5-2018

Simulating composite delamination with a damage-type cohesive zone model

Haodong Du
Purdue University

SIMULATING COMPOSITE DELAMINATION WITH A DAMAGE-TYPE COHESIVE ZONE MODEL

A Thesis

Submitted to the Faculty

of

Purdue University

by

Haodong Du

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Aeronautics and Astronautics

May 2018

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Wenbin Yu, Chair

School of Aeronautics and Astronautics

Dr. Marisol Koslowski

Department of Mechanical Engineering

Dr. Vikas Tomar

School of Aeronautics and Astronautics

Approved by:

Dr. Wayne Chen

Head of the School Graduate Program

ACKNOWLEDGMENTS

The author would like to thank Dr. Wenbin Yu for his support and academic advice. The author would also like to thank Dr. Liang Zhang for his guidance through the project and valuable advice on academic writing. The author is also thankful for Dr. Marisol Koslowski and Dr. Vikas Tomar as committee memebers.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SYMBOLS

a	initial crack length
A	damage accumulation conjugate foce
b	width of specimen
c	lever length
d	damage parameter for measuring degradation
f	damage function
G_{Ic}, G_{IIc}	mode I, mode II critical energy release rate
h	half thickness of specimen
J	path dependence function
$\tilde{K}$	initial elastic stiffness
K	actual elastic stiffness
k	initial damage threshold
$\tilde{K}_c$	penalty stiffness in compression
L	half length of specimen
$\boldsymbol{L}$	consistent Jacobian
P	applied load
Q, b	damage accumulation parameters
u	load point displacement
y	damage conjugate force
α	damage parameter for measuring damage accumulation
$\boldsymbol{\beta}$	mode mix vector
$\tilde{\boldsymbol{\gamma}}$	effective separation vector
$\boldsymbol{\gamma}$	separation vector
γ_{Ic}	mode I critical separation at delamination onset
$\hat{\boldsymbol{\gamma}}$	direction vector of separation

λ	Lagrange multiplier
$\boldsymbol{\tau}$	traction vector
$\tilde{\boldsymbol{\tau}}$	effective traction vector
τ_{Imax}, γ_{Imax}	mode I interfacial strength and corresponding separation
Φ	dissipation per unit area
Ψ	Helmholtz free energy per unit area
Ψ_e, Ψ_d	elastic and damage part of Helmholtz free energy per unit area

ABSTRACT

Du, Haodong MSAA, Purdue University, May 2018. Simulating composite delamination with a damage-type cohesive zone model. Major Professor: Wenbin Yu.

Interlaminar damage (delamination) is one of the predominant forms of failure in laminated composites, which is broadly used in aerospace, astronautical and automobile industry and many other fields. Engineering problems about damage tolerance and structure durability requires the ability to simulate mixed mode delamination in laminated composites.

The objective of the research is to develop an implicit scheme for a recently developed damage-type cohesive zone model (CZM) with an associated systematic calibration method. The CZM is formulated based on thermodynamics, and the damage evolution is derived with the principle of maximum dissipation.

A stable implicit scheme using the Newton–Raphson method is developed to solve the model iteratively. A finite element framework consisting of double-cantilever beam (DCB) , end-notched flexure (ENF) and mixed-mode beam (MMB) models and properly chosen mesh density is built to incorporate the present CZM. A systematic calibration method is then established to calibrate the damage parameters from experimental results of interfacial parameters and flexural tests.

The present model is found to yield consistent and accurate results in finite element simulations. Specifically, it's shown to be able to reproduce the critical energy release rates and maximum loads that the structure can endure. The maximum loads are found to be also affected by the interfacial strenghth. Conclusively, the present model could be used in engineering practice because of its superior accuracy and stability.

1. INTRODUCTION

1.1 Motivation

Laminated composites are broadly used in aerospace and automobile industry and other fields because of its superior properties like large strength-weight ratio and long fatigue life. It's crucial to study the failure mechanism of laminated composites, among which delamination is one of the predominant forms of failure. The existence of delamination could affect mechanical properties such as stiffness and strength of the structures. The traditional way to solve these problems is to use finite element method in conjunction with linear elastic fracture mechanics (LEFM), which requires that the fracture process zone (FPZ) is small compared with any characteristic length. However, in the case of composite delamination the FPZ could be comparable to characteristic length. The cohesive zone model (CZM) is able to handle large FPZ and has been intensively researched for the simulation of composite delamination. In spite that various CZMs have been proposed, there are some challenges that need to be further addressed:

1. The CZM should be able to simulate delamination along complex separation paths reliably and consistently;

2. The critical energy release rates and maximum loads should be reproduced to high accuracy for an arbitrarily mixed mode.

1.2 Literature Review

In experimental studies, delamination is characterized within the framework of fracture mechanics, in which the energy releas rates G_c is a commonly used property. Test methods have been proposed to study G_c in different delamination modes, such

as double cantilever beam (DCB) for mode I, end-notch flexure (ENF) for mode II and mixed-mode bending (MMB) for mixed mode I and mode II. These methods have been standardized in ASTM standards [1–3]. The models are depicted in Figure 1.1. The specimen is a composite laminate with a pre-crack between two layers in the middle. In the DCB model, two vertical forces in opposite directions are applied to the free ends of cantilever beams; in the ENF model, a vertical force is applied at the center of a simply supported beam; in the MMB model, an upward force at the free end and a downward force at the center are applied to the specimen, and the mode mix ratio is adjusted by the force ratio, which is realized using a lever with adjustable length. Once the delamination parameters are determined from experiments, numerical approach is then utilized to study sophisticated delamination behaviors.

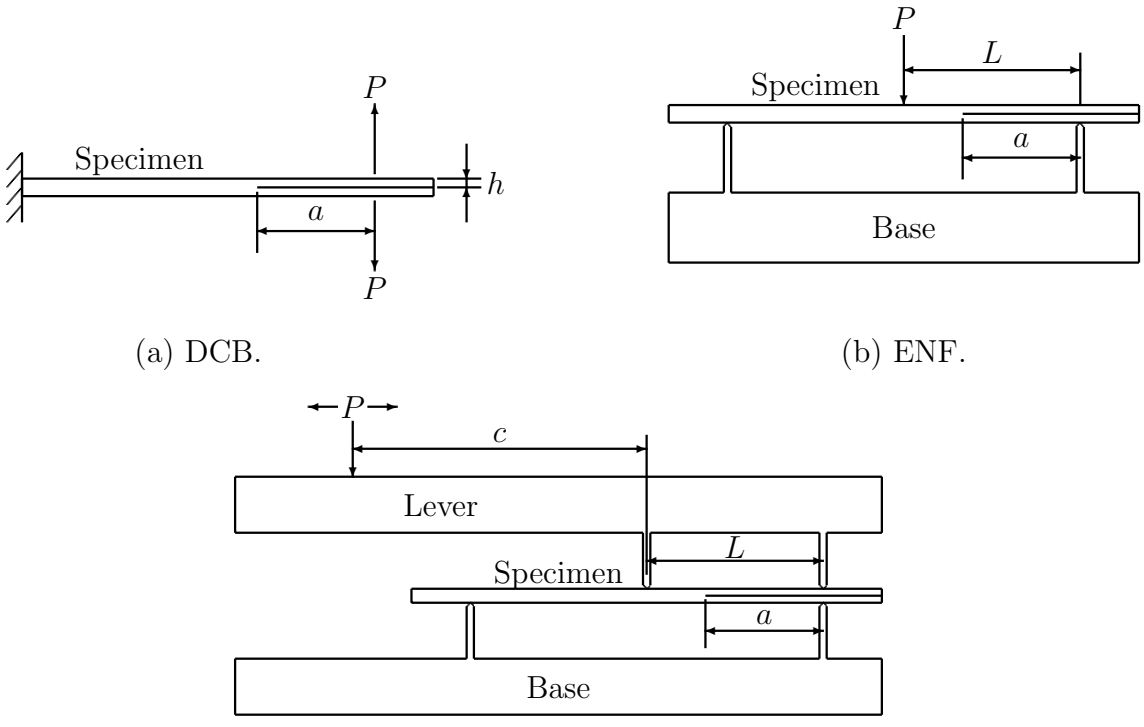
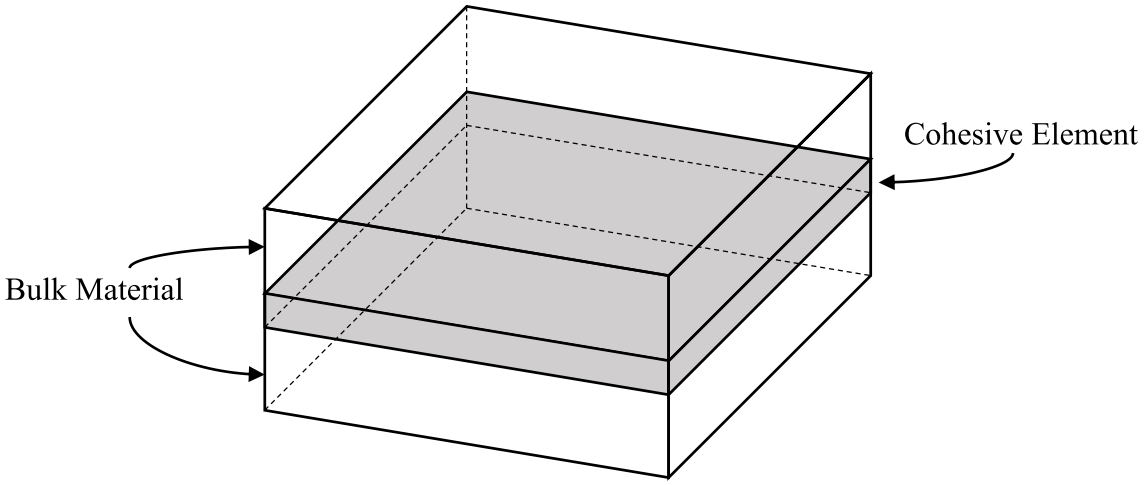


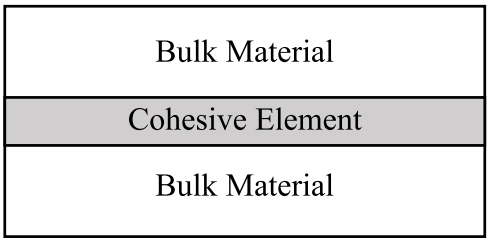
Figure 1.1. Schematics of DCB, ENF and MMB models.

The CZM has been shown to perform well in predicting crack growth in composites [4–6], which consists of a FPZ comparable to the characteristic length. Dugdale [7] and Barenblatt [8] first introduced the concept of CZM by limiting the material nonlinearity in a thin sheet ahead of the crack tip. A cohesive element could be a brick element in 3D or a quad element in 2D. Theoretically, the element has a zero thickness while in Abaqus a very small geometric thickness could be assigned and the nominal strains would be calculated by relative separation displacements [9]. The structures of cohesive elements are illustrated in Figure 1.2. In real models, the cohesive elements usually have a much smaller thickness than the elements shown in Figure 1.2. Thereafter, a cohesive law is constructed to relate the interfacial traction and the intelaminar separation. One of the most-known and simplest traction–separation relation is the bilinear model by Camanho and Dávila [10], which includes a linear increasing portion and a linear softening portion and eventually breaks when the traction is very low. Various advanced CZMs have more complex traction–separation relation but they share the similar trend of increasing traction and subsequent softening.

Current CZMs are categorized into potential-based CZMs and damage-type CZMs. For a potential-based CZM, the cohesive relation is obtained from a potential function, which is properly designed to characterize the traction–separation relationship (see Park and Paulino's paper [5] for a review). CZMs with predetermined traction-separation profiles are also mostly categorized into potential-based CZMs. The related cohesive laws include polynomial [11], exponential [12, 13], bilinear [14], trapezoidal functions [15]. As an example, Xu and Needleman proposed a potential function with exponential functions. This CZM has some advantages: the normal compression control is automatically achieved; the tractions and their derivatives are continuous [16]. However, the irreversibility and path dependence are not accounted for. Since they developed this CZM, numerous attempts have been made to extend it. Ortiz and Pandolfi incorporated irreversibility into exponential cohesive law by introducing an effective separation [13]. Bosch et al. took into account the path dependence based on the CZM [16] by considering potential functions for the tangential direction and



(a) 3D



(b) 2D

Figure 1.2. Schematics of cohesive elements.

the normal direction separately, which leads to more model parameters. These improvements have their own limitations:

1. Ad hoc assumptions other than the cohesive potential functions are needed;

2. The critical energy release rates cannot be recovered precisely for mixed mode delamination.

As a matter of fact, delamination is a damage process. To extend the potential-based CZMs, some concepts of damage is introduced to consider the irreversibility. The thesis would be focused on damage-type CZMs which are directly rooted in principles of thermodynamics. A damage-type CZM generally involves a properly developed damage criteria and the corresponding damage evolution law. Most damage-type CZMs have a damage evolution law constructed phenomenologically and have few restrictions on the damage criteria [17–19]. It turns out the damage evolution rule could be derived from the damage criteria following the principle of maximum dissipation [20]. Specifically, we want a generalized standard damage model [21], which possesses the following features:

1. The relation between each pair of state variable and its conjugated thermodynamic driving force (e.g., the damage factor and the damage conjugate force) is specified by a potential function;

2. The damage initiation is specified by a damage criterion;

3. The material obeys the principle of maximum disspation.

Several attempts have been made to make a damage-type CZM generalized standard. Turon et al. [22] developed a damage-type CZM similar to a generalized standard one except that an effective separation took the place of a damage conjugate force in the damage evolution law. Mosler and Scheider proposed another class of damage-type CZMs by requiring the element following the so-called principle of minimum stress power. Despite improvements, the CZMs are still not generalized standard. For more

detailed discussions about a generalized standard damage model, see Zhang and Gao's paper [21].

Numerical methods for time-dependent nonlinear problems are classified into implicit and explicit methods. The explicit method estimates the state at the next time step from the current state. Many CZM papers utilized explicit methods in solving the CZM equations [23, 24]. Despite its simplicity, the explicit method has intrinsic error and numerical oscilation [20, 25, 26]. Therefore, the implicit method is needed to improve accuracy and stability. The implicit methods involve the state equations at both the current and the next time step. The simultaneous equations would be nonlinear equations and could be solved iteratively with the Newton–Raphson method. Mathematically, the implicit method is more accurate, but it encounters convergence difficulties. The implicit method has been used in models with determined traction separation curves. Camanho used the implicit method for the bilinear model [10]; Gao and Bower used the implicit method for the exponential-shaped traction–separation law and they added viscous dissipation to deal with convergence difficulty [27]. It is straightforward to derive an implicit scheme for models with determined traction separation curves as it is just finding the tangent line on the hyper-surface made up of the traction separation curves. For a model with determined traction separation, the discrepancy between damage-type representation and potential-based representation is not essential. As an example, the bilinear model was present in the damage formulation [10], while it could also be integrated and present with a potential function [5]. However, the present model involves complex separation path and thus the traction separation curves depend on the loading history. To the best knowledge of the author, this is the first implicit model to deal with path dependence behavior. The present model could be solved with the Newton–Raphson method effectively. It yields convergent and consistent results and no artificial viscous dissipation is needed so accuracy and efficiency are balanced.

Calibration is critical to get results in accordance with experimental data. For simple models such as the bilinear model, the critical energy release rates and the

interfacial strength obtained from experiments are directly incorporated in the models [10, 24]. Some criteria are proposed to recover the critical energy release rates in mixed mode delamination. The most popular ones are the power law criterion [28] and the B-K criterion [29]. Some potential-based CZMs includes artificial parameters [30]. While they succeeded in reproducing complex behaviors like path dependence phenomenon, some parameters don't have clear physical meanings and are artificially chosen. Moreover, the critical energy release rate can not be precisely determined with experiments. There have been some concerns about the ASTM standards [1–3]:

1. Linear elastic behavior is assumed. Therefore, the FPZ has to be smaller than any characteristic length, which does not apply as we will show numerically.

2. Modified beam theory is used to establish the relationship between level length and mode mix ratio. The relationship is shown to be not precise with the angle of inclination.

3. ASTM standard is good for engineering purposes since it recommends the most conservative theory for calculating the critical energy release rate. The value might not be suitable for obtaining precise results numerically.

Some of the concerns are also discussed in previous papers [31].

Beyond the ASTM standards, improvements have been made for the analytical solutions [32–34]. By taking into account the nonlinear cohesive law and finite thickness, the models give a reasonable estimate of the FPZ length and can serve as a tool to justify the numerical solutions. However, it is hard to solve CZM with complicated cohesive laws analytically, and path dependence cannot be dealt with conveniently. In other words, the state does not rely on the loading history. In the present model, we try to keep the fundamental formulations straightforward and introduce some parameters with specific physical meanings to incorporate the complexity of the delamination behavior. The critical energy release rate is not directly included in the model, but it is used for calibration. A set of calibration procedures are also established and implemented in programs. The model with the calibrated parameters is

able to reproduce the experimental data to higher accuracy and we believe it to be more generally applicable.

1.3 Objectives

The objective of the research is to develop an implicit scheme for a recently developed damage-type cohesive zone model with an associated systematic calibration method. The formulation and derivation of the CZM together with the principle of maximum dissipation are briefly introduced. Subsequently an implicit scheme of the CZM is proposed based on the Newton–Raphson method, and implemented in a UMAT subroutine for Abaqus/Standard. A finite element framework is constructed to incorporate the CZM. A systematic calibration method is established to calibrate the damage parameters from interfacial parameters and flexural test results. The present CZM with parameters calibrated by the aforementioned systematic calibration method is found to be able to simulate complex separation path and reproduce the maximum loads. The interfacial strength is also found to have an effect on the delamination process.

2. COHESIVE ZONE MODEL

2.1 String-based CZM

In this section, the fundamentals of a string-based CZM together with the damage model will be introduced. Figure 2.1 depicts the crazing ahead of a crack tip during the delamination of a polymer interface. During a crazing process, it's observed that:

1. The interface become crazing once it cannot sustain higher imposed traction;
2. More and more crazes nucleate, grow perpendicular to the adherends, and form into an array of main fibrils, connected by cross-tie fibrils;
3. Once main fibrils fail, delamination occurs.

Accordingly, the cohesive zone can be idealized as a fibrillated region where each cohesive element represent a fibril (or a fibril bundle). Each fibril can further be idealized as a deformable string such that:

1. It can only sustain uniaxial stress in its longitudinal direction [35];
2. Its damage behavior is path-dependent due to some micromechanisms (e.g., the existence of cross-tie fibrils).

According to the first point, each cohesive element should have initial elasticity tensor

$$\tilde{\boldsymbol{K}} = \tilde{K}\boldsymbol{I}, \tag{2.1}$$

where $\tilde{K}$ denotes its initial elastic stiffness, and $\boldsymbol{I}$ denotes the second-order identity tensor.

Figure 2.2 depicts a cohesive element and its associated local corotational material coordinate system, $\boldsymbol{x} = (x_1, x_2, x_3)$. The x_1-, the x_2-, and the x_3-directions are chosen

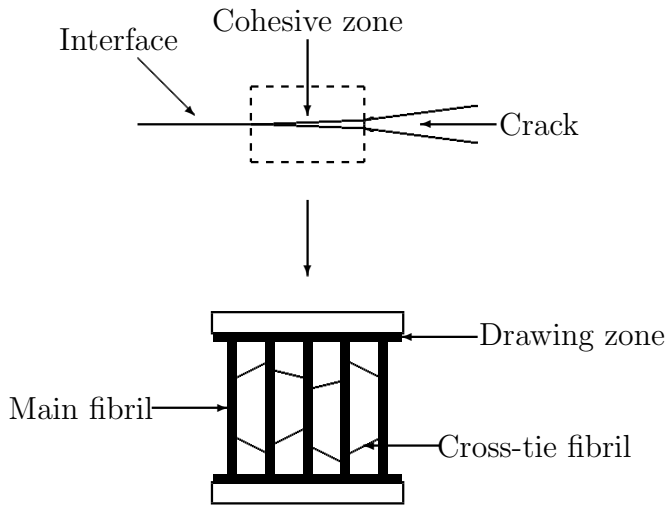
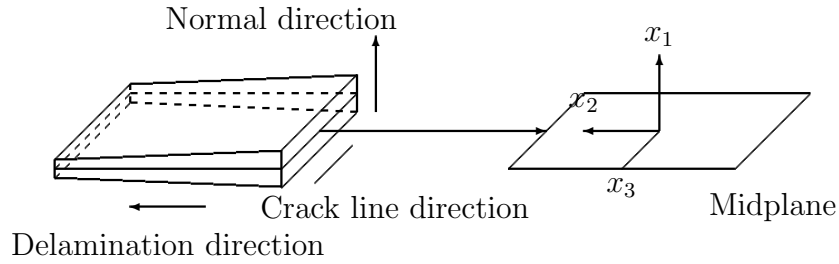


Figure 2.1. Crazing ahead of a crack tip (amended from [36]).



(a) Deformed cohesive element. (b) Local coordinate system.

Figure 2.2. Schematic of a deformed cohesive element.

such that they are associated with mode I, mode II, and mode III fracture, respectively (see Appendix A for more details).

According to the thermodynamics of continuum, the Helmholtz free energy per unit area of the element, say Ψ, can treated as a function of a suitable set of independent state variables, e.g.,

$$\Psi = \Psi\left(\boldsymbol{\gamma}, d, \alpha\right), \tag{2.2}$$

where $\boldsymbol{\gamma}$ denotes the separation vector, d is a scalar measuring the degradation of the element, namely the damage factor, and α is a scalar describing damage accumulation. Hereafter set

$$\boldsymbol{\gamma} = \begin{bmatrix} \langle\gamma_1\rangle & \gamma_2 & \gamma_3 \end{bmatrix}^T \tag{2.3}$$

unless otherwise specified, where $\langle\cdot\rangle$ denotes the Macaulay bracket, i.e.,

$$\langle x\rangle = \begin{cases} 0 & x < 0, \\ x & x \geq 0. \end{cases} \tag{2.4}$$

This implies that a negative γ_1 does not promote damage or delamination. Assume that Ψ can be decomposed into its elastic and damage accumulation parts as

$$\Psi\left(\boldsymbol{\gamma}, d, \alpha\right) = \Psi_e\left(\boldsymbol{\gamma}, d\right) + \Psi_d\left(\alpha\right). \tag{2.5}$$

By definition, the thermodynamic forces conjugate to $\boldsymbol{\gamma}$, d, and α are given by

$$\boldsymbol{\tau} = \frac{\partial\Psi}{\partial\boldsymbol{\gamma}} = \frac{\partial\Psi_e}{\partial\boldsymbol{\gamma}}, \quad y = -\frac{\partial\Psi}{\partial d} = -\frac{\partial\Psi_e}{\partial d}, \quad A = \frac{\partial\Psi}{\partial\alpha} = \frac{\mathrm{d}\Psi_d}{\mathrm{d}\alpha}, \tag{2.6}$$

where $\boldsymbol{\tau}$ denotes the traction vector, y denotes the damage conjugate force, and A denotes the damage accumulation conjugate force. For an isothermal process, the Clausius–Duhem inequality can be expressed as

$$\Phi = \boldsymbol{\tau}\cdot\dot{\boldsymbol{\gamma}} - \dot{\Psi} \geq 0, \tag{2.7}$$

where Φ denotes the dissipation per unit area, and the overdot denotes the time derivative of a quantity. Combining Eqs. (2.5)–(2.7) gives

$$\Phi = y\dot{d} - A\dot{\alpha} \geq 0. \tag{2.8}$$

Following Simo [37], assume that each element obeys the following principle of maximum dissipation: for fixed but otherwise arbitrary $\dot{d}$ and $\dot{\alpha}$, the actual y and A should maximize Φ subject to constraint $f \leq 0$, or mathematically speaking, should make Lagrange functional

$$L\left(y, A, \dot{\lambda}; \dot{d}, \dot{\alpha}\right) = y\dot{d} - A\dot{\alpha} - \dot{\lambda}f \tag{2.9}$$

attain its extremum, where $\dot{\lambda}$ is a positive Lagrange multiplier, and $f = f(y, A)$ is referred to as the damage function. In other words, one could obtain associated damage evolution laws

$$\dot{d} = \dot{\lambda}\frac{\partial f}{\partial y} \quad \text{and} \quad \dot{\alpha} = -\dot{\lambda}\frac{\partial f}{\partial A} \tag{2.10}$$

subject to loading/unloading conditions

$$f \leq 0, \quad \dot{\lambda} \geq 0, \quad \dot{\lambda}f = 0. \tag{2.11}$$

The concepts of effective space and energy equivalence can facilitate the derivation. The effective space is where a fictitious, undamaged cohesive element is obtained from a real, damaged one by removing all damages so that a set of so-called effective quantities can be defined [38]. Following Cordebois and Sidoroff [39], let the energy equivalence hypothesis follow the definition of the effective traction vector: the effective traction vector should be applied to an undamaged element such that it produces the same elastic Helmholtz free energy as those observed on a damaged element subject to the apparent traction vector, i.e.,

$$\Psi_e = \frac{1}{2}K\boldsymbol{\gamma} \cdot \boldsymbol{\gamma} = \frac{1}{2}\tilde{K}\tilde{\boldsymbol{\gamma}} \cdot \tilde{\boldsymbol{\gamma}}, \tag{2.12}$$

where K denotes the apparent elastic stiffness, and the overtilde denotes an effective quantity (e.g., the initial elastic stiffness, $\tilde{K}$, remains constant throughout separation). Following Betten [40], let $\tilde{\boldsymbol{\tau}}$ be related to $\boldsymbol{\tau}$ by

$$\tilde{\boldsymbol{\tau}} = \frac{\boldsymbol{\tau}}{1-d} \quad \text{or} \quad \boldsymbol{\tau} = (1-d)\,\tilde{\boldsymbol{\tau}}. \tag{2.13}$$

Hooke's law can be expressed as

$$\boldsymbol{\tau} = K\boldsymbol{\gamma} \quad \text{and} \quad \tilde{\boldsymbol{\tau}} = \tilde{K}\tilde{\boldsymbol{\gamma}} \tag{2.14}$$

in the apparent and the effective spaces, respectively. Combining Eqs. (2.12), (2.13), and (2.14) gives

$$K = (1-d)^2 \tilde{K}. \tag{2.15}$$

The damage model describes the element behavior under mixed-mode I/II/III delamination. Its associated parameters ought to be calibrated via a series of flexural tests, which include but are not limited to standard DCB, ENF, and MMB tests. Assume that the element obeys damage criterion

$$f = J(\hat{\boldsymbol{\gamma}})\, y - k - A(\alpha) \leq 0, \tag{2.16}$$

where J is referred to as the path dependence function,

$$\hat{\boldsymbol{\gamma}} = \frac{\boldsymbol{\gamma}}{\sqrt{\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}}} \equiv \frac{\boldsymbol{\gamma}}{\gamma} \tag{2.17}$$

denotes the unit vector in the direction of $\boldsymbol{\gamma}$ with γ denoting the magnitude of $\boldsymbol{\gamma}$, and k denotes the initial damage threshold. The principle of maximum dissipation requires the damage surface to be convex (see Simo and Hughes [37] for more details). This imposes no restrictions on $J(\hat{\boldsymbol{\gamma}})$. More details about J are introduced in Appendix B. Substituting Eq. (2.12) into the second equation of Eq. (2.6) gives

$$y = -\frac{\partial \Psi_e}{\partial d} = (1-d)\,\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}. \tag{2.18}$$

Let y_0 denote the value of y at damage initiatio and let $\boldsymbol{e}_1$ represent the unit vector in the x_1 direction. Set $J(\boldsymbol{e}_1) = 1$ so that:

1. For a test producing mode I delamination, $k = y_0|_{\hat{\boldsymbol{\gamma}}=\boldsymbol{e}_1}$;

2. For a test during which $\hat{\boldsymbol{\gamma}} = \hat{\boldsymbol{\gamma}}_i$ (where $(\cdot)_i$ denotes the value of a quantity for the ith test), $J(\hat{\boldsymbol{\gamma}}_i) = k \Big/ \Big($

Suppose that a set of n data points $(\hat{\boldsymbol{\gamma}}_i, J_i)$ have been calibrated via a series of flexural tests during which $\hat{\boldsymbol{\gamma}}_i$'s remained constant. $J(\hat{\boldsymbol{\gamma}})$ is then a multivariate Lagrange polynomials constructed from these data points (see Appendix A and B for more

details). The damage accumulation law, $A = A(\alpha)$, must be an increasing function. Numerical experiments indicate that

$$A = Q\left[\exp(b\alpha) - 1\right] \tag{2.19}$$

serves the purpose well, where Q and b are two positive damage accumulation parameters to be calibrated. Last but not least, substituting Eq. (2.16) into Eq. (2.10) gives

$$\dot{d} = \dot{\lambda} J \quad \text{and} \quad \dot{\alpha} = \dot{\lambda}. \tag{2.20}$$

Till now, the damage model has been full specified, and its associated parameters, such as k, Q, b, and J_i's, are to be calibrated.

2.2 Implicit Integration Scheme

Nowadays, a CZM can be easily implemented in the commercial finite element code Abaqus/Standard via user subroutine UMAT for structural analysis. UMAT can: 1. define the traction–separation relation for a cohesive element; 2. use and update any state variables, either passed in or stored elsewhere [9]. Let $(\cdot)_n$ denote a quantity at a given instant of time, t_n, and let $\Delta(\cdot)$ denotes the increment in a quantity over time interval $[t_n, t_{n+1}]$. When calling UMAT at an integration point, Abaqus/Standard passes in $\boldsymbol{\tau}_n$, $\boldsymbol{\gamma}_n$ and $\Delta\boldsymbol{\gamma}$ and gets back $\boldsymbol{\tau}_{n+1}$ and the consistent Jacobian, $\partial\Delta\boldsymbol{\tau}/\partial\Delta\boldsymbol{\gamma}$. Suppose that all the variables at t_n, as well as $\Delta\boldsymbol{\gamma}$, are known. The implicit integration scheme should update the other variables and return the consistent Jacobian.

For notational convenience, omit the subscript $n+1$ on each quantity at t_{n+1}. Note that Eqs. (2.16) and (2.18) should be satisfied at each instant of time. Substituting Eq. (2.18) into Eq. (2.16) gives

$$f = J(\hat{\boldsymbol{\gamma}}) \cdot (1-d)\,\tilde{K}\boldsymbol{\gamma} \cdot \boldsymbol{\gamma} - k - A(\alpha) \leq 0. \tag{2.21}$$

By definition,

$$d = d_n + \Delta d \quad \text{and} \quad \alpha = \alpha_n + \Delta\alpha. \tag{2.22}$$

Eq. (2.20) can be rewritten as

$$\Delta d = J\left(\hat{\boldsymbol{\gamma}}\right)\Delta\lambda = J\left(\hat{\boldsymbol{\gamma}}\right)\Delta\alpha. \tag{2.23}$$

It turns out that, once Eq. (2.11) is met, the integration scheme can be formulated as solving the following equation for $\Delta\alpha$:

$$f\left(\Delta\alpha\right) = \left(J\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}\right)\left(1-d_n-J\Delta\alpha\right)-k-A\left(\Delta\alpha\right)=0. \tag{2.24}$$

Here the Newton–Raphson method is used for problem solving. Require

$$f\left(\Delta\alpha_{\text{old}}+\mathrm{d}\Delta\alpha\right) = f\left(\Delta\alpha_{\text{old}}\right)+f'\left(\Delta\alpha\right)\mathrm{d}\Delta\alpha = 0, \tag{2.25}$$

where

$$f'\left(\Delta\alpha\right) = -J^2\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}-A'\left(\Delta\alpha\right). \tag{2.26}$$

Substituting Eq. (2.26) into Eq. (2.25) and solving for $\mathrm{d}\Delta\alpha$ give

$$\mathrm{d}\Delta\alpha = \frac{f\left(\Delta\alpha_{\text{old}}\right)}{J^2\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}+A'\left(\Delta\alpha\right)}. \tag{2.27}$$

The corrections can then be computed and added to the solutions, i.e.,

$$\Delta\alpha_{\text{new}} = \Delta\alpha_{\text{old}}+\mathrm{d}\Delta\alpha. \tag{2.28}$$

Once the process is iterated to convergence, the variables are updated as

$$\alpha = \alpha_n+\Delta\alpha, \quad d = d_n+J\left(\hat{\boldsymbol{\gamma}}\right)\Delta\alpha, \quad \boldsymbol{\tau} = \left(1-d\right)^2\tilde{K}\boldsymbol{\gamma}. \tag{2.29}$$

In Eq. (2.29), d is explicitly updated because $J\left(\hat{\boldsymbol{\gamma}}\right)$ may vary with time. Accordingly, the implicit integration scheme is only semi-implicit and conditionally stable. In other words, it may be unstable when the time increment size is too large. Fortunately, Abaqus/Standard automatically adjusts the time increment size using so-called automatic incrementation control [9].

The consistent Jacobian awaits determination. Note that hereafter $\boldsymbol{\gamma}$ is no longer held fixed. Accordingly, f in Eq. (2.24) becomes a function of $\Delta\alpha$ and $\boldsymbol{\gamma}$, i.e.,

$$f\left(\Delta\alpha,\boldsymbol{\gamma}\right) = \left[J\left(\hat{\boldsymbol{\gamma}}\right)\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}\right]\left(1-d_n-J\left(\hat{\boldsymbol{\gamma}}\right)\Delta\alpha\right]-k-A\left(\Delta\alpha\right)=0. \tag{2.30}$$

It is beneficial to relate $\mathrm{d}\hat{\boldsymbol{\gamma}}$ to $\mathrm{d}\boldsymbol{\gamma}$. Totally differentiating both sides of Eq. (2.17) gives

$$\mathrm{d}\hat{\boldsymbol{\gamma}} = \frac{\mathrm{d}\boldsymbol{\gamma}}{\gamma} - \frac{\mathrm{d}\gamma}{\gamma^2}\boldsymbol{\gamma}. \tag{2.31}$$

Following Benzerga and Besson [41], express $\mathrm{d}\gamma$ as

$$\mathrm{d}\gamma = \frac{\mathrm{d}\gamma^2}{2\gamma} = \frac{\mathrm{d}\left(\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}\right)}{2\gamma} = \frac{\boldsymbol{\gamma}\cdot\mathrm{d}\boldsymbol{\gamma}}{\gamma}. \tag{2.32}$$

Substituting Eq. (2.32) into Eq. (2.31) gives

$$\mathrm{d}\hat{\boldsymbol{\gamma}} = \frac{\mathrm{d}\boldsymbol{\gamma}}{\gamma} - \frac{\boldsymbol{\gamma}\cdot\mathrm{d}\boldsymbol{\gamma}}{\gamma^3}\boldsymbol{\gamma} = \frac{\left(\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}\right)\boldsymbol{I} - \boldsymbol{\gamma}\otimes\boldsymbol{\gamma}}{\gamma^3}\cdot\mathrm{d}\boldsymbol{\gamma} \equiv \boldsymbol{A}\cdot\mathrm{d}\boldsymbol{\gamma}. \tag{2.33}$$

On one hand, totally differentiating both sides of Eq. (2.30) with respect to $\Delta\alpha$ and $\boldsymbol{\gamma}$ gives

$$\mathrm{d}f = \frac{\partial f}{\partial\Delta\alpha}\mathrm{d}\Delta\alpha + \frac{\partial f}{\partial\boldsymbol{\gamma}}\cdot\mathrm{d}\boldsymbol{\gamma} = 0, \tag{2.34}$$

where

$$\frac{\partial f}{\partial\Delta\alpha} = -J^2\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma} - A'\left(\Delta\alpha\right), \tag{2.35}$$

$$\frac{\partial f}{\partial\boldsymbol{\gamma}} = \left(1-d_n-2J\Delta\alpha\right)\left(\tilde{K}\boldsymbol{\gamma}\cdot\boldsymbol{\gamma}\right)\frac{\partial J}{\partial\hat{\boldsymbol{\gamma}}}\cdot\boldsymbol{A} + 2\left(1-d_n-J\Delta\alpha\right)J\tilde{K}\boldsymbol{\gamma}. \tag{2.35$'$}$$

Solving Eq. (2.34) for $\mathrm{d}\Delta\alpha$ gives

$$\mathrm{d}\Delta\alpha = -\frac{\frac{\partial f}{\partial\boldsymbol{\gamma}}}{\frac{\partial f}{\partial\Delta\alpha}}\cdot\mathrm{d}\boldsymbol{\gamma} \equiv \boldsymbol{B}\cdot\boldsymbol{\gamma} \tag{2.36}$$

On the other hand, totally differentiating both sides of the first equation of Eq. (2.14) with respect to Δd and $\boldsymbol{\gamma}$ gives

$$\mathrm{d}\boldsymbol{\tau} = \mathrm{d}K\boldsymbol{\gamma} + K\mathrm{d}\boldsymbol{\gamma}. \tag{2.37}$$

Totally differentiating both sides of Eq. (2.15) gives

$$\mathrm{d}K = -2\left(1-d\right)\tilde{K}\mathrm{d}\Delta d. \tag{2.38}$$

It is beneficial to relate $\mathrm{d}\Delta d$ to $\mathrm{d}\Delta\alpha$. Totally differentiating both sides of Eq. (2.23) gives

$$\mathrm{d}\Delta d = J\mathrm{d}\Delta\alpha + \Delta\alpha\frac{\partial J}{\partial\hat{\boldsymbol{\gamma}}}\cdot\boldsymbol{A}\cdot\mathrm{d}\boldsymbol{\gamma}. \tag{2.39}$$

Substituting Eq. (2.36) into Eq. (2.39) gives

$$\mathrm{d}\Delta d = \left(J\boldsymbol{B} + \Delta\alpha \frac{\partial J}{\partial \hat{\boldsymbol{\gamma}}} \cdot \boldsymbol{A} \right) \cdot \mathrm{d}\boldsymbol{\gamma}. \tag{2.40}$$

Combining Eqs. (2.37), (2.38), and (2.40) gives

$$\mathrm{d}\boldsymbol{\tau} = \left[2\left(1-d\right)\tilde{K}\boldsymbol{\gamma} \otimes \left(J\boldsymbol{B} + \Delta\alpha \frac{\partial J}{\partial \hat{\boldsymbol{\gamma}}} \cdot \boldsymbol{A} \right) + K\boldsymbol{I} \right] \cdot \mathrm{d}\boldsymbol{\gamma} \equiv \boldsymbol{L} \cdot \mathrm{d}\boldsymbol{\gamma}. \tag{2.41}$$

Last but not least, the definition of $\boldsymbol{\gamma}$ (Eq. (2.3)) indicates that, when $\gamma_1 < 0$, the relationship between τ_1 and γ_1 is unspecified. To prevent crack face penetration, set each cohesive element to have a small thickness so that a negative γ_1 is therefore admissible. Let $\tilde{K}_c$ denote the penalty contact stiffness such that

$$\tau_1 = \tilde{K}_c \gamma_1 \quad \text{or} \quad \mathrm{d}\tau_1 = \tilde{K}_c \mathrm{d}\gamma_1 \tag{2.42}$$

when $\gamma_1 < 0$. Numerical experiments indicate that $\tilde{K}_c = 1000\tilde{K}$ serves the purpose well. Let

$$\boldsymbol{I}^+ = \begin{bmatrix} H\left(\gamma_1\right) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.43}$$

where $H\left(x\right)$ denotes the Heaviside step function, i.e.,

$$H\left(x\right) = \begin{cases} 0 & x < 0, \\ 1 & x \geq 0. \end{cases} \tag{2.44}$$

Combining Eqs. (2.41)–(2.43) gives

$$\begin{Bmatrix} \mathrm{d}\tau_1 \\ \mathrm{d}\tau_2 \\ \mathrm{d}\tau_3 \end{Bmatrix} = \left[\boldsymbol{I}^+ \cdot \boldsymbol{L} + \tilde{K}_c \left(\boldsymbol{I} - \boldsymbol{I}^+ \right) \right] \begin{Bmatrix} \mathrm{d}\gamma_1 \\ \mathrm{d}\gamma_2 \\ \mathrm{d}\gamma_3 \end{Bmatrix} \equiv \boldsymbol{L}^* \begin{Bmatrix} \mathrm{d}\gamma_1 \\ \mathrm{d}\gamma_2 \\ \mathrm{d}\gamma_3 \end{Bmatrix}, \tag{2.45}$$

where $\boldsymbol{L}^*$ turns out to be the consistent Jacobian. It can be verified that:

1. When $\gamma_1 \geq 0$, $\boldsymbol{L}^* = \boldsymbol{L}$;

2. Otherwise, the relationship between $\mathrm{d}\tau_1$ and $\mathrm{d}\gamma_1$ obeys Eq. (2.42).

3. FINITE ELEMENT FRAMEWORK

In this chapter, discussions are made on experimental setups and finite element models of DCB, ENF, and MMB specimens. The way to determine the finite element solver (i.e. time integration scheme) and mesh density to get stable and convergent results is also presented. The actual models and parameters are presented more detailedly in Chapter 5.

3.1 Experimental Apparatus

The test methods for determination of interlaminar fracture toughness G_c (equivalently, critical energy release rate) of continuous fiber-reinforced composite materials at various mode I to mode II mix ratio have been standardized [1–3]. It would be helpful to illustrate the setup of the MMB test first as the DCB and ENF tests can be considered as a simplified version of the MMB test.

The MMB test apparatus shown in Fig. 1.1 is used to to measure G_c at various mix ratio for a laminated composite. The test specimen is made up of a rectangular unidirectional laminated composite with a non-adhesive insert at the mid-plane to serve as a pre-crack. Loads are applied through a lever connected to a tab applied near the ends of the pre-crack and a roller placed at the middle in the non-delamination region. The roller acts as a fulcrum so the tab is pulled up when the lever is pushed down at one end. The forces at the tab and the roller are opposite in directions and the ratio between them could be adjusted with the lever length c. The applied load is recorded versus the load point displacement and the critical energy release rate G_c could be calculated thereafter. As is pointed out in the Chapter 1 and would be further illustrated in Chapter 5, the theoretical foundation of ASTM standards is not

perfect. Despite this fact, the experimental apparatus is still useful and the G_c values calibrated by the ASTM standard would serve as an initial guess.

The lever length c should be determined by the desired mode mix ratio prior to experiments. Using the simple beam theory, an analytical relation is derived by Camanho and Dávila [10]:

$$c = L\frac{\frac{1}{2}\sqrt{3\left(\left(\frac{1-\kappa}{\kappa}+1\right)\right.}}{3-\frac{1}{2}\sqrt{3\left(\left(\frac{1-\kappa}{\kappa}\right)\right.}}, \tag{3.1}$$

where

$$\kappa = \frac{G_{II}}{G_I + G_{II}} \tag{3.2}$$

denotes the mode mix ratio. Detailed discussion about energy release rate G_I and G_{II} could be found in Appendix A. The ASTM standard made corrections to the result by adopting the modified beam theory, which introduced a virtual crack length χh to account for the deflection and rotation at the crack tip [42]. The corrected lever length is given by

$$c = \frac{12\beta^2 + 3\alpha + 8\beta\sqrt{3\alpha}}{36\beta^2 - 3\alpha}L, \tag{3.3}$$

where

$$\begin{aligned} \alpha &= \frac{1-\kappa}{\kappa} = \frac{G_I}{G_{II}}, \\ \beta &= \frac{a+\chi h}{a+0.42\chi h}, \end{aligned} \tag{3.4}$$

and a denotes the delamination length.

The DCB and ENF tests share the same test specimen except that the lever is removed. For DCB, the loads are applied at the bottom and top beam in the delamination region. For ENF, the load is applied with a roller at the center in the nondelamination region.

3.2 Finite Element Models

The test specimen could be modeled with a 3D solid model or a 2D plane stress model. Both models have been built and the UMAT subroutine is designed and tested to work in either case. The model consists of a two layer composite plate with a cohesive layer between them. The laminate is modeled with brick elements for 3D or quad elements for 2D while the cohesive layer is modeled with the built-in cohesive element in Abaqus. A schematic 2D finite element model of MMB is shown in Figure 3.1.

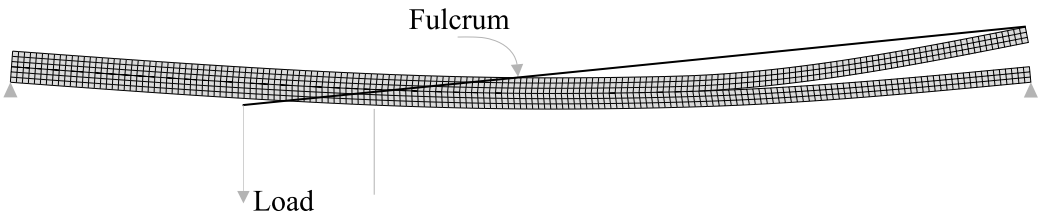


Figure 3.1. Schematic of a MMB finite element model.

The lever is modeled as a rigid body. One end of the lever is tied to a beam in the delamination region. Meanwhile, a small region of the lever near the center of the laminate serves as the fulcrum since it is constrained to slide freely on the surface of the laminate and other parts of the lever are set to have no interaction with the laminate. The load is applied at the far end of the lever as shown in the figure.

By prescribing that the cohesive elements have the same size in the longitudinal direction as the adjacent laminate elements, it is straightforward to bond the cohesive elements to laminate elements. In the thickness direction, a fairly small element size is assigned for the cohesive elements (less than 1/1000 of the laminate element thickness). The fictitious geometric thickness has no effect on the final result according to the implementation of cohesive elements in Abaqus [9].

3.3 Time Integration Scheme

An implicit solving process is preferred but not commonly used. As a summary, finite element solvers are classified into implicit solvers and explicit solvers by their time integration scheme. There are some major differences between them:

1. Implicit solvers apply to static problems while explicit solvers apply to dynamics problems. However, quasi-static problems could be handled with either solver.

2. Theoretically, implicit solvers are more accurate since it tries to find the exact solution by iterations. It is tricky to deal with kinetic energy and artificial viscosity when explicit solvers are used to solve a quasi-static problem.

Numerous efforts have been made in simulating CZM with explicit methods. Oscillation is a common problem because of the dynamic nature of the explicit solver [20, 25, 26]. The reasons people prefer explicit solvers might be:

1. The implicit solver often experiences convergence difficulties. The computation is most likely to fail when a cohesive element reaches the critical energy release rate.

2. The implicit solver requires an implicit material model to make it truly implicit. The process often involves iteration and is not as straightforward as solving the material model using Euler method explicitly.

Efforts are made to ensure the convergence of internal material model iteration. The Newton method with line search is used to solve the material model [43]. A very strict convergence criterion is enforced when possible and it is relaxed a little when the element reaches the critical point of breakage. Finite element calculations are also conducted with the Newton method in Abaqus/Standard. The Riks method [43] which theoretically has better convergence performance proves to be unnecessary in the specific problem of CZMs.

3.4 Mesh Study

The CZM simulation is very sensitive to mesh density. Coarse mesh could lead to solving failure or an unconvergent result. Considerations when meshing the structures include:

1. There are supposed to be enough elements in the thickness direction of the beam. Meanwhile, the reduced integration could be used.

2. The FPZ of the cohesive layer needs to be adequately resolved. This could be achieved by estimating the FPZ size analytically and insert enough elements in this region.

Last but not least, the result could be justified through a mesh refinement study. Refine the mesh, and compare the result until a convergent result is achieved.

4. SYSTEMATIC CALIBRATION METHOD

ASTM standard DCB, ENF, and MMB tests are among the most widely used tests for measuring mode I, mode II, and mixed-mode I/II interface properties. Although the present CZM can handle mixed-mode I/II/III delamination, there is still a lack of standard flexural tests capable of producing such delamination. Hereafter assume that a cohesive element has the same properties in all directions in the x_2x_3 plane. In this case, $J(\hat{\boldsymbol{\gamma}})$ reduces to a univariate Lagrange polynomial (see Appendix A for more details). Figure 4.1 depicts a systematic method for calibrating interface parameters via ASTM standard flexural tests. It consists of three major steps:

1. Create an initial guess for the CZM;
2. Implement the initial guess in Abaqus/Standard for structural analysis;
3. Calibrate the interface parameters through trial and error.

In the remaining of this chapter, these three steps will be described in detail.

4.1 Initial Guess

An initial guess can be created based on:

1. An estimate of the G_c–β_2 relationship, e.g., that predicted by the Euler–Bernoulli beam theory along with the Griffith theory of brittle fracture;
2. For the DCB test, estimates of the interface strength and the corresponding separation, say $\tau_{\mathrm{I}\max}$ and $\gamma_{\mathrm{I}\max}$, either observed during experiments or obtained through numerical experiments.

Following Zhang et al. [21], assume that a cohesive element exhibits softening right after damage initiation. This assumption, together with Eq. (2.14), implies that

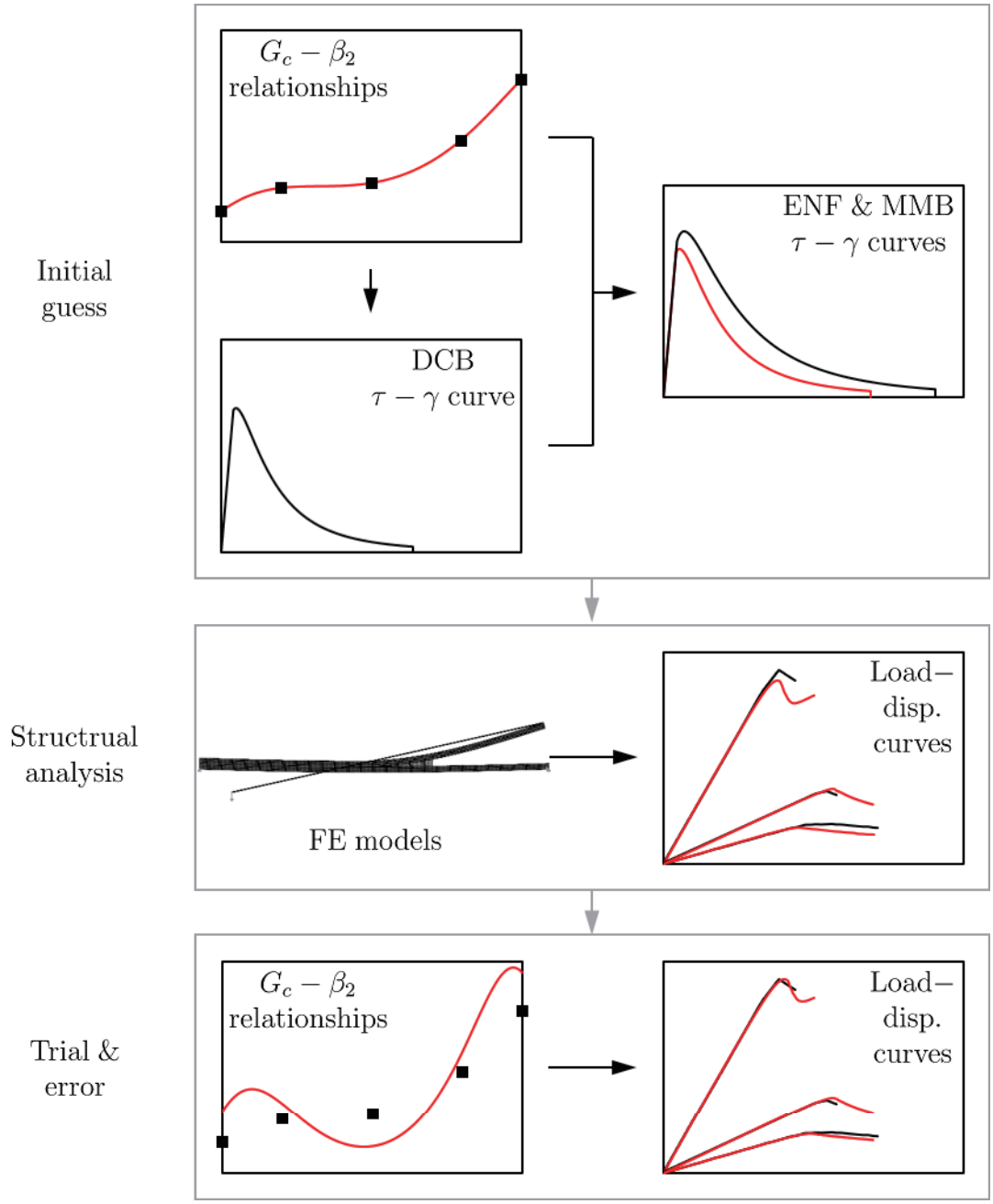


Figure 4.1. Systematic calibration method.

$\tilde{K} \approx \tau_{\mathrm{I}\max}/\gamma_{\mathrm{I}\max}$. $\tilde{K}$ is a thermodynamic quantity here rather than a penalty stiffness proposed by Camanho [10]. Its value should be properly chosen to produce realistic damage evolution. Otherwise, the element may undergo a sudden decrease in d (from 0 to almost 1) right after damage initiation. Let $\gamma_{\mathrm{I}c}$ denote the critical separation at the onset of delamination, during the DCB test. For a bilinear CZM, $G_{\mathrm{I}c} = \frac{1}{2}\tau_{\mathrm{I}\max}\gamma_{\mathrm{I}c}$. For the present CZM, although $G_{\mathrm{I}c}$, $\tau_{\mathrm{I}\max}$, and $\gamma_{\mathrm{I}c}$ are implicitly linked, for fixed $G_{\mathrm{I}c}$, $\tau_{\mathrm{I}\max}$ still increases with decreasing $\gamma_{\mathrm{I}c}$. Also for fixed $G_{\mathrm{I}c}$, an extremely high $\tau_{\mathrm{I}\max}$ corresponds to a stiff but brittle interface, while an extremely low $\tau_{\mathrm{I}\max}$ a flexible but ductile interface. A former interface often causes unstable delamination during the ENF and some MMB tests, while a latter interface often causes considerable specimen degradation even before the visually observed onset of delamination. Although the load–displacement curves are found to be insensitive to the interface strength (see Chapter 5 for more details), the assumed interface strengths should at least have the same order of magnitude as the real one, to avoid unrealistic predictions.

Given $G_{\mathrm{I}c}$, $\tau_{\mathrm{I}\max}$, and $\gamma_{\mathrm{I}\max}$, k can be determined as shown in Section 2.1 and Q and b in Eq. (2.19) await calibration. Here the present implicit integration scheme is implemented in an incremental–iterative procedure for time integration, and the method of nonlinear least squares is used for calibration. The latter method is not guaranteed to converge to the global optimum and often gets "lost" if started far from the solution. Setting the starting values close to the solution can greatly improve the convergence. Here such starting values are obtained through a series of Monte Carlo experiments. The inputs to the incremental–iterative procedure are then $\tau_{\mathrm{I}\max}$, $\gamma_{\mathrm{I}\max}$, k, Q, and b. To facilitate the calibration, invoke the following assumption only at this step: during each test, each element is deformed along the same proportional separation path. This is the equivalent of setting $\hat{\boldsymbol{\gamma}}$ to be constant everywhere. With this assumption, $\boldsymbol{A}$ in Eq. (2.41) vanishes, and performing the incremental–iterative procedure along $\hat{\boldsymbol{\gamma}} = \boldsymbol{e}_1$ gives the DCB traction–separation curve, which suggests $G_{\mathrm{I}c}$ and $\tau_{\mathrm{I}\max}$ different from the estimated ones. Q and b can then be calibrated by iteratively adjusting their values till they make the predicted $G_{\mathrm{I}c}$ and $\tau_{\mathrm{I}\max}$ have

the best fit to the estimated ones. After this, J_i's (expect J_1) await calibration. Performing the incremental–iterative procedure along $\hat{\boldsymbol{\gamma}} = \hat{\boldsymbol{\gamma}}_i$ gives the ith traction–separation curve, which is the ENF or an MMB curve. Since the ith curve depends on J_i rather than its neighborhood, J_i's can then be calibrated one by one, from the estimated G_c–β_2 relationship. The calibration process here is similar to the above one expect that J_i is the fitted parameter each time. After this, the initial guess for the CZM is ready to use.

4.2 Structural Analysis

The present CZM is implemented in Abaqus/Standard via UMAT, for structural analysis. The implementation has been briefly described in Chapter 2, and more details can be found in Appendix C. Different chosen options on the finite element models of test specimens lead to different predictions. The most rigorous option is to choose 3D models. Several authors [20, 22, 25, 26, 44, 45] used explicit finite element analyzers along with such models but obtained some oscillatory load–displacement curves. Numerical experiments indicate that it is very time consuming for Abaqus/Standard, along with a bilinear CZM, to simulate the DCB test, not to mention more complex MMB tests. One major reason is that a complex finite element model is often accompanied with a low convergence rate and low accuracy. This leads one to choose simplified models. Noting that the laminate thickness is much smaller than the span length, the second option becomes idealizing each laminate as a plate [10,46,47]. Despite improved efficiency, a plate theory oversimplifies the stress distribution over the thickness direction of each laminate. As mentioned above, even though the laminate thickness is small, it is still comparable to the FPZ length and therefore non-negligible. Noting that each laminate has a small width and is unconstrained in its width direction, the third option becomes idealizing each laminate as a 2D beam and assuming that plane stress conditions prevail [4, 45, 48]. Numerical results indicate that the stress fields produced by such models are more sophisticated

than those predicted by the Euler–Bernoulli beam theory, especially around interface crack tips. Therefore, the third option is chosen throughout structural analysis. With the initial guess and the chosen option, Abaqus/Standard can predict a series of load–displacement curves. These curves, however, most likely deviate from the experimental data, indicating that the initial guess needs to be tuned.

4.3 Trial and Error

Let $(P_{i\max}, u_{i\max})$ denote the peak point of the ith load–displacement curve, where $P_{i\max}$ and $u_{i\max}$ denote the peak load and the corresponding displacement, respectively. Further let $\bar{P}_{i\max} = P_{i\max}/P_{1\max}$ and $\bar{u}_{i\max} = u_{i\max}/u_{1\max}$. Step 3 can then be reformulated as finding the values of some interface parameters making the predicted $\left(\bar{P}_{i\max}, \bar{u}_{i\max}\right)$ have the best fit to the experimental ones. The predicted load–displacement curves provide some valuable insights into the problem. Numerical results indicate that:

1. During the DCB or the ENF test, each element is deformed along the same proportional separation path (i.e., $\hat{\boldsymbol{\gamma}} = \boldsymbol{e}_1$ during the DCB test and $\hat{\boldsymbol{\gamma}} = \boldsymbol{e}_2$ during the ENF test);

2. During each MMB test, each element is deformed along its respective nonproportional separation path.

This allows one to accomplish the task as follows:

1. For the DCB test, hold $\tau_{1\max}$, $\bar{Q} = Q/k$, and b fixed, and iteratively adjust $\tilde{K}$ till the predicted $(P_{1\max}, u_{1\max})$ have the best fit to the experimental one;

2. For all other tests, calibrate J_i's from the estimated G_c–β_2 relationship, as described above;

3. From the second to the nth test, in turn adjust each J_i using interpolation and extrapolation, and iterate this process till the predicted $(P_{i\max}, u_{i\max})$'s have the best fit to the experimental ones.

At Step 1, the reasons for adjusting $\tilde{K}$ rather than the other parameters include:

1. As mentioned above, $\tau_{\mathrm{I}\max}$ is properly chosen to avoid unrealistic predictions, and it is therefore preferable to hold $\tau_{\mathrm{I}\max}$ fixed;

2. Numerical experiments indicate that holding $\gamma_{\mathrm{I}\max}/\gamma_{\mathrm{I}c}$ fixed makes $\bar{Q}$ and b a kind of eigenvalues (i.e., given $\bar{Q}$ and b, all traction–separation curves exhibit similar trends regardless the values of J_i's);

3. On one hand, $\tilde{K}$ turns out to be the most flexible parameter; on the other hand, a good initial guess of the CZM makes an adjusted $\tilde{K}$ always acceptable.

Step 2 is necessary because:

1. A change in $\tilde{K}$ suggests a change in $G_{\mathrm{I}c}$ and therefore changes in J_i's ($i \neq 1$, because J_1 must equal 1);

2. Step 2 ensures that the current J_i's are still good initial guesses.

On one hand, since $\hat{\boldsymbol{\gamma}}$ remains constant throughout the DCB test, the DCB load–displacement curve remain unchanged at Step 3. On the other hand, since each element's $\hat{\boldsymbol{\gamma}}$ keeps varying during an MMB test, adjusting one J_i ($i \neq 1$) inevitably affects all predicted $(P_{i\max}, u_{i\max})$'s ($i \neq 1$). Fortunately, the task is formulated such that there exists a unique set of J_i's whose corresponding $(P_{i\max}, u_{i\max})$'s are the best fits.

5. RESULTS AND DISCUSSION

Reeder and Crews [49] performed a series of DCB, ENF, and MMB tests on a type of unidirectional APC-2 PEEK composite beams, measured the applied load–load point displacement curve during each test, and computed the critical energy release rate–mode mixture ratio relationship using the the Euler–Bernoulli beam theory along with the Griffith theory of brittle fracture. In this chapter, the interface parameters associated with the present CZM will be calibrated from the experimental data, using the systematic calibration method presented in Chapter 4. For comparison purposes, the predictions by the present CZM will be compared with those by a bilinear CZM. Meanwhile, the separation path during each test will be investigated, and the effect of the interfacial strength on a load–displacement curve will be evaluated.

5.1 Initial Guess

In this section, the initial guess of model parameters is obtained following the aforementioned systematic calibration process. Table 5.1 lists experimental values including critical traction at damage initiation $\tau_{\mathrm{I}0}$ and combinations of mode mix ratios (β_2) and the corresponding critical energy release rates (G_c). Interfacial stiffness $\tilde{K}$ and initial damage threshold k also listed in Table 5.1 are determined through simple calculations as described in Section 4.1. A reasonable value of $3.3 \times 10^4\,\mathrm{N/mm^3}$ is chosen while the value recommended by Camanho and Dávila [10] is $10^6\,\mathrm{N/mm^3}$. A smoother transition into damage period is observed with the chosen $\tilde{K}$ value. The damage parameters Q, b and the path dependence function J calibrated from interfacial strength $\tau_{\mathrm{I}0}$ and critical energy release rates G_c through nonlinear least square are listed in Table 5.2.

Table 5.1. Predetermined interface parameters.

(a) Estimated G_c–β_2 relationship

	DCB	20% MMB	50% MMB	80% MMB	ENF
β_2	0.0	0.2	0.5	0.8	1.0
$G_c\,(\mathrm{J/m^2})$	969	1103	1131	1376	1719

(b) Other parameters.

$\tilde{K}\,(\mathrm{N/mm^3})$	$\tau_{\mathrm{I}0}\,(\mathrm{MPa})$	$k\,(\mathrm{J/m^2})$	$\tau_{\mathrm{Imax}}\,(\mathrm{MPa})$	d_c
3.30237×10^4	80.0	193.8	81.2	0.95

Table 5.2. Fitted interface parameters.

(a) Fitted values of Q and b.

$Q\,(\mathrm{J/m^2})$	b
332.86	2.15227

(b) Fitted values of J

	DCB	20% MMB	50% MMB	80% MMB	ENF
J	1.0	0.951839	0.942956	0.877952	0.812779

It is beneficial to investigate the traction separation curves and compare the results with the results of the bilinear model. The results are shown in Figure 5.2. There are two major differences between the present CZM and the bilinear model. First, the present model fails at higher strain, which means the element is more ductile. Second, the bilinear model fails to distinguish the DCB specimen and the 20% MMB specimen while they are distinct in the present model. The defect of the bilinear model is due to the limitation of B-K criterion. The reason is better illustrated in Figure 5.2, in which the bilinear model cannot reproduce the critical energy release rates and has the largest discrepancy for 20% MMB.

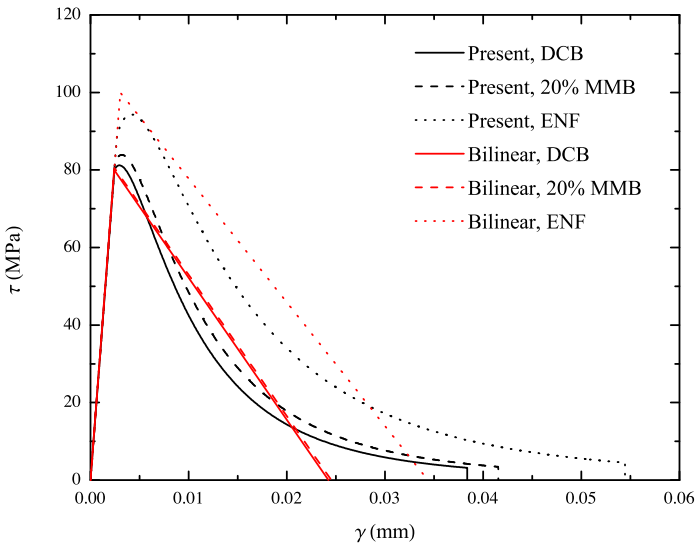


Figure 5.1. Estimated traction–separation curves.

5.2 Structural Analysis

Structural analysis is conducted to test the chosen model parameters. The experimental setup in Reeder and Crews's experiments are modeled with the present model [49]. The experimental setups are shown in Figure 1.1. Each laminate has s

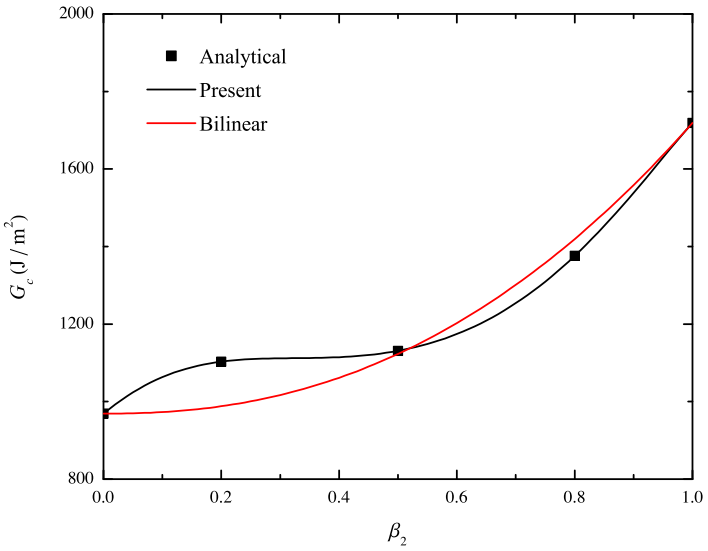


Figure 5.2. Estimated critical energy release rate–mode mix ratio curves.

span length (L) of 102 mm, a width (b) of 25.4 mm, and a thickness (h) of 1.56 mm. Table 5.3 lists the elastic constants, initial delamination lengths (a_0) and experimental lever lengths (c) of the test specimens. Abaqus/Standard with a UMAT subroutine is then used to simulate the delamination problem. Plane stress is assumed as discussed previously. Each beam of the laminate is meshed with 510×6 4-node quadrilateral elements (CPS4). The cohesive layer is meshed with 510 4-node quadrilateral elements (COH2D4). An illustrative finite element model with coarse mesh is shown in Figure 3.1. In order to justify the mesh, the FPZ is estimated to be [50]

$$l_{s,\mathrm{I}} = M \left(E_1 \frac{G_{\mathrm{Ic}}}{\tau_{\mathrm{I}0}^2} h^3 \right)^{1/4}. \tag{5.1}$$

The scaling factor M is theoretically close to unity but a value of 0.5 is suggested by Harper and Hallett [26] to better estimate the actual $l_{s,\mathrm{I}}$. A conservative estimate of the FPZ length is 1.45 mm, which is about than 7 times of the cohesive elements length. According to Turon et al. [51], the element density should be adequate to resolve the FPZ. The suggestion is also verified by mesh convergence study.

Table 5.3. Experimental parameters.

(a) Elastic constants of each laminate.

E_1 (GPa)	$E_2 = E_3$ (GPa)	$G_{12} = G_{13}$ (GPa)	G_{23} (GPa)	$\nu_{12} = \nu_{13}$	ν_{23}
122.7	10.1	5.5	3.7	0.25	0.45

(b) Initial delamination lengths and lever lengths.

	DCB	20% MMB	50% MMB	80% MMB	ENF
a_0 (mm)	32.9	33.7	34.1	31.4	39.3
c (mm)	–	97.4	42.2	27.6	–

The load–displacement curves are shown in Figure 5.3, including the experimental results, simulation results based on the present model, and simulation results based on the bilinear model reported by Camanho and Dávila [10]. The curve for the DCB specimen is actually two times the displacement of a single beam. As shown in the

figure, the results from the present model is no worse than the bilinear model for now. One noteworthy imperfection is the 80% MMB specimen curve. The model fails even in the linear part. The discrepancy is also reported by Balzani and Wagner [46] for the same group of experimental data. A possible explanation is defects in the experimental data, such as inaccurate initial delamination length. The present model is superior in that:

1. The present model is more robust. It yields convergent results for any specimen while the bilinear model has convergence difficulty in simulating the MMB specimens with the implicit time integration scheme;

2. The present model has a solid thermodynamic foundation and it is able to reproduce the designated critical energy release rates.

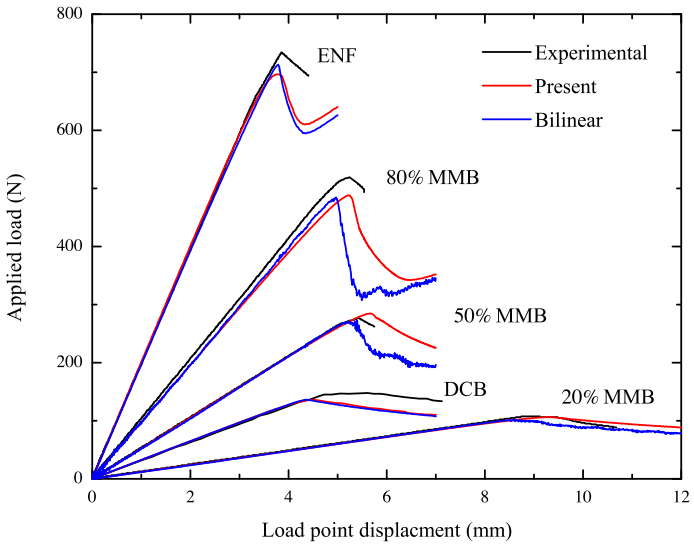


Figure 5.3. Estimated applied load–load point displacement curves.

5.3 Trial and Error

Due to the aforementioned imperfections of ASTM standards, the recommended values of the critical energy release rate is not an accurate measure. In structural design, the load carrying capability is the ultimate goal. Therefore, the path dependence function J is calibrated accordingly. In this section, the calibrated interfacial values and the modified flexural simulation results are presented.

The calibrated parameters are listed in Table 5.4. The stiffness constant $\tilde{K}$ is adjusted to be softer than the predetermined value. Changes in J_i's lead to changes in the theoretical critical energy release rate G_c's. Table 5.5 lists the estimated values together with the calibrated values of G_c, which is obtained with the adjusted J and by assuming a proportional separation path. The critical energy release rate G_c's by different methods are also shown in Figure 5.4. Despite the capability of recovering G_c, the path dependence function J is a preferred damage parameter in the present model. Eventually, the load–displacement curves are shown in Figure 5.5. Apparently, the present model agrees better than the bilinear model. The experimental values of maximum loads and predicted values by different models are listed in Table 5.6. The data for the bilinear model are reported by Camanho and Dávila [10]. It can be observed that the present model with implicit integration scheme is able to predict the maximum loads to high accuracy.

Table 5.4. Further tuned interface parameters.

(a) Fitted values of J

	DCB	20% MMB	50% MMB	80% MMB	ENF
J	1.0	0.98557	1.0780	0.90683	0.82618

(b) Values of other parameters.

$Q\,(\mathrm{J/m^2})$	b	$\tilde{K}\,(\mathrm{N/mm^3})$	$k\,(\mathrm{J/m^2})$
393.66	2.15226	27923.2	229.2

Table 5.5. Estimated and calibrated G_c's (J/m^2).

	DCB	20% MMB	50% MMB	80% MMB	ENF
Estimated	0.969	1.103	1.131	1.376	1.719
Calibrated	1.146	1.190	0.948	1.487	1.937

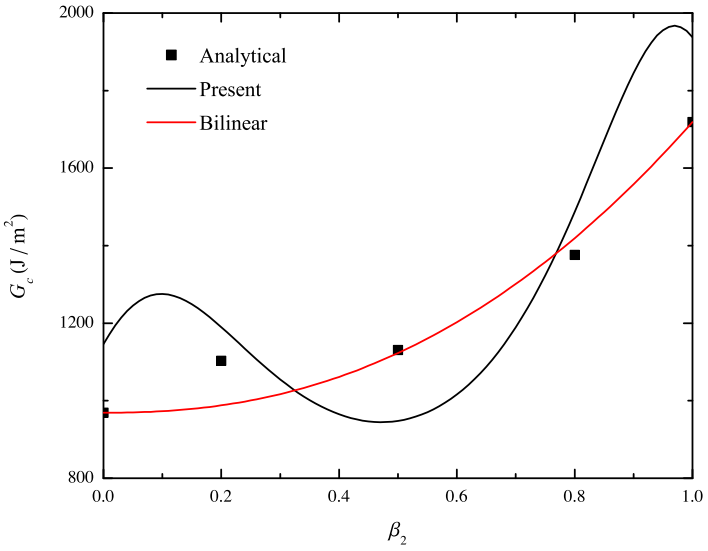


Figure 5.4. Calibrated critical energy release rate–mode mixture ratio curve.

Table 5.6. Peak applied loads (N) predicted by different CZMs.

	Experimental		Present		Bilinear	
	Load	Error	Load	Error	Load	Error
DCB	147.11	–	147.07	−0.02%	153.27	−4.2%
20% MMB	108.09	–	106.80	−1.2%	86.95	19.6%
50% MMB	275.35	–	281.70	2.3%	236.60	14.1%
80% MMB	518.66	–	510.52	−1.57%	479.86	7.5%
ENF	733.96	–	734.08	0.017%	695.94	5.2%

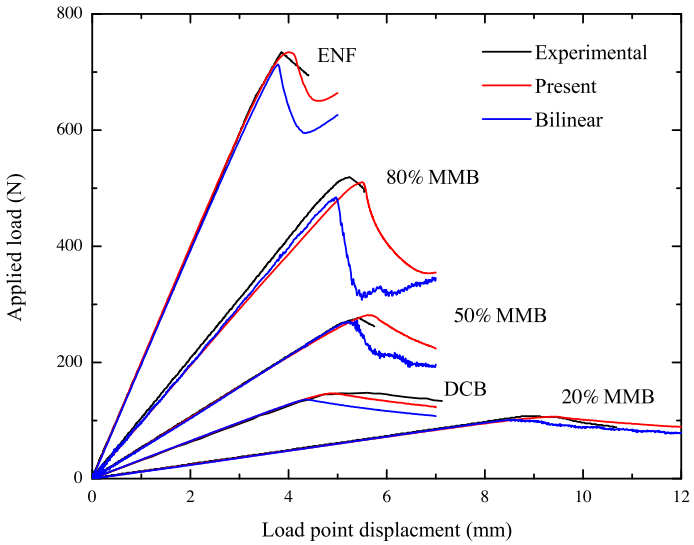


Figure 5.5. Fitted applied load–load point displacement curves.

5.4 Separation Paths

As mentioned before, Turon reported that the mode mix ratio of a MMB specimen varies as the damage grows [4]. Equivalently speaking, the separation path of a MMB specimen cannot be fixed. Therefore, it's beneficial to look into the separation path variation. Let $\theta = \tan^{-1}(\gamma_2/\gamma_1)$ denotes the angle of inclination of a cohesive element, which is the angle between the direction of the element and the x_1 axis. The separation path is constant if θ remains unchanged throughout the delamination.

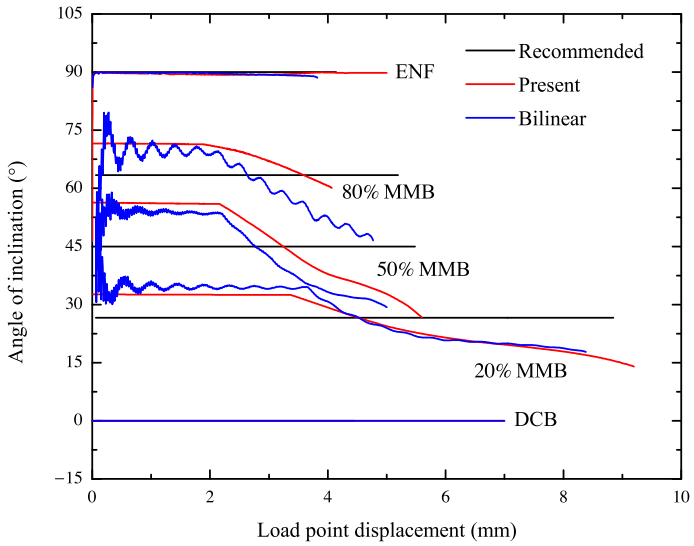


Figure 5.6. Angle of inclination for the rightmost elements.

Figure 5.6 shows the variation of orientation for the rightmost element in each test. The predicted curves for MMB lie above the recommended angle by the beam model and drop during the nonlinear region. The curves for DCB and ENF follow the expected value strictly. This explains the difficulty in modeling mixed mode specimen. On one hand, the varying mode mix ratio requires the model to capture the varying critical energy release rates. On the other hand, the phenomenon of separation path variation is not well considered in the current experimental standard. Therefore, the

critical energy release rates obtained from experiments are not strictly corresponding to the specified mode mix ratio. In order to compensate for the difficulty, we choose to use the parameters based on beam model as an initial guess and then adjust them to better fit to the critical loads. The procedure is detailed in Chapter 4. The adjusted parameters are listed in Table 5.4. From these values, the critical energy release rate G_c could be determined (see Table 5.5), which we believe better reflect the properties of each presumptive fixed separation path.

5.5 Parametric Study

The critical energy release rate G_c is the most common control parameter in fracture mechanics. However, several papers show that the interfacial strength also affects the delamination process [4, 48, 52, 53]. In this section, the effect of interfacial strength on load-displacement curves is studied by numerical experiments.

Three sets of material properties with different interfacial strength are considered. As described in Chapter 4, some parameters are predetermined from the assumed interfacial strength, while others are calibrated using the adjusted fracture toughness listed in Table 5.5. Apparently, the fitted parameters don't depend on the interfacial strength. Figure 5.7 shows mode I traction–separation for different values of interfacial strength. As the interfacial strength becomes higher, the element becomes stronger (higher maximum traction) but less ductile (larger final separation). The result is as expected since we constrain the critical energy release rates to be the same. Flexural tests for DCB, ENF, and 50% MMB are conducted numerically. The load-displacement curves are shown in Figure 5.8. The maximum loads are close but not the same with varying interfacial strengths. Specifically, the maximum load increase a little with increasing interfacial strength, especially for ENF specimen.

The results indicate the limitation of linear elastic fracture mechanics, in which the cohesive zone size must be much smaller than all characteristic length of the structure so that the stress field at regions other than the small FPZ could be calculated

with the theory of elasticity. On one hand, the cohesive zone size of an adhesive layer exists in reality and is primarily determined by the adhesive properties and deformation. On the other hand, during a DCB, an ENF, or an MMB test, the laminate thickness must be much smaller than the span length to keep geometric nonlinear effects small [2]. There is no guarantee that the cohesive zone size is much smaller than the small laminate thickness. Apart from G_c commonly used in LEFM, we also need the interfacial strength as a interfacial parameter, which is considered in calibration process as you can see in Section 4.

Table 5.7. Interface parameters in three cases.

(a) Predetermined parameters.

	$\tilde{K}\,(\mathrm{N/mm^3})$	$\tau_{\mathrm{I}0}\,(\mathrm{MPa})$	$k\,(\mathrm{J/m^2})$	$\tau_{\mathrm{I}\max}\,(\mathrm{MPa})$	d_c
1	13961.6	56.57	229.2	57.42	0.95
2	27923.2	80.00	229.2	81.2	0.95
3	40209.4	96.00	229.2	97.44	0.95

(b) Fitted values of Q and b.

	$Q\,(\mathrm{J/m^2})$	b
1	393.66	2.15226
2	393.66	2.15226
3	393.66	2.15227

(c) Fitted values of J

	DCB	20% MMB	50% MMB	80% MMB	ENF
1	1.0	0.98557	1.0780	0.90683	0.82618
2	1.0	0.98557	1.0780	0.90683	0.82618
3	1.0	0.98557	1.0780	0.90683	0.82618

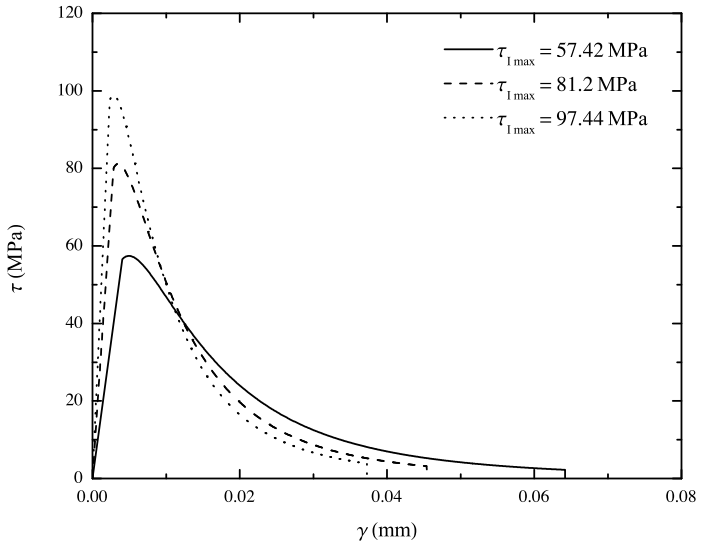
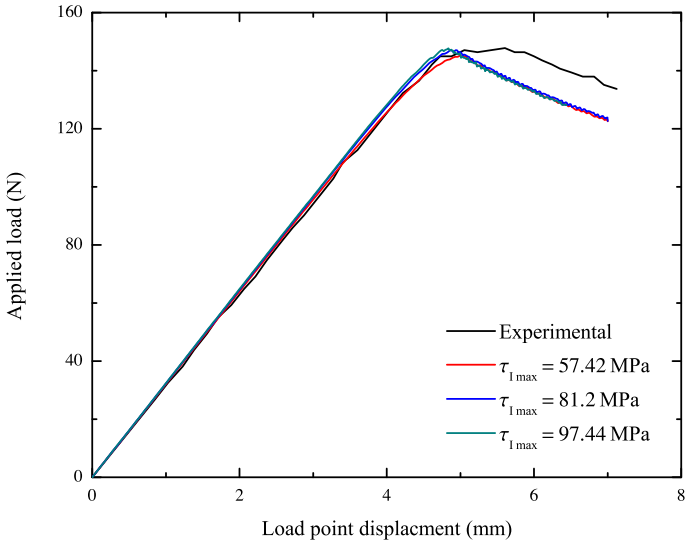
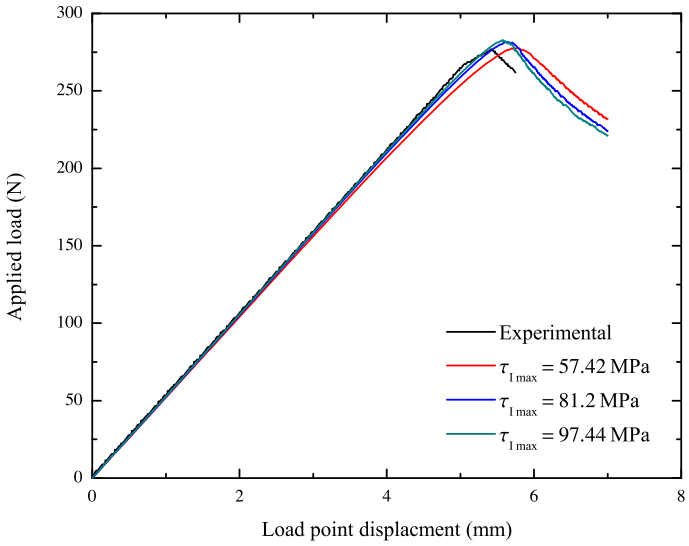


Figure 5.7. Mode I traction–separation curves for different interfacial strengths.



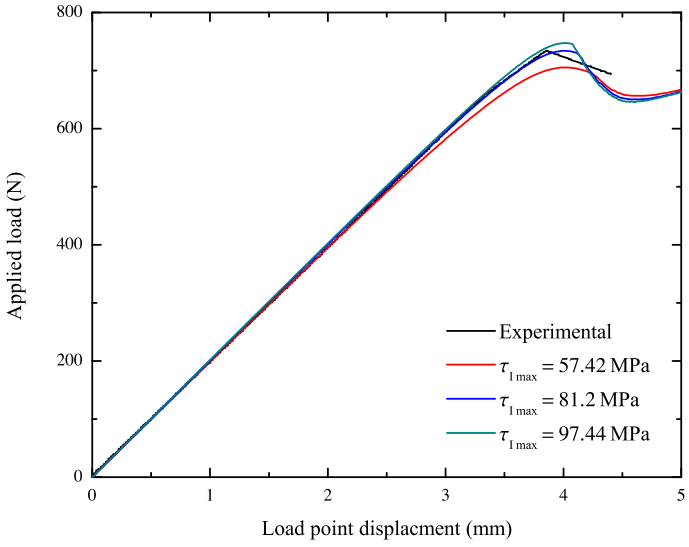
(a) DCB.

Figure 5.8. Load–displacement curves for different interfacial strengths.



(b) 50% MMB.

Figure 5.8. Load–displacement curves for different interfacial strengths (continued).



(c) ENF.

Figure 5.8. Load–displacement curves for different interfacial strengths.

6. CONCLUSIONS

The research presents an implicit scheme for a damage-type CZM with an associated systematic calibration method. The damage-type CZM takes into account the path dependence and is derived with the principle of maximum dissipation. The implicit scheme is constructed based on the Newton–Raphson method. The calibration method is established such that flexural test results could be precisely reproduced. The following findings are discovered:

1. The present CZM is able to yield consistent and reliable results for flexural tests with complex separation path;
2. The systematic calibration method is found to provide reasonable parameters for the CZM to yield accurate results;
3. The implicit scheme is found to be stable enough in complex separation path delamination.

The following conclusions could be drawn from the findings:

1. The present CZM together with the implicit scheme and calibration method could be applicable in real applications;
2. The systematic calibration method could be adjusted to provide parameters for other similar damage models.

REFERENCES

[1] ASTM. ASTM D552813 standard test method for mode I interlaminar fracture toughness of unidirectional fiber-reinforced polymer matrix composites, 2013.

[2] ASTM. ASTM D6671/D6671M13E1 standard test method for mixed mode I–mode II interlaminar fracture toughness of unidirectional fiber reinforced polymer matrix composites, 2013.

[3] ASTM. ASTM D7905/D7905M14 standard test method for determination of the mode II interlaminar fracture toughness of unidirectional fiber-reinforced polymer matrix composites, 2014.

[4] A. Turon, P.P. Camanho, J. Costa, and J. Renart. Accurate simulation of delamination growth under mixed-mode loading using cohesive elements: definition of interlaminar strengths and elastic stiffness. *Composite Structures*, 92(8):1857–1864, 2010.

[5] K. Park and G.H. Paulino. Cohesive zone models: a critical review of traction-separation relationships across fracture surfaces. *Applied Mechanics Reviews*, 64(6):060802, 2011.

[6] F. Bianchi and X. Zhang. A cohesive zone model for predicting delamination suppression in z-pinned laminates. *Composites Science and Technology*, 71(16):1898–1907, 2011.

[7] D.S. Dugdale. Yielding of steel sheets containing slits. *Journal of the Mechanics and Physics of Solids*, 8(2):100–104, 1960.

[8] G.I. Barenblatt. The mathematical theory of equilibrium cracks in brittle fracture. *Advances in Applied Mechanics*, 7:55–129, 1962.

[9] Simulia. Abaqus 6.13 documentation, 2013.

[10] P.P. Camanho and C.G. Dávila. Mixed-mode decohesion finite elements for the simulation of delamination in composite materials. Technical Report 211737, USA: National Aeronautics and Space Administration, 2002.

[11] V. Tvergaard. Effect of fibre debonding in a whisker-reinforced metal. *Materials Science and Engineering: A*, 125(2):203–213, 1990.

[12] X.P. Xu and A. Needleman. Void nucleation by inclusion debonding in a crystal matrix. *Modelling and Simulation in Materials Science and Engineering*, 1(2):111, 1993.

[13] M. Ortiz and A. Pandolfi. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. *International Journal for Numerical Methods in Engineering*, 44(9):1267–1282, 1999.

[14] F.H. Wittmann, K. Rokugo, E. Brühwiler, H. Mihashi, and P. Simonin. Fracture energy and strain softening of concrete as determined by means of compact tension specimens. *Materials and Structures*, 21(1):21–32, 1988.

[15] V. Tvergaard and J.W. Hutchinson. The influence of plasticity on mixed mode interface toughness. *Journal of the Mechanics and Physics of Solids*, 41(6):1119–1135, 1993.

[16] M.J. Van den Bosch, P.J.G. Schreurs, and M.G.D. Geers. An improved description of the exponential xu and needleman cohesive zone law for mixed-mode decohesion. *Engineering Fracture Mechanics*, 73(9):1220–1234, 2006.

[17] A. Corigliano. Formulation, identification and use of interface models in the numerical analysis of composite delamination. *International Journal of Solids and Structures*, 30(20):2779–2811, 1993.

[18] O. Allix and A. Corigliano. Modeling and simulation of crack propagation in mixed–modes interlaminar fracture specimens. *International Journal of Fracture*, 77(2):111–140, 1996.

[19] G. Alfano and M.A. Crisfield. Finite element interface models for the delamination analysis of laminated composites: mechanical and computational issues. *International Journal for Numerical Methods in Engineering*, 50(7):1701–1736, 2001.

[20] L.C.T. Overgaard, E. Lund, and P.P. Camanho. A methodology for the structural analysis of composite wind turbine blades under geometric and material induced instabilities. *Computers & Structures*, 88(19):1092–1109, 2010.

[21] L. Zhang, Z.Y. Gao, and W.B. Yu. A string-based cohesive zone model for interlaminar delamination. *Engineering Fracture Mechanics*, 180:1–22, 2017.

[22] 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. *Mechanics of Materials*, 38(11):1072–1089, 2006.

[23] R. Borg, L. Nilsson, and K. Simonsson. Simulating DCB, ENF and MMB experiments using shell elements and a cohesive zone model. *Composites Science and Technology*, 64(2):269–278, 2004.

[24] C. Fan, P.Y.B. Jar, and J.J.R. Cheng. Cohesive zone with continuum damage properties for simulation of delamination development in fibre composites and failure of adhesive joints. *Engineering Fracture Mechanics*, 75(13):3866–3880, 2008.

[25] S.T. Pinho, L. Iannucci, and P. Robinson. Formulation and implementation of decohesion elements in an explicit finite element code. *Composites Part A: Applied Science and Manufacturing*, 37(5):778–789, 2006.

[26] P.W. Harper and S.R. Hallett. Cohesive zone length in numerical simulations of composite delamination. *Engineering Fracture Mechanics*, 75(16):4774–4792, 2008.

[27] Y.F. Gao and A.F. Bower. A simple technique for avoiding convergence problems in finite element simulations of crack nucleation and growth on cohesive interfaces. *Modelling and Simulation in Materials Science and Engineering*, 12(3):453, 2004.

[28] E.M. Wu and R.C. Reuter Jr. Crack extension in fiberglass reinforced plastics. Technical Report 275, University of Illinois, 1965.

[29] M.L. Benzeggagh and M. Kenane. Measurement of mixed-mode delamination fracture toughness of unidirectional glass/epoxy composites with mixed-mode bending apparatus. *Composites Science and Technology*, 56(4):439–449, 1996.

[30] K. Park, G.H. Paulino, and J.R. Roesler. A unified potential-based cohesive model of mixed-mode fracture. *Journal of the Mechanics and Physics of Solids*, 57(6):891–908, 2009.

[31] A. Airoldi and C.G. Dávila. Identification of material parameters for modelling delamination in the presence of fibre bridging. *Composite Structures*, 94(11):3240–3249, 2012.

[32] J.W. Xie, A.M. Waas, and M. Rassaian. Closed-form solutions for cohesive zone modeling of delamination toughness tests. *International Journal of Solids and Structures*, 88:379–400, 2016.

[33] J.W. Xie, A.M. Waas, and M. Rassaian. Estimating the process zone length of fracture tests used in characterizing composites. *International Journal of Solids and Structures*, 100:111–126, 2016.

[34] W. Xu and A.M. Waas. Multiple solutions in cohesive zone models of fracture. *Engineering Fracture Mechanics*, 177:104–122, 2017.

[35] A.N. Gent and R.P. Petrich. Adhesion of viscoelastic materials to rigid substrates. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 310, pages 433–448. The Royal Society, 1969.

[36] C.Y. Hui, A. Ruina, R. Long, and A. Jagota. Cohesive zone models and fracture. *The Journal of Adhesion*, 87(1):1–52, 2011.

[37] J.C. Simo and T.J.R. Hughes. *Computational Inelasticity*, volume 7 of *Interdisciplinary Applied Mathematics*. Springer, New York, 1998.

[38] G.Z. Voyiadjis and P.I. Kattan. *Advances in Damage Mechanics: Metals and Metal Matrix Composites*. Elsevier Science, 1999.

[39] J. Cordebois and F. Sidoroff. Anisotropic damage in elasticity and plasticity. *Journal de Mécanique Théorique et Appliquée*, pages 45–60, 1982.

[40] J. Betten. Damage tensors in continuum mechanics. *Journal de Mecanique Theorique et Appliquee*, 2(1):13–32, 1983.

[41] A.A. Benzerga and J. Besson. Plastic potentials for anisotropic porous solids. *European Journal of Mechanics, A/Solids*, 20(3):397–434, 2001.

[42] S. Hashemi, A.J. Kinloch, and J.M. Williams. The analysis of interlaminar fracture in uniaxial fibre-polymer composites. volume 427, pages 173–199. The Royal Society, 1990.

[43] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipies in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[44] A.M Elmarakbi, N. Hu, and H. Fukunaga. Finite element simulation of delamination growth in composite materials using ls-dyna. *Composites Science and Technology*, 69(14):2383–2391, 2009.

[45] Z. Gao, L. Zhang, and W. Yu. Simulating the mixed-mode progressive delamination in composite laminates. In *American Society of Composites-30th Technical Conference*, 2015.

[46] C. Balzani and W. Wagner. An interface element for the simulation of delamination in unidirectional fiber-reinforced composite laminates. *Engineering Fracture Mechanics*, 75(9):2597–2615, 2008.

[47] C.G. Dávila, P.P. Camanho, and A. Turon. Effective simulation of delamination in aeronautical structures using shells and cohesive elements. *Journal of Aircraft*, 45(2):663–672, 2008.

[48] D. Xie and A.M. Waas. Discrete cohesive zone model for mixed-mode fracture using finite element analysis. *Engineering Fracture Mechanics*, 73(13):1783–1796, 2006.

[49] J.R. Reeder and J.H. Crews Jr. Mixed-mode bending method for delamination testing. *AIAA Journal*, 28(7):1270–1276, 1990.

[50] Q.D. Yang and B. Cox. Cohesive models for damage evolution in laminated composites. *International Journal of Fracture*, 133(2):107–137, 2005.

[51] A. Turon, C.G. Dávila, P.P. Camanho, and J. Costa. An engineering solution for using coarse meshes in the simulation of delamination with cohesive zone models. Technical Report 213547, USA: National Aeronautics and Space Administration, 2005.

[52] A. Turon, C.G. Davila, P.P. Camanho, and J. Costa. An engineering solution for mesh size effects in the simulation of delamination using cohesive zone models. *Engineering Fracture Mechanics*, 74(10):1665–1682, 2007.

[53] C. Sarrado, A. Turon, J. Renart, and I. Urresti. Assessment of energy dissipation during mixed-mode delamination growth using cohesive zone models. *Composites Part A: Applied Science and Manufacturing*, 43(11):2128–2136, 2012.

[54] R. Borg, L. Nilsson, and K. Simonsson. Modeling of delamination using a discretized cohesive zone and damage formulation. *Composites Science and Technology*, 62(10):1299–1314, 2002.

A. ENERGY RELEASE RATE

Following Borg et al. [54], the energy release rates are defined by

$$G = \int_0^t \boldsymbol{\tau} \cdot \dot{\boldsymbol{\gamma}} \mathrm{d}t - \Psi_e = \int_0^t \boldsymbol{\tau} \cdot \dot{\boldsymbol{\gamma}} \mathrm{d}t - \frac{1}{2}\boldsymbol{\tau} \cdot \boldsymbol{\gamma} \tag{A.1}$$

The energy release rate could be generalized to the energy release vector

$$\boldsymbol{G} = [G_I, G_{II}, G_{III}]^T = \int_0^t \boldsymbol{\tau} \cdot \boldsymbol{\mathcal{I}} \cdot \dot{\boldsymbol{\gamma}} \mathrm{d}t - \frac{1}{2}\boldsymbol{\tau} \cdot \boldsymbol{\mathcal{I}} \cdot \boldsymbol{\gamma}, \tag{A.2}$$

where

$$\boldsymbol{\mathcal{I}} = \sum_{i=1}^{3} \boldsymbol{e}_i \otimes \boldsymbol{e}_i \otimes \boldsymbol{e}_i. \tag{A.3}$$

G could be related to $\boldsymbol{G}$ as

$$G = G_I + G_{II} + G_{III}. \tag{A.4}$$

It's common to assume delamination occurs when the damage variable d reach a specific value. In this CZM, d has a maximum of unity and delamination is assumed to happen when $d = d_c < 1$. d_c is a value very close to unity because d would approach but never reach unity in the generalized CZM. With the delamination criterion, the instant of delamination initiation t_c could be determined. Notice that the elastic part Ψ_e vanishes at complete delamination, and hence the critical energy release rate and critical energy release rate vector are obtained as

$$G_c = \int_0^{t_c} \boldsymbol{\tau} \cdot \dot{\boldsymbol{\gamma}} \mathrm{d}t \quad \text{and} \quad \boldsymbol{G}_\mathrm{c} = \int_0^{t_c} \boldsymbol{\tau} \cdot \boldsymbol{\mathcal{I}} \cdot \dot{\boldsymbol{\gamma}} \mathrm{d}t. \tag{A.5}$$

The critical energy release rate are used to characterize the ability to resist fracture. So it's also called fracture toughness.

Define the mode mix vector

$$\boldsymbol{\beta} = \frac{\boldsymbol{G}}{G}. \tag{A.6}$$

As discussed previously, proportional separation paths are assumed in most CZMs, but non-proportional separation paths are observed even in a simple MMB specimen (see Figure 1.1 for MMB specimen setting, and Sarrado et al. [53] for more details). It is beneficial to find a intuitive form of the mode mix ratio, which is hard to obtain for general case but possible for a proportional separation path. Specifically, set $\dot{\hat{\boldsymbol{\gamma}}} = \mathbf{0}$. Substituting Eq. (2.14) into Eq. (A.1) gives

$$G = \int_0^t K\gamma\dot{\gamma}\mathrm{d}t - \frac{1}{2}K\gamma^2. \tag{A.7}$$

Similarly,

$$\begin{aligned} \boldsymbol{G} &= \int_0^t K\boldsymbol{\gamma}\cdot\boldsymbol{\mathcal{I}}\cdot\dot{\boldsymbol{\gamma}}\mathrm{d}t - \frac{1}{2}K\boldsymbol{\gamma}\cdot\boldsymbol{\mathcal{I}}\cdot\boldsymbol{\gamma} \\ &= \left(\int_0^t K\gamma\dot{\gamma}\mathrm{d}t - \frac{1}{2}K\gamma^2\right)\hat{\boldsymbol{\gamma}}\cdot\boldsymbol{\mathcal{I}}\cdot\hat{\boldsymbol{\gamma}} = G \quad \sum_{i=1}^{3}\left(\gamma_i^2\boldsymbol{e}_i\right). \end{aligned} \tag{A.8}$$

Substituting Eqs. (A.7) and (A.8) into Eq. (A.6) gives

$$\boldsymbol{\beta} = \sum_{i=1}^{3}\gamma_i^2\boldsymbol{e}_i. \tag{A.9}$$

From the above Eq. (A.9), we notice that $\dot{\hat{\boldsymbol{\gamma}}} = \mathbf{0}$ is a sufficient condition of $\dot{\boldsymbol{\beta}} = \mathbf{0}$, i.e. proportional separation path leads to constant mode mix ratio.

B. PATH DEPENDENCE FUNCTION

Following Zhang et al. [21], the Lagrange basis polynomials and normalized Lagrange basis polynomials are introduced as

$$l_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right) = \prod_{j=1;j\neq i}^{n} \frac{\left(\hat{\boldsymbol{\gamma}}^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)\cdot\left(\hat{\boldsymbol{\gamma}}_i^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)}{\left(\hat{\boldsymbol{\gamma}}_i^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)\cdot\left(\hat{\boldsymbol{\gamma}}_i^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)} \equiv \prod_{j=1;j\neq i}^{n} \frac{\left(\hat{\boldsymbol{\gamma}}^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)\cdot\left(\hat{\boldsymbol{\gamma}}_i^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2}\right)}{\hat{\boldsymbol{\gamma}}_i^{\circ 2} - \hat{\boldsymbol{\gamma}}_j^{\circ 2\ 2}} \tag{B.1}$$

$$\hat{l}_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right) = \frac{l_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right)}{\sum_{j=1}^{n} l_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right)} \equiv \frac{l_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right)}{l\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right)} \tag{B.2}$$

where $(\cdot)^{\circ 2}$ denote the Hadamard power of a vector, e.g.

$$\hat{\boldsymbol{\gamma}}^{\circ 2} = \sum_{i=1}^{3} \hat{\gamma}_i^2 \boldsymbol{e}_i. \tag{B.3}$$

Finally, $J\left(\hat{\boldsymbol{\gamma}}\right)$ are obtained through interpolation,

$$J\left(\hat{\boldsymbol{\gamma}}\right) = \sum_{i=1}^{n} \hat{l}_i\left(\hat{\boldsymbol{\gamma}}^{\circ 2}\right) J_i. \tag{B.4}$$

It could be verified that Eq. (B.4) reproduces the value of experimental data $(J_i, \hat{\boldsymbol{\gamma}}_i)$. Generally speaking, material is most likely to posses a specific symmetry, among which transverse isotropy is a rather common one. That means the material exhibits the same property in all direction within a plane. Specifically, J is a function of $\hat{\gamma}_1$ only in many real delamination configurations. Similarly, we introduce Lagrange basis polynomials of one variable

$$l_i\left(\hat{\gamma}_1^2\right) = \prod_{j=1;j\neq i}^{n} \frac{\hat{\gamma}_1^2 - \left(\hat{\gamma}_1\right)_j^2}{\left(\hat{\gamma}_1\right)_i^2 - \left(\hat{\gamma}_1\right)_j^2}, \tag{B.5}$$

and $J\left(\hat{\gamma}_1\right)$ could be obtained through interpolation as

$$J\left(\hat{\gamma}_1\right) = \sum_{i=1}^{n} l_i\left(\hat{\gamma}_1^2\right) J_i. \tag{B.6}$$

Furthermore, the derivative of J is needed:

$$\frac{\partial J}{\partial \hat{\boldsymbol{\gamma}}} = \frac{\partial J}{\partial \hat{\gamma}_1} \boldsymbol{e}_1 = \left(\sum_{i=1}^{n} 2\hat{\gamma}_1 \frac{\partial l_i}{\partial \hat{\gamma}_1^2} J_i \right) \boldsymbol{e}_1, \tag{B.7}$$

where

$$\frac{\partial l_i}{\partial \hat{\gamma}_1^2} = \sum_{j=1; j\neq i}^{n} \left[\prod_{k=1; k\neq i,j}^{n} \left(\frac{\hat{\gamma}_1^2 - (\hat{\gamma}_1)_k^2}{(\hat{\gamma}_1)_i^2 - (\hat{\gamma}_1)_k^2} \right] \frac{1}{(\hat{\gamma}_1)_i^2 - (\hat{\gamma}_1)_j^2}. \tag{B.8}$$

C. UMAT ALGORITHM

Let $(\cdot)_{\text{old}}$ denote the value of a quantity at a given instant of time, t_n, and let $(\cdot)_{\text{new}}$ denote that at $t_{n+1} = t_n + \Delta t$, where $\Delta(\cdot)$ denotes the increment in a quantity over this time interval. When calling UMAT at a cohesive element, Abaqus/Standard passes in $\boldsymbol{\gamma}_{\text{old}}$, and $\Delta\boldsymbol{\gamma}$ and gets back $\boldsymbol{\tau}_{\text{new}}$ and the consistent tangent stiffness matrix. The UMAT algorithm is described as follows (see Figure C.1 for a flowchart of the algorithm):

1. Read $\boldsymbol{\gamma}_{\text{old}}$, and $\Delta\boldsymbol{\gamma}$, which are passed in by Abaqus/Standard, and also read α_{old} and d_{old}, which are saved as state variables for each element;
2. Check whether the loading/unloading conditions (Eq. (2.11)) are met;
3. If yes, first compute α_{new} with Eqs. (2.27) (2.28), then update d_{new} and $\boldsymbol{\tau}_{\text{new}}$ with Eqs.(2.29), finally compute the tangent stiffness matrix with Eq. (2.45);
4. Otherwise, first update $\boldsymbol{\gamma}_{\text{new}}$, then set $d_{\text{new}} = d_{\text{old}}$, and finally compute $\boldsymbol{\tau}_{\text{new}}$ with Eq. (2.29);
5. Save α_{new} and d_{new} for the present element, and return $\boldsymbol{\tau}_{\text{new}}$ and the consistent tangent stiffness matrix to Abaqus/Standard.

Once the delamination criterion is met, UMAT will mark the present element as failed and set its stiffness to zero hereafter. Interested readers can refer to Abaqus manual [9] for more details on Abaqus/Standard and UMAT.

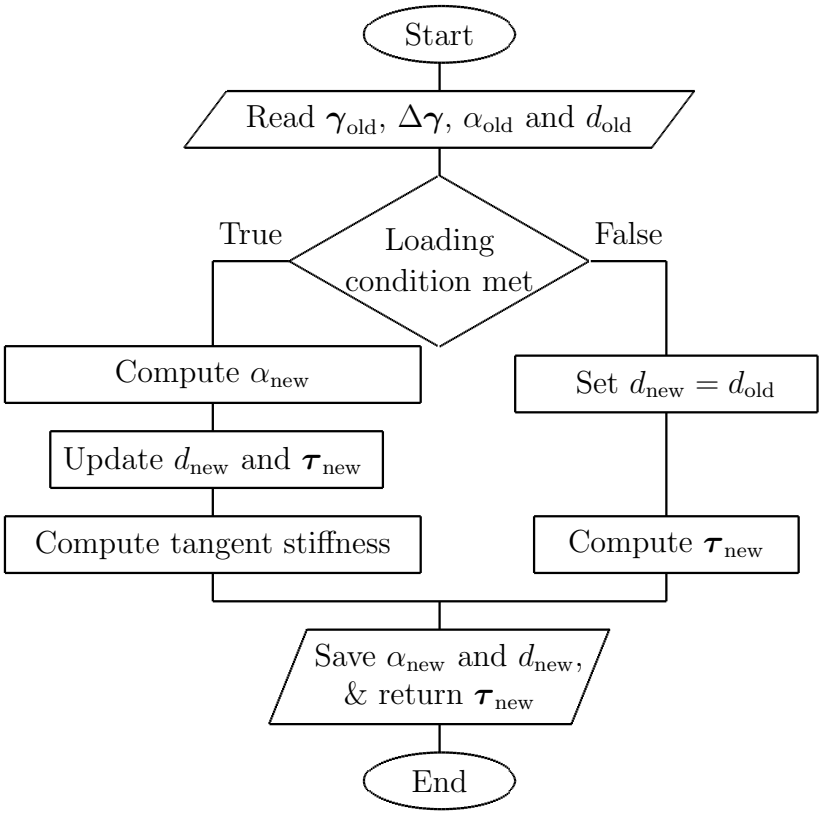


Figure C.1. UMAT algorithm.

The FORTRAN codes of the UMAT is attched:

```
SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,&
  RPL,DDSDDT,DRPLDE,DRPLDT,&
  STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,&
  NDI,NSHR,NTENS,NSTATV,PROPS,NPROPS,COORDS,DROT,PNEWDT,&
  CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,JSTEP,KINC)

  INCLUDE 'ABA_PARAM.INC'

  CHARACTER*80 CMNAME
  DIMENSION STRESS(NTENS),STATEV(NSTATV),&
    DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),&
    STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),&
    PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),&
    JSTEP(4)
  PARAMETER (ONE=1.0D0,TWO=2.0D0,THREE=3.0D0)
  INTEGER::K1,K2,ENDD
  INTEGER,PARAMETER::DBL=SELECTED_REAL_KIND(15)
  INTEGER,PARAMETER::ntest=5
  REAL(DBL),PARAMETER::TOLERANCE=1.0D-15
  !Self defined parameters
  REAL(DBL)::s_gp(NTENS),e_gp(NTENS),hard_F(2)
  REAL(DBL)::scale,s01,G1c,emax,e01,K_const,k0_const,esq
  REAL(DBL)::beta1i(ntest),Ji(ntest),L_gp(NTENS,NTENS)
  REAL(DBL)::pJpe(1,NTENS),J0,I2(NTENS,NTENS),K_compress
  LOGICAL::Compress

  !Initialize
  e_gp=STRAN+DSTRAN
```

```
Compress=.FALSE.
s_gp=0.d0
DDSDDE=0.d0
K_compress=1.d0
ENDD=2+NTENS

!SET UP NECESSARY PARAMETERS
!G1c values and J values subject to change
!depending on the calibration results
scale = SQRT(1.0d0)
s01 = scale*80.0d0
G1c = 1.146d0
emax = TWO*G1c/s01
e01 = 0.1d0*emax
K_const = s01/e01
k0_const = K_const*e01*e01

Q_const = 1.7175515678826703d0*k0_const
b_const = 2.1522633821928245d0

beta1i(1)=1.0d0
beta1i(2)=0.8d0
beta1i(3)=0.5d0
beta1i(4)=0.2d0
beta1i(5)=0.0d0

Ji(1)=1.0d0
Ji(2)=0.9855707482949242d0
Ji(3)=1.078d0
```

```
Ji(4)=0.9330861131488021d0
Ji(5)=0.8261750152328744d0

I2=0.0d0
DO K1=1,NTENS
  I2(K1,K1)=1.0d0
ENDDO
!After failure
hard_F=STATEV(1:2)
IF (hard_F(2) >= 0.95d0) THEN
  IF (e_gp(1) < 0.d0) THEN
    s_gp(1)=K_compress*K_const*e_gp(1)
    DDSDDE(1,1)=K_compress*K_const
    WRITE(*,*) K_compress
  ENDIF
  STRESS=STRESS+s_gp-STATEV(3:5)
  STATEV(3:ENDD)=s_gp
  RETURN
ENDIF
!UPDATE STRESS

esq=DOT_PRODUCT(e_gp,e_gp)
IF ((KINC .EQ. 1) .OR. (esq .LT. TOLERANCE)) THEN
  J0=1.0d0
ELSE
  CALL IntJ(e_gp,J0,pJpe)
ENDIF

IF (DOT_PRODUCT(e_gp,e_gp) .LT. k0_const/J0/K_const) THEN
```

```
  s_gp=K_const*e_gp
  DDSDDE=0.0d0
  DDSDDE=K_const*I2
  IF (e_gp(1) < 0.d0) THEN
    s_gp(1)=K_compress*K_const*e_gp(1)
    DDSDDE(1,1)=DDSDDE(1,1)*K_compress
  ENDIF
  STRESS=STRESS+s_gp-STATEV(3:ENDD)
  STATEV(3:ENDD)=s_gp
  RETURN
END IF

CALL NewtonRaphsonMethod(hard_F)
IF (hard_F(2)>0.955) THEN
  PNEWDT=0.5
  RETURN
ENDIF

IF (e_gp(1) <0.d0) THEN
  s_gp(1)=K_compress*K_const*e_gp(1)
  Compress=.TRUE.
ENDIF
STRESS=STRESS+s_gp-STATEV(3:ENDD)

!UPDATE JACOBIAN
DDSDDE=0.0d0
CALL Ddmg(e_gp,hard_F(2),hard_F(1),L_gp)
DDSDDE=L_gp
```

```
  !UPDATE STATEV
  STATEV(1:2)=hard_F
  STATEV(3:ENDD)=s_gp

CONTAINS

SUBROUTINE IntJ(e_gp,J0,pJpe)
  REAL(DBL),INTENT(IN):: e_gp(:)

  REAL(DBL),INTENT(OUT):: J0
  REAL(DBL),INTENT(OUT):: pJpe(:,:)

  REAL(DBL)::beta(NTENS),sum,beta1,l(ntest),plpb(ntest)
  REAL(DBL)::denominator,numerator
  INTEGER:: i,j,k

  sum=0.0d0
  DO i=1,NTENS
    beta(i)=e_gp(i)*e_gp(i)
    sum=sum+beta(i)
  ENDDO
  beta=beta/sum
  beta1=beta(1)

  DO i=1,ntest
    l(i)=1.0d0
    plpb(i)=1.0d0
    sum=0.0d0
    denominator=1.0d0
```

```
    DO j=1,ntest
      IF (j /= i) THEN
        l(i)=l(i)*(beta1-beta1i(j))/(beta1i(i)-beta1i(j))
        denominator=denominator*(beta1i(i)-beta1i(j))
        numerator=1.0d0
        DO k=1,ntest
          IF (k /= i .and. k /= j) numerator=numerator*(beta1-beta1i(k))
        ENDDO
        sum=sum+numerator
      ENDIF
    ENDDO
    plpb(i)=sum/denominator
  ENDDO

  J0=DOT_PRODUCT(l,Ji)
  pJpe=0.0d0
  pJpe(1,1)=2.0d0*e_gp(1)*DOT_PRODUCT(plpb,Ji)
END SUBROUTINE IntJ

SUBROUTINE Ddmg(e_gp,d_gp,alpha_gp,Ddmg2)

  REAL(DBL),INTENT(IN):: e_gp(:)
  REAL(DBL),INTENT(IN):: d_gp
  REAL(DBL),INTENT(IN):: alpha_gp

  REAL(DBL),INTENT(OUT):: Ddmg2(:,:)

  REAL(DBL):: e_total(1,NTENS),y_total,e_eq2,e_eq,k_gp
  REAL(DBL):: M0,K0,A2(NTENS,NTENS),dAda,S0,pJpe(1,NTENS)
```

```
REAL(DBL):: R1(1,NTENS),pype(1,NTENS)
REAL(DBL):: pypd,E1(1,NTENS),J0

e_total(1,:)=e_gp(1:NTENS)
k_gp = k0_const + Q_const*(EXP(b_const*alpha_gp)-1)

M0=1.0d0-d_gp
K0=M0*M0*K_const

e_eq2=DOT_PRODUCT(e_total(1,:),e_total(1,:))
e_eq=SQRT(e_eq2)
A2=(e_eq2*I2-MATMUL(TRANSPOSE(e_total),e_total))/(e_eq2*e_eq)
CALL IntJ(e_total(1,:),J0,pJpe)

IF (Compress) Then
  e_total(1,1)=0.d0
  e_eq2=DOT_PRODUCT(e_total(1,:),e_total(1,:))
  e_eq=SQRT(e_eq2)
  I2(1,1)=0.d0
ELSE
  I2(1,1)=1.d0
ENDIF

y_total=M0*K_const*e_eq2
E1=0.0d0
Ddmg2=0.d0

IF(J0*y_total-k_gp>=-TOLERANCE*1.0d5 .OR. Compress) THEN
!Stresses are on the yield surface
```

```
    dAda=b_const*Q_const*EXP(b_const*alpha_gp)
    S0=J0*J0/dAda
    R1=J0*y_total*MATMUL(pJpe,A2)/dAda

    pype=2.0d0*M0*K_const*e_total
    pypd=-K_const*e_eq2
    E1=(S0*pype+R1)/(1.0d0-S0*pypd)
  ELSE
    WRITE(*,*) "Warning: NOT ON YIELD SURFACE"
    WRITE(*,*) J0*y_total-k_gp
  ENDIF

  Ddmg2=-2.0d0*M0*K_const*MATMUL(TRANSPOSE(e_total),E1)+K0*I2

  IF (Compress) Ddmg2(1,1)=Ddmg2(1,1)+K_compress*K_const

END SUBROUTINE Ddmg

FUNCTION YieldFunc(d,alpha,J0)
  REAL(DBL)::YieldFunc

  REAL(DBL)::d,alpha,J0,y,A

  A = Q_const*(EXP(b_const*alpha)-1)
  y = (1-d)*K_const*DOT_PRODUCT(e_gp,e_gp)

  YieldFunc = J0*y - k0_const -A
  RETURN
END FUNCTION YieldFunc
```

```
SUBROUTINE NewtonRaphsonMethod(hard_F)
  REAL(DBL),INTENT(INOUT)::hard_F(2)

  REAL(DBL)::pJpe(1,NTENS),J0
  !damage parameter
  REAL(DBL)::k_gp,alpha_gp,d_gp,A,delta_alpha,y,delta_d
  REAL(DBL)::alpha_gp_tmp,d_gp_tmp,f_crit,dAda
  !Variable for line search algorithm
  REAL(DBL)::lam,lam_prev,lam_prev2,yield_new,yield_old
  REAL(DBL)::yield_prev,yield_prev2,d_g,aa,b
  REAL(DBL)::D_f,f_new,f_old,f_prev,f_prev2,diff,counter
  !solution control parameter
  REAL(DBL),PARAMETER::a_crit = 1.0d-4,MAXITER = 20

  !Initialize damage parameter from last step
  alpha_gp_tmp = hard_F(1)
  d_gp_tmp = hard_F(2)

  CALL IntJ(e_gp,J0,pJpe)

  !count number of iterations
  counter = 0
  DO
    A=Q_const*(EXP(b_const*alpha_gp_tmp)-1)
    dAda=b_const*Q_const*EXP(b_const*alpha_gp_tmp)
    y=(1-d_gp_tmp)*K_const*DOT_PRODUCT(e_gp,e_gp)
    f_crit = J0*y-k0_const-A
```

```
IF (ABS(f_crit) .LE. TOLERANCE) EXIT
IF (counter .EQ. MAXITER) THEN
  WRITE(*,*) "Warning: MAXITER achieved,&
   &not converged. f_dmg=",f_crit
  EXIT
ENDIF
counter = counter + 1

delta_d=(J0**2*y-J0*k0_const-J0*A)/&
 &(dAda+J0**2*K_const*DOT_PRODUCT(e_gp,e_gp))
delta_alpha=delta_d/J0

IF (ABS(delta_d) .LE. TOLERANCE) THEN
  WRITE(*,*) "Warning: f_crit accuracy not converged.&
   &But delta_d too small. f_dmg=",f_crit
  EXIT
ENDIF

!Line search method
lam = 1.0d0
yield_old=YieldFunc(d_gp_tmp,alpha_gp_tmp,J0)
D_f = (-J0*K_const*DOT_PRODUCT(e_gp,e_gp) - dAda/J0)*yield_old
DO
  yield_new=YieldFunc(d_gp_tmp+lam*delta_d,&
   &alpha_gp_tmp+lam*delta_alpha,J0)
  f_new = yield_new**2
  f_old = yield_old**2
  diff = f_new-f_old
  IF (ABS(delta_d) .LT. TOLERANCE) THEN
```

```
  diff = (yield_new+yield_old)*&
   &(-K_const*J0*DOT_PRODUCT(e_gp,e_gp)*delta_d-dAda*delta_alpha)
ENDIF
IF (diff .LE. 2.0d0*a_crit*(D_f*lam*delta_d)) EXIT

d_g = D_f*delta_d
IF (lam == 1.0d0) THEN
  lam_prev2 = lam_prev
  lam_prev = lam
  f_prev2 =f_prev
  f_prev = f_new
  lam = -d_g/(2.0d0*(f_new-f_old-d_g))
  IF (lam>0.5*lam_prev) lam = 0.5*lam_prev
  IF (lam<0.1*lam_prev) lam = 0.1*lam_prev
ELSE
  lam_prev2 = lam_prev
  lam_prev = lam
  f_prev2 = f_prev
  f_prev = f_new
  aa = 1/lam_prev**2*(f_prev-d_g*lam_prev-f_old)-1/&
   &lam_prev2**2*(f_prev2-d_g*lam_prev2-f_old)
  b = -lam_prev2/lam_prev**2*(f_prev-d_g*lam_prev-f_old)+&
   &lam_prev/lam_prev2**2*(f_prev2-d_g*lam_prev2-f_old)
  aa = aa/(lam_prev-lam_prev2)
  b = b/(lam_prev-lam_prev2)
  lam = (-b+SQRT(b**2-3*aa*d_g))/(3*aa)
  IF (lam>0.5*lam_prev) lam = 0.5*lam_prev
  IF (lam<0.1*lam_prev) lam = 0.1*lam_prev
ENDIF
```

```
    ENDDO
    !update varibale
    alpha_gp_tmp = alpha_gp_tmp + lam*delta_alpha
    d_gp_tmp = d_gp_tmp + lam*delta_d

  ENDDO
  hard_F(1) = alpha_gp_tmp
  hard_F(2) = d_gp_tmp
  s_gp = K_const*(1-d_gp_tmp)**2*e_gp
  RETURN
END SUBROUTINE NewtonRaphsonMethod

END SUBROUTINE UMAT
```