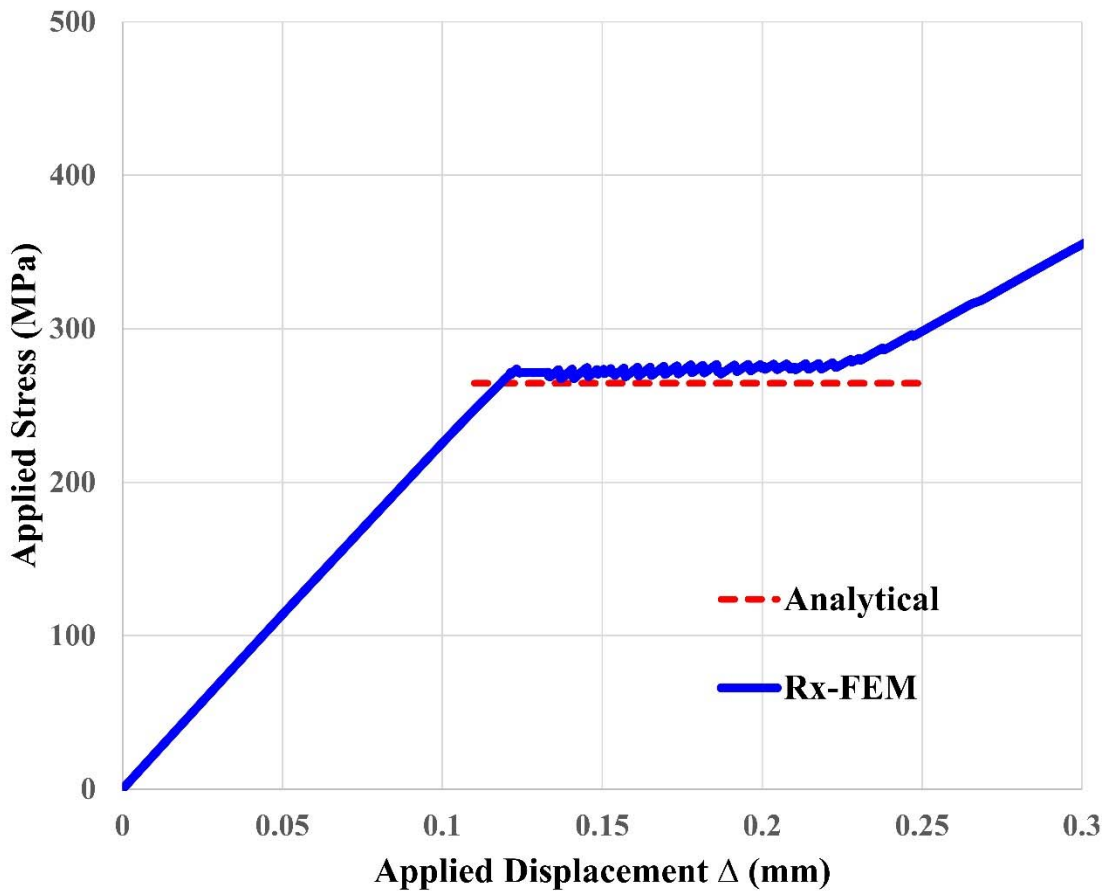
Figure 28. Applied average stress-displacement curve of the TCT test.

When the displacement loading is applied, the applied stress initially increases linearly until the delamination initiation starting from two tips of the middle crack. Then these delaminations start to propagate in a stable manner while the applied stress remains constant.

Finally, the applied stress starts increasing again after the delaminations reach the grips, but with a different slope since the middle ply of the TCT specimen is completely unloaded. As one can see the numerical simulation in the TCT test shows a good agreement with the analytical solution for predicting the applied stress value of the delamination initiation starting from the tips of the matrix crack.

5-7 Bending of 2 plies composite laminate

Consider a 2 plies laminate with L = 50 mm and h = 3 mm consisting of a 0° unidirectional top ply, 90° unidirectional bottom ply and a loading roller with r = 3 mm as shown in Figure 29.
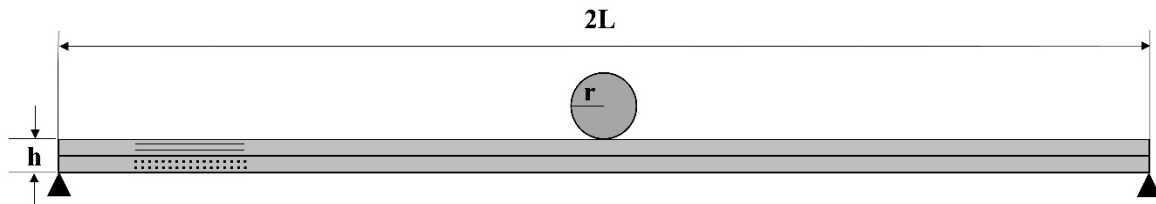
Figure 29. Geometry of the composite laminate.

Both the left end node and right end node of the bottom surface are pinned boundary condition. The material properties are given in Table 5. The Abaqus native element CPE4 for original and duplicate elements, the MIC UEL for MIC propagation, and the interface UEL for delamination are adopted in the model.

Table 5. Material properties for the composite laminate.

Property	Value
E_{11} (MPa)	157600
E_{22} (MPa)	8977
G_{12} (MPa)	5000
v_{12}	0.32
Y_t (MPa)	92
S (MPa)	89
G_{IC} (N/mm)	0.24
G_{IIC} (N/mm)	0.739

The Abaqus built-in contact feature with "Frictionless" and "Hard contact" is applied to six different contact pairs including four pairs for the interface between top ply and bottom ply as discussed previously in the ENF test, and two pairs for the following:

- The bottom surface of the original top beam and the top surface of the original bottom beam

- The bottom surface of the original top beam and the top surface of the duplicate bottom beam

- The bottom surface of the duplicate top beam and the top surface of the original bottom beam

- The bottom surface of the duplicate top beam and the top surface of the duplicate bottom beam

- Roller surface and the top surface of the original top ply

- Roller surface and the top surface of the duplicate top ply

In this test, a roller will apply two step displacement loading on the top surface of the composite laminate.
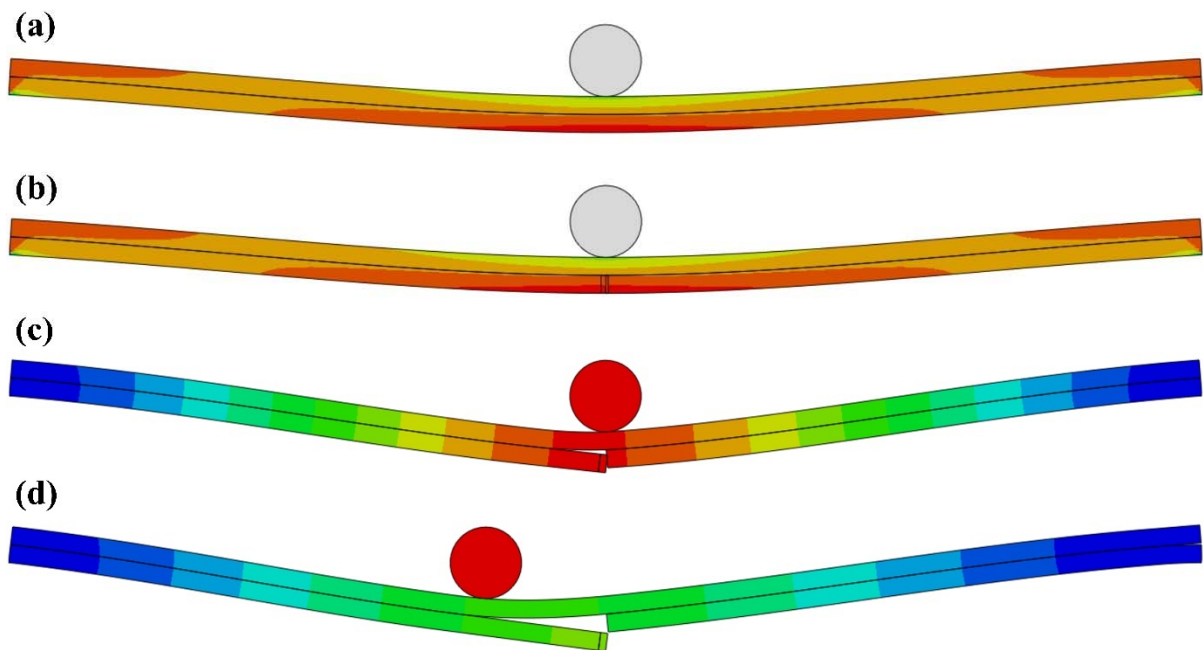
Figure 30. Progressive damage analysis in composite laminate. The σ_{22} stress field just (a) before a MIC insertion and (b) after, and the displacement field (c) in the end of 1st step and (d) 2nd step.

The simulation begins without any initial matrix crack and delamination. In the first step, the roller applies 6 mm displacement loading along the -y direction. Figure 30(b) shows the normal transverse stress component σ_{22} just after a MIC insertion, where the location

coincides with the maximum stress location at the center point of the bottom surface just before the matrix crack initiation as shown in Figure 30(a). As one can see that the MIC forms a matrix crack and the delamination starts to propagation along the ply interface in the end of the first step in Figure 30(c) which shows the displacement field. Please note again that it appears to have a real matrix crack since any element having $\widetilde{H}(x) < 0.5$ at every integration point is hidden by using the SDVs in Abaqus visualization. In the second step, the roller is sliding 10 mm left along the -x direction. When the roller is moving left, further delamination grows as expected as shown in Figure 30(d). The numerical simulation in Figure 30 proves that the proposed implementation has the ability for modeling the progressive damage analysis in laminated composites which consists of the arbitrary matrix crack insertion and propagation, and delamination due to the matrix cracks.

CHAPTER VI

CONCLUSIONS

A successful implementation of the Rx-FEM methodology in the commercial finite element software Abaqus is demonstrated in two-dimensional formulation for modeling progressive damage analysis in laminated composite materials. The proposed implementation consists of the implementation of MIC technique to modeling arbitrary transverse matrix cracks in each ply of the composite laminate, and delamination between plies by using mixed-mode cohesive formulation which is also used to describe the MIC matrix crack propagation. The implementation of MIC modeling is based on by superimposing two native Abaqus elements which are connected by a cohesive MIC UEL. In order to facilitate the regularized Heaviside step function based energy conditions for the MIC propagation, a UMAT is used in the superimposed elements. In addition, an interface UEL is developed to connect two composite plies with MICs.

Three verification models including a simple bar, unnotched plate, and an OHT coupon are considered for verifying accurate insertion and propagation of the arbitrary transverse matrix crack in the MIC implementation. The numerical simulations for unnotched plate and OHT coupon show that one or more MICs are correctly inserted parallel to the fiber orientation at the failure locations when the LaRC04 failure criterion [44] is met or exceeded at the end of each loading increment. The propagation of each MIC, governed by a mixed-mode cohesive

model implemented in the MIC UEL, has good agreement compared with the experiments data for the splitting phenomenon of the OHT 0° coupon.

Benchmark solutions were used to successfully verify interface UEL accuracy for Mode I, Mode II propagation and the interaction between initial matrix crack and delamination in geometrically linear and non-linear formulation for DCB, ENF and TCT specimens. Finally, a bending example of a 2 plies composite laminate was considered bringing together the developed capabilities in geometrically nonlinear formulation to simulate matrix cracking and delamination under sliding roller contact displaying the ability of the present implementation to utilize features of the host software without modification of the Rx-FEM implementation.

APPENDIX A

ABAQUS INPUT FILE OF THE PROPOSED IMPLEMENTATION

The present appendix is the Abaqus input file of 9 elements bar example for the proposed implementation. Further detail of this example is provided for reader in section 5-1 Uniaxial Tension of a 9 Elements Bar.

```
**MODEL DEFINITION
*node
**nodes 1 to 20 for original elements, nodes 21 to 40 for duplicate elements
1, 0, 0
2, 0.1, 0
3, 0.2, 0
4, 0.3, 0
5, 0.4, 0
6, 0.5, 0
7, 0.6, 0
8, 0.7, 0
9, 0.8, 0
10, 0.9, 0
11, 0.9, 0.1
12, 0.8, 0.1
13, 0.7, 0.1
14, 0.6, 0.1
15, 0.5, 0.1
16, 0.4, 0.1
17, 0.3, 0.1
```

18, 0.2, 0.1

19, 0.1, 0.1

20, 0, 0.1

21, 0, 0

22, 0.1, 0

23, 0.2, 0

24, 0.3, 0

25, 0.4, 0

26, 0.5, 0

27, 0.6, 0

28, 0.7, 0

29, 0.8, 0

30, 0.9, 0

31, 0.9, 0.1

32, 0.8, 0.1

33, 0.7, 0.1

34, 0.6, 0.1

35, 0.5, 0.1

36, 0.4, 0.1

37, 0.3, 0.1

38, 0.2, 0.1

39, 0.1, 0.1

40, 0, 0.1

*element, type=CPS4, elset=original

1, 1, 2, 19, 20

2, 2, 3, 18, 19

3, 3, 4, 17, 18

4, 4, 5, 16, 17

5, 5, 6, 15, 16

6, 6, 7, 14, 15

7, 7, 8, 13, 14

8, 8, 9, 12, 13

9, 9, 10, 11, 12

*element, type=CPS4, elset=duplicate

10, 21, 22, 39, 40

11, 22, 23, 38, 39

12, 23, 24, 37, 38

13, 24, 25, 36, 37

14, 25, 26, 35, 36

15, 26, 27, 34, 35

16, 27, 28, 33, 34

17, 28, 29, 32, 33

18, 29, 30, 31, 32

*orientation, name=ori-1

1., 0., 0., 0., 1., 0.

3, 90.

*solid section, elset=original, orientation=ori-1, material=umat1

1

* solid section, elset=duplicate, orientation=ori-1, material=umat2

1

*material, name=umat1

*user material, constants=4, type=mechanical

9000, 9000, 3080, 0.3

*depvar

4

* material, name=umat2

*user material, constants=4, type=mechanical

9000, 9000, 3080, 0.3

*depvar

4

*user element, type=U111, nodes=8, coordinates=2, properties=5, variables=42

1, 2

*element, type=U111, elset=mic_uel

19, 1, 2, 19, 20, 21, 22, 39, 40

20, 2, 3, 18, 19, 22, 23, 38, 39

21, 3, 4, 17, 18, 23, 24, 37, 38

22, 4, 5, 16, 17, 24, 25, 36, 37

23, 5, 6, 15, 16, 25, 26, 35, 36

24, 6, 7, 14, 15, 26, 27, 34, 35

25, 7, 8, 13, 14, 27, 28, 33, 34

26, 8, 9, 12, 13, 28, 29, 32, 33

27, 9, 10, 11, 12, 29, 30, 31, 32

*uel property, elset=mic_uel

**properties for cohesive model

0.2, 0.2, 60, 60, 1000000

*step, nlgeom=no, inc=1000

step 1

*static, stabilize, factor=0.002, allsdtol=0.05, continue=no

0.001, 1.0, 1e-20, 0.01

*controls, reset

*controls, parameters=time incrementation

8, 10, , , , , , 50, , ,

*Controls, parameters=field, field=displacement

0.025, 0.05, , , 0.1, , ,

*boundary

1, 2, 2, 0

21, 2, 2, 0

10, 2, 2, 0

30, 2, 2, 0

1, 1, 1, -0.1

21, 1, 1, -0.1

20, 1, 1, -0.1

40, 1, 1, -0.1

10, 1, 1, 0.1

30, 1, 1, 0.1

11, 1, 1, 0.1

31, 1, 1, 0.1

*node print

U

RF

*el print

S

*el print

SDV

*output, field

*node output, variable=preselect

U, RF

*element output, directions=yes, variable=preselect

S, E, SDV

*end step

APPENDIX B

ABAQUS USER-DEFINED MATERIAL CODE OF THE PROPOSED

IMPLEMENTATION

The present appendix is the Abaqus user-defined material (UMAT) code of the proposed implementation. This UMAT code is employed to multiply the material elasticity matrix by regularized Heaviside step function $\tilde{H}(x)$ and $\left(1 - \tilde{H}(x)\right)$ which correspond to the first and second terms in Eq. (40), respectively. Further detail of the UMAT is provided for reader in section 4-1 Implementation of the Mesh-Independent Crack.

```
!abaqus 2d UMAT user subroutine definition
!multiply material stiffness by regularized H
subroutine H_Mat (PROPS,DDSDDE,H)
        include 'ABA_PARAM.INC'
        dimension PROPS(*), DDSDDE(3,3)

        !!isotropic material
        ! real(8) :: E, nu, G, H

        ! E = PROPS(1)
        ! nu = PROPS(2)
        ! G = E/(2.0D0*(1.0D0+nu))

        ! DDSDDE(1,1) = H*E/(1.0D0-nu**2)
        ! DDSDDE(1,2) = H*E/(1.0D0-nu**2)*nu
```

```fortran
! DDSDDE(2,1) = H*E/(1.0D0-nu**2)*nu

! DDSDDE(2,2) = H*E/(1.0D0-nu**2)

! DDSDDE(3,3) = H*G


!Composite material
real(8) :: E1, E2, G12, nu12, H


E1 = PROPS(1)
E2 = PROPS(2)
G12 = PROPS(3)
nu12 = PROPS(4)
nu21 = (E2*nu12)/E1


DDSDDE(1,1) = H*E1/(1.0D0-nu12*nu21)
DDSDDE(1,2) = H*(E1*nu21)/(1.0D0-nu12*nu21)
DDSDDE(2,1) = DDSDDE(1,2)
DDSDDE(2,2) = H*E2/(1.0D0-nu12*nu21)
DDSDDE(3,3) = H*G12


return
end subroutine
```

APPENDIX C

ABAQUS USER-DEFINED ELEMENT CODE OF THE PROPOSED

IMPLEMENTATION

The present appendix is the Abaqus user-defined element (UEL) code of the proposed implementation. The MIC UEL implemented to account for the cohesive forces corresponding to energy M_n in Eq. (40) contains no independent or additional degrees of freedom and only ties together the physical degrees of freedom of the kinematic elements involved akin to a typical interface UEL often build for interface failure modeling in standard element analysis. Further detail of the MIC UEL and interface UEL is provided for reader in section 4-1 Implementation of the Mesh-Independent Crack and 4-2 Implementation of Delamination, respectively.

```
!abaqus 2D UEL user subroutine definition
subroutine uel(RHS, AMATRX, SVARS, ENERGY, NDOFEL, NRHS, NSVARS, &
        PROPS, NPROPS, COORDS, MCRD, NNODE, U, DU, V, A, JTYPE, TIME, &
        DTIME, KSTEP, KINC, JELEM, PARAMS, NDLOAD, JDLTYP, ADLMAG, &
        PREDEF, NPREDF, LFLAGS, MLVARX, DDLMAG, MDLOAD, PNEWDT, JPROPS, &
        NJPRO, PERIOD)

    use rxfem_core
    include 'ABA_PARAM.INC'

    dimension RHS(MLVARX, *), AMATRX(NDOFEL, NDOFEL), PROPS(*), &
```

```fortran
      SVARS(*), ENERGY(8), COORDS(MCRD, NNODE), U(NDOFEL), &
      DU(MLVARX, *), V(NDOFEL), A(NDOFEL), TIME(2), PARAMS(*), &
      JDLTYP(MDLOAD, *), ADLMAG(MDLOAD, *), DDLMAG(MDLOAD, *), &
      PREDEF(2, NPREDF, NNODE), LFLAGS(*), JPROPS(*)


!variables used in the UEL subroutine
dimension Sc(ndofel, ndofel), Fc(ndofel, nrhs), &
      T(mcrd, nrhs), T_d(mcrd, mcrd), R(mcrd, mcrd), &
      Bc(mcrd, ndofel), Bct(ndofel, mcrd), ShapeN(nnode), Co_de(mcrd, nnode), &
      del(mcrd), GP_W1(4), GP_W2(2), tmp(ndofel, mcrd), Co_de_l(mcrd, nnode), &
      Transformation_M(ndofel, ndofel), Transformation_M_T(ndofel, ndofel), &
      tmp1(ndofel, ndofel), Sc_global(ndofel, ndofel), Fc_global(ndofel, nrhs), H(4)


!variables for CZM model
double precision Gn, Gt, Tn_m, Tt_m, th, lambda_ncr, lambda_tcr, &
      T_t, T_n, Jacobian, delta_h, N1, N2, N3, N4, del1, del2, del3, &
      del4, del5, del6, del7, del8, deln_max, delt_max, el_length, damage


integer :: i, n_GP


type(element), pointer :: elem  !the element from elems_a which this MIC UEL shares a facet with
      real(8), allocatable :: crack_norm(:)        !the crack_normal vector for the element
      real(8), allocatable :: elem_mcs(:,:)        !the material coordinate system for the element
      real(8), allocatable :: crack_cs(:,:)        !the crack surface coordinate system
      real(8) :: vec3(3)           !a dummy vector
      real(8) :: mag           !dummy for magnitude of vec3


!MIC UEL for connection elem_a and elem_b
if (JTYPE .eq. 111) then

      !initializations
      n_GP = 4
      GP_W1 = (/1.d0 , 1.d0 , 1.d0 , 1.d0/)
```

80

```fortran
call k_Matrix_Zero(RHS, ndofel, nrhs)

call k_Matrix_Zero(AMATRX, NDOFEL, NDOFEL)

Transformation_M = 0.d0

Transformation_M_T = 0.d0

Co_de = 0.d0


!properties for CZM model

Gn              = PROPS(1)

Gt              = PROPS(2)

Tn_m            = PROPS(3)

Tt_m            = PROPS(4)

K               = PROPS(5)


!Change from the global coordinates to the local coordinates

do i = 1, mcrd

        do j = 1, nnode

                Co_de(i, j) = COORDS(i, j) + U(2*(j-1)+i)

        end do

end do


do i =1, n_GP

        !first get the gradient of the step function at this integration point

        call get_elem(JELEM-4*size(groups(1)%elems_a), elem)


        !now obtain the norm of the element's step function gradient

        delta_h = elem%norm_grad_stepf_int(i)


        !get rid of small values in delta_h

        if (abs(delta_h) .lt. 1e-10) then

                delta_h = 0.d0

        end if


        if ((allocated(elem%crack_norm)) .and. (delta_h .ne. -1.d0)) then
```

```fortran
!now obtain the set of basis vectors aligned with the crack surface
if (allocated(crack_norm) .ne. .true.) then
        allocate(crack_norm(size(elem%crack_norm)))
        allocate(elem_mcs(size(elem%orient,1),size(elem%orient,2)))
        allocate(crack_cs(size(elem_mcs,1),size(elem_mcs,2)))
end if


!pull the element attributes into dummy parameters
crack_norm(:) = elem%crack_norm(:)
elem_mcs(:,:) = elem%orient(:,:)


!fill the first axis of the crack c.s. with the crack normal
crack_cs(1,1:) = crack_norm(1:size(crack_cs,2))
!fill the second axis of the crack c.s. with the fiber direction
crack_cs(2,1:) = elem_mcs(1:,1)


!transpose the crack_cs so that its columns are the basis vectors of the
crack_cs
crack_cs = transpose(crack_cs)
R = transpose(crack_cs)
else
        R(1,:) = (/1.d0, 0.d0/)
        R(2,:) = (/0.d0, 1.d0/)
end if


!now get the tangential and normal separations at this integration point
call k_Coords_Transform (PROPS, R, el_length, COORDS, U, ndofel, nnode, delta_h,
& mcrd, Jacobian, Co_de_l, Transformation_M, Transformation_M_T, Co_de, c_r,
c_s, i)


del1 = Co_de_l(1, 5) - Co_de_l(1, 1)
del2 = Co_de_l(2, 5) - Co_de_l(2, 1)
del3 = Co_de_l(1, 6) - Co_de_l(1, 2)
del4 = Co_de_l(2, 6) - Co_de_l(2, 2)
```

del5 = Co_de_l(1, 7) - Co_de_l(1, 3)

del6 = Co_de_l(2, 7) - Co_de_l(2, 3)

del7 = Co_de_l(1, 8) - Co_de_l(1, 4)

del8 = Co_de_l(2, 8) - Co_de_l(2, 4)

!Numerical integration to compute RHS and AMATRX

N1 = (1 - c_r)*(1 - c_s)*0.25

N2 = (1 + c_r)*(1 - c_s)*0.25

N3 = (1 + c_r)*(1 + c_s)*0.25

N4 = (1 - c_r)*(1 + c_s)*0.25

del(1) = N1*del1 + N2*del3 + N3*del5 + N4*del7

del(2) = N1*del2 + N2*del4 + N3*del6 + N4*del8

!for cohesive law

damage = SVARS(i)

call cohesive_law2 (PROPS, T, T_d, del, SVARS, damage, delta_h)

!store CZM damage into elem_a

elem%damage(i) = damage

ShapeN(1) = -N1

ShapeN(2) = -N2

ShapeN(3) = -N3

ShapeN(4) = -N4

ShapeN(5) = N1

ShapeN(6) = N2

ShapeN(7) = N3

ShapeN(8) = N4

Bc(1, 1) = ShapeN(1)

Bc(1, 2) = 0.d0

Bc(1, 3) = ShapeN(2)

Bc(1, 4) = 0.d0

Bc(1, 5) = ShapeN(3)

Bc(1, 6) = 0.d0

Bc(1, 7) = ShapeN(4)

Bc(1, 8) = 0.d0

Bc(1, 9) = ShapeN(5)

Bc(1, 10) = 0.d0

Bc(1, 11) = ShapeN(6)

Bc(1, 12) = 0.d0

Bc(1, 13) = ShapeN(7)

Bc(1, 14) = 0.d0

Bc(1, 15) = ShapeN(8)

Bc(1, 16) = 0.d0

Bc(2, 1) = 0.d0

Bc(2, 2) = ShapeN(1)

Bc(2, 3) = 0.d0

Bc(2, 4) = ShapeN(2)

Bc(2, 5) = 0.d0

Bc(2, 6) = ShapeN(3)

Bc(2, 7) = 0.d0

Bc(2, 8) = ShapeN(4)

Bc(2, 9) = 0.d0

Bc(2, 10) = ShapeN(5)

Bc(2, 11) = 0.d0

Bc(2, 12) = ShapeN(6)

Bc(2, 13) = 0.d0

Bc(2, 14) = ShapeN(7)

Bc(2, 15) = 0.d0

Bc(2, 16) = ShapeN(8)

call k_Matrix_Transpose (Bc, Bct, mcrd, ndofel)

call k_Matrix_Multiply (Bct, T_d, tmp, ndofel, mcrd, mcrd)

call k_Matrix_Multiply (tmp, Bc, Sc, ndofel, mcrd, ndofel)

call k_Matrix_Multiply (Transformation_M_T, Sc, tmp1, ndofel, ndofel, ndofel)

```fortran
                call k_Matrix_Multiply (tmp1, Transformation_M, Sc_global, ndofel, ndofel, ndofel)

                call k_Matrix_PlusScalar (AMATRX, Sc_global, Jacobian, ndofel, ndofel)

                call k_Matrix_Multiply (Bct, T, Fc, ndofel, mcrd, nrhs)

                call k_Matrix_Multiply (Transformation_M_T, Fc, Fc_global, ndofel, ndofel, nrhs)

                call k_Matrix_PlusScalar (RHS, -Fc_global, Jacobian, ndofel, nrhs)


                !update the SVARS for cohesive law
                SVARS(i) = damage


        end do


!interface UEL between plies
else if (JTYPE .eq. 112) then


        !initializations
        n_GP = 2
        GP_W2 = (/1.d0, 1.d0/)
        call k_Matrix_Zero(RHS, ndofel, nrhs)
        call k_Matrix_Zero(AMATRX, NDOFEL, NDOFEL)
        Transformation_M = 0.d0
        Transformation_M_T = 0.d0
        Co_de = 0.d0
        H(4) = 0.d0


        !properties for CZM model
        Gn              = PROPS(1)
        Gt              = PROPS(2)
        Tn_m            = PROPS(3)
        Tt_m            = PROPS(4)
        K               = PROPS(5)


        do i = 1, mcrd
                do j = 1, nnode
```

```fortran
            Co_de(i, j) = COORDS(i, j) + U(2*(j-1)+i)

        end do

end do


do i =1, n_GP


        !get element information
        call get_elem(JELEM-5*size(groups(1)%elems_a), elem)


        H(1) = elem%nodes(4)%node%stepf

        H(2) = elem%nodes(3)%node%stepf

        H(3) = elem%nodes(2)%node%stepf

        H(4) = elem%nodes(1)%node%stepf


        !now get the tangential and normal separations at this integration point
        call k_Coords_Transform_interface (PROPS, R, el_length, COORDS, U, ndofel,
        nnode, delta_h, & mcrd, Jacobian, Co_de_l, Transformation_M,
        Transformation_M_T, Co_de, c_r, c_s, i)


        del1 = H(4)*Co_de_l(1, 4) + (1-H(4))*Co_de_l(1, 8) - H(1)*Co_de_l(1, 1) - (1-
        H(1))*Co_de_l(1, 5)

        del2 = H(4)*Co_de_l(2, 4) + (1-H(4))*Co_de_l(2, 8) - H(1)*Co_de_l(2, 1) - (1-
        H(1))*Co_de_l(2, 5)

        del3 = H(3)*Co_de_l(1, 3) + (1-H(3))*Co_de_l(1, 7) - H(2)*Co_de_l(1, 2) - (1-
        H(2))*Co_de_l(1, 6)

        del4 = H(3)*Co_de_l(2, 3) + (1-H(3))*Co_de_l(2, 7) - H(2)*Co_de_l(2, 2) - (1-
        H(2))*Co_de_l(2, 6)


        !Numerical integration to compute RHS and AMATRX
        N1 = (1 - c_s)*0.5

        N2 = (1 + c_s)*0.5

        del(1) = N1*del1 + N2*del3

        del(2) = N1*del2 + N2*del4


        !for cohesive_law
        damage = SVARS(i)
```

call cohesive_law_interface (PROPS, T, T_d, del, SVARS, damage)

!Bc matrix

Bc(1, 1) = -H(1)*N1

Bc(1, 2) = 0.d0

Bc(1, 3) = -H(2)*N2

Bc(1, 4) = 0.d0

Bc(1, 5) = H(3)*N2

Bc(1, 6) = 0.d0

Bc(1, 7) = H(4)*N1

Bc(1, 8) = 0.d0

Bc(1, 9) = (-1+H(1))*N1

Bc(1, 10) = 0.d0

Bc(1, 11) = (-1+H(2))*N2

Bc(1, 12) = 0.d0

Bc(1, 13) = (1-H(3))*N2

Bc(1, 14) = 0.d0

Bc(1, 15) = (1-H(4))*N1

Bc(1, 16) = 0.d0

Bc(2, 1) = 0.d0

Bc(2, 2) = -H(1)*N1

Bc(2, 3) = 0.d0

Bc(2, 4) = -H(2)*N2

Bc(2, 5) = 0.d0

Bc(2, 6) = H(3)*N2

Bc(2, 7) = 0.d0

Bc(2, 8) = H(4)*N1

Bc(2, 9) = 0.d0

Bc(2, 10) = (-1+H(1))*N1

Bc(2, 11) = 0.d0

Bc(2, 12) = (-1+H(2))*N2

Bc(2, 13) = 0.d0

```fortran
        Bc(2, 14) = (1-H(3))*N2
        Bc(2, 15) = 0.d0
        Bc(2, 16) = (1-H(4))*N1


        call k_Matrix_Transpose (Bc, Bct, mcrd, ndofel)
        call k_Matrix_Multiply (Bct, T_d, tmp, ndofel, mcrd, mcrd)
        call k_Matrix_Multiply (tmp, Bc, Sc, ndofel, mcrd, ndofel)
        call k_Matrix_Multiply (Transformation_M_T, Sc, tmp1, ndofel, ndofel, ndofel)
        call k_Matrix_Multiply (tmp1, Transformation_M, Sc_global, ndofel, ndofel, ndofel)
        call k_Matrix_PlusScalar (AMATRX, Sc_global, Jacobian, ndofel, ndofel)
        call k_Matrix_Multiply (Bct, T, Fc, ndofel, mcrd, nrhs)
        call k_Matrix_Multiply (Transformation_M_T, Fc, Fc_global, ndofel, ndofel, nrhs)
        call k_Matrix_PlusScalar (RHS, -Fc_global, Jacobian, ndofel, nrhs)

        !update the SVARS for cohesive_law2
        SVARS(i) = damage


      end do


    end if


    return
end subroutine
```

APPENDIX D

ABAQUS USER-DEFINED EXTERNAL DATABASES CODE OF THE PROPOSED

IMPLEMENTATION

The present appendix is the Abaqus user-defined external databases (UEXTERNALDB) code of the proposed implementation. The UEXTERNALDB is used to manage user-defined external databases and calculate model-independent history information. Further detail of the UEXTERNALDB is provided in [35].

```fortran
!abaqus UEXTERNALDB user subroutine definition
subroutine uexternaldb(lop, lrestart, time, dtime, kstep, kinc)

        use rxfem_core
        include 'aba_param.inc'

        !inputs
        integer :: lop, lrestart, kstep, kinc
        real(8) :: time(:), dtime

        !working parameters
        character(300) :: path, file
        integer :: len_path, len_file, i
        type(string) :: a

        !if beginning of analysis, read in the input file and populate rxfem model constructs
        if (lop .eq. 0) then
```

```fortran
        !get the working directory
        call getoutdir(path, len_path)
        a = trim(path)


        !get the input file name
        call getjobname(file, len_file)
        a = a + '\' + trim(file) + '.inp'


        !call delete entity
        call deleteentity()


        !setup the rxfem model containers
        call process_input_file(a%chars)


end if


!print the increment number in the beginning of every increment
if (lop .eq. 1) then
        print *, 'working on load increment: ', kinc
end if


!check failure criterion to insert crack
if (lop .eq. 2) then
        !calculate the failure index of every integration point
        call LaRC04()


        !check the failure index to insert crack
        call check2insertcrack()


end if


!if end of analysis
if (lop .eq. 3) then
```

```fortran
            !call delete entity
            call deleteentity()
      end if


      return


end subroutine uexternaldb
```

REFERENCES

[1]   E. V. Iarve, M. R. Gurvich, D. H. Mollenhauer, C. A. Rose, and C. G. Dávila, "Mesh-independent matrix cracking and delamination modeling in laminated composites," *Int. J. Numer. Methods Eng.*, 2011.

[2]   R. Talreja, "CONTINUUM MECHANICS CHARACTERIZATION OF DAMAGE IN COMPOSITE MATERIALS.," *Proc. R. Soc. London, Ser. A Math. Phys. Sci.*, 1985.

[3]   J. W. Lee, D. H. Allen, and C. E. Harris, "Internal State Variable Approach for Predicting Stiffness Reductions in Fibrous Laminated Composites with Matrix Cracks," *J. Compos. Mater.*, 1989.

[4]   B. N. Nguyen, "Three-dimensional modeling of damage in laminated composites containing a central hole," *J. Compos. Mater.*, 1997.

[5]   G. Alfano and M. A. Crisfield, "Finite element interface models for the delamination analysis of laminated composites: Mechanical and computational issues," *Int. J. Numer. Methods Eng.*, 2001.

[6]   W. G. Jiang, S. R. Hallett, B. G. Green, and M. R. Wisnom, "A concise interface constitutive law for analysis of delamination and splitting in composite materials and its application to scaled notched tensile specimens," *Int. J. Numer. Methods Eng.*, 2007.

[7]   A. Turon, P. P. Camanho, J. Costa, and C. G. Dávila, "A damage model for the simulation of delamination in advanced composites under variable-mode loading," *Mech. Mater.*, 2006.

[8]   R. Krueger, "Virtual crack closure technique: History, approach, and applications," *Appl. Mech. Rev.*, 2004.

[9]   L. R. Deobald, G. E. Mabson, B. Dopker, D. M. Hoyt, J. Baylor, and D. Graesser, "Interlaminar fatigue elements for crack growth based on virtual crack closure technique," in *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2007.

[10]   T. E. Tay, G. Liu, V. B. C. Tan, X. S. Sun, and D. C. Pham, "Progressive failure analysis of composites," *J. Compos. Mater.*, 2008.

[11]   P. Maimí, P. P. Camanho, J. A. Mayugo, and C. G. Dávila, "A continuum damage model for composite laminates: Part I - Constitutive model," *Mech. Mater.*, 2007.

[12]   P. Maimí, P. P. Camanho, J. A. Mayugo, and C. G. Dávila, "A continuum damage model for composite laminates: Part II - Computational implementation and validation," *Mech. Mater.*, 2007.

[13]   Y. Nikishkov, A. Markeev, and G. Seon, "Simulation of damage in composites based on solid finite elements," *J. Am. Helicopter Soc.*, 2010.

[14]   K. Song, Y. Li, and C. A. Rose, "Continuum damage mechanics models for the analysis of progressive failure in open-hole tension laminates," in *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and*

Materials Conference, 2011.

[15]   C. S. Lopes, S. Sádaba, C. González, J. Llorca, and P. P. Camanho, "Physically-sound simulation of low-velocity impact on fiber reinforced laminates," in *International Journal of Impact Engineering*, 2016.

[16]   E. V. Iarve, D. Mollenhauer, and R. Kim, "Theoretical and experimental investigation of stress redistribution in open hole composite laminates due to damage accumulation," in *Composites Part A: Applied Science and Manufacturing*, 2005.

[17]   D. Mollenhauer, E. V. Iarve, R. Kim, and B. Langley, "Examination of ply cracking in composite laminates with open holes: A moiré interferometric and numerical study," *Compos. Part A Appl. Sci. Manuf.*, 2006.

[18]   S. R. Hallett, W. G. Jiang, B. Khan, and M. R. Wisnom, "Modelling the interaction between matrix cracks and delamination damage in scaled quasi-isotropic specimens," *Compos. Sci. Technol.*, 2008.

[19]   F. P. Van Der Meer and L. J. Sluys, "Continuum models for the analysis of progressive failure in composite laminates," *J. Compos. Mater.*, 2009.

[20]   O. Allix, D. Guédra-Degeorges, S. Guinard, and A. Vinet, "Analyse de la tenue aux impacts à faible vitesse et faible énergie des stratifiés composites par la mécanique de l'endommagement," *Mec. Ind.*, 2000.

[21]   N. Moës, J. Dolbow, and T. Belytschko, "A finite element method for crack growth without remeshing," *Int. J. Numer. Methods Eng.*, 1999.

[22]   M. J. Swindeman, E. V. Iarve, R. A. Brockman, D. H. Mollenhauer, and S. R. Hallett,

"Strength prediction in open hole composite laminates by using discrete damage modeling," *AIAA J.*, 2013.

[23]   E. V. Iarve, "Mesh independent modelling of cracks by using higher order shape functions," *Int. J. Numer. Methods Eng.*, 2003.

[24]   B. Patzák and M. Jirásek, "Process zone resolution by extended finite elements," *Eng. Fract. Mech.*, 2003.

[25]   E. Benvenuti, A. Tralli, and G. Ventura, "A regularized XFEM model for the transition from continuous to discontinuous displacements," *Int. J. Numer. Methods Eng.*, 2008.

[26]   E. Benvenuti, "A regularized XFEM framework for embedded cohesive interfaces," *Comput. Methods Appl. Mech. Eng.*, 2008.

[27]   H. K. Adluru, K. H. Hoos, E. V. Iarve, and J. G. Ratcliffe, "Delamination initiation and migration modeling in clamped tapered laminated beam specimens under static loading," *Compos. Part A Appl. Sci. Manuf.*, 2019.

[28]   E. V. Iarve, K. H. Hoos, Y. Nikishkov, and A. Makeev, "Discrete damage modeling of static bearing failure in laminated composites," *Compos. Part A Appl. Sci. Manuf.*, 2018.

[29]   K. Hoos, E. V. Iarve, M. Braginsky, E. Zhou, and D. H. Mollenhauer, "Static strength prediction in laminated composites by using discrete damage modeling," *J. Compos. Mater.*, 2017.

[30]   E. V. Iarve, K. Hoos, M. Braginsky, E. Zhou, and D. H. Mollenhauer, "Progressive

failure simulation in laminated composites under fatigue loading by using discrete damage modeling," *J. Compos. Mater.*, 2017.

[31]   H. K. Adluru, K. H. Hoos, and E. V. Iarve, "Discrete damage modelling of delamination migration in clamped tapered laminated beam specimens," in *32nd Technical Conference of the American Society for Composites 2017*, 2017.

[32]   F. P. van der Meer and L. J. Sluys, "A phantom node formulation with mixed mode cohesive law for splitting in laminates," in *International Journal of Fracture*, 2009.

[33]   F. P. van der Meer, "Mesolevel Modeling of Failure in Composite Laminates: Constitutive, Kinematic and Algorithmic Aspects," *Arch. Comput. Methods Eng.*, 2012.

[34]   A. B. Harman, D. Rapking, D. Mollenhauer, and E. Iarve, "Examination of a small radius of curvature composite notch with a novel chevron feature to improve damage tolerance," *Compos. Part A Appl. Sci. Manuf.*, 2019.

[35]   D. Simulia, "Abaqus 6.17 Documentation.," *Provid. RI, USA DS SIMULIA Corp*, 2017.

[36]   E. Giner, N. Sukumar, J. E. Tarancón, and F. J. Fuenmayor, "An Abaqus implementation of the extended finite element method," *Eng. Fract. Mech.*, 2009.

[37]   B. Y. Chen, S. T. Pinho, N. V. De Carvalho, P. M. Baiz, and T. E. Tay, "A floating node method for the modelling of discontinuities in composites," *Eng. Fract. Mech.*, 2014.

[38]   W. Liu, Q. D. Yang, S. Mohammadizadeh, and X. Y. Su, "An efficient augmented

finite element method for arbitrary cracking and crack interaction in solids," *Int. J. Numer. Methods Eng.*, 2014.

[39]   D. Ling, Q. Yang, and B. Cox, "An augmented finite element method for modeling arbitrary discontinuities in composite materials," *Int. J. Fract.*, 2009.

[40]   Y. J. Liang, E. V. Iarve, and J. S. McQuien, "The implementation of regularized extended finite element method (RX-FEM) in ABAQUS," in *AIAA Scitech 2019 Forum*, 2019.

[41]   Y.-J. LIANG, J. S. MCQUIEN, and E. V. IARVE, "An ABAQUS Implementation of Regularized Extended Finite Element Method (Rx-FEM) for Modeling the Interaction Between Matrix Cracking and Delamination in Composites," in *American Society for Composites 2019*, 2019.

[42]   A. Hansbo and P. Hansbo, "A finite element method for the simulation of strong and weak discontinuities in solid mechanics," *Comput. Methods Appl. Mech. Eng.*, 2004.

[43]   I. Babuška and J. M. Melenk, "The partition of unity method," *Int. J. Numer. Methods Eng.*, 1997.

[44]   S. T. Pinho, C. G. Dávila, P. P. Camanho, L. Iannucci, and P. Robinson, "Failure Models and Criteria for FRP Under In-Plane or Three-Dimensional Stress States Including Shear Non-linearity," *Nasa/Tm-2005-213530*, 2005.

[45]   E. V. Iarve and D. H. Mollenhauer, "Mesh-independent matrix cracking and delamination modeling in advanced composite materials," in *Numerical Modelling of Failure in Advanced Composite Materials*, 2015.

[46] X. Lu, M. Ridha, B. Y. Chen, V. B. C. Tan, and T. E. Tay, "On cohesive element parameters and delamination modelling," *Eng. Fract. Mech.*, 2019.

[47] M. König, R. Krüger, K. Kussmaul, M. von Alberti, and M. Gädke, "Characterizing Static and Fatigue Interlaminar Fracture Behavior of a First Generation Graphite/Epoxy Composite," in *Composite Materials: Testing and Design, Thirteenth Volume*, 2009.