

Copyright  
by  
Michael Johns Borden  
2012

The Dissertation Committee for Michael Johns Borden  
certifies that this is the approved version of the following dissertation:

**Isogeometric Analysis of Phase-Field Models for Dynamic Brittle  
and Ductile Fracture**

Committee:

---

Thomas J. R. Hughes, Supervisor

---

Omar Ghattas

---

Chad M. Landis

---

Krishnaswamy Ravi-Chandar

---

Mary F. Wheeler

**Isogeometric Analysis of Phase-Field Models for Dynamic Brittle  
and Ductile Fracture**

**by**

**Michael Johns Borden, B.S.; M.S.; M.S.C.A.M.**

**DISSERTATION**

Presented to the Faculty of the Graduate School of  
The University of Texas at Austin  
in Partial Fulfillment  
of the Requirements  
for the Degree of

**DOCTOR OF PHILOSOPHY**

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2012

## **Acknowledgments**

I am thankful for the many people that contributed to the completion of this dissertation. In particular, I would like to thank the following: Professor Thomas J.R. Hughes, my advisor, for providing guidance and encouragement as I explored the topics presented here. Professors Omar Ghattas, Chad M. Landis, K. Ravi-Chandar, and Mary F. Wheeler for their time and service as my committee members. My many colleagues and collaborators for their contributions and the many discussions that have helped shape the ideas presented in this dissertation. And finally, I am most thankful for the patience and support of my family, Amanda, Bryce, and Brandon.

# **Isogeometric Analysis of Phase-Field Models for Dynamic Brittle and Ductile Fracture**

Michael Johns Borden, Ph.D.

The University of Texas at Austin, 2012

Supervisor: Thomas J. R. Hughes

To date, efforts to model fracture and crack propagation have focused on two broad approaches: discrete and continuum damage descriptions. The discrete approach incorporates a discontinuity into the displacement field that must be tracked and updated. Examples of this approach include XFEM, element deletion, and cohesive zone models. The continuum damage, or smeared crack, approach incorporates a damage parameter into the model that controls the strength of the material. An advantage of this approach is that it does not require interface tracking since the damage parameter varies continuously over the domain.

An alternative approach is to use a phase-field to describe crack propagation. In the phase-field approach to modeling fracture the problem is reformulated in terms of a coupled system of partial differential equations. A continuous scalar-valued phase-field is introduced into the model to indicate whether the material is in the unfractured or fractured “phase”. The evolution of the phase-field is governed by a partial differential equation that includes a driving force that is a function of the strain energy of the body in

question. This leads to a coupling between the momentum equation and the phase-field equation. The phase-field model also includes a length scale parameter that controls the width of the smooth approximation to the discrete crack. This allows discrete cracks to be modeled down to any desired length scale. Thus, this approach incorporates the strengths of both the discrete and continuum damage models, i.e., accurate modeling of individual cracks with no interface tracking.

The research presented in this dissertation focuses on developing phase-field models for dynamic fracture. A general formulation in terms of the usual balance laws supplemented by a microforce balance law governing the evolution of the phase-field is derived. From this formulation, small-strain brittle and large-deformation ductile models are then derived. Additionally, a fourth-order theory for the phase-field approximation of the crack path is postulated. Convergence and approximation results are obtained for the proposed theories. In this work, isogeometric analysis, and particularly T-splines, plays an important role by providing a smooth basis that allows local refinement. Several numerical simulations have been performed to evaluate the proposed theories. These results show that phase-field models are a powerful tool for predicting fracture.

# Table of Contents

<b>Acknowledgments</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>List of Tables</b>	<b>xi</b>
<b>List of Figures</b>	<b>xii</b>
<b>Chapter 1. Introduction</b>	<b>1</b>
1.1 Review of computational damage and fracture mechanics . . . . .	2
1.1.1 Discrete fracture models . . . . .	2
1.1.2 Continuum damage models . . . . .	3
1.1.3 Phase-field models . . . . .	5
1.2 Research contributions . . . . .	7
1.3 Organization of this dissertation . . . . .	9
<b>Chapter 2. Isogeometric analysis preliminaries</b>	<b>11</b>
2.1 B-splines and NURBS . . . . .	12
2.1.1 Bernstein polynomials and Bézier curves . . . . .	13
2.1.2 Knot vectors and B-splines . . . . .	13
2.1.3 Knot insertion . . . . .	14
2.1.4 NURBS . . . . .	15
2.2 T-splines . . . . .	16
2.2.1 The T-mesh . . . . .	17
2.2.2 The T-spline basis . . . . .	20
2.2.2.1 Local knot interval vectors . . . . .	20
2.2.2.2 The local basis function domain . . . . .	22
2.2.2.3 T-spline basis functions . . . . .	22
2.2.3 The T-spline element structure . . . . .	23
2.2.3.1 The local basis function mesh . . . . .	23
2.2.3.2 The elemental T-mesh . . . . .	26

2.2.3.3	The IEN array . . . . .	27
2.2.3.4	Restricting the global basis to the parent element domain . . . . .	28
2.2.3.5	The T-spline element geometric map . . . . .	32
2.3	Bézier extraction . . . . .	33
2.3.1	The Bernstein basis . . . . .	35
2.3.2	Computing the element extraction operator for NURBS . . . . .	37
2.3.2.1	Surfaces and solids . . . . .	40
2.3.3	Computing the element extraction operator for T-splines . . . . .	42
2.3.3.1	Comparing extraction for NURBS and T-splines . . . . .	42
2.3.3.2	The extended knot vector . . . . .	43
2.3.3.3	Univariate extraction . . . . .	44
2.3.3.4	Multivariate extraction . . . . .	46
2.3.3.5	A Bézier extraction algorithm for T-splines . . . . .	47
2.3.4	Bézier extraction and the finite element framework . . . . .	53
2.3.4.1	Incorporating $C^e$ into the finite element formulation . . . . .	54
2.3.4.2	The element shape function routine . . . . .	55
<b>Chapter 3. Formulation</b>		<b>58</b>
3.1	Kinematics . . . . .	58
3.2	Phase-field approximations of the fracture surface . . . . .	60
3.2.1	Second-order phase-field theory . . . . .	60
3.2.2	Fourth-order phase-field theory . . . . .	61
3.2.3	Energy approximation . . . . .	62
3.3	Microforce derivation of the phase-field fracture model . . . . .	63
3.4	Small strain brittle fracture . . . . .	66
3.4.1	Second-order formulation . . . . .	69
3.4.1.1	Weak and semidiscrete Galerkin form . . . . .	71
3.4.2	Fourth-order formulation . . . . .	74
3.4.2.1	Weak form . . . . .	75
3.5	Large deformation ductile fracture . . . . .	76
3.5.1	Finite deformation hyperelastic model . . . . .	79
3.5.2	Elastoplastic constitutive model . . . . .	82
3.5.2.1	Return-mapping algorithm . . . . .	84
3.6	Irreversibility . . . . .	88

3.6.1	Strain-history functional . . . . .	89
3.7	Time discretization . . . . .	90
3.7.1	Monolithic generalized- $\alpha$ time discretization . . . . .	90
3.7.2	Staggered time discretization . . . . .	93
3.8	A note on $\Gamma$ -convergence . . . . .	97
<b>Chapter 4. Analysis</b>		<b>99</b>
4.1	Stress degradation function . . . . .	99
4.1.1	Quadratic degradation function . . . . .	100
4.1.2	Cubic degradation function . . . . .	102
4.2	1D analysis . . . . .	104
4.2.1	Non-homogenous solution for the second-order theory . . . . .	104
4.2.2	Convergence study of the phase-field equation . . . . .	107
4.2.3	Surface energy approximation . . . . .	110
4.2.3.1	Second-order theory . . . . .	112
4.2.3.2	Fourth-order theory . . . . .	114
4.2.4	Strain energy approximation . . . . .	116
4.3	Double cantilever beam . . . . .	120
<b>Chapter 5. Numerical results</b>		<b>127</b>
5.1	Quasi-static results . . . . .	128
5.1.1	2D Quasi-static tension load . . . . .	128
5.1.2	2D Quasi-static shear load . . . . .	130
5.2	Two-Dimensional Dynamic Crack Branching . . . . .	132
5.2.1	Reversible vs. Irreversible . . . . .	138
5.2.2	Multiple branches and fragmentation . . . . .	139
5.3	Dynamic shear loading . . . . .	144
5.3.1	Effect of $\sigma_c$ on the material response . . . . .	146
5.3.2	Adaptive refinement scheme . . . . .	152
5.3.3	Analysis-suitable local refinement of T-splines . . . . .	153
5.4	Pressurized cracks . . . . .	158
5.5	Pressurized cylinder with solid elements . . . . .	165
5.6	Ductile fracture of a plate subject to an impulsive load . . . . .	168
5.7	3D block with random nucleation sites . . . . .	176

<b>Chapter 6. Conclusions and future work</b>	<b>179</b>
<b>Appendices</b>	<b>182</b>
<b>Appendix A. Dimensionless form of the phase-field equations</b>	<b>183</b>
A.1 Non-dimensional parameters for branching crack problem. . . . .	184
<b>Appendix B. Numerical Miscellany</b>	<b>186</b>
B.1 Modeling a preexisting crack in a continuous body . . . . .	186
B.2 Higher order derivatives of basis functions . . . . .	187
<b>Bibliography</b>	<b>191</b>

## List of Tables

2.1	The IEN array is constructed using the information from Figure 2.8. The IEN array maps the local basis function number ( $a$ ) and the element number ( $e$ ) to the corresponding global control point ( $A$ ). The local basis function number indexes the T-spline basis functions supported by the element in question. Note that there can be different numbers of T-spline basis functions associated with different elements. . . . .	30
2.2	The action of extraction operators on the Bézier element basis computed from the extended knot vector from Figure 2.15. The full operators would be computed in the case of NURBS, but only the highlighted rows for the T-spline basis function. . . . .	46
4.1	Predicted convergence rates in the $H^s$ norm for the fourth-order problem for piecewise polynomials of degree $p$ . . . . .	109
4.2	Error in the computed surface energy for the one-dimensional problems shown in (4.34) and (4.40). For these results we have used $\ell_0 = 1/8$ . . .	116
4.3	Material properties. . . . .	121
4.4	The data for each variation of the phase-field model studied in this section. For the quadratic degradation function we have used $u_0 = 0.431$ and for the cubic degradation function we have used $u_0 = 0.425$ with $s = 10^{-4}$ . The crack tip is taken as the contour of the phase-field at $c = 0.2$ . . . . .	123
4.5	The error in the critical energy release rate and slope of the displacement vs. crack length curve, $m = \Delta u / \Delta a$ . The reference slope, $m_0 = 1.35 \times 10^{-2}$ , was calculated using standard linear elastic fracture mechanics. .	124
4.6	A comparison of the energy balance $W_{\Delta u} = \tilde{\Psi}_{SE} + \tilde{\Psi}_c$ for each variation of the model. . . . .	126
5.1	The mesh sizes and time steps used for the dynamic crack branching example. The mesh is a uniform, quadratic NURBS in each case. To ensure accurate results, the time step size is chosen to be roughly $h/v_R$ . .	134
5.2	The number basis functions before refinement and the number of elements that were flagged for refinement for each mesh. . . . .	154
5.3	Model parameters for the pressure function (5.11). . . . .	172

## List of Figures

1.1	An example of a simulation result using a discrete fracture model. This simulation was performed using the cohesive zone method. Discrete discontinuities were inserted into the mesh as the cracks advanced. The contours show the intensity of the horizontal stress component. . . . .	3
1.2	An example of a simulation result using a continuum damage model. These results have been produced using a higher-order gradient damage model. The contours indicate the evolution of the damage parameter. Notice the widening of the damage zone as the simulation progresses. . . . .	4
2.1	The domain $\Omega \subset \mathbb{R}^2$ for a bivariate ( $d_p = 2$ ), cubic ( $p = 3$ ) T-spline. The curved boundaries are exact quarter circles. The hash marks on the left indicate homogeneous Dirichlet boundary conditions. . . . .	17
2.2	A T-mesh defining a bicubic T-spline geometry. The large red circles are the T-junctions for this T-mesh. The indexing identifies the T-mesh control points. . . . .	19
2.3	A valid knot interval configuration for the bicubic T-mesh in Figure 2.2. The triangles correspond to a knot interval of 0, the squares correspond to a knot interval of $\frac{1}{2}$ , and the pentagons correspond to a knot interval of 1. A valid knot interval configuration for a T-mesh element is shown in the element callout. Notice that the knot intervals along opposing sides of the element sum to the same value. . . . .	19
2.4	The construction of T-spline basis function $N_{18}$ which is associated with T-mesh vertex $P_{18}$ . Starting at the upper left and going clockwise we have: Inferrence of the local knot vectors from the T-mesh, the resulting local basis function mesh, the local basis function domain, and the T-spline basis function. . . . .	24
2.5	The construction of T-spline basis function $N_{33}$ which is associated with the T-junction T-mesh vertex $P_{33}$ . Starting at the upper left and going clockwise we have: Inferrence of the local knot vectors from the T-mesh, the resulting local basis function mesh, the local basis function domain, and the T-spline basis function. . . . .	25
2.6	The mapping of $T_{33}$ onto the global T-mesh $T$ . The dashed edges correspond to lines of reduced continuity from $T_{33}$ which are not in $T$ . . . . .	27
2.7	The T-spline elements in the elemental T-mesh $T_f$ of $T$ . . . . .	28
2.8	T-mesh elements 1, 10, 11, and 17 in the elemental T-mesh and the T-mesh control points whose corresponding T-spline basis functions are non-zero over these T-spline elements. . . . .	29

2.9	The mappings between parent, element, and basis function coordinate systems for element 17. For simplicity only some of the mappings are shown. $\tilde{\Phi}^{17}$ maps from the parent coordinate system into the element coordinate system. $\hat{\Phi}_a^{17}$ maps from the element coordinate system of local basis function number $a$ into the basis function coordinate system of global control point $A = \text{IEN}(a, 17)$ . The solid dot ( $\bullet$ ) on element 17 shows the rotations incorporated in the mappings. . . . .	31
2.10	A schematic diagram illustrating the central role played by Bézier extraction in unifying CAGD and FEA. . . . .	34
2.11	Bézier extraction for a cubic B-spline curve. The B-spline curve and basis functions are shown on the left. The action of the extraction operator, $C_2$ , for element $\Omega_2$ is illustrated on the right. The transpose of the extraction operator defines the control points, $Q_2 = \{Q_{2,a}\}_{a=1}^4$ , of the Bézier element from the control points, $P_2 = \{P_a\}_{a=2}^5$ of the B-spline curve. The B-spline basis functions, $N_2 = \{N_a\}_{a=2}^5$ , can be computed over the element by applying the extraction operator to Bernstein polynomial basis functions, $B = \{B_a\}_{a=1}^4$ , defined on the Bézier element. Note that the Bernstein basis is the same for each element. This allows the formation of element arrays to be standardized in a shape function routine. . . . .	35
2.12	Bernstein basis functions for polynomial degree $p = 1, 2, 3$ . . . . .	37
2.13	Ordering of the two-dimensional Bernstein polynomials on a bicubic Bézier element. . . . .	37
2.14	All Bézier elements in the support of T-spline basis function $N_{40}$ in the elemental T-mesh, $T_f$ (left) and the local basis function mesh, $T_{40}$ (right). . . . .	43
2.15	A univariate T-spline basis function (solid line) with local knot vector $\Xi = \{0, 1, 2, 3, 4\}$ . The extended knot vector $\bar{\Xi} = \{0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4\}$ adds the additional functions (dashed lines) that will be used to construct the basis functions for the Bézier elements. . . . .	44
2.16	The resulting basis functions of the Bézier elements after knot insertion and refinement of the basis functions shown in Figure 2.15. . . . .	45
2.17	The extraction of a single T-spline basis function. Extraction is decoupled into two univariate extraction operations and then reassembled into a single row of $C^e$ . (The T-spline basis function is the one associated with control point 40, and element $e$ corresponds to element 13 in the elemental T-mesh shown in Figure 2.7.) . . . . .	48

2.18	The result of extracting $N_{40}(\xi_{40})$ over element $e$ , where $e = 13$ . On the top left, the position of element 13 in the local basis function space of $N_{40}(\xi_{40})$ is shown. On the top right, Bézier extraction of $N_{40}$ over element $e$ (see Figure 2.17) generates $c_{14}^{13^T}$ where 14 is the local index of global control point 40 for element 13. In other words, $40 = \text{IEN}(14, 13)$ . Note that this represents a single row of the element extraction operator $\mathbf{C}^{13}$ . On the bottom left, $N_{40}(\xi_{40})$ is plotted. The shaded region is the restriction of $N_{40}(\xi_{40})$ to the domain of element 13. On the bottom right, a close-up of element $e$ is shown which portrays graphically the relationship between the extraction coefficients and Bernstein basis functions. Note that $N_{40}(\xi_{40}) _{13} = c_{14}^{13^T} \mathbf{B}(\hat{\xi})$ . See equation (2.51). . . . .	49
2.19	Integration is performed by Gaussian quadrature on each element. The physical element is pulled back first to the parametric domain through the geometrical mapping and then to the parent element through an affine mapping. For NURBS more efficient quadrature rules may be devised (see Hughes et al. [55]). . . . .	56
3.1	A schematic representation of (a) a solid body $\Omega_t$ with an internal discontinuity boundary $\Gamma_t$ and (b) an approximation of an internal discontinuity by a phase-field $c(x, t)$ . The model parameter $\ell_0$ controls the width of the phase-field approximation. . . . .	59
3.2	The one-dimensional phase-field approximation to the crack surface $\Gamma$ . . . . .	61
3.3	The smooth approximation to the crack surface $\Gamma$ for the fourth-order theory. . . . .	63
4.1	Uniaxial (a) stress-strain and (b) phase-field-strain curves for the homogeneous solution using the quadratic degradation function (4.3). Note that the value of $\ell_0$ influences the maximum tensile stress, see (4.6). . . . .	101
4.2	Uniaxial (a) stress-strain and (b) phase-field-strain curves for the homogeneous solution using the cubic degradation function (4.8). In this case, we have used $s = 10^{-4}$ . For comparison, the dashed lines show the plots for the quadratic degradation function. . . . .	104
4.3	One-dimensional solution of the phase field formulation for various values of the stress ratio $\beta = \frac{\sigma}{\sigma_c} = \frac{\sigma^*}{\sigma_c^*}$ . Note that, except for the limiting case $\beta = 0$ , the phase-field is smooth at the center of the crack ( $x^* = 0$ ). . . . .	107
4.4	Convergence plots of the error in the $L_2$ norm for (a) the second-order phase-field and (b) the fourth-order phase-field. . . . .	110
4.5	Convergence plots of the error in the $H^1$ norm for (a) the second-order phase-field and (b) the fourth-order phase-field. . . . .	111
4.6	Convergence plots of the error in the $H^2$ norm for the fourth-order phase-field. . . . .	112
4.7	Convergence plots for the error in the computed strain energy term. . . . .	119

4.8	The geometry and boundary conditions for the double cantilever beam problem. . . . .	121
5.1	Boundary and load conditions and T-spline Bézier mesh for the single notch in tension benchmark test. The T-spline contains 1370 cubic basis functions and the effective element size is between $h_{min} = 3.906 \times 10^{-3}$ mm and $h_{max} = 1.25 \times 10^{-1}$ mm. . . . .	129
5.2	Crack progression for tension test at several load increments. . . . .	129
5.3	Force-displacement curve for the notched tension benchmark test. T-spline 1 is shown in Figure 5.1b. T-spline 2 is a single global refinement of T-spline 1. The results obtained using the T-spline discretizations are compared to those reported by Miehe, Hofacker, and Welschinger [68] using standard $C^0$ finite elements. . . . .	130
5.4	Input model for the quasi-static shear benchmark test: (a) geometry and boundary conditions and (b) the Bézier element representation of the T-spline. The T-spline contains 5587 cubic basis functions and the effective element size is between $h_{min} = 3.906 \times 10^{-3}$ mm and $h_{max} = 6.25 \times 10^{-2}$ mm. The refinement was performed <i>a priori</i> based on knowledge of the expected crack path. . . . .	131
5.5	Crack progression for the quasi-static shear test at several load increments.	131
5.6	Force-displacement curve for the quasi-static shear test. The results obtained using the T-spline discretization are compared to those reported by Miehe, Hofacker, and Welschinger [68] using standard $C^0$ finite elements. . . . .	132
5.7	The geometry and boundary conditions for the crack branching example. In the simulation the initial crack is modeled by introducing an initial strain history field that induces a phase field at the initial crack location, and the geometry and displacement field are $C^1$ continuous throughout (see Appendix B.1). . . . .	133
5.8	Results for the crack branching example at $t = 80 \mu\text{s}$ . For all three meshes $\ell_0 = 2.5 \times 10^{-4}$ m. . . . .	135
5.9	Plots over time of (a) the elastic strain energy as defined by (5.1) and (b) the dissipated fracture energy as defined by (5.2). . . . .	136
5.10	Plots over time of the crack tip velocity. After branching, the reported crack tip velocity is that of the upper branch. . . . .	137
5.11	A post-processed plot of mesh 2 at $t = 70 \mu\text{s}$ . The displacements have been scaled by a factor of 50 and areas of model where $c < 0.05$ have been removed from the plot. Pressure is measured in Pascals. . . . .	137
5.12	A comparison between the model with irreversibility enforced through the strain energy history functional and reversible behavior allowed for mesh 2. In the reversible case the decrease in the phase-field parameter is much more local to the crack. Qualitatively, however, both results are similar. . . . .	139

5.13	Plots over time of (a) the elastic strain energy and (b) the dissipated fracture energy. . . . .	140
5.14	Plots over time of the crack tip velocity for mesh 2 with and without irreversibility enforced. . . . .	141
5.15	The branching crack problem with $\sigma = 1.95$ MPa. . . . .	142
5.16	The branching crack problem with $\sigma = 2.3$ MPa. . . . .	143
5.17	The geometry and boundary conditions for the dynamic shear loading example. The crack is modeled by an actual discontinuity in the mesh with a zero radius crack tip. The load is applied as a velocity condition that is ramped up from 0 to 16.5 m/s in one microsecond and then held constant for the duration of the simulation. . . . .	145
5.18	Evolution of the crack through time for a uniform $1024 \times 1024$ quadratic NURBS mesh with 1,055,242 control points and $\ell_0 = 1.95 \times 10^{-4}$ m. The resulting crack propagation angle is somewhat greater than $65^\circ$ and close to the experimentally observed angle of about $70^\circ$ reported by Kalthoff and Winkler [58] and Kalthoff [57]. . . . .	147
5.19	A post-processed plot of the dynamic shear loading example at $t = 75 \mu\text{s}$ . The displacements have been scaled by a factor of 5 and areas of model where $c < 0.05$ have been removed from the plot. Pressure is measured in Pascals. . . . .	148
5.20	Crack tip velocity for dynamic shear loading example. . . . .	148
5.21	Kalthoff results at $t = 87 \mu\text{s}$ . Initiation of spurious crack at approximately (a) $t = 37 \mu\text{s}$ , (b) $t = 48 \mu\text{s}$ , and (c) $t = 56 \mu\text{s}$ . $\ell_0^0 = 1.5625 \times 10^{-3}$ m. . . . .	149
5.22	Plots of the (a) mode I opening stress and (b) phase-field values over time at the lower right corner of the dynamic shear model. The cubic case indicates that the cubic degradation function was used. All other cases used the quadratic degradation function. Note that $\ell_0^0 = 1.5625 \times 10^{-3}$ m. . . . .	150
5.23	A comparison of the crack topology for (a) the quadratic degradation function with $\ell_0 = \ell_0^0/8$ on a uniform $1024 \times 1024$ mesh and (b) the cubic degradation function with $\ell_0 = \ell_0^0/4$ on a uniform $512 \times 512$ mesh. . . . .	151
5.24	A schematic representation of an adaptive refinement scheme based on T-splines, analysis-suitable local refinement, and Bézier extraction. . . . .	153
5.25	Kalthoff mesh refinement results. Mesh 1 is a $128 \times 128$ cubic T-spline mesh. Bézier elements were flagged for refinement if $c < 0.8$ at any quadrature point inside the element and $h = \sqrt{a} > 1.94 \times 10^{-4}$ m where $a$ is the element area. . . . .	155
5.26	The first four meshes in the local refinement sequence. The elements in red are those that were selected to be refined at each step. . . . .	156

5.27	The (a) elastic strain energy, $\int_{\Omega} \{(1 - k)c^2 + k\}\psi_e^+ + \psi_e^- \} d\mathbf{x}$ , and (b) dissipated energy, $\int_{\Omega} \mathcal{G}_c [(c - 1)^2/(4\ell_0) + \ell_0 \nabla c ^2] d\mathbf{x}$ , for the sequence of refined meshes shown in Figure 5.25. The reference mesh is the uniformly refined mesh from Figure 5.18. . . . .	157
5.28	Geometry for the pressurized crack example. The geometry is made large enough such that stress waves reflected from the boundary do not reach the propagating crack before the end of the simulation. The dashed lines indicate the extents of the area in which crack propagation will occur before the simulation end time. . . . .	158
5.29	The mesh for the pressurized crack problem. The mesh is constructed as a $C^2$ -continuous cubic T-spline and refined locally in the area where the crack will propagate (the area that appears solid black in the figure). The coarse elements have a mesh size of $h = 0.01$ m, which transitions to a fine mesh size of $h = 1.5265 \times 10^{-4}$ m. This mesh contains 258,879 bi-cubic basis functions. . . . .	159
5.30	The pressure function is varied temporally as shown in (a). To prevent the pressure load from being applied to the newly created crack, the applied pressure load is also limited to a small area around the initial crack as shown in (b). . . . .	160
5.31	Phase-field plots for (a) $p_0 = 4$ MPa and (b) $p_0 = 8$ MPa computed using the second-order phase-field theory. The results for the fourth-order phase-field theory are qualitatively similar and are not shown here. . . . .	162
5.32	Elastic strain energy over time for (a) $p_0 = 4$ MPa and (b) $p_0 = 8$ MPa. The result for the second and fourth-order phase-field theories are compared. . . . .	163
5.33	Crack tip velocity over time for (a) $p_0 = 4$ MPa and (b) $p_0 = 8$ MPa. The result for the second and fourth-order phase-field theories are compared. The fourth-order theory consistently gives higher crack tip velocities. . . . .	164
5.34	Geometry and symmetry conditions for the pressure vessel simulation. The mesh is a three-dimensional thickened T-spline. . . . .	165
5.35	The final mesh for the pressurized cylinder example problem. The volumetric mesh was constructed by thickening a mid-surface mesh. The refinement was performed using the adaptive refinement scheme described in Section 5.3.2 which resulted in a final mesh containing 862,100 basis functions. . . . .	166
5.36	The results of the pressurized cylinder example. The phase-field is shown. . . . .	167
5.37	A post-processed plot of the pressure vessel example at $t = 1.76 \times 10^{-3}$ s. The displacements have been scaled by a factor of 5 and areas of model where $c < 0.05$ have been removed from the plot. Displacement is measured in meters. . . . .	168
5.38	Cross section views showing the three-dimensional phase-field profiles of the crack surfaces. . . . .	169

5.39	Experimental results showing several crack patterns for a plate subjected to an impulsive load (pictures courtesy of Roshdy Barsoum). . . . .	170
5.40	The (a) geometry and boundary conditions and (b) NURBS patches for the impulsively loaded plate example. The boundary conditions were modeled as both sliding ( $u_z = 0$ ) and clamped ( $u_x = u_y = u_z$ ). The dotted line in (b) shows the edge of the displacement boundary condition.	171
5.41	The distribution of the impulse load as a function of (a) time and (b) radius, $r$ , measured from the center of the plate. . . . .	172
5.42	The crack patterns for the sliding and clamped boundary conditions. The patterns depend on both the initial state and boundary conditions. .	174
5.43	The equivalent plastic strain (EPS) and displacement profiles for the sliding boundary condition. Areas of the model where $c < 0.15$ have been removed from the displacement plot. . . . .	175
5.44	The equivalent plastic strain (EPS) and displacement profiles for the clamped boundary conditions. Areas of the model where $c < 0.15$ have been removed from the displacement plot. . . . .	175
5.45	Three-dimensional crack propagation over time. The initial geometry (a) is seeded with randomly distributed nucleation sites. The progression of the crack from (b) to (f) shows the ability of the phase-field model to capture branching, turning, and merging of cracks in three dimensions. . . . .	178

# **Chapter 1**

## **Introduction**

The prevention of fracture-induced failure is a major constraint in engineering designs, and the numerical simulation of fracture processes often plays a key role in design decisions. As a consequence, a wide variety of fracture models have been proposed. A key feature of many of these models is a fracture energy term that relates energy dissipation to the formation of new crack surfaces. A particularly successful example is Griffith's theory for brittle fracture. In this theory, the potential energy of a body is dependent on the sum of the elastic strain energy and the surface energy of the newly formed cracks. When a critical strain energy value is reached the crack extends.

Numerical methods play a crucial role in fracture analyses. Due to the complexity of fracture processes in engineering applications, however, there are significant challenges to accurately predicting crack behavior. One of the difficulties that numerical methods have with fracture models is the need to track discontinuous displacements along unknown crack paths. In this dissertation we study a method that removes this problem by modeling the crack as a continuous transition from uncracked to fully-cracked material. Models of this type are commonly called phase-field models for fracture.

## 1.1 Review of computational damage and fracture mechanics

### 1.1.1 Discrete fracture models

To date, efforts to model fracture and crack propagation have focused on two broad approaches: discrete and continuum damage descriptions. The discrete description of fracture incorporates a discontinuity into the displacement field. Examples of this approach include element deletion, the extended finite element method (XFEM) introduced by Moës et al. [71], the cohesive zone (or interelement crack) methods introduced by Xu and Needleman [107] and Camacho and Ortiz [31], and the cohesive segments method introduced by Remmers et al. [84]. All of these approaches represent cracks as discrete discontinuities, either by inserting discontinuities by means of remeshing strategies, or by enriching the displacement field with discontinuities using the partition of unity method of Babuška and Melenk [8]. These approaches have been studied in depth and it has been shown that they successfully model crack propagation behavior in two dimensions. An example is shown in Figure 1.1. The simulation shown here is a result of the cohesive zone method (see Verhoosel et al. [99] for details). In this method, the displacement discontinuities are explicitly introduced into the mesh, i.e., the mesh is split as the cracks advance.

One disadvantage of discrete fracture models is the need to track the evolution of the discontinuities during simulations. This requires algorithmic checks of the current and predicted crack path that must resolve crack intersections and bifurcations. In three dimensions this has proven to be difficult and, to date, no completely satisfactory solution has been found.

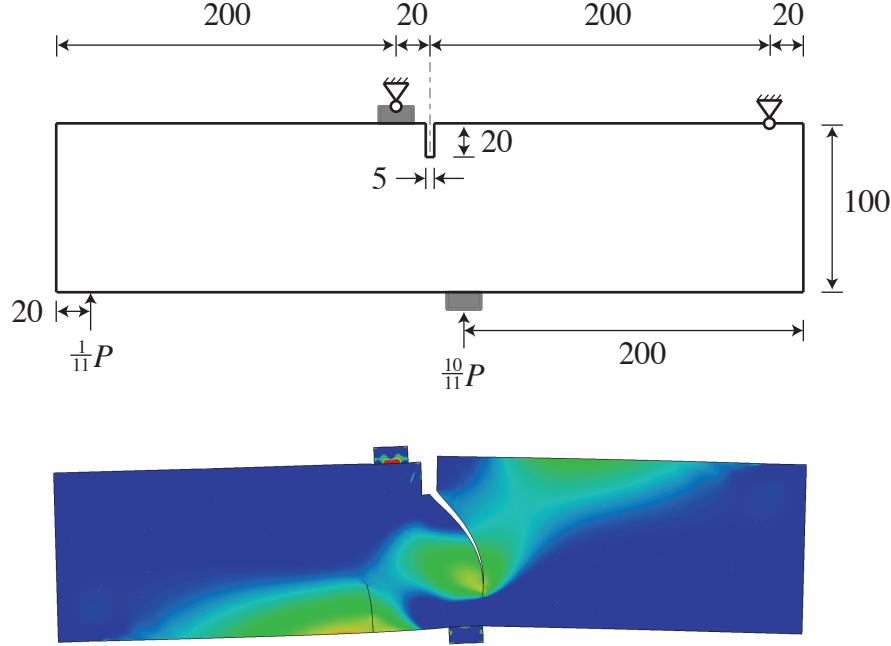


Figure 1.1: An example of a simulation result using a discrete fracture model. This simulation was performed using the cohesive zone method. Discrete discontinuities were inserted into the mesh as the cracks advanced. The contours show the intensity of the horizontal stress component.

### 1.1.2 Continuum damage models

The continuum damage, or smeared crack approach, incorporates a damage parameter into the model that controls the strength of the material (see de Borst [38] for a detailed overview). These models are widely used to simulate diffuse fracture processes. To overcome mesh dependency problems, non-local gradient enhanced theories have been developed which introduce a length scale into the model. For example, Peerlings et al. [75] introduce a gradient formulation for quasi-brittle fracture. In these theories a non-local strain measure is an additional unknown that depends on a local strain measure. The addition of this unknown results in a system of two partial differential equation that govern the response of the model. An example is shown in Figure 1.2. These results

have been produced using a higher-order gradient damage model (see Verhoosel et al. [101] for details).

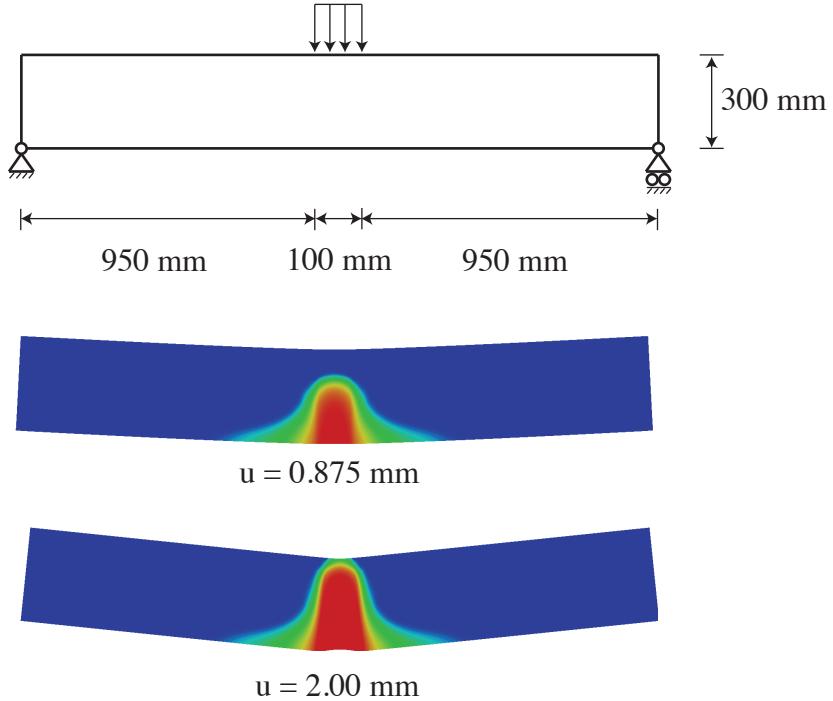


Figure 1.2: An example of a simulation result using a continuum damage model. These results have been produced using a higher-order gradient damage model. The contours indicate the evolution of the damage parameter. Notice the widening of the damage zone as the simulation progresses.

One drawback to these models is that the damage zone tends to widen in a direction normal to the crack direction as the simulation proceeds (see Figure 1.2). This behavior limits the effectiveness of continuum damage models when modeling large dominant cracks. Geers et al. [45] introduced a formulation to address this problem in which the length scale parameter is not constant but is dependent on the strain state. de Borst et al. [39] has proposed the use of cohesive segments as a way to unify both smeared and discrete cracks and has demonstrated that this approach can describe the

transition from distributed micro-cracking to a dominant crack in concrete models. This later approach, however, requires additional checks on the criteria for inserting the cohesive segments into the model.

### 1.1.3 Phase-field models

In recent years, alternative methods for the numerical simulation of fracture have emerged. In these approaches, discontinuities are not introduced into the displacement field or geometry. Instead, the fracture surface is approximated by a phase-field. In the phase-field approach the problem is reformulated in terms of a coupled system of PDEs. A continuous scalar-valued phase-field is introduced into the model to indicate whether the material is in the unfractured or fractured “phase”. This phase-field parameter plays a similar role to the damage parameter in continuum damage models in that it is used to reduce material strength in the area of the crack. The evolution of the phase-field is governed by a PDE that includes a driving force that is a function of the strain energy of the body in question. This leads to a coupling between the momentum equation and the phase-field equation. The phase-field model also includes a length scale parameter that controls the width of the continuous approximation to the discrete crack. This allows discrete cracks to be modeled down to any desired length scale. Thus, this approach incorporates the strengths of both the discrete and continuum damage models, i.e., accurate modeling of individual cracks with no interface tracking.

A phase-field model for quasi-static brittle fracture emanated from the work of Francfort and co-workers on the variational formulation for Griffith-type fracture models. The genesis of this approach is due to Francfort and Marigo [43], and that of the phase-field implementation to Bourdin, Francfort, and Marigo [25] (see Bourdin, Francfort, and Marigo [26] for a comprehensive overview). The variational formula-

tion for quasi-static brittle fracture leads to an energy functional that closely resembles the potential presented by Mumford and Shah [72], which is encountered in image segmentation. A phase-field approximation of the Mumford-Shah potential, based on the theory of  $\Gamma$ -convergence, was presented by Ambrosio and Tortorelli [3]. As discussed by Bourdin, Francfort, and Marigo [26], this approximation was adopted to facilitate the numerical solution of their variational formulation. Recently, this model has been applied in a dynamic setting by Bourdin, Larsen, and Richardson [27]; Larsen, Ortner, and Süli [64]; and Larsen [63], but application to structures of engineering interest has not been considered.

An alternative quasi-static formulation of this phase-field approximation has been presented in the recent work of Miehe, Hofacker, and Welschinger [68] and Miehe, Welschinger, and Hofacker [69]. In this formulation, the phase-field approximation follows from continuum mechanics and thermodynamic arguments. Besides the provision of an alternative derivation, Miehe, Hofacker, and Welschinger [68] also added various features to the model that are key to its application to engineering structures.

Independent of the phase-field formulation based on Griffith's theory, dynamic phase-field fracture models have been developed based on Landau-Ginzburg type phase-field evolution equations, e.g., Karma et al. [59]. The link between these theories and Griffith's theory, however, is not clear. In this dissertation, we favor the phase-field formulation of the Bourdin-type since the physical properties of Griffith's theory are well understood and have proven useful in engineering applications.

## 1.2 Research contributions

In this dissertation, phase-field models for dynamic fracture are developed. The starting point of this work is the quasi-static model for brittle fracture described by Bourdin et al. [26] and Miehe et al. [68]. The model is generalized to both brittle and ductile dynamic fracture. Additionally, a fourth-order phase-field approximation that improves the accuracy of the model is introduced. The models are studied in the context of isogeometric analysis, which provided a framework for conveniently computing with smooth basis functions on exact geometric representations. As part of this work, the following contributions have been made.

1. The formulation of the phase-field model has been generalized by supplementing the usual balance laws with an additional microforce balance law that governs the evolution of the phase-field. This general formulation provides a thermodynamically stable setting from which a new fourth-order phase-field approximation and ductile fracture model have been derived. Additionally, a cubic stress degradation function has been introduced into the formulation. With this new function, the stress-strain relationship exhibits a much more linear behavior prior to crack formation than with previous stress degradation functions.
2. The development of numerical algorithms for the efficient solution of the coupled momentum/phase-field problem. This includes two algorithms for the temporal discretization. These algorithms are based on the generalized- $\alpha$  method and provide the option to solve the equations as a monolithic or staggered problem. An adaptive  $h$ -refinement scheme for the spatial discretization has also been developed.

3. Development and implementation of a general isogeometric analysis code based on Bézier extraction. This code is built on Trilinos (a library developed at Sandia National Laboratories the provides data structures and algorithms for the solution of large systems) and MPI and is capable of handling the mesh resolution required to accurately model crack propagation.
4. The computation of several numerical results that demonstrate the behavior of the phase-field model. These results include both brittle and ductile fracture problems. Large, three-dimensional simulations relevant to real-world engineering questions have been completed. The potential application of phase-field models to pressurized crack problems (e.g., hydraulic fracture) has also been demonstrated. And finally, we have demonstrated the use of T-splines—a recently developed geometric modeling technology—in large, three-dimensional simulations.

These contributions represent the confluence of many disciplines. For example, isogeometric analysis is the unification of Computer Aided Geometric Design technologies (Computer Science) and the finite element method (Mathematics, Mechanics, Engineering, Computer Science). Phase-field models have a strong history in the simulation of the chemical processes associated with binary alloys (Chemistry, Material Science), and the phase-field models studied in this dissertation also have roots in image segmentation (Computer Science). Convergence properties for these models have been established by others via  $\Gamma$ -convergence (Mathematics). The models for the physical process of fracture are based on well known energetic crack growth criteria (Fracture Mechanics). Finally, accurate spatial and temporal discretization techniques are used to compute numerical solutions (Numerical Analysis).

The phase-field models that are developed in this dissertation are based on energetic approaches to predicting fracture. The behavior of crack growth, including nucleation, bifurcation, and crack tip speed and direction changes, is completely determined by the model without the need for additional ad hoc rules. Because crack behavior does not depend on ad hoc rules, the implementation of these models is much simpler and robust than is the case for alternative approaches. This point is particularly important for three-dimensional problems where crack topologies can become very complex. The three-dimensional capabilities of phase-field models provide a powerful tool for predicting fracture.

### 1.3 Organization of this dissertation

Chapter 2 presents aspects of isogeometric analysis that are important for developments presented in later chapters. Non-Uniform Rational B-Splines (NURBS) and T-splines are described and the important development of Bézier extraction is presented. Chapter 3 presents the formulation of several variations of the phase-field model. A general microforce derivation is presented. This is followed by the formulation of small-strain brittle and large-deformation ductile fracture models. The new fourth-order phase-field approximation is introduced. The monolithic and staggered time discretization algorithms are described. Chapter 4 presents a detailed analysis of the phase-field model. The cubic stress degradation function is introduced and an analysis of the one-dimensional solution is given. The remainder of the chapter explores the properties of the model and focuses particularly on the fourth-order approximation. Chapter 5 presents several results for numerical simulations that focus on problems that exhibit complex crack behavior, e.g., bifurcations and direction changes. The adaptive

*h*-refinement scheme is presented. A method for modeling pressurized cracks is introduced. Three-dimensional volumetric problems are solved. Chapter 6 discussions conclusions and outlines directions for future research.

## Chapter 2

### Isogeometric analysis preliminaries

Isogeometric analysis was introduced by Hughes et al. [53] as a generalization of finite elements and has been described in detail by Cottrell et al. [35]. The basic idea of the isogeometric concept is to use the same basis for analysis as is used to describe the geometric domain of the model. The use of non-uniform rational B-splines (NURBS) and T-splines [90], which are a generalization of NURBS, in analysis provides the ability to conveniently compute with a smooth basis. This is in contrast to the basis typically used for standard finite element analysis, which is continuous but not smooth. Furthermore, in the case of T-splines, the local density and smoothness of the basis can be controlled by inserting T-junctions, which are a generalization of hanging nodes in standard finite elements. Also, when using T-splines, efficient mesh refinement strategies exist that preserve the exact CAD geometry. These properties have allowed for innovative approaches to solving existing problems.

The use of isogeometric analysis has been studied in many areas including turbulence [2, 9, 11, 15], fluid-structure interaction [10, 12, 13, 108], incompressibility [6, 7, 42], structural analysis and dynamics [36, 37, 54], shells [18, 60], phase-field analysis [48, 49], large deformation with mesh distortion [66], shape optimization [73, 74, 79, 102], and electromagnetics [29]. The smoothness of the basis also provides a more accurate representation of stresses, which has been observed to yield efficient spatial discretizations for discrete (see Verhoosel et al. [99]) and smeared (see Verhoosel

et al. [100]) fracture models. Another study by Benson et al. [21] has shown that the accuracy of the spatial discretization of NURBS allows for more efficient time integration for explicit dynamics in the context of large deformation shells. There have also been efforts to study geometric considerations relating to isogeometric analysis such as understanding geometry requirements for analysis [34, 61, 65] and the construction of surface and volumetric parameterizations [1, 67, 103, 104, 105, 109].

In this chapter we provide an overview of NURBS and T-spline basis functions most commonly used in isogeometric analysis. We also describe a computational framework called Bézier extraction that provides the canonical element representation for these basis functions.

## 2.1 B-splines and NURBS

To establish definitions for later development, we provide a brief overview of the construction of B-splines and NURBS. For a more detailed description see Piegl and Tiller [78] and Rogers [86]. We index element (i.e., local) basis functions and control points with lower case indices  $a, b, c, \dots$  and global (i.e., patch) basis functions and control points with upper case indices  $A, B, C, \dots$ . The indices  $i, j, k$ , are used for various things, the context making use clear. The global cases considered here consist of a single patch. However, the generalization to the multi-patch case is straightforward. It just involves a transformation between the control point indices of each patch and the corresponding global control points.

### 2.1.1 Bernstein polynomials and Bézier curves

A degree  $p$  Bézier curve is defined by a linear combination of  $p + 1$  Bernstein polynomial basis functions. We define the set of basis functions as  $\mathbf{B}(\xi) = \{B_{a,p}(\xi)\}_{a=1}^{p+1}$  where  $\xi$  is a parametric coordinate defined below. The corresponding set of vector valued control points are  $\mathbf{P} = \{\mathbf{P}_a\}_{a=1}^{p+1}$  where each  $\mathbf{P}_a \in \mathbb{R}^d$ ,  $d$  being the number of spatial dimensions, and  $\mathbf{P}$  is a matrix of dimension  $n \times d$ , viz.,

$$\mathbf{P} = \begin{bmatrix} P_1^1 & P_1^2 & \dots & P_1^d \\ P_2^1 & P_2^2 & \dots & P_2^d \\ \vdots & \vdots & & \vdots \\ P_n^1 & P_n^2 & \dots & P_n^d \end{bmatrix}. \quad (2.1)$$

The Bézier curve can then be written as

$$C(\xi) = \sum_{a=1}^{p+1} \mathbf{P}_a B_{a,p}(\xi) = \mathbf{P}^T \mathbf{B}(\xi) \quad \xi \in [0, 1]. \quad (2.2)$$

The Bernstein polynomials can be defined recursively for  $\xi \in [0, 1]$  as

$$B_{a,p}(\xi) = (1 - \xi)B_{a,p-1}(\xi) + \xi B_{a-1,p-1}(\xi) \quad (2.3)$$

where

$$B_{1,0}(\xi) \equiv 1 \quad (2.4)$$

and

$$B_{a,p}(\xi) \equiv 0 \quad \text{if } a < 1 \text{ or } a > p + 1. \quad (2.5)$$

### 2.1.2 Knot vectors and B-splines

A univariate B-spline basis is defined by a knot vector. The knot vector is a set of non-decreasing parametric coordinates written as  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$  where

$\xi_A \in \mathbb{R}$  is the  $A^{th}$  knot,  $p$  is the polynomial degree of the B-spline basis functions, and  $n$  is the number of basis functions. B-spline basis functions for a given degree,  $p$ , are defined recursively over the parametric domain by the knot vector. Piecewise constants are first defined as

$$N_{A,0}(\xi) = \begin{cases} 1 & \xi_A \leq \xi < \xi_{A+1} \\ 0 & \text{otherwise.} \end{cases} \quad (2.6)$$

For  $p > 0$  the basis functions are define by the Cox-de Boor recursion formula

$$N_{A,p}(\xi) = \frac{\xi - \xi_A}{\xi_{A+p} - \xi_A} N_{A,p-1}(\xi) + \frac{\xi_{A+p+1} - \xi}{\xi_{A+p+1} - \xi_{A+1}} N_{A+1,p-1}(\xi). \quad (2.7)$$

A B-spline *curve* of degree  $p$  in  $\mathbb{R}^d$  is defined by a set of B-spline basis functions,  $\mathbf{N}(\xi) = \{N_{A,p}(\xi)\}_{A=1}^n$ , and control points,  $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^n$ , as

$$T(\xi) = \sum_{A=1}^n \mathbf{P}_A N_{A,p}(\xi) = \mathbf{P}^T \mathbf{N}(\xi). \quad (2.8)$$

### 2.1.3 Knot insertion

Knots may be inserted into a knot vector without changing the geometric or parametric properties of the curve. Let  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$  be a given knot vector. Inserting a new knot  $\bar{\xi} \in [\xi_k, \xi_{k+1}]$  with  $k > p$  into the knot vector requires that  $n + 1$  new basis functions be defined using (2.6) and (2.7) with the new knot vector  $\bar{\Xi} = \{\xi_1, \xi_2, \dots, \xi_k, \bar{\xi}, \xi_{k+1}, \dots, \xi_{n+p+1}\}$ . The  $m = n + 1$  new control points,  $\{\bar{\mathbf{P}}_A\}_{A=1}^m$ , are formed from the original control points,  $\{\mathbf{P}_A\}_{A=1}^n$ , by

$$\bar{\mathbf{P}}_A = \begin{cases} \mathbf{P}_1 & A = 1 \\ \alpha_A \mathbf{P}_A + (1 - \alpha_A) \mathbf{P}_{A-1} & 1 < A < m \\ \mathbf{P}_n & A = m \end{cases} \quad (2.9)$$

where

$$\alpha_A = \begin{cases} 1 & 1 \leq A \leq k-p \\ \frac{\bar{\xi} - \xi_A}{\xi_{A+p} - \xi_A} & k-p+1 \leq A \leq k \\ 0 & A \geq k+1. \end{cases} \quad (2.10)$$

Knot values may be inserted multiple times but the continuity of the basis is reduced by one for each repetition of a given knot value. However, by choosing control variables as in (2.9) and (2.10) the continuity of the *curve* is preserved.

#### 2.1.4 NURBS

A NURBS (Non-Uniform Rational B-Spline) is defined by a knot vector  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$ , a set of rational basis functions  $\mathbf{R} = \{R_{A,p}\}_{A=1}^n$ , and a set of control points  $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^n$  as

$$T(\xi) = \sum_{A=1}^n \mathbf{P}_A R_{A,p}(\xi). \quad (2.11)$$

The NURBS basis functions are defined as

$$R_{A,p}(\xi) = \frac{w_A N_{A,p}(\xi)}{W(\xi)} \quad (2.12)$$

where

$$W(\xi) = \sum_{B=1}^n w_B N_{B,p}(\xi) \quad (2.13)$$

is the weight function and  $w_B$  is the weight corresponding to the  $B^{th}$  basis function.

For more efficient computation, a rational curve in  $\mathbb{R}^n$  can be represented by a polynomial curve in  $\mathbb{R}^{n+1}$ . The higher dimensional space is referred to as the projective space. As an example, if  $\mathbf{P}_A$  is a control point of a NURBS curve then the corresponding homogeneous control point in projective space is  $\tilde{\mathbf{P}}_A = \{w_A \mathbf{P}_A, w_A\}^T$ . Thus, given a

NURBS curve defined in  $\mathbb{R}^n$  by (2.11) the corresponding B-spline curve defined in  $\mathbb{R}^{n+1}$  is

$$T(\xi) = \sum_{A=1}^n \tilde{\mathbf{P}}_A N_{A,p}(\xi). \quad (2.14)$$

Working in the projective coordinate system allows the algorithms which operate on B-splines to be applied to NURBS. Once new control variables are computed for the B-spline in the projective coordinate system, simply dividing through by the weight yields the control variables for the NURBS.

## 2.2 T-splines

T-splines, which emanate from computer aided geometrid design (CAGD), overcome the tensor product restriction inherent in NURBS (see Sederberg et al. [90]). In fact, NURBS are a restricted subset of T-splines. Additionally, T-splines can be locally refined (see Sederberg et al. [92]) and can generate analysis-suitable models of arbitrary topological complexity (see Sederberg et al. [93]). This makes T-splines an ideal basis for isogeometric analysis. The extension of the isogeometric framework to the more general T-spline setting was initiated by Bazilevs et al. [14] and Dörfel et al. [41]. T-spline discretizations have been successfully applied to fracture and damage by Verhoosel et al. [99] and Verhoosel et al. [100]. In these applications, efficient local refinement plays an important role. Initial investigations extending T-spline-based isogeometric analysis to the arbitrary topology setting in the context of shells by Benson et al. [20] have also been promising.

T-splines are used for several of the numerical results that are presented in Chapter 5. The fundamental structure of T-splines in terms of isogeometric analysis was presented by Scott et al. [88]. In this section we provide an overview of T-splines (for a

full discussion of the use of T-splines in analysis see Scott [87]). We will illustrate these concepts using the physical domain  $\Omega \subset \mathbb{R}^2$  shown as the shaded area in Figure 2.1. Throughout this section we use  $p$  to indicate polynomial degree,  $d_p$  to indicate the number of parametric dimensions, and  $d_s$  to indicate the number of spatial dimensions.

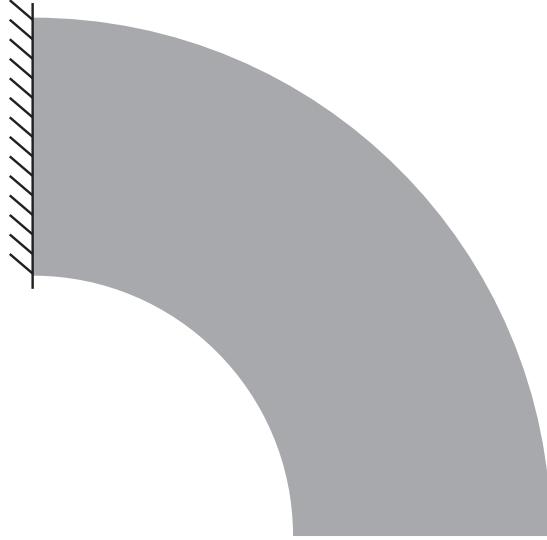


Figure 2.1: The domain  $\Omega \subset \mathbb{R}^2$  for a bivariate ( $d_p = 2$ ), cubic ( $p = 3$ ) T-spline. The curved boundaries are exact quarter circles. The hash marks on the left indicate homogeneous Dirichlet boundary conditions.

### 2.2.1 The T-mesh

A T-spline is constructed from a T-mesh, denoted by  $\mathbf{T}$ . For surfaces, i.e.  $d_p = 2$ , the T-mesh is a mesh of quadrilateral elements<sup>1</sup> which permit T-junctions. In finite element terminology, a T-junction is analogous to a “hanging node.” An element with T-junctions is composed of four corner vertices and any number of additional vertices on any side. The spatial representation of a T-mesh is called a T-spline control mesh

---

<sup>1</sup>What we refer to as T-mesh “elements” are usually referred to as T-mesh “faces” in the CAGD literature [90].

(here we use T-mesh and T-spline control mesh interchangeably). Every vertex in the T-mesh is assigned a control point,  $\mathbf{P}_A \in \mathbb{R}^{d_s}$ , and control weight,  $w_A \in \mathbb{R}$ , where the index  $A$  is used to denote a global control point number.

A T-mesh for the domain  $\Omega$  in Figure 2.1 is shown in Figure 2.2. The black and red circles are T-mesh vertices or, equivalently, control points (see Appendix A of Scott et al. [88] for values of the control point coordinates and weights). The T-junctions in Figure 2.2 are the large red circles  $\mathbf{P}_{25}$  and  $\mathbf{P}_{33}$ .

**Remark 2.1.** This simple geometry could be represented more concisely with NURBS or T-splines. In the case of NURBS, as few as six control points are capable of representing the exact geometry (the tensor product of quadratic and linear basis functions), and for bicubic T-splines, as few as 16 are required. The additional control points in the T-mesh of Figure 2.2 are representative of the fact that finite element analysis will typically require many more degrees of freedom than geometric design.

A T-mesh does not contain enough information to define a T-spline basis. A valid knot interval configuration must also be assigned. Knot intervals (see Sederberg et al. [91]) provide a way to assign local parametric information to a T-mesh. A knot interval is a non-negative real number assigned to an edge. A valid knot interval configuration requires that the knot intervals on opposite sides of every T-mesh element sum to the same value.

A valid knot interval configuration for the T-mesh in Figure 2.2 is shown in Figure 2.3. Notice that an outer ring of zero-length knot intervals has been assigned to the T-mesh. These zero-length knot intervals play a similar role to open knot vectors in NURBS and ease the imposition of boundary conditions.

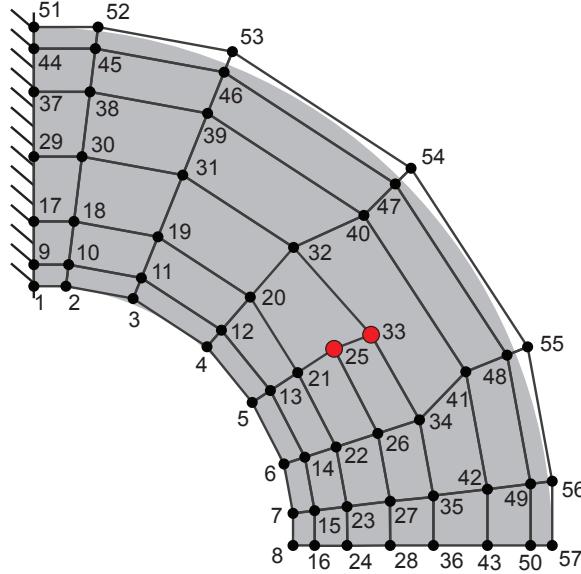


Figure 2.2: A T-mesh defining a bicubic T-spline geometry. The large red circles are the T-junctions for this T-mesh. The indexing identifies the T-mesh control points.

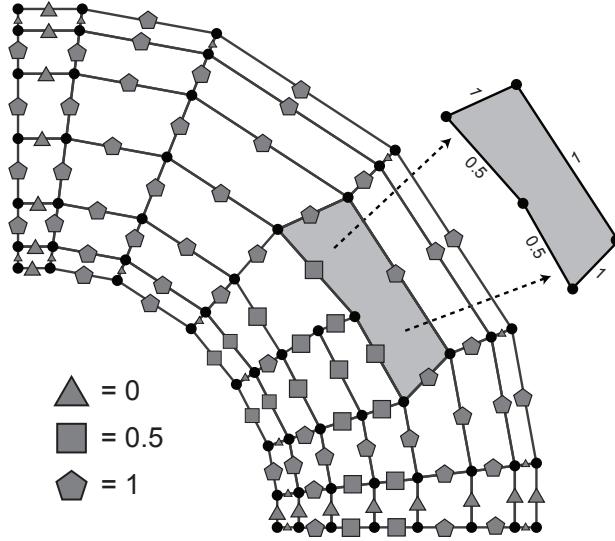


Figure 2.3: A valid knot interval configuration for the bicubic T-mesh in Figure 2.2. The triangles correspond to a knot interval of 0, the squares correspond to a knot interval of  $\frac{1}{2}$ , and the pentagons correspond to a knot interval of 1. A valid knot interval configuration for a T-mesh element is shown in the element callout. Notice that the knot intervals along opposing sides of the element sum to the same value.

## 2.2.2 The T-spline basis<sup>2</sup>

T-spline basis functions can be inferred from a T-mesh with a valid knot interval configuration. A T-spline basis function is associated with each vertex in the T-mesh. We illustrate the construction of the T-spline basis functions associated with  $\mathbf{P}_{18}$  and  $\mathbf{P}_{33}$  in Figures 2.4 and 2.5, respectively.

### 2.2.2.1 Local knot interval vectors

The first step in constructing a T-spline basis function is inferring sequences of knot intervals from the T-mesh in the neighborhood of the associated vertex. These knot intervals are organized into local knot interval vectors. A local knot interval vector is a sequence of knot intervals,  $\Delta\Xi = \{\Delta\xi_1, \Delta\xi_2, \dots, \Delta\xi_{p+1}\}$ , such that  $\Delta\xi_i = \xi_{i+1} - \xi_i$ , and a local knot vector, derivable from any  $\Delta\Xi$ , is a non-decreasing knot sequence,  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{p+2}\}$ . A local knot interval vector possesses all the information in a local knot vector except an origin. For example, if a knot interval vector is  $\{1, 3, 2, 1\}$ , then we can set the origin to zero and the corresponding local knot vector is  $\{0, 1, 4, 6, 7\}$ . If instead we are given a local knot vector, then the local knot interval vector is simply the difference between adjacent knots. In general, for T-splines, knot intervals are the method of choice for assigning and retrieving parameter information to and from the T-mesh since no origin is required. It should be noted that all classical B-spline algorithms can be rewritten in terms of knot intervals.

To every vertex,  $A$ , in the T-mesh we assign a set of local knot interval vectors,  $\Delta\Xi_A = \{\Delta\xi_A^i\}_{i=1}^{d_p}$ , from which a corresponding set of local knot vectors,  $\Xi_A =$

---

<sup>2</sup>The mathematical term “basis” implies linear independence. The proof that T-spline blending functions in fact constitute a basis has recently been given for a class of T-splines [65].

$\{\Xi_A^i\}_{i=1}^{d_p}$ , can be derived. The local knot interval vectors  $\Delta\Xi_A$  are constructed by marching through the T-mesh in each topological direction, starting at the vertex  $A$ , until  $p - 1$  vertices or perpendicular edges are intersected. If a vertex or perpendicular edge is encountered, the knot interval distance traversed since the last intersection is placed in the local knot interval vector. If a T-mesh boundary is crossed before  $p - 1$  knot intervals are encountered, knot intervals are appended to complete the knot interval vector. In analysis these appended knot intervals are often chosen to be equal to zero to create an open knot vector structure along the boundary of the T-mesh.

In Figure 2.4, in the upper left corner, the knot intervals used to construct the local knot vectors for T-mesh vertex  $P_{18}$  are shown. Notice that this vertex is near the boundary of the T-mesh. When marching to the left only a single knot interval is encountered before reaching the boundary of the T-mesh. As a result an additional zero knot interval is added to the front of the local knot interval vector. The knot interval vectors for  $P_{18}$  are given by

$$\Delta\Xi_{18} = \begin{bmatrix} 0, 0, 1, 1 \\ 0, 1, 1, 1 \end{bmatrix}. \quad (2.15)$$

In Figure 2.5, in the upper left corner, the knot intervals used to construct the local knot interval vectors for T-mesh vertex  $P_{33}$  are shown. Notice that this is a T-junction vertex. In this case, T-mesh elements must be traversed to form the local knot interval vectors. When an element is traversed the knot interval sum associated with the sides of the element which are parallel to the traversal direction are inserted into the local knot interval vector. For  $P_{33}$ , the knot interval vectors are given by,

$$\Delta\Xi_{33} = \begin{bmatrix} 1, \frac{1}{2}, \frac{1}{2}, 1 \\ \frac{1}{2}, \frac{1}{2}, 1, 1 \end{bmatrix}. \quad (2.16)$$

### 2.2.2.2 The local basis function domain

Each set of local knot vectors  $\Xi_A$  defines a local basis function domain,  $\widehat{\Omega}_A \subset \mathbb{R}^{d_p}$ , over which a single T-spline basis function is defined. The local basis function domain is defined as

$$\widehat{\Omega}_A = \bigotimes_{i=1}^{d_p} \widehat{\Omega}_A^i, \quad (2.17)$$

where  $\widehat{\Omega}_A^i = [0, \Delta\xi_1^i + \Delta\xi_2^i + \Delta\xi_3^i + \Delta\xi_4^i] \subset \mathbb{R}$ . Each local basis function domain carries a coordinate system  $\xi_A = (\xi_A^1, \xi_A^2) = (\xi_A, \eta_A)$ . This coordinate system is called the basis coordinate system.

Local basis function domains  $\widehat{\Omega}_{18}$  and  $\widehat{\Omega}_{33}$  are shown in the bottom right corners of Figures 2.4 and 2.5, respectively. In the case of  $\widehat{\Omega}_{18}$  we have that  $\widehat{\Omega}_{18}^1 = [0, 2]$  and  $\widehat{\Omega}_{18}^2 = [0, 3]$ . In the case of  $\widehat{\Omega}_{33}$  we have that  $\widehat{\Omega}_{33}^1 = [0, 3]$  and  $\widehat{\Omega}_{33}^2 = [0, 3]$ .

### 2.2.2.3 T-spline basis functions

Over each local basis function domain  $\widehat{\Omega}_A$  we define a single T-spline basis function,  $N_A : \widehat{\Omega}_A \rightarrow \mathbb{R}^+ \cup 0$ . This is done by forming the tensor product of the univariate basis functions  $\{N_A^i(\xi_A | \Xi_A^i)\}_{i=1}^{d_p}$  as

$$N_A(\xi_A | \Xi_A) \equiv \prod_{i=1}^{d_p} N_A^i(\xi_A^i | \Xi_A^i). \quad (2.18)$$

The univariate T-spline basis function,  $N_A^i : \widehat{\Omega}_A^i \rightarrow \mathbb{R}^+ \cup 0$ , is defined using a recurrence relation, starting with the piecewise constant ( $p = 0$ ) basis function

$$N_A^i(\xi_A^i | \xi_{A,1}^i, \xi_{A,2}^i) = \begin{cases} 1 & \text{if } \xi_{A,1}^i \leq \xi_A^i < \xi_{A,2}^i \\ 0 & \text{otherwise} \end{cases}, \quad (2.19)$$

where  $\xi_{A,k}^i$  is the  $k^{\text{th}}$  knot value in the localized knot vector  $\Xi_A^i$ . For  $p > 0$ , the basis function is defined using the Cox-de Boor recursion formula:

$$\begin{aligned} N_A^i(\xi_A^i | \xi_{A,1}^i, \xi_{A,2}^i, \dots, \xi_{A,p+2}^i) &= \frac{\xi_A^i - \xi_{A,1}^i}{\xi_{A,p+1}^i - \xi_{A,1}^i} N_A^i(\xi_A^i | \xi_{A,1}^i, \xi_{A,2}^i, \dots, \xi_{A,p+1}^i) + \\ &\quad + \frac{\xi_{A,p+2}^i - \xi_A^i}{\xi_{A,p+2}^i - \xi_{A,2}^i} N_A^i(\xi_A^i | \xi_{A,2}^i, \xi_{A,3}^i, \dots, \xi_{A,p+2}^i). \end{aligned} \quad (2.20)$$

The T-spline basis functions  $N_{18}$  and  $N_{33}$  are shown in the lower left corner of Figures 2.4 and 2.5, respectively. Between knots, the one-dimensional basis functions (2.20) are  $C^\infty$  continuous. At knots the continuity is reduced (see Cottrell et al. [35]).

**Remark 2.2.** It should be noted that while algorithms exist for computing T-spline basis functions according to (2.20) (see Piegl and Tiller [78]), using the recursive definition in finite element shape function routines is expensive when compared to the extracted element technology introduced in Section 2.3 below.

### 2.2.3 The T-spline element structure

We now develop the finite element structure underlying T-splines. A T-spline element  $\Omega^e \subset \mathbb{R}^{d_s}$  is a region in physical space which is bounded by knot lines, which are lines of reduced continuity in the T-spline basis (we use the terminologies knot lines and lines of reduced continuity interchangeably throughout this chapter). The basis functions restricted to the interior of the T-spline element are  $C^\infty$ -continuous.

#### 2.2.3.1 The local basis function mesh

From any set of local knot interval vectors  $\Delta\Xi_A$  (and corresponding local knot vectors  $\Xi_A$ ) a local basis function mesh,  $T_A$ , can be defined as the tensor product mesh

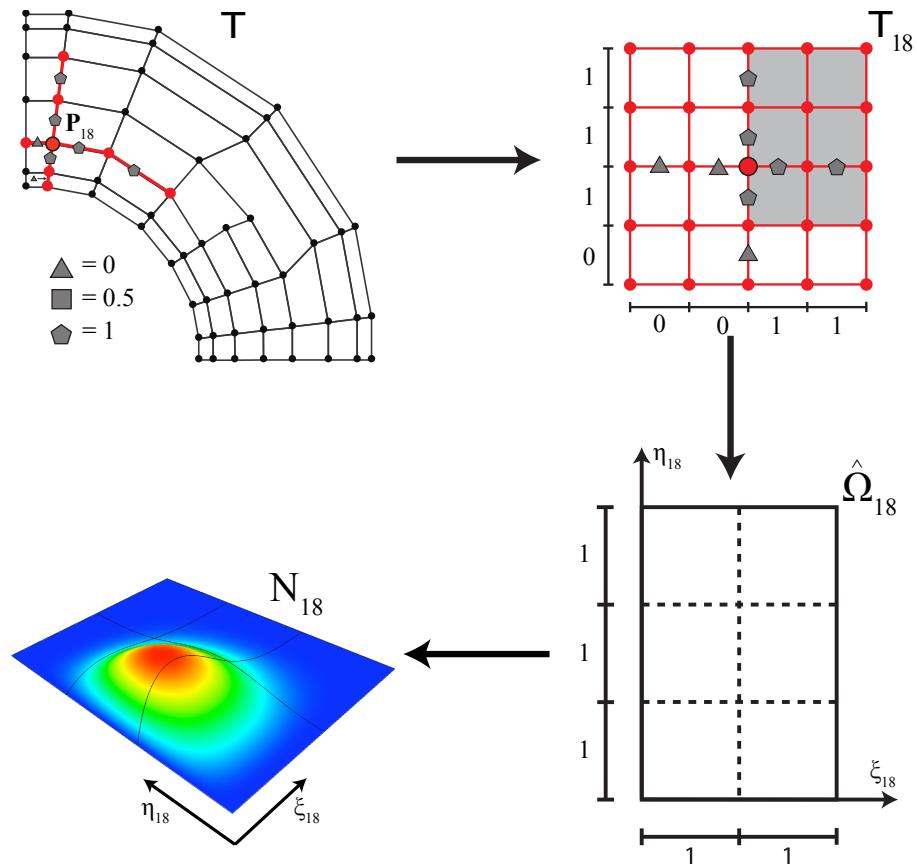


Figure 2.4: The construction of T-spline basis function  $N_{18}$  which is associated with T-mesh vertex  $P_{18}$ . Starting at the upper left and going clockwise we have: Inference of the local knot vectors from the T-mesh, the resulting local basis function mesh, the local basis function domain, and the T-spline basis function.

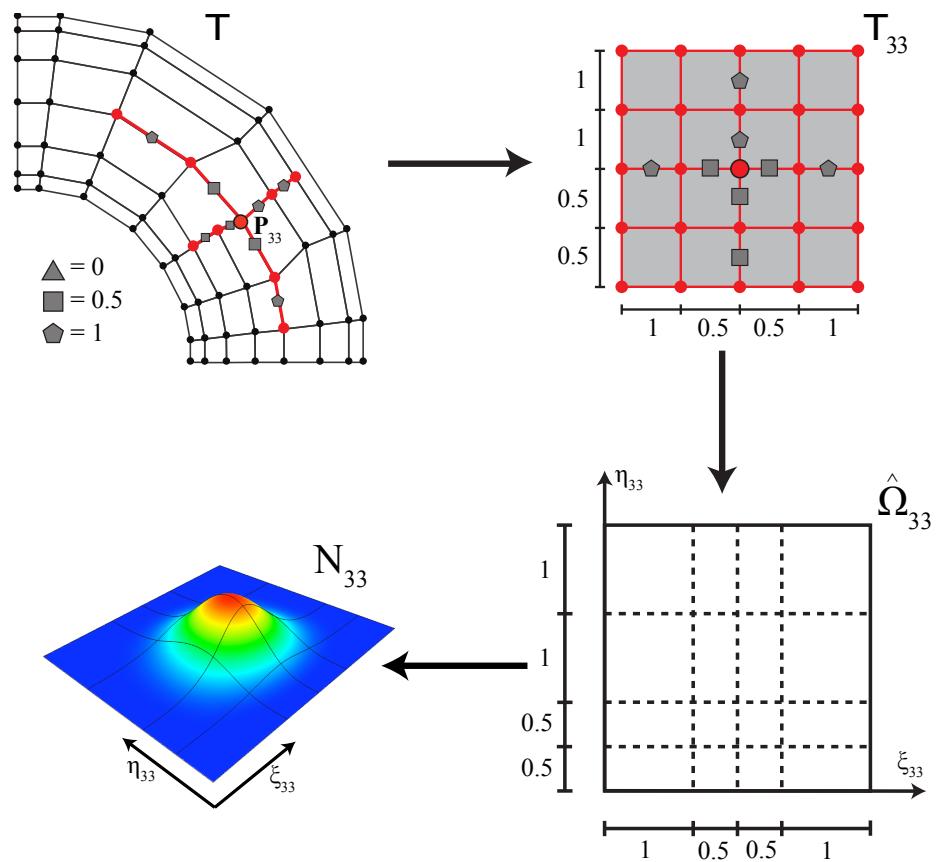


Figure 2.5: The construction of T-spline basis function  $N_{33}$  which is associated with the T-junction T-mesh vertex  $P_{33}$ . Starting at the upper left and going clockwise we have: Inference of the local knot vectors from the T-mesh, the resulting local basis function mesh, the local basis function domain, and the T-spline basis function.

representation of the local knot vectors

$$\mathsf{T}_A = \bigotimes_{i=1}^{d_p} \Xi_A^i. \quad (2.21)$$

Each edge in  $\mathsf{T}_A$  continues to carry the appropriate knot interval from  $\Delta\Xi_A$ .

Figure 2.4, in the upper right corner, shows the local basis function mesh,  $\mathsf{T}_{18}$ , generated from  $\Delta\Xi_{18}$ . The shaded region indicates the local basis function mesh elements with positive parametric area. Figure 2.5, in the upper right corner, shows the local basis function mesh,  $\mathsf{T}_{33}$ , generated from  $\Delta\Xi_{33}$ . In this case every element has positive parametric area.

### 2.2.3.2 The elemental T-mesh

A T-mesh element does not necessarily have a one-to-one correspondence with a T-spline element. We recall that a T-mesh element is a quadrilateral in the T-mesh or T-spline control mesh and the T-spline element is a region of the T-spline surface bounded by lines of reduced continuity in the T-spline basis. This can be seen by drawing the local basis function mesh  $\mathsf{T}_{33}$  on top of the T-mesh, as in Figure 2.6. The dashed lines in Figure 2.6 indicate edges which exist in  $\mathsf{T}_{33}$  but not in  $\mathsf{T}$ . Each knot line in the T-spline basis is present in at least one basis function. However, not all these knot lines are represented by corresponding lines in the T-mesh. We form the *elemental T-mesh* by augmenting the T-mesh with all these lines. The elemental T-mesh  $\mathsf{T}_f$  of  $\mathsf{T}$  is shown in Figure 2.7. The dashed edges indicate lines of reduced continuity which were not present in the original T-mesh.

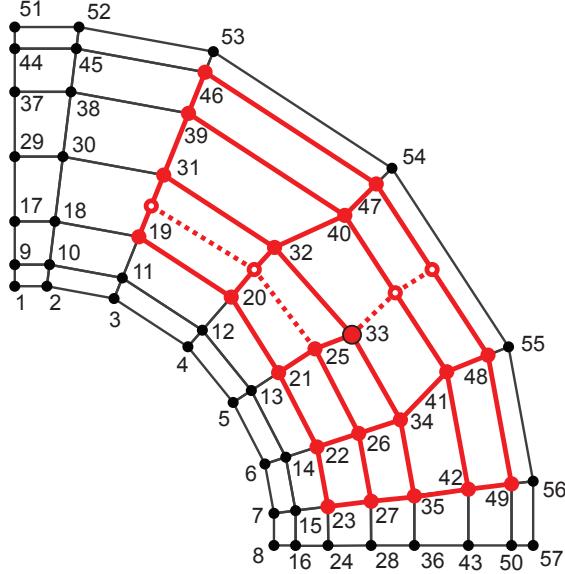


Figure 2.6: The mapping of  $T_{33}$  onto the global T-mesh  $T$ . The dashed edges correspond to lines of reduced continuity from  $T_{33}$  which are not in  $T$ .

### 2.2.3.3 The IEN array

Over every element domain there exists a set of T-spline basis functions which are non-zero. The T-spline basis functions are in one-to-one correspondence with the T-mesh control points and are indexed by the global control point numbers. The IEN array maps the local basis function number,  $a$ , and the element number,  $e$ , to the corresponding global control point number  $A$ . There can be different numbers of T-spline basis functions supported by each element. This is in contrast to NURBS where all elements are in the support of exactly  $(p + 1)^{d_p}$  NURBS basis functions. The complete IEN array for the T-spline elements in Figure 2.7 is shown in Table 2.1. Notice that elements 9 and 10 are in the supports of 17 T-spline basis functions. Figure 2.8 shows T-spline elements 1, 10, 11, and 17 and the T-mesh control points whose corresponding basis functions are non-zero over each of these elements.

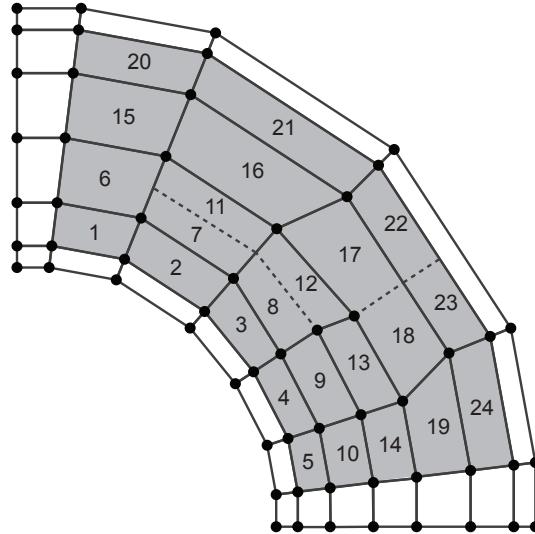


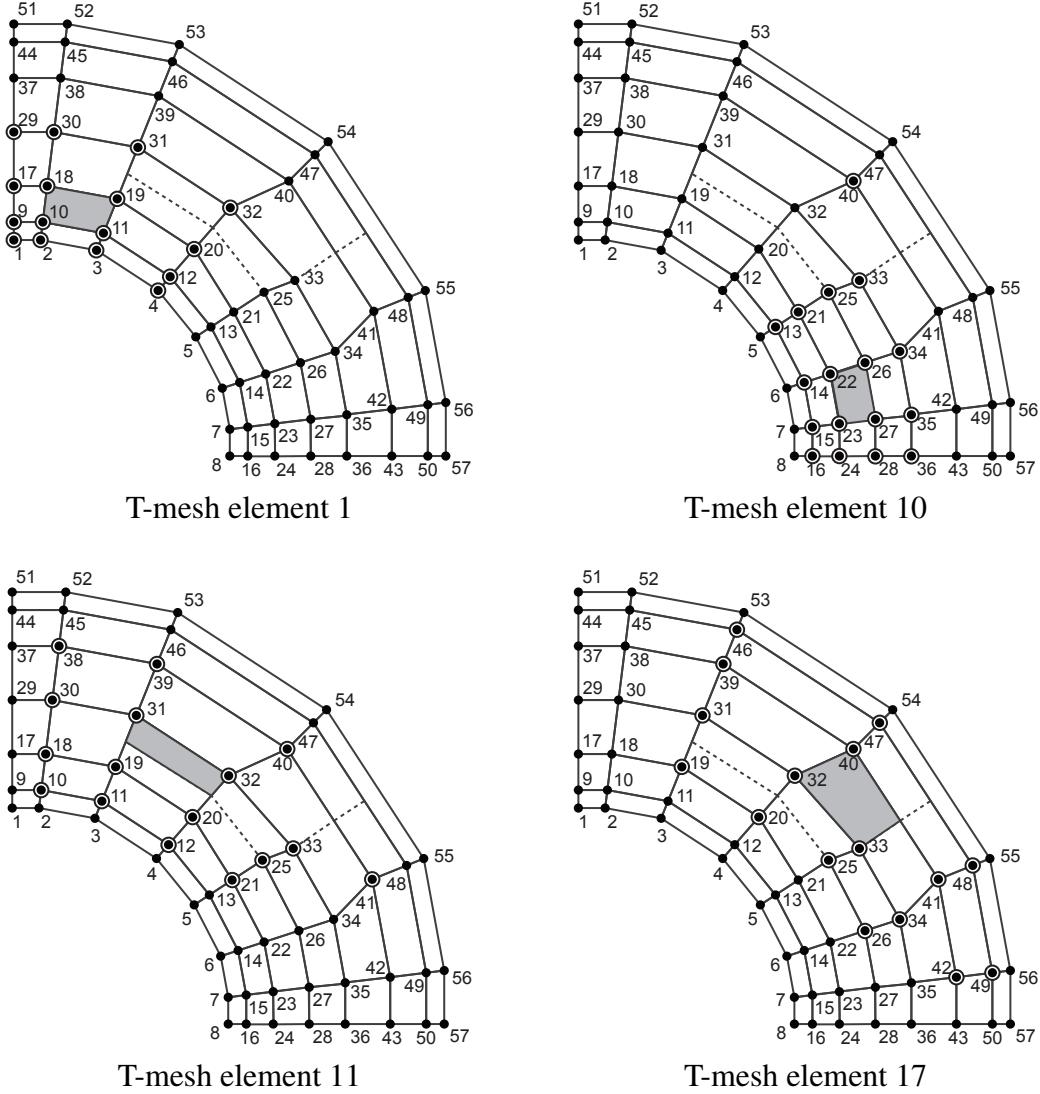
Figure 2.7: The T-spline elements in the elemental T-mesh  $T_f$  of  $T$ .

#### 2.2.3.4 Restricting the global basis to the parent element domain

We develop the finite element point-of-view for T-splines by defining the T-spline basis functions over the parent element domain. The non-local and T-junction structure of the T-spline basis requires additional machinery not found in traditional finite element constructions. We define a set,  $S^e = \{\tilde{\Phi}^e, \{\hat{\Phi}_a^e\}_{a=1}^{n_e}\}$ , of affine mappings for the element under consideration where

- $\tilde{\Phi}^e : \tilde{\Omega} \rightarrow \hat{\Omega}^e$  is a one-to-one and onto affine map from the parent element domain onto the element domain:

$$\xi^e = \tilde{\Phi}^e(\tilde{\xi}). \quad (2.22)$$



- Indicates T-spline basis functions that are supported by the highlighted element

Figure 2.8: T-mesh elements 1, 10, 11, and 17 in the elemental T-mesh and the T-mesh control points whose corresponding T-spline basis functions are non-zero over these T-spline elements.

- $\hat{\Phi}_a^e : \widehat{\Omega}^e \rightarrow \widehat{\Omega}_A$ , for  $a = 1, \dots, n_e$  is a one-to-one affine mapping from the element domain into the local basis function domain for basis function  $A = \text{IEN}(a, e)$ :

$$\xi_A = \hat{\Phi}_a^e(\xi^e). \quad (2.23)$$

These mappings can be determined using the elemental T-mesh and the IEN array and are used to map from the parent element into each local basis function domain which corresponds to a T-spline basis function which is non-zero over element  $e$ . The action of some but not all of the affine maps in  $S^{17}$  is shown in Figure 2.9.

$e$	Element function number ( $a$ )																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	9	10	11	12	17	18	19	20	29	30	31	32	
2	2	3	4	5	10	11	12	13	18	19	20	21	25	30	31	32	
3	3	4	5	6	11	12	13	14	19	20	21	22	25	26	31	32	
4	4	5	6	7	12	13	14	15	20	21	22	23	25	26	27	32	
5	5	6	7	8	13	14	15	16	21	22	23	24	25	26	27	28	
6	9	10	11	12	17	18	19	20	29	30	31	32	37	38	39	40	
7	10	11	12	13	18	19	20	21	25	30	31	32	33	38	39	40	
8	11	12	13	14	19	20	21	22	25	26	31	32	33	34	39	40	
9	12	13	14	15	20	21	22	23	25	26	27	32	33	34	35	39	40
10	13	14	15	16	21	22	23	24	25	26	27	28	33	34	35	36	40
11	10	11	12	18	19	20	21	25	30	31	32	33	38	39	40	41	
12	11	12	19	20	21	22	25	26	31	32	33	34	39	40	41	42	
13	12	20	21	22	23	25	26	27	32	33	34	35	39	40	41	42	
14	21	22	23	24	25	26	27	28	33	34	35	36	40	41	42	43	
15	17	18	19	20	29	30	31	32	37	38	39	40	44	45	46	47	
16	18	19	20	25	30	31	32	33	38	39	40	41	45	46	47	48	
17	19	20	25	26	31	32	33	34	39	40	41	42	46	47	48	49	
18	20	25	26	27	32	33	34	35	39	40	41	42	46	47	48	49	
19	25	26	27	28	33	34	35	36	40	41	42	43	47	48	49	50	
20	29	30	31	32	37	38	39	40	44	45	46	47	51	52	53	54	
21	30	31	32	33	38	39	40	41	45	46	47	48	52	53	54	55	
22	31	32	33	34	39	40	41	42	46	47	48	49	53	54	55	56	
23	32	33	34	35	39	40	41	42	46	47	48	49	53	54	55	56	
24	33	34	35	36	40	41	42	43	47	48	49	50	54	55	56	57	

$$A = \text{IEN}(a, e)$$

Table 2.1: The IEN array is constructed using the information from Figure 2.8. The IEN array maps the local basis function number ( $a$ ) and the element number ( $e$ ) to the corresponding global control point ( $A$ ). The local basis function number indexes the T-spline basis functions supported by the element in question. Note that there can be different numbers of T-spline basis functions associated with different elements.

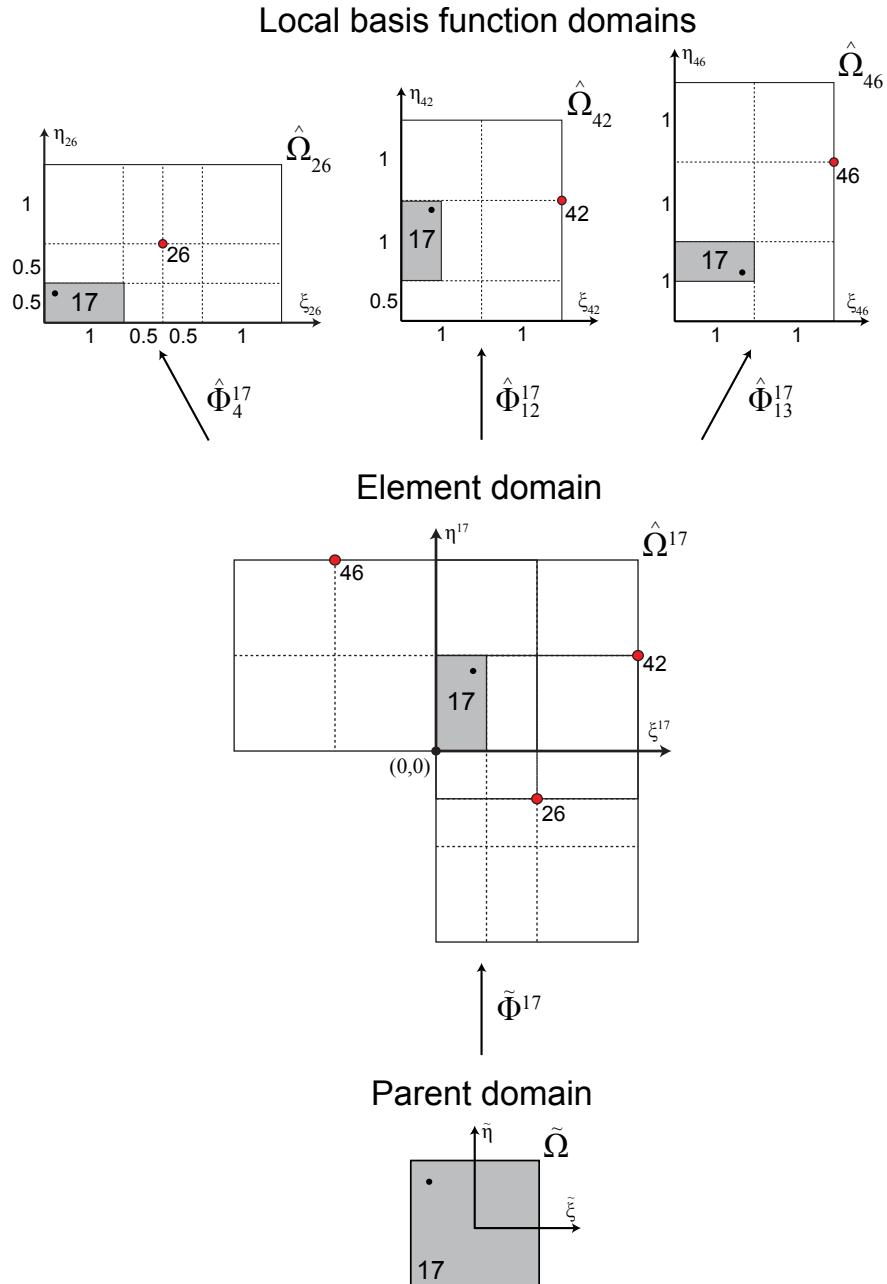


Figure 2.9: The mappings between parent, element, and basis function coordinate systems for element 17. For simplicity only some of the mappings are shown.  $\tilde{\Phi}^{17}$  maps from the parent coordinate system into the element coordinate system.  $\hat{\Phi}_a^{17}$  maps from the element coordinate system of local basis function number  $a$  into the basis function coordinate system of global control point  $A = \text{IEN}(a, 17)$ . The solid dot ( $\bullet$ ) on element 17 shows the rotations incorporated in the mappings.

Using  $\mathbf{S}^e$  and the IEN array we can define a localized, element-based definition for the global T-spline basis functions as

$$N_A(\boldsymbol{\xi}_A)|_e = N_A\left(\hat{\Phi}_a^e(\boldsymbol{\xi}^e)\right)|_e = N_A\left(\hat{\Phi}_a^e\left(\tilde{\Phi}^e(\tilde{\boldsymbol{\xi}})\right)\right)|_e = N_a^e(\tilde{\boldsymbol{\xi}}) \quad (2.24)$$

where  $|_e$  indicates restriction to the domain of element number  $e$ .

### 2.2.3.5 The T-spline element geometric map

With the basis functions defined over the parent element by equation (2.24), we now define the element geometric map,  $\tilde{\mathbf{x}}^e : \tilde{\Omega} \rightarrow \Omega^e$ , from the parent element domain onto the physical domain as

$$\tilde{\mathbf{x}}^e(\tilde{\boldsymbol{\xi}}) = \frac{\sum_{a=1}^{n_e} \mathbf{P}_a^e w_a^e N_a^e(\tilde{\boldsymbol{\xi}})}{W^e(\tilde{\boldsymbol{\xi}})} = \sum_{a=1}^{n_e} \mathbf{P}_a^e R_a^e(\tilde{\boldsymbol{\xi}}) \quad (2.25)$$

where

$$W^e(\tilde{\boldsymbol{\xi}}) = \sum_{a=1}^{n_e} w_a^e N_a^e(\tilde{\boldsymbol{\xi}}) \quad (2.26)$$

is the element weight function, and  $\mathbf{P}_a^e = \mathbf{P}_{\text{IEN}(a,e)}$  and  $w_a^e = w_{\text{IEN}(a,e)}$  are the control point and weight, respectively, corresponding to the  $a^{\text{th}}$  T-spline basis function over element  $e$ .  $R_a^e$  is the rational form of the T-spline basis function because it includes the weight function in its denominator. Defining the element weight vector  $\mathbf{w}^e = \{w_a^e\}_{a=1}^{n_e}$ , the diagonal weight matrix  $\mathbf{W}^e = \text{diag}(\mathbf{w}^e)$ , and element control points  $\mathbf{P}^e$  as a matrix of dimension  $n_e \times d_s$ ,

$$\mathbf{P}^e = \begin{bmatrix} P_1^{e,1} & P_1^{e,2} & \dots & P_1^{e,d_s} \\ P_2^{e,1} & P_2^{e,2} & \dots & P_2^{e,d_s} \\ \vdots & \vdots & & \vdots \\ P_{n_e}^{e,1} & P_{n_e}^{e,2} & \dots & P_{n_e}^{e,d_s} \end{bmatrix}, \quad (2.27)$$

we can generate the corresponding matrix representation of the geometric map as

$$\tilde{\mathbf{x}}^e(\tilde{\boldsymbol{\xi}}) = \frac{1}{(\mathbf{w}^e)^T \mathbf{N}^e(\tilde{\boldsymbol{\xi}})} (\mathbf{P}^e)^T \mathbf{W}^e \mathbf{N}^e(\tilde{\boldsymbol{\xi}}) \quad (2.28)$$

$$= (\mathbf{P}^e)^T \mathbf{R}^e(\tilde{\boldsymbol{\xi}}), \quad (2.29)$$

where  $\mathbf{N}^e = \{N_a^e\}_{a=1}^{n_e}$  and  $\mathbf{R}^e = \{R_a^e\}_{a=1}^{n_e}$  are the vectors of polynomial and rational T-spline basis functions, respectively, which are non-zero over element  $e$ .

## 2.3 Bézier extraction

In order to facilitate computation using isogeometric basis functions, Bézier extraction was introduced for NURBS by Borden et al. [22] and for T-splines by Scott et al. [88]. Bézier extraction constructs a set of Bézier elements defining a NURBS or T-spline. A Bézier element is a region of the physical domain in which the basis functions are  $C^\infty$ -continuous and over which integration is performed. For NURBS, the Bézier elements correspond to non-zero knot spans in the parametric domain. For T-splines, they correspond to the element structure introduced in the previous section.

In this section we describe the construction of isogeometric Bézier elements and the Bézier extraction operator. These constructs provide an element structure for isogeometric analysis that can be incorporated into existing finite element codes. In fact, the same data processing arrays utilized in finite element analysis, namely, the IEN, ID, and LM arrays ( see Hughes [52]), are also sufficient for isogeometric Bézier elements. The centrality of Bézier extraction in unifying common CAD representations and FEA is illustrated in Figure 2.10.

The Bézier extraction operator maps a piecewise Bernstein polynomial basis onto a B-spline basis. This transformation makes it possible to use piecewise  $C^0$  Bézier

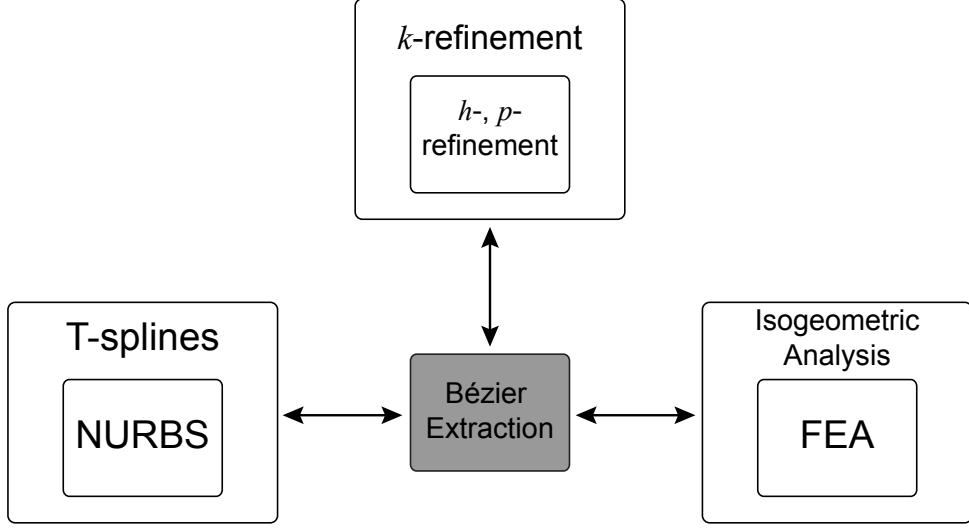


Figure 2.10: A schematic diagram illustrating the central role played by Bézier extraction in unifying CAGD and FEA.

elements as the finite element representation of a NURBS or T-spline. The idea is illustrated in Figure 2.11 for a cubic B-spline curve. Over each knot interval,  $\Omega_e$ , we can compute an element extraction operator,  $\mathbf{C}^e$ . The action of this operator is shown for knot interval  $\Omega_2$  in Figure 2.11. In general, we can write the B-spline basis over each interval,  $e$ , as

$$\mathbf{N}^e = \mathbf{C}^e \mathbf{B} \quad (2.30)$$

where  $\mathbf{N}^e = \{N_a\}_{a=1}^{n_e}$  are the B-spline basis functions supported by element  $e$  and  $\mathbf{B} = \{B_a\}_{a=1}^{p+1}$  are the Bernstein polynomials of degree  $p$ . For a rational basis with  $\mathbf{w}^e = \{w_a\}_{a=1}^{n_e}$  being the weights for  $n_e$  basis functions supported by element  $e$  we have

$$\mathbf{R}^e = \mathbf{W}^e \frac{\mathbf{N}^e}{W} = \mathbf{W}^e \frac{\mathbf{C}^e \mathbf{B}}{W} \quad (2.31)$$

where  $\mathbf{W}^e = \text{diag}(\mathbf{w}^e)$  and  $W$  is the weight function.

In this section we begin by showing how the Bézier element representation of a NURBS can be computed. We then show the same for T-splines.

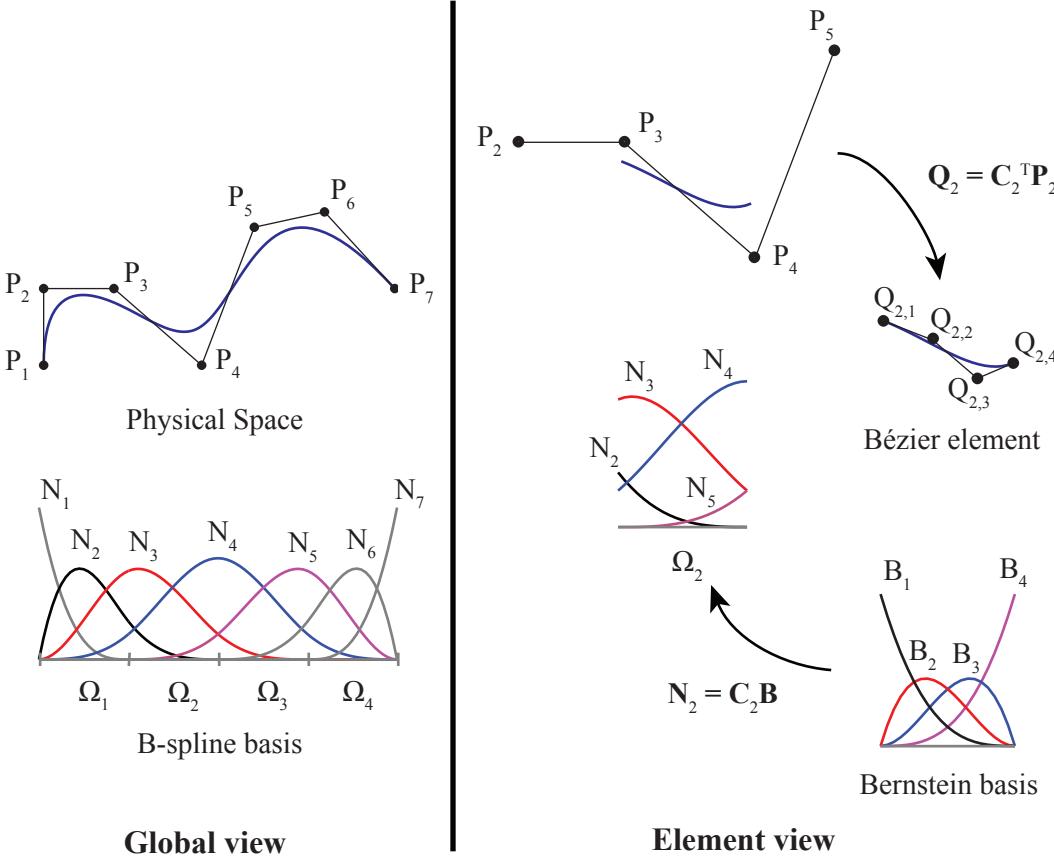


Figure 2.11: Bézier extraction for a cubic B-spline curve. The B-spline curve and basis functions are shown on the left. The action of the extraction operator,  $C_2$ , for element  $\Omega_2$  is illustrated on the right. The transpose of the extraction operator defines the control points,  $Q_2 = \{Q_{2,a}\}_{a=1}^4$ , of the Bézier element from the control points,  $P_2 = \{P_a\}_{a=2}^5$  of the B-spline curve. The B-spline basis functions,  $N_2 = \{N_a\}_{a=2}^5$ , can be computed over the element by applying the extraction operator to Bernstein polynomial basis functions,  $B = \{B_a\}_{a=1}^4$ , defined on the Bézier element. Note that the Bernstein basis is the same for each element. This allows the formation of element arrays to be standardized in a shape function routine.

### 2.3.1 The Bernstein basis

The Bernstein polynomials form a basis for the Bézier element. The univariate Bernstein basis functions are defined over the biunit interval  $[-1, 1]$  as

$$B_{i,p}^k(\tilde{\xi}^k) = \frac{1}{2^p} \binom{p}{i-1} (1 - \tilde{\xi}^k)^{p-(i-1)} (1 + \tilde{\xi}^k)^{i-1} \quad (2.32)$$

where the binomial coefficient  $\binom{p}{i-1} = \frac{p!}{(i-1)!(p+1-i)!}$ ,  $1 \leq i \leq p+1$ . In CAGD, The Bernstein polynomials are usually defined over the unit interval  $[0, 1]$ , but in finite element analysis the biunit interval is preferred to take advantage of the usual domains for Gauss quadrature. The univariate Bernstein basis functions for  $p = 1, 2$ , and  $3$  are plotted in Figure 2.12. The univariate Bernstein basis has the following properties:

- Partition of unity.

$$\sum_{i=1}^{p+1} B_{i,p}^k(\tilde{\xi}^k) = 1 \quad \forall \tilde{\xi}^k \in [-1, 1] \quad (2.33)$$

- Pointwise nonnegativity.

$$B_{i,p}^k(\tilde{\xi}^k) \geq 0 \quad \forall \tilde{\xi}^k \in [-1, 1] \quad (2.34)$$

- Endpoint interpolation.

$$B_{1,p}^k(-1) = B_{p+1,p}^k(1) = 1 \quad (2.35)$$

- Symmetry.

$$B_{i,p}^k(\tilde{\xi}^k) = B_{p+1-i,p}^k(-\tilde{\xi}^k) \quad \forall \tilde{\xi}^k \in [-1, 1] \quad (2.36)$$

The multivariate Bernstein basis functions of degree  $p$ ,  $B_{a,p} : \tilde{\Omega} \rightarrow \mathbb{R}^+ \cup 0$ , with  $a = 1, \dots, (p+1)^{d_p}$ , are formed as the tensor-product of univariate basis functions  $B_{i,p}^k : [-1, 1] \rightarrow \mathbb{R}^+ \cup 0$ , with  $i = 1, \dots, p+1$ . In the bivariate case we have

$$B_{a(i,j),p}(\tilde{\xi}) = B_{i,p}^1(\tilde{\xi}^1) B_{j,p}^2(\tilde{\xi}^2). \quad (2.37)$$

with

$$a(i, j) = (p+1)(j-1) + i. \quad (2.38)$$

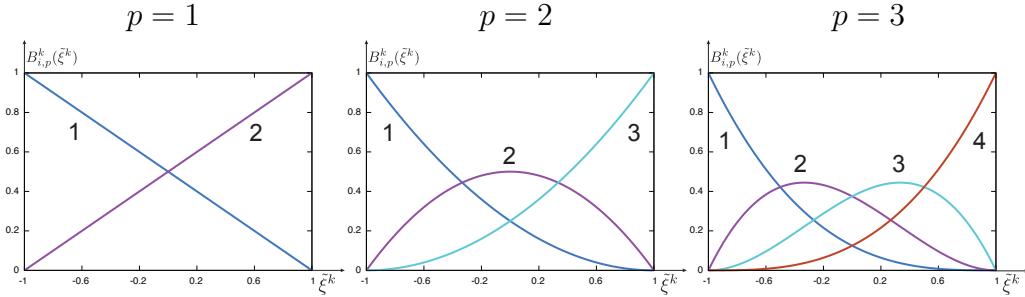


Figure 2.12: Bernstein basis functions for polynomial degree  $p = 1, 2, 3$ .

Recall that  $\tilde{\xi} = (\tilde{\xi}^1, \tilde{\xi}^2) = (\tilde{\xi}, \tilde{\eta})$  is the coordinate system assigned to the parent element. From the formulas, it is clear that the basis functions are numbered from left to right in one dimension. In two dimensions each row is numbered from left to right, starting with the bottom row and moving upward. See Figure 2.13.

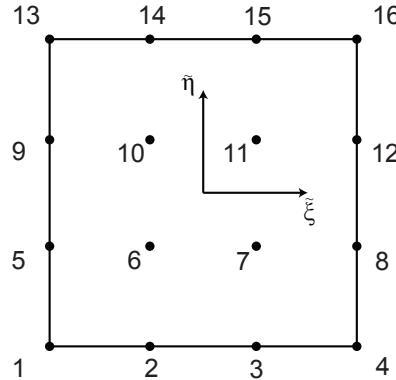


Figure 2.13: Ordering of the two-dimensional Bernstein polynomials on a bicubic Bézier element.

### 2.3.2 Computing the element extraction operator for NURBS

To compute the Bézier elements and extraction operators of a NURBS we use Bézier decomposition. Technically, Bézier decomposition is accomplished by repeating

all interior knots of a knot vector until they have a multiplicity equal to  $p + 1$ . For our purposes, however, a multiplicity of  $p$  is sufficient. The result of this lower multiplicity is that neighboring Bézier elements will share control points. Since we are projecting to a smooth, continuous basis this has no effect and slightly reduces computation cost. We note that although multiple applications of knot insertion may be used for Bézier decomposition there are more efficient algorithms available (see Piegl and Tiller [78] for details).

The element extraction operators,  $\mathbf{C}^e$ , can be computed directly by modifying existing Bézier decomposition algorithms. Algorithm 2.1 is based on the Bézier decomposition algorithm presented by Piegl and Tiller [78]. In its original form, this algorithm used (2.9) and (2.10) to compute the control points of the Bézier elements of a curve. We have modified this algorithm so that it uses only (2.10) to compute the coefficients of the element extraction operators. Note that by inserting knots from left to right we are able to reduce the total computational cost by updating overlapping coefficients between neighboring elements.

**Algorithm 2.1** An algorithm to compute the local extraction operators for a one-dimensional B-spline parametric domain. Note that the input does not require the geometric information of the control variables.

**input** Knot vector,  $\mathbf{U} = \{u_1, \dots, u_m\}$   
          Number of knots,  $m$   
          Curve degree,  $p$

**output** Number of elements,  $nb$   
          Element extraction operators,  $\mathbf{C}^e$ ,  $e = 1, 2, \dots, nb$

```

// Initializations:
a = p+1;
b = a+1;
nb = 1;
C1 = I;
while b < m do
    // Initialize the next extraction operator.
    Cnb+1 = I;
    i = b;

    // Count multiplicity of the knot at location b.
    while b < m && U(b+1) == U(b) do b = b+1;
    mult = b-i+1;

    if mult < p do
        // Use (2.10) to compute the alphas.
        numer = U(b)-U(a);
        for j = p,p-1,...,mult+1 do
            alphas(j-mult) = numer / (U(a+j)-U(a));
        end
        r = p-mult;
        // Update the matrix coefficients
        // for r new knots
        for j=1,2,...,r do
            save = r-j+1;
            s = mult+j;
            for k=p+1,p,...,s+1 do
                alpha = alphas(k-s);

```

```

// The following line corresponds to (2.9).
Cnb(:, k) = alpha*Cnb(:, k)
+ (1.0-alpha)*Cnb(:, k-1);

end

if b < m do
    // Update overlapping coefficients
    // of the next operator.
    Cnb+1(save:j+save, save) = Cnb(p-j+1:p+1, p+1);
end

end

// Finished with the current operator.

nb = nb + 1;
if b < m do
    // Update indices for the next operator.
    a = b;
    b = b+1;
end
end

end

```

### 2.3.2.1 Surfaces and solids

It is convenient to take advantage of the tensor product structure of higher dimension NURBS to compute the Bézier extraction operators for higher dimension. For consistency, we introduce a mapping,  $\tilde{A}$ , between the tensor product space and the global indexing of the basis functions and control points. Let  $i = 1, 2, \dots, n$ ,  $j = 1, 2, \dots, m$ , and  $k = 1, 2, \dots, l$  then in two dimensions we define

$$\tilde{A}(i, j) = m(i - 1) + j \quad (2.39)$$

and in three dimensions

$$\tilde{A}(i, j, k) = (l \times m)(i - 1) + l(j - 1) + k. \quad (2.40)$$

NURBS basis functions for surfaces and solids are defined by the tensor product of univariate B-spline basis functions. If  $N_{i,p}(\xi)$ ,  $M_{j,q}(\eta)$ , and  $L_{l,r}(\zeta)$  are univariate B-spline basis functions, then in two dimensions with  $A = \tilde{A}(i, j)$  and  $\hat{A} = \tilde{A}(\hat{i}, \hat{j})$

$$R_A^{p,q}(\xi, \eta) = \frac{M_{i,q}(\eta)N_{j,p}(\xi)w_A}{\sum_{\hat{i}=1}^n \sum_{\hat{j}=1}^m M_{\hat{i},q}(\eta)N_{\hat{j},p}(\xi)w_{\hat{A}}} \quad (2.41)$$

and in three dimensions with  $A = \tilde{A}(i, j, k)$  and  $\hat{A} = \tilde{A}(\hat{i}, \hat{j}, \hat{k})$  depending on the spatial dimension

$$R_A^{p,q,r}(\xi, \eta, \zeta) = \frac{L_{i,r}(\zeta)M_{j,q}(\eta)N_{k,p}(\xi)w_A}{\sum_{\hat{i}=1}^n \sum_{\hat{j}=1}^m \sum_{\hat{k}=1}^l L_{\hat{i},r}(\zeta)M_{\hat{j},q}(\eta)N_{\hat{k},p}(\xi)w_{\hat{A}}} \quad (2.42)$$

are the surface and solid NURBS basis functions respectively. For  $i = 1, 2, \dots, n$  and  $j = 1, 2, \dots, m$ , a NURBS surface is defined by a given control net  $\{\mathbf{P}_A\}$ ,  $A = 1, 2, \dots, (n \times m)$ , and knot vectors  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$  and  $\mathbf{H} = \{\eta_1, \eta_2, \dots, \eta_{m+q+1}\}$  as

$$S(\xi, \eta) = \sum_{A=1}^{n \times m} R_A^{p,q}(\xi, \eta) \mathbf{P}_A. \quad (2.43)$$

Similarly for a NURBS solid with  $i = 1, 2, \dots, n$ ,  $j = 1, 2, \dots, m$ ,  $k = 1, 2, \dots, l$ ,  $\{\mathbf{P}_A\}$ ,  $A = 1, 2, \dots, (n \times m \times l)$ , and knot vectors  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$ ,  $\mathbf{H} = \{\eta_1, \eta_2, \dots, \eta_{m+q+1}\}$ , and  $\mathcal{Z} = \{\zeta_1, \zeta_2, \dots, \zeta_{l+r+1}\}$  we have

$$S(\xi, \eta, \zeta) = \sum_{A=1}^{n \times m \times l} R_A^{p,q,r}(\xi, \eta, \zeta) \mathbf{P}_A. \quad (2.44)$$

To define the surface and solid element extraction operators we let  $\mathbf{C}_\xi^i$ ,  $\mathbf{C}_\eta^j$ , and  $\mathbf{C}_\zeta^k$  be the  $i^{th}$ ,  $j^{th}$ , and  $k^{th}$  univariate element extraction operators in the  $\xi$ ,  $\eta$ , and  $\zeta$

direction. Then we have for a surface and solid respectively

$$\mathbf{C}_A^e = \mathbf{C}_\eta^i \odot \mathbf{C}_\xi^j \quad (2.45)$$

and

$$\mathbf{C}_A^e = \mathbf{C}_\zeta^i \odot \mathbf{C}_\eta^j \odot \mathbf{C}_\xi^k \quad (2.46)$$

where  $\odot$  is defined for two matrices  $\mathbf{A}$  and  $\mathbf{B}$ , which may have different dimensions, as

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} A_{11}\mathbf{B} & A_{12}\mathbf{B} & \cdots \\ A_{21}\mathbf{B} & A_{22}\mathbf{B} & \\ \vdots & & \ddots \end{bmatrix}. \quad (2.47)$$

### 2.3.3 Computing the element extraction operator for T-splines

For T-splines, no global tensor product structure exists but, as was discussed in Section 2.2.2.2, a local tensor product parameter domain can be defined for each individual basis function. Thus, for T-splines, the computation of the element extraction operators is performed function-by-function, where each basis function contributes a row to each of the extraction operators corresponding to the Bézier elements in its support. For example, Figure 2.14 shows the Bézier elements in the support of T-spline basis function  $N_{40}$ . Bézier extraction of  $N_{40}$  contributes a row to the element extraction operator for each of the elements in its support.

#### 2.3.3.1 Comparing extraction for NURBS and T-splines

Similarly to NURBS, multivariate extraction operators for T-splines can be computed as products of univariate extraction operators. A major difference between NURBS and T-splines is that where NURBS have a global parameterization, T-splines have a local parameterization. This leads to two differences in how the extraction operators are

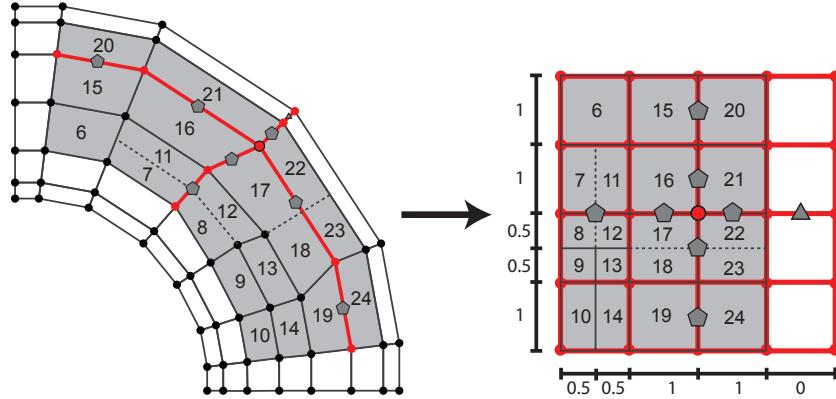


Figure 2.14: All Bézier elements in the support of T-spline basis function  $N_{40}$  in the elemental T-mesh,  $T_f$  (left) and the local basis function mesh,  $T_{40}$  (right).

computed. First, with T-splines, we work with local knot vectors that are, in general, not open, i.e., the first and last knot may have multiplicity less than  $p + 1$ . Second, since the local knot vectors correspond to individual functions, as a local knot vector is processed, we compute a single row for each of the corresponding element extraction operators, as opposed to computing the entire element extraction operator at once. Additionally, since there may be T-junctions in a T-spline, knots may need to be inserted into the knot spans of the local knot vector as part of the extraction process.

### 2.3.3.2 The extended knot vector

To handle the local knot vectors we introduce the extended knot vector,  $\bar{\Xi}_A$ . The extended knot vector is constructed by repeating the first and last knots in the local knot vector until they have a multiplicity equal to  $p + 1$ . The extended knot vector does not change the parametric description of the original T-spline basis function, but it adds functions to the local basis function domain. For example, Figure 2.15 shows the univariate T-spline basis function (solid line) with local knot vector  $\Xi = \{0, 1, 2, 3, 4\}$

and the additional basis functions (dashed lines) added by the extended knot vector  $\bar{\Xi} = \{0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4\}$ . When the functions are numbered from left to right the T-spline basis function will be numbered  $n_t + 1$  where  $n_t$  is the number of knots added to the front of the local knot vector when constructing the extended knot vector.

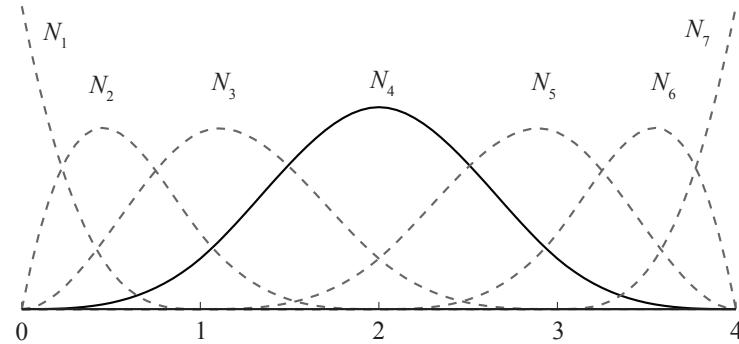


Figure 2.15: A univariate T-spline basis function (solid line) with local knot vector  $\Xi = \{0, 1, 2, 3, 4\}$ . The extended knot vector  $\bar{\Xi} = \{0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4\}$  adds the additional functions (dashed lines) that will be used to construct the basis functions for the Bézier elements.

### 2.3.3.3 Univariate extraction

We now describe how the univariate extraction operators are computed for T-splines. Once the extended knot vector has been constructed for the original T-spline basis function, the corresponding rows of the univariate extraction operators can be computed. These extraction operator rows are computed by performing knot insertion on the extended knot vector. Conceptually, the process is similar to computing the extraction operator for NURBS. For example, if the extended knot vector for the function in Figure 2.15 is considered as a knot vector for a NURBS curve, Bézier extraction would produce the basis functions shown in Figure 2.16, which are the same basis functions that are produced by extraction for the T-spline basis function. The difference between

extraction for NURBS and T-splines is that for T-splines we only compute a single row of the operator. For comparison, Table 2.2 shows the full extraction operators that would be computed in the case of NURBS with the rows corresponding to the T-spline basis function from Figure 2.15 highlighted. The extraction operator rows are computed by Algorithm 2.2, which is a modified version of Algorithm 2.1.

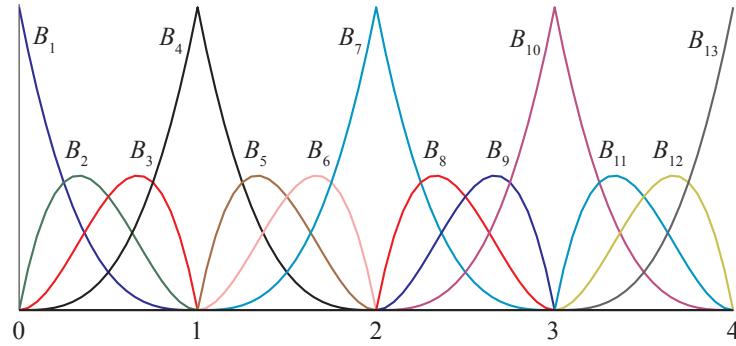


Figure 2.16: The resulting basis functions of the Bézier elements after knot insertion and refinement of the basis functions shown in Figure 2.15.

As a result of the extraction process we have a representation of each T-spline function in terms of the basis functions of the Bézier elements. This can be seen by considering Bézier element  $e$  in the support of the T-spline basis function  $A$ . The T-spline basis function  $A$  restricted to element  $e$  is related to the Bernstein basis functions by

$$N_A(\xi_A)|_e = N_a^e(\tilde{\xi}) = \mathbf{e}_a^T \mathbf{N}^e(\tilde{\xi}) = \mathbf{e}_a^T \mathbf{C}^e \mathbf{B}(\tilde{\xi}) = \mathbf{c}_a^{e^T} \mathbf{B}(\tilde{\xi}) \quad (2.48)$$

where  $\mathbf{e}_a$  is a unit vector which is equal to 1 in entry  $a$  and zero elsewhere. The vector  $\mathbf{c}_a^e$  extracts basis function  $A = \text{IEN}(a, e)$  from Bézier element  $e$ .

$$\begin{aligned}
\left\{ \begin{array}{c} N_1 \\ N_2 \\ N_3 \\ N_4 \end{array} \right\} &= \left[ \begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 1/2 & 1/4 \\ 0 & 0 & 1/2 & 7/12 \\ 0 & 0 & 0 & 1/6 \end{array} \right] \left\{ \begin{array}{c} B_1 \\ B_2 \\ B_3 \\ B_4 \end{array} \right\} \\
\left\{ \begin{array}{c} N_2 \\ N_3 \\ N_4 \\ N_5 \end{array} \right\} &= \left[ \begin{array}{cccc} 1/4 & 0 & 0 & 0 \\ 7/12 & 2/3 & 1/3 & 1/6 \\ 1/6 & 1/3 & 2/3 & 2/3 \\ 0 & 0 & 0 & 1/6 \end{array} \right] \left\{ \begin{array}{c} B_4 \\ B_5 \\ B_6 \\ B_7 \end{array} \right\} \\
\left\{ \begin{array}{c} N_3 \\ N_4 \\ N_5 \\ N_6 \end{array} \right\} &= \left[ \begin{array}{cccc} 1/6 & 0 & 0 & 0 \\ 2/3 & 2/3 & 1/3 & 1/6 \\ 1/6 & 1/3 & 2/3 & 7/12 \\ 0 & 0 & 0 & 1/4 \end{array} \right] \left\{ \begin{array}{c} B_7 \\ B_8 \\ B_9 \\ B_{10} \end{array} \right\} \\
\left\{ \begin{array}{c} N_4 \\ N_5 \\ N_6 \\ N_7 \end{array} \right\} &= \left[ \begin{array}{cccc} 1/6 & 0 & 0 & 0 \\ 7/12 & 1/2 & 0 & 0 \\ 1/4 & 1/2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right] \left\{ \begin{array}{c} B_{10} \\ B_{11} \\ B_{12} \\ B_{13} \end{array} \right\}
\end{aligned}$$

Table 2.2: The action of extraction operators on the Bézier element basis computed from the extended knot vector from Figure 2.15. The full operators would be computed in the case of NURBS, but only the highlighted rows for the T-spline basis function.

### 2.3.3.4 Multivariate extraction

Multivariate Bézier extraction operators  $\mathbf{C}^e$  for a T-spline element  $e$  are constructed as products of univariate Bézier extraction operators. In the bivariate case, T-spline basis function  $N_A$  restricted to element  $e$  can be decomposed into two univariate T-spline basis functions as

$$N_A(\xi_A)|_e = N_a^e(\tilde{\xi}) = N_a^{e,1}(\tilde{\xi}^1)N_a^{e,2}(\tilde{\xi}^2), \quad (2.49)$$

where  $a$  and  $e$  are such that  $A = \text{IEN}(a, e)$  and the superscripts indicate the local parametric direction. Using the extraction procedure outlined above for univariate T-spline basis functions, we can construct extraction operators for the basis functions  $N_a^{e,1}(\tilde{\xi}^1)$

and  $N_a^{e,2}(\tilde{\xi}^2)$ . By substituting equation (2.48) into (2.49) we obtain

$$N_a^e(\tilde{\xi}) = \left[ \mathbf{c}_a^{e,1^T} \mathbf{B}^1(\tilde{\xi}^1) \right] \left[ \mathbf{c}_a^{e,2^T} \mathbf{B}^2(\tilde{\xi}^2) \right]. \quad (2.50)$$

Using the index mapping from equation (2.38) it can be shown that this becomes

$$N_a^e(\tilde{\xi}) = \mathbf{c}_a^{e^T} \mathbf{B}(\tilde{\xi}), \quad (2.51)$$

where now  $\mathbf{c}_a^e$  corresponds to the  $a^{\text{th}}$  row of the bivariate element extraction operator  $\mathbf{C}^e$  in equation (2.48). This process is repeated for each T-spline basis function supported by element  $e$  to generate the full extraction operator for the element. Importantly, once the element extraction operators are computed, a finite element code never needs to know anything about T-mesh topology to process the global T-spline basis.

The process and result of extracting  $N_{40}(\xi_{40})$  over an element  $e$  in its support are shown in Figures 2.17 and 2.18, respectively. In Figure 2.17 the univariate components of the T-spline basis function are shown on the left and right. Note that in both cases a knot interior to a knot span of the extended knot vector must be inserted  $p$  times to compute the extraction operator for element  $e$ . In Figure 2.18 we show the resulting extraction coefficients. In Appendix C of Scott et al. [88], full extraction operators corresponding to elements 1, 10, 11 and 17 are presented.

### 2.3.3.5 A Bézier extraction algorithm for T-splines

A Bézier extraction algorithm for T-splines consists of the following steps:

1. Infer the T-spline basis from the T-mesh. See Section 2.2.2.
2. Refine the T-mesh to construct the elemental T-mesh. See Section 2.2.3.2.

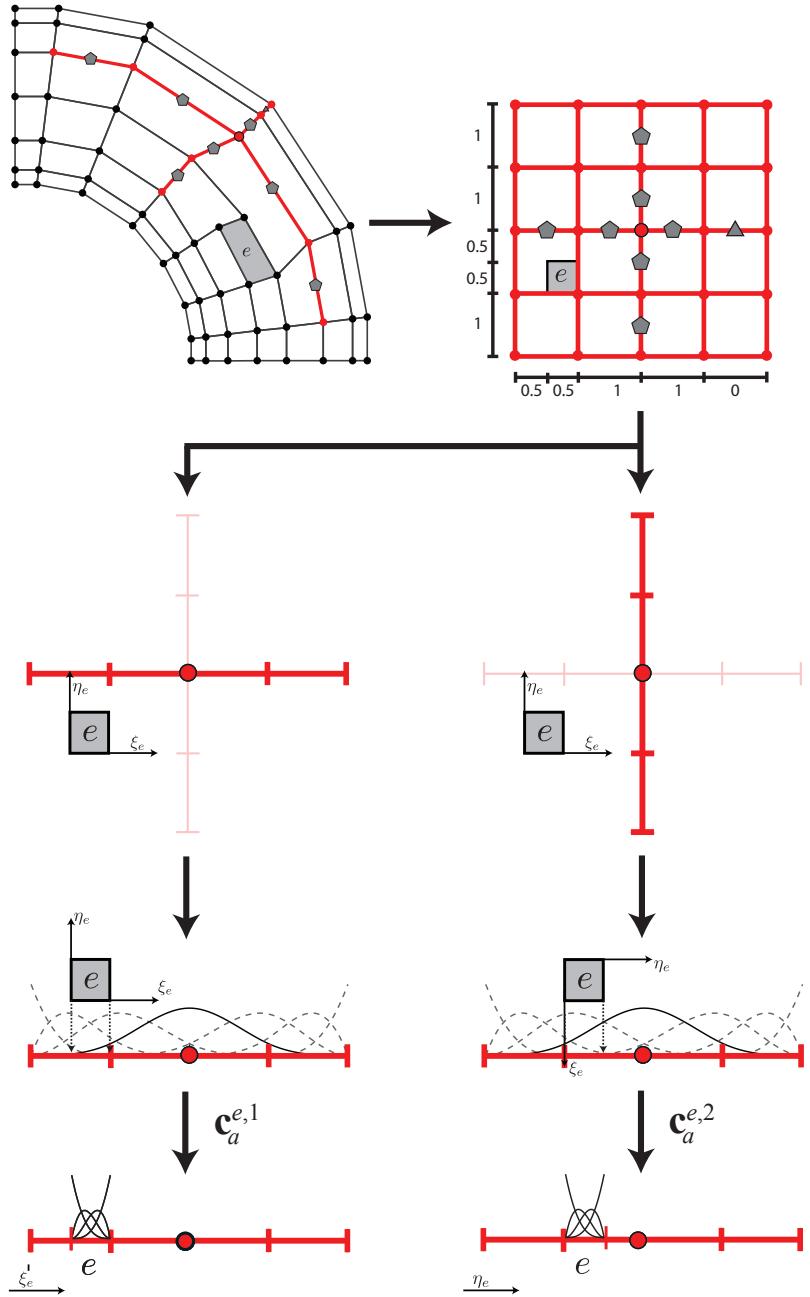


Figure 2.17: The extraction of a single T-spline basis function. Extraction is decoupled into two univariate extraction operations and then reassembled into a single row of  $\mathbf{C}^e$ . (The T-spline basis function is the one associated with control point 40, and element  $e$  corresponds to element 13 in the elemental T-mesh shown in Figure 2.7.)

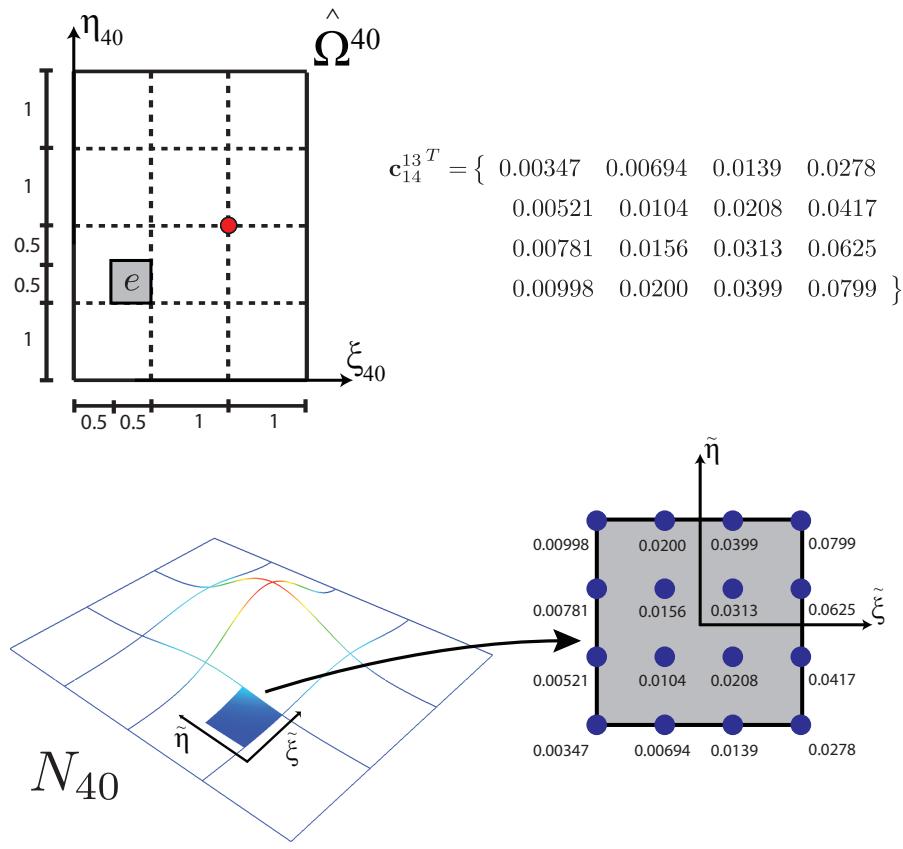


Figure 2.18: The result of extracting  $N_{40}(\xi_{40})$  over element  $e = 13$ . On the top left, the position of element 13 in the local basis function space of  $N_{40}(\xi_{40})$  is shown. On the top right, Bézier extraction of  $N_{40}$  over element  $e$  (see Figure 2.17) generates  $\mathbf{c}_{14}^{13 T}$  where 14 is the local index of global control point 40 for element 13. In other words,  $40 = \text{IEN}(14, 13)$ . Note that this represents a single row of the element extraction operator  $\mathbf{C}^{13}$ . On the bottom left,  $N_{40}(\xi_{40})$  is plotted. The shaded region is the restriction of  $N_{40}(\xi_{40})$  to the domain of element 13. On the bottom right, a close-up of element  $e$  is shown which portrays graphically the relationship between the extraction coefficients and Bernstein basis functions. Note that  $N_{40}(\xi_{40})|_{13} = \mathbf{c}_{14}^{13 T} \mathbf{B}(\tilde{\xi})$ . See equation (2.51).

3. For a T-spline basis function determine the Bézier elements which are in its support. See Section 2.3.3.
4. For a T-spline basis function perform Bézier extraction. See Sections 2.3.3.3 and 2.3.3.4.
5. Repeat steps 3 and 4 for each T-spline basis function.

We note that the topological characterization of a T-mesh described in Section 2.2.1 is analogous to a quadrilateral mesh with hanging nodes. Thus, traditional quadrilateral meshing data structures and refinement schemes that allow hanging nodes are compatible with T-splines and can be used to perform the first three steps of the algorithm.

The fourth step is the most critical and is not mesh dependent. The primary operation in this step is the univariate extraction of the extended local knot vectors for a T-spline basis function as described in Section 2.3.3.3. This routine is presented in Algorithm 2.2.

**Algorithm 2.2** An algorithm to compute the univariate extraction operator rows corresponding to a single univariate T-spline basis function.

```

input Knot vector,  $\Xi = \{\xi_1, \dots, \xi_{p+2}\}$ 
        Interior knots to be inserted into  $\Xi$ ,  $U = \{\bar{\xi}_1, \dots, \bar{\xi}_m\}$ 
        Knot spans in  $\Xi$  where new knots will be inserted,
        spans =  $\{\bar{s}_1, \dots, \bar{s}_m\}$ 
        Number of interior knots, m
        Curve degree, p
    
```

**output** Element extraction operator rows  $c^e$ ,  $e = 1, 2, \dots, p+1+m$

```
// Construct the extended knot vector and count
// the number of knots added to the front of the
// knot vector.
call compute_extended_knot_vector

input: Ξ
output: Ubar, nt

// Initialization variables:
a = p+1;
b = a+1;
nb = 1;
c1 = 0;
c1(nt+1) = 1;
mbar = p + 2 + nt + m;
ki = 1;
si = 1;

while b < mbar do
    // Initialize the next extraction operator row.
    cnb+1 = 0;

    // Count multiplicity of the knot at location b.
    add = 0;
    if si <= m && spans(si) = ki do
        mult = 0;
        add = 1;
    // Add the new knot to the knot vector.
```

```

Ubar(b+1:mbar+p-m+si) = Ubar(b:mbar+p-m+si-1);
Ubar(b) = U(si);
si = si + 1;

else
    ki = ki+1;
    i = b;
    while b < m && Ubar(b+1) == Ubar(b) do
        b = b+1;
    end
    mult = b-i+1;
end

if mult < p do
    numer = Ubar(b)-Ubar(a);
    for j = p,p-1,...,mult+1 do
        alphas(j-mult)
        = numer/ (Ubar(a+j+add)-Ubar(a));
    end
    r = p-mult;
    // Update the matrix coefficients for
    // r new knots
    for j=1,2,...,r do
        save = r-j+1;
        s = mult+j;
        for k=p+1,p,...,s+1 do
            alpha = alphas(k-s);
            cnb(k) = alpha*cnb(k) + (1.0-alpha)*cnb(k-1);
    end

```

```

if b < m do
    // Update overlapping coefficients of the
    // next operator row.
    cnb+1(save) = cnb(p+1);
end
end
// Finished with the current operator.
nb = nb + 1;
if b < m do
    // Update indices for the next operator.
    a = b;
    b = b+1;
end
end
end

```

#### 2.3.4 Bézier extraction and the finite element framework

Bézier extraction provides an element structure that can be incorporated into existing finite element frameworks. In the sections which follow we explore this application of Bézier extraction.

We assume that we have a geometric domain defined by 2.25 where  $\mathbf{P} = \{\mathbf{P}_A\}_{A=1}^n$  is a set of control points and  $\mathbf{R} = \{R_A\}_{A=1}^n$  is a basis for which the Bézier extraction operators,  $\mathbf{C}^e$ , and Bernstein basis,  $\mathbf{B} = \{B_A\}_{A=1}^m$ , can be computed.

### 2.3.4.1 Incorporating $\mathbf{C}^e$ into the finite element formulation

We now show how the element extraction operator,  $\mathbf{C}^e$ , can be incorporated into the finite element formulation. We begin with an abstract weak formulation. Letting  $\mathcal{S}$  be the trial solution space and  $\mathcal{V}$  be the space of weighting functions we have

$$(W) \left\{ \begin{array}{l} \text{Given } f, \text{ find } u \in \mathcal{S} \text{ such that for all } w \in \mathcal{V} \\ a(w, u) = (w, f) \end{array} \right. \quad (2.52)$$

where  $a(\cdot, \cdot)$  is a bilinear form and  $(\cdot, \cdot)$  is the  $L^2$  inner-product. Both  $\mathcal{S}$  and  $\mathcal{V}$  are assumed to be subspaces of the Sobolev space  $H^1$  (see [52] for further details). Galerkin's method consists of constructing finite-dimensional approximations of  $\mathcal{S}$  and  $\mathcal{V}$ . In an isogeometric setting we construct the finite-dimensional subspaces  $\mathcal{S}^h \subset \mathcal{S}$  and  $\mathcal{V}^h \subset \mathcal{V}$  from the basis which describes the geometry. The Galerkin formulation is then

$$(G) \left\{ \begin{array}{l} \text{Given } f, \text{ find } u^h \in \mathcal{S}^h \text{ such that for all } w^h \in \mathcal{V}^h \\ a(w^h, u^h) = (w^h, f) \end{array} \right. \quad (2.53)$$

In isogeometric analysis, the isoparametric concept is invoked, that is, the field in question is represented in terms of the geometric basis. We can write  $u^h$  and  $w^h$  as

$$w^h = \sum_{A=1}^n c_A R_A \quad (2.54)$$

$$u^h = \sum_{B=1}^m d_B R_B \quad (2.55)$$

where  $c_A$  and  $d_B$  are control variables. Substituting these into (2.53) yields the matrix form of the problem

$$\mathbf{K}\mathbf{d} = \mathbf{F} \quad (2.56)$$

where

$$\mathbf{K} = [K_{AB}], \quad (2.57)$$

$$\mathbf{F} = \{F_A\}, \quad (2.58)$$

$$\mathbf{d} = \{d_B\}, \quad (2.59)$$

$$K_{AB} = a(R_A, R_B), \quad (2.60)$$

$$F_A = (R_A, f). \quad (2.61)$$

The preceding formulation applies to scalar-valued partial differential equations, such as the heat conduction equation. The generalization to vector-valued partial differential equations, such as elasticity, follows standard procedures described in [52].

### 2.3.4.2 The element shape function routine

As in standard finite elements, the global stiffness matrix,  $\mathbf{K}$ , and force vector,  $\mathbf{F}$ , can be computed by performing integration over the Bézier elements to form the element stiffness matrices and force vectors— $\mathbf{k}^e$  and  $\mathbf{f}^e$  respectively—and assembling these into their global counterparts. The element form of (2.60) and (2.61) is

$$k_{ab}^e = a_e(R_a^e, R_b^e), \quad (2.62)$$

$$f_a^e = (R_a^e, f)_e \quad (2.63)$$

where  $a_e(\cdot, \cdot)$  denotes the bilinear form restricted to the element,  $(\cdot, \cdot)_e$  is the  $L^2$  inner-product restricted to the element, and  $R_a^e$  are the element shape functions. The integration is usually performed by Gaussian quadrature. As shown in Figure 2.19 for the two dimensional case, the integrals are pulled back, first onto the parametric element and then onto a bi-unit parent element. This requires the evaluation of the global basis

functions, their derivatives, and the Jacobian determinate of the pullback from the physical space to the parent element at each quadrature point in the parent element. These evaluations are done in an element shape function routine.

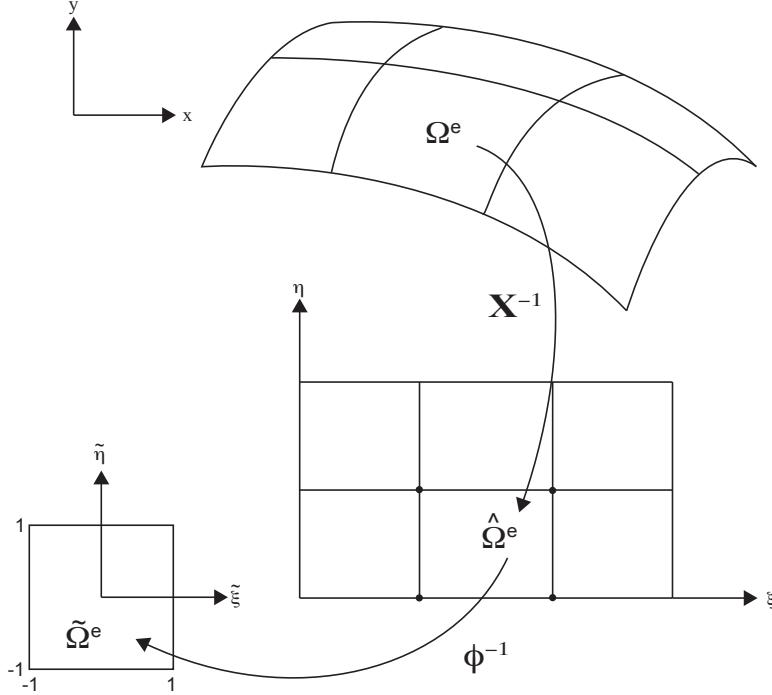


Figure 2.19: Integration is performed by Gaussian quadrature on each element. The physical element is pulled back first to the parametric domain through the geometrical mapping and then to the parent element through an affine mapping. For NURBS more efficient quadrature rules may be devised (see Hughes et al. [55]).

In order to perform the shape function routine evaluations we recall from (2.31) that

$$\mathbf{R}^e(\xi) = \mathbf{W}^e \mathbf{C}^e \frac{\mathbf{B}^e(\xi)}{W^b(\xi)}. \quad (2.64)$$

Thus, the derivatives of  $\mathbf{R}^e$  with respect to the parametric coordinates,  $\xi_i$ , are

$$\frac{\partial \mathbf{R}^e(\boldsymbol{\xi})}{\partial \xi_i} = \mathbf{W}^e \mathbf{C}^e \frac{\partial}{\partial \xi_i} \left( \frac{\mathbf{B}^e(\boldsymbol{\xi})}{W^b(\boldsymbol{\xi})} \right) = \mathbf{W}^e \mathbf{C}^e \left( \frac{1}{W^b(\boldsymbol{\xi})} \frac{\partial \mathbf{B}^e(\boldsymbol{\xi})}{\partial \xi_i} - \frac{\partial W^b(\boldsymbol{\xi})}{\partial \xi_i} \frac{\mathbf{B}^e(\boldsymbol{\xi})}{(W^b(\boldsymbol{\xi}))^2} \right). \quad (2.65)$$

To compute the derivatives with respect to the physical coordinates,  $(x_1, x_2, x_3)$ , we apply the chain rule to get

$$\frac{\partial \mathbf{R}^e(\boldsymbol{\xi})}{\partial x_i} = \sum_{j=1}^3 \frac{\partial \mathbf{R}^e(\boldsymbol{\xi})}{\partial \xi_j} \frac{\partial \xi_j}{\partial x_i}. \quad (2.66)$$

To compute  $\partial \boldsymbol{\xi} / \partial \mathbf{x}$  we first compute  $\partial \mathbf{x} / \partial \boldsymbol{\xi}$  using (2.25) and (2.65) and then take its inverse. Since we are integrating over the parent element we must also compute the Jacobian determinant of the mapping from the parent element to the physical space,  $J$ .

It is computed as

$$J = \left| \frac{\partial \mathbf{x}}{\partial \tilde{\boldsymbol{\xi}}} \right| = \left| \frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}} \frac{\partial \boldsymbol{\xi}}{\partial \tilde{\boldsymbol{\xi}}} \right|. \quad (2.67)$$

Higher-order derivatives can also be computed as described in [35]. An element shape function routine is presented in Appendix A of Borden et al. [22].

**Remark 2.3.** If  $\mathbf{R}^e$  and  $\mathbf{C}^e$  are computed as the tensor product of univariate components — as described in Section 2.3.2.1 for NURBS and Section 2.3.3.4 for T-splines — their tensor product structure can be exploited to reduce computational cost when computing the matrix products in (2.64) and (2.65). The following procedure can be used: (i) pre-compute the univariate Bernstein basis functions and derivatives at the prescribed quadrature points in each direction, (ii) at each call of the shape function routine, compute the univariate results for (2.64) and (2.65), and (iii) depending on dimension, use (2.43) or (2.44) to compute tensor product values of these results.

## Chapter 3

### Formulation

Consider an arbitrary body,  $\Omega_t$ , with an evolving internal discontinuity boundary,  $\Gamma_t$ , that represents a set of discrete cracks, as shown in Figure 3.1a. From a numerical standpoint, tracking  $\Gamma_t$  often requires complex and costly computations. This is particularly so when interactions between multiple cracks (even in two dimensions), or complex shaped cracks in three dimensions are considered. Also of interest in the case of dynamic fracture simulations, as considered in this work, is the ability to robustly model crack branching. In this work we pursue a formulation which is capable of handling cracks of arbitrary topological complexity. The underlying idea is to approximate  $\Gamma_t$  with a continuous scalar field,  $c$ , as shown in Figure 3.1b. As will be shown, with this approach the fracture problem can be reformulated as a system of partial differential equation that completely determine the evolution of the cracks. There are no ad hoc additions needed to determine crack nucleation, propagation, or bifurcation.

### 3.1 Kinematics

We let  $\Omega_0 \subset \mathbb{R}^d$  (with  $d \in \{1, 2, 3\}$ ) be the reference configuration of an arbitrary body with external boundary  $\partial\Omega_0$  and internal discontinuity boundary  $\Gamma_0$ . The current configuration at time  $t$  and its external boundary and internal discontinuity boundary are denoted as  $\Omega_t$ ,  $\partial\Omega_t$ , and  $\Gamma_t$  respectively (see Figure 3.1a). The time dependent deformation map  $\phi : (\Omega_0 \times \mathbb{R}) \rightarrow \Omega_t$  maps a point  $\mathbf{X} \in \Omega_0$  at time  $t \in [0, T]$  to a point

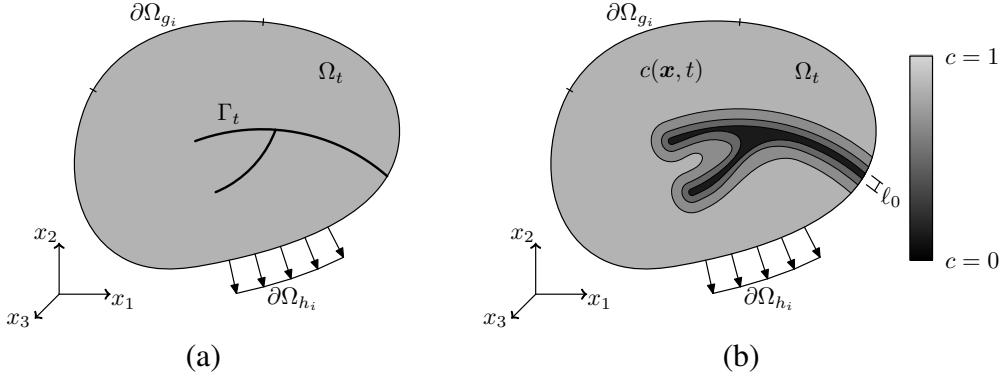


Figure 3.1: A schematic representation of (a) a solid body  $\Omega_t$  with an internal discontinuity boundary  $\Gamma_t$  and (b) an approximation of an internal discontinuity by a phase-field  $c(\mathbf{x}, t)$ . The model parameter  $\ell_0$  controls the width of the phase-field approximation.

$\mathbf{x} = \phi(\mathbf{X}, t) \in \Omega_t$ . The deformation gradient is a second-order tensor defined as

$$\mathbf{F} = \nabla^{\mathbf{X}} \phi(\mathbf{X}, t) = \frac{\partial \phi(\mathbf{X}, t)}{\partial \mathbf{X}} \quad (3.1)$$

where  $\nabla^{\mathbf{X}}$  is the gradient operator with respect to the reference coordinates. The displacement of a point  $\mathbf{X}$  at time  $t$  is denoted by  $\mathbf{u}(\mathbf{X}, t) = \phi(\mathbf{X}, t) - \mathbf{X}$ . The displacement field satisfies time-dependent Dirichlet boundary conditions,  $u_i(\mathbf{X}, t) = g_i(\mathbf{X}, t)$ , on  $\partial\Omega_{g_i} \subseteq \partial\Omega_t$ , and time-dependent Neumann boundary conditions on  $\partial\Omega_{h_i} \subseteq \partial\Omega_t$ .

The evolving internal discontinuity boundary,  $\Gamma_0(t)$ , represents a set of discrete cracks. In accordance with energetic approaches to fracture, the energy required to create a unit area of fracture surface is equal to the critical fracture energy density  $\mathcal{G}_c$ <sup>1</sup>.

The total potential energy of the body,  $\Psi$ , is then given by

$$\Psi(\mathbf{F}, \Gamma) = \int_{\Omega_0} \psi_e(\mathbf{F}) d\Omega_0 + \int_{\Gamma_0} \mathcal{G}_c d\Gamma_0. \quad (3.2)$$

where  $\psi_e$  is the elastic strain energy density function and the fracture energy contribution is merely the critical fracture energy density integrated over the fracture surface.

---

<sup>1</sup>This critical fracture energy density is commonly referred to as the critical energy release rate, or, in the context of cohesive zone models, the fracture toughness.

Irreversibility of the fracture process dictates that  $\Gamma_0(t) \subseteq \Gamma_0(t + \Delta t)$  for all  $\Delta t > 0$ . Hence, translation of cracks through the domain is prohibited, but cracks can extend, branch, and merge.

## 3.2 Phase-field approximations of the fracture surface

In order to circumvent the difficulty associated with numerically tracking a discontinuity as the crack path evolves, we approximate the discrete fracture surface by a phase-field,  $c(\mathbf{X}, t) \in [0, 1]$ , that approaches 1 away from the crack and is 0 inside the crack (see Figure 3.1b). The phase-field approximation introduces an  $n$ th order crack density functional,  $\Gamma_{c,n}$ , that is dependent only on a length scale parameter  $\ell_0$ , the phase-field  $c$ , and derivatives of  $c$  up to order  $n$  such that

$$\tilde{\Psi}_c = \int_{\Omega_0} \mathcal{G}_c \Gamma_{c,n} d\Omega_0 \approx \int_{\Gamma_0} \mathcal{G}_c d\Gamma_0. \quad (3.3)$$

The parameter  $\ell_0 \in \mathbb{R}^+$  determine the width of the smooth approximation of the crack. It can also be considered as a material property governing the critical value of stress required for nucleation and propagation of cracks (see Section 4.1).

### 3.2.1 Second-order phase-field theory

To approximate the crack with a continuous crack density functional, we follow a similar line of argument as presented by Miehe et al. [68]. We first introduce the second-order crack density functional

$$\Gamma_{c,2} = \frac{1}{4\ell_0} \left[ (c - 1)^2 + 4\ell_0^2 |\nabla^{\mathbf{X}} c|^2 \right], \quad (3.4)$$

which includes a gradient term for  $c$ . Minimizing  $\int \Gamma_{c,2} dx$  under the assumption that  $c(0) = 0$  and  $c'(\mathbf{X}) \rightarrow 0$  as  $|\mathbf{X}| \rightarrow \infty$  leads to the Euler equation

$$c - 1 - 4\ell_0^2 \Delta^{\mathbf{X}} c = 0. \quad (3.5)$$

The solution to this equation,  $c$ , is the solution to the minimization problem: find

$$\operatorname{argmin}_c (I(c)), \quad I(c) = \int \Gamma_{c,2} dx. \quad (3.6)$$

In one dimension the solution to this differential equation is

$$c(x) = 1 - e^{-|x|/2\ell_0}, \quad (3.7)$$

which is shown in Figure 3.2. In this case, it can be shown that (3.4) with (3.7) satisfies (3.3). In the general case, we use (3.4) to approximate the fracture energy as

$$\int_{\Gamma_0} \mathcal{G}_c d\Gamma_0 \approx \int_{\Omega_0} \mathcal{G}_c \left[ \frac{(1-c)^2}{4\ell_0} + \ell_0 |\nabla^{\mathbf{X}} c|^2 \right] d\Omega_0. \quad (3.8)$$

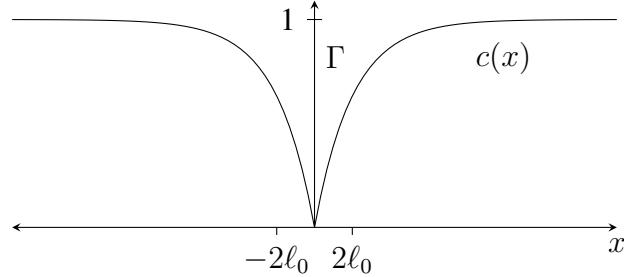


Figure 3.2: The one-dimensional phase-field approximation to the crack surface  $\Gamma$ .

### 3.2.2 Fourth-order phase-field theory

The smooth bases of isogeometric analysis provide a convenient computational framework in which to explore higher-order theories. In this section we introduce a fourth-order theory for the crack density functional.

We start by introducing the crack density functional with fourth-order derivatives of  $c$

$$\Gamma_{c,4} = \frac{1}{4\ell_0} [(c - 1)^2 + 2\ell_0^2 |\nabla^{\mathbf{X}} c|^2 + \ell_0^4 (\Delta^{\mathbf{X}} c)^2]. \quad (3.9)$$

Following the same ideas presented above we minimize  $\int \Gamma_{c,4} d\Omega_0$ , under the assumption that  $c(0) = c'(0) = 0$  and all derivatives of  $c$  go to zero as  $|\mathbf{X}|$  goes to  $\infty$ , to get the Euler equation

$$c(x) - 1 - 2\ell_0^2 \Delta c(x) + \ell_0^4 \Delta^2 c(x) = 0. \quad (3.10)$$

Solving this differential equation in one dimension by splitting it into two parts with the crack at  $x = 0$  and  $c(0) = 0$ ,  $c'(0) = 0$ , and  $c^{(\alpha)}(x) \rightarrow 0$  as  $x \rightarrow \pm\infty$  for all  $\alpha > 0$  we get

$$c(x) = 1 - e^{-|x|/\ell_0} \left( 1 + \frac{|x|}{\ell_0} \right) \quad (3.11)$$

(see Figure 3.3). Again, it can be shown that for one dimension (3.3) holds if (3.9) is substituted into the left-hand side with (3.11). In the general case, we use (3.9) to approximate the fracture energy as

$$\int_{\Gamma_0} \mathcal{G}_c d\Gamma_0 \approx \int_{\Omega_0} \mathcal{G}_c \left[ \frac{(1 - c)^2}{4\ell_0} + \frac{\ell_0}{2} |\nabla^{\mathbf{X}} c|^2 + \frac{\ell_0^3}{4} |\Delta^{\mathbf{X}} c|^2 \right] d\Omega_0. \quad (3.12)$$

### 3.2.3 Energy approximation

To model the loss of material stiffness in the crack we assume the elastic strain energy density function takes the general form

$$\psi_e(\mathbf{F}, c) = g(c)\psi_e^+(\mathbf{F}) + \psi_e^-(\mathbf{F}) \quad (3.13)$$

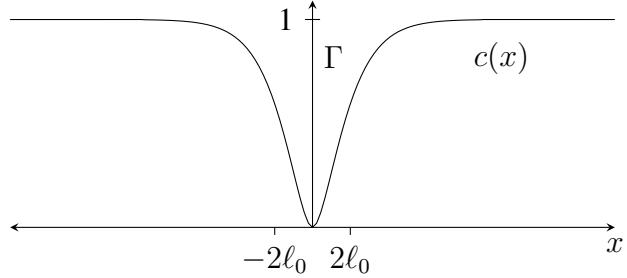


Figure 3.3: The smooth approximation to the crack surface  $\Gamma$  for the fourth-order theory.

where  $g(c)$  is the stress degradation function and the split between  $\psi_e^+$  and  $\psi_e^-$  provides a distinction between tensile and compressive energy. From this, the phase-field approximation of the potential energy becomes

$$\tilde{\Psi}_c(\mathbf{F}, c) = \int_{\Omega_0} g(c)\psi_e^+(\mathbf{F}) + \psi_e^-(\mathbf{F})d\mathbf{x} + \int_{\Omega_0} \mathcal{G}_c \Gamma_{c,n} d\mathbf{x}. \quad (3.14)$$

The Helmholtz free energy function is then

$$\psi = g(c)\psi_e^+ + \psi_e^- + \mathcal{G}_c \Gamma_{c,n}. \quad (3.15)$$

The stress degradation function,  $g(c)$ , is discussed in detail in Section 4.1.

**Remark 3.1.** By only applying the degradation function to the tensile part of the elastic energy density, we prohibit crack propagation under compression. In some cases, this feature has been observed to be important. This is particularly true for dynamic simulations, as compressive stress waves reflecting from domain boundaries tend to create physically unrealistic fracture patterns in the case where the degradation function is applied to the compressive energy.

### 3.3 Microforce derivation of the phase-field fracture model

In this section we introduce a microforce balance theory that supplements the usual momentum balance equations with a microforce balance law that accounts for the

kinematics of the phase-field. Microforce theories have been applied to other phase-field models. For example, Gurtin [50] has derived the Cahn-Hilliard equations from a microforce balance law. In addition, Stumpf and Hackl [98] introduced a microforce derivation for damage laws that provides a thermodynamically consistent framework for thermo-viscoelastic and quasi-brittle materials. The phase-field model can be considered as a damage law and we follow this derivation, but we restrict ourselves to isothermal processes.

We begin the derivation of the phase-field model with the usual linear and angular momentum balance laws:

**Linear momentum:**

$$\frac{D}{Dt} \int_{\Omega_0} \rho_0 \dot{\mathbf{U}} d\Omega_0 = \int_{\Omega_0} \mathbf{B} d\Omega_0 + \int_{\partial\Omega_0} \mathbf{P} \cdot \mathbf{N} d\partial\Omega_0 \quad (3.16)$$

$$\Rightarrow P_{iJ,J} + B_i = \rho_0 \ddot{U}_i \quad (3.17)$$

**Angular momentum:**

$$\frac{D}{Dt} \int_{\Omega_0} \mathbf{x} \times \rho_0 \dot{\mathbf{U}} d\Omega_0 = \int_{\Omega_0} \mathbf{x} \times \mathbf{B} d\Omega_0 + \int_{\partial\Omega_0} \mathbf{x} \times (\mathbf{P} \cdot \mathbf{N}) d\partial\Omega_0 \quad (3.18)$$

$$\Rightarrow P_{iJ}F_{jJ} = P_{jJ}F_{iJ} \quad (3.19)$$

where  $\mathbf{P}$  is the first Piola-Kirchhoff stress tensor and  $\mathbf{N}$  is the outward unit normal of  $\partial\Omega_0$ ,  $\rho_0$  the mass density,  $\mathbf{B}$  a body force, and  $\mathbf{U}$  the displacement.

To derive the microforce balance law, we assume that at each time step the phase-field can be characterized by an internal microforce  $\pi(\mathbf{X}, t) \in \mathbb{R}$ , an external microforce  $l(\mathbf{X}, t) \in \mathbb{R}$ , and a microforce  $\lambda(\mathbf{X}, t) \in \mathbb{R}$  acting on the surface of the body. Furthermore, we assume that there are no inertial terms associated with the phase-field. Given these assumptions, we postulate the microforce balance law to be

**Microforce:**

$$\int_{\partial\Omega_0} \lambda d\Omega_0 + \int_{\Omega_0} l d\Omega_0 + \int_{\Omega_0} \pi d\Omega_0 = 0 \quad (3.20)$$

$$\lambda = \xi_I N_I \quad \text{on} \quad \partial\Omega_0 \quad (3.21)$$

$$\Rightarrow \xi_{I,I} + \pi + l = 0 \quad (3.22)$$

where  $\xi_I$  is a microforce traction vector acting on  $\partial\Omega_0$ .

The formulation of the phase-field fracture model must be thermodynamically consistent. The thermodynamics are outlined as follows:

**Energy Balance** Letting  $e_0$  denote the internal energy per unit reference volume and using (3.17), energy balance can be written as

$$\frac{D}{Dt} \int_{\Omega_0} \left( \frac{1}{2} \rho_0 \dot{U}_i \dot{U}_i + e_0 \right) d\Omega_0 \quad (3.23)$$

$$= \int_{\partial\Omega_0} P_{iJ} N_J \dot{U}_i d\partial\Omega_0 + \int_{\Omega_0} B_i \dot{U}_i d\Omega_0 + \int_{\partial\Omega_0} \lambda \dot{c} d\partial\Omega_0 + \int_{\Omega_0} l \dot{c} d\Omega_0 \quad (3.24)$$

$$\Rightarrow \int_{\Omega_0} \dot{e}_0 d\Omega_0 = \int_{\Omega_0} P_{iJ} \dot{U}_{i,J} d\Omega_0 + \int_{\partial\Omega_0} \lambda \dot{c} d\partial\Omega_0 + \int_{\Omega_0} l \dot{c} d\Omega_0 \quad (3.25)$$

where we have used the fact that  $\dot{c}$  is work conjugate to the microforces.

**Second law** Let  $s_0$  be entropy per unit reference volume. Then the second law of thermodynamics can be stated as

$$\frac{D}{Dt} \int_{\Omega_0} s_0 d\Omega_0 \geq - \int_{\partial\Omega_0} \frac{Q_I N_i}{\Theta} d\partial\Omega_0 + \int_{\Omega_0} \frac{r_0}{\Theta} d\Omega_0 \quad (3.26)$$

$$\Rightarrow \Theta \dot{s}_0 \geq \frac{1}{\Theta} Q_I \Theta_{,I} - Q_{I,I} + r_0 \quad (3.27)$$

where  $\Theta$  is absolute temperature,  $Q$  is the heat flux, and  $r_0$  is a heat source. For isothermal and adiabatic processes (the case considered in this dissertation) this simplifies to

$$\Theta \dot{s}_0 \geq 0. \quad (3.28)$$

**Dissipation inequality** Letting  $\phi$  be the Helmholtz free energy the Legendre transformation states

$$e_0 = \psi + \Theta s_0. \quad (3.29)$$

It follows from the second law of thermodynamics that the dissipation inequality is given by

$$\Rightarrow \dot{\psi} \leq \dot{e}_0. \quad (3.30)$$

Substituting (3.21) into the microforce surface term of (3.25), applying the divergence theorem, integrating by parts, and using (3.22), this inequality leads to

$$\begin{aligned} \int_{\Omega_0} \dot{\psi} d\Omega_0 &\leq \int_{\Omega_0} \left( P_{IJ} \dot{F}_{IJ} + \xi_I \dot{c}_{,I} - \pi \dot{c} \right) d\Omega_0 \\ &= \int_{\Omega_0} \left( \frac{1}{2} S_{IJ} \dot{C}_{IJ} + \xi_I \dot{c}_{,I} - \pi \dot{c} \right) d\Omega_0 \end{aligned} \quad (3.31)$$

where  $\mathbf{S} = \mathbf{F}^{-1} \mathbf{P}$  is the second Piola-Kirchhoff stress tensor and  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  is the right Cauchy-Green deformation tensor.

This derivation provides a general formulation for phase-field fracture models. As long as the free energy,  $\psi$ , is defined such that (3.31) holds the model will be thermodynamically consistent. In what follows we derive several models that satisfy this condition.

### 3.4 Small strain brittle fracture

We begin by deriving a small strain formulation for brittle fracture. We assume small deformations and deformation gradients, and define the infinitesimal strain tensor,  $\boldsymbol{\varepsilon}(\mathbf{x}, t) \in \mathbb{R}^{d \times d}$ , with components

$$\varepsilon_{ij} = u_{(i,j)} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (3.32)$$

as an appropriate deformation measure. We assume isotropic linear elasticity, such that the undamaged elastic energy density is given by

$$\psi_e(\boldsymbol{\varepsilon}) = \frac{1}{2} \lambda \varepsilon_{ii} \varepsilon_{jj} + \mu \varepsilon_{ij} \varepsilon_{ij} \quad (3.33)$$

with  $\lambda$  and  $\mu$  the Lamé constants. Note that we use the Einstein summation convention on repeated indices.

To model a tensile/compressive split of the strain energy we follow Miehe, Hockecker, and Welschinger [68] and define the elastic energy as

$$\psi_e(\boldsymbol{\varepsilon}, c) = g(c) \psi_e^+(\boldsymbol{\varepsilon}) + \psi_e^-(\boldsymbol{\varepsilon}). \quad (3.34)$$

where  $\psi_e^+$  and  $\psi_e^-$  are the strain energies computed from the positive and negative components of the strain tensor, respectively, defined through a spectral decomposition of strain. Let

$$\boldsymbol{\varepsilon} = \mathbf{P} \boldsymbol{\Lambda} \mathbf{P}^T \quad (3.35)$$

where  $\mathbf{P}$  consists of the orthonormal eigenvectors of  $\boldsymbol{\varepsilon}$  and  $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$  is a diagonal matrix of principal strains. We define

$$\boldsymbol{\varepsilon}^+ = \mathbf{P} \boldsymbol{\Lambda}^+ \mathbf{P}^T \quad (3.36)$$

$$\boldsymbol{\varepsilon}^- = \mathbf{P} \boldsymbol{\Lambda}^- \mathbf{P}^T \quad (3.37)$$

where

$$\boldsymbol{\Lambda}^+ = \text{diag}(\langle \lambda_1 \rangle, \langle \lambda_2 \rangle, \langle \lambda_3 \rangle) \quad (3.38)$$

$$\boldsymbol{\Lambda}^- = \boldsymbol{\Lambda} - \boldsymbol{\Lambda}^+ \quad (3.39)$$

and

$$\langle x \rangle = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}. \quad (3.40)$$

Then

$$\psi_e^+(\boldsymbol{\varepsilon}) = \frac{1}{2}\lambda\langle \text{tr } \boldsymbol{\varepsilon} \rangle^2 + \mu \text{tr}[(\boldsymbol{\varepsilon}^+)^2] \quad (3.41)$$

and

$$\psi_e^-(\boldsymbol{\varepsilon}) = \frac{1}{2}\lambda(\text{tr } \boldsymbol{\varepsilon} - \langle \text{tr } \boldsymbol{\varepsilon} \rangle)^2 + \mu \text{tr}[(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^+)^2]. \quad (3.42)$$

This decomposition is somewhat different from the one used in Amor, Marigo, and Maurini [5]. In our notation, the model used by Amor, Marigo, and Maurini [5] would be

$$\psi_e(\boldsymbol{\varepsilon}, c) = \frac{\kappa}{2}(\text{tr } \boldsymbol{\varepsilon} - \langle \text{tr } \boldsymbol{\varepsilon} \rangle)^2 + (c^2 + k) \left( \frac{\kappa}{2}\langle \text{tr } \boldsymbol{\varepsilon} \rangle^2 + \mu \text{tr}[(\boldsymbol{\varepsilon}^{\text{dev}})^2] \right) \quad (3.43)$$

where  $\kappa$  is the bulk modulus and  $\boldsymbol{\varepsilon}^{\text{dev}}$  is the deviatoric strain tensor. In the Amor, Marigo, and Maurini [5] model, expansive volumetric strain energy, as measured by  $\langle \text{tr } \boldsymbol{\varepsilon} \rangle$ , and deviatoric strain energy, are subjected to damage, but not compressive volumetric strain energy. We note a difference of this model compared with the model we use is that, as is evident from (3.43), all three principal strains could be negative, but the deviatoric strain energy would experience damage, unlike the model we use. However, the intent of both models is similar, that is, to maintain resistance in compression and, in particular, during crack closure. Del Piero, Lancioni, and March [40] provide a large deformation definition of strain energy, but they do not decompose the strain energy into positive and negative parts to maintain resistance to compression. Rather, they rely on residual elasticity provided by the parameter  $k$  and the fact that as the Jacobian determinant goes to zero, the strain energy approaches positive infinity.

With the strain energy density defined above for the small strain case the Helmholtz free energy function becomes

$$\psi = g(c)\psi_e^+ + \psi_e^- + \mathcal{G}_c\Gamma_{c,n}. \quad (3.44)$$

Under the small strain assumption, equation (3.31) becomes

$$\int_{\Omega} \dot{\psi} d\Omega \leq \int_{\Omega} (\sigma_{ij}\dot{\varepsilon}_{ij} + \xi_i\dot{c}_{,i} - \pi\dot{c}) d\Omega. \quad (3.45)$$

### 3.4.1 Second-order formulation

For the second-order phase-field formulation we substitute  $\Gamma_{c,2}$  into (3.44) to get

$$\psi = g(c)\psi_e^+ + \psi_e^- + \mathcal{G}_c \left[ \frac{(1-c)^2}{4\ell_0} + \ell_0|\nabla c|^2 \right]. \quad (3.46)$$

Substituting this into the dissipation inequality (3.31) and assuming that  $\psi$  may depend on  $\dot{c}$  we get

$$\begin{aligned} \int_{\Omega} \dot{\psi} d\Omega &= \int_{\Omega} \left( \frac{\partial\psi}{\partial\varepsilon_{ij}}\dot{\varepsilon}_{ij} + \frac{\partial\psi}{\partial c}\dot{c} + \frac{\partial\psi}{\partial c_{,i}}\dot{c}_{,i} + \frac{\partial\psi}{\partial\ddot{c}}\ddot{c} \right) d\Omega \\ &\leq \int_{\Omega} (\sigma_{ij}\dot{\varepsilon}_{ij} + \xi_i\dot{c}_{,i} - \pi\dot{c}) d\Omega. \end{aligned} \quad (3.47)$$

This inequality must hold for arbitrary variations of  $\dot{\varepsilon}$ ,  $\dot{c}$ ,  $\dot{c}_{,i}$ , and  $\ddot{c}$ . Assuming that  $\pi$  can depend linearly on  $\dot{c}$  to account for dissipation, this leads to

$$\frac{\partial\psi}{\partial\dot{c}} = 0 \quad (3.48)$$

$$\frac{\partial\psi}{\partial\varepsilon_{ij}} = \sigma_{ij} \quad (3.49)$$

$$\frac{\partial\psi}{\partial c_{,i}} = \xi_i \quad (3.50)$$

$$\pi + \frac{\partial\psi}{\partial c} \leq 0 \Rightarrow \pi = -\frac{\partial\psi}{\partial c} - \beta\dot{c} \quad (3.51)$$

where  $\beta = \beta(\varepsilon, c, c_{,i}) \geq 0$  is a dissipation constant. Substituting these results into the balance laws, we get

**Linear momentum:**

$$\left( \frac{\partial \psi}{\partial \varepsilon_{ij}} \right)_{,j} + b_i = \rho \ddot{u}_i \quad (3.52)$$

**Microforce:**

$$\left( \frac{\partial \psi}{\partial c_{,i}} \right)_{,i} + l - \frac{\partial \psi}{\partial c} = \beta \dot{c} \quad (3.53)$$

Now, assuming  $l = 0$  and  $\beta = 0$  we get, after some manipulations, the strong form of the second-order formulation to be

$$(S) \begin{cases} \sigma_{ij,j} + b_i = \rho \ddot{u}_i & \text{on } \Omega \times ]0, T[ \\ \frac{2\ell_0 g'(c) \psi_e^+}{\mathcal{G}_c} + c - 4\ell_0^2 c_{,ii} = 1 & \text{on } \Omega \times ]0, T[. \end{cases} \quad (3.54)$$

where

$$\sigma_{ij} = g(c) \frac{\partial \psi_e^+}{\partial \varepsilon_{ij}} + \frac{\partial \psi_e^-}{\partial \varepsilon_{ij}}. \quad (3.55)$$

These equations of motion can be solved to find both the displacement field  $\mathbf{u}(\mathbf{x}, t)$  and phase-field  $c(\mathbf{x}, t)$ .

The equations of motion are subject to the boundary conditions

$$(S: BC) \begin{cases} u_i = g_i & \text{on } \partial\Omega_{g_i} \times ]0, T[ \\ \sigma_{ij} n_j = h_i & \text{on } \partial\Omega_{h_i} \times ]0, T[ \\ c_{,i} n_i = 0 & \text{on } \partial\Omega \times ]0, T[ \end{cases} \quad (3.56)$$

with  $g_i(\mathbf{x}, t)$  and  $h_i(\mathbf{x}, t)$  being prescribed on  $\partial\Omega_{g_i}$  and  $\partial\Omega_{h_i}$ , respectively, for all  $t \in ]0, T[$ , and with  $\mathbf{n}(\mathbf{x})$  being the outward-pointing normal vector of the boundary. The

boundary condition on the phase-field is the homogeneous natural boundary condition that arises from the derivation of the weak form.

In addition, the equations of motion (3.54) are supplemented with initial conditions

$$(S: IC) \left\{ \begin{array}{ll} \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) & \mathbf{x} \in \Omega \\ \dot{\mathbf{u}}(\mathbf{x}, 0) = \mathbf{v}_0(\mathbf{x}) & \mathbf{x} \in \Omega \end{array} \right. \quad (3.57)$$

for both the displacement field and the phase-field.

**Remark 3.2.** At this point in our formulation we have not addressed the issue of irreversibility. There is nothing in the above formulation to prevent cracks from healing if loads are removed (the formulation is, none the less, thermodynamically consistent). This issue will be addressed in detail in Section 3.6.

**Remark 3.3.** By letting  $\beta = 0$  we have assumed that there are no viscous effects acting on the evolution of the phase-field. Miehe et al. [68] include this term and study its effect. In the quasi-static case it can improve numerical stability, but we have found no need for it in the dynamic case.

### 3.4.1.1 Weak and semidiscrete Galerkin form

The numerical solution of (3.54) requires a spatial and temporal discretization. In this section we formulate the spatial discretization for the second order theory by means of the Galerkin method.

For the weak form of the problem we define the trial solution spaces  $\mathcal{S}_t$  for the

displacements and  $\tilde{\mathcal{S}}_t$  for the phase-field as

$$\mathcal{S}_t = \{\mathbf{u}(t) \in (H^1(\Omega))^3 \mid u_i(t) = g_i \text{ on } \partial\Omega_{g_i}\} \quad (3.58)$$

$$\tilde{\mathcal{S}}_t = \{c(t) \in H^1(\Omega)\}. \quad (3.59)$$

Similarly, the weighting function spaces are defined as

$$\mathcal{V} = \{\mathbf{w} \in (H^1(\Omega))^3 \mid w_i = 0 \text{ on } \partial\Omega_{g_i}\} \quad (3.60)$$

$$\tilde{\mathcal{V}} = \{q \in H^1(\Omega)\}. \quad (3.61)$$

Multiplying the equations in (3.54) by the appropriate variational functions and applying integration by parts leads to the weak form:

$$(W) \left\{ \begin{array}{l} \text{Given } \mathbf{g}, \mathbf{h}, \mathbf{u}_0, \dot{\mathbf{u}}_0, \text{ and } c_0 \text{ find } \mathbf{u}(t) \in \mathcal{S}_t \text{ and } c(t) \in \tilde{\mathcal{S}}_t, t \in [0, T], \text{ such that} \\ \text{for all } \mathbf{w} \in \mathcal{V} \text{ and for all } q \in \tilde{\mathcal{V}}, \\ (\rho \ddot{\mathbf{u}}, \mathbf{w})_\Omega + (\boldsymbol{\sigma}, \nabla \mathbf{w})_\Omega = (\mathbf{h}, \mathbf{w})_{\partial\Omega_h} \\ \left( \frac{2\ell_0 g'(c) \psi_e^+}{\mathcal{G}_c} + c, q \right)_\Omega + (4\ell_0^2 \nabla c, \nabla q)_\Omega = (1, q)_\Omega \\ (\rho \mathbf{u}(0), \mathbf{w})_\Omega = (\rho \mathbf{u}_0, \mathbf{w})_\Omega \\ (\rho \dot{\mathbf{u}}(0), \mathbf{w})_\Omega = (\rho \dot{\mathbf{u}}_0, \mathbf{w})_\Omega \end{array} \right. \quad (3.62)$$

where  $(\bullet, \bullet)_\Omega$  is the  $\mathcal{L}_2$  inner product on  $\Omega$ .

Following the Galerkin method, we let  $\mathcal{S}_t^h \subset \mathcal{S}_t$ ,  $\mathcal{V}^h \subset \mathcal{V}$ ,  $\tilde{\mathcal{S}}_t^h \subset \tilde{\mathcal{S}}_t$ , and  $\tilde{\mathcal{V}}^h \subset \tilde{\mathcal{V}}$  be the usual finite-dimensional approximations to the function spaces of the weak form (see Hughes [52] for details). The simidiscrete Galerkin form of the problem

is then given as

$$(G) \left\{ \begin{array}{l} \text{Given } \mathbf{g}, \mathbf{h}, \mathbf{u}_0, \dot{\mathbf{u}}_0, \text{ and } c_0 \text{ find } \mathbf{u}^h(t) \in \mathcal{S}_t^h \text{ and } c^h(t) \in \tilde{\mathcal{S}}_t^h, t \in [0, T], \text{ such that} \\ \text{for all } \mathbf{w}^h \in \mathcal{V}^h \text{ and for all } q^h \in \tilde{\mathcal{V}}^h, \\ (\rho \ddot{\mathbf{u}}^h, \mathbf{w}^h)_\Omega + (\boldsymbol{\sigma}, \nabla \mathbf{w}^h)_\Omega = (\mathbf{h}, \mathbf{w}^h)_{\partial\Omega_h} \\ \left( \frac{2\ell_0 g'(c^h) \psi_e^+}{\mathcal{G}_c} + c^h, q^h \right)_\Omega + (4\ell_0^2 \nabla c^h, \nabla q^h)_\Omega = (1, q^h)_\Omega \\ (\rho \mathbf{u}^h(0), \mathbf{w}^h)_\Omega = (\rho \mathbf{u}_0, \mathbf{w}^h)_\Omega \\ (\rho \dot{\mathbf{u}}^h(0), \mathbf{w}^h)_\Omega = (\rho \dot{\mathbf{u}}_0, \mathbf{w}^h)_\Omega \end{array} \right. \quad (3.63)$$

The explicit representations of  $\mathbf{u}^h$ ,  $\mathbf{w}^h$ ,  $c^h$ , and  $q^h$  in terms of the basis functions and control variables values are

$$u_i^h = \sum_A^{n_b} N_A(x) d_{iA} \quad (3.64)$$

$$w_i^h = \sum_A^{n_b} N_A(x) c_{iA} \quad (3.65)$$

$$c^h = \sum_A^{n_b} N_A(x) \phi_A \quad (3.66)$$

$$q^h = \sum_A^{n_b} N_A(x) \chi_A \quad (3.67)$$

where  $n_b$  is the dimension of the discrete space, the  $N_A$ 's are the global basis functions,  $i$  is the spatial degree-of-freedom number, and  $d_{iA}$ ,  $c_{iA}$ ,  $\phi_A$ , and  $\chi_A$  are control variable degrees-of-freedom. Note that we have assumed that both the trial solution and variational finite dimensional function spaces are defined over the same set of basis functions.

### 3.4.2 Fourth-order formulation

For the fourth-order phase-field formulation we substitute  $\Gamma_{c,4}$  into (3.44) to get

$$\psi = g(c)\psi_e^+ + \psi_e^- + \mathcal{G}_c \left[ \frac{(1-c)^2}{4\ell_0} + \frac{\ell_0}{2} |\nabla c|^2 + \frac{\ell_0^3}{4} (\Delta c)^2 \right] \quad (3.68)$$

Substituting this into the dissipation inequality and again assuming  $\psi$  may depend on  $\dot{c}$  we get

$$\begin{aligned} \int_{\Omega} \dot{\psi} d\Omega &= \int_{\Omega} \left( \frac{\partial \psi}{\partial \varepsilon_{ij}} \dot{\varepsilon}_{ij} + \frac{\partial \psi}{\partial c} \dot{c} + \frac{\partial \psi}{\partial c_{,i}} \dot{c}_{,i} + \frac{\partial \psi}{\partial c_{,jj}} \dot{c}_{,ii} + \frac{\partial \psi}{\partial \dot{c}} \ddot{c} \right) d\Omega \\ &\leq \int_{\Omega} (\sigma_{ij} \dot{\varepsilon}_{ij} + \xi_i \dot{c}_{,i} - \pi \dot{c}) d\Omega. \end{aligned} \quad (3.69)$$

To handle the term involving the Laplacian of  $c$  we use the divergence theorem to get

$$\begin{aligned} \int_{\Omega} \frac{\partial \psi}{\partial c_{,jj}} \dot{c}_{,ii} d\Omega &= \int_{\Omega} \left[ \left( \frac{\partial \psi}{\partial c_{,jj}} \dot{c}_{,i} \right)_{,i} - \left( \frac{\partial \psi}{\partial c_{,jj}} \right)_{,i} \dot{c}_{,i} \right] d\Omega \\ &= \int_{\partial\Omega} \frac{\partial \psi}{\partial c_{,jj}} \dot{c}_{,i} n_i d\partial\Omega - \int_{\Omega} \left( \frac{\partial \psi}{\partial c_{,jj}} \right)_{,i} \dot{c}_{,i} d\Omega. \end{aligned} \quad (3.70)$$

Following similar arguments as for the second-order case, and noting that the normal component  $\dot{c}_{,n} = \dot{c}_{,i} n_i$  can be varied arbitrarily on  $\partial\Omega$ , this leads to

$$\frac{\partial \psi}{\partial \dot{c}} = 0 \quad (3.71)$$

$$\frac{\partial \psi}{\partial \varepsilon_{ij}} = \sigma_{ij} \quad (3.72)$$

$$\frac{\partial \psi}{\partial c_{,i}} - \left( \frac{\partial \psi}{\partial c_{,jj}} \right)_{,i} = \xi_i \quad (3.73)$$

$$\frac{\partial \psi}{\partial c_{,jj}} = 0 \quad \text{on } \partial\Omega \quad (3.74)$$

$$\pi + \frac{\partial \psi}{\partial c} \leq 0 \Rightarrow \pi = -\frac{\partial \psi}{\partial c} - \beta \dot{c} \quad (3.75)$$

where  $\beta = \beta(\varepsilon, c, c_{,i}, c_{,ii}) \geq 0$  is again a dissipation constant. The linear momentum balance law is unchanged from the second-order formulation. The microforce balance

law becomes

$$\left( \frac{\partial \psi}{\partial c_{,i}} \right)_{,i} - \left( \frac{\partial \psi}{\partial c_{,jj}} \right)_{,ii} + l - \frac{\partial \psi}{\partial c} = \beta \dot{c} \quad (3.76)$$

Assuming  $l = 0$  and  $\beta = 0$  and defining the Cauchy stress tensor the same as for the second-order formulation we get, after some manipulations, the strong form to be

$$(S) \left\{ \begin{array}{ll} \sigma_{ij,j} = \rho \ddot{u}_i & \text{on } \Omega \times ]0, T[ \\ \frac{2\ell_0 g'(c) \psi_e^+}{\mathcal{G}_c} + c - 2\ell_0^2 c_{,ii} + \ell_0^4 c_{,iijj} = 1 & \text{on } \Omega \times ]0, T[ \\ u_i = g_i & \text{on } \partial\Omega_{g_i} \times ]0, T[ \\ \sigma_{ij} n_j = h_i & \text{on } \partial\Omega_{h_i} \times ]0, T[ \\ c_{,ii} = 0 & \text{on } \partial\Omega \times ]0, T[ \\ (\ell_0^4 c_{,iij} - 2\ell_0^2 c_{,j}) n_j = 0 & \text{on } \partial\Omega \times ]0, T[ \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) & \mathbf{x} \in \Omega \\ \dot{\mathbf{u}}(\mathbf{x}, 0) = \mathbf{v}_0(\mathbf{x}) & \mathbf{x} \in \Omega \end{array} \right. \quad (3.77)$$

The additional boundary condition on the phase-field is a homogeneous natural boundary that arises from the derivation of the weak form.

### 3.4.2.1 Weak form

For the weak form of the problem we define the trial solution spaces  $\mathcal{S}_t$  for the displacements and  $\tilde{\mathcal{S}}_t$  for the phase-field as

$$\mathcal{S}_t = \{ \mathbf{u}(t) \in (H^1(\Omega))^3 \mid u_i(t) = g_i \text{ on } \partial\Omega_{g_i} \} \quad (3.78)$$

$$\tilde{\mathcal{S}}_t = \{ c(t) \in H^2(\Omega) \}. \quad (3.79)$$

Similarly, the weighting function spaces are defined as

$$\mathcal{V} = \{ \mathbf{w} \in (H^1(\Omega))^3 \mid w_i = 0 \text{ on } \partial\Omega_{g_i} \} \quad (3.80)$$

$$\tilde{\mathcal{V}} = \{ q \in H^2(\Omega) \}. \quad (3.81)$$

Multiplying the equations in (3.77) by the appropriate variational functions and applying integration by parts leads to the weak form:

$$(W) \left\{ \begin{array}{l} \text{Given } \mathbf{g}, \mathbf{h}, \mathbf{u}_0, \dot{\mathbf{u}}_0, \text{ and } c_0 \text{ find } \mathbf{u}(t) \in \mathcal{S}_t \text{ and } c(t) \in \tilde{\mathcal{S}}_t, t \in [0, T], \text{ such that} \\ \text{for all } \mathbf{w} \in \mathcal{V} \text{ and for all } q \in \tilde{\mathcal{V}}, \\ (\rho \ddot{\mathbf{u}}, \mathbf{w})_{\Omega} + (\boldsymbol{\sigma}, \nabla \mathbf{w})_{\Omega} = (\mathbf{h}, \mathbf{w})_{\partial \Omega_h} \\ \left( \frac{2\ell_0 g'(c) \psi_e^+}{\mathcal{G}_c} + c, q \right)_{\Omega} + (2\ell_0^2 \nabla c, \nabla q)_{\Omega} + (\ell_0^4 \Delta c, \Delta q)_{\Omega} = (1, q)_{\Omega} \\ (\rho \mathbf{u}(0), \mathbf{w})_{\Omega} = (\rho \mathbf{u}_0, \mathbf{w})_{\Omega} \\ (\rho \dot{\mathbf{u}}(0), \mathbf{w})_{\Omega} = (\rho \dot{\mathbf{u}}_0, \mathbf{w})_{\Omega} \end{array} \right. \quad (3.82)$$

where  $(\bullet, \bullet)_{\Omega}$  is the  $\mathcal{L}_2$  inner product on  $\Omega$ . The Galerkin form follows in the same way as for the second-order theory.

**Remark 3.4.** Technically, the function spaces for the displacements are not required to be as smooth as the spaces for the fourth-order phase-field. In practice, however, we typically use the same spaces for both the displacement and phase-field.

### 3.5 Large deformation ductile fracture

A major contribution of this dissertation is an initial formulation of a phase-field model for ductile fracture. Ductile behavior occurs in many materials such as, for example, low-strength steel and copper. Failure of these materials is of interest in many engineering structures. For example, the behavior of ship and submarine hulls subject to explosive loads is of great interest to the Navy.

We restrict the development of our formulation to the second-order phase-field theory. The extension to higher-order models is straightforward and follows the same developments as the small strain formulation. The formulation is based on the assumption of an intermediate deformation from which the elastic deformation is characterized.

This assumption leads to a local multiplicative decomposition of the deformation gradient into elastic and plastic parts as

$$\mathbf{F} = \mathbf{F}^e \mathbf{F}^p \quad (3.83)$$

where  $\mathbf{F}^e$  and  $\mathbf{F}^p$  are the elastic and plastic deformations gradients respectively (see Simo and Hughes [95] for details). From this decomposition we define the plastic right Cauchy-Green deformation tensor as

$$\mathbf{C}^p = \mathbf{F}^{pT} \mathbf{F}^p. \quad (3.84)$$

We assume the Helmholtz free energy function takes the form  $\psi = \psi(\mathbf{C}, \mathbf{C}^p, \mathbf{Q}, c, c_{,I}, \dot{c})$ , where  $\mathbf{Q}$  is a set of internal plastic variables. Substituting this into (3.31) we get

$$\begin{aligned} & \int_{\Omega_0} \left( \frac{\partial \psi}{\partial C_{IJ}} \dot{C}_{IJ} + \frac{\partial \psi}{\partial C_{IJ}^p} \dot{C}_{IJ}^p + \frac{\partial \psi}{\partial Q_I} \dot{Q}_I + \frac{\partial \psi}{\partial c} \dot{c} + \frac{\partial \psi}{\partial c_{,I}} \dot{c}_{,I} + \frac{\partial \psi}{\partial \dot{c}} \ddot{c} \right) d\Omega_0 \\ & \leq \int_{\Omega_0} \left( \frac{1}{2} S_{IJ} \dot{C}_{IJ} + \xi_I \dot{c}_{,I} - \pi \dot{c} \right) d\Omega_0. \end{aligned} \quad (3.85)$$

For purely elastic deformations, we assume that the plastic dissipation, defined as

$$\mathbb{D}^p(\mathbf{C}, \mathbf{C}^p, \mathbf{Q}; \dot{\mathbf{C}}^p, \dot{\mathbf{Q}}) = - \frac{\partial \psi}{\partial C_{IJ}^p} \dot{C}_{IJ}^p - \frac{\partial \psi}{\partial Q_I} \dot{Q}_I, \quad (3.86)$$

is zero. Following the same arguments as given in Section 3.4.1, this assumption leads to the following conditions:

$$\frac{\partial \psi}{\partial \dot{c}} = 0 \quad (3.87)$$

$$\frac{\partial \psi}{\partial c_{,I}} = \xi_I \quad (3.88)$$

$$2 \frac{\partial \psi}{\partial C_{IJ}} = S_{IJ} \quad (3.89)$$

$$\mathbb{D}^p = - \frac{\partial \psi}{\partial C_{IJ}^p} \dot{C}_{IJ}^p - \frac{\partial \psi}{\partial Q_I} \dot{Q}_I \geq 0 \quad (3.90)$$

$$\left( \pi + \frac{\partial \psi}{\partial c} \right) \dot{c} \leq 0 \Rightarrow \pi = - \frac{\partial \psi}{\partial c} - \beta \dot{c} \quad (3.91)$$

where  $\beta = \beta(\mathbf{C}, \mathbf{C}^p, c, c_{,I}) \geq 0$  is a dissipation constant. Substituting these into the balance laws, we get:

**Linear momentum:**

$$\left( 2F_{iK} \frac{\partial\psi}{\partial C_{KJ}} \right)_{,J} + B_i = \rho_0 \ddot{U}_i \quad (3.92)$$

**Microforce:**

$$\left( \frac{\partial\psi}{\partial c_{,I}} \right)_{,I} + l - \frac{\partial\psi}{\partial c} = \beta \dot{c} \quad (3.93)$$

With the governing equations of motion and the second-order approximation from Section 3.2 in hand, we can derive the strong form of the phase-field model for ductile fracture. The Helmholtz free-energy is given by

$$\psi(C_{IJ}, C_{IJ}^p, Q_I, c, c_{,I}) = \tilde{W}(C_{IJ}, C_{IJ}^p, Q_I) + \mathcal{G}_c \left[ \frac{(1-c)^2}{4\ell_0} + \ell_0 c_{,I} c_{,I} \right] \quad (3.94)$$

where we have left the elastic strain energy density function  $\tilde{W}$  to be defined later. Then

$$\left( 2F_{iK} \frac{\partial\psi}{\partial C_{KJ}} \right)_{,J} = \left( 2F_{iK} \frac{\tilde{W}}{\partial C_{KJ}} \right)_{,J} \quad (3.95)$$

$$\left( \frac{\partial\psi}{\partial c_{,I}} \right)_{,I} = 2\mathcal{G}_c \ell_0 (c_{,I})_{,I} \quad (3.96)$$

$$\frac{\partial\psi}{\partial c} = \frac{\partial\tilde{W}}{\partial c} - \mathcal{G}_c \frac{1-c}{2\ell_0}. \quad (3.97)$$

Now, assuming  $l = 0$ , and  $\beta = 0$  and substituting these back into (3.92) and (3.93) we get the strong form of the equations of motion

$$(S) \begin{cases} \left( 2F_{iK} \frac{\tilde{W}}{\partial C_{KJ}} \right)_{,J} + B_i = \rho_0 \ddot{U}_i & \text{on } \Omega_0 \times ]0, T[ \\ \frac{2\ell_0}{\mathcal{G}_c} \frac{\partial\tilde{W}}{\partial c} + c - 4\ell_0^2 c_{,I} c_{,I} = 1 & \text{on } \Omega_0 \times ]0, T[ \end{cases} \quad (3.98)$$

These equations of motion can be solved to find both the displacement field  $\mathbf{U}(\mathbf{X}, t)$  and phase-field  $c(\mathbf{X}, t)$ . The equations of motion (3.98) are subject to the boundary conditions

$$(S: BC) \begin{cases} U_i = G_i & \text{on } \partial\Omega_{0,G_i} \times ]0, T[ \\ T_i = H_i & \text{on } \partial\Omega_{0,H_i} \times ]0, T[ \\ c_{,I} N_I = 0 & \text{on } \partial\Omega_0 \times ]0, T[ \end{cases} \quad (3.99)$$

with  $G_i(\mathbf{X}, t)$  and  $H_i(\mathbf{X}, t)$  being presciped on  $\partial\Omega_{0,G_i}$  and  $\partial\Omega_{0,H_i}$ , respectively, for all  $t \in ]0, T[$ , and with  $\mathbf{N}$  being the unit outward normal on  $\partial\Omega_0$  and  $T_i = P_{iI} N_I$  the first Piola-Kirchhoff traction vector. In addition, the equations of motions (3.98) are supplemented with initial conditions

$$(S: IC) \begin{cases} \mathbf{U}(\mathbf{X}, 0) = \mathbf{U}_0(\mathbf{X}) & \mathbf{X} \in \Omega_0 \\ \dot{\mathbf{U}}(\mathbf{X}, 0) = \mathbf{V}_0(\mathbf{X}) & \mathbf{X} \in \Omega_0 \end{cases} \quad (3.100)$$

for both the displacement and velocity field.

### 3.5.1 Finite deformation hyperelastic model

From the elastic/plastic decomposition of the deformation gradient,  $\mathbf{F} = \mathbf{F}^e \mathbf{F}^p$ , we define the elastic left Cauchy-Green tensor as

$$\mathbf{b}^e = \mathbf{F}^e \mathbf{F}^{eT} \quad (3.101)$$

and the determinate of the elastic deformation gradient as

$$J^e = \det \mathbf{F}^e. \quad (3.102)$$

We also define the deviatoric left Cauchy-Green tensor as

$$\bar{\mathbf{b}}^e = (J^e)^{-2/3} \mathbf{b}^e \quad (3.103)$$

so that  $\det \bar{\mathbf{b}}^e = 1$ .

We then begin with the assumption that the undamaged elastic stress response is characterized by the elastic strain energy density function

$$W = U(J^e) + \bar{W}(\bar{\mathbf{b}}^e) \quad (3.104)$$

where  $U(J^e)$  is the volumetric contribution and  $\bar{W}(\bar{\mathbf{b}}^e)$  is the deviatoric contribution. We follow the strain energy decomposition introduced by Amor et al. [5] and split this into tensile and compressive parts as

$$W^+ = \begin{cases} U(J^e) + \bar{W}(\bar{\mathbf{b}}^e) & J^e \geq 1 \\ \bar{W}(\bar{\mathbf{b}}^e) & J^e < 1 \end{cases} \quad (3.105)$$

$$W^- = \begin{cases} 0 & J^e \geq 1 \\ U(J^e) & J^e < 1 \end{cases}. \quad (3.106)$$

The damaged strain energy is then given by

$$\tilde{W}(J^e, \bar{\mathbf{b}}^e) = g(c)W^+(J^e, \bar{\mathbf{b}}^e) + W^-(J^e). \quad (3.107)$$

For the formulation presented here, we use the specific forms

$$U(J^e) = \frac{1}{2}\kappa \left[ \frac{1}{2}(J^{e2} - 1) - \ln J^e \right], \quad (3.108)$$

$$\bar{W}(\bar{\mathbf{b}}^e) = \frac{1}{2}\mu (\text{tr} [\bar{\mathbf{b}}^e] - 3) \quad (3.109)$$

where  $\kappa$  and  $\mu$  are the bulk and shear modulus respectively. The resulting constitutive model is summarized in Box 3.1.

Box 3.1: A finite deformation hyperelastic constitutive model based on a split of the volumetric part of the strain energy.

Positive, negative and damaged strain-energy density functions

$$\begin{aligned} U(J^e) &= \frac{1}{2}\kappa \left[ \frac{1}{2}(J^{e2} - 1) - \ln J^e \right] \\ \bar{W}(\bar{\mathbf{b}}^e) &= \frac{1}{2}\mu (\text{tr}[\bar{\mathbf{b}}^e] - 3) \\ W^+(J^e, \bar{\mathbf{b}}^e) &= \begin{cases} U(J^e) + \bar{W}(\bar{\mathbf{b}}^e) & J^e \geq 1 \\ \bar{W}(\bar{\mathbf{b}}^e) & J^e < 1 \end{cases} \\ W^-(J^e) &= \begin{cases} 0 & J^e \geq 1 \\ U(J^e) & J^e < 1 \end{cases} \\ \tilde{W}(J^e, \bar{\mathbf{b}}^e) &= g(c)W^+(J^e, \bar{\mathbf{b}}^e) + W^-(J^e) \end{aligned}$$

Positive, negative and damaged stress tensors

$$\begin{aligned} \boldsymbol{\sigma}^+ &= \begin{cases} U'(J^e)\mathbf{I} + J^{e-1}\mu \text{dev}[\bar{\mathbf{b}}^e] & J^e \geq 1 \\ J^{e-1}\mu \text{dev}[\bar{\mathbf{b}}^e] & J^e < 1 \end{cases} \\ \boldsymbol{\sigma}^- &= \begin{cases} \mathbf{0} & J^e \geq 1 \\ U'(J^e)\mathbf{I} & J^e < 1 \end{cases} \\ \tilde{\boldsymbol{\sigma}} &= g(c)\boldsymbol{\sigma}^+ + \boldsymbol{\sigma}^- \end{aligned}$$

**Remark 3.5.** The split of the elastic strain energy given in (3.105) and (3.106) assumes that in the case of isochoric plastic flow ( $J^e$  is constant) the evolution of the phase-field is determined by the deviatoric part of the elastic deformation. We realize that actual physical processes are much more complex and may include dependencies on accumulated plastic work and thermal effects, among others. We plan to investigate the inclusion of other mechanisms in future work.

**Remark 3.6.** Using the relationship  $\mathbf{C}^p = \mathbf{F}^T \mathbf{b}^{e-1} \mathbf{F}$  we get

$$\text{tr} [\bar{\mathbf{b}}^e] = \text{tr} [J^{-2/3} \mathbf{F} \mathbf{C}^{p-1} \mathbf{F}^T] \quad (3.110)$$

$$= J^{-2/3} \mathbf{1} : [\mathbf{F} \mathbf{C}^{p-1} \mathbf{F}^T] \quad (3.111)$$

$$= J^{-2/3} \mathbf{C} : \mathbf{C}^{p-1} \quad (3.112)$$

$$= \bar{\mathbf{C}} : \mathbf{C}^{p-1}. \quad (3.113)$$

With this, we can rewrite the undamaged elastic strain energy density function as

$$W = W(\mathbf{C}, \mathbf{C}^p) = U(J) + \bar{W}(\mathbf{C}, \mathbf{C}^p). \quad (3.114)$$

This allows the constitutive theory in Box 3.1 to be written in an equivalent form in the reference configuration. We make use of this fact in the next section.

### 3.5.2 Elastoplastic constitutive model

In this section we describe an associative elastoplastic constitutive model based on  $J_2$  flow theory with isotropic hardening. We assume an isotropic stress response governed by the damage/elastic constitutive model given in Box 3.1. We also assume the plastic flow to be isochoric, i.e.,

$$\det \mathbf{F}^p = 1 \Rightarrow J = \det \mathbf{F}^e. \quad (3.115)$$

The associative flow rule is derived by making use of the principle of maximum plastic dissipation. This principle states that for any given fixed internal plastic variables  $\{\mathbf{C}^p, \mathbf{Q}\}$  and yield function  $f(\mathbf{C}, \mathbf{C}^p, \mathbf{Q})$  the actual deformation tensor,  $\mathbf{C}$ , satisfies

$$\begin{aligned} \mathbb{D}^p(\mathbf{C}, \mathbf{C}^p, \mathbf{Q}; \dot{\mathbf{C}}^p, \dot{\mathbf{Q}}) &\geq \mathbb{D}^p(\hat{\mathbf{C}}, \mathbf{C}^p, \mathbf{Q}; \dot{\mathbf{C}}^p, \dot{\mathbf{Q}}) \\ \text{for all } \hat{\mathbf{C}} \text{ s.t. } f(\hat{\mathbf{C}}, \mathbf{C}^p, \mathbf{Q}) &\leq 0 \end{aligned} \quad (3.116)$$

where  $\mathbb{D}^p$  is defined in (3.86). Assuming an uncoupled free energy function in the internal variables,  $\mathbf{Q}$ , the principle of maximum plastic dissipation leads to the associative flow rule

$$-4 \frac{\partial^2 \tilde{W}}{\partial \mathbf{C} \partial \mathbf{C}^p} : \dot{\mathbf{C}}^p = 2\dot{\gamma} \frac{\partial f}{\partial \mathbf{C}} \quad (3.117)$$

and the Kuhn-Tucker loading/unloading conditions

$$\dot{\gamma} \geq 0, \quad f \leq, \quad \dot{\gamma}f = 0. \quad (3.118)$$

For a detailed derivation see Simo [94].

For the current formulation, we assume that yielding is governed by the Mises-Huber yield condition in terms of the damaged Kirchhoff stress,  $\tilde{\boldsymbol{\tau}}$ , by the yield function

$$f(\tilde{\boldsymbol{\tau}}, \alpha) = \|\tilde{\mathbf{s}}\| - \sqrt{\frac{2}{3}}k(\alpha) \quad (3.119)$$

where  $\alpha$  is the hardening parameter and  $k(\alpha)$  the isotropic hardening modulus. The hardening law is determined by the choice of  $k(\alpha)$  as, for example,

$$k(\alpha) = \begin{cases} \sigma_y & \text{(Perfect plasticity)} \\ \sigma_y + K\alpha & \text{(Linear hardening)} \end{cases} \quad (3.120)$$

where  $\sigma_y$  is the flow stress and  $K$  is the linear hardening coefficient.

Given the hyperelastic stress response from Box 3.1 the yield function can be rewritten in the form  $f(\bar{\mathbf{C}}, \mathbf{C}^p, \mathbf{Q})$  where  $\bar{\mathbf{C}} = J^{-2/3}\mathbf{C}$ . Then, again following the derivation given by Simo [94], the flow rule is

$$J^{-2/3} \text{DEV} \left[ 2 \frac{\partial^2 g(c)\bar{W}}{\partial \bar{\mathbf{C}} \partial \mathbf{C}^p} : \dot{\mathbf{C}}^p \right] = -\dot{\gamma} J^{-2/3} \text{DEV} \left[ \frac{\partial f}{\partial \bar{\mathbf{C}}} \right] \quad (3.121)$$

where  $\text{DEV}[\cdot] = (\cdot) - \frac{1}{3}[\mathbf{C} : (\cdot)]\mathbf{C}^{-1}$  is the deviator in the reference configuration.

From the fact that

$$\frac{\partial f}{\partial \bar{\mathbf{C}}} = \frac{\partial}{\partial \bar{\mathbf{C}}} (g(c)\mu \text{dev} [\bar{\mathbf{b}}^e]) \quad (3.122)$$

and that  $g(c)$  is independent of  $\bar{\mathbf{C}}$  and  $\mathbf{C}^p$ , we see that  $g(c)$  can be factored out of (3.121). Thus, the flow rule is independent of  $c$ . In the reference configuration, the flow rule simplifies to

$$\dot{\mathbf{C}}^{p-1} = -\frac{2}{3}\dot{\gamma}\mathbf{C} : \mathbf{C}^{p-1}\mathbf{N} \quad (3.123)$$

$$\mathbf{N} = \mathbf{F}^{-1}\mathbf{n}\mathbf{F}^{-T} \quad (3.124)$$

$$\mathbf{n} = \frac{\mathbf{s}}{\|\mathbf{s}\|} \quad (3.125)$$

which can be pushed forward to the current configuration to get

$$L_v \mathbf{b}^e = -\frac{2}{3}\dot{\gamma} \operatorname{tr} [\mathbf{b}^e] \mathbf{n} \quad (3.126)$$

where  $L_v \mathbf{b}^e$  is the Lie derivative of  $\mathbf{b}^e$  (see Simo and Hughes [95] for details). The complete elastic/plastic constitutive model is summarized in Box 3.2.

### 3.5.2.1 Return-mapping algorithm

Numerical integration of the constitutive theory summarized in Box 3.2 is performed by a backward Euler scheme that leads to the construction of a radial return-mapping algorithm. Following the procedure outlined by Simo and Hughes [95, Chapter 9] and taking into account the definition of the damage/elastic stress response from Box 3.1, the phase-field parameter is incorporated as follows. For each time interval,  $[t_n, t_{n+1}]$ , we assume the initial states  $\phi_n$ ,  $\bar{\mathbf{b}}_n^e$ ,  $\alpha_n$ , and  $\mathbf{F}_n$  are known. The configuration at time  $t_{n+1}$  is obtained from the incremental displacement field  $\mathbf{u}_n : \Omega_n \rightarrow \mathbb{R}^3$  as

$$\phi_{n+1}(\mathbf{X}) = \phi_n(\mathbf{X}) + \mathbf{u}_n(\phi_n(\mathbf{X})) \quad (3.127)$$

Box 3.2: A finite deformation rate-independent elastoplastic constitutive model based on  $J_2$  flow theory with an associative flow rule. The stress response is governed by the hyperelastic constitutive model from Box 3.1.

Elastic-plastic strain decomposition

$$\mathbf{F} = \mathbf{F}^e \mathbf{F}^p$$

Finite deformation damage/elastic constitutive model (see Box 3.1)

$$\tilde{\boldsymbol{\sigma}} = g(c)\boldsymbol{\sigma}^+ + \boldsymbol{\sigma}^-, \quad \tilde{\boldsymbol{\tau}} = J^e \tilde{\boldsymbol{\sigma}}$$

Mises-Huber yield function

$$f = \|\tilde{\mathbf{s}}\| - \sqrt{\frac{2}{3}k(\alpha)}, \quad \tilde{\mathbf{s}} = \text{dev } \tilde{\boldsymbol{\tau}}$$

Flow rule

$$\text{dev}[L_v \mathbf{b}^e] = -\frac{2}{3}\dot{\gamma} \text{tr}[\mathbf{b}^e] \mathbf{n}, \quad \mathbf{n} = \frac{\tilde{\mathbf{s}}}{\|\tilde{\mathbf{s}}\|}$$

Hardening law

$$\dot{\alpha} = \sqrt{\frac{2}{3}\dot{\gamma}}$$

Kuhn-Tucker loading/unloading conditions

$$f \leq 0, \quad \dot{\gamma} \geq 0, \quad f\dot{\gamma} = 0$$

and the relative deformation gradient is given as

$$\mathbf{f}_{n+1} = \mathbf{I} + \nabla_{x_n} \mathbf{u}_n. \quad (3.128)$$

Assuming an elastic trial response, *i.e.*,  $\alpha_{n+1}^{\text{trial}} = \alpha_n$ , the trial deformation is given by

$$\bar{\mathbf{f}}_{n+1} = \det[\mathbf{f}_{n+1}]^{-1/3} \mathbf{f}_{n+1} \quad (3.129)$$

$$\bar{\mathbf{b}}_{n+1}^{\text{e trial}} = \bar{\mathbf{f}}_{n+1} \bar{\mathbf{b}}_n^e \bar{\mathbf{f}}_{n+1}^T. \quad (3.130)$$

The trial damaged deviatoric stress is then computed from the relationship for the damaged Kirchhoff stress,  $\tilde{\boldsymbol{\tau}} = J\tilde{\boldsymbol{\sigma}}$ , as

$$\tilde{\mathbf{s}}_{n+1}^{\text{trial}} = \text{dev}[\tilde{\boldsymbol{\tau}}_{n+1}^{\text{trial}}] = g(c)\mu \text{dev}[\bar{\mathbf{b}}_{n+1}^{e \text{ trial}}]. \quad (3.131)$$

Now, letting  $\Delta\gamma$  be the algorithmic counterpart to  $\dot{\gamma}$ , the discrete Kuhn-Tucker loading/unloading conditions are

$$f(\tilde{\boldsymbol{\tau}}_{n+1}, \alpha_{n+1}) \leq 0, \quad \Delta\gamma \geq 0, \quad \Delta\gamma f(\tilde{\boldsymbol{\tau}}_{n+1}, \alpha_{n+1}) = 0. \quad (3.132)$$

These conditions give rise to two alternatives based on the result of the evaluation of the yield function for the given trial state. We define

$$f_{n+1}^{\text{trial}} := f(\tilde{\boldsymbol{\tau}}_{n+1}^{\text{trial}}, \alpha_n) = \|\tilde{\mathbf{s}}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}k(\alpha_n). \quad (3.133)$$

If  $f_{n+1}^{\text{trial}} \leq 0$ , then (3.132) is satisfied with  $\Delta\gamma = 0$  and the trial damage/elastic state is accepted as the solution at  $t_{n+1}$ . If, on the other hand,  $f_{n+1}^{\text{trial}} \geq 0$ , then (3.132) is not satisfied, implying that the trial stress is not admissible. In this case, we need to find  $\Delta\gamma > 0$  that satisfies the condition  $f(\tilde{\boldsymbol{\tau}}_{n+1}, \alpha_{n+1}) = 0$ . To do this, we make use of the constitutive relation between the stress and left Cauchy-Green tensor and the discrete evolution equations

$$\bar{\mathbf{b}}_{n+1}^e = \bar{\mathbf{b}}_{n+1}^{e \text{ trial}} - \frac{2}{3}\Delta\gamma \text{tr}[\bar{\mathbf{b}}_{n+1}^e]\mathbf{n}_{n+1} \quad (3.134)$$

$$\alpha_{n+1} = \alpha_n + \sqrt{\frac{2}{3}}\Delta\gamma \quad (3.135)$$

where

$$\mathbf{n}_{n+1} = \frac{\tilde{\mathbf{s}}_{n+1}}{\|\tilde{\mathbf{s}}_{n+1}\|}. \quad (3.136)$$

Using relationship (3.131) and (3.134), with  $\text{tr}[\mathbf{n}_{n+1}] = 0$  implying  $\text{tr}[\bar{\mathbf{b}}_{n+1}^e] = \text{tr}[\bar{\mathbf{b}}_{n+1}^{e\text{ trial}}]$ , we get

$$\tilde{\mathbf{s}}_{n+1} = \tilde{\mathbf{s}}_{n+1}^{\text{trial}} - \frac{2}{3}g(c)\mu\Delta\gamma \text{tr}[\bar{\mathbf{b}}_{n+1}^{e\text{ trial}}]\mathbf{n}_{n+1} \quad (3.137)$$

and

$$\|\tilde{\mathbf{s}}_{n+1}\| = \|\tilde{\mathbf{s}}_{n+1}^{\text{trial}}\| - \frac{2}{3}g(c)\mu\Delta\gamma \text{tr}[\bar{\mathbf{b}}_{n+1}^{e\text{ trial}}]. \quad (3.138)$$

We then find  $\Delta\gamma$  by solving the (possibly nonlinear) scalar valued equation

$$\hat{f}(\Delta\gamma) = \|\tilde{\mathbf{s}}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}k(\alpha_n + \sqrt{\frac{2}{3}}\Delta\gamma) - \frac{2}{3}g(c)\mu\Delta\gamma \text{tr}[\bar{\mathbf{b}}_{n+1}^{e\text{ trial}}] = 0 \quad (3.139)$$

and update the trial state using equations (3.131) and (3.135). The return-mapping algorithm is summarized in Box 3.3.

**Box 3.3:** Return-mapping algorithm for the finite deformation rate-independent elastoplastic constitutive model from Box 3.2.

1. Update the current configuration

$$\phi_{n+1}(\mathbf{X}) = \phi_n(\mathbf{X}) + \mathbf{u}_n(\phi_n(\mathbf{X}))$$

$$\mathbf{f}_{n+1} = \mathbf{I} + \nabla_{x_n} \mathbf{u}_n$$

$$\mathbf{F}_{n+1} = \mathbf{f}_{n+1}\mathbf{F}_n$$

2. Compute the damage/elastic predictor

$$\bar{\mathbf{f}}_{n+1} = \det[\mathbf{f}_{n+1}]^{-1/3} \mathbf{f}_{n+1}$$

$$\bar{\mathbf{b}}_{n+1}^{e\text{ trial}} = \bar{\mathbf{f}}_{n+1} \bar{\mathbf{b}}_n^e \bar{\mathbf{f}}_{n+1}^T$$

$$\tilde{\mathbf{s}}_{n+1}^{\text{trial}} = g(c)\mu \text{dev} [\bar{\mathbf{b}}_{n+1}^{e\text{ trial}}]$$

3. Check for plastic loading

```

 $f_{n+1}^{\text{trial}} = \|\tilde{\mathbf{s}}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}k(\alpha_n)$ 
IF  $f_{n+1}^{\text{trial}} \leq 0$  THEN
    Set  $(\cdot)_{n+1} = (\cdot)_{n+1}^{\text{trial}}$ , and EXIT
ELSE
    Go to 4 (return-mapping)
ENDIF

```

4. Return-mapping

$$\bar{\mu} = \frac{1}{3}g(c)\mu \text{tr} [\bar{\mathbf{b}}_{n+1}^{e \text{ trial}}]$$

Solve:  $\hat{f}(\Delta\gamma) = \|\tilde{\mathbf{s}}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}k(\alpha_n + \sqrt{\frac{2}{3}}\Delta\gamma) - 2\bar{\mu}\Delta\gamma = 0$

$$\tilde{\mathbf{s}}_{n+1} = \tilde{\mathbf{s}}_{n+1}^{\text{trial}} - 2\bar{\mu}\Delta\gamma \mathbf{n}$$

$$\alpha_{n+1} = \alpha_n + \sqrt{\frac{2}{3}}\Delta\gamma$$

5. Update the Kirchhoff stress

$$J_{n+1} = \det[\mathbf{F}_{n+1}]$$

$$\tilde{p}_{n+1} = \begin{cases} g(c)U'(J_{n+1}) & J^e \geq 1 \\ U'(J_{n+1}) & J^e < 1 \end{cases}$$

$$\tilde{\boldsymbol{\tau}}_{n+1} = J_{n+1}\tilde{p}_{n+1}\mathbf{I} + \tilde{\mathbf{s}}_{n+1}$$

6. Update the intermediate configuration

$$\bar{\mathbf{b}}_{n+1}^e = \frac{1}{g(c)\mu} (\tilde{\mathbf{s}}_{n+1} + \bar{\mu}\mathbf{I})$$

### 3.6 Irreversibility

To be physically consistent, the phase-field fracture model must enforce irreversibility of the fracture process. In other words, all cracks must only extend over time. This can be written in the discontinuous case as  $\Gamma_0(t) \subseteq \Gamma_0(t + \Delta t)$  for all  $\Delta t > 0$ . For the phase-field model, irreversibility can be achieved by enforcing monotonicity of  $c$ , i.e., satisfying  $\dot{c} \leq 0$  for all  $t$ . In this case, the solution of the phase-field equation

becomes a constrained minimization problem with the associated additional computational cost. An alternative approach suggested by Bourdin, Francfort, and Marigo [26] constrains a subset of the discrete phase-field control variables that meet a minimum value threshold. This approach can be stated as follows. Let  $\Phi = \{\phi_A\}_A^{n_b}$  be the set of discrete phase-field control variables (see (3.66)). Define

$$K = \{\phi_A \in \Phi \text{ s.t. } \phi_A \leq \eta\} \quad (3.140)$$

where  $\eta > 0$  is a small parameter. The phase-field equation is then solved subject to the constraint

$$\phi_A = 0 \quad \forall \phi_A \in K. \quad (3.141)$$

This approach is much easier to solve numerically but requires the selection of the small parameter  $\eta$  (the selection of  $\eta$  can be guided by the discussion presented in Section 4.1). Also, for non-interpolatory bases, such as those used in isogeometric analysis, the value of the control-variable is not the value of the solution at that location. This makes it difficult to interpret the meaning of  $K$ .

### 3.6.1 Strain-history functional

Miehe et al. [68] introduced the strain-history functional

$$\mathcal{H}(\mathbf{X}, t) = \max_{\tau \leq t} (\psi_e^+(\mathbf{F}(\tau))) \quad (3.142)$$

as another approach to enforce irreversibility. This functional replaces  $\psi_e^+$  in the phase-field equation and prevents the reaction term from decreasing. Motivation for this functional is based on an analysis of a one-dimensional local damage model (see Miehe et al. [68, Appendix A]). One advantage of this functional is computational efficiency. The

only computational cost is a floating point comparison and it only requires storing one history variable per integration point. Another advantage of this approach is that an initial history functional can be used to model initial cracks (see Appendix B.1).

The thermodynamic justification for the local damage model, however, does not necessarily hold for the non-local model where the derivative terms of  $c$  effect the solution. In fact, results presented in Section 4.3 indicate that this approach requires further study to determine if it is indeed thermodynamically consistent. In many practical cases, however, this approach may still be a good choice.

## 3.7 Time discretization

In this section we introduce two temporal discretization schemes: a monolithic implicit scheme and a two-stage staggered scheme in which the momentum equation is integrated using an implicit-explicit scheme.

### 3.7.1 Monolithic generalized- $\alpha$ time discretization

The monolithic time integration scheme is based on the generalized- $\alpha$  method introduced by Chung and Hulbert [33]. Let  $\mathbf{R}^u$  and  $\mathbf{R}^c$  be the residuals of the discrete momentum and phase-field equations respectively. For time step  $n$ , let  $\mathbf{d}_n$  and  $\phi_n$  be the vectors of control variable degrees-of-freedom of the displacements and phase-field, respectively (see (3.64) and (3.66)). We then define  $\mathbf{v}_n = \dot{\mathbf{d}}_n$ , and  $\mathbf{a}_n = \ddot{\mathbf{d}}_n$ . The monolithic generalized- $\alpha$  time integration scheme is then stated as follows: given

$(\mathbf{d}_n, \mathbf{v}_n, \mathbf{a}_n)$ , find  $(\mathbf{d}_{n+1}, \mathbf{v}_{n+1}, \mathbf{a}_{n+1}, \mathbf{d}_{n+\alpha_f}, \mathbf{v}_{n+\alpha_f}, \mathbf{a}_{n+\alpha_m}, \phi_{n+1})$  such that

$$\mathbf{R}^u(\mathbf{d}_{n+\alpha_f}, \mathbf{v}_{n+\alpha_f}, \mathbf{a}_{n+\alpha_m}, \phi_{n+1}) = 0, \quad (3.143)$$

$$\mathbf{R}^c(\mathbf{d}_{n+\alpha_f}, \phi_{n+1}) = 0, \quad (3.144)$$

$$\mathbf{d}_{n+\alpha_f} = \mathbf{d}_n + \alpha_f(\mathbf{d}_{n+1} - \mathbf{d}_n), \quad (3.145)$$

$$\mathbf{v}_{n+\alpha_f} = \mathbf{v}_n + \alpha_f(\mathbf{v}_{n+1} - \mathbf{v}_n), \quad (3.146)$$

$$\mathbf{a}_{n+\alpha_m} = \mathbf{a}_n + \alpha_m(\mathbf{a}_{n+1} - \mathbf{a}_n), \quad (3.147)$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \Delta t((1 - \gamma)\mathbf{a}_n + \gamma\mathbf{a}_{n+1}), \quad (3.148)$$

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t\mathbf{v}_n + \frac{(\Delta t)^2}{2}((1 - 2\beta)\mathbf{a}_n + 2\beta\mathbf{a}_{n+1}), \quad (3.149)$$

where  $\Delta t = t_{n+1} - t_n$  is the time step and the parameters  $\alpha_f, \alpha_m, \beta$ , and  $\gamma$  define the method. These parameters will be discussed below.

At each time step, the solution is obtained using a Newton-Raphson method to solve the nonlinear equations above. Letting  $i$  be the Newton iteration, the residual vector and consistent tangent matrix for the linearized system are defined by

$$\frac{\partial \mathbf{R}_i^u}{\partial \mathbf{a}_{n+1}} \Delta \mathbf{a} + \frac{\partial \mathbf{R}_i^u}{\partial \phi_{n+1}} \Delta \phi = -\mathbf{R}_i^u, \quad (3.150)$$

$$\frac{\partial \mathbf{R}_i^c}{\partial \mathbf{a}_{n+1}} \Delta \mathbf{a} + \frac{\partial \mathbf{R}_i^c}{\partial \phi_{n+1}} \Delta \phi = -\mathbf{R}_i^c, \quad (3.151)$$

where

$$\mathbf{R}_i^u = \mathbf{R}^u(\mathbf{d}_{n+\alpha_f}^i, \mathbf{v}_{n+\alpha_f}^i, \mathbf{a}_{n+\alpha_m}^i, \phi_{n+1}), \quad (3.152)$$

and

$$\mathbf{R}_i^c = \mathbf{R}^c(\mathbf{d}_{n+\alpha_f}, \phi_{n+1}). \quad (3.153)$$

For each iteration, the linearized system defined by (3.150) and (3.151) is solved and iteration continues until convergence of the residual vectors occurs. For the examples discussed in this dissertation, we have defined convergence as

$$\max \left\{ \frac{\|\mathbf{R}_i^u\|}{\|\mathbf{R}_0^u\|}, \frac{\|\mathbf{R}_i^c\|}{\|\mathbf{R}_0^c\|} \right\} \leq tol \quad (3.154)$$

where  $\|\cdot\|$  denotes the Euclidean norm. For the simulations considered in this work,  $tol = 10^{-4}$  has been observed to be an appropriate choice.

As an example, the residual vectors for the small-strain second-order theory are defined as

$$\mathbf{R}^u = \{R_{A,i}^u\}, \quad (3.155)$$

$$R_{A,i}^u = (\mathbf{h}, N_A \mathbf{e}_i)_{\partial\Omega_h} - (\rho \ddot{\mathbf{u}}^h, N_A \mathbf{e}_i)_\Omega - \left( \sigma_{jk}, B_A^{ijk} \right)_\Omega \quad (3.156)$$

and

$$\mathbf{R}^c = \{R_A^c\}, \quad (3.157)$$

$$R_A^c = (1, N_A)_\Omega - \left( \frac{2\ell_0 g'(c^h) \mathcal{H}}{\mathcal{G}_c} + c^h, N_A \right)_\Omega - \left( 4\ell_0^2 \frac{\partial c^h}{\partial x_i}, \frac{\partial N_A}{\partial x_i} \right)_\Omega, \quad (3.158)$$

where  $\mathcal{H}$  is the strain history functional introduced in (3.142),  $\mathbf{e}_i$  is the  $i$ th Euclidean basis vector and,

$$B_A^{ijk} = \frac{1}{2} \left( \frac{\partial N_A}{\partial x_j} \delta_{ik} + \frac{\partial N_A}{\partial x_k} \delta_{ij} \right) \quad (3.159)$$

so that  $\varepsilon_{jk} = \sum_A B_A^{ijk} d_{A,i}$ . Similar arrays can be defined for the higher-order and elastic/plastic theories.

**Parameter selection** In Chung and Hulbert [33] it was shown that  $\alpha_f$  and  $\alpha_m$  can be parametrized by the spectral radius,  $\rho_\infty$ , of the amplification matrix at  $\Delta t = \infty$  such

that second-order accuracy and unconditional stability are achieved for a second-order linear problem if

$$\alpha_f = \frac{1}{\rho_\infty + 1}, \quad (3.160)$$

$$\alpha_m = \frac{2 - \rho_\infty}{\rho_\infty + 1}, \quad (3.161)$$

$$\beta = \frac{1}{4}(1 + \alpha_m - \alpha_f)^2, \quad (3.162)$$

$$\gamma = \frac{1}{2} + \alpha_m - \alpha_f. \quad (3.163)$$

### 3.7.2 Staggered time discretization

For the staggered time integration scheme, the momentum and phase-field equations are solved independently. At a given time step, the momentum equation is solved first to get the updated displacements. Using the updated displacements, the phase-field equation is solved. In addition to reducing the problem to solving two (possibly linear) systems, this scheme also allows greater flexibility in how the momentum equation is solved, i.e., we can use either implicit or explicit schemes. This scheme can also be generalized to a predictor/multicorrector format where additional Newton-Raphson iterations can be performed within a time step. Below we present a general predictor/multicorrector algorithm.

Defining the residual vectors for the momentum and phase-field equations as before, and again letting  $\mathbf{d}$  and  $\phi$  be arrays of the control variable coefficients in (3.64) and (3.66), the staggered predictor/multicorrector time integration scheme is stated as follows: given  $(\mathbf{d}_n, \mathbf{v}_n, \mathbf{a}_n, \phi_n)$ , solve

## Predictor stage

$$i = 0 \quad (\text{iteration counter}) \quad (3.164)$$

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t(1 - \gamma)\mathbf{a}_n \quad (3.165)$$

$$\tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t\mathbf{v}_n + \frac{(\Delta t)^2}{2}(1 - 2\beta)\mathbf{a}_n \quad (3.166)$$

$$\mathbf{a}_{n+1}^{(i)} = 0 \quad (3.167)$$

$$\mathbf{v}_{n+1}^{(i)} = \tilde{\mathbf{v}}_{n+1} \quad (3.168)$$

$$\mathbf{d}_{n+1}^{(i)} = \tilde{\mathbf{d}}_{n+1} \quad (3.169)$$

$$\phi_{n+1}^{(i)} = \phi_n \quad (3.170)$$

## Multicorrector stage

$$\mathbf{a}_{n+\alpha_m}^{(i)} = \mathbf{a}_n + \alpha_m(\mathbf{a}_{n+1}^{(i)} - \mathbf{a}_n) \quad (3.171)$$

$$\mathbf{v}_{n+\alpha_f}^{(i)} = \mathbf{v}_n + \alpha_f(\mathbf{v}_{n+1}^{(i)} - \mathbf{v}_n) \quad (3.172)$$

$$\mathbf{d}_{n+\alpha_f}^{(i)} = \mathbf{d}_n + \alpha_f(\mathbf{d}_{n+1}^{(i)} - \mathbf{d}_n) \quad (3.173)$$

$$\mathbf{M}^* \Delta \mathbf{a} = \mathbf{R}^u(\mathbf{d}_{n+\alpha_f}^{(i)}, \mathbf{v}_{n+\alpha_f}^{(i)}, \mathbf{a}_{n+\alpha_m}^{(i)}, \phi_{n+1}^{(i)}) \quad (3.174)$$

$$\mathbf{a}_{n+1}^{(i+1)} = \mathbf{a}_{n+1}^{(i)} + \Delta \mathbf{a} \quad (3.175)$$

$$\mathbf{v}_{n+1}^{(i+1)} = \tilde{\mathbf{v}}_{n+1} + \Delta t \gamma \mathbf{a}_{n+1}^{(i+1)} \quad (3.176)$$

$$\mathbf{d}_{n+1}^{(i+1)} = \tilde{\mathbf{d}}_{n+1} + (\Delta t)^2 \beta \mathbf{a}_{n+1}^{(i+1)} \quad (3.177)$$

$$\mathbf{K}^{cc} \Delta \phi = \mathbf{F}^c \quad (3.178)$$

$$\phi_{n+1}^{(i+1)} = \Delta \phi \quad (3.179)$$

Again, we define the arrays for the small-strain second-order theory. The phase-field arrays in (3.178) are defined as

$$\mathbf{K}^{cc} = [K_{AB}], \quad (3.180)$$

$$K_{AB} = \left( \left( \frac{2\ell_0 g''(c^h)\mathcal{H}}{\mathcal{G}_c} + 1 \right) N_B, N_A \right)_\Omega + \left( 4\ell_0^2 \frac{\partial N_B}{\partial x_i}, \frac{\partial N_A}{\partial x_i} \right)_\Omega, \quad (3.181)$$

$$\mathbf{F}^c = \{F_A\}, \quad (3.182)$$

$$F_A = (1, N_A), \quad (3.183)$$

and  $\Delta t = t_{n+1} - t_n$  is the time step and the parameters  $\alpha_m, \alpha_f, \beta$ , and  $\gamma$ , which define the method, are selected as described below.

If the linearized momentum equation (3.174) is being solved implicitly then

$$\mathbf{M}^* = -\frac{\partial \mathbf{R}_i^u}{\partial \mathbf{a}_{n+1}} = \alpha_m \mathbf{M} + \alpha_f \beta (\Delta t)^2 \mathbf{K}, \quad (3.184)$$

where  $\mathbf{M}$  is the consistent mass matrix and

$$\mathbf{K} = [K_{AB,ij}^{uu}], \quad (3.185)$$

$$K_{AB,ij}^{uu} = \left( \frac{\partial \sigma_{lk}}{\partial \varepsilon_{mn}} B_B^{jmn}, B_A^{ilk} \right)_\Omega \quad (3.186)$$

is the consistent damage-elastic tangent stiffness matrix. If we let  $\mathbf{M}^* = \alpha_m \widetilde{\mathbf{M}}$  where  $\widetilde{\mathbf{M}}$  is the lumped mass matrix then the linearized momentum equation is solved explicitly. When computing with NURBS and T-splines, we compute  $\widetilde{\mathbf{M}}$  using the row-sum technique as described in Hughes [52]. Due to the fact that NURBS and T-spline basis functions are point-wise positive, the row-sum lumped mass matrix is guaranteed to be positive. Furthermore, it is also mass conservative.

**Remark 3.7.** In the small-strain implicit case (3.174) is linear in the displacements if  $\psi_e = \psi_e^+$  and  $\psi_e^- = 0$ , i.e., there is no tensile/compressive split of the strain energy. The

linearity of (3.178) depends on  $g(c)$ . If  $g'(c)$  is linear then  $\mathbf{K}^{cc}$  will not depend on  $c$  and (3.178) will be linear.

**Parameter selection** For the staggered solution strategy, the choice of  $\alpha_m$ ,  $\alpha_f$ , and  $\mathbf{M}^*$  provides several options for the type of algorithm that is used to solve the linear momentum problem. For the fully implicit case ( $\mathbf{M}^* = \alpha_m \mathbf{M} + \alpha_f \beta (\Delta t)^2 \mathbf{K}$ ) we use the generalized- $\alpha$  method described above. For the fully explicit case ( $\mathbf{M}^* = \alpha_m \widetilde{\mathbf{M}}$ ) we use either the HHT- $\alpha$  of Hilber, Hughes, and Tayler [51] or the explicit generalized- $\alpha$  method of Hulbert and Chung [56]. The HHT- $\alpha$  method, parameterized by  $\alpha$ , provides a second-order accurate family of algorithms for linear second-order equations if  $\alpha \in [-\frac{1}{3}, 0]$  and

$$\alpha_m = 1, \quad (3.187)$$

$$\alpha_f = 1 + \alpha, \quad (3.188)$$

$$\beta = \frac{(1 - \alpha)^2}{4}, \quad (3.189)$$

$$\gamma = \frac{1 - 2\alpha}{2} \quad (3.190)$$

(see Miranda, Ferencz, and Hughes [70]). The explicit generalized- $\alpha$  method, as shown by Hulbert and Chung [56], is a one-parameter family of explicit algorithms that provides optimal numerical dissipation and is second-order accurate for linear problems if  $\alpha_f = 0$  and

$$\alpha_m = \frac{2 - \rho_b}{1 + \rho_b}, \quad (3.191)$$

$$\beta = \frac{5 - 3\rho_b}{(1 + \rho_b)^2(2 - \rho_b)}, \quad (3.192)$$

$$\gamma = \alpha_m + \frac{1}{2}, \quad (3.193)$$

where  $\rho_b$  is the spectral radius value at the bifurcation limit of the principal roots of the characteristic equation.

### 3.8 A note on $\Gamma$ -convergence

The notion of  $\Gamma$ -convergence has played an important role in establishing the second-order phase-field theory as a valid model for brittle fracture. The important  $\Gamma$ -convergence results related to phase-field approximations for fracture are summarized from Giovanni [47] as follows:

**Definition 3.1.** ( $\Gamma$ -convergence) Let  $X$  be a metric space. A sequence of functionals,  $F_{\ell_0} : X \rightarrow [0, \infty]$ ,  $\ell_0 > 0$ , on  $X$  is said to  $\Gamma$ -converge to the  $\Gamma$ -limit  $F : X \rightarrow [0, \infty]$  if and only if

- (Lower bound inequality) For every  $u \in X$  and every sequence  $(u_{\ell_0}) \in X$  such that  $u_{\ell_0} \rightarrow u$  in  $X$  as  $\ell_0 \rightarrow 0$ ,

$$F(u) \leq \liminf_{\ell_0 \rightarrow 0} F_{\ell_0}(u_{\ell_0}) \quad (3.194)$$

- (Upper bound inequality) For every  $u \in X$ , there exists a sequence  $(u_{\ell_0}) \in X$  converging to  $u$  such that

$$F(u) \geq \limsup_{\ell_0 \rightarrow 0} F_{\ell_0}(u_{\ell_0}) \quad (3.195)$$

**Theorem 3.1.** *If  $F_{\ell_0}$   $\Gamma$ -converges to  $F$  and  $u_{\ell_0}$  minimizes  $F_{\ell_0}$  over  $X$ , then every cluster point of  $(u_{\ell_0})$  minimizes  $F$  over  $X$ .*

In the case of the phase-field model for brittle fracture,  $\Gamma$ -convergence implies that the minimizing solution to  $\tilde{\Psi}_c$  will converge to a minimizing solution of  $\Psi$  as  $\ell_0$

goes to zero. Chambolle [32] has provided a  $\Gamma$ -convergence proof for the case of brittle fracture in linear elasticity that shows that as  $\ell_0$  approaches zero the second-order approximate potential energy

$$\tilde{\Psi}_c = \int_{\Omega_0} (c^2 + k) \psi_e(\boldsymbol{\varepsilon}) d\boldsymbol{x} + \int_{\Omega_0} \mathcal{G}_c \left[ \frac{(1-c)^2}{4\ell_0} + \ell_0 |\nabla c|^2 \right] d\boldsymbol{x} \quad (3.196)$$

$\Gamma$ -converges to the total potential energy for Griffith's theory given by (3.2). The model parameter  $k \ll 1$  was introduced in Ambrosio and Tortorelli [4] as a way to avoid ill-posedness. Although it has been used in numerous numerical implementations and theoretical papers (see Bellettini and Coscia [16], Bourdin [23], Bourdin, Francfort, and Marigo [25], Giacomini [46], Bourdin [24], Del Piero, Lanchioni, and March [40], Lanchioni and Royer-Carfagni [62], Amor, Marigo, and Maurini [5], Burke, Ortner, and Süli [30]), Braides [28] proved later it was not necessary in order to obtain  $\Gamma$ -convergence. We have tested the necessity of setting  $k > 0$  in our formulation numerically and we have also found its inclusion to be unnecessary. In fact, all calculations presented in this dissertation set  $k = 0$  and it has not been included in the formulations presented here.

A  $\Gamma$ -convergence proof for the fourth-order theory presented in Section 3.4.2 has not yet been established. In Chapter 4, however, we show that this theory is well behaved at the length scales for  $\ell_0$  that are practical for computing.

# Chapter 4

## Analysis

In this chapter we analyze the phase-field model developed in Chapter 3. This analysis will help to establish important properties of the fourth-order approximation presented in Section 3.2.2.

### 4.1 Stress degradation function

The formulations presented in Chapter 3 have been derived in terms of a general stress degradation function,  $g : [0, 1] \rightarrow [0, +\infty)$ . This function is assumed to be increasing, lower semicontinuous with

$$g(0) = 0, \quad g(1) = 1, \quad g(c) > 0 \text{ for } c \neq 0, \quad g'(0) = 0, \quad g'(1) > 0. \quad (4.1)$$

The first and second condition are the limits for fully damaged and undamaged material, respectively. The third condition prevents a fully damaged state for  $c \neq 0$ . The fourth condition ensures that the energy converges to a finite value in the fully broken state, i.e.,  $c = 0$ . The fifth condition ensures that the phase-field will evolve when starting from a uniform initial state where  $c = 1$  everywhere. The necessity of the fifth condition will be made clear in the formulations that follow. A function meeting these requirements will satisfy the assumptions placed on the degradation function in the  $\Gamma$ -convergence proofs presented by Braides [28].

This function plays an important role in the evolution of both the stress and phase-field parameter. We will study the degradation function in terms of the one-dimensional homogeneous solution by ignoring all spatial derivatives of  $c$ . We will also restrict our study to small-strain kinematics. This approach has been previously utilized by others (see e.g. Bourdin [24], Amor, Marigo, and Maurini [5], Francfort, Le, and Serfaty [44], and Pham et al. [77]) to investigate the stress-strain response of damage-elastic models. Under the above assumptions, the one-dimensional coupled problem can be stated as

$$\left\{ \begin{array}{ll} \frac{d\sigma}{dx} = 0 & \text{on } \mathbb{R} \times ]0, T[ \\ \frac{g'(c)\ell_0 E \varepsilon^2}{\mathcal{G}_c} + c = 1 & \text{on } \mathbb{R} \times ]0, T[ \end{array} \right. \quad (4.2)$$

where the stress is given by  $\sigma = g(c)E\varepsilon$  with  $\varepsilon = du/dx$  being the strain. Note also that we are ignoring irreversibility in this section under the assumption of monotonically increasing loads.

#### 4.1.1 Quadratic degradation function

A simple quadratic degradation function that is common in the literature and meets the requirements stated above is

$$g(c) = c^2. \quad (4.3)$$

Substituting this into (4.2) we obtain

$$c_{\text{hom}} = \left( \frac{2\ell_0 E}{\mathcal{G}_c} \varepsilon_{\text{hom}}^2 + 1 \right)^{-1} \quad (4.4)$$

where  $c_{\text{hom}}$  and  $\varepsilon_{\text{hom}}$  are the homogeneous phase-field and strain, respectively. Substitution of this result into the constitutive equation yields the homogeneous stress as a

function of the homogeneous strain

$$\sigma_{\text{hom}} = \left( \frac{2\ell_0 E}{G_c} \varepsilon_{\text{hom}}^2 + 1 \right)^{-2} E \varepsilon_{\text{hom}}. \quad (4.5)$$

Figure 4.1(a) shows a characteristic plot of the homogeneous stress versus the homogeneous strain. Figure 4.1(b) shows the corresponding evolution of the homogeneous phase-field parameter. It is observed that as the strain is increased, initially the stress also increases. This increase in stress is accompanied by a decrease of the phase-field parameter. At some point, a critical stress level,  $\sigma_c$ , is reached after which both the stress and the phase-field decrease in value upon an increase in strain.

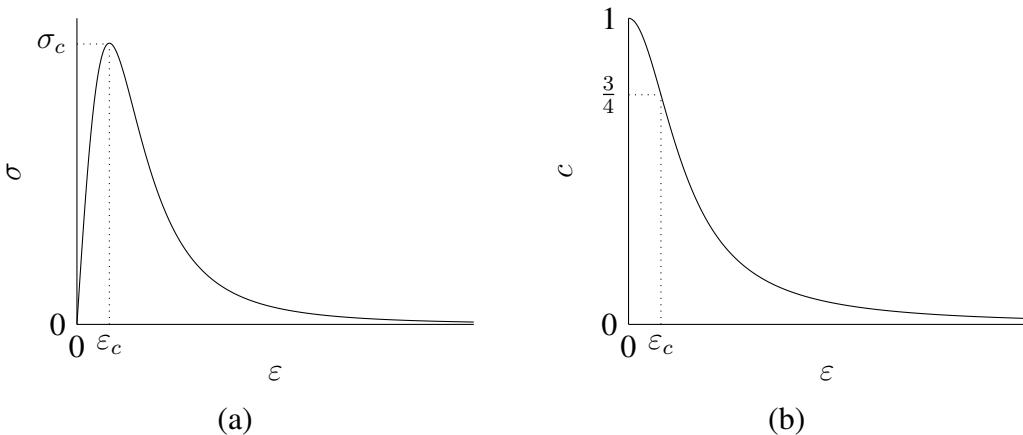


Figure 4.1: Uniaxial (a) stress-strain and (b) phase-field-strain curves for the homogeneous solution using the quadratic degradation function (4.3). Note that the value of  $\ell_0$  influences the maximum tensile stress, see (4.6).

The critical value for the stress, and corresponding value for the strain, are found as

$$\sigma_c = \frac{9}{16} \sqrt{\frac{E G_c}{6\ell_0}}, \quad \varepsilon_c = \sqrt{\frac{G_c}{6\ell_0 E}} \quad (4.6)$$

From these expressions it is clear that the critical stress will increase as  $\ell_0$  decreases. In the limit as  $\ell_0$  goes to zero, i.e., when the phase-field formulation coincides with

the discrete fracture formulation, the crack nucleation stress becomes infinite. This observation is consistent with the properties of Griffith's theory, which only allows for crack nucleation at stress singularities. It is interesting to note that the critical value for the phase-field is independent of the model and material parameters

$$c_c = \sqrt{\frac{\sigma_c}{E\varepsilon_c}} = \frac{3}{4}. \quad (4.7)$$

This implies that, no matter what parameters are used, a reduction in the phase-field of approximately 25 percent is established prior to crack nucleation at places where a crack emerges.

By virtue of the preceding analysis, we view  $\ell_0$  as a material parameter since it influences the critical stress. The issue as to whether or not  $\ell_0$  is a numerical regularization parameter or a material parameter has been discussed in Amor, Marigo, and Maurini [5]. See also the important results presented in Pham et al. [77].

#### 4.1.2 Cubic degradation function

One potential drawback of the quadratic degradation function presented above is the nonlinearity of the stress-strain curve prior to crack nucleation. This is evident in the stress-strain curve shown in Figure 4.1(a). This nonlinear behavior is a result of the decrease in  $c$  for small values of strain as shown in Figure 4.1(b). A consequence of this is that any non-zero strain will contribute to the computed crack surface energy term (see for example equation (3.8)). This includes strains that may be far from a crack. To address these issues we propose a parameterized cubic degradation function given by

$$g(c) = s(c^3 - c^2) + 3c^2 - 2c^3 \quad (4.8)$$

where  $s > 0$  determines the slope of  $g$  at  $c = 1$ . Note that the quadratic degradation function is a special case of this function for  $s = 2$ . Substituting this equation into (4.2)

and solving for the positive root of the phase-field equation, we get for  $s \neq 2$

$$c_{\text{hom}} = \frac{\ell_0 E \varepsilon_{\text{hom}}^2 (6 - 2s) + \mathcal{G}_c + \sqrt{[\ell_0 E \varepsilon_{\text{hom}}^2 (6 - 2s) + \mathcal{G}_c]^2 + 4\ell_0 E \varepsilon_{\text{hom}}^2 \mathcal{G}_c (3s - 6)}}{2\ell_0 E \varepsilon_{\text{hom}}^2 (6 - 3s)}. \quad (4.9)$$

This expression can be used to get an expression for the homogeneous stress in terms of the homogeneous strain and the slope parameter  $s$ . We can then compute the critical value of strain, which is given by

$$\varepsilon_c = \left( \sqrt{\frac{9 - 2s - \sqrt{s(18 - 5s)}}{(s - 3)^2}} \right) \sqrt{\frac{\mathcal{G}_c}{6\ell_0 E}}. \quad (4.10)$$

Similarly to the quadratic degradation function, the critical strain is a function of the material parameters and the additional slope parameter. We omit the full expression for the critical stress due to its complexity, but it can be written in the form

$$\sigma_c = f(s) \sqrt{\frac{E\mathcal{G}_c}{6\ell_0}} \quad (4.11)$$

where  $f(s)$  is a function of  $s$  only. Taking the limit as  $s \rightarrow 0$  we find

$$\lim_{s \rightarrow 0} \varepsilon_c = \sqrt{\frac{\mathcal{G}_c}{6\ell_0 E}}, \quad \lim_{s \rightarrow 0} \sigma_c = \sqrt{\frac{E\mathcal{G}_c}{6\ell_0}} = E \left( \lim_{s \rightarrow 0} \varepsilon_c \right), \quad \lim_{s \rightarrow 0} c_c = 1 \quad (4.12)$$

In other words, as  $s$  gets small the stress-strain relationship becomes linear up to the critical stress. This can be seen in Figure 4.2(a) where the stress-strain curve is plotted for  $s = 10^{-4}$ . As can be seen from Figure 4.2(b), this is a result of the nearly constant phase-field up to the critical strain. For comparison, the plots for the quadratic degradation function are included in the Figure. An analysis of numerical results using both the quadratic and cubic degradation function will be presented in Section 4.3.

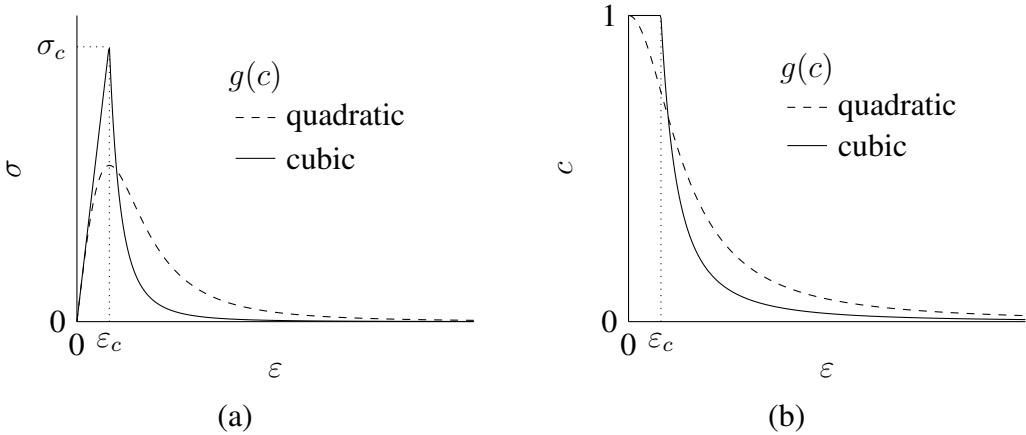


Figure 4.2: Uniaxial (a) stress-strain and (b) phase-field-strain curves for the homogeneous solution using the cubic degradation function (4.8). In this case, we have used  $s = 10^{-4}$ . For comparison, the dashed lines show the plots for the quadratic degradation function.

## 4.2 1D analysis

To illustrate various properties of the phase-field formulation for brittle fracture, we study the analytical solution to the boundary value problem for the second-order small-strain theory introduced in Chapter 3. We restrict our analysis to the one-dimensional domain  $\Omega \subset \mathbb{R}$  ( $d = 1$ ) and ignore all temporal derivatives. We assume the strain field to be non-compressive (i.e.,  $\psi_e^- = 0$  such that  $\psi_e = c^2 \frac{1}{2} E \varepsilon^2$ ).

### 4.2.1 Non-homogenous solution for the second-order theory

Interesting features of the model can be observed from the non-homogeneous solution of the one-dimensional static problem

$$\left\{ \begin{array}{ll} \frac{d\sigma}{dx} = 0 & \text{on } \mathbb{R} \times ]0, T[ \\ \frac{g'(c)\ell_0 E \varepsilon^2}{\mathcal{G}_c} + c - 4\ell_0^2 \frac{d^2 c}{dx^2} = 1 & \text{on } \mathbb{R} \times ]0, T[. \end{array} \right. \quad (4.13)$$

Letting  $g(c) = c^2$  and using  $\sigma = c^2 E \varepsilon$ , we can combine the constitutive equation for the stress and phase-field equations to yield the non-linear ordinary differential equation

$$\left( \frac{2\ell_0\sigma^2}{c^4 E G_c} + 1 \right) c - 4\ell_0^2 \frac{d^2 c}{dx^2} = 1. \quad (4.14)$$

A solution for the phase-field,  $c(x)$ , with a crack at  $x = 0$  is found by supplementing this differential equation with the boundary condition  $\lim_{x \rightarrow \pm\infty} c(x) = c_{\text{hom}}(\sigma)$ , which implies that the phase-field gradient vanishes far away from the crack. It is important to note that the homogeneous solution to the phase-field is dependent on the stress,  $\sigma$ , as elaborated in the previous section. In addition to the far-field boundary condition, we require the solution to be symmetric and differentiable at every point except for  $x = 0$ .

The first step in finding the non-homogeneous solution to the phase-field problem is to multiply equation (4.14) with  $\frac{dc}{dx}$  and make use of the fact that, by (4.13),  $\sigma$  is constant, to obtain

$$\frac{d}{dx} \left[ -\frac{\ell_0\sigma^2}{c^2 E G_c} + \frac{c^2}{2} - 2\ell_0^2 \left( \frac{dc}{dx} \right)^2 - c \right] = 0. \quad (4.15)$$

Since we require the solution to be symmetric around  $x = 0$ , we integrate this expression from  $x$  to infinity for positive values of  $x$ , and from minus infinity to  $x$  when  $x$  is negative. Since we have specified the crack to be centered at  $x = 0$ , we require the phase-field to have a minimum at  $x = 0$ . From these requirements, we obtain

$$\frac{dc}{dx} = \text{sgn}(x) \sqrt{\frac{1}{2\ell_0^2} \left( -\frac{\ell_0\sigma^2}{c^2 E G_c} + \frac{c^2}{2} - c - a \right)} \quad (4.16)$$

where the coefficient  $a$  follows from the far-field boundary condition as

$$a = -\frac{\ell_0\sigma^2}{c_{\text{hom}}^2 E G_c} + \frac{c_{\text{hom}}^2}{2} - c_{\text{hom}}. \quad (4.17)$$

By definition, substitution of the homogeneous solution  $c_{\text{hom}}(\sigma)$  into equation (4.16) yields a zero phase-field gradient. An analytical non-homogeneous solution to equation

(4.14) can be found for the case of a fully developed crack, i.e. for  $\sigma = 0$  (no stress carrying capability) and  $c_{\text{hom}}(0) = 1$ . In this case, equation (4.16) reduces to

$$\frac{dc}{dx} = \text{sgn}(x) \frac{1-c}{2\varepsilon}. \quad (4.18)$$

and, assuming  $c(0) = 0$ , we find

$$c = 1 - \exp\left(-\frac{|x|}{2\ell_0}\right) \quad (4.19)$$

as the solution that satisfies the specified boundary conditions. Note that this is the exact function used by Miehe et al. [68] to derive their phase field formulation. This function is exactly the construction of the recovery sequence for the  $\Gamma$ -convergence proof, originally given in Ambrosio and Tortorelli [4]. Note that its construction is also justified in much less technical terms in Bourdin, Francfort, and Marigo [26, Section 8.1.1].

The unique non-homogeneous solution can also be constructed for  $\sigma > 0$ . In this case, a second admissible phase-field value, smaller than the homogeneous solution,  $c_{\text{hom}}$ , for which the gradient is equal to zero can be found. This phase-field value corresponds to the value of the phase-field at the center of the crack, and is denoted by  $c_{\text{crack}}$ . Since the phase-field increases monotonically from  $c_{\text{crack}}$  at the center of the crack (at  $x = 0$ ) to  $c_{\text{hom}}$  far away from the crack ( $x = \pm\infty$ ), we find the coordinates  $0 < \pm x < \infty$  at which the phase-field is equal to  $c_{\text{crack}} < c(x) < c_{\text{hom}}$  by evaluation of

$$x = \pm \int_{c_{\text{crack}}}^{c(x)} \left[ \frac{1}{2\ell_0^2} \left( -\frac{\ell_0 \sigma^2}{\bar{c}^2 E G_c} + \frac{\bar{c}^2}{2} - \bar{c} - a \right) \right]^{-\frac{1}{2}} d\bar{c}. \quad (4.20)$$

In non-dimensional form with the non-dimensional constant  $C_h = 1$  (see Appendix A),

(4.20) becomes

$$x^* = \pm \int_{c_{\text{crack}}}^{c(L_0 x^*)} \left[ \frac{1}{2(\ell_0^*)^2} \left( -\frac{\ell_0^*(\sigma^*)^2}{\bar{c}^2} + \frac{\bar{c}^2}{2} - \bar{c} - a^* \right) \right]^{-\frac{1}{2}} d\bar{c}. \quad (4.21)$$

Evaluating this integral numerically for  $\sigma = \beta\sigma_c$  with various values of  $\beta \in ]0, 1[$  we get the solutions shown in Figure 4.3 (note that the presented results have again been non-dimensionalized and merely depend on the parameter  $\beta$ ). It is interesting to note that, except for the limiting case of a fully developed crack, the solution to the phase-field is smooth at the center of the crack.

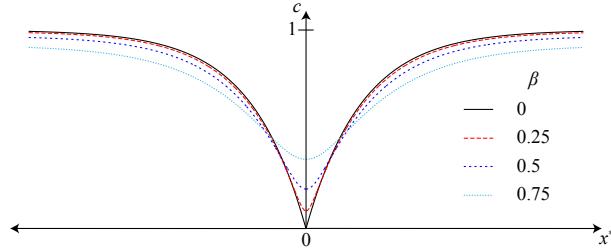


Figure 4.3: One-dimensional solution of the phase field formulation for various values of the stress ratio  $\beta = \frac{\sigma}{\sigma_c} = \frac{\sigma^*}{\sigma_c^*}$ . Note that, except for the limiting case  $\beta = 0$ , the phase-field is smooth at the center of the crack ( $x^* = 0$ ).

For further results on non-homogeneous solutions of one-dimensional problems see Pham and Marigo [76] and Pham, Amor, Marigo, and Maurini [77]

#### 4.2.2 Convergence study of the phase-field equation

In the next two sections we will examine the one-dimensional phase-field equation in isolation from the linear momentum equation. We will proceed by assuming a given displacement or strain field and then examine certain properties of the resulting phase-field. We start by letting  $\mathcal{G}_c = 1$ . This leads to the following second and fourth-order problems defined on the domain  $\Omega = (-L, L)$ .

**Second-order theory:**

$$\left\{ \begin{array}{l} (4\ell_0\mathcal{H} + 1)c - 4\ell_0^2 c'' = 1 \quad -L < x < L \\ c'(-L) = c'(L) = 0 \end{array} \right. \quad (4.22)$$

**Fourth-order theory:**

$$\left\{ \begin{array}{l} (4\ell_0\mathcal{H} + 1)c - 2\ell_0^2 c'' + \ell_0^4 c^{(4)} = 1 \quad -L < x < L \\ c''(-L) = c''(L) = 0 \\ \ell_0^2 c^{(3)}(-L) - 2c'(-L) = 0 \\ \ell_0^2 c^{(3)}(L) - 2c'(L) = 0 \end{array} \right. \quad (4.23)$$

In this section we will study the convergence properties of the solution of the discrete Galerkin problems corresponding to (4.22) and (4.23). We begin by defining the reaction term,  $\mathcal{H}$ , as

$$\mathcal{H}(x) = \begin{cases} 0 & x < -2\ell_0 \\ B & -2\ell_0 \leq x < 2\ell_0 \\ 0 & x \geq 2\ell_0 \end{cases} \quad (4.24)$$

and choose  $L = 1$ ,  $B = 500$ , and  $\ell_0 = 0.025$ . This choice of  $\mathcal{H}$  allows us to find an exact solution to both (4.22) and (4.23) so that we can compute convergence rates. The exact solutions will be in the Sobolev spaces  $H^2(-1, 1)$  and  $H^4(-1, 1)$  for the second and fourth-order theory respectively.

For comparison, we compute the theoretical convergence rates by the Aubin-Nitsche method. The relevant results are presented by Strang and Fix [97], among others, and are summarized as follows: Let  $2m$  be the largest number of derivatives in the strong form of the PDE that is being approximated. Then, suppose that the finite element approximation space is composed of piecewise polynomials of degree  $k - 1$ . Letting  $c^h$

be the finite element approximation to the exact solution  $c$  and defining  $e = c - c^h$  to be the error, the optimal convergence rate is

$$\|e\|_s \leq Ch^\alpha \|c\|_k \quad (4.25)$$

$$\alpha = \begin{cases} k-s & s \geq 2m-k \\ 2(k-m) & s \leq 2m-k \end{cases} \quad (4.26)$$

where  $\|\cdot\|_s$  is the norm associated with the  $H^s$  Sobolev space,  $h$  is the mesh size, and  $C$  is a constant independent of  $h$ . Note that for the second-order theory  $k = 2$  and for the fourth-order theory  $k = \{3, 4\}$ . For the second-order theory the approximation predicts quadratic convergence in the  $L_2$  norm and linear convergence in the  $H^1$  norm for  $p \geq 1$ . Convergence rates for the fourth-order theory are given in Table 4.1.

		$s$		
		0	1	2
$p = 2$		2	2	1
$p \geq 3$		4	3	2

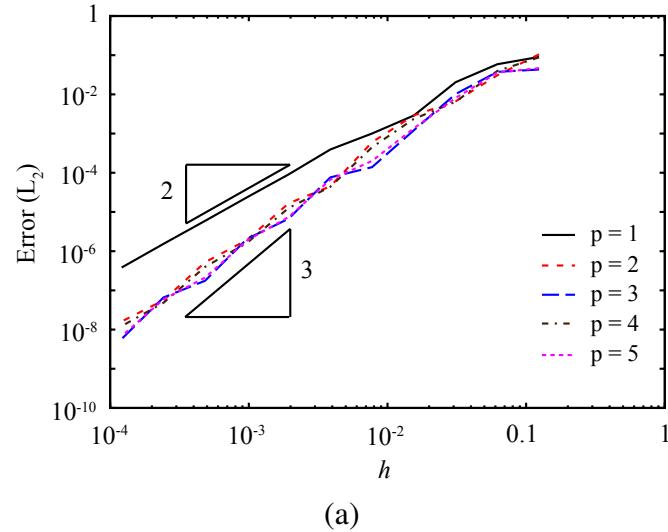
Table 4.1: Predicted convergence rates in the  $H^s$  norm for the fourth-order problem for piecewise polynomials of degree  $p$ .

The convergence results for the numerical solutions to (4.22) and (4.23) are shown in Figures 4.4 – 4.6. In all cases the domain has been discretized with uniform B-spline basis functions with no repeated knots. Notice that the second-order theory achieves slightly higher convergence rates than expected. This is not fully understood but may be due to local effects where the exact solution is smooth ( $C^\infty$ ). The results for the fourth-order theory match well with the predicted rates.

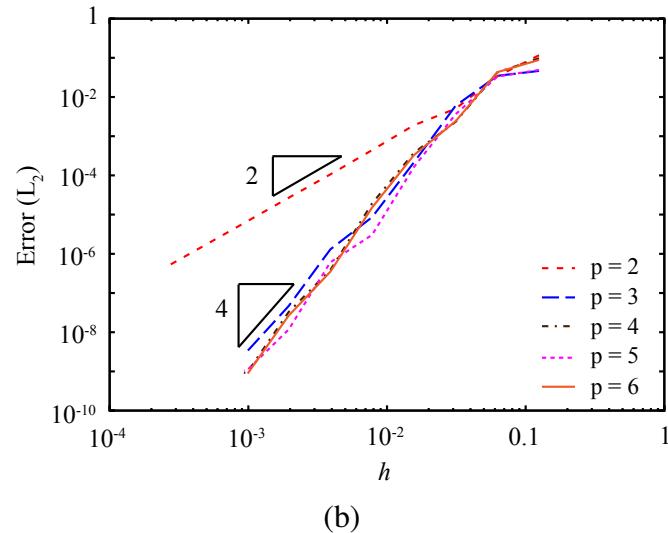
### 4.2.3 Surface energy approximation

One quantity that is important for interpreting numerical results is the effective critical energy release rate defined as

$$\mathcal{G}_c^{\text{eff}} = \frac{\tilde{\Psi}_c}{a} \quad (4.27)$$

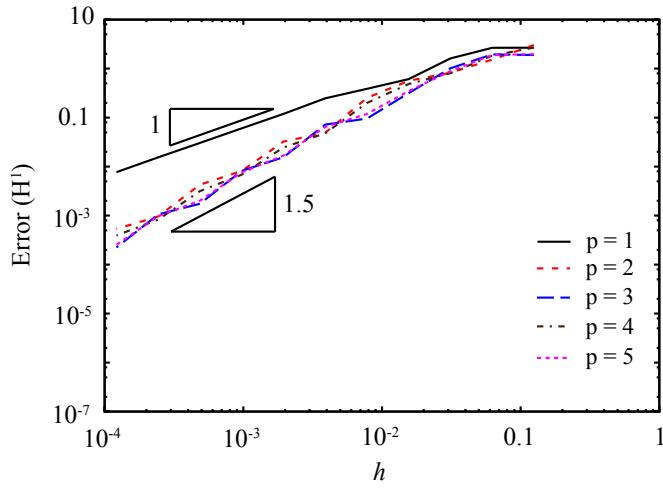


(a)

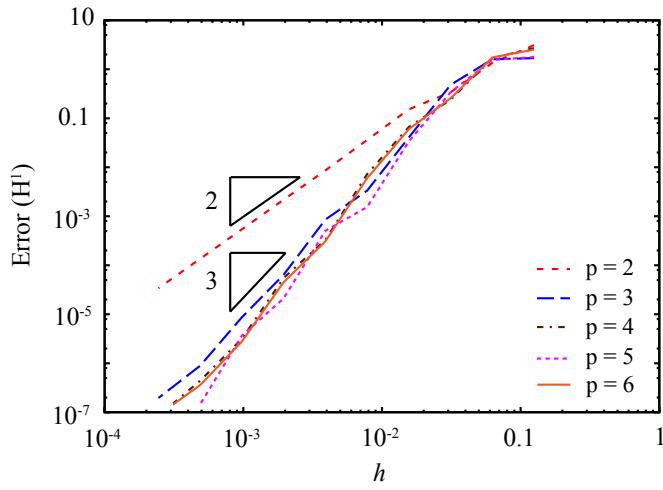


(b)

Figure 4.4: Convergence plots of the error in the  $L_2$  norm for (a) the second-order phase-field and (b) the fourth-order phase-field.



(a)



(b)

Figure 4.5: Convergence plots of the error in the  $H^1$  norm for (a) the second-order phase-field and (b) the fourth-order phase-field.

where  $\tilde{\Psi}_c$  is the energy associated with the crack surfaces defined in (3.3) and  $a$  is the total crack length (in one dimension  $a = 1$  for each crack). For the second-order phase-field theory, Bourdin et al. [26] showed that when discretizing with linear elements  $\tilde{\Psi}_c$  effectively amplifies the critical energy release rate by a factor of roughly  $1 + h/4\ell_0$ .

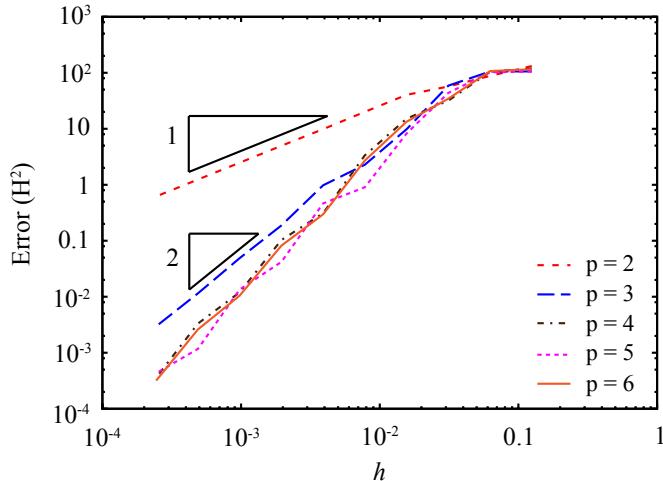


Figure 4.6: Convergence plots of the error in the  $H^2$  norm for the fourth-order phase-field.

Using this estimate, we define the predicted effective critical energy release rate as

$$\mathcal{G}_c^{\text{pred}} = \left(1 + \frac{h}{4\ell_0}\right) \mathcal{G}_c. \quad (4.28)$$

No estimates have yet been developed for the fourth-order theory but we provide a numerical estimate below.

#### 4.2.3.1 Second-order theory

We begin with the second-order theory as a reference point. We let  $\Omega = (-100, 100)$  and assume a crack exists at  $x = 0$ . For the ideal fully developed crack, i.e.,  $du/dx = 0$  for  $x \neq 0$ , we can compute the optimal phase-field profile (for the right half of the domain) from the following boundary value problem:

$$\left\{ \begin{array}{ll} c - 4\ell_0^2 c'' = 1 & 0 < x < 100 \\ c(0) = 0 \\ c'(100) = 0 \end{array} \right. \quad (4.29)$$

Substituting the solution to this problem into the surface energy term we get

$$\tilde{\Psi}_c = 2\mathcal{G}_c \int_0^{100} \left\{ \frac{(1-c)^2}{4\ell_0} + \ell_0(c')^2 \right\} dx = \mathcal{G}_c \left( \frac{\exp\left(\frac{100}{\ell_0}\right) - 1}{\exp\left(\frac{100}{\ell_0}\right) + 1} \right). \quad (4.30)$$

It is clear from this expression that as  $\ell_0$  goes to zero the right hand side goes to  $\mathcal{G}_c$ , i.e., we recover the exact surface energy.

To study the numerical approximation to the problem above, we define the idealized continuous piecewise linear displacement field

$$u = \begin{cases} 0 & x < -h \\ \frac{u_0}{2h}(x+h) & -h \leq x < h \\ u_0 & x \geq h \end{cases} \quad (4.31)$$

where  $h$  and  $u_0$  are given parameters. From this we get

$$\frac{du}{dx} = \begin{cases} 0 & x < -h \\ \frac{u_0}{2h} & -h \leq x < h \\ 0 & x \geq h \end{cases} \quad (4.32)$$

and

$$\psi_e^+ = \begin{cases} 0 & x < -h \\ \frac{1}{2}E \left( \frac{u_0}{2h} \right)^2 & -h \leq x < h \\ 0 & x \geq h. \end{cases} \quad (4.33)$$

As  $h$  goes to zero this strain energy converges to  $du/dx = 0$  for  $x \neq 0$ , which is the strain energy that was assumed for the boundary value problem (4.29).

Letting  $E = 1$  and  $\mathcal{G}_c = 1$  and using (4.33) we find the solution to

$$\begin{cases} (4\ell_0\psi_e^+ + 1)c - 4\ell_0^2 c'' = 1 & -100 < x < 100 \\ c'(-100) = c'(100) = 0 \end{cases} \quad (4.34)$$

where we have chosen  $\ell_0 = 1/8$  and  $u_0 = 10$ . This problem was solved numerically using one-dimensional linear finite elements, and the domain was discretized with 819,200 uniform intervals. The results showing the error in the effective critical energy release rate,  $\mathcal{G}_c^{\text{eff}} - 1$ , are shown in Table 4.2. As expected, the solution is converging linearly with  $h$ . For comparison, the predicted error in the critical strain energy release rate,  $\mathcal{G}_c^{\text{pred}} - 1$ , is included in the table.

#### 4.2.3.2 Fourth-order theory

We follow the same strategy for the fourth-order theory. Starting with the same idealized crack assumption as above we can compute the optimal phase-field profile from the following boundary value problem:

$$\left\{ \begin{array}{ll} c - 2\ell_0^2 c'' + \ell_0^4 c^{(4)} = 1 & 0 < x < 100 \\ c(0) = 0 \\ c'(0) = 0 \\ c''(100) = 0 \\ \ell_0^2 c^{(3)}(100) - 2c'(100) = 0. \end{array} \right. \quad (4.35)$$

Substituting the solution to this problem into the surface energy term we get

$$\begin{aligned} \tilde{\Psi}_c &= 2\mathcal{G}_c \int_0^{100} \left\{ \frac{(1-c)^2}{4\ell_0} + \frac{\ell_0}{2}(c')^2 + \frac{\ell_0^3}{4}(c'')^2 \right\} dx \\ &= 2\mathcal{G}_c \left( \frac{3\ell_0^2 \exp\left(\frac{400}{\ell_0}\right) - 400\ell_0 \exp\left(\frac{200}{\ell_0}\right) - 3\ell_0^2}{6\ell_0^2 \exp\left(\frac{400}{\ell_0}\right) + 20 \exp\left(\frac{200}{\ell_0}\right) (4000 + \ell_0^2) + 6\ell_0^2} \right). \end{aligned} \quad (4.36)$$

Considering the dominant terms of the numerator and denominator it is clear that the right hand side converges to  $\mathcal{G}_c$  as  $\ell_0$  goes to zero.

The numerical solution of the fourth-order theory requires a basis with continuous first derivatives. Assuming a displacement field that is in the same function space

as the phase-field we define the idealized piecewise quadratic displacements field to be

$$u = \begin{cases} 0 & x < -h \\ \frac{u_0}{2h^2}(x^2 + 2hx + h^2) & -h \leq x < 0 \\ \frac{u_0}{2h^2}(-x^2 + 2hx + h^2) & 0 \leq x < h \\ u_0 & x \geq h \end{cases} \quad (4.37)$$

where  $h$  and  $u_0$  are given parameters. From this we get

$$\frac{du}{dx} = \begin{cases} 0 & x < -h \\ \frac{u_0}{h^2}(h+x) & -h \leq x < 0 \\ \frac{u_0}{h^2}(h-x) & 0 \leq x < h \\ 0 & x \geq h \end{cases} \quad (4.38)$$

and

$$\psi_e^+ = \begin{cases} 0 & x < -h \\ \frac{1}{2}E \left[ \frac{u_0}{h^2}(h+x) \right]^2 & -h \leq x < 0 \\ \frac{1}{2}E \left[ \frac{u_0}{h^2}(h-x) \right]^2 & 0 \leq x < h \\ 0 & x \geq h. \end{cases} \quad (4.39)$$

Again, as  $h$  goes to zero this strain energy converges to  $du/dx = 0$  for  $x \neq 0$ , which is the strain energy that was assumed for the boundary value problem (4.35).

Letting  $E = 1$  and  $\mathcal{G}_c = 1$  and using (4.39) we find the solution to

$$\left\{ \begin{array}{l} (4\ell_0\psi_e^+ + 1)c - 2\ell_0^2 c'' + \ell_0^4 c^{(4)} = 1 \quad -100 < x < 100 \\ c''(-100) = c''(100) = 0 \\ \ell_0^2 c^{(3)}(-100) - 2c'(-100) = 0 \\ \ell_0^2 c^{(3)}(100) - 2c'(100) = 0 \end{array} \right. \quad (4.40)$$

where we have again chosen  $\ell_0 = 1/8$  and  $u_0 = 10$ . This problem was solved numerically using a one-dimensional quadratic B-spline, and the domain was discretized

with 819,200 uniform knot spans. The results showing the error in the effective critical energy release rate,  $\mathcal{G}_c^{\text{eff}} - 1$ , are shown in Table 4.2. These results indicate that the convergence rate of the error in the surface energy for the fourth-order theory is approaching  $O(h^3)$  or better.

h	2nd-order			4th-order	
	$\mathcal{G}_c^{\text{pred}} - 1$	Error ( $\mathcal{G}_c^{\text{eff}} - 1$ )	Conv. Rate	Error ( $\mathcal{G}_c^{\text{eff}} - 1$ )	Conv. Rate
$\ell_0$	$2.500 \times 10^{-1}$	$4.180 \times 10^{-1}$		$2.722 \times 10^{-1}$	
$\ell_0/2$	$1.250 \times 10^{-1}$	$2.119 \times 10^{-1}$	0.980	$1.106 \times 10^{-1}$	1.299
$\ell_0/4$	$6.250 \times 10^{-2}$	$1.059 \times 10^{-1}$	1.001	$2.938 \times 10^{-2}$	1.912
$\ell_0/8$	$3.125 \times 10^{-2}$	$5.305 \times 10^{-2}$	0.997	$4.837 \times 10^{-3}$	2.603
$\ell_0/16$	$1.563 \times 10^{-2}$	$2.655 \times 10^{-2}$	0.999	$6.864 \times 10^{-4}$	2.817
$\ell_0/32$	$7.813 \times 10^{-3}$	$1.329 \times 10^{-2}$	0.998	$1.159 \times 10^{-4}$	2.566
$\ell_0/64$	$3.906 \times 10^{-3}$	$6.667 \times 10^{-3}$	0.995	$1.730 \times 10^{-5}$	2.744
$\ell_0/128$	$1.953 \times 10^{-3}$	$3.377 \times 10^{-3}$	0.981	$1.724 \times 10^{-6}$	3.327

Table 4.2: Error in the computed surface energy for the one-dimensional problems shown in (4.34) and (4.40). For these results we have used  $\ell_0 = 1/8$ .

**Remark 4.1.** Given the piecewise constant strain energy defined in (4.32) it is possible to find the exact solutions to (4.34). The errors computed from the exact solution agree with the numerical solution up to the number of digits shown in the table above. For the fourth-order theory with the strain energy defined by (4.39), however, it would be difficult to find an exact solution to (4.40). For this reason, we have chosen to solve the problems numerically using isogeometric analysis.

#### 4.2.4 Strain energy approximation

Having shown the approximation properties of the phase-field equation in the previous two sections we now focus on the approximation of the strain energy term. We begin our analysis with the exact solution for the phase-field from (4.29) and (4.35) for

the second and fourth-order theories respectively. Given the exact phase-field, we wish to solve the following boundary value problem for the displacements:

$$\left\{ \begin{array}{l} \frac{d}{dx} \left( c^2 E \frac{du}{dx} \right) = 0 \quad -L < x < L \\ u(-L) = 0 \\ u(L) = u_0. \end{array} \right. \quad (4.41)$$

We get the exact solution for this problem by noting that  $c(0) = 0$  and  $c^2 du/dx$  is constant, therefor it must be equal to zero. This implies that  $du/dx = 0$  for  $x \neq 0$ . It follows that the exact displacements are given by

$$u = \begin{cases} 0 & L \leq x < 0 \\ u_0 & 0 < x \leq L. \end{cases} \quad (4.42)$$

To study the approximation to the strain energy, we define the strain energy norm as:

$$\|u\|_e^2 = \int_{-L}^L c^2 E \left( \frac{du}{dx} \right)^2 dx. \quad (4.43)$$

We now let  $u$  be the exact solution given in (4.42),  $\mathcal{S}^h$  be a finite-dimensional trial function space, and  $u^h \in \mathcal{S}^h$  be the solution of the Galerkin finite element problem. By the best approximation property of the Galerkin method we know

$$\|u^h - u\|_e^2 \leq \|w^h - u\|_e^2 \quad \forall w^h \in \mathcal{S}^h. \quad (4.44)$$

For the second-order theory we choose  $\mathcal{S}^h$  to be the set of piecewise linear functions and then prescribe the displacements to be

$$w^h = \begin{cases} 0 & -L \leq x < -h \\ \frac{u_0}{2h}(h + x) & -h \leq x < h \\ u_0 & h \leq x \leq L. \end{cases} \quad (4.45)$$

By the best approximation property the error given by this displacement will bound the error of the actual discrete solution. Now, taking advantage of the symmetry of the phase-field and the strain energy about  $x = 0$  we get

$$\|w^h - u\|_e^2 = 2 \int_0^L c^2 E \left( \frac{dw^h}{dx} - \frac{du}{dx} \right)^2 dx = 2 \int_0^h c^2 E \left( \frac{u_0}{2h} \right)^2 dx. \quad (4.46)$$

Taking the Taylor expansion of the second-order phase-field (3.7) and considering only the dominant term we get that  $c$  is proportional to  $x$  near  $x = 0$ . From this we get

$$\|w^h - u\|_e^2 = C \int_0^h x^2 \left( \frac{1}{h} \right)^2 dx = Ch \quad (4.47)$$

where  $C$  is a constant independent of  $h$ . Thus, given the exact solution for the second-order phase-field theory, the convergence rate of the strain energy is bounded by  $1/2$ .

A similar analysis can be carried out for the fourth-order theory. In this case we take  $\mathcal{S}^h$  to be the set of piecewise quadratic functions with continuous first derivatives (e.g., quadratic B-splines) and prescribe the displacements to be

$$w^h = \begin{cases} 0 & -L \leq x < -h \\ \frac{u_0}{2h^2}(x^2 + 2hx + h^2) & -h \leq x < 0 \\ \frac{u_0}{2h^2}(-x^2 + 2hx + h^2) & 0 \leq x < h \\ u_0 & h \leq x \leq L. \end{cases} \quad (4.48)$$

Again, taking advantage of symmetry we get

$$\begin{aligned} \|w^h - u\|_e^2 &= 2 \int_0^L c^2 E \left( \frac{dw^h}{dx} \right)^2 dx = 2 \int_0^h c^2 E \left[ \frac{u_0}{h^2}(h-x) \right]^2 dx \\ &\leq 2 \int_0^h c^2 E \left( \frac{u_0}{h} \right)^2 dx. \end{aligned} \quad (4.49)$$

Taking the Taylor expansion of the fourth-order phase-field (3.11) and considering only the dominant term we get that  $c$  is proportional to  $x^2$  near  $x = 0$ . From this we get

$$\|w^h - u\|_e^2 \leq C \int_0^h x^4 \left( \frac{1}{h} \right)^2 dx = Ch^3 \quad (4.50)$$

where  $C$  is again a constant independent of  $h$ . Thus, given the exact solution for the fourth-order phase-field theory, the convergence rate of the strain energy is bounded by  $3/2$ .

Figure 4.7 shows convergence plots for the finite element solution to (4.41) with  $E = 100$ ,  $\ell_0 = 1/8$ ,  $L = 5$  and  $u_0 = 10^{-5}$ . Piecewise linear basis function were used for the second-order problem and quadratic B-splines for the fourth-order problem. The computed results match well with the estimates presented above.

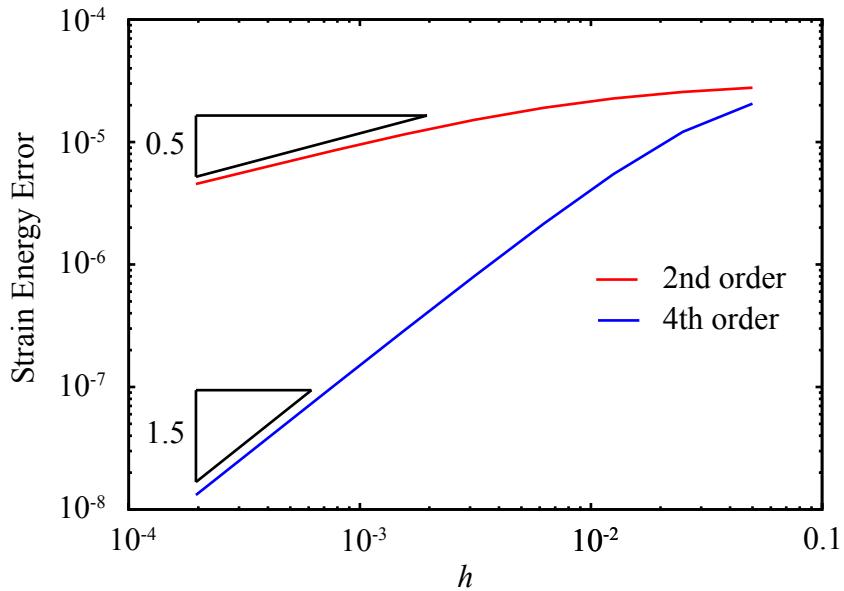


Figure 4.7: Convergence plots for the error in the computed strain energy term.

**Remark 4.2.** These results suggest that in numerical calculations the crack will always carry some residual non-zero stress. We typically use meshes such that  $\ell_0/h = 2$ . At this ratio the stresses in the crack may be large enough to act as a cohesive crack. In fact, at the length scales that can be realistically achieved for numerical calculations one might be justified in considering the phase-field model to be a cohesive crack model. This is a topic that deserves further investigation.

**Remark 4.3.** If we let  $\mathcal{S}^h$  be the set of piecewise functions of polynomial order  $p$ , we can use the inverse estimate

$$\left\| \frac{dw^h}{dx} \right\|_{L^\infty} \leq \frac{C(p)}{h} \|w^h\|_{L^\infty} \quad (4.51)$$

to get

$$\frac{dw^h}{dx} - \frac{du}{dx} \propto \frac{1}{h}. \quad (4.52)$$

Thus, for an  $n$ th-order phase-field theory the bound on the error in the strain energy norm is

$$\|w^h - u\|_e^2 = Ch^{(n-1)} \quad (4.53)$$

which gives a convergence rate of  $(n - 1)/2$ .

### 4.3 Double cantilever beam

In this section we present results for simulations of the two-dimensional double cantilever beam problem shown in Figure 4.8. When displacement boundary conditions are applied to this configuration as shown in the figure the crack growth is stable, i.e., small increases in displacements lead to small increases in crack length. This configuration can also be analyzed using standard linear elastic fracture mechanics, providing benchmark results that we can then compare with results from the phase-field model.

The material properties used for the analyses in this section are shown in Table 4.3 and plane strain conditions are assumed. We have used the small strain brittle fracture formulation presented in Section 3.4 and have also assumed quasi-static behavior. In all cases, the mesh is constructed as a contiguous (no discontinuities at the crack) quadratic B-spline. Non-uniform knot spacing has been used to provide more

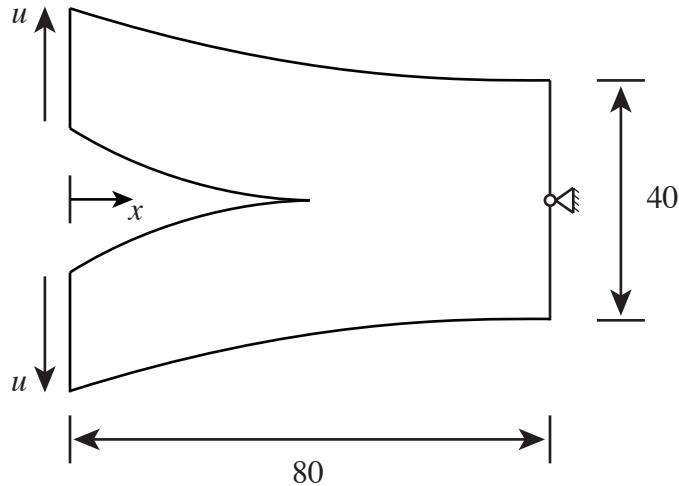


Figure 4.8: The geometry and boundary conditions for the double cantilever beam problem.

mesh resolution near the crack with fewer elements than a uniform mesh would require. All results are computed using the monolithic strategy discussed in Section 3.7.1. We assume that the solution is dominated by tensile strains and use a strain energy that is not split, i.e.,  $\psi_e^+ = \psi_e$  and  $\psi_e^- = 0$ .

Property	Value
E	100
$\nu$	0.25
$\mathcal{G}_c$	0.01
$\ell_0$	1/2

Table 4.3: Material properties.

In order to initiate phase-field localization in the contiguous mesh we set a Dirichlet condition on the phase-field. This condition forces  $c = 0$  along the crack for  $0 < x < 40$ . We then displace the left ends of the beam until a sufficient amount of strain localization has occurred to maintain the phase-field localization. The vertical displacements of the left boundary are then fixed and the Dirichlet condition is removed to allow the solution to reach an unconstrained (in the phase-field) state.

With the initial crack established, the displacements of the left ends are again increased. For a given displacement we define the crack tip as the largest  $x$  coordinate of the phase-field contour for  $c = 0.2$ . Starting from an initial displacement,  $u_0$ , we add 500 increments of size  $10^{-4}$  to get a total change in displacement of  $\Delta u = 0.05$ . We then measure the change in crack length,  $\Delta a$ , and compute the change in surface energy,  $\tilde{\Psi}_c = \tilde{\Psi}_c(u_0 + \Delta u) - \tilde{\Psi}_c(u_0)$ , the change in strain energy,  $\tilde{\Psi}_{SE} = \tilde{\Psi}_{SE}(u_0 + \Delta u) - \tilde{\Psi}_{SE}(u_0)$ , and the work done by the boundary conditions,  $W_{\Delta u}$ , with

$$\tilde{\Psi}_c = \int_{\Omega} \mathcal{G}_c \Gamma_{c,n} d\mathbf{x} \quad (4.54)$$

$$\tilde{\Psi}_{SE} = \int_{\Omega} (c^2 \psi^+ + \psi^-) d\mathbf{x} \quad (4.55)$$

$$W_{\Delta u} = 2 \int_{u_0}^{u_0 + \Delta u} F^{\text{ext}}(u - u_0) du \quad (4.56)$$

where  $F^{\text{ext}} = F^{\text{ext}}(u)$  is the total external force required to produce the given displacement of the upper half of the beam.

Simulations were preformed for three levels of mesh refinement,  $h_{\min} = \{\ell_0/2, \ell_0/4, \ell_0/8\}$ . For each mesh, six variations of the phase-field model were studied. The data computed for each case are shown in Table 4.4. For the quadratic degradation function we have used  $u_0 = 0.431$  and for the cubic degradation function  $u_0 = 0.425$  with  $s = 10^{-4}$  ( $s$  being the slope of the cubic degradation function at  $c = 1$ ). For the reversible results we have placed no constraint on  $\dot{c}$  (i.e.,  $\dot{c} > 0$  is allowed). The irreversible results use the strain history functional discussed in Section 3.6. The 2nd and 4th order results refer to the second and fourth-order theories discussed in Sections 3.4.1 and 3.4.2 respectively.

In Table 4.5 we compute the error for two quantities for each simulation. The first is the error in the effective critical energy release rate,  $\mathcal{G}_c^{\text{eff}} = \Delta \tilde{\Psi}_c / \Delta a$ . Recall

		$h_{\min}$	$a_0$	$\Delta a$	$\Delta \tilde{\Psi}_c$	$\Delta \tilde{\Psi}_{SE}$
Quad. deg. func.	Reversible	2nd	$\ell_0/2$	39.077	2.866	0.0387494
			$\ell_0/4$	40.523	3.025	0.0356936
			$\ell_0/8$	41.321	3.120	0.0341935
	4th		$\ell_0/2$	40.514	2.962	0.0368772
			$\ell_0/4$	41.612	3.119	0.0340304
			$\ell_0/8$	42.023	3.201	0.0329217
	Irrev.	2nd	$\ell_0/2$	40.346 <sup>a</sup>	2.128	0.0348035
			$\ell_0/4$	40.525	2.176	0.0312358
			$\ell_0/8$	41.036	2.303	0.0303668
Cubic deg. func.	Reversible	2nd	$\ell_0/2$	40.495	3.073	0.0408824
			$\ell_0/4$	42.076	3.233	0.0376239
			$\ell_0/8$	42.960	3.340	0.0361355
	4th		$\ell_0/2$	41.651	3.137	0.0392788
			$\ell_0/4$	42.944	3.296	0.0365048
			$\ell_0/8$	43.511	3.391	0.0350732
	Irrev.	2nd	$\ell_0/2$	40.524	3.079	0.0422469
			$\ell_0/4$	41.878	3.144	0.0402489
			$\ell_0/8$	42.605	3.211	0.0385206

<sup>a</sup> For the second-order irreversible case using the quadratic degradation function we have used  $u_0 = 0.453$  for the  $h_{\min} = \ell_0/2$ .

Table 4.4: The data for each variation of the phase-field model studied in this section. For the quadratic degradation function we have used  $u_0 = 0.431$  and for the cubic degradation function we have used  $u_0 = 0.425$  with  $s = 10^{-4}$ . The crack tip is taken as the contour of the phase-field at  $c = 0.2$ .

from Section 4.2.3 that we expect  $\mathcal{G}_c^{\text{eff}}$  to converge linearly to  $\mathcal{G}_c$  for the second-order theory and faster for the fourth-order theory (see Table 4.2). These trends are apparent for the reversible case. For the irreversible case, however, the rate is much lower than expected. This is not surprising since the surface energy term,  $\tilde{\Psi}_c$ , is dependent on the strain history. In general,  $\mathcal{G}_c^{\text{eff}}$  will not converge to  $\mathcal{G}_c$  for the irreversible case.

Table 4.5 also shows the error in the slope of the displacement vs. crack length curve. Here we have assumed that the displacement vs. crack length curve is linear and we have defined the slope as  $m = \Delta u / \Delta a$ . From a standard linear elastic fracture mechanics calculation we calculate the expected slope to be  $m_0 = 1.35 \times 10^{-2}$ . The normalized error in the slope is reported in the table. For this measure of error the cubic degradation function is more accurate than the quadratic functions, especially in the irreversible case. This can, at least in part, be explained by the fact that the cubic

	$h_{\min}$	$\mathcal{G}_c^{\text{eff}} = \frac{\Delta \tilde{\Psi}_c}{\Delta a}$	$\frac{\mathcal{G}_c^{\text{eff}} - \mathcal{G}_c}{\mathcal{G}_c}$	Conv. Rate	$m = \frac{\Delta u}{\Delta a}$	$\frac{m - m_0}{m_0}$	
Quad. deg. func.	Reversible	$\ell_0/2$	0.013520	0.3520	$1.745 \times 10^{-2}$	0.293	
		$\ell_0/4$	0.011780	0.1780	$1.653 \times 10^{-2}$	0.224	
		$\ell_0/8$	0.010959	0.0959	$1.603 \times 10^{-2}$	0.187	
	4th	$\ell_0/2$	0.012450	0.2450	$1.688 \times 10^{-2}$	0.250	
		$\ell_0/4$	0.010911	0.0911	$1.603 \times 10^{-2}$	0.187	
		$\ell_0/8$	0.010285	0.0285	$1.562 \times 10^{-2}$	0.157	
Irrev.	2nd	$\ell_0/2$	0.016355	0.6355	$2.350 \times 10^{-2}$	0.741	
		$\ell_0/4$	0.014355	0.4355	$2.298 \times 10^{-2}$	0.702	
		$\ell_0/8$	0.013186	0.3186	$2.171 \times 10^{-2}$	0.608	
	Cubic deg. func.	Reversible	$\ell_0/2$	0.013304	0.3304	$1.627 \times 10^{-2}$	0.205
			$\ell_0/4$	0.011637	0.1637	$1.547 \times 10^{-2}$	0.146
			$\ell_0/8$	0.010819	0.0819	$1.497 \times 10^{-2}$	0.109
Cubic deg. func.	4th	$\ell_0/2$	0.012521	0.2521	$1.594 \times 10^{-2}$	0.181	
		$\ell_0/4$	0.011075	0.1074	$1.517 \times 10^{-2}$	0.124	
		$\ell_0/8$	0.010343	0.0343	$1.474 \times 10^{-2}$	0.092	
	Irrev.	$\ell_0/2$	0.013721	0.3721	$1.624 \times 10^{-2}$	0.203	
		$\ell_0/4$	0.012802	0.2802	$1.590 \times 10^{-2}$	0.178	
		$\ell_0/8$	0.011996	0.1996	$1.557 \times 10^{-2}$	0.153	

Table 4.5: The error in the critical energy release rate and slope of the displacement vs. crack length curve,  $m = \Delta u / \Delta a$ . The reference slope,  $m_0 = 1.35 \times 10^{-2}$ , was calculated using standard linear elastic fracture mechanics.

degradation function results in a model that is closer to the linear elastic brittle fracture model. It has a stress-strain curve that is nearly linear and a higher critical stress.

**Remark 4.4.** We note that for the reversible second-order theory the convergence rate for the error in  $\mathcal{G}_c^{\text{eff}}$  appears to be decreasing for the quadratic degradation function. A possible explanation for this is that strains exist in the body away from the crack. These strains cause a decrease in the phase-field and a corresponding increase in the surface energy term,  $\mathcal{G}_c^{\text{eff}}$ , that is not related to the crack length. This effect is much less for the cubic degradation function (see Figure 4.2) which may account for the slightly higher convergence rates.

**Remark 4.5.** From the results presented in Table 4.5 it appears that the choice of degradation function has only a minor influence on  $\mathcal{G}_c^{\text{eff}}$  for the reversible case. This is expected since the degradation function does not appear in the surface energy term,  $\tilde{\Psi}_c$ . In fact, the estimates by Bourdin et al. [26] are established completely independent of the degradation function.

A comparison of the energy balance,  $W_{\Delta u} = \Delta\tilde{\Psi}_c + \Delta\tilde{\Psi}_{SE}$  is shown in Table 4.6. For simulations under the reversible case, the error in the energy balance is within less than a tenth of a percent. This difference is likely due to numerical tolerances in the Newton-Raphson iterations used to solve the monolithic problem, and is acceptably small. For the reversible case, however, it appears that energy is being created. Care should be taken in interpreting these results. Recall that the history functional used to enforce irreversibility in these results is an ad-hoc modification of the formulation after it has been derived from the free energy functional. This makes it difficult to interpret the energy terms,  $\tilde{\Psi}_{SE}$  and  $\tilde{\Psi}_c$ , precisely. A more thorough investigation is needed to

determine if the history functional is a thermodynamically consistent mechanism for enforcing irreversibility.

	$h_{\min}$	$W_{\Delta u}$	$\frac{\Delta \tilde{\Psi}_{SE}}{W_{\Delta u}}$	$\frac{\Delta \tilde{\Psi}_c}{W_{\Delta u}}$	$\frac{\Delta \tilde{\Psi}_{SE} + \Delta \tilde{\Psi}_c}{W_{\Delta u}}$
Quad. deg. func.	Reversible	$\ell_0/2$	0.0505784	0.2337	0.7661
		$\ell_0/4$	0.0459658	0.2233	0.7765
		$\ell_0/8$	0.0435711	0.2151	0.7848
	4th	$\ell_0/2$	0.484844	0.2391	0.7606
		$\ell_0/4$	0.0442621	0.2309	0.7688
		$\ell_0/8$	0.0421374	0.2185	0.7813
	Irrev.	$\ell_0/2$	0.0497445	0.3126	0.6996
		$\ell_0/4$	0.0465710	0.3455	0.6707
		$\ell_0/8$	0.0446493	0.3406	0.6801
Cubic deg. func.	Reversible	$\ell_0/2$	0.0526557	0.2236	0.7764
		$\ell_0/4$	0.0477124	0.2114	0.7886
		$\ell_0/8$	0.0451421	0.1995	0.8005
	4th	$\ell_0/2$	0.0505462	0.2227	0.7772
		$\ell_0/4$	0.0462552	0.2105	0.7892
		$\ell_0/8$	0.0439444	0.2017	0.7981
	Irrev.	$\ell_0/2$	0.0525657	0.2219	0.8038
		$\ell_0/4$	0.0478437	0.1984	0.8413
		$\ell_0/8$	0.0456216	0.2003	0.8444

Table 4.6: A comparison of the energy balance  $W_{\Delta u} = \tilde{\Psi}_{SE} + \tilde{\Psi}_c$  for each variation of the model.

# Chapter 5

## Numerical results

In this chapter we investigate the numerical performance of the phase-field fracture model. All geometries have been discretized spatially using either NURBS or T-spline basis functions (which we refer to as the global smooth basis). The numerical computation for all the models was performed using the Bézier extraction methods described in Section 2.3. To integrate arrays over the Bézier elements we use Gaussian quadrature with a  $p + 1$  rule in each parametric direction, where  $p$  is the polynomial degree of the basis functions. Thus, in two-dimensions we use a 3-by-3 quadrature rule for quadratic basis functions and a 4-by-4 quadrature rule for cubic basis functions. Hughes, Reali, and Sangalli [55] have shown that smooth bases allow for more efficient quadrature rules. This is beyond the scope of the work presented here and we acknowledge that the quadrature rules we use may represent overkill.

For the examples below, the reported mesh sizes,  $h$ , are computed on the Bézier elements as  $h = \sqrt[d]{a}$  where  $a$  is the area of an element in two dimensions and the volume of an element in three dimensions and  $d$  is the number of spatial dimensions. In most cases, the mesh is such that  $h = \ell_0/2$  in the area where a crack has formed. Experience has shown that this relationship between  $h$  and  $\ell_0$  provides sufficient accuracy without over resolving the crack. Unless otherwise noted, irreversibility was enforced using the strain energy history functional discussed in Section 3.6, and the quadratic degradation function (4.3) was used.

## 5.1 Quasi-static results

In this section we consider two quasi-static benchmark tests reported by Miehe, Hofacker, and Welschinger [68] in order to compare the results obtained by classical  $C^0$  finite elements and isogeometric finite elements. For both examples, the material parameters are  $E = 210$  GPa,  $\nu = 0.3$ , and  $\mathcal{G}_c = 2700$  J/m<sup>2</sup> and plane strain is assumed. The length scale was chosen to be  $\ell_0 = 7.5 \times 10^{-6}$  m and we did not include a viscous damping term on the phase-field.

We use  $C^2$ -continuous cubic T-splines for the spatial discretization of the models and the staggered quasi-static solution strategy described by Miehe, Hofacker, and Welschinger [68] to obtain the solution at each load increment. As can be seen from Figures 5.1b and 5.4b, T-splines can be locally refined in the area where the crack forms. This is in contrast to a NURBS, where refinement propagates globally to produce a much denser mesh. The T-splines were refined *a priori* based on the expected solution. The initial crack is modeled as a discrete discontinuity in the geometry and the T-spline contains  $C^0$  lines that radiate out from the crack tip so that the mesh is divided into four equal square subdomains of  $C^2$  continuity. These  $C^0$  lines were included to facilitate modeling the sharp crack tip as a  $C^0$  geometric feature. An alternative would have been to use a globally  $C^2$ -continuous T-spline on the entire domain and to introduce the crack through an induced crack in the phase-field (see Appendix B.1).

### 5.1.1 2D Quasi-static tension load

The geometry and boundary conditions of the this model are shown in Figure 5.1. The locally refined T-spline contained 1370 cubic basis functions. The calculations of Miehe, Hofacker, and Welschinger [68] utilized 20,000 linear triangles.

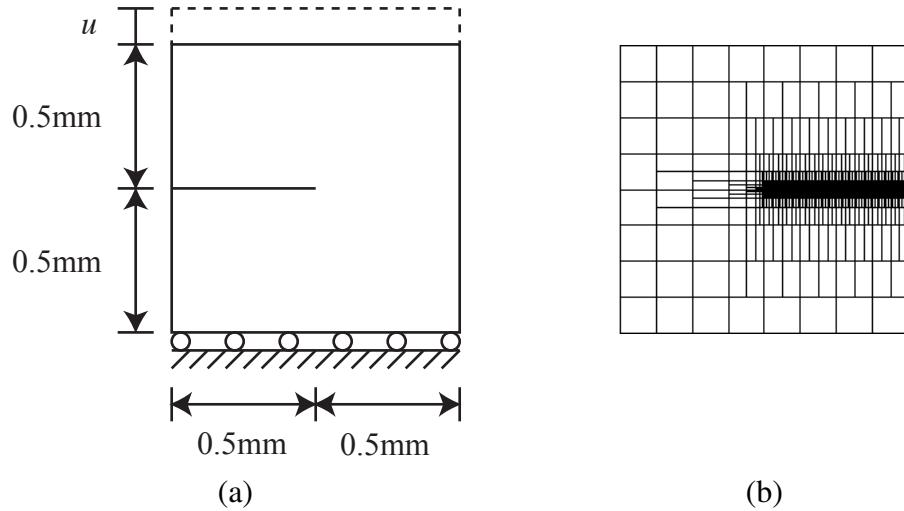


Figure 5.1: Boundary and load conditions and T-spline Bézier mesh for the single notch in tension benchmark test. The T-spline contains 1370 cubic basis functions and the effective element size is between  $h_{min} = 3.906 \times 10^{-3}$  mm and  $h_{max} = 1.25 \times 10^{-1}$  mm.

Figure 5.2 shows the progression of the crack at several load levels and the load-displacement curve is shown in Figure 5.3 with a comparison to the results reported by Miehe, Hofacker, and Welschinger [68]. The curve labeled T-spline 1 is the result for the original T-spline and the curve labeled T-spline 2 is the result of one global refinement of T-spline 1. The results are close to those published by Miehe et al. [68].

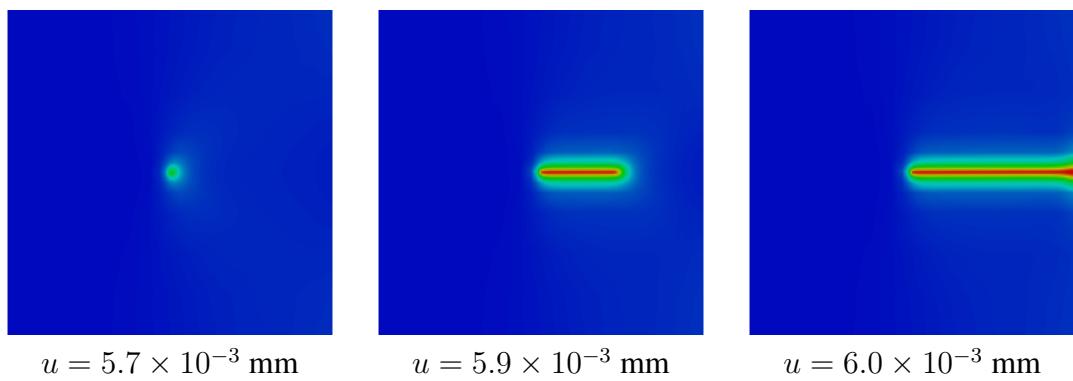


Figure 5.2: Crack progression for tension test at several load increments.

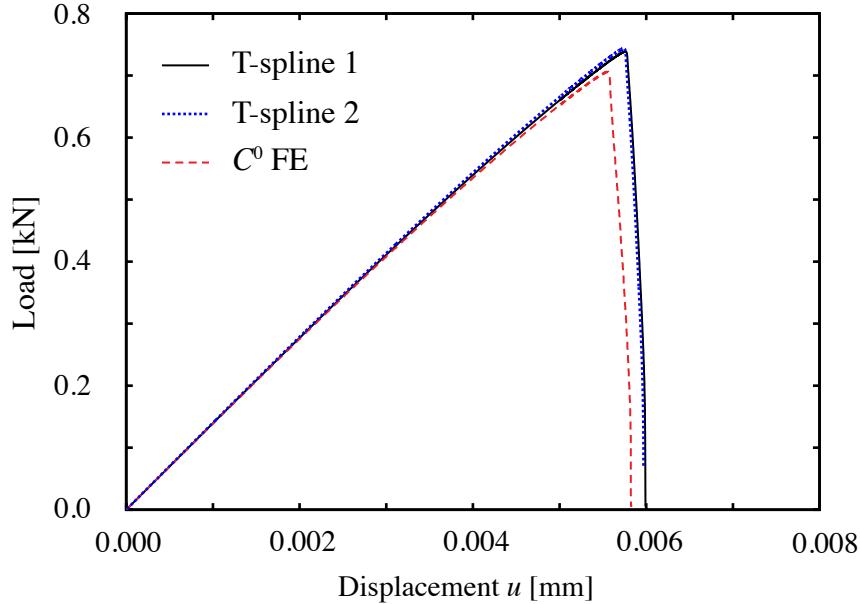


Figure 5.3: Force-displacement curve for the notched tension benchmark test. T-spline 1 is shown in Figure 5.1b. T-spline 2 is a single global refinement of T-spline 1. The results obtained using the T-spline discretizations are compared to those reported by Miehe, Hofacker, and Welschinger [68] using standard  $C^0$  finite elements.

### 5.1.2 2D Quasi-static shear load

The geometry and boundary conditions of the this model are shown in Figure 5.4. The locally refined T-spline contained 5,587 cubic basis functions. The calculations of Miehe, Hofacker, and Welschinger [68] utilized 30,000 linear triangles.

Figure 5.5 shows the progression of the crack at several load levels and the load-displacement curve is shown in Figure 5.6 with a comparison to the results reported by Miehe, Hofacker, and Welschinger [68]. As can be seen from the load-displacement curve, the results obtained from the T-spline mesh are in good agreement with those obtained from standard  $C^0$  finite elements, but with far fewer degrees-of-freedom. We attribute this to the smooth description of the stress fields, which was also observed to be beneficial in cohesive zone modeling by Verhoosel et al. [99] and gradient damage

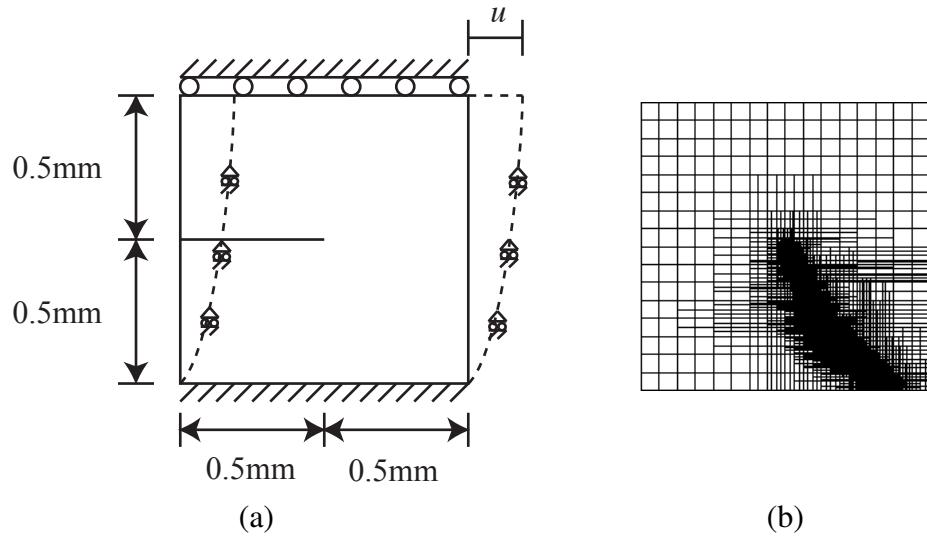


Figure 5.4: Input model for the quasi-static shear benchmark test: (a) geometry and boundary conditions and (b) the Bézier element representation of the T-spline. The T-spline contains 5587 cubic basis functions and the effective element size is between  $h_{min} = 3.906 \times 10^{-3}$  mm and  $h_{max} = 6.25 \times 10^{-2}$  mm. The refinement was performed *a priori* based on knowledge of the expected crack path.

modeling by Verhoosel et al. [100].

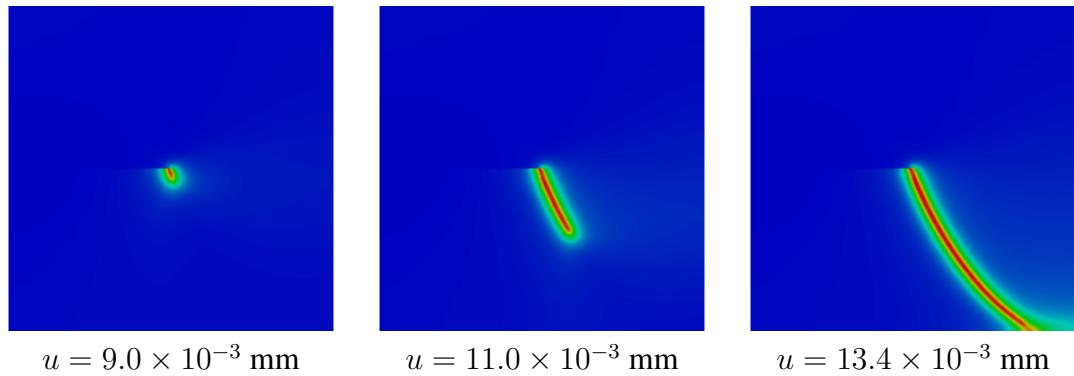


Figure 5.5: Crack progression for the quasi-static shear test at several load increments.

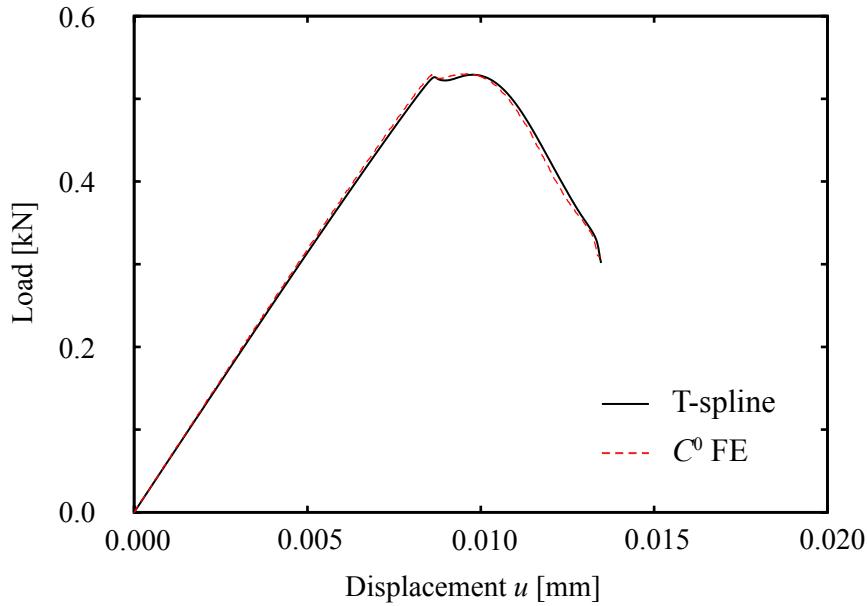


Figure 5.6: Force-displacement curve for the quasi-static shear test. The results obtained using the T-spline discretization are compared to those reported by Miehe, Hofacker, and Welschinger [68] using standard  $C^0$  finite elements.

## 5.2 Two-Dimensional Dynamic Crack Branching

In this example, we model a pre-notched rectangular plate loaded dynamically in tension. The geometry and boundary conditions of the problem are shown in Figure 5.7. A traction load is applied to the top and bottom surface at the initial time step and held constant throughout the simulation. All other surfaces have a zero traction condition applied. This load condition is such that crack branching will occur (see Song, Wang, and Belytschko [96] for a report of results for this problem using several other methods of dynamic fracture analysis). The initial crack is induced by an initial strain-history field (see Appendix B.1) allowing the geometry to be modeled as a single quadratic  $C^1$ -continuous NURBS patch.

The model parameters are  $\rho = 2450 \text{ kg/m}^3$ ,  $E = 32 \text{ GPa}$ ,  $\nu = 0.2$ ,  $\mathcal{G}_c = 3 \text{ J/m}^2$ ,

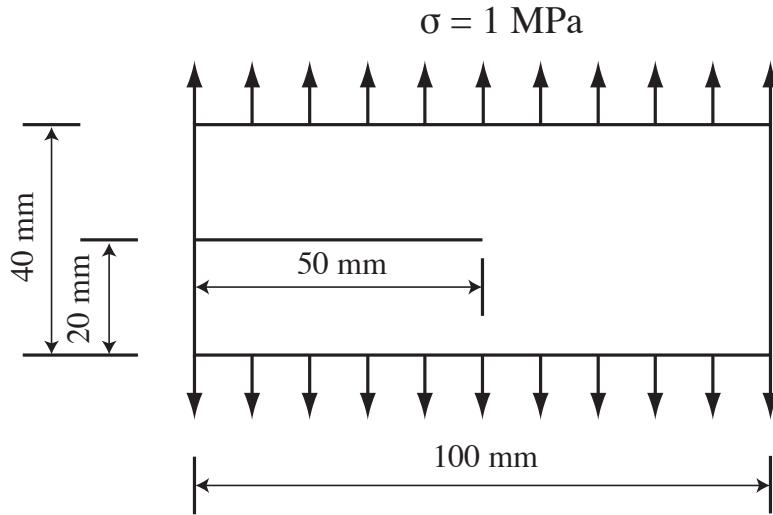


Figure 5.7: The geometry and boundary conditions for the crack branching example. In the simulation the initial crack is modeled by introducing an initial strain history field that induces a phase field at the initial crack location, and the geometry and displacement field are  $C^1$  continuous throughout (see Appendix B.1).

and plane strain is assumed. The corresponding dilatational, shear, and Rayleigh wave speeds are  $v_d = 3810$  m/s,  $v_s = 2333$  m/s,  $v_R = 2125$  m/s. The length scale was chosen to be  $\ell_0 = 2.5 \times 10^{-4}$  m. The monolithic generalized- $\alpha$  time integration scheme discussed in section 3.7.1 was used with  $\rho_\infty = 0.5$ .

Table 5.1 lists the parameters for three successively finer meshes used for this example. A uniform mesh was used to remove any effect from mesh distribution and size variation. To ensure accurate results, the time step was chosen for each mesh such that  $\Delta t \approx h/v_R$ .

In order to make a comparison between the results from the three meshes we define the elastic strain energy as

$$\tilde{\Psi}_{SE} = \int_{\Omega} \{ [(1 - k)c^2 + k]\psi_e^+(\boldsymbol{\varepsilon}) + \psi_e^-(\boldsymbol{\varepsilon}) \} \, d\boldsymbol{x} \quad (5.1)$$

	$h$	$\Delta t$
Mesh 1	$2.5 \times 10^{-4}$ m	$1 \times 10^{-7}$ s
Mesh 2	$1.25 \times 10^{-4}$ m	$5 \times 10^{-8}$ s
Mesh 3	$6.25 \times 10^{-5}$ m	$2.5 \times 10^{-8}$ s

Table 5.1: The mesh sizes and time steps used for the dynamic crack branching example. The mesh is a uniform, quadratic NURBS in each case. To ensure accurate results, the time step size is chosen to be roughly  $h/v_R$ .

and the dissipated energy as

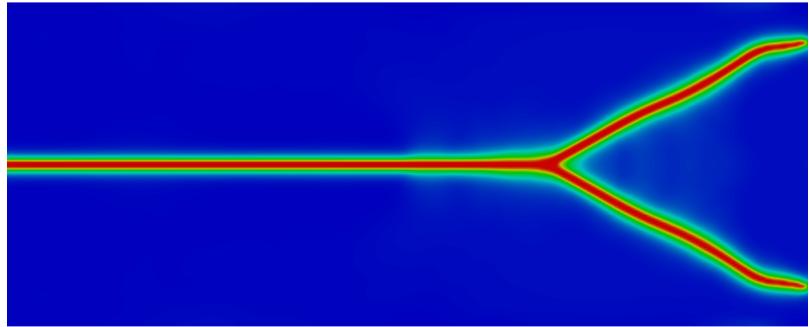
$$\tilde{\Psi}_c = \int_{\Omega} \mathcal{G}_c \left[ \frac{(c - 1)^2}{4\ell_0} + \ell_0 \frac{\partial c}{\partial x_i} \frac{\partial c}{\partial x_i} \right] dx. \quad (5.2)$$

These values provide a measure of the global response of the model.

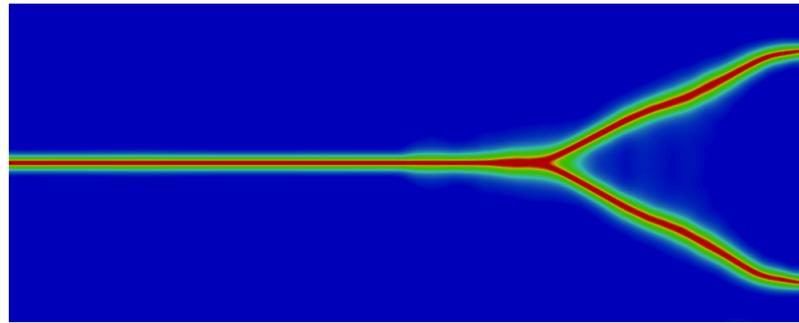
The final phase-field results for each mesh are shown in Figure 5.8. As can be seen, the crack path is similar for all meshes although mesh 1 ( $h = \ell_0$ ) fails to reach the boundary in the same amount of time as the other two. This is typical of the behavior seen when the mesh is too coarse to adequately resolve the phase-field transition near the crack. Coarse meshes tend to limit the crack tip velocity. It is difficult to see a difference in the crack path between mesh 2 and mesh 3.

Figure 5.9(a) shows the elastic strain energy as defined by (5.1) and Figure 5.9(b) show the energy dissipated by the formation of the crack as defined by (5.2). These plots show that  $h = \ell_0/2$  provides enough resolution to capture the response of the model. For  $h = \ell_0$  the mesh is too coarse to capture the gradients of the phase-field near the crack. This causes the material to behave as if it is too stiff and too much energy is dissipated. The crack tip velocity is plotted in Figure 5.10. In all cases, the velocity stays well below 60% of the Rayleigh wave speed as has been commonly observed in experiments (see Ravi-Chandar and Knauss [82]).

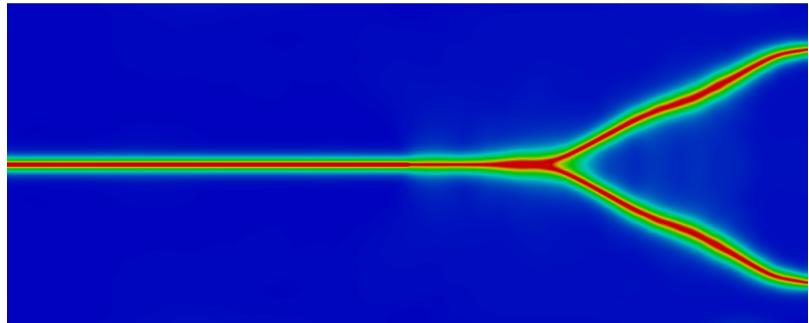
In Figure 5.11 we show a post-processed plot of mesh 2 at  $t = 70 \mu s$ . In this



(a) Mesh 1 with  $h = 2.5 \times 10^{-4}$  m



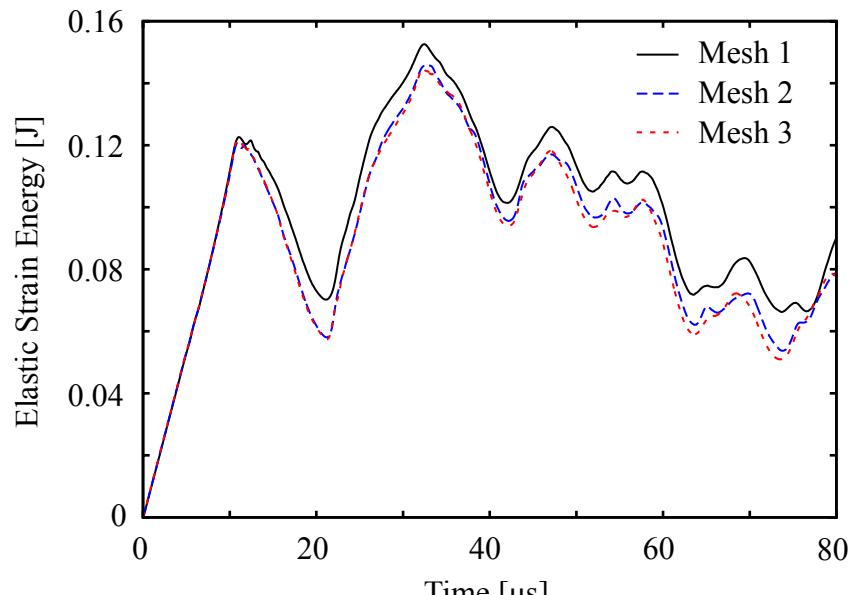
(b) Mesh 2 with  $h = 1.25 \times 10^{-4}$  m



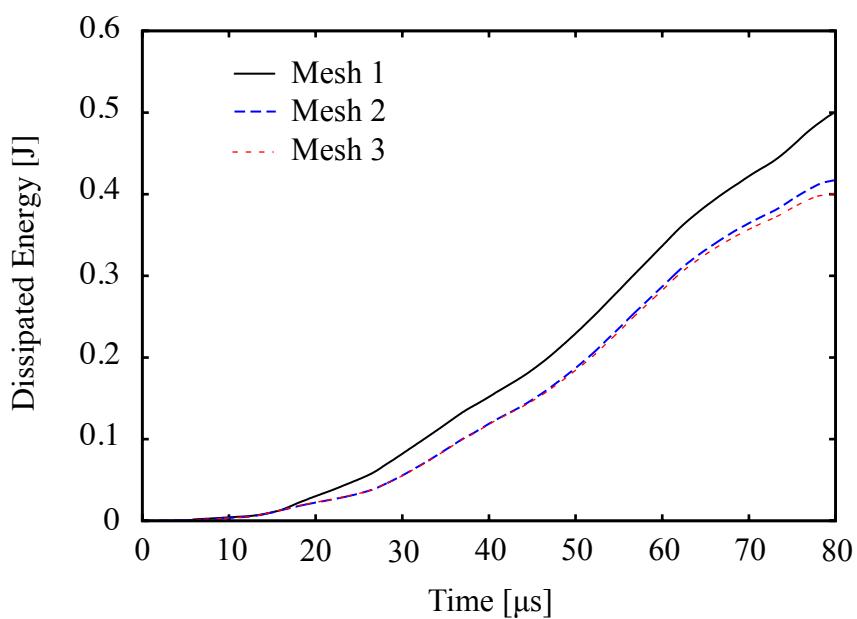
(c) Mesh 3 with  $h = 6.25 \times 10^{-5}$  m

Figure 5.8: Results for the crack branching example at  $t = 80 \mu\text{s}$ . For all three meshes  $\ell_0 = 2.5 \times 10^{-4}$  m.

figure we have scaled the displacements by a factor of 50 and removed areas of the model from the plot where  $c < 0.05$  in order to show a representation of the cracked geometry.



(a)



(b)

Figure 5.9: Plots over time of (a) the elastic strain energy as defined by (5.1) and (b) the dissipated fracture energy as defined by (5.2).

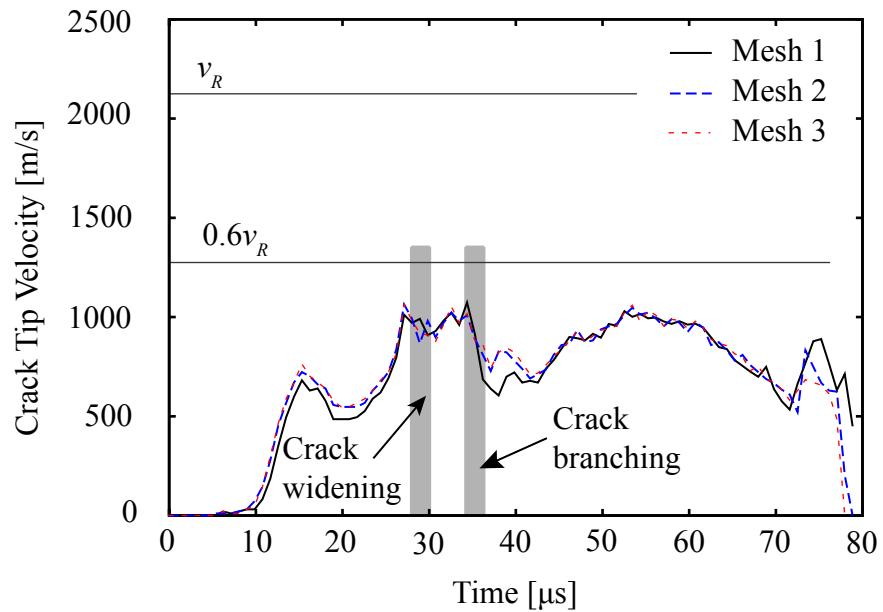


Figure 5.10: Plots over time of the crack tip velocity. After branching, the reported crack tip velocity is that of the upper branch.

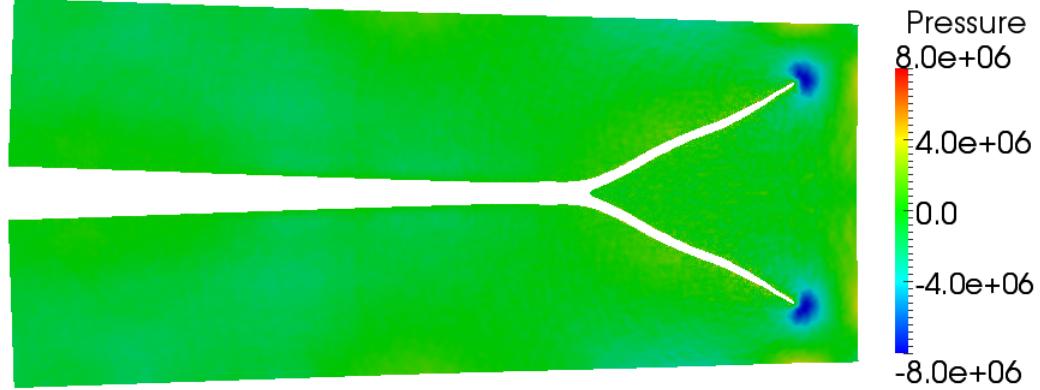


Figure 5.11: A post-processed plot of mesh 2 at  $t = 70 \mu\text{s}$ . The displacements have been scaled by a factor of 50 and areas of model where  $c < 0.05$  have been removed from the plot. Pressure is measured in Pascals.

**Remark 5.1.** For the monolithic time integration scheme the size of  $\Delta t$  is limited by accuracy and convergence. For the dynamic crack branching example discussed here,

choosing a time step such that  $\Delta t \leq 2h/v_R$  produced acceptable accuracy. For  $\Delta t = 8h/v_R$  the Newton-Raphson method failed to converge near the time of the initial crack propagation.

**Remark 5.2.** Since the crack is not tracked algorithmically, the velocity of the crack tip is measured as a post-processing step. For the values reported here, the location of the crack tip,  $x$ , is found on an iso-curve of the phase-field with value 0.25. The velocity is then computed as  $v_n = (x_{n+1} - x_n)/\Delta t$ .

### 5.2.1 Reversible vs. Irreversible

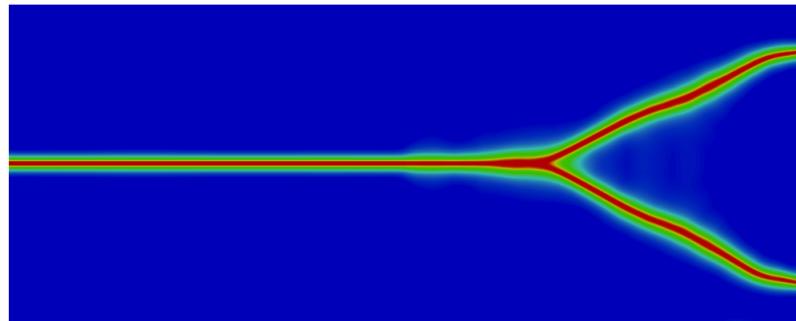
Since the displacements for the branching crack problem behave in a generally monotonic behavior, up to reflected waves, we can assume that it is not necessary to enforce irreversibility. With this in mind, we repeat the simulations for mesh 2 without enforcing irreversibility. The final phase-field results are shown in Figure 5.12. It is clear from these results that enforcing irreversibility through the strain energy history functional,  $\mathcal{H}$ , has an effect on the phase-field. For the reversible case the decrease in the phase-field parameter is much more local to the crack. Qualitatively, however, both results are similar.

Figures 5.13(a) and 5.13(b) show a comparison of the elastic strain and dissipated energies for the reversible and irreversible cases. We can clearly see that the irreversible case leads to more dissipated energy as a result of the wider transition zone near the crack. This also causes a decrease in the elastic strain energy since there is less undamaged material. Interestingly, the crack tip velocity, reported in Figure 5.14, is similar for both cases.

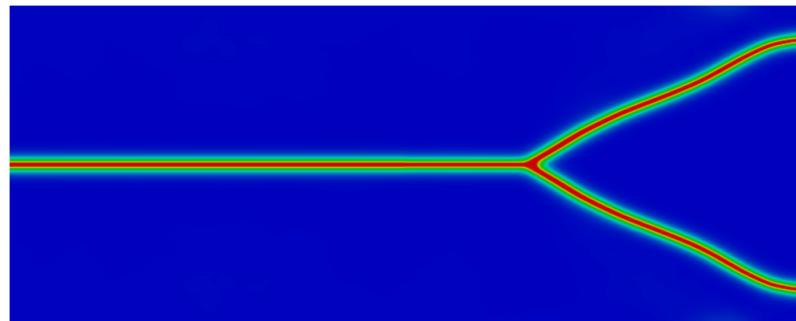
### 5.2.2 Multiple branches and fragmentation

In dynamic fracture the branching behavior of cracks is related to the applied loads. Here we study the effect of increasing the applied traction,  $\sigma$  (see Figure 5.7). In addition, we increase the total simulation time to see the effects of stress waves as they reflect through the body after the cracks have formed.

We performed the simulations on mesh 2 using the staggered solution strategy discussed in Section 3.7.2. The explicit HHT- $\alpha$  method was used for the time integration of the momentum equation with  $\alpha = -0.3$ . An adaptive time step of  $\Delta t = 0.9\Delta t_{\text{crit}}$

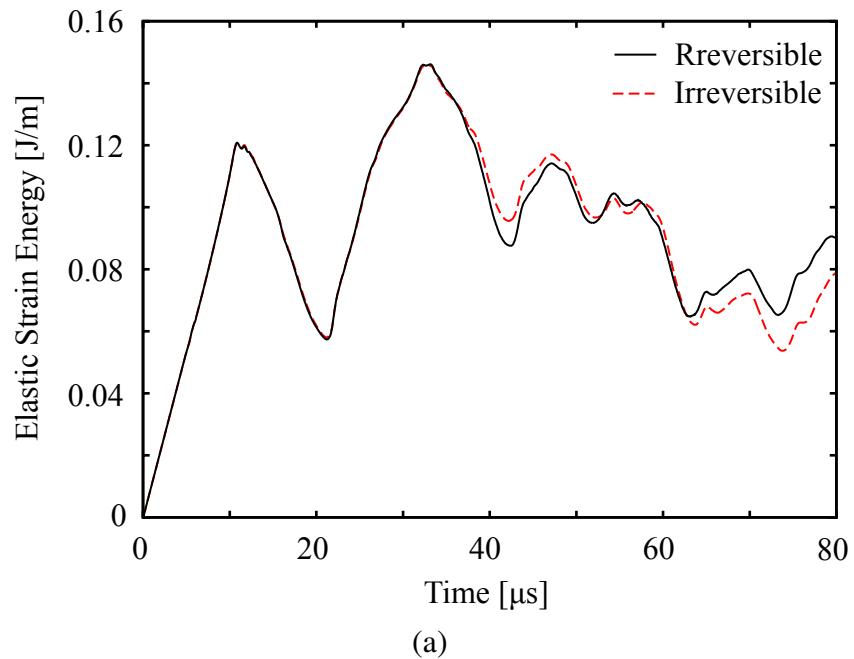


(a) Irreversible

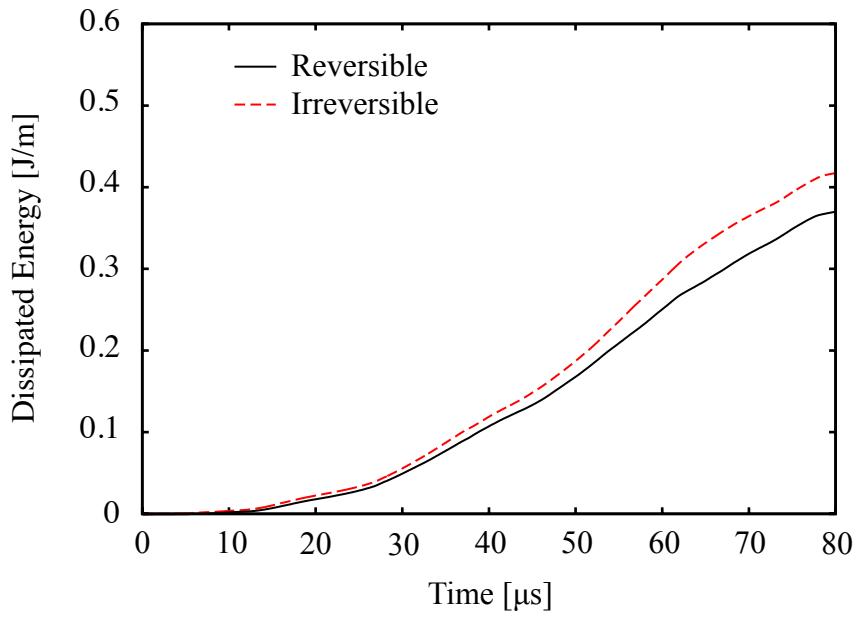


(b) Reversible

Figure 5.12: A comparison between the model with irreversibility enforced through the strain energy history functional and reversible behavior allowed for mesh 2. In the reversible case the decrease in the phase-field parameter is much more local to the crack. Qualitatively, however, both results are similar.



(a)



(b)

Figure 5.13: Plots over time of (a) the elastic strain energy and (b) the dissipated fracture energy.

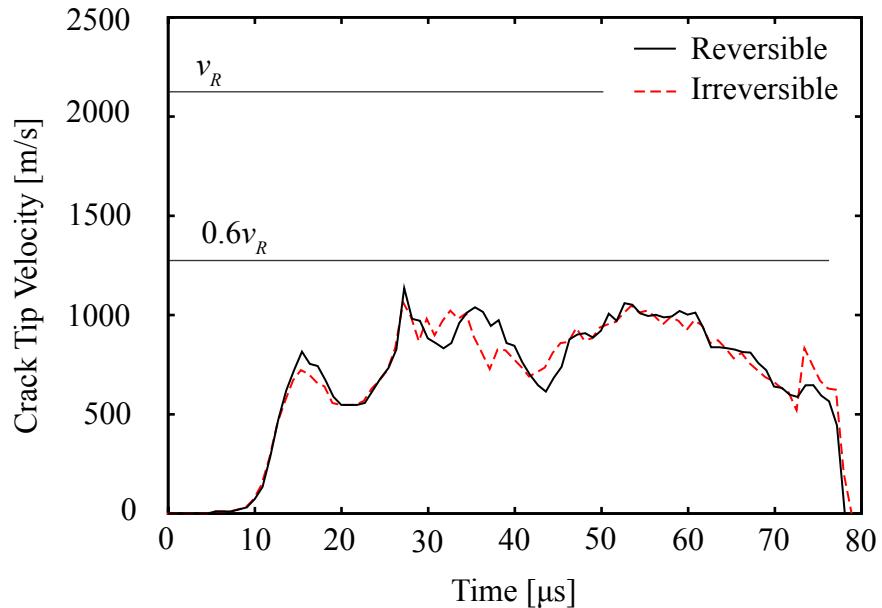


Figure 5.14: Plots over time of the crack tip velocity for mesh 2 with and without irreversibility enforced.

was used (see Section 5.3 for a definition of  $\Delta t_{\text{crit}}$ ). To compute  $\Delta t_{\text{crit}}$ , we used the power iteration algorithm presented by Benson [19].

Two cases are shown in Figures 5.15 and 5.16. For the results shown in Figure 5.15 we let  $\sigma = 1.95$  MPa. In this case, the crack is initially similar to the case for  $\sigma = 1.0$  MPa except that the branching angle is smaller. At a point in time after the crack has reached the right edge of the domain, however, tensile stress accumulates along the crack branches. This causes additional crack branching. New branches continue to form as tensile stresses accumulate in other areas of the domain.

For the results shown in Figure 5.16 we let  $\sigma = 2.3$  MPa. In this case, additional branching occurs before the crack reaches the right edge of the domain. Similar to the previous case, stress accumulation causes additional branching as time progresses, but the crack pattern is much more complex.

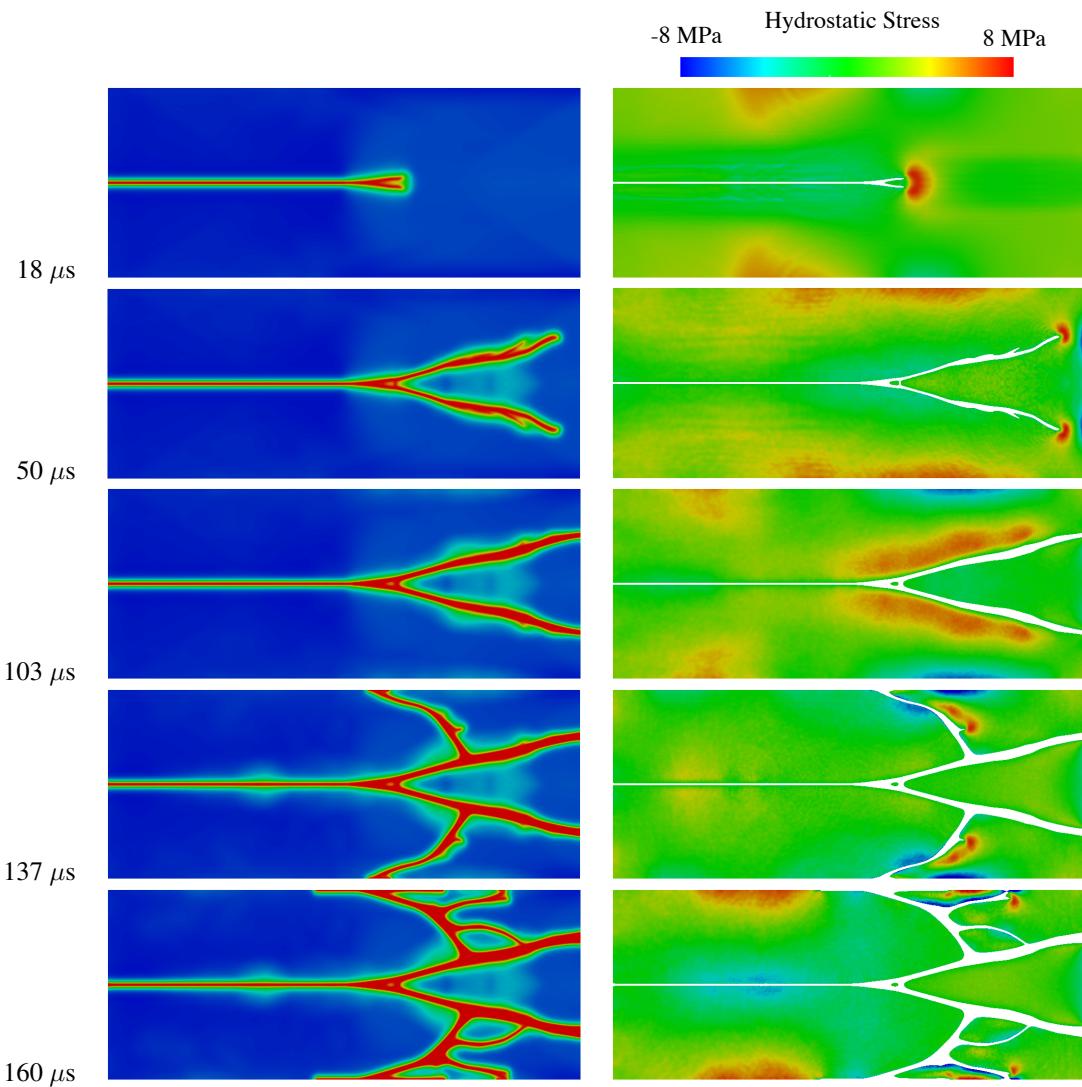


Figure 5.15: The branching crack problem with  $\sigma = 1.95$  MPa.

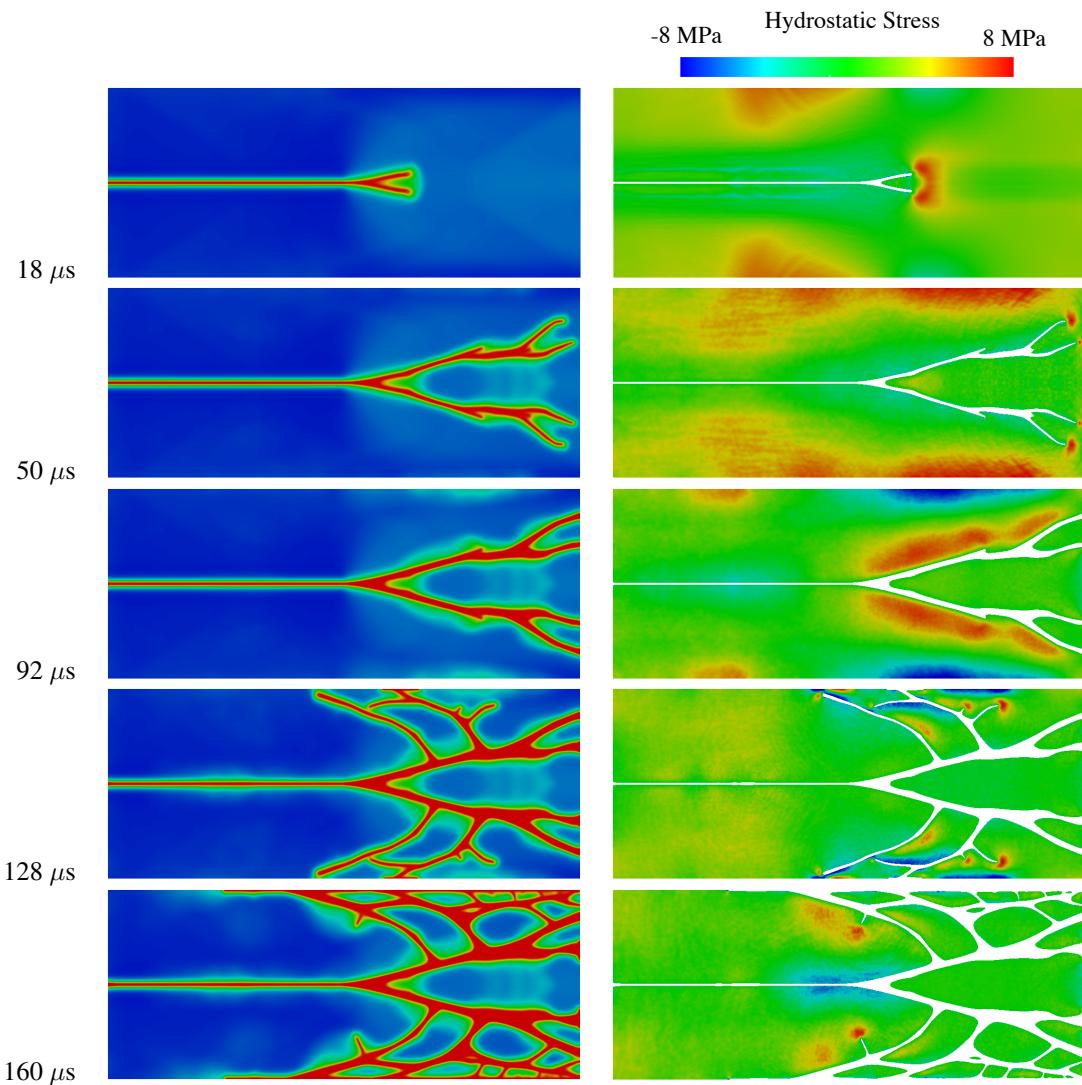


Figure 5.16: The branching crack problem with  $\sigma = 2.3$  MPa.

### 5.3 Dynamic shear loading

In this example we model crack initiation and propagation under a dynamic shear load. The model is based on experimental results reported by Kalthoff and Winkler [58] and Kalthoff [57]. Previous numerical results of this problem based on XFEM have been reported by Belytschko et al. [17], among others, and a comparison of results between XFEM, the element deletion method, and the interelement crack method have been reported by Song, Wang, and Belytschko [96]. Numerical results of a similar experiment reported by Ravi-Chandar [80] have been reported by Remmers, de Borst, and Needleman [85] where cohesive segments have been used to model the crack.

The input geometry and loading conditions for the simulation are shown in Figure 5.17, where symmetry has been employed to reduce the computational cost. In the experiment, the load was applied by firing a projectile at a prenotched specimen. In our simulation we modeled the case where the projectile was fired with a velocity of 33 m/s by applying the kinematic velocity

$$v = \begin{cases} \frac{t}{t_0} v_0 & t \leq t_0 \\ v_0 & t > t_0 \end{cases} \quad (5.3)$$

where  $v_0 = 16.5$  m/s and  $t_0 = 1\mu\text{s}$ . A no traction boundary condition was applied to all unspecified surfaces. We modeled the geometry using quadratic NURBS basis functions. The initial crack was modeled by a discontinuity in the geometry in order to introduce a sharp crack tip as in Section 5.1.2.

The model parameters are  $\rho = 8000 \text{ kg/m}^3$ ,  $E = 190 \text{ GPa}$ ,  $\nu = 0.3$ ,  $\mathcal{G}_c = 2.213 \times 10^4 \text{ J/m}^2$ ,  $k = 0$ , and plane strain is assumed. The corresponding dilatational, shear, and Rayleigh wave speeds are  $v_d = 5654 \text{ m/s}$ ,  $v_s = 3022 \text{ m/s}$ ,  $v_R = 2803 \text{ m/s}$ . The length scale was chosen to be  $\ell_0 = 1.95 \times 10^{-4} \text{ m}$  leading to a maximum uniaxial

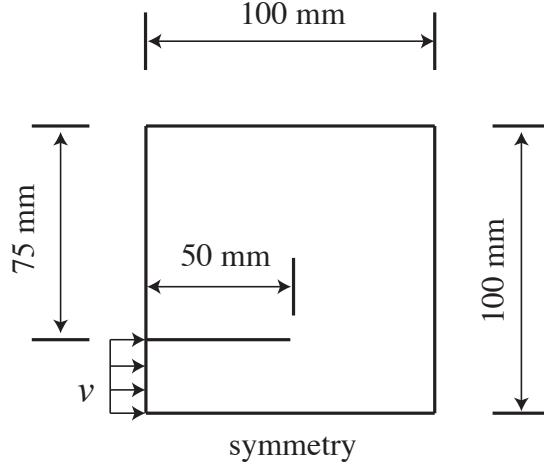


Figure 5.17: The geometry and boundary conditions for the dynamic shear loading example. The crack is modeled by an actual discontinuity in the mesh with a zero radius crack tip. The load is applied as a velocity condition that is ramped up from 0 to 16.5 m/s in one microsecond and then held constant for the duration of the simulation.

stress of 1.07 GPa (see Section 4.1). The mesh has  $1024 \times 1024$  uniform quadratic elements so that  $h \approx \ell_0/2$ .

The simulations were performed using the staggered integration scheme described in Section 3.7.2 with the momentum equation being solved explicitly using the HHT- $\alpha$  method with  $\alpha = -0.1$ . A fixed time step of  $\Delta t = 1.25 \times 10^{-8}$  was chosen, which is slightly less than  $0.9\Delta t_{\text{crit}}$ , where the critical time step,  $\Delta t_{\text{crit}}$ , is computed as

$$\Delta t_{\text{crit}} = \frac{\Omega_{\text{crit}}}{\omega_{\max}} \quad (5.4)$$

with  $\omega_{\max}$  the maximum natural frequency of the momentum equation determined from the undamped eigenproblem and (considering only the undamped case)

**HHT- $\alpha$**  (see Miranda, Ferencz, and Hughes [70])

$$\Omega_{\text{crit}} = \frac{\sqrt{2(\gamma + 2\alpha(\gamma - \beta)}}}{\gamma + 2\alpha(\gamma - \beta)} \quad (5.5)$$

**Explicit generalized- $\alpha$**  (see Hulbert and Chung [56])

$$\Omega_{\text{crit}} = \sqrt{\frac{12(1 + \rho_b)^3(2 - \rho_b)}{10 + 15\rho_b - \rho_b^2 + \rho_b^3 - \rho_b^4}}. \quad (5.6)$$

The resulting phase-field is shown in Figure 5.18. Note that, initially, the crack starts to propagate at a larger angle than the angle decreases as the crack propagates. The average angle from the initial crack tip to the point where the crack intersects the boundary is somewhat greater than  $65^\circ$ . This is in fairly good agreement with the experimental results, which show the crack propagating at about  $70^\circ$ . The velocity of the crack tip is shown in Figure 5.20. As can be seen, the crack quickly accelerates to a velocity just below 60% of the Rayleigh wave speed and maintains this velocity until it decreases as the crack approaches the top surface. Although no crack tip velocity information is reported for the experimental results, this velocity behavior is in good agreement with behavior reported by Ravi-Chandar [80] for a similar experiment.

In Figure 5.19 we show a post-processed plot of the model at  $t = 70 \mu\text{s}$ . In this figure we have scaled the displacements by a factor of 5 and removed areas of the model from the plot where  $c < 0.05$  in order to show a representation of the cracked geometry.

**Remark 5.3.** Experience has shown that the action of the phase-field does not effect the critical time step of the explicit algorithm. Choosing  $\Delta t \leq 0.9\Delta t_{\text{crit}}$  has proven sufficient to maintain stability and is often conservative, i.e, letting  $\Delta t = \Delta t_{\text{crit}}$  often results in a stable time step.

### 5.3.1 Effect of $\sigma_c$ on the material response

In Section 4.1 we showed that the critical stress,  $\sigma_c$ , depends on  $\ell_0$  and the stress degradation function  $g(c)$ . Figure 5.21 shows how this can effect crack topology and

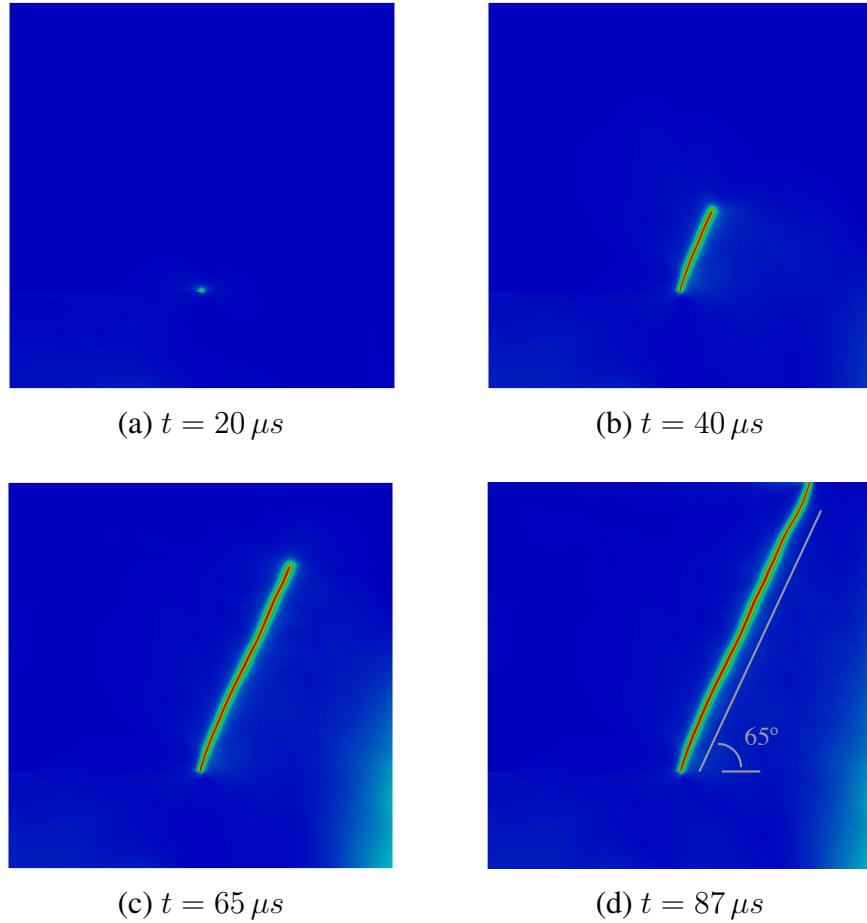


Figure 5.18: Evolution of the crack through time for a uniform  $1024 \times 1024$  quadratic NURBS mesh with 1,055,242 control points and  $\ell_0 = 1.95 \times 10^{-4}$  m. The resulting crack propagation angle is somewhat greater than  $65^\circ$  and close to the experimentally observed angle of about  $70^\circ$  reported by Kalthoff and Winkler [58] and Kalthoff [57].

propagation. This figure shows results of the dynamic shear example with different values of  $\ell_0$  and uniformly refined meshes such that  $h = \ell_0/2$  for each mesh. When  $\ell_0$  is large, a secondary crack nucleates opposite the impact site. Similar numerical results have been reported by Belytschko et al. [17]. This behavior is consistent with experimental results reported by Ravi-Chandar et al. [83] in which a mode I crack nucleates opposite the initial crack in a weakened specimen. In the experiment, this nucleation

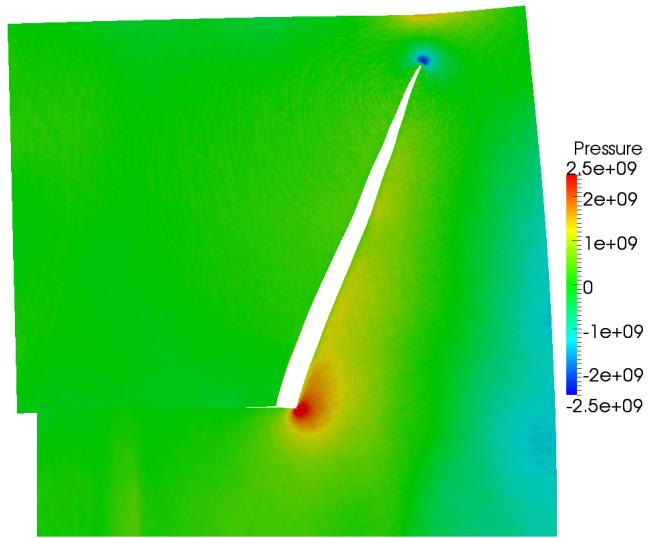


Figure 5.19: A post-processed plot of the dynamic shear loading example at  $t = 75 \mu\text{s}$ . The displacements have been scaled by a factor of 5 and areas of model where  $c < 0.05$  have been removed from the plot. Pressure is measured in Pascals.

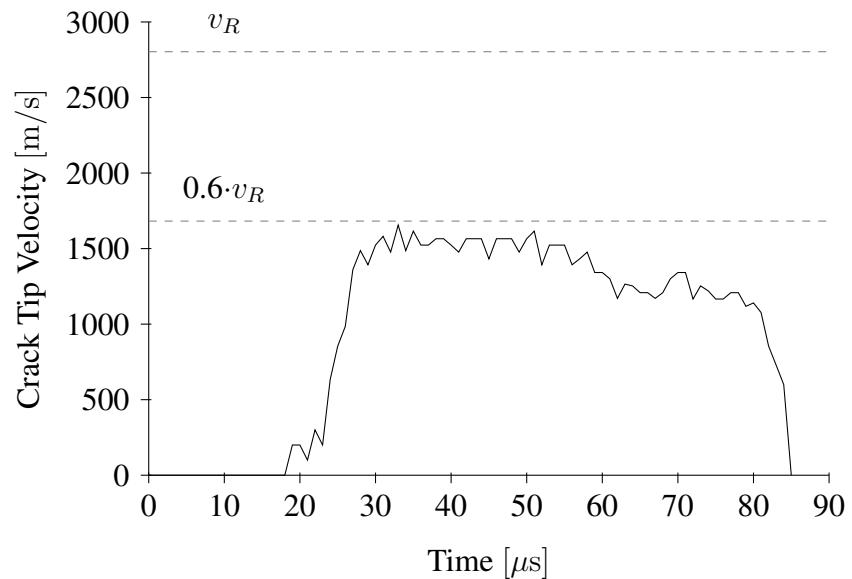


Figure 5.20: Crack tip velocity for dynamic shear loading example.

is caused by the arrival of a compressive stress wave, which reflects back from the free surface as a tensile wave.

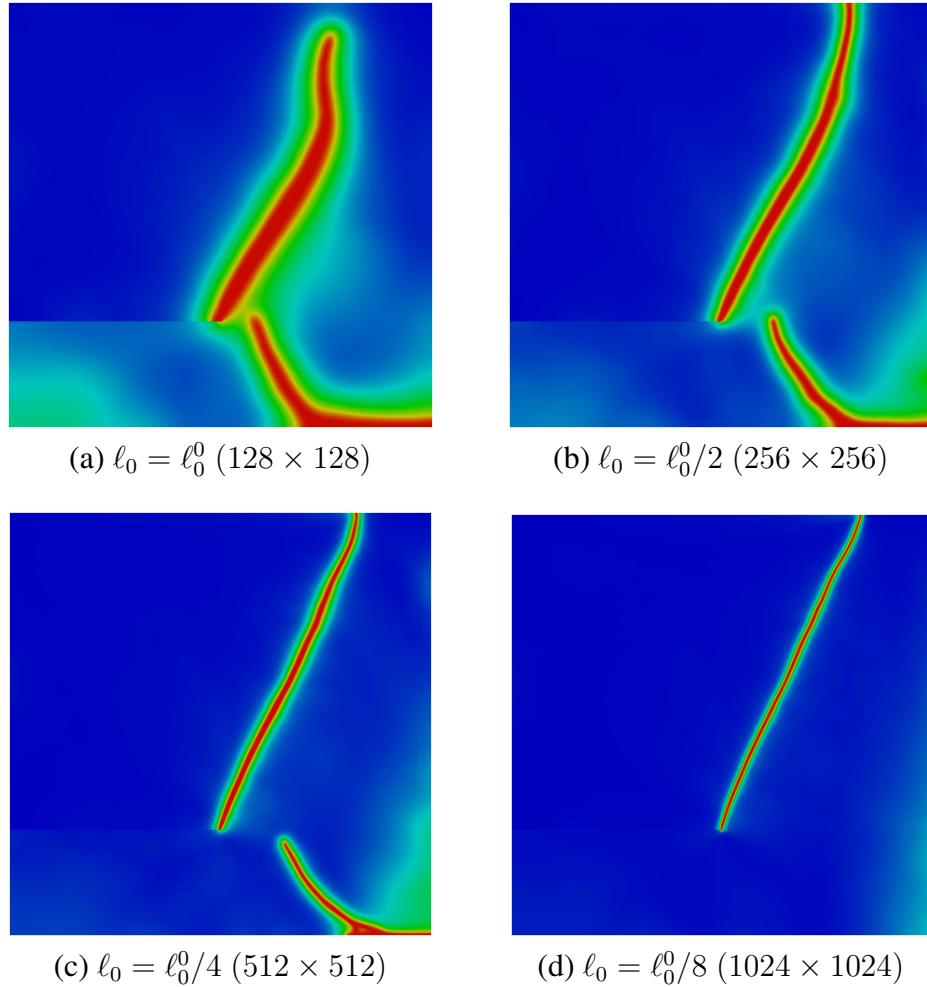
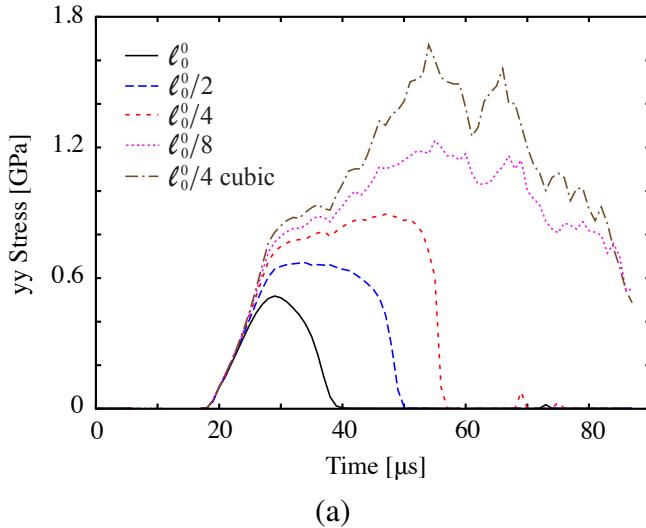


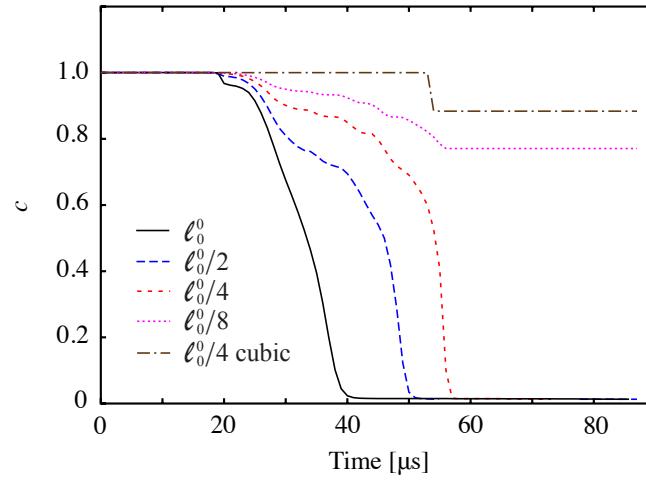
Figure 5.21: Kalthoff results at  $t = 87 \mu s$ . Initiation of spurious crack at approximately (a)  $t = 37 \mu s$ , (b)  $t = 48 \mu s$ , and (c)  $t = 56 \mu s$ .  $\ell_0^0 = 1.5625 \times 10^{-3} \text{ m}$ .

Figure 5.22 shows plots of the mode I opening stress and phase-field values over time for the point at the lower right corner of the geometry. It is clear from Figure 5.22(a) that for the three largest values of  $\ell_0$  the critical tensile stress is reached, which leads to crack nucleation. From Figure 5.22(b) it can be seen that the phase-field value for the

smallest  $\ell_0$  does not drop below the critical value of 0.75 indicating that the maximum tensile stress does not reach the critical value at which nucleation occurs.



(a)



(b)

Figure 5.22: Plots of the (a) mode I opening stress and (b) phase-field values over time at the lower right corner of the dynamic shear model. The cubic case indicates that the cubic degradation function was used. All other cases used the quadratic degradation function. Note that  $\ell_0^0 = 1.5625 \times 10^{-3}$  m.

We repeat this simulation with the cubic degradation function (4.8) with  $s =$

$10^{-4}$ . We let  $\ell_0 = \ell_0^0/4$  and use a uniform quadratic NURBS mesh with  $512 \times 512$  elements. Recall that the cubic degradation function has a critical stress that is about twice that of the quadratic degradation function. Thus, we expect the result to be similar to those shown in Figure 5.21(d). As shown in Figure 5.23, this is the case. In fact, the cubic degradation function leads to much less accumulated damage in the lower right corner.

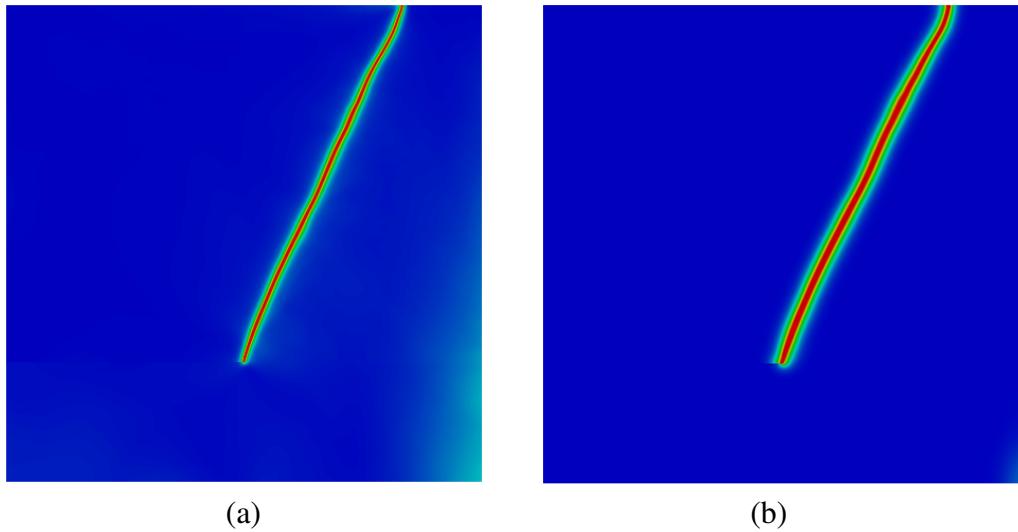


Figure 5.23: A comparison of the crack topology for (a) the quadratic degradation function with  $\ell_0 = \ell_0^0/8$  on a uniform  $1024 \times 1024$  mesh and (b) the cubic degradation function with  $\ell_0 = \ell_0^0/4$  on a uniform  $512 \times 512$  mesh.

Plots of the mode I opening stress and phase-field for the cubic degradation case are included in Figure 5.22. Notice that the peak stress for this case corresponds to the small decrease in  $c$  indicating that the material was near the threshold for crack nucleation. Also, similarly to the one-dimensional homogeneous case (Figure 4.2),  $c$  stays near the value one and then decreases sharply to a lower level.

### 5.3.2 Adaptive refinement scheme

The length scale parameter,  $\ell_0$ , plays two roles in the phase-field model: first, it determines the width of the approximation to the crack, and second, as shown in Section 4.1, it influences the magnitude of the tensile stress required for crack nucleation. Thus, in order to capture fine scale details of a crack, or model materials with high nucleation stresses, a small value for  $\ell_0$  is needed. We have shown seen this with the dynamic shear loading example. If the maximum uniaxial shear stress is too low then a secondary crack will nucleate at the surface opposite the initial crack (see Ravi-Chandar et al. [83]). Thus, a small value of  $\ell_0$  is required to accurately capture the crack topology for this problem. This in turn requires a fine mesh in areas where the crack is located.

To efficiently compute with fine meshes, as needed to accurately resolve a crack for small values of  $\ell_0$ , we introduce an adaptive refinement scheme. For this scheme we choose the phase-field parameter as a convenient measure for determining the need for refinement. Away from the crack the value of the phase-field stays close to one. By choosing a critical threshold of the phase-field that is higher than the value at which crack nucleation occurs ( $c = 0.75$ ) the area near the crack is easily identified. Using a larger value for the critical threshold results in a greater area of refinement (we have found  $c = 0.8$  to be a good choice). The adaptive refinement scheme we have developed proceeds as follows:

1. Run the dynamic simulation to some termination point
2. Flag elements where the phase-field is below the critical threshold
3. Refine the flagged elements
4. Rerun the simulation with the locally refined mesh

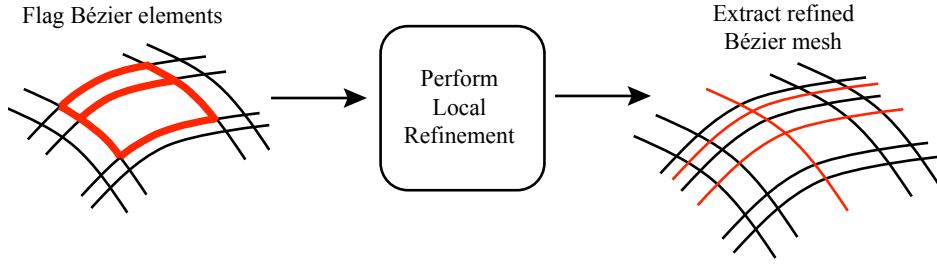


Figure 5.24: A schematic representation of an adaptive refinement scheme based on T-splines, analysis-suitable local refinement, and Bézier extraction.

5. Repeat steps 2—4 until convergence

### 5.3.3 Analysis-suitable local refinement of T-splines

A distinguishing feature of T-splines is the presence of T-junctions (hanging nodes) in the mesh. T-junctions maintain locality in the context of refinement. This is in contrast with NURBS where all refinement is global. A highly localized and efficient refinement algorithm for analysis-suitable T-splines was developed by Scott et al. [89]. This algorithm avoids introducing superfluous control points, preserves exact geometry, generates smooth nested spaces, and maintains the properties of an analysis-suitable space.

In the context of isogeometric analysis, the adaptive refinement scheme is based on Bézier extraction and analysis-suitable local refinement as shown schematically in Figure 5.24. First, the flagged Bézier elements are used to determine the basis functions of the T-spline that will be refined. Analysis-suitable local refinement is then applied to generate the refined set of T-spline basis functions. Bézier extraction is then applied to the refined T-spline to generate the new set of Bézier elements.

We apply this refinement scheme to the dynamic shear loading example. We start with a coarse initial  $C^2$ -continuous cubic T-spline that has  $128 \times 128$  Bézier elements. For all meshes,  $\ell_0$  is set to  $1.95 \times 10^{-4}$  m. Elements are flagged for refinement if the phase-field parameter is less than 0.8 at any quadrature point within the element. The sequence of the results is shown in Figure 5.25 where each simulation was terminated at  $t = 100 \mu s$ . Figure 5.26 shows the sequence of meshes with the elements that have been flagged for refinement at the end of each iteration. Note that when the mesh is too coarse, the crack propagation is restricted and the direction is incorrect. It is not until mesh 3, when  $h = \ell_0$ , that the mesh is fine enough to capture the correct crack path.

Figure 5.27 compares the elastic strain energy and dissipated energy at each refinement iteration to the solution from Section 5.3, which we call the reference solution. The elastic strain energy, shown in Figure 5.27(a), is over predicted for the coarse meshes as a result of restricted crack propagation. This plot shows that it is not until mesh 4, when most of the element along the propagation path are such that  $h = \ell_0/2$ , that the elastic strain energy agrees well with the reference solution. This is also true for the dissipated energy shown in Figure 5.27(b).

Table 5.2 lists the total number of functions for each mesh in the refinement sequence and the number of elements that were flagged at the end of each simulation. Note that the final mesh has 53,032 cubic basis functions. This is compared to 1,055,242 quadratic basis functions in the uniformly refined reference solution.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Mesh 5
Number of functions	17,755	19,992	27,032	47,824	53,032
Flagged elements	589	2,001	6,257	1,446	8

Table 5.2: The number basis functions before refinement and the number of elements that were flagged for refinement for each mesh.

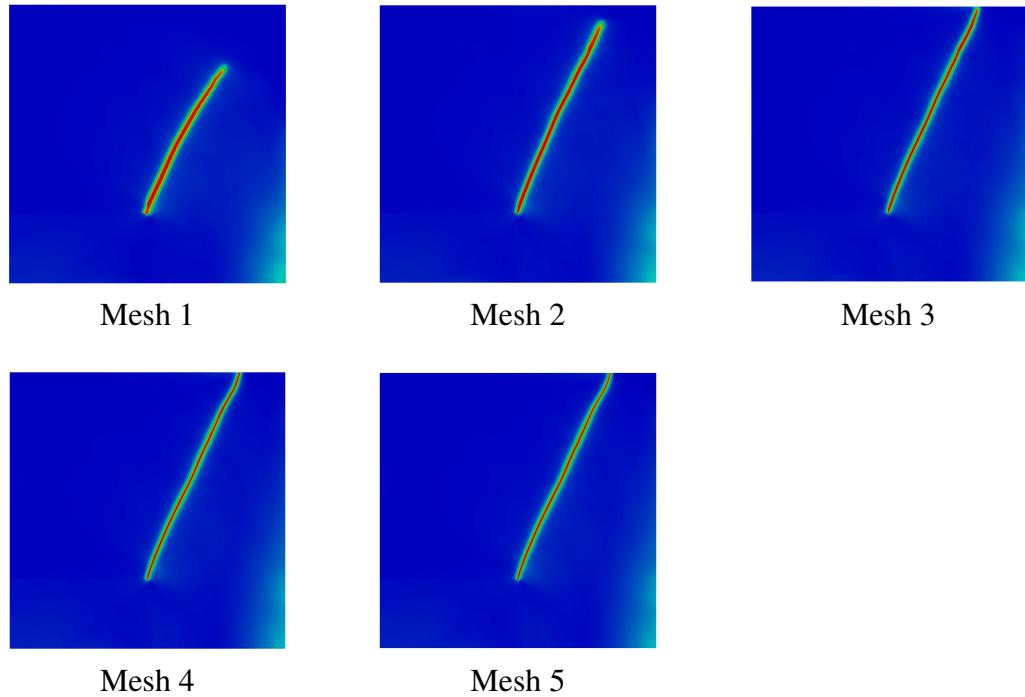
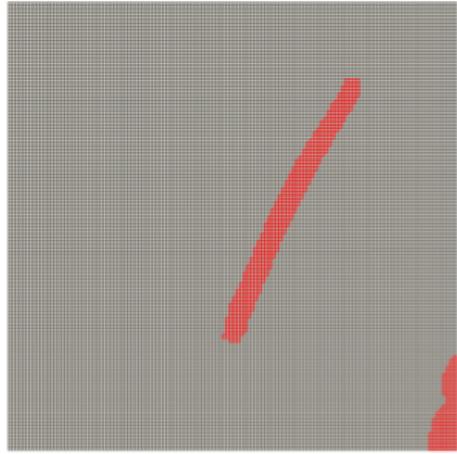
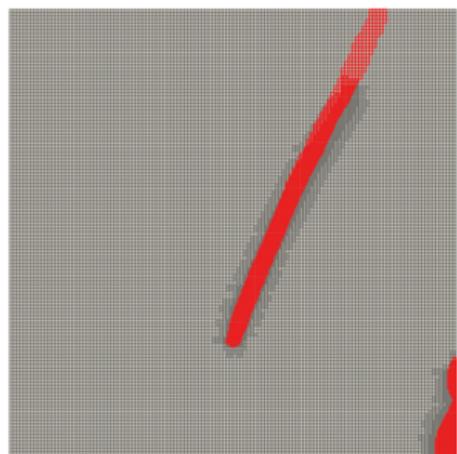


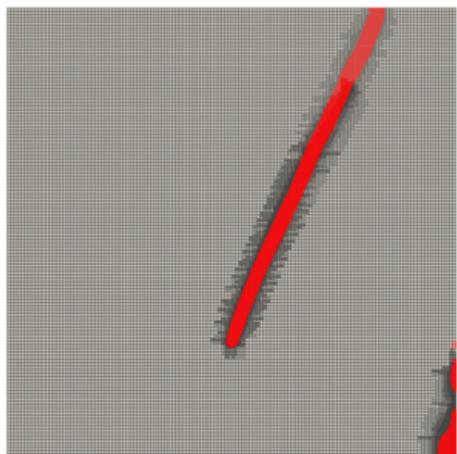
Figure 5.25: Kalthoff mesh refinement results. Mesh 1 is a  $128 \times 128$  cubic T-spline mesh. Bézier elements were flagged for refinement if  $c < 0.8$  at any quadrature point inside the element and  $h = \sqrt{a} > 1.94 \times 10^{-4}$  m where  $a$  is the element area.



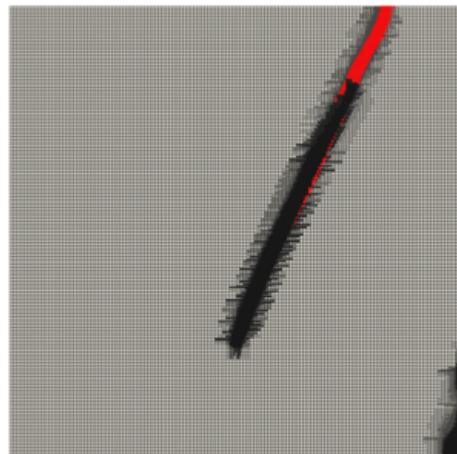
Mesh 1



Mesh 2

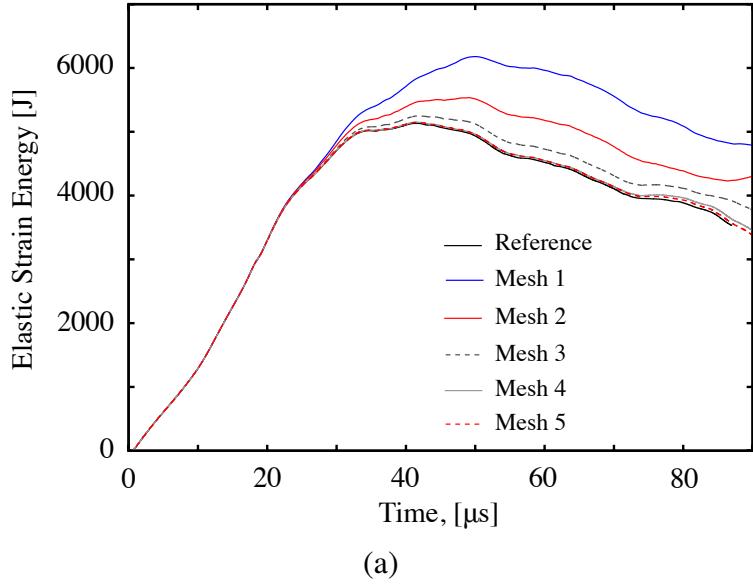


Mesh 3

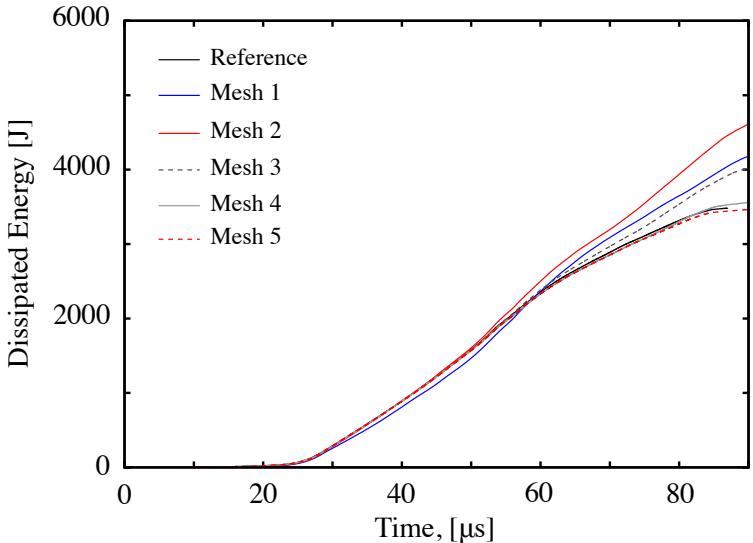


Mesh 4

Figure 5.26: The first four meshes in the local refinement sequence. The elements in red are those that were selected to be refined at each step.



(a)



(b)

Figure 5.27: The (a) elastic strain energy,  $\int_{\Omega} \{[(1 - k)c^2 + k]\psi_e^+ + \psi_e^- \} dx$ , and (b) dissipated energy,  $\int_{\Omega} \mathcal{G}_c [(c - 1)^2 / (4\ell_0) + \ell_0 |\nabla c|^2] dx$ , for the sequence of refined meshes shown in Figure 5.25. The reference mesh is the uniformly refined mesh from Figure 5.18.

## 5.4 Pressurized cracks

In this example we show that cracks defined by a phase-field can be pressurized, *i.e.*, a pressure load can be applied to the crack "surfaces" even though there is no explicit geometric surface representation. This model is based on experiments reported by Ravi Chandar and Knauss [81] and Ravi-Chandar and Knauss [82]. The input geometry is shown in Figure 5.28. The geometry is made large enough such that stress waves reflected from the boundary do not reach the propagating crack before the end of the simulation. The dashed lines in Figure 5.28 indicate the area in which the cracks will propagate during the simulation.

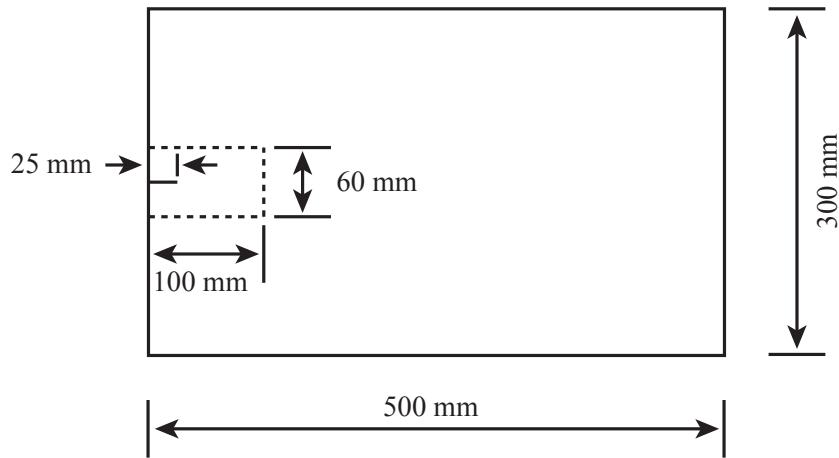


Figure 5.28: Geometry for the pressurized crack example. The geometry is made large enough such that stress waves reflected from the boundary do not reach the propagating crack before the end of the simulation. The dashed lines indicate the extents of the area in which crack propagation will occur before the simulation end time.

The model parameters are  $\rho = 1230 \text{ kg/m}^3$ ,  $E = 4.55 \text{ GPa}$ ,  $\nu = 0.31$ , and  $G_c = 50 \text{ J/m}^2$  and plane strain is assumed. The corresponding dilatational, shear, and Rayleigh wave speeds are  $v_d = 2264 \text{ m/s}$ ,  $v_s = 1188 \text{ m/s}$ ,  $v_R = 1104 \text{ m/s}$ . The length scale was chosen to be  $\ell_0 = 5 \times 10^{-4} \text{ m}$  leading to a uniaxial critical stress of 4.9 MPa

(see Section 4.1).

The mesh is constructed as a  $C^2$ -continuous cubic T-spline. To improve computational efficiency, an initial coarse mesh is constructed and then refined *a priori* only in the area indicated by the dashed lines in Figure 5.28. The initial mesh, with a mesh size of  $h = 0.01$  m, was refined locally to achieve a mesh size of  $h = 1.5265 \times 10^{-4}$  m in the area inside the dashed lines. The final mesh is shown in Figure 5.29. This mesh contains 258,879 bi-cubic basis functions.

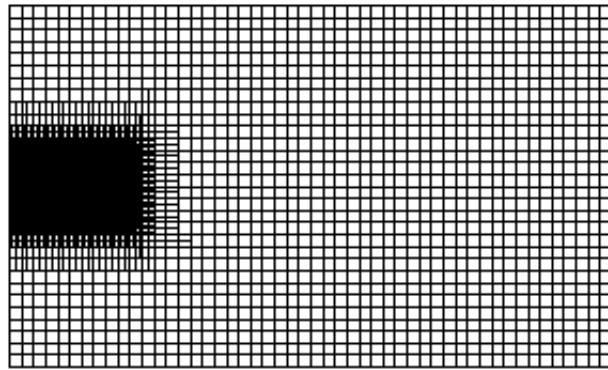


Figure 5.29: The mesh for the pressurized crack problem. The mesh is constructed as a  $C^2$ -continuous cubic T-spline and refined locally in the area where the crack will propagate (the area that appears solid black in the figure). The coarse elements have a mesh size of  $h = 0.01$  m, which transitions to a fine mesh size of  $h = 1.5265 \times 10^{-4}$  m. This mesh contains 258,879 bi-cubic basis functions.

A pressure load inside the crack is used to induce crack propagation. To apply a pressure load to a crack defined by a phase-field we consider the weak form of the momentum equation:

$$(\rho \ddot{\mathbf{u}}, \mathbf{w})_{\Omega} + (\boldsymbol{\sigma}, \nabla \mathbf{w})_{\Omega} = (\mathbf{h}, \mathbf{w})_{\partial\Omega} + (\mathbf{b}, \mathbf{w})_{\Omega} \quad (5.7)$$

where  $\mathbf{b}$  is an applied external body force. Letting  $p$  be the pressure load, we then define the body force  $\mathbf{b} = \nabla [(1 - c)(-p)]$  such that the body force is applied when  $c$  is near

zero and is negligible when  $c$  is near one. Substituting this body force back into the weak form equation and integrating by parts we get

$$\begin{aligned} & (\rho \ddot{\mathbf{u}}, \mathbf{w})_{\Omega} + (\boldsymbol{\sigma}, \nabla \mathbf{w})_{\Omega} \\ &= (\mathbf{h}, \mathbf{w})_{\partial\Omega} + ((1 - c)p, \nabla \cdot \mathbf{w})_{\Omega} - ((1 - c)p, \mathbf{w} \cdot \mathbf{n})_{\partial\Omega}. \end{aligned} \quad (5.8)$$

The pressure load applied to the current example was varied over time as shown in Figure 5.30a. The load was also restricted spatially to an area around the initial crack (see Figure 5.30b) so that the propagating crack was not pressurized.

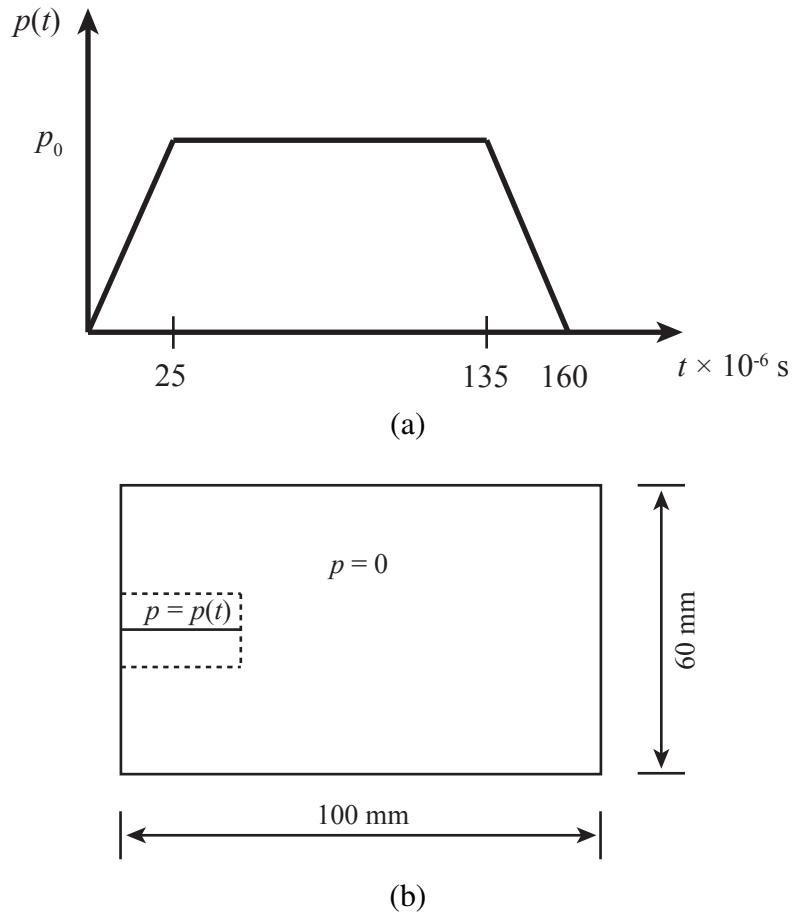
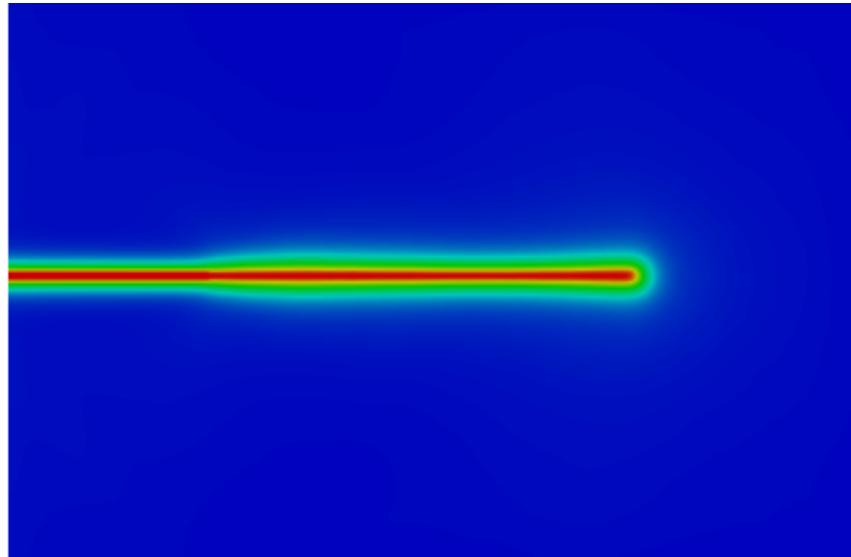


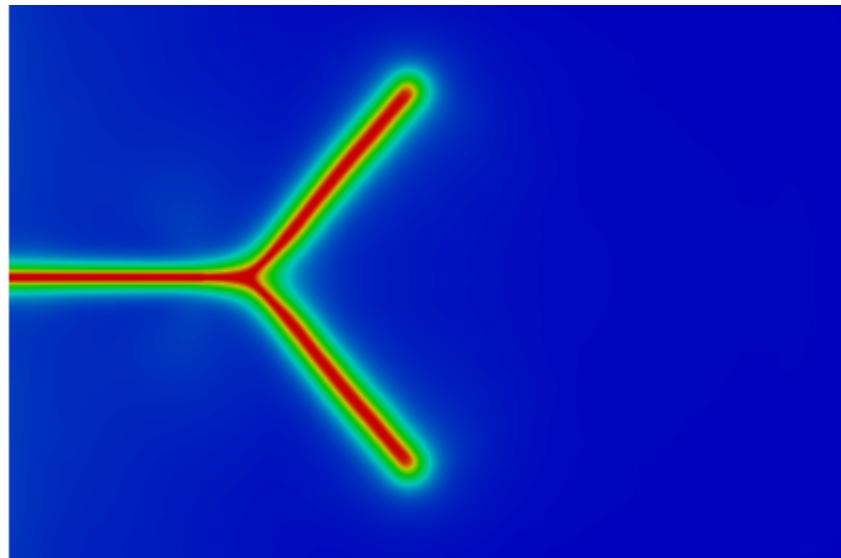
Figure 5.30: The pressure function is varied temporally as shown in (a). To prevent the pressure load from being applied to the newly created crack, the applied pressure load is also limited to a small area around the initial crack as shown in (b).

Two loading conditions were studied for this problem:  $p_0 = 4 \text{ MPa}$  and  $p_0 = 8 \text{ MPa}$ . A single crack propagated without branching under the former load and branching occurred under the latter load as shown in Figure 5.31. We emphasize again that no additional numerical techniques are required to capture this behavior.

For this example, we also compare the second and fourth-order phase-field theories. Figure 5.32 shows plots of the elastic strain energy over time for both theories and pressure loads. These plots show that there is good agreement in the elastic response. Figure 5.33 shows plots of the crack tip velocity over time for both theories and pressure loads. As can be seen, the fourth-order theory tends to give slightly higher crack tip velocities, but the overall trend is similar. The maximum velocity is also in good agreement with experiments (see Ravi-Chandar and Knauss [82]).

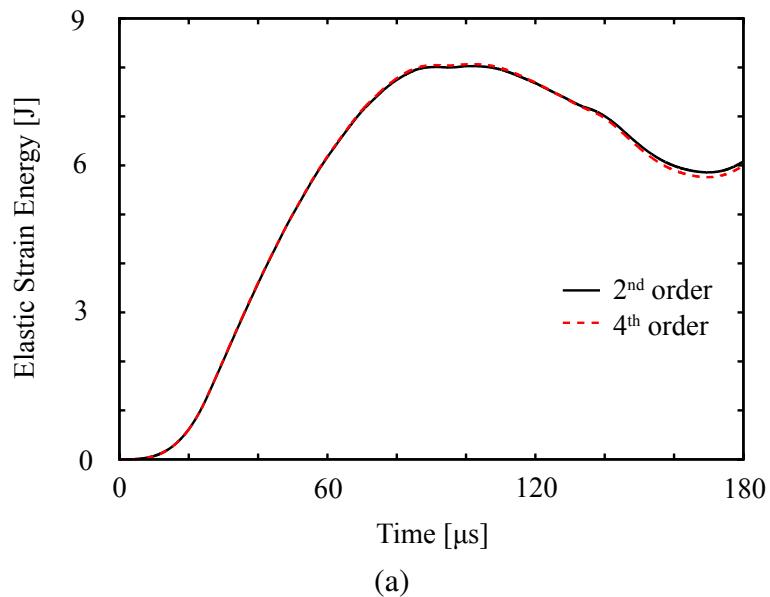


(a)

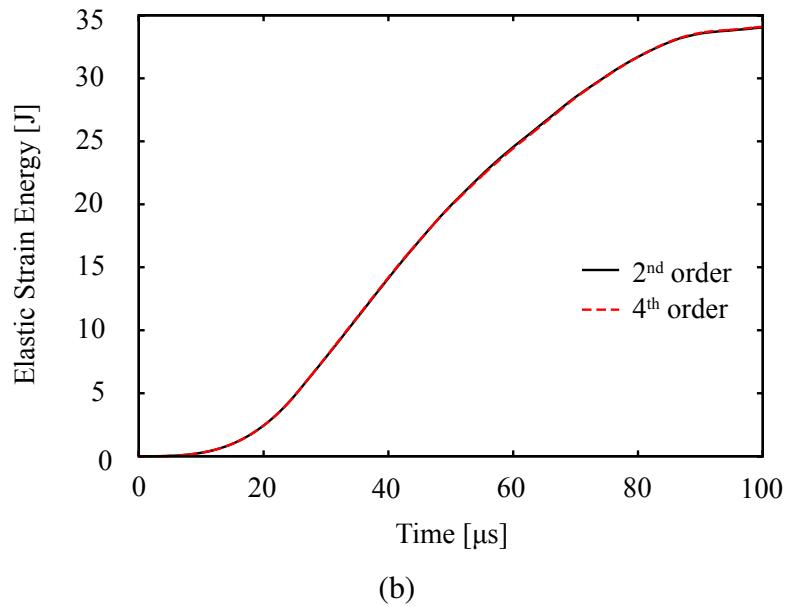


(b)

Figure 5.31: Phase-field plots for (a)  $p_0 = 4$  MPa and (b)  $p_0 = 8$  MPa computed using the second-order phase-field theory. The results for the fourth-order phase-field theory are qualitatively similar and are not shown here.



(a)



(b)

Figure 5.32: Elastic strain energy over time for (a)  $p_0 = 4 \text{ MPa}$  and (b)  $p_0 = 8 \text{ MPa}$ . The result for the second and fourth-order phase-field theories are compared.

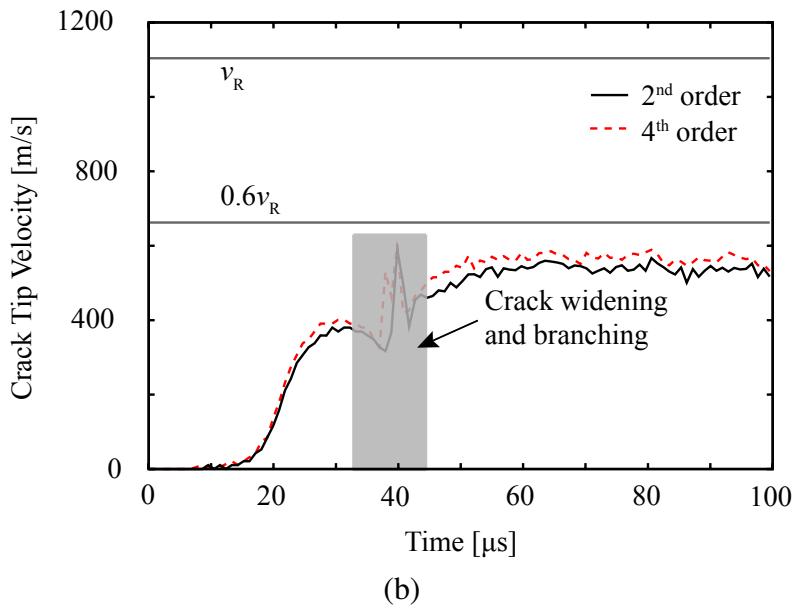
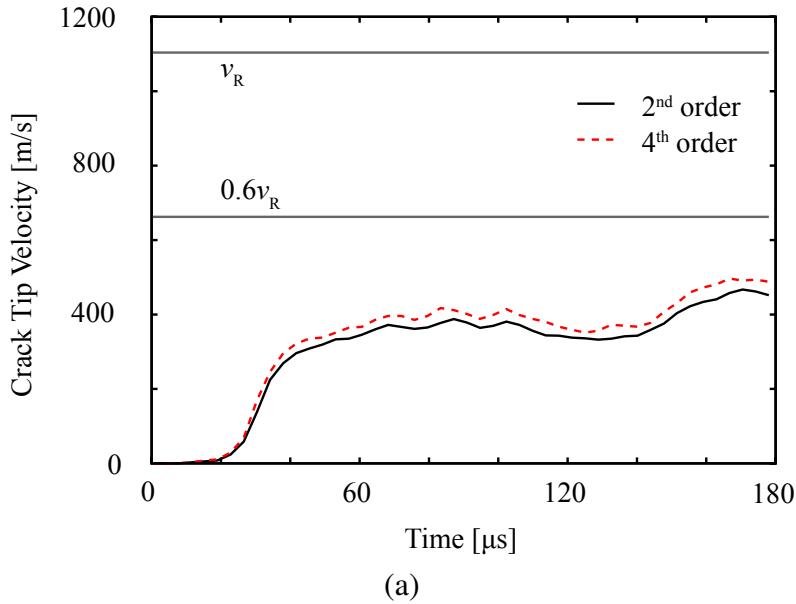


Figure 5.33: Crack tip velocity over time for (a)  $p_0 = 4$  MPa and (b)  $p_0 = 8$  MPa. The result for the second and fourth-order phase-field theories are compared. The fourth-order theory consistently gives higher crack tip velocities.

## 5.5 Pressurized cylinder with solid elements

A major benefit of the phase-field formulation presented here is that it extends easily to three dimensions. In this example, we show a three-dimensional computation of a pressurized cylinder with a spherical end cap. The input geometry for the simulation is shown in Figure 5.34 where symmetry was used to reduce the computational cost. The initial crack was modeled by an induced phase-field (see Appendix B.1). A linearly increasing hydrostatic pressure load,  $p$ , was applied to the inner surface as  $p = 50t$  MPa where  $t$  is the current time.

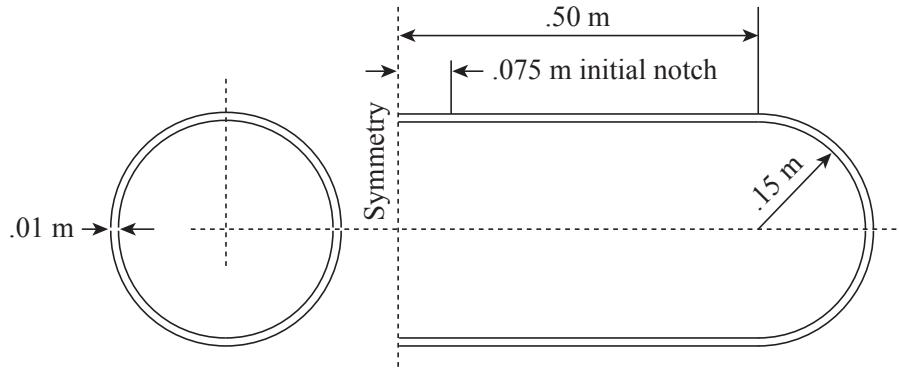


Figure 5.34: Geometry and symmetry conditions for the pressure vessel simulation. The mesh is a three-dimensional thickened T-spline.

The model parameters are  $\rho = 8000 \text{ kg/m}^3$ ,  $E = 190 \text{ GPa}$ ,  $\nu = 0.3$ , and  $\mathcal{G}_c = 2.213 \times 10^4 \text{ J/m}^2$ . The corresponding dilatational, shear, and Rayleigh wave speeds are  $v_d = 5654 \text{ m/s}$ ,  $v_s = 3022 \text{ m/s}$ ,  $v_R = 2803 \text{ m/s}$ . The length scale was chosen to be  $\ell_0 = 2.5 \times 10^{-3} \text{ m}$  leading to a uniaxial critical stress of 298 MPa (see Section 4.1).

To construct the mesh, a  $C^2$ -continuous cubic T-spline mid-surface was first modeled in Rhino, a commercial CAD software package, using the AutoDesk T-Splines Plug-in for Rhino. The initial mid-surface mesh had a mesh size of  $h \approx 0.01 \text{ m}$ . Af-

ter export, the surface was thickened with  $C^1$ -continuous quadratic functions such that there were eight Bézier elements (eleven functions) through the thickness. To get the final mesh, we used the adaptive refinement scheme described in Section 5.3.2. The refinement was applied to the mid-surface mesh at each iteration until  $h \approx \ell_0/2$  in the area of the crack. A new volume mesh was created from the updated mid-surface mesh at each iteration. The final mesh is shown in Figure 5.35. This mesh contains 862,100 basis functions.

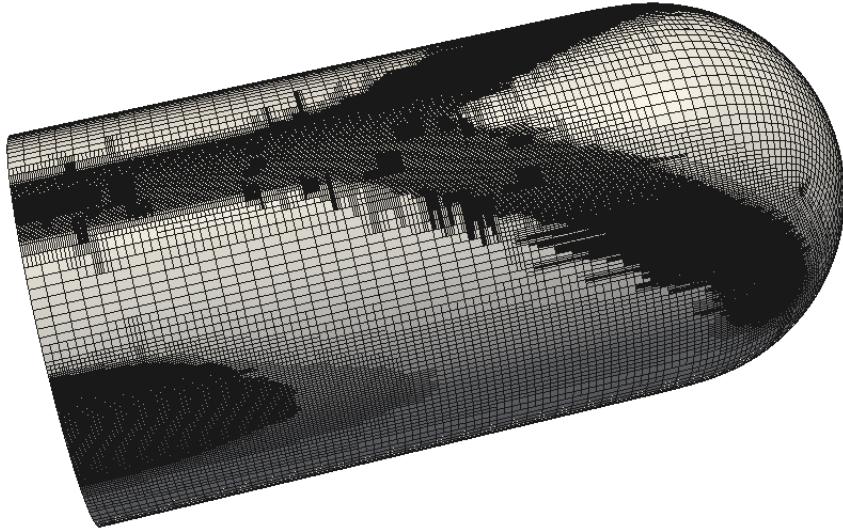


Figure 5.35: The final mesh for the pressurized cylinder example problem. The volumetric mesh was constructed by thickening a mid-surface mesh. The refinement was performed using the adaptive refinement scheme described in Section 5.3.2 which resulted in a final mesh containing 862,100 basis functions.

The simulations were performed using the staggered integration scheme described in Section 3.7.2 with the momentum equation being solved explicitly using the HHT- $\alpha$  method with  $\alpha = -0.3$ . An adaptive time step of  $\Delta t = 0.9\Delta t_{\text{crit}}$  was used (see Section 5.3 for a definition of  $\Delta t_{\text{crit}}$ ). To compute  $\Delta t_{\text{crit}}$ , we used the power iteration algorithm presented by Benson [19]. The resulting phase-field is shown at several time

intervals in Figure 5.36. A post-processed plot of the model at  $t = 1.76 \times 10^{-3}$  s is shown in Figure 5.37 with the displacements scaled by a factor of 5 and the area of the model where  $c < 0.05$  removed from the visualization.

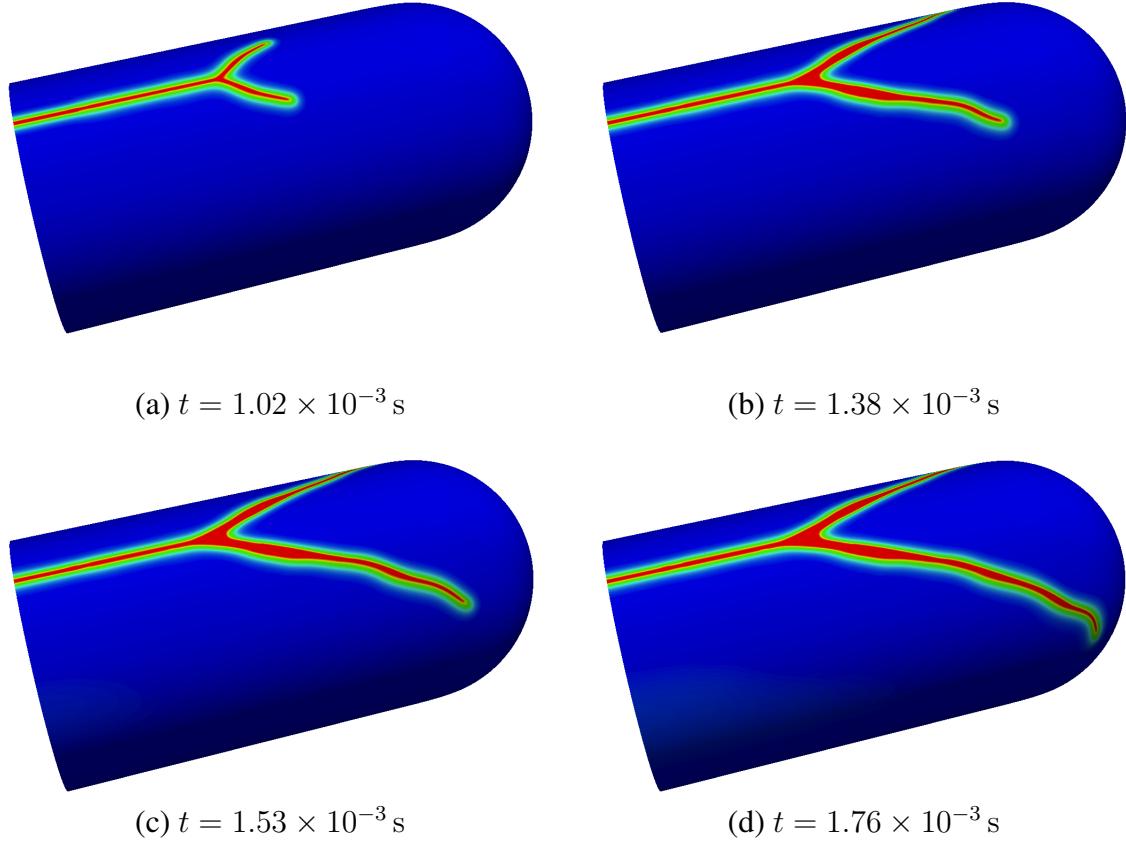


Figure 5.36: The results of the pressurized cylinder example. The phase-field is shown.

Figure 5.38 shows two cross-section views of the crack at different times in the simulation. These cross-sections show the ability of the phase-field model to capture three-dimensional characteristics of a crack. We emphasize that the computation for this three-dimensional model did not require any additional algorithmic complexity compared to the two-dimensional models shown previously.

**Remark 5.4.** This model demonstrates several key features of isogeometric analysis.

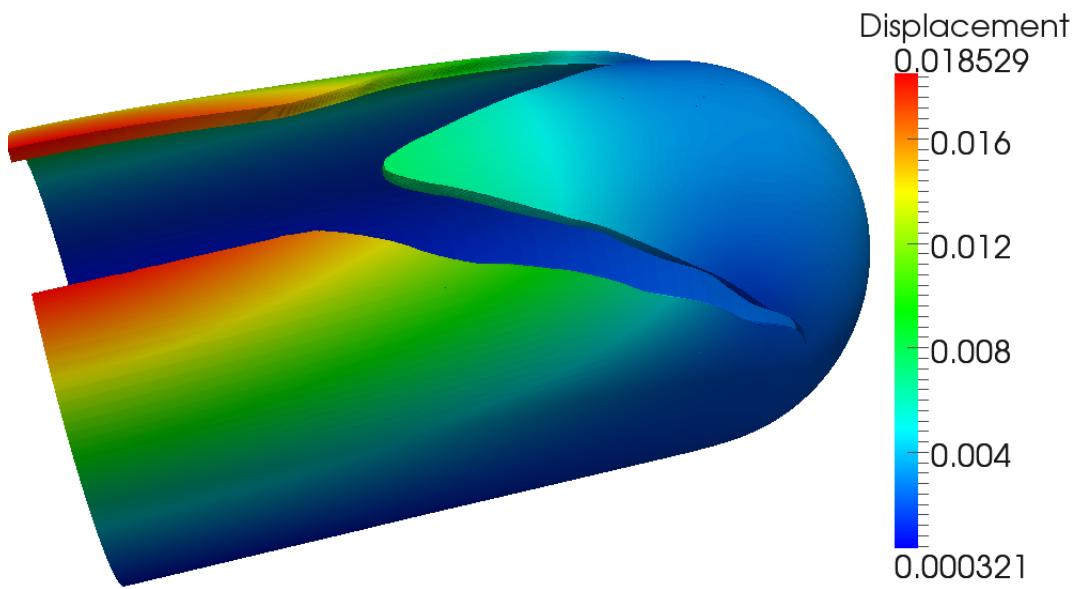


Figure 5.37: A post-processed plot of the pressure vessel example at  $t = 1.76 \times 10^{-3}$  s. The displacements have been scaled by a factor of 5 and areas of model where  $c < 0.05$  have been removed from the plot. Displacement is measured in meters.

First, the initial mid-surface model was constructed in a commercial CAD software package. This model was used directly to construct the analysis model, *i.e.*, there is no intermediate meshing step. Second, the smoothness of the CAD geometry is represented exactly by the analysis model. Finally, refinement was performed directly on the CAD model and the exact CAD geometry was maintained at each iteration. This is illustrated in Figure 5.35.

## 5.6 Ductile fracture of a plate subject to an impulsive load

In this example we model the ductile fracture of a steel plate subject to an impulsive load. The model is based on underwater explosion experiments described by Webster [106]. Examples of the experimental results are shown in Figure 5.39. The different fracture patterns evident in the figure depend on material parameters and loading

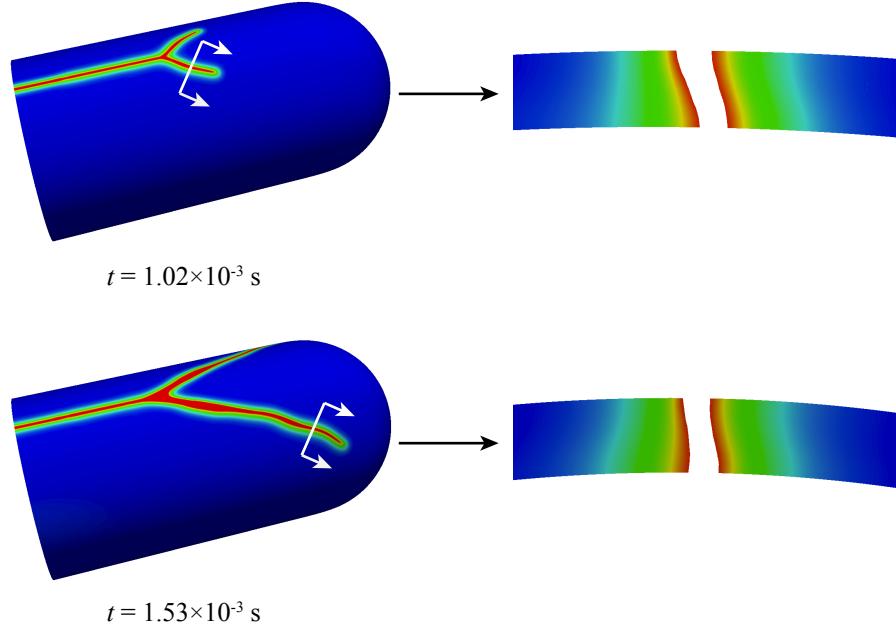


Figure 5.38: Cross section views showing the three-dimensional phase-field profiles of the crack surfaces.

conditions.

The input geometry and boundary conditions are shown in Figure 5.40. We model two boundary condition cases: sliding ( $u_z = 0$ ) and clamped ( $u_x = u_y = u_z$ ). The geometry is modeled using multiple quadratic volumetric NURBS patches. This allows us to represent the shape of the circular plate exactly. The boundary of the patches is shown in Figure 5.40(b). We note the basis is  $C^0$ -continuous across the patch boundaries. The discretized model contains 113,936 trivariate quadratic NURBS basis functions with a nominal size of  $h = 6.35 \times 10^{-3}$  m. This results in two quadratic elements (four functions) through the thickness of the plate.

The elastoplastic material parameters are (see Webster [106])  $\rho = 7780 \text{ kg/m}^3$ ,  $E = 206 \text{ GPa}$ ,  $\nu = 0.281$ ,  $\sigma_y = 355 \text{ MPa}$ , and perfect plasticity is assumed. The

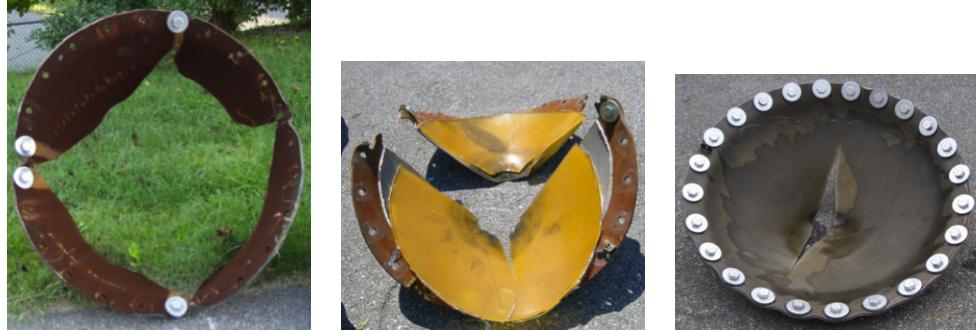


Figure 5.39: Experimental results showing several crack patterns for a plate subjected to an impulsive load (pictures courtesy of Roshdy Barsoum).

corresponding elastic dilatational, shear, and Rayleigh wave speeds are  $v_d = 5825$  m/s,  $v_s = 3214$  m/s,  $v_R = 2972$  m/s. For this initial simulation, the length scale was chosen to be  $\ell_0 = 1.27 \times 10^{-2}$  m and the critical energy release rate to be  $\mathcal{G}_c = 10^6$  J/m. In the one-dimensional homogeneous case this gives  $\sigma_c = 2.8\sigma_y$ .

The load condition is modeled using an impulse pressure function combined with a Gaussian distribution centered on the face of the plate. The impulsive load is given by (5.9) and the distribution by (5.10). The parameters  $A$ ,  $B$ ,  $t_0$ , and  $t_1$  are used to fit the curve to the desired profile and  $p_0$  is a residual pressure. The parameters used for this example are shown in Table 5.3. These parameters give a peak pressure of 45 MPa at  $t = 20 \mu\text{s}$ . The impulse and distribution functions are shown in Figure 5.41.

$$\tilde{p}(t) = At \exp(-Bt) + \begin{cases} 0 & t < t_0 \\ p_0 \frac{1}{2} \left[ 1 + \sin \left( \pi \frac{t - t_0}{t_1 - t_0} - \frac{\pi}{2} \right) \right] & t_0 \leq t < t_1 \\ p_0 & t \geq t_1 \end{cases} \quad (5.9)$$

$$g(r) = \exp \left( -\frac{r^2}{2\tau^2} \right) \quad (5.10)$$

$$p(t, r) = \tilde{p}(t) \times g(r) \quad (5.11)$$

The simulations were performed using the staggered integration scheme described in Section 3.7.2. The momentum equation was solved implicitly using the HHT- $\alpha$  method with  $\alpha = -0.3$ . A fixed time step of  $\Delta t = 2 \mu\text{s}$  was used.

For large deformation simulations mesh elements that have lost all stiffness may

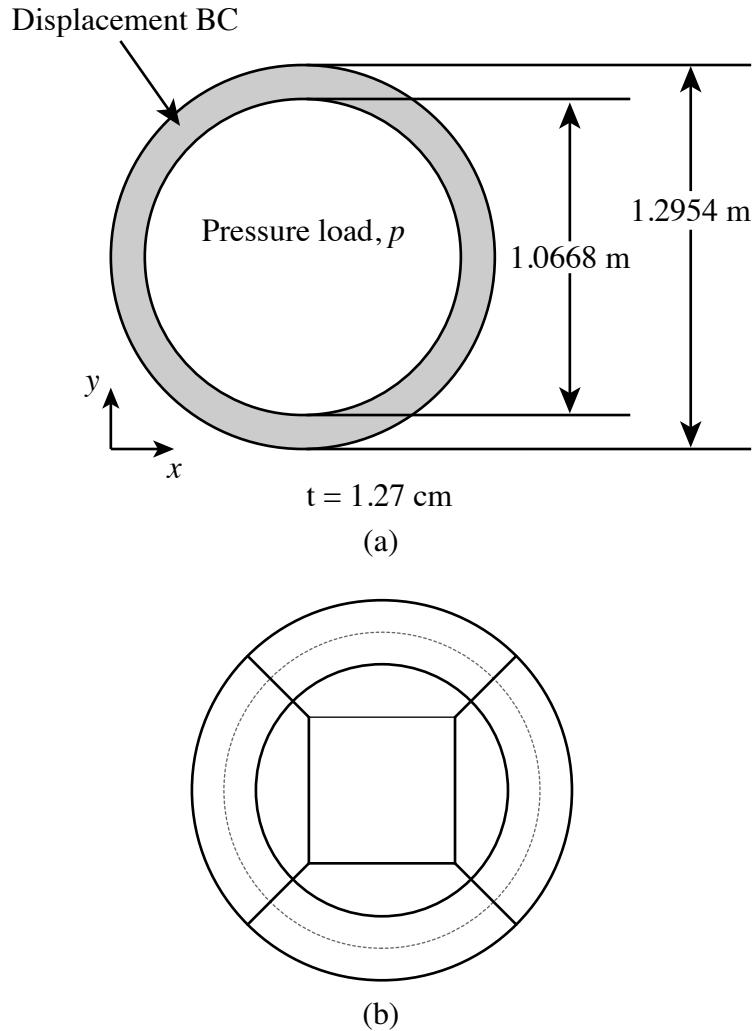


Figure 5.40: The (a) geometry and boundary conditions and (b) NURBS patches for the impulsively loaded plate example. The boundary conditions where modeled as both sliding ( $u_z = 0$ ) and clamped ( $u_x = u_y = u_z$ ). The dotted line in (b) shows the edge of the displacement boundary condition.

Parameter	Value
$A$	$2.25 \times 10^{12} \exp(1) \text{ Pa/s}$
$B$	50,000
$t_0$	$20 \mu\text{s}$
$t_1$	$100 \mu\text{s}$
$p_0$	2 MPa
$\tau$	0.30

Table 5.3: Model parameters for the pressure function (5.11).

become inverted. To prevent these elements from adversely effecting the simulation they are removed from the stiffness calculation. To determine when an element should be removed from the calculation we check the value of the phase-field at each integration point. If the phase-field is below a value of 0.1 (only one percent of the material strength remains) at each integration point of an element, then the element is removed from the stiffness calculation. Its mass remains, however, to prevent ill-conditioning of the system.

The symmetry of the geometry, boundary conditions, and load prevent cracks

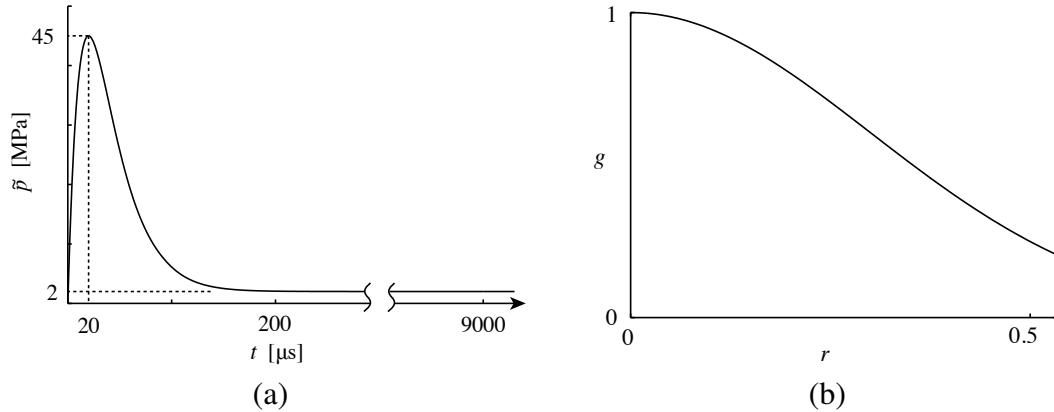


Figure 5.41: The distribution of the impulse load as a function of (a) time and (b) radius,  $r$ , measured from the center of the plate.

from nucleating in numerical simulations. To overcome this problem, we have created initial cracks at the center of the plate using the history functional as described in Appendix B.1. We have modeled three initial states as shown in the top row of Figure 5.42. The resulting crack topology is also shown in this figure for both the sliding and clamped boundary conditions. Notice that both the initial crack topology and the boundary condition influences the crack behavior.

Figures 5.43 and 5.44 show the equivalent plastic strain (EPS) and displaced profiles for the sliding and clamped boundary condition. As expected, there is a clear ductile zone around the cracks. Significant yielding is also present in areas where cracks have not formed. Comparing these results with the experimental results shown in Figure 5.39 we see qualitative similarities with the sliding boundary case. The differences could be partially attributed to inaccurately capturing the true boundary condition. Including other fracture mechanisms as discussed in Remark 3.5 may also improve the accuracy of the results.

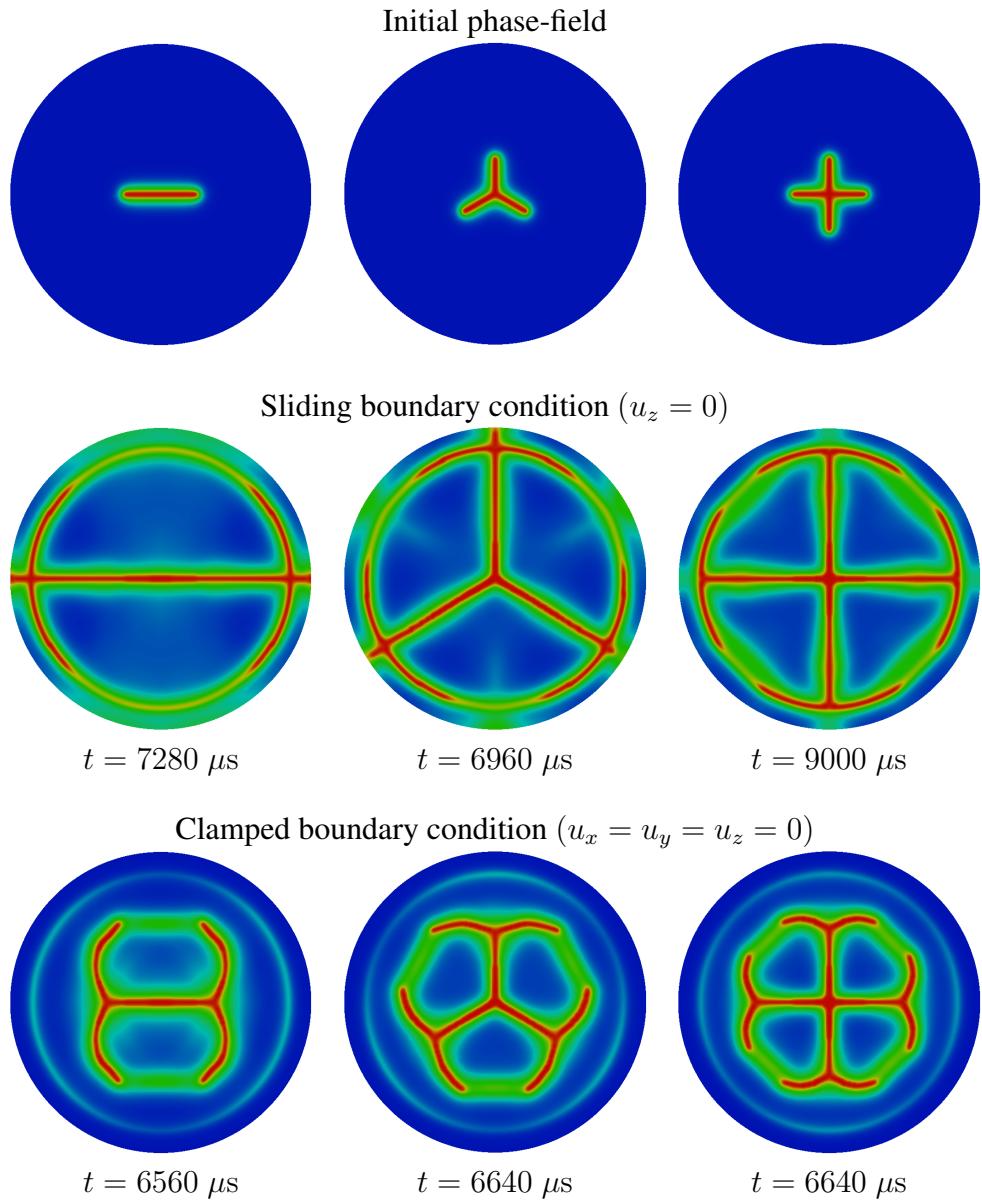


Figure 5.42: The crack patterns for the sliding and clamped boundary conditions. The patterns depend on both the initial state and boundary conditions.

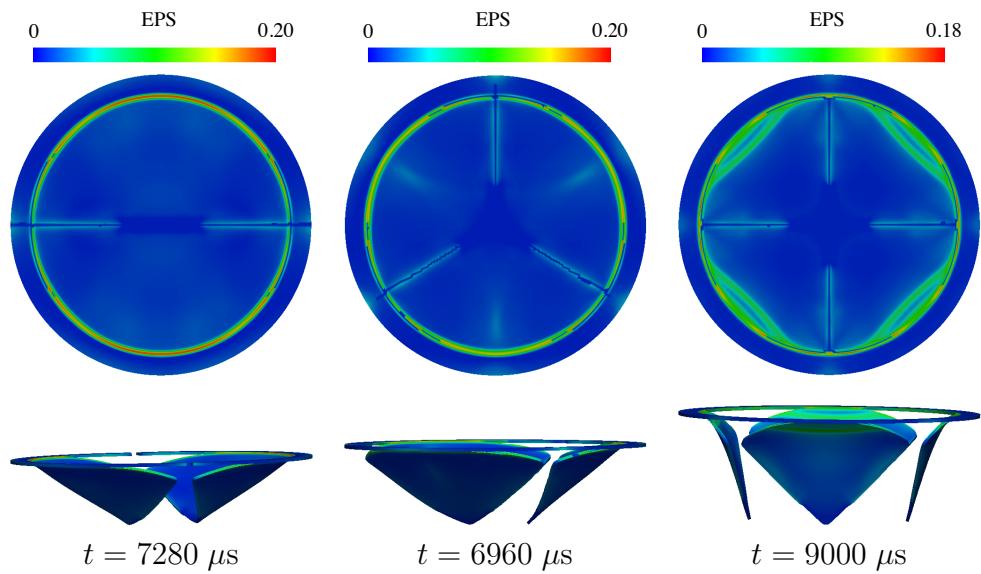


Figure 5.43: The equivalent plastic strain (EPS) and displacement profiles for the sliding boundary condition. Areas of the model where  $c < 0.15$  have been removed from the displacement plot.

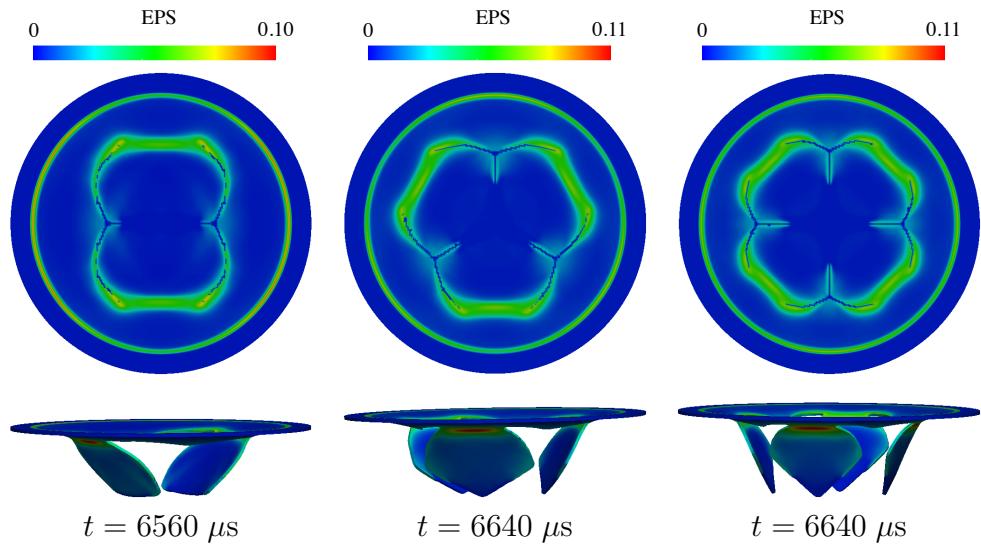


Figure 5.44: The equivalent plastic strain (EPS) and displacement profiles for the clamped boundary conditions. Areas of the model where  $c < 0.15$  have been removed from the displacement plot.

## 5.7 3D block with random nucleation sites

This final example demonstrates that phase-field models have the potential to become a predictive tool for complex three-dimensional crack initiation and propagation problems. The geometry for this example is a rectangular block that is 0.01 m high by 0.01 m wide by 0.005 m deep. Several nucleation sites—spherical areas where an initial strain energy history is imposed as described in Appendix B.1—are randomly distributed throughout the body (see Figure 5.45(a)). These sites create a representation of an inhomogeneous material and allow stress concentrations to develop. A displacement boundary condition is imposed on the top and bottom surfaces by applying a velocity,  $v$ , in the direction normal to the surfaces where

$$v = \begin{cases} \frac{t}{2.5 \mu\text{s}} 2.5 \text{ m/s} & t \leq 2.5 \mu\text{s} \\ 2.5 \text{ m/s} & t > 2.5 \mu\text{s}. \end{cases} \quad (5.12)$$

We use the fourth-order small strain brittle formulation introduced in Section 3.2.2 and the geometry is discretized by a single quadratic B-spline patch with 16,384,000 uniform Bézier elements ( $h = \ell_0/2$ ). The model parameters are  $\rho = 8000 \text{ kg/m}^3$ ,  $E = 190 \text{ GPa}$ ,  $\nu = 0.3$ , and  $\mathcal{G}_c = 2.213 \times 10^4 \text{ J/m}^2$  and the length scale was chosen to be  $\ell_0 = 6.25 \times 10^{-5} \text{ m}$ . The staggered scheme from Section 3.7.2 was used with the explicit HHT- $\alpha$  method with  $\alpha = -0.3$  to perform the time integration of the momentum equation.

A sequence of figures showing the crack growth over time for this problem is presented in Figure 5.45(a)-(f). The initial randomly distributed nucleation sites are shown in Figure 5.45(a). Figure 5.45(b) shows the crack beginning to propagate from one of the nucleation sites. In Figure 5.45(c)-(d) we see multiple branches developing as the crack progresses. In Figure 5.45(e)-(f) we can see separate branches of the crack merging and additional crack fronts developing and changing direction in the upper and lower corners on the left side of the block.

There are several things about this example that are worth reemphasizing. First, the strain energy history function provides a simple, mesh independent mechanism for adding initial defects to a model. Second, as demonstrated previously, once the mesh is sufficiently fine to capture the gradients of the phase-field the crack behavior is mesh independent. In this example, there is no noticeable mesh dependency on the shape of the cracks. And finally, all of the behavior of the cracks is completely determined by the coupled momentum and phase-field equations. There are no ad hoc rules to determine when a crack propagates, bifurcates, merges with another crack, or changes direction. These properties result in a remarkably stable and robust numerical model.

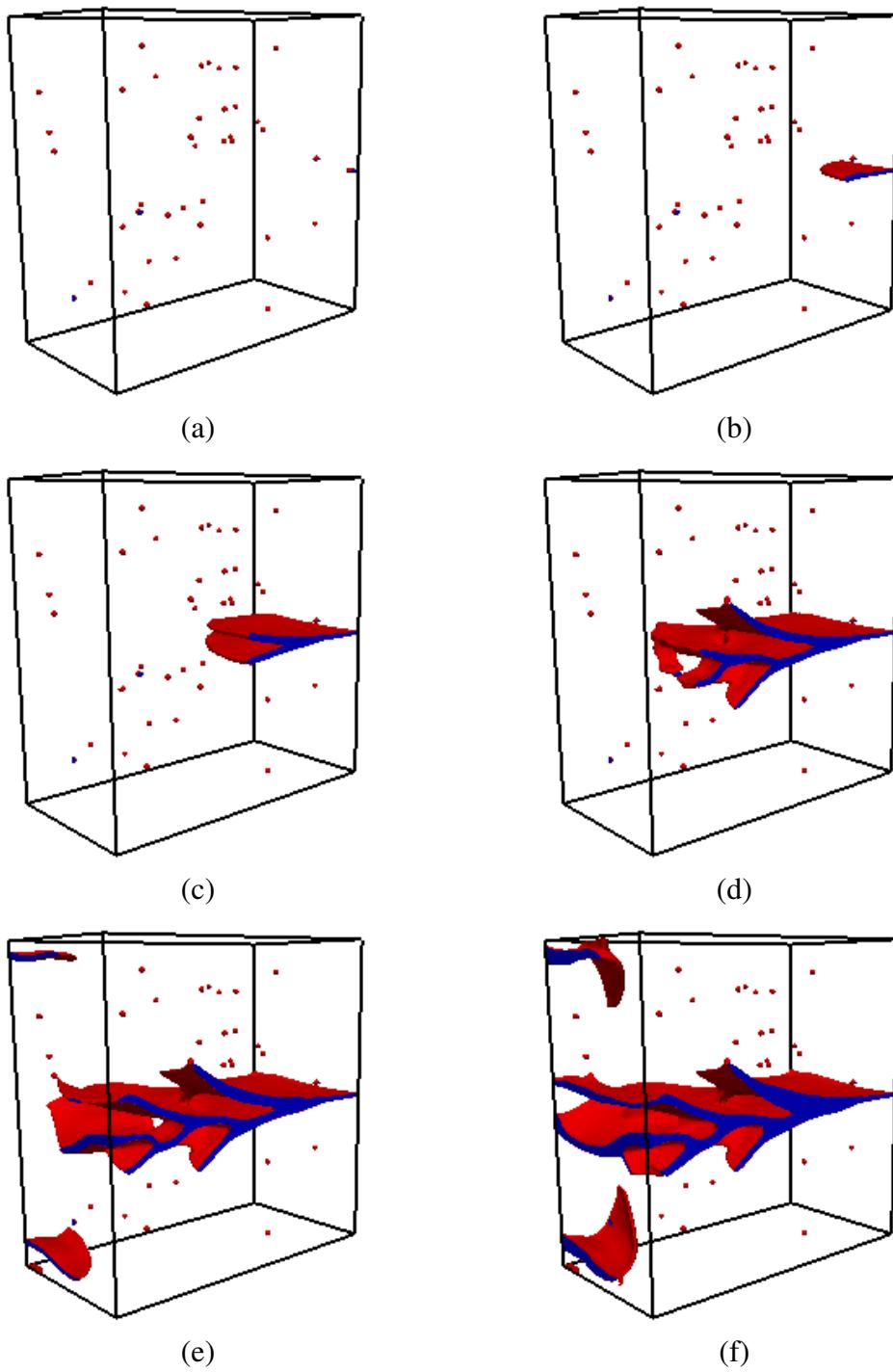


Figure 5.45: Three-dimensional crack propagation over time. The initial geometry (a) is seeded with randomly distributed nucleation sites. The progression of the crack from (b) to (f) shows the ability of the phase-field model to capture branching, turning, and merging of cracks in three dimensions.

# **Chapter 6**

## **Conclusions and future work**

In this dissertation, phase-field models for dynamic brittle and ductile fracture have been presented. The phase-field models provide a continuous representation of a crack and remove the need to numerically track discontinuities in the displacement field. Additionally, the models account for complex crack behavior such as bifurcation and direction changes without the need of additional ad hoc rules. This greatly reduces the algorithmic complexity of the implementation, especially in three dimensions. This work has also shown that phase-field models are capable of accurately modeling real world fracture problems.

The contributions in this dissertation satisfy the required components of the Computational Science Engineering and Mathematics program. Grouped by component, the contributions are summarized as

### **Area A: Applicable mathematics**

- Generalization of the dynamic formulation via a microforce derivation
- Formulation and analysis of a fourth-order phase-field approximation
- Formulation of a ductile fracture model

### **Area B: Numerical analysis and scientific computation**

- Implementation of a general isogeometric analysis code based on Bézier extraction
- Development of time discretization algorithms for the coupled system of the momentum/phase-field equations
- Development of an adaptive  $h$ -refinement scheme for the spatial discretization

### **Area C: Mathematical modeling and applications**

- Simulations of dynamic brittle and ductile fracture problems
- Simulations of large, three-dimensional fracture problems
- Application of the phase-field model to pressurized cracks
- Use of T-splines for adaptive refinement and simulation of volumetric models

There are many potential research directions remaining for this work. We have shown that the fourth-order theory improves convergence properties at the length scales typical of numerical calculations, but  $\Gamma$ -convergence remains an open problem. In the case of ductile fracture there are many mechanisms that could potentially play an important role in the fracture process. For example, viscoplasticity and thermal effects may influence ductile fracture. Plastic work may also play an important role in crack propagation. These effects may be included in the model through constitutive theory or by modifying the term that drives the evolution of the phase-field. A more thorough study of the strain energy history functional,  $\mathcal{H}$ , is also needed. This functional provided an efficient mechanism for enforcing irreversibility, but as we have seen, it

changes the behavior of the model in ways that are not fully understood. Finally, on a more practical side, there is much potential in applying the phase-field model to pressurized crack problems, particularly in the geosciences where complex, three-dimensional crack topologies and interactions are common.

## **Appendices**

## Appendix A

### Dimensionless form of the phase-field equations

To improve the conditioning and scaling of the monolithic strategy discussed in Section 3.7.1, we consider the strong form equations (3.54) in their dimensionless form. To arrive at the dimensionless formulation we define a length scale  $L_0$  and time scale  $T_0$  as

$$L_0 = \frac{\mathcal{G}_c}{C_h E}, \quad T_0 = L_0 \sqrt{\frac{\rho}{C_h E}} \quad (\text{A.1})$$

where the non-dimensional constant  $C_h$  is used to control the scaling of the problem. By introducing non-dimensional space and time coordinates, and a non-dimensional displacement field as

$$x^* = x/L_0, \quad t^* = t/T_0, \quad u^* = u/L_0 \quad (\text{A.2})$$

we arrive at

$$\left\{ \begin{array}{l} \frac{\partial \sigma_{ij}^*}{\partial x_j^*} = \frac{\partial^2 u_i^*}{\partial (t^*)^2} \\ [4\ell_0^*(1-k)\mathcal{H}^* + 1] c - 4(\ell_0^*)^2 \frac{\partial^2 c}{\partial (x_i^*)^2} = 1 \end{array} \right. \quad (\text{A.3})$$

with

$$\sigma^* = \frac{\sigma}{C_h E}, \quad \mathcal{H}^* = \frac{\mathcal{H}}{C_h E}, \quad \ell_0^* = \frac{\ell_0}{L_0} \quad (\text{A.4})$$

In practice, we have found that choosing  $C_h$  such that the size of the elements in the mesh have an area equal to one yields good results.

To compute the Lagrangian from the non-dimensional solution we note that

$$\begin{aligned} L_{\ell_0}(\dot{\mathbf{u}}, \mathbf{u}, c) &= L_{\ell_0}^*(\dot{\mathbf{u}}^*, \mathbf{u}^*, c) \\ &= \hat{E} L^N \left\{ \int_{\Omega_0^*} \frac{1}{2} \dot{u}_i^* \dot{u}_i^* - \{(1-k)c^2 + k\} \psi_e^{+,*}(\boldsymbol{\gamma}) - \psi_e^{-,*}(\boldsymbol{\gamma}) d\Omega_0^* \right. \\ &\quad \left. - \int_{\Omega_0^*} \left( \frac{(1-c)^2}{4\ell_0^*} + \ell_0^* \frac{\partial c}{\partial X_i^*} \frac{\partial c}{\partial X_i^*} \right) d\Omega_0^* \right\} \end{aligned} \quad (\text{A.5})$$

and for the normalized (by  $\mathcal{G}_c$ ) dissipation rate

$$\begin{aligned} \dot{\Gamma} &= \frac{d\Gamma}{dt} = \frac{d}{dt} \int_{\Omega_0} \left( \frac{(1-c)^2}{4\ell_0} + \ell_0 \frac{\partial c}{\partial X_i} \frac{\partial c}{\partial X_i} \right) d\Omega_0 \\ &= L^{N-2} \left( \frac{\rho}{\hat{E}} \right)^{-1/2} \frac{d}{dt^*} \int_{\Omega_0^*} \left( \frac{(1-c)^2}{4\ell_0^*} + \ell_0^* \frac{\partial c}{\partial X_i^*} \frac{\partial c}{\partial X_i^*} \right) d\Omega_0^*. \end{aligned} \quad (\text{A.6})$$

**Remark:** The same implementation can be used to compute with both the dimensional and non-dimensional forms of the equations. If the dimensional form of the equations has been implemented then the non-dimensional form can be computed by setting the material parameters  $\mathcal{G}_c = 1$ ,  $\rho = 1$ , and  $E = 1/C_h$ ; scaling the input geometry by  $1/L_0$ ; scaling the time steps by  $1/T_0$ ; and using  $\ell_0^*$  in place of  $\ell_0$ .

## A.1 Non-dimensional parameters for branching crack problem.

The example discussed in Section 5.2 was solved using the monolithic strategy from Section 3.7.1. As such, the problem was non-dimentionalized, and the non-dimensional parameters that were used are:

$$C_h = 10^{-6}, \quad \hat{E} = C_h E = 32 \times 10^3 \text{ N/m}^2,$$

$$L = \mathcal{G}_c / \hat{E} = 9.375 \times 10^{-5} \text{ m}, \quad T = L \sqrt{\rho / \hat{E}} = 2.594 \times 10^{-5} \text{ s},$$

$$x^* = x/L = 1066.7, \quad y^* = y/L = 426.67,$$

$$\sigma^* = \sigma / \hat{E} = 31.25, \quad t_{\text{end}}^* = 3.115$$

The following length scales (dimensional and non-dimensional) and non-dimensional time steps where used in the numerical simulations:

$$\ell_0^0 = 5 \times 10^{-4} \text{ m}, \quad \ell_0^1 = 2.5 \times 10^{-4} \text{ m}, \quad \ell_0^2 = 1.25 \times 10^{-4} \text{ m},$$

$$\ell_0^{0*} = 5.3333, \quad \ell_0^{1*} = 2.6667, \quad \ell_0^{2*} = 1.3333,$$

$$\Delta t_0^* = 3.5 \times 10^{-3} \text{ s}, \quad \Delta t_1^* = 1.75 \times 10^{-3} \text{ s}, \quad \Delta t_2^* = 8.75 \times 10^{-4} \text{ s}$$

## Appendix B

### Numerical Miscellany

#### B.1 Modeling a preexisting crack in a continuous body

For the numerical examples discussed in this paper, a preexisting crack is used to initialize crack propagation. The initial crack is model as either a discrete crack in the geometry or as an induced crack in the phase-field. For the induced crack, an initial strain-history field is specified such that an initial crack in the phase-field is defined. To define the initial strain-history field we let  $l$  be a line that represents the discrete crack we wish to include and  $d(\mathbf{x}, l)$  is the closest distance from  $\mathbf{x}$  to the line  $l$ . The strain-history field is then defined as

$$\mathcal{H}_0(\mathbf{x}) = B \begin{cases} \frac{\mathcal{G}_c}{4\ell_0} \left( 1 - \frac{d(\mathbf{x}, l)}{\ell_0} \right) & d(\mathbf{x}, l) \leq \ell_0 \\ 0 & d(\mathbf{x}, l) > \ell_0 \end{cases} \quad (\text{B.1})$$

where the magnitude of the scalar  $B$  can be determined by letting  $d = 0$  and substituting  $\mathcal{H}_0$  into (3.54) with  $\partial^2 c / \partial x_i = 0$  to get

$$B = \frac{1}{c} - 1. \quad (\text{B.2})$$

In the examples presented above, we have chosen  $c = 10^{-3}$  to be the value of the phase-field in the initial crack so that  $B = 10^3$ .

There are two important advantages to the this approach: first and foremost, initial cracks can be located anywhere in the domain without reference to the mesh. We believe this may prove highly advantageous in specifying complex surface cracks

in three-dimensional bodies. A second advantage is that it can be easily utilized with non-interpolatory basis functions, such as are used in meshless and, in the examples presented here, isogeometric analysis.

## B.2 Higher order derivatives of basis functions

To compute the solution for the higher-order phase-field theories we need to take higher-order derivatives of the basis functions. In this section we derive the formulas to compute the second derivatives of the basis functions at the element level. This allows solutions for the fourth-order phase-field theory.

We begin with the equation to compute the physical coordinates,  $\mathbf{x} = (x, y, z)$ , in each element:

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{a=1}^{n_{en}} N_a(\boldsymbol{\xi}) \mathbf{x}_a^e \quad (\text{B.3})$$

where  $\boldsymbol{\xi} = (\xi, \eta, \zeta)$  are the coordinates in the parent domain (for two dimensions the  $z$  and  $\zeta$  are dropped). In two dimensions, the first derivatives of the basis functions with respect to the physical coordinates can be computed as

$$N_{a,x} = N_{a,\xi} \xi_{,x} + N_{a,\eta} \eta_{,x} \quad (\text{B.4})$$

$$N_{a,y} = N_{a,\xi} \xi_{,y} + N_{a,\eta} \eta_{,y}, \quad (\text{B.5})$$

which requires the derivatives of the parent coordinates with respect to the physical coordinates. These can be computed in the usual way as

$$\begin{bmatrix} \xi_{,x} & \xi_{,y} \\ \eta_{,x} & \eta_{,y} \end{bmatrix} = \frac{1}{j} \begin{bmatrix} y_{,\eta} & -x_{,\eta} \\ -y_{,\xi} & x_{,\xi} \end{bmatrix} = \begin{bmatrix} x_{,\xi} & x_{,\eta} \\ y_{,\xi} & y_{,\eta} \end{bmatrix}^{-1} \quad (\text{B.6})$$

where  $j$  is the determinant of the Jacobian of  $(x, y)$ .

To compute the second derivatives of the basis functions with respect to the physical coordinates, we start by computing the derivatives with respect to the parent coordinates as

$$N_{a,\xi} = N_{a,x}x_{,\xi} + N_{a,y}y_{,\xi} \quad (\text{B.7})$$

$$N_{a,\eta} = N_{a,x}x_{,\eta} + N_{a,y}y_{,\eta}. \quad (\text{B.8})$$

The second derivatives with respect to the parent coordinates are then

$$N_{a,\xi\xi} = N_{a,xx}x_{,\xi}^2 + N_{a,x}x_{,\xi\xi} + 2N_{a,xy}x_{,\xi}y_{,\xi} + N_{a,y}y_{,\xi\xi} + N_{a,yy}y_{,\xi}^2 \quad (\text{B.9})$$

$$N_{a,\xi\eta} = N_{a,xx}x_{,\xi}x_{,\eta} + N_{a,x}x_{,\xi\eta} + N_{a,xy}(x_{,\xi}y_{,\eta} + x_{,\eta}y_{,\xi}) + N_{a,y}y_{,\xi\eta} + N_{a,yy}y_{,\xi}y_{\eta} \quad (\text{B.10})$$

$$N_{a,\eta\eta} = N_{a,xx}x_{,\eta}^2 + N_{a,x}x_{,\eta\eta} + 2N_{a,xy}x_{,\eta}y_{,\eta} + N_{a,y}y_{,\eta\eta} + N_{a,yy}y_{,\eta}^2. \quad (\text{B.11})$$

These equations leads to the  $3 \times 3$  system

$$\begin{bmatrix} x_{,\xi}^2 & 2x_{,\xi}y_{,\xi} & y_{,\xi}^2 \\ x_{,\xi}x_{,\eta} & x_{,\xi}y_{,\eta} + x_{,\eta}y_{,\xi} & y_{,\xi}y_{\eta} \\ x_{,\eta}^2 & 2x_{,\eta}y_{,\eta} & y_{,\eta}^2 \end{bmatrix} \begin{Bmatrix} N_{a,xx} \\ N_{a,xy} \\ N_{a,yy} \end{Bmatrix} = \begin{Bmatrix} N_{a,\xi\xi} - N_{a,xx}x_{,\xi\xi} - N_{a,y}y_{,\xi\xi} \\ N_{a,\xi\eta} - N_{a,x}x_{,\xi\eta} - N_{a,y}y_{,\xi\eta} \\ N_{a,\eta\eta} - N_{a,xx}x_{,\eta\eta} - N_{a,y}y_{,\eta\eta} \end{Bmatrix}. \quad (\text{B.12})$$

The coefficient matrix and right hand side are known so that we can solve for the second derivatives of the basis functions with respect to the physical coordinates.

In three dimensions, the approach is the same. The derivative of the basis functions with respect to the parent coordinates are

$$N_{a,\xi} = N_{a,x}x_{,\xi} + N_{a,y}y_{,\xi} + N_{a,z}z_{,\xi} \quad (\text{B.13})$$

$$N_{a,\eta} = N_{a,x}x_{,\eta} + N_{a,y}y_{,\eta} + N_{a,z}z_{,\eta} \quad (\text{B.14})$$

$$N_{a,\zeta} = N_{a,x}x_{,\zeta} + N_{a,y}y_{,\zeta} + N_{a,z}z_{,\zeta}, \quad (\text{B.15})$$

and the second derivative with respect to the parent coordinates are

$$\begin{aligned}
N_{a,\xi\xi} &= N_{a,xx}x_{,\xi}^2 + N_{a,yy}y_{,\xi}^2 + N_{a,zz}z_{,\xi}^2 \\
&+ 2N_{a,xy}x_{,\xi}y_{,\xi} + 2N_{a,xz}x_{,\xi}z_{,\xi} + 2N_{a,yz}y_{,\xi}z_{,\xi} \\
&+ N_{a,x}x_{,\xi\xi} + N_{a,y}y_{,\xi\xi} + N_{a,z}z_{,\xi\xi}
\end{aligned} \tag{B.16}$$

$$\begin{aligned}
N_{a,\eta\eta} &= N_{a,xx}x_{,\eta}^2 + N_{a,yy}y_{,\eta}^2 + N_{a,zz}z_{,\eta}^2 \\
&+ 2N_{a,xy}x_{,\eta}y_{,\eta} + 2N_{a,xz}x_{,\eta}z_{,\eta} + 2N_{a,yz}y_{,\eta}z_{,\eta} \\
&+ N_{a,x}x_{,\eta\eta} + N_{a,y}y_{,\eta\eta} + N_{a,z}z_{,\eta\eta}
\end{aligned} \tag{B.17}$$

$$\begin{aligned}
N_{a,\zeta\zeta} &= N_{a,xx}x_{,\zeta}^2 + N_{a,yy}y_{,\zeta}^2 + N_{a,zz}z_{,\zeta}^2 \\
&+ 2N_{a,xy}x_{,\zeta}y_{,\zeta} + 2N_{a,xz}x_{,\zeta}z_{,\zeta} + 2N_{a,yz}y_{,\zeta}z_{,\zeta} \\
&+ N_{a,x}x_{,\zeta\zeta} + N_{a,y}y_{,\zeta\zeta} + N_{a,z}z_{,\zeta\zeta}
\end{aligned} \tag{B.18}$$

$$\begin{aligned}
N_{a,\xi\eta} &= N_{a,xx}x_{,\xi}x_{,\eta} + N_{a,yy}y_{,\xi}y_{,\eta} + N_{a,zz}z_{,\xi}z_{,\eta} \\
&+ N_{a,xy}(x_{,\xi}y_{,\eta} + x_{,\eta}y_{,\xi}) \\
&+ N_{a,xz}(x_{,\xi}z_{,\eta} + x_{,\eta}z_{,\xi}) \\
&+ N_{a,yz}(y_{,\xi}z_{,\eta} + y_{,\eta}z_{,\xi}) \\
&+ N_{a,x}x_{,\xi\eta} + N_{a,y}y_{,\xi\eta} + N_{a,z}z_{,\xi\eta}
\end{aligned} \tag{B.19}$$

$$\begin{aligned}
N_{a,\xi\zeta} &= N_{a,xx}x_{,\xi}x_{,\zeta} + N_{a,yy}y_{,\xi}y_{,\zeta} + N_{a,zz}z_{,\xi}z_{,\zeta} \\
&+ N_{a,xy}(x_{,\xi}y_{,\zeta} + x_{,\zeta}y_{,\xi}) \\
&+ N_{a,xz}(x_{,\xi}z_{,\zeta} + x_{,\zeta}z_{,\xi}) \\
&+ N_{a,yz}(y_{,\xi}z_{,\zeta} + y_{,\zeta}z_{,\xi}) \\
&+ N_{a,x}x_{,\xi\zeta} + N_{a,y}y_{,\xi\zeta} + N_{a,z}z_{,\xi\zeta}
\end{aligned} \tag{B.20}$$

$$\begin{aligned}
N_{a,\eta\zeta} &= N_{a,xx}x_{,\eta}x_{,\zeta} + N_{a,yy}y_{,\eta}y_{,\zeta} + N_{a,zz}z_{,\eta}z_{,\zeta} \\
&+ N_{a,zz}(x_{,\eta}y_{,\zeta} + x_{,\zeta}y_{,\eta})
\end{aligned}$$

$$\begin{aligned}
& + N_{a,xz}(x_{,\eta}z_{,\zeta} + x_{,\zeta}z_{,\eta}) \\
& + N_{a,yz}(y_{,\eta}z_{,\zeta} + y_{,\zeta}z_{,\eta}) \\
& + N_{a,x}x_{,\eta\zeta} + N_{a,y}y_{,\eta\zeta} + N_{a,z}z_{,\eta\zeta}.
\end{aligned} \tag{B.21}$$

These equations lead to the  $6 \times 6$  system

$$\left[ \begin{array}{cccccc}
x_{,\xi}^2 & y_{,\xi}^2 & z_{,\xi}^2 & 2x_{,\xi}y_{,\xi} & 2x_{,\xi}z_{,\xi} & 2y_{,\xi}z_{,\xi} \\
x_{,\eta}^2 & y_{,\eta}^2 & z_{,\eta}^2 & 2x_{,\eta}y_{,\eta} & 2x_{,\eta}z_{,\eta} & 2y_{,\eta}z_{,\eta} \\
x_{,\zeta}^2 & y_{,\zeta}^2 & z_{,\zeta}^2 & 2x_{,\zeta}y_{,\zeta} & 2x_{,\zeta}z_{,\zeta} & 2y_{,\zeta}z_{,\zeta} \\
x_{,\xi}x_{,\eta} & y_{,\xi}y_{,\eta} & z_{,\xi}z_{,\eta} & (x_{,\xi}y_{,\eta} + x_{,\eta}y_{,\xi}) & (x_{,\xi}z_{,\eta} + x_{,\eta}z_{,\xi}) & (y_{,\xi}z_{,\eta} + y_{,\eta}z_{,\xi}) \\
x_{,\xi}x_{,\zeta} & y_{,\xi}y_{,\zeta} & z_{,\xi}z_{,\zeta} & (x_{,\xi}y_{,\zeta} + x_{,\zeta}y_{,\xi}) & (x_{,\xi}z_{,\zeta} + x_{,\zeta}z_{,\xi}) & (y_{,\xi}z_{,\zeta} + y_{,\zeta}z_{,\xi}) \\
x_{,\eta}x_{,\zeta} & y_{,\eta}y_{,\zeta} & z_{,\eta}z_{,\zeta} & (x_{,\eta}y_{,\zeta} + x_{,\zeta}y_{,\eta}) & (x_{,\eta}z_{,\zeta} + x_{,\zeta}z_{,\eta}) & (y_{,\eta}z_{,\zeta} + y_{,\zeta}z_{,\eta})
\end{array} \right] \\
\left\{ \begin{array}{c}
N_{a,xx} \\
N_{a,yy} \\
N_{a,zz} \\
N_{a,xy} \\
N_{a,xz} \\
N_{a,yz}
\end{array} \right\} = \left\{ \begin{array}{c}
N_{a,\xi\xi} - N_{a,x}x_{,\xi\xi} - N_{a,y}y_{,\xi\xi} - N_{a,z}z_{,\xi\xi} \\
N_{a,\eta\eta} - N_{a,x}x_{,\eta\eta} - N_{a,y}y_{,\eta\eta} - N_{a,z}z_{,\eta\eta} \\
N_{a,\zeta\zeta} - N_{a,x}x_{,\zeta\zeta} - N_{a,y}y_{,\zeta\zeta} - N_{a,z}z_{,\zeta\zeta} \\
N_{a,\xi\eta} - N_{a,x}x_{,\xi\eta} - N_{a,y}y_{,\xi\eta} - N_{a,z}z_{,\xi\eta} \\
N_{a,\xi\zeta} - N_{a,x}x_{,\xi\zeta} - N_{a,y}y_{,\xi\zeta} - N_{a,z}z_{,\xi\zeta} \\
N_{a,\eta\zeta} - N_{a,x}x_{,\eta\zeta} - N_{a,y}y_{,\eta\zeta} - N_{a,z}z_{,\eta\zeta}
\end{array} \right\}. \tag{B.22}$$

which can be solved to get the second derivatives of the basis functions with respect to the physical coordinates.

## Bibliography

- [1] M. Aigner, C. Heinrich, B. Juettler, E. Pilgerstorfer, B. Simeon, and A. V. Vuong. Swept Volume Parameterization for Isogeometric Analysis. In Hancock, E. R. and Martin, R. R. and Sabin, M. A., editor, *Mathematics of Surfaces XIII*, volume 5654 of *Lecture Notes in Computer Science*, pages 19–44, 2009.
- [2] I. Akkerman, Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and S. Hulshoff. The role of continuity in residual-based variational multiscale modeling of turbulence. *Computational Mechanics*, 41:371–378, 2008.
- [3] L. Ambrosio and V. M. Tortorelli. Approximation of functional depending on jumps by elliptic functional via  $\Gamma$ -convergence. *Communications on Pure and Applied Mathematics*, 43(8):999–1036, 1990.
- [4] L. Ambrosio and V. M. Tortorelli. On the approximation of free discontinuity problems. *Boll. Un. Mat. Ital. B* (7), 6(1):105–123, 1992.
- [5] H. Amor, J. J. Marigo, and C. Maurini. Regularized formulation of the variational brittle fracture with unilateral contact: Numerical experiments. *Journal of the Mechanics and Physics of Solids*, 57(8):1209–1229, 2009.
- [6] F. Auricchio, L. B. da Veiga, A. Buffa, C. Lovadina, A. Reali, and G. Sangalli. A fully “locking-free” isogeometric approach for plane linear elasticity problems: A stream function formulation. *Computer Methods in Applied Mechanics and Engineering*, 197:160–172, 2007.

- [7] F. Auricchio, L. B. da Veiga, C. Lovadina, and A. Reali. The importance of the exact satisfaction of the incompressibility constraint in nonlinear elasticity: mixed FEMs versus NURBS-based approximations. *Computer Methods in Applied Mechanics and Engineering*, 199:314–323, 2010.
- [8] I. Babuška and J. M. Melenk. The partition of unity method. *International Journal for Numerical Methods in Engineering*, 40(4):727–758, 1997.
- [9] Y. Bazilevs and I. Akkerman. Large eddy simulation of turbulent Taylor-Couette flow using isogeometric analysis and residual-based variational multiscale method. *Journal of Computational Physics*, 229:3402–3414, 2010.
- [10] Y. Bazilevs, V. M. Calo, Y. Zhang, and T. J. R. Hughes. Isogeometric fluid-structure interaction analysis with application to arterial blood flow. *Computational Mechanics*, 38:310–322, 2006.
- [11] Y. Bazilevs, V. M. Calo, J. A. Cottrell, T. J. R. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197:173–201, 2007.
- [12] Y. Bazilevs, V.M. Calo, T. J. R. Hughes, and Y. Zhang. Isogeometric fluid-structure interaction: theory, algorithms, and computations. *Computational Mechanics*, 43(1):3–37, 2008.
- [13] Y. Bazilevs, J. R. Gohean, T. J. R. Hughes, R. D. Moser, and Y. Zhang. Patient-specific isogeometric fluid-structure interaction analysis of thoracic aortic blood

- flow due to implantation of the Jarvik 2000 left ventricular assist device. *Computer Methods in Applied Mechanics and Engineering*, 198(45-46):3534–3550, 2009.
- [14] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipson, M. A. Scott, and T. W. Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5–8):229–263, 1 2010.
- [15] Y. Bazilevs, C. Michler, V. M. Calo, and T. J. R. Hughes. Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes. *Computer Methods in Applied Mechanics and Engineering*, 199(13-16):780–790, 2010.
- [16] G. Bellettini and A. Coscia. Discrete approximation of a free discontinuity problem. *Numerical Functional Analysis and Optimization*, 15(3-4):201–224, 1994.
- [17] T. Belytschko, H. Chen, J. Xu, and G. Zi. Dynamic crack propagation based on loss of hyperbolicity and a new discontinuous enrichment. *International Journal for Numerical Methods in Engineering*, 58(12):1873–1905, 2003.
- [18] D. J. Benson, Y. Bazilevs, M. C. Hsu, and T. J. R. Hughes. Isogeometric shell analysis: the Reissner-Mindlin shell. *Computer Methods in Applied Mechanics and Engineering*, 199:276–289, 2009.
- [19] D. J. Benson. Stable time step estimation for multi-material Eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering*, 167(1-2):191–205, 1998.

- [20] D. J. Benson, Y. Bazilevs, E. De Luycker, M.-C. Hsu, M. Scott, T. J. R. Hughes, and T. Belytschko. A generalized finite element formulation for arbitrary basis functions: From isogeometric analysis to XFEM. *International Journal for Numerical Methods in Engineering*, 83(6):765–785, 2010.
- [21] D. J. Benson, Y. Bazilevs, M. C. Hsu, and T. J. R. Hughes. A large deformation, rotation-free, isogeometric shell. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1367–1378, 2011.
- [22] M. J. Borden, M. A. Scott, J. A. Evans, and T. J. R. Hughes. Isogeometric finite element data structures based on Bézier extraction of NURBS. *International Journal for Numerical Methods in Engineering*, 87(1-5):15–47, 2011.
- [23] B. Bourdin. Image segmentation with a finite element method. *M2AN Mathematical Modelling and Numerical Analysis*, 33(2):229–244, 1999.
- [24] B. Bourdin. Numerical implementation of the variational formulation for quasi-static brittle fracture. *Interfaces and Free Boundaries*, 9(3):411–430, 2007.
- [25] B. Bourdin, G. A. Francfort, and J-J. Marigo. Numerical experiments in revisited brittle fracture. *Journal of the Mechanics and Physics of Solids*, 48(4):797–826, 2000.
- [26] B. Bourdin, G. A. Francfort, and J. J. Marigo. The variational approach to fracture. *Journal of Elasticity*, 91(1-3):5–148, 2008.
- [27] B. Bourdin, C. Larsen, and C. Richardson. A time-discrete model for dynamic fracture based on crack regularization. *International Journal of Fracture*, 168(2):133–143, 2011.

- [28] A Braides. *Approximation of Free-Discontinuity Problems*. Springer, Berlin, 1998.
- [29] A. Buffa, G. Sangalli, and R. Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1143–1152, 2010.
- [30] S. Burke, C. Ortner, and E. Süli. An adaptive finite element approximation of a variational model of brittle fracture. *SIAM Journal on Numerical Analysis*, 48(3):980–1012, 2010.
- [31] G. T. Camacho and M. Ortiz. Computational modelling of impact damage in brittle materials. *International Journal of Solids and Structures*, 33(20-22):2899 – 2938, 1996.
- [32] A. Chambolle. An approximation result for special functions with bounded deformation. *Journal de Mathématiques Pures et Appliquées*, 83(7):929–954, 2004.
- [33] J. Chung and G. M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-alpha method. *Journal of Applied Mechanics*, 60(2):371–375, 1993.
- [34] E. Cohen, T. Martin, R. M. Kirby, T. Lyche, and R. F. Riesenfeld. Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):334–356, 2010.
- [35] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric analysis: Toward integration of CAD and FEA*. Wiley, 2009.

- [36] J.A. Cottrell, A. Reali, Y. Bazilevs, and T.J.R. Hughes. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195(41-43):5257 – 5296, 2006.
- [37] J.A. Cottrell, T.J.R. Hughes, and A. Reali. Studies of refinement and continuity in isogeometric structural analysis. *Computer Methods in Applied Mechanics and Engineering*, 196:4160–4183, 2007.
- [38] R. de Borst. Damage, material instabilities, and failure. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics Volume 2: Solids and Structures*, pages 335–373. John Wiley & Sons, Ltd., 2004.
- [39] R. de Borst, J. J. C. Remmers, A. Needleman, and M.-A. Abellán. Discrete vs smeared crack models for concrete fracture: bridging the gap. *International Journal for Numerical and Analytical Methods in Geomechanics*, 28(7-8):583–607, 2004.
- [40] G. Del Piero, G. Lancioni, and R. March. A variational model for fracture mechanics: Numerical experiments. *Journal of the Mechanics and Physics of Solids*, 55(12):2513–2537, 2007.
- [41] M. R. Dörfel, B. Jüttler, and B. Simeon. Adaptive isogeometric analysis by local h-refinement with T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5–8):264–275, 2010.
- [42] T. Elguedj, Y. Bazilevs, V. M. Calo, and T. J. R. Hughes. B-bar and F-bar projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements. *Computer Methods in Applied Mechanics and Engineering*, 197:2732–2762, 2008.

- [43] G. A. Francfort and J. J. Marigo. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids*, 46(8):1319–1342, 1998.
- [44] G. A. Francfort, N. Q. Le, and S. Serfaty. Critical points of Ambrosio-Tortorelli converge to critical points of Mumford-Shah in the one-dimensional Dirichlet case. *ESAIM: Control, Optimisation and Calculus of Variations*, 15(3):576–598, 2009.
- [45] M. G. D. Geers, R. de Borst, W. A. M. Brekelmans, and R. H. J. Peerlings. Strain-based transient-gradient damage model for failure analyses. *Computer Methods in Applied Mechanics and Engineering*, 160(1-2):133–153, 1998.
- [46] A. Giacomini. Ambrosio-Tortorelli approximation of quasi-static evolution of brittle fractures. *Calculus of Variations and Partial Differential Equations*, 22(2):129–172, 2005.
- [47] A. Giovanni. Variational models for phase transitions. An approach via  $\Gamma$ -convergence. In G. Buttazzo, A. Marino, and M. K. V. Murthy, editors, *Calculus of variations and partial differential equations. Topics on geometrical evolution problems and degree theory*, pages 95–114. Springer-Verlag, Berlin-Heidelberg, 2000.
- [48] H. Gómez, V. M. Calo, Y. Bazilevs, and T. J. R. Hughes. Isogeometric analysis of the Cahn-Hilliard phase-field model. *Computer Methods in Applied Mechanics and Engineering*, 197(49-50):4333–4352, 2008.

- [49] H. Gómez, T. J. R. Hughes, X. Nogueira, and V. M. Calo. Isogeometric analysis of the isothermal Navier-Stokes-Korteweg equations. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1828–1840, 2010.
- [50] M. E. Gurtin. Generalized Ginzburg-Landau and Cahn-Hilliard equations based on a microforce balance. *Physica D: Nonlinear Phenomena*, 92(3-4):178–192, 1996.
- [51] H. M. Hilber, T. J. R. Hughes, and R. L. Tayler. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5:283–292, 1977.
- [52] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, Mineola, NY, 2000.
- [53] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194:4135–4195, 2005.
- [54] T. J. R. Hughes, A. Reali, and G. Sangalli. Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS. *Computer Methods in Applied Mechanics and Engineering*, 197(49-50):4104–4124, 2008.
- [55] T. J. R. Hughes, A. Reali, and G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):301–313, 2010.

- [56] G. M. Hulbert and J. Chung. Explicit time integration algorithms for structural dynamics with optimal numerical dissipation. *Computer Methods in Applied Mechanics and Engineering*, 137:175–188, 1996.
- [57] J. Kalthoff. Modes of dynamic shear failure in solids. *International Journal of Fracture*, 101(1):1–31, 2000.
- [58] J. F. Kalthoff and S. Winkler. Failure mode transition at high rates of shear loading. In C. Y. Chiem, H. D. Kunze, and L. W. Meyer, editors, *Proceedings of the International Conference on Impact Loading and Dynamic Behavior of Materials*, volume 1, pages 185–195, 1987.
- [59] A. Karma, D. A. Kessler, and H. Levine. Phase-field model of mode III dynamic fracture. *Physical Review Letters*, 87(4):045501, 2001.
- [60] J. Kiendl, Y. Bazilevs, M. C. Hsu, R. Wüchner, and K. U. Bletzinger. The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches. *Computer Methods in Applied Mechanics and Engineering*, 199(37-40):2403–2416, 2010.
- [61] H. J. Kim, Y. D. Seo, and S. K. Youn. Isogeometric analysis for trimmed CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 198(37-40):2982–2995, 2009.
- [62] G. Lancioni and G. Royer-Carfagni. The variational approach to fracture mechanics. a practical application to the French *Panthéon* in Paris. *Journal of Elasticity*, 95(1):1–30, 2009.

- [63] C. J. Larsen. Models for dynamic fracture based on griffith's criterion. In K. Hackl, editor, *IUTAM Symposium on Variational Concepts with Applications to the Mechanics of Materials*, volume 21, pages 131–140. Springer Netherlands, 2010.
- [64] C. J. Larsen, C. Ortner, and E. Süli. Existence of solutions to a regularized model of dynamic fracture. *Mathematical Methods and Models in Applied Sciences*, 20(7):1021–1048, 2010.
- [65] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, and M. A. Scott. On linear independence of T-spline blending functions. *Computer Aided Geometric Design*, 29(1):63–76, 2012.
- [66] S. Lipton, J. A. Evans, Y. Bazilevs, T. Elguedj, and T. J. R. Hughes. Robustness of isogeometric structural discretizations under severe mesh distortion. *Computer Methods in Applied Mechanics and Engineering*, 199:357–373, 2010.
- [67] T. Martin, E. Cohen, and R. M. Kirby. Volumetric parameterization and trivariate B-spline fitting using harmonic functions. *Computer Aided Geometric Design*, 26(6):648–664, 2009.
- [68] C. Miehe, M. Hofacker, and F. Welschinger. A phase field model for rate-independent crack propagation: Robust algorithmic implementation based on operator splits. *Computer Methods in Applied Mechanics and Engineering*, 199(45-48):2765–2778, 2010.
- [69] C. Miehe, F. Welschinger, and M. Hofacker. Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field FE imple-

- mentations. *International Journal for Numerical Methods in Engineering*, 83(10):1273–1311, 2010.
- [70] I. Miranda, R. M. Ferencz, and T. J. R. Hughes. An improved implicit-explicit time integration method for structural dynamics. *Earthquake Engineering and Structural Dynamics*, 18:643–653, 1989.
- [71] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, 1999.
- [72] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.
- [73] A. P. Nagy, M. M. Abdalla, and Z. Gürdal. On the variational formulation of stress constraints in isogeometric design. *Computer Methods in Applied Mechanics and Engineering*, 199(41-44):2687–2696, 2010.
- [74] A. P. Nagy, M. M. Abdalla, and Z. Gürdal. Isogeometric sizing and shape optimisation of beam structures. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1216–1230, 2010.
- [75] R. H. J. Peerlings, R. De Borst, W. A. M. Brekelmans, and J. H. P De Vree. Gradient enhanced damage for quasi-brittle materials. *International Journal for Numerical Methods in Engineering*, 39(19):3391–3403, 1996.
- [76] K. Pham and J. J. Marigo. Construction and analysis of localized responses for

gradient damage models in a 1d setting. *Vietnam Journal of Mechanics*, 31(3-4):233–246, 2009.

- [77] K. Pham, H. Amor, J. J. Marigo, and C. Maurini. Gradient damage models and their use to approximate brittle fracture. *International Journal of Damage Mechanics*, 20(4):618–652, 2011.
- [78] L. Piegl and W. Tiller. *The NURBS Book (Monographs in Visual Communication)*, 2nd ed. Springer-Verlag, New York, 1997.
- [79] X. Qian. Full analytical sensitivities in NURBS based isogeometric shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 199(29–32):2059–2071, 2010.
- [80] K. Ravi-Chandar. Dynamic fracture of nominally brittle materials. *International Journal of Fracture*, 90(1):83–102, 1998.
- [81] K. Ravi Chandar and W. G. Knauss. Dynamic crack-tip stresses under stress wave loading — a comparison of theory and experiment. *International Journal of Fracture*, 20(3):209–222, 1982.
- [82] K. Ravi-Chandar and W. G. Knauss. An experimental investigation into dynamic fracture: III. On steady-state crack propagation and crack branching. *International Journal of Fracture*, 26(2):141–154, 1984.
- [83] K. Ravi-Chandar, J. Lu, B. Yang, and Z. Zhu. Failure mode transitions in polymers under high strain rate loading. *International Journal of Fracture*, 101:33–72, 2000.

- [84] J. J. C. Remmers, R. de Borst, and A. Needleman. A cohesive segments method for the simulation of crack growth. *Computational Mechanics*, 31(1):69–77, 2003.
- [85] J. J. C. Remmers, R. de Borst, and A. Needleman. The simulation of dynamic crack propagation using the cohesive segments method. *Journal of the Mechanics and Physics of Solids*, 56(1):70–92, 2008.
- [86] D. F. Rogers. *An Introduction to NURBS With Historical Perspective*. Academic Press, San Diego, CA, 2001.
- [87] M. A. Scott. *T-splines as a Design-Through-Analysis Technology*. PhD thesis, The University of Texas at Austin, 2011.
- [88] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, and T. J. R. Hughes. Isogeometric finite element data structures based on Bézier extraction of T-splines. *International Journal for Numerical Methods in Engineering*, 88(2):126–156, 2011.
- [89] M. A. Scott, X. Li, T. W. Sederberg, and T. J. R. Hughes. Local refinement of analysis-suitable T-splines. *Computer Methods in Applied Mechanics and Engineering*, 213–216:206–222, 2012.
- [90] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and T-NURCCSs. *ACM Transactions on Graphics*, 22 (3):477–484, 2003.
- [91] T. W. Sederberg, J. Zheng, and X. Song. Knot intervals and multi-degree splines. *Computer Aided Geometric Design*, 20:455–468, 2003.

- [92] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche. T-spline simplification and local refinement. *ACM Transactions on Graphics*, 23(3):276–283, 2004.
- [93] T. W. Sederberg, G. T. Finnigan, X. Li, H. Lin, and H. Ipson. Watertight trimmed NURBS. *ACM Transactions on Graphics*, 27(3):79:1–79:8, 2008.
- [94] J. C. Simo. A framework for finite strain elastoplasticity based on maximum plastic dissipation and the multiplicative decomposition: Part I. continuum formulation. *Computer Methods in Applied Mechanics and Engineering*, 66(2):199 – 219, 1988.
- [95] J. C. Simo and T. J. R. Hughes. *Computational Inelasticity*. Springer-Verlag, New York, 1998.
- [96] S.-H. Song, H. Wang, and T. Belytschko. A comparative study on finite element methods for dynamic fracture. *Computational Mechanics*, 42(2):239–250, 2008.
- [97] G. Strang and G. Fix. *An Analysis of the Finite Element Method*. Wellesly-Cambridge, Wellesly, MA, 2008.
- [98] H. Stumpf and K. Hackl. Micromechanical concept for the analysis of damage evolution in thermo-viscoelastic and quasi-brittle materials. *International Journal of Solids and Structures*, 40(6):1567–1584, 2003.
- [99] C. V. Verhoosel, M. A. Scott, R. de Borst, and T. J. R. Hughes. An isogeometric approach to cohesive zone modeling. *International Journal for Numerical Methods in Engineering*, 87(1-5):336–360, 2010.

- [100] C. V. Verhoosel, M. A. Scott, T. J. R. Hughes, and R. de Borst. An isogeometric analysis approach to gradient damage models. *International Journal for Numerical Methods in Engineering*, 86(1):115–134, 2011.
- [101] C. V. Verhoosel, M. A. Scott, M. J. Borden, T. J. R. Hughes, and R. de Borst. Discretization of higher-order gradient damage models using isogeometric finite elements. In G. Pijaudier-Cabot and F. Dufour, editors, *Damage Mechanics of Cementitious Materials and Structures*, pages 89–120. Wiley-ISTE, 2012.
- [102] W. A. Wall, M. A. Frenzel, and C. Cyron. Isogeometric structural shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 197:2976–2988, 2008.
- [103] W. Wang and Y. Zhang. Wavelets-based NURBS simplification and fairing. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):290–300, 2010.
- [104] W. Wang, Y. Zhang, M. A. Scott, and T. J. R. Hughes. Converting an unstructured quadrilateral mesh to a standard T-spline surface. *Computational Mechanics*, 48:477–498, 2011.
- [105] W. Wang, Y. Zhang, G. Xu, and T. J. R. Hughes. Converting an unstructured quadrilateral/hexahedral mesh to a rational T-spline. *Computational Mechanics*, 50(1):65–84, 2012.
- [106] K. G. Webster. Investigation of close proximity underwater explosion effects on a ship-like structure using the multi-material arbitrary Lagrangian Eulerian finite element method. Master’s thesis, Virginia Polytechnic Institute and State University, January 2007.

- [107] X.-P. Xu and A. Needleman. Numerical simulations of fast crack growth in brittle solids. *Journal of the Mechanics and Physics of Solids*, 42(9):1397 – 1434, 1994.
- [108] Y. Zhang, Y. Bazilevs, S. Goswami, C. Bajaj, and T. J. R. Hughes. Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow. *Computer Methods in Applied Mechanics and Engineering*, 196:2943–2959, 2007.
- [109] Y. Zhang, W. Wang, and T. J. R. Hughes. Solid T-spline construction from boundary representations for genus-zero geometry. *Computer Methods in Applied Mechanics and Engineering*, In press: doi: 10.1016/j.cma.2012.01.014, 2012.