Purdue University
Purdue e-Pubs

Open Access Dissertations | Theses and Dissertations

Fall 2014

Algebraic level sets for CAD/CAE integration and moving boundary problems

Kritika Upreti
Purdue University

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_dissertations

 Part of the Computer Sciences Commons, and the Mechanical Engineering Commons

Recommended Citation
Upreti, Kritika, "Algebraic level sets for CAD/CAE integration and moving boundary problems" (2014). *Open Access Dissertations*. 376.
https://docs.lib.purdue.edu/open_access_dissertations/376

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Kritika Upreti

Entitled
Algebraic Level Sets for CAD/CAE Integration and Moving Boundary Problems

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Ganesh Subbarayan

Anil Bajaj

Marisol Koslowski

Ananth Grama

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Ganesh Subbarayan

Approved by Major Professor(s): _____

_____

Approved by: Ganesh Subbarayan 10/16/2014

Head of the Graduate Program Date

ALGEBRAIC LEVEL SETS FOR CAD/CAE INTEGRATION AND MOVING BOUNDARY PROBLEMS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Kritika Upreti

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2014

Purdue University

West Lafayette, Indiana

To my parents, sister and husband

for their love, support and patience and most of all for believing in my choices

ACKNOWLEDGMENTS

First of all, I am grateful to my advisor, Professor Ganesh Subbarayan, for giving me the opportunity to pursue my Doctoral studies at Purdue University. During the past five years, he not only guided me towards my technical goals, but also mentored me to become successful in all aspects of my professional life. He taught me how to become an effective teacher, mentor, presenter, writer, colleague and a better individual. I am extremely thankful to him for his support at challenging times.

I would also like to thank my committee members Professor Anil Bajaj, Professor David Anderson, Professor Marisol Koslowski, Professor Ananth Grama and Professor Christoph Hoffmann for their time and valuable suggestions. I am also thankful to all the professors who taught me during my stay at Purdue. It was a privilege to be taught by one of the best in their respective fields.

I am grateful to my internship manager Dr. Ravi Shenoy at Qualcomm for being a wonderful mentor and sharing his industry experiences and knowledge that helped me shape my career goals. I would also like to thank my internship managers Dr. Xiaobai Li and Dr. Edward Knudsen at Bosch research who gave me the opportunity to work with some of the best researchers in industry.

I would like to acknowledge the support of Semiconductor Research Corporation (SRC) for part of the thesis work. I am thankful to SRC for this support. I also extend my gratitude to the SRC project co-investigators: Professor Bahgat Sammakia and Dr. Dae Young Jung for their insightful suggestions and help over the course of the project.

The past five years were made easier by colleagues and friends at Purdue. I would like to thank all members of our lab with whom I interacted during the course of my PhD. I would like to thank lab-alumni - Kaushik, Nikhil and Simranjeet for their help and support during my initial days in the lab. I am thankful to lab-alumni - Wole

for his guidance and readiness to help, Dennis for his sense of humor, Chaitra for her friendship, cheerful smile and demeanor and Hung-Yun for being an inspiration. I would also like to thank our lab-alumni Abhishek who was a mentor to me after Professor. His guidance and our technical discussions were very helpful in shaping my thesis. I am thankful to my current labmates - Subramanya and Yuvraj for their friendship and support, Anirudh for interesting discussions, Tao for our collaboration. I am also thankful to lab-alumni Allen, Mesut and Phillip.

I would like to express my gratitude to all friends at Purdue who were my support system and family away from home; Mitro, Rangha, Shashank, Aarti, Susmita, Suchi, Romi, Neha to name a few. I am greatly thankful to my parents and sister for supporting my five year long journey, believing in me and encouraging me during this time. I am the most grateful to my husband, Manu, who is my pillar of strength, my best friend, my shrink and my constant source of inspiration and motivation.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

ABSTRACT

Upreti, Kritika. Ph.D., Purdue University, December 2014. Algebraic Level Sets for CAD/CAE Integration and Moving Boundary Problems. Major Professor: Ganesh Subbarayan, School of Mechanical Engineering.

Boundary representation (B-rep) of CAD models obtained from solid modeling kernels are commonly used in design, and analysis applications outside the CAD systems. Boolean operations between interacting B-rep CAD models as well as analysis of such multi-body systems are fundamental operations on B-rep geometries in CAD/CAE applications. However, the boundary representation of B-rep solids is, in general, not a suitable representation for analysis operations which lead to CAD/CAE integration challenges due to the need for conversion from B-rep to volumetric approximations. The major challenges include intermediate mesh generation step, capturing CAD features and associated behavior exactly and recurring point containment queries for point classification as inside/outside the solid. Thus, an ideal analysis technique for CAD/CAE integration that can enable direct analysis operations on B-rep CAD models while overcoming the associated challenges is desirable.

Further, numerical surface intersection operations are typically necessary for boolean operations on B-rep geometries during the CAD and CAE phases. However, for non-linear geometries, surface intersection operations are non-trivial and face the challenge of simultaneously satisfying the three goals of accuracy, efficiency and robustness. In the class of problems involving multi-body interactions, often an implicit knowledge of the boolean operation is sufficient and explicit intersection computation may not be needed. Such implicit boolean operations can be performed by point containment queries on B-rep CAD models. However, for complex non-linear B-rep geometries, the point containment queries may involve numerical iterative point projection op-

erations which are expensive. Thus, there is a need for inexpensive, non-iterative techniques to enable such implicit boolean operations on B-rep geometries.

Moreover, in analysis problems with evolving boundaries (or *moving boundary problems*), interfaces or cracks, blending functions are used to enrich the underlying domain with the known behavior on the enriching entity. The blending functions are typically dependent on the distance from the evolving boundaries. For boundaries defined by free form curves or surfaces, the distance fields have to be constructed numerically. This may require either a polytope approximation to the boundary and/or an iterative solution to determine the exact distance to the boundary.

In this work a purely algebraic, and computationally efficient technique is described for constructing signed distance measures from Non-Uniform Rational B-Splines (NURBS) boundaries that retain the geometric exactness of the boundaries while eliminating the need for iterative and non-robust distance calculation. The proposed technique exploits the NURBS geometry and algebraic tools of *implicitization*. Such a signed distance measure, also referred to as the *Algebraic Level Sets*, gives a volumetric representation of the B-rep geometry constructed by purely non-iterative algebraic operations on the geometry. This in turn enables both the implicit boolean operations and analysis operations on B-rep geometries in CAD/CAE applications. Algebraic level sets ensure exactness of geometry while eliminating iterative numerical computations. Further, a geometry-based analysis technique that relies on hierarchical partition of unity field compositions (HPFC) theory [1] and its extension to enriched field modeling [2] is presented. The proposed technique enables direct analysis of complex physical problems without meshing, thus, integrating CAD and CAE. The developed techniques are demonstrated by constructing algebraic level sets for complex geometries, geometry-based analysis of B-rep CAD models and a variety of fracture examples culminating in the analysis of steady state heat conduction in a solid with arbitrary shaped three-dimensional cracks.

The proposed techniques are lastly applied to investigate the risk of fracture in the ultra low-k (ULK) dies due to copper (Cu) wirebonding process. Maximum damage

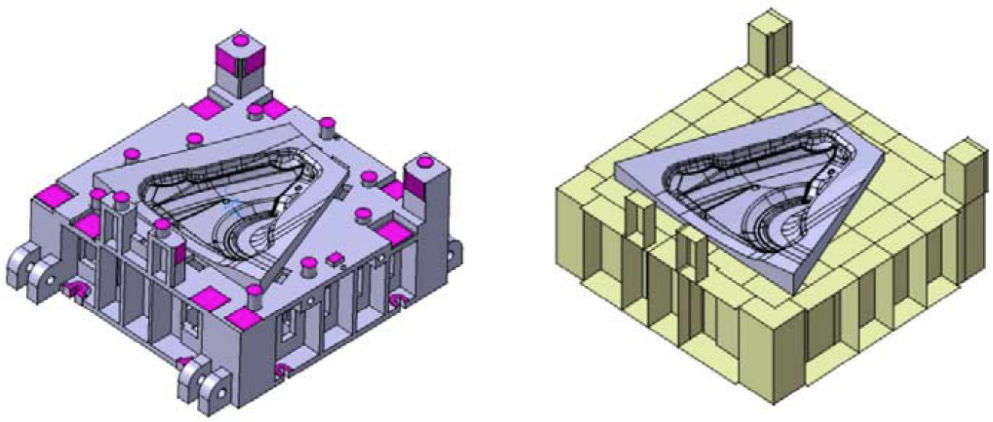
induced in the interlayer dielectric (ILD) stack during the process steps is proposed as an indicator of the reliability risk. Numerical techniques based on enriched isogeometric approximations are adopted to model damage in the ULK stacks using a cohesive damage description. A damage analysis procedure is proposed to conduct damage accumulation studies during Cu wirebonding process. Analysis is carried out to identify weak interfaces and potential sites for crack nucleation as well as damage nucleation patterns. Further, the critical process condition is identified by analyzing the damage induced during the impact and ultrasonic excitation stages. Also, representative ILD stack designs with varying Cu percentage are compared for risk of fracture.

1. INTRODUCTION

Computer Aided Design (CAD) and Computer Aided Engineering (CAE) are two integral phases of Product Design. CAD systems enable design synthesis, which includes a model of the geometrical shape and its visualization in a computer based environment. CAE systems on the other hand analyze the synthesized design to check if it satisfies the design requirements. This is achieved by solving boundary/initial value problems defined over the geometric domain represented by the CAD model. Hence, both CAD and CAE phases are executed in a sequence multiple times to iteratively improve the design. For such an execution to be efficient, it is important to integrate the CAD and CAE phases seamlessly.

In the basic design-analysis sequence, the product geometry is first generated using CAD tools and then, the physical behavior of the product is predicted using the analysis tools. A number of integration challenges arise in this basic sequence. Firstly, the developed CAD models are complex with detailed geometric features (Fig. 1.1(a)) and may be unsuitable for analysis using the current commercial tools (Finite Element Analysis, FEA, being the most popular choice). Since, the mathematical representations of geometry in CAD and CAE systems are often distinct, the ability to integrate CAD and CAE is greatly aided by a common mathematical representation as well as construction procedures that mirror each other.

The simplification or defeaturing of the CAD models (Fig. 1.1(b)) to make them suitable for analysis is a tricky problem. An important goal of the design-analysis sequence is to capture the behavior of the product correctly in addition to documenting the geometrical state of design. The physical model may depend on known or specified behavior such as boundaries with specified boundary conditions (essential or natural), crack surfaces, multi-material interfaces or singular points. Both the geometry of the boundaries and known behavior need to be accurately captured. But, at



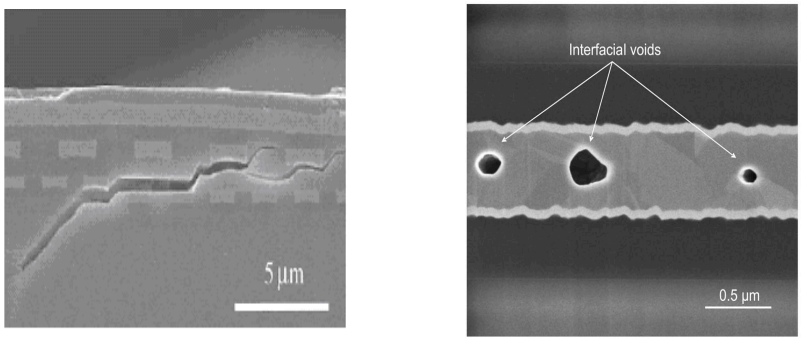
(a) (b)

Figure 1.1. Geometric models in CAD and CAE systems. (a) Detailed design model from CAD systems, (b) Abstracted analysis model (adapted from [3]).

the same time the computational expense increases with increase in the complexity of the analysis model. Hence, a number of factors determine the efficiency of CAD and CAE integration such as the scale, scope and purpose of the CAE analysis, the nature and dimensionality of the CAD model, and the amount of detail required for the CAE application [3]. These challenges are further aggravated by the iterative nature of the design problem such as in shape/topology optimization problems or physics driven moving boundary problems such as crack propagation and void evolution (Fig. 1.2).

The problem of automatically converting the B-rep CAD model into a finite element mesh is a widely studied research area [6,7]. Even if such an integration may be automated (i.e., even if complex part geometry is automatically meshed), the analysis model as represented by the mesh does not preserve the original geometry, which is critical in physical problems where computable normals and tangents are critical. Further, the mesh needs to be reconstructed when geometry is evolved for any reason.

Another important challenge to CAD/CAE integration is posed by the geometric operation of *Surface-Surface Intersection*. Although often not recognized as such,



(a) (b)

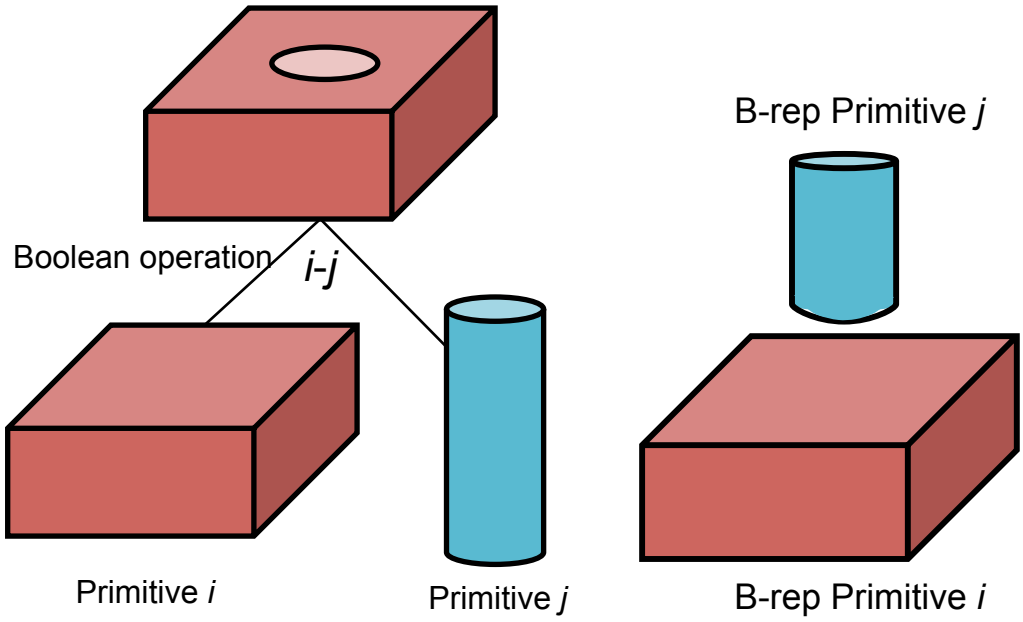
Figure 1.2. Moving boundary problems. (a) Cracks in semiconductor chip dielectric stacks (adapted from [4]), (b) Electromigration driven void growth (adapted from [5]).

surface intersection is a fundamental and recurring operation during CAD *as well as* CAE phases beginning with geometric modeling, followed by mesh generation and, finally, during the analysis process (see Fig. 1.3). A good surface intersection algorithm should satisfy the three goals of *accuracy*, *robustness* and *efficiency* [8–11]. However, this problem has been an area of research for more than three decades and yet simultaneous advancement of all the three goals remains a challenge [8].

A combined CAD/CAE modeling strategy which is able to overcome these challenges will enable tighter integration in the design process.

1.1 Survey of Computational Techniques for CAD/CAE Integration

A survey of computational techniques that attempt to integrate design and analysis is presented and the associated challenges are discussed. The computational schemes for design and analysis can be compared on the basis of multiple criteria. A comparison of known mesh-based and mesh-free approaches to CAD/CAE integration, focusing on basic computational tasks that support complete CAD-CAE integration can be found in [12]. Further, a compilation of geometry-based isopara-



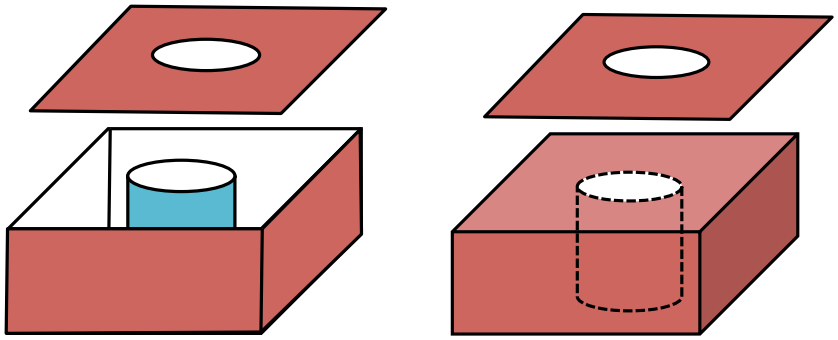
(a) (b)

Figure 1.3. Surface-surface intersection operations in CAD/CAE. (a) Boolean operations in CAD geometry construction, (b) Interference detection between interacting primitives in contact problem.

metric approaches towards CAD/CAE integration is presented in [13]. In this section, the existing computational modeling schemes are compared with respect to the challenges identified in the previous section. In the following discussion, the techniques are classified based on the CAD geometric representations (see Fig. 1.4) and the corresponding approximation spaces in the analysis domain. The classification is summarized in Fig. 1.5.

1.1.1 Boundary Representation

In this approach solid objects are modeled using their boundaries comprising of faces, edges and vertices. The geometries obtained from commercial CAD systems



(a) (b)

Figure 1.4. Representation of CAD geometry constructed in Fig. 1.3(a). (a) Boundary representation, (b) Volumetric representation.

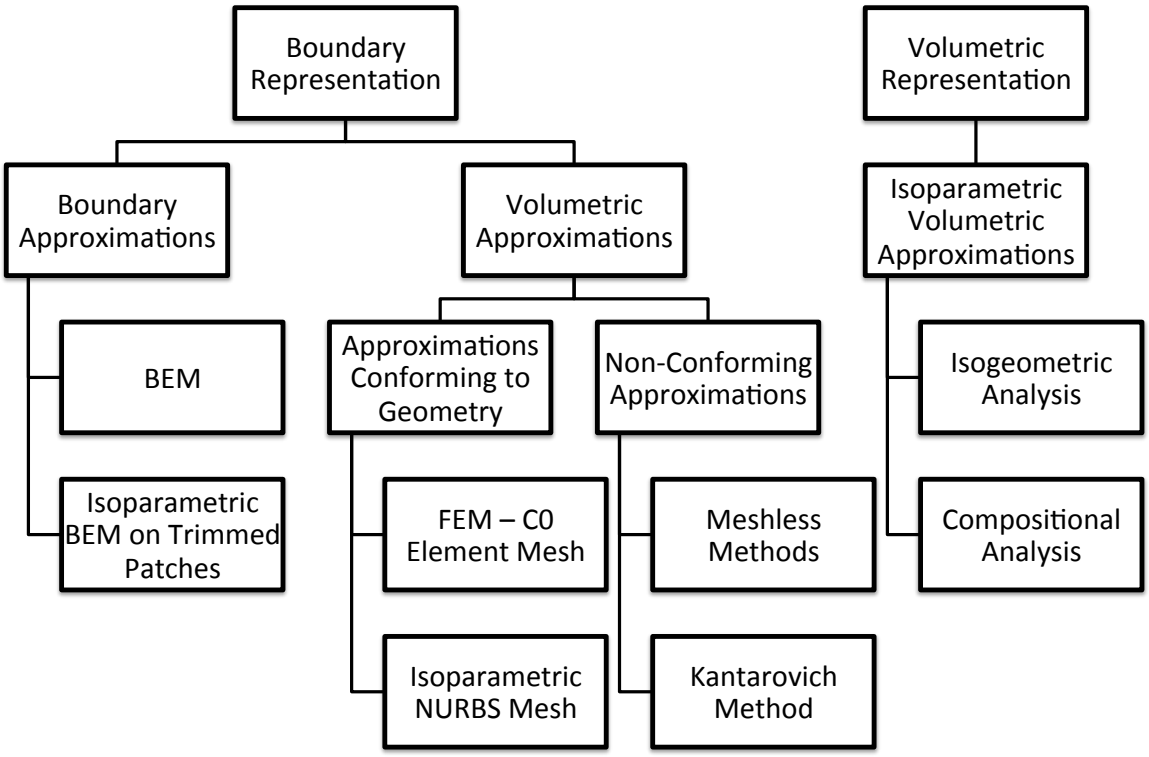
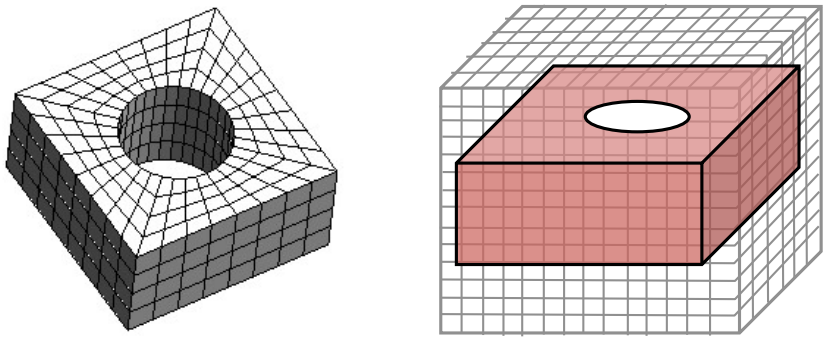


Figure 1.5. Classification of computational techniques for CAD/CAE integration.

are in general boundary representations. This representation does not parameterize the interior of the solid, hence no volumetric information is associated with it.

Volumetric Approximations

In this class of methods, volumetric approximations of solid objects are built from their B-rep CAD models for analysis applications. The volumetric approximations are further classified based on their construction (see Fig. 1.6).



(a) (b)

Figure 1.6. Volumetric approximations. (a) Approximations conforming to geometry, (b) Non-conforming approximations.

Approximations Conforming to Geometry In this class of methods, the approximation space of analysis domain need to conform to the geometry of CAD model. The construction of such geometry conforming approximation is achieved by an intermediate mesh generation step.

Traditional Finite Element Mesh. The Finite Element Method (FEM) is the most popular technique among the ones which use volumetric approximations for analysis. A finite element mesh approximates the volumetric domain of the CAD model by approximately conforming to its boundary [14,15]. The efficiency of design-analysis integration in this case depends on the efficiency of the automated mesh

generation schemes. While the problem of automated mesh generation has seen great advances [7, 16], there are still a number of unsolved issues related to robustness [17]. Analysis of complicated geometries and detailed features lead to a large number of degrees of freedom since FE approximation to CAD model is accurate only in the limit of mesh refinement. Also, the analysis model, which is only an approximation to the CAD model, may lose important geometric features influencing the behavior of the design. Further, remeshing at every step of geometric evolution is a challenge with FEM. These challenges motivate the research for alternate modeling techniques.

Isoparameteric NURBS Representation. Ideally, design-analysis integration should be done at the level of mathematical representation of geometry and analysis. This implies that the functions that represent geometry of the CAD model should also represent the analysis geometry and field approximations. The CAD geometry may be represented by parametric representations such as Bezier, B-spline and NURBS. Kagan [18] developed a B-spline based finite element scheme. Another such development was the B-spline finite element method by Sabin [19]. Renken and Subbarayan [20] used NURBS to represent the shape of droplets as well as to calculate the solid-liquid and liquid-vapor interaction energies at the surfaces. Recently, NURBS based design and analysis is termed as Isogeometric analysis [21] and has become a growing research area. However, the isogeometric approximations are volumetric in nature and require construction of meshes from the boundary representations of geometry. The isogeometric "meshes" have the advantage of eliminating all geometric representation errors. However, conversion from the boundary representation to a volumetric representation is still a mesh generation problem and will suffer from similar limitations.

Non-Conforming Approximations In this class of methods, the approximation space of analysis domain need not conform to the geometry of CAD model. The approximations can be constructed on a background uniform grid or in other similar manner while the geometric model is implicitly used to evaluate the queries needed for analysis, for example, point containment and distance calculation. This eliminates the

problems associated with conforming meshes to geometry or remeshing during iterations.

Meshless Methods. Smooth particle hydrodynamics (SPH) [22, 23], the hp-cloud method [24] and many others, construct approximations such that they cover the domain by supports of radial basis functions which decay inversely to the distance from center of the support. A detailed review of such techniques is presented in [25]. These methods eliminate the problems associated with mesh based techniques, but complicate the enforcement of boundary conditions and volumetric integration.

Kantarovich Method. Another class of methods in this category construct approximations on uniform grids. These methods use basis functions which satisfy prescribed boundary conditions and are also easy to integrate over the volumetric domain. Kantarovich [26] proposed the construction of such basis functions from any other basis function by using a sufficiently smooth function that vanishes at the dirichlet boundary. Rvachev [27] generalized Kantarovich's idea of applying homogeneous boundary conditions to any and all types of boundary value problems using the concept of R-function method (RFM) solution structure. Other extensions to this method are presented in [28–30]. Höllig proposed the method of weighted extended B-splines (WEB-splines) to construct approximations on the uniform grid in order to improve stability and convergence of the method [31]. Recently, authors in [32] have proposed a query-based approach towards CAD and CAE integration. While this class of methods looks promising when compared to the other techniques, it is important to note that the B-rep geometry is implicitly modeled in the analysis domain. Hence, exactness of geometry during analysis of behavior is not ensured.

Boundary Approximations

In this class of methods, boundary approximations of solid objects are built from their B-rep CAD models for analysis applications.

Boundary Element Method In the boundary element method (BEM) [33], approximations are built only over the boundary of the domain. Hence, this method has considerable advantage over the traditional FEM with respect to the number of degrees of freedom.

However, BEM suffers from certain intrinsic limitations. It produces fully populated matrices which makes the method computationally expensive and has computational advantage over FEM only in the problems with small surface/volume ratio. Non-linearities can only be handled by volume integrals, hence BEM is inefficient in such problems. BEM is applicable to problems for which Green's functions can be calculated. This places considerable restrictions on the generality of its applications.

Isoparameteric Boundary Element Method Isoparametric boundary approximations lead to design-analysis integration at the level of mathematical representation of geometry and analysis in the true sense. Casale [34, 35] proposed the integration of geometric modeling and structural analysis by using BEM on trimmed patches on the boundary. Developments in this area followed after advent of isogeometric analysis [36, 37].

The isoparameteric boundary approximations overcome the challenge of mesh generation associated with isoparametric volumetric approximations and analysis is done directly on the geometry. However, it still suffers from the limitations of BEM discussed in previous section resulting in limited applications of the technique.

1.1.2 Volumetric Representation

With the integration at mathematical level, the class of methods using isoparametric approximations to geometry theoretically overcome the issues of mesh generation and regeneration as analysis is performed directly on the geometry. The challenge however is that most of the CAD models are boundary representations while isogeometric analysis requires volumetric NURBS approximations. Thus, there is a need to develop volumetric representations for CAD models that parameterize the bound-

ary as well as interior domain of solids. It is proposed that new ab-initio design methods that support analysis aware representations for CAD models need to be developed [38]. Hence, there is an increasing interest in trivariate solid construction motivated by isogeometric analysis that is exemplified by references [39–41]. However, the research in this area is still at a nascent stage and the most important challenge is that current B-rep CAD modeling systems need to be completely replaced with new CAD modeling kernels that enable trivariate CAD geometry construction.

Isoparametric Volumetric Approximations

The isoparametric volumetric approximation is same as the isoparameteric NURBS representation described in the previous sections. The basic difference in this is that a volumetric representation of geometry such as trivariate B-spline or NURBS parameterization is assumed. Hence, in this case, there is no need for conversion of boundary representation to volumetric representation. This class of methods achieves CAD/CAE integration at the mathematical representation level in the true sense. Further, the intrinsic challenges associated with BEM are also overcome due to the volumetric nature of this approximation.

Compositional Analysis A further step towards CAD/CAE integration was proposed by authors in [42]. They proposed CAD/CAE integration both at mathematical as well as procedural level by mirroring the constructive solid geometry (CSG) procedure for defining the analysis problem as well as using NURBS to represent both geometry and analysis functions. This technique was further extended to develop the HPFC theory [1]. The compositional procedure mirroring CSG procedure is illustrated in Fig. 1.7.

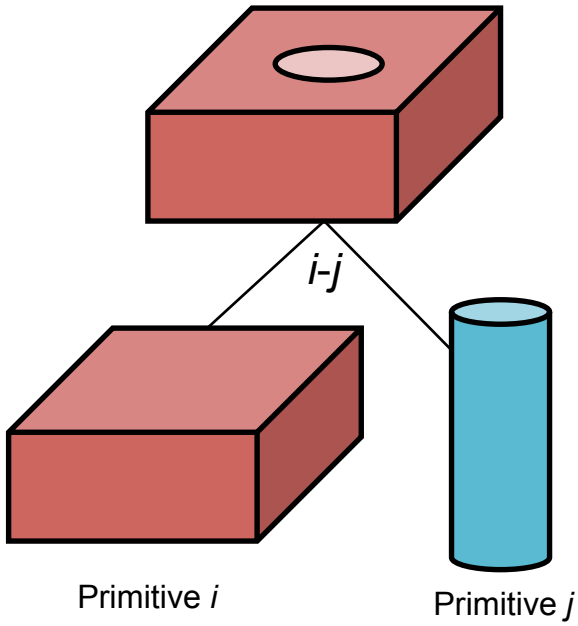


Figure 1.7. Compositional isoparametric approximations mirroring CSG procedure.

1.2 Core Challenges to CAD/CAE Integration

The core challenges to the computational techniques for CAD/CAE integration are summarized in this section. The computational techniques discussed in the previous section address one or more of these challenges but not all.

1.2.1 Mesh Generation Problem

There are two main factors that drive the need for an intermediate mesh generation step between the design and analysis processes.

Need for Volumetric Information. In general, all analysis techniques except for the boundary element method are volumetric in nature and need conversion of B-rep CAD model into an analysis amenable volumetric representation through mesh generation.

Difference in the Mathematical Representation. This is applicable to all approximations (volumetric or boundary) that are non-isoparametric in nature to the CAD geometry.

The mesh generation problem is theoretically eliminated by the analysis approximations non-conforming to the geometry and the isogeometric approximations.

1.2.2 Exactness of Geometry

The CAD geometry may have certain geometric features which have associated behavior that is important for the analysis problem. For example, boundaries with specified field values, singular points or crack surfaces, multi-material interfaces. These geometric features must be captured exactly to model its behavior accurately. However, in mesh generation schemes, often the CAD geometries are simplified or defeatured, leading to inaccurately capturing potentially important geometric features. Further, in non-conforming approximations, the geometry is captured implicitly. Hence, the behavior of such geometric features is captured only in the limit of refinement. The exactness of geometric features with relevant behavior is solved only by isoparametric boundary or volumetric approximations that explicitly capture the exact geometry.

1.2.3 Point Containment Problem

A fundamental problem associated with bounded solids is to determine if a query point is contained within the approximation of the solid. This problem is trivial if volumetric approximation of the solid is used for analysis since it has a parameteric representation for the interior of the solid. However, for analysis approximations distinct from the CAD geometry, only the boundary representation of the geometry is available. Hence, numerical techniques that may involve point projection on to the surface of the solid are needed to check for point containment.

1.2.4 Surface Intersection Problem

Another fundamental problem in CAD/CAE integration is the boolean operations occurring in multibody problems such as contact analysis and machining simulations. In general, for both boundary as well as volumetric approximations of geometry, boolean operations involve surface intersection computations. In existing CAD systems, B-rep CAD models are represented by a collection of trimmed spline patches such as NURBS. Boolean operations on such a representation involves numerical surface-surface intersection operation between the trimmed NURBS patches. The problem of surface intersection between two parametric spline patches leads to an underconstrained system of non-linear equations which is typically solved using numerical methods. Surface intersection solution techniques are broadly classified into four categories: algebraic [43, 44], subdivision [11, 45–48], lattice [49] and marching methods [50–52]. Hybrid methods have also been developed recently combining features of the above methods. In general, in the surface intersection techniques, it is challenging to *simultaneously* achieve the goals of *accuracy*, *robustness* and *efficiency* [8–11]. Greater accuracy in general is associated with greater data proliferation and computational cost. Numerical inaccuracies may lead to detection of wrong topology of solution. Further, the intersection techniques need to detect all curve branches, loops and singularities to ensure robustness. However, robustness is dependent on heuristic numerical parameter choices such as step size in marching methods or number of subdivision steps. Most algorithms use conservative step size that makes them slow. Hence, it is very difficult to satisfy the conflicting goals of accuracy, robustness and efficiency simultaneously using the existing methods.

The problem of CAD/CAE integration faces the challenges of mesh generation problem, capturing geometric features exactly, point containment problem and surface intersection problem. The mesh generation problem is eliminated by building non-conforming approximations. However, such approximations have difficulty in

Table 1.1. Comparison of CAD/CAE integration techniques.

Geoemtric Representation	**Boundary**		**Volumetric**		**Volumetric**	
Analysis Approximations	**Boundary**		**Volumetric**		**Isoparametric Volumetric**	
	BEM	Isoparame-tric BEM	FEA/IGA Mesh	Meshless	IGA	HPFC
Mesh Generation Problem	×	✓	×	✓	✓	✓
Exactness of Geometry	×	✓	×	×	✓	✓
Point Containment Problem	✓	✓	✓	×	✓	✓
Surface Intersection Problem	×	×	×	×	×	×

capturing the geometric features exactly as well as the numerical challenge of point containment. On the other hand, isogeometric approximations can theoretically remove the mesh generation problem as well as handle exactness of geometry and point containment. But the major challenge with this class of analysis techniques is the need for trivariate analysis ready CAD models. The existing CAD systems cannot provide such a representation, hence, such geometries need to be constructed from scratch and new CAD modeling tools need to be developed. Further, none of the current techniques address the numerical challenge of surface intersection for implicit boolean operations. These challenges are summarized in Table 1.1.

1.3 Summary of Research Gaps

An ideal computational technique to overcome the CAD/CAE integration challenges will possess the following characteristics:

1. Enable direct analysis of complex physical problems without meshing
2. Utilize and ensure exactness of geometry
3. Eliminate surface intersection problem
4. Eliminate point containment problem

Towards this end, a purely algebraic solution to the problem of geometrical and behavioral modeling is proposed here. The algebraic tools of *implicitization* will be utilized to construct a signed distance measure, named as *Algebraic Level Sets*, for the parametric geometry. In the proposed technique, there will be no numerical computation of intersection curve and all boolean compositions in CAD and CAE will be addressed algebraically using the algebraic level sets. This technique will utilize the HPFC theory for direct analysis of geometric models [1] and isogeometric enriched field approximations [2] for analysis of solids with enriching entities such as cracks, voids, or inclusions.

1.4 Research Overview

The main goal of this work is to develop a CAD/CAE integration technique that will (1) enable direct analysis of B-rep CAD models without meshing, (2) ensure exactness of geometry to capture known behavior on boundaries, (3) address surface intersection and point containment problems through algebraic non-iterative solution. To this end, *Algebraic Level Sets*, an algebraic measure of signed distance from NURBS boundaries is proposed. The proposed technique exploits the NURBS geometry and algebraic tools of *implicitization* to construct the signed distance measure

for the NURBS solid. Such a signed distance measure gives a volumetric representation of the B-rep geometry constructed by purely non-iterative algebraic operations on the geometry. Further, it also ensures exactness of the geometry in the analysis procedure while eliminating iterative numerical computations. A geometry-based analysis technique that relies on HPFC theory [1] and its extension to enriched field modeling [2] is presented. The proposed technique enables direct analysis of complex physical problems without meshing, thus, integrating CAD and CAE. In this technique approximations on underlying domain are composed with the approximations on explicitly modeled B-rep CAD boundaries with known behavior. The blending of the approximations on the B-rep geometry and underlying domain is achieved by using weight functions that satisfy the partition of unity property [53]. These weight functions are constructed using the proposed algebraic distance measure from boundaries. Further, all implicit boolean operations and point containment queries during analysis are performed by querying the sign of the algebraic level set of B-rep solids. There is no numerical computation of exact intersection curve, hence, eliminating numerical surface intersection problem. The developed techniques are demonstrated by constructing algebraic level sets for complex geometries, geometry-based analysis of B-rep CAD models and a variety of fracture examples culminating in the analysis of steady state heat conduction in a solid with arbitrary shaped three-dimensional cracks. Finally, the proposed techniques are applied to a microelectronics packaging problem to investigate the risk of fracture in the ULK dies due to Cu wirebonding process.

1.5 Outline

In Chapter 2, the theoretical concepts essential for development of algebraic level sets are presented. In Chapter 3, detailed algorithms for constructing unsigned algebraic distance measure for NURBS and Bezier geometries are discussed. The properties of the algebraic distance field are examined to determine the quality of its ap-

proximation to exact distance in Chapter 4. In Chapter 5, algorithms for constructing signed distance measures for bounded B-rep solids are discussed. The algebraic sign calculation is a by-product of the algebraic distance field calculation technique. Next, the algebraic sign calculation algorithms are applied to solve the problem of implicit boolean operations in Chapter 6. Further, in Chapter 7, a computational technique for CAD/CAE integration based on the HPFC theory and the enriched field modeling is presented. This technique enables direct analysis of complex physical problems without meshing, thus, integrating CAD and CAE. In Chapter 8, the application of algebraic distance field for modeling fracture, damage and crack propagation are demonstrated using the geometrically explicit and behaviorally implicit enrichment scheme [2]. Further, in Chapter 9, application of algebraic level sets is demonstrated in a wirebonding problem encountered in the semiconductor industry. A comprehensive study of risk of fracture in ULK dies during the Cu wirebonding process is presented using simulations of damage and fracture in layered ULK structures. Finally, in Chapter 10, the thesis is summarized listing the novel contributions and proposed future work.

2. ALGEBRAIC LEVEL SETS : THEORY

In general, in moving boundary problems, the motion of complex boundaries need to be tracked within the domain. Examples of such problems arise in many fields, including fluid mechanics, solid mechanics, optimal design, computer vision and image processing. Commonly, an Eulerian framework in which the geometry of the boundaries is inferred implicitly as the zero level set of an evolving field is used to numerically solve these class of problems [54]. Since the level sets implicitize both the geometry of the boundary as well as the distance from the boundary, in other computational procedures for moving boundary problems, there is a common need for explicitly calculated distance fields.

In general, in these problems, distance from the boundary or interface serves as a measure of influence of the behavior on the boundary at a point in the underlying domain. Computational procedures relying on distance fields are many. For example, the use of distance fields for image processing is well established [55]. In fluid-structure interaction problems, signed distance functions have been used to represent fluid boundaries such as the fluid-structure interface or the free surface within the computational domain [56, 57]. Distance fields are also useful in contact problems [58–60] for defining the gap function for contact detection. Recently, in the Isogeometric Analysis (IGA, [20, 21, 42]) literature, an approximate distance field has been used to enrich the base approximations with those on lower-dimensional geometrical features, enabling application of boundary conditions as well as simulations of crack propagation [2]. Mathematical representation of graded materials have also used distance fields to describe the desired blended material distributions [61, 62]. Thus, while the use of distance field as a measure of influence of the boundaries on the domain has enabled the numerical solution to complex problems, inexpensive distance field calculations are essential for the analysis to be computationally efficient.

Since the notion of distance fields is fundamental to many computational solution procedures, there exists a significant established literature aimed at improving the techniques for calculating the distance field. Iterative methods such as the Newton-Raphson technique are usually required for finding continuously varying distance from spatial points to a parametric surface [63–66]. While the iterative schemes make distance computations expensive, the calculated distance field is unfortunately not sufficiently smooth for many engineering applications. This is since the distance function is not differentiable at points that are equidistant to two or more points on the surface (cut locus of the surface).

Iterative distance calculation methods in general become necessary due to the geometrical nonlinearity of the surface. Hence, piecewise planar approximations of a parametric surface using, for instance, triangular mesh are popular in computer graphics applications (see for instance [67]). The piecewise linearization of geometry makes the distance calculation relatively straightforward, albeit by introducing a new combinatorial problem – that of determining the nearest planar surface from among the many possible ones at a spatial point. Therefore, the algorithms relating to piecewise planar approximations mainly deal with efficient data structures to determine the triangular face closest to the point of interest (see for instance, [68, 69]) or algorithmically propagating distances from calculations closer to the surface [70, 71]. The disadvantage of the piecewise planar approximations is that the geometrical exactness of the boundary is not preserved making the calculated distance accurate only in the limit of refinement of the planar approximations. But, more critically, the calculated distance field is non-smooth since a subset of the spatial points will project to an edge or vertex of the triangular face. This in turn implies that one cannot rely on distance calculations on planar approximations to generate smooth, continuous distance measures.

In general, distance measures may be thought of as extending approximations of fields from the boundaries into the underlying domain. For constructing such a distance measure, the exact distance to the boundary is not as critical as a smooth and

monotonically increasing function of distance that serves as a measure of influence of the boundary on the underlying domain. Smoothness of such an approximate distance measure would ensure robustness of any calculations based on the constructed field. Hence, an approximate distance measure that is smooth is sufficient for these applications. Towards building such an approximate measure, Biswas and Shapiro [72] described a method using piecewise planar geometric approximations of a parametric curve. They constructed an approximate distance field for each linear segment and combined the distance field of each linear segment using R-functions into a global approximate distance field. The advantage of this technique over the previous distance calculation techniques is that this enables desired smoothness in the distance field. However, this technique relies on piecewise linearization of the geometry and hence, compromises the exactness of the boundary.

In this work, a purely algebraic, and therefore computationally efficient, technique for constructing an approximate distance function that avoids the iterative, and therefore inefficient, distance computations is developed. The technique preserves the geometric exactness of low-degree (two or three) NURBS surfaces. Such purely algebraic distance computation procedures that preserve the geometric exactness do not appear to exist in prior literature. The proposed technique overcomes the need for the iterative exact distance computations at every quadrature point during analysis while providing smoother, more robust distance field relative to the numerically computed distance.

In this chapter, the theoretical concepts essential for development of the algebraic distance field are presented. Rational Bezier and NURBS parametrizations are considered as they are most popular in Computer aided geometric design (CAGD) and Isogeometric Analysis.

2.1 Distance Field: Definition and Properties

Given a set $S \in \mathbb{R}^n$, for any point $x \in \mathbb{R}^n$ a distance field $d(x)$ of S is defined as the minimum distance from the point x to an element $p \in S$ given by

$$d_S(x) = \inf_{p \in S} \parallel x - p \parallel \tag{2.1}$$

The distance function d is continuous at all $x \in \mathbb{R}^n$ and is differentiable almost everywhere in $\mathbb{R}^n$. The points where the field is not differentiable are those where minimum distance corresponds to non-unique points $p \in S$. An important property associated with the gradient of the field d is

$$\parallel \nabla d_S(x) \parallel = 1 \tag{2.2}$$

Thus, by definition, the *distance field* refers to exact distance to the given geometry. In this paper, the same terminology is followed, and any approximation to the exact distance field is referred to as *approximate distance field*, *distance measure*, *algebraic level sets*, or *algebraic distance field* depending on the context.

2.2 Requirements on Distance Field for Explicit Modeling of Evolving Boundaries

As a motivation for identifying the requirements on distance measures so they may be useful for analysis, various types of boundaries and interfaces that occur in engineering design and analysis problems are summarized in Fig. 2.1. Ideally, the distance measure would preserve the geometric exactness of the explicit boundary so that the behavior on these complex boundaries is accurately captured by the analysis scheme. Hence, a fundamental requirement that drives the current work is that the distance field needs to be exact on and very near the boundaries. At the same time, it is also required that the distance field is sufficiently smooth for modeling the analysis problem. Thus, the following requirements on distance field are imposed in this study:

1. Exact on the boundary

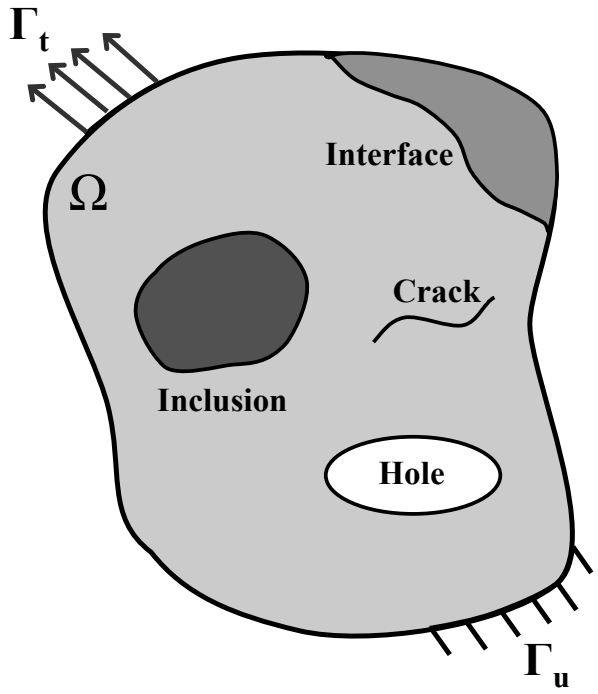


Figure 2.1. Illustration of the different types of boundaries and interfaces encountered during engineering analysis that may be modeled as enrichments on an underlying field.

2. Montonic function of exact distance

3. Sufficiently smooth for engineering applications within the domain

4. Efficiently obtained through non-iterative calculations

2.3 Implicitization of Parametric Geometry using Elimination Theory

An implicit representation of a geometry is an equation of the form $f(x_1, ..., x_n) = 0$ in n-dimensional space. Given the implicit representation $f(x_1, ..., x_n) = 0$, a level set of the function f is of the form $\mathcal{L}(f) = \{(x_1, ..., x_n) \,|\, f(x_1, ..., x_n) = c\}$, that is a set for which the function f takes on some constant value c. Such level sets of a geometry give monotonic measure of distance from the geometry as shown in Fig.

2.2. Hence, by obtaining an implicit representation of the parametric geometry, it is possible to construct a utilizable distance measure from the geometry.

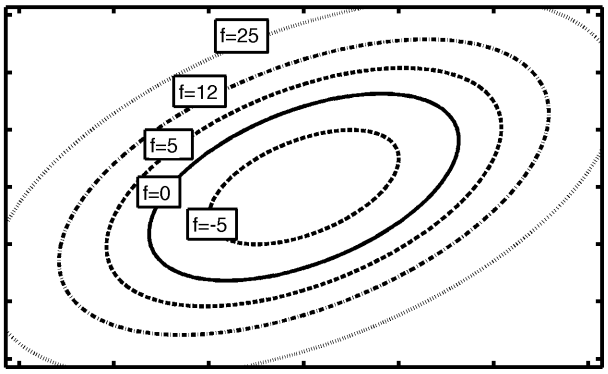


Figure 2.2. Level curves of an ellipse with implicit equation $f(x, y) = x^2 + y^2 - xy - 8$.

Algebraic techniques based on elimination theory [73, 74] enable the conversion of parametric geometry to its implicit representation $f(x_1, ..., x_n) = 0$. Elimination theory investigates the conditions under which sets of polynomials have common roots. For instance, a linear system of equations $Ax = 0$ will have a non-trivial solution if and only if the determinant of the coefficient matrix vanishes, i.e., $|A| = 0$. An expression of this form $|A|$ involving the coefficients of the polynomials is a *Resultant*. The vanishing of this resultant is a necessary and sufficient condition for the set of polynomials to have a common non-trivial root. Implicitization tools are based on construction of such resultants.

2.3.1 Implicitization of Parametric Curves

Sederberg [73] discussed in detail the various resultants that may be used for implicitization. In this work, Bezout's resultant is used for implicitization of parametric curves as it is efficient to compute compared to the other resultants. We briefly illustrate Bezout's resultant with an example but, prior to that, we observe that for

a rational parametric curve $C(X(t), Y(t), W(t))$ or $C(x, y)$ with $x = \frac{X(t)}{W(t)}, y = \frac{Y(t)}{W(t)}$, two auxiliary polynomials may be formed as:

$$\begin{aligned} q_1(x,t) &= W(t)x - X(t) = 0 \\ q_2(y,t) &= W(t)y - Y(t) = 0 \end{aligned} \tag{2.3}$$

The resultant of the above polynomials (treating x and y as constants) yields the implicit curve corresponding to the parametric entity. Now, specifically, consider the following two polynomials

$$\begin{aligned} q_1(t) &= a_3t^3 + a_2t^2 + a_1t + a_0 \\ q_2(t) &= b_3t^3 + b_2t^2 + b_1t + b_0 \end{aligned} \tag{2.4}$$

where, from the form of Equation (2.3), it is clear that a_i and b_j are linear functions of x and y respectively. By the following algebraic manipulations, three equations are obtained in t^2, t and 1.

$$\begin{aligned} &a_3q_2 - b_3q_1 = (a_3b_2)t^2 + (a_3b_1)t + (a_3b_0) = 0 \\ &(a_3t + a_2)q_2 - (b_3t + b_2)q_1 = (a_3b_1)t^2 + [(a_3b_0) + (a_2b_1)]t + (a_2b_0) = 0 \\ &(a_3t^2 + a_2t + a_1)q_2 - (b_3t^2 + b_2t + b_1)q_1 = (a_3b_0)t^2 + (a_2b_0)t + (a_1b_0) = 0 \end{aligned} \tag{2.5}$$

In matrix form, this system of equations can be written as $A\hat{x} = 0$, or

$$\begin{bmatrix} (a_3b_2) & (a_3b_1) & (a_3b_0) \\ (a_3b_1) & (a_3b_0) + (a_2b_1) & (a_2b_0) \\ (a_3b_0) & (a_2b_0) & (a_1b_0) \end{bmatrix} \begin{Bmatrix} t^2 \\ t \\ 1 \end{Bmatrix} = 0 \tag{2.6}$$

Then, Bezout's resultant of the two polynomials $q_1(t)$ and $q_2(t)$ is $|A|$ by definition.

In general, Bezout's resultant of two degree r polynomials is determinant of a $r \times r$ symmetric matrix:

$$\begin{bmatrix} (a_rb_{r-1}) & \cdots & (a_rb_0) \\ \vdots & & \vdots \\ (a_rb_0) & \cdots & (a_1b_0) \end{bmatrix} \begin{Bmatrix} t^{r-1} \\ \vdots \\ 1 \end{Bmatrix} = 0 \tag{2.7}$$

where $(a_ib_j) = a_ib_j - a_jb_i$, a and b are linear functions of x and y respectively. For the values of t for which $q_1(t) = 0$ and $q_2(t) = 0$, there exists points $P(x, y)$ lying on

the parametric curve $C(X(t), Y(t), W(t))$ such that $f(x, y) = 0$. This is the implicit equation of the parametric curve C. Fig. 2.3 illustrates implicitized curves of different degrees.

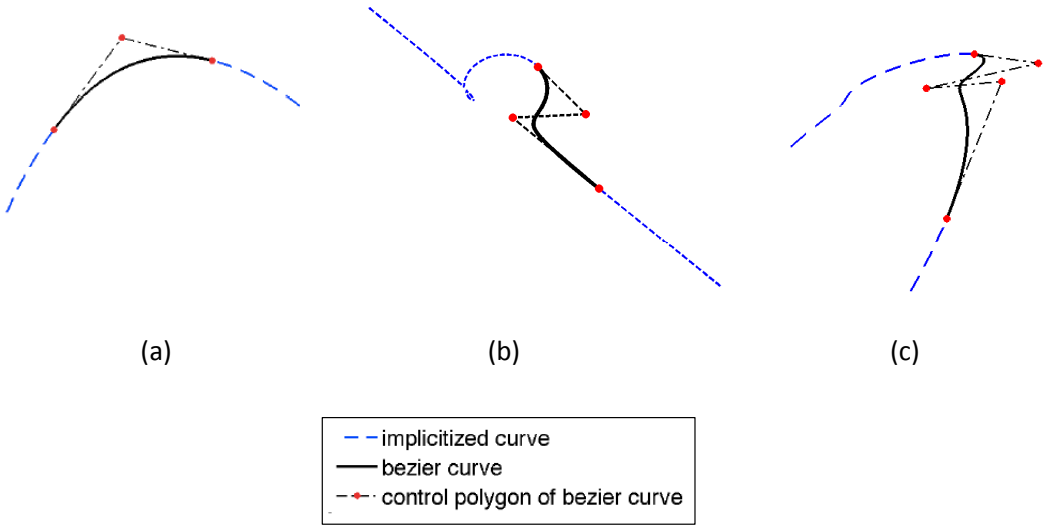


Figure 2.3. Implicitization examples. (a) Degree 2 Bezier curve with homogeneous control points (-3,-2,0,1), (0,2,0,1), (4,1,0,1), (b) Degree 3 Bezier curve with homogeneous control points (4,-3,0,1), (-1,2,0,4), (5,1,0,1.5), (2,6,0,2), (c) Degree 4 Bezier curve with homogeneous control points (0,-3,0,1), (4,4,0,2), (-3,7,0,4), (5,4,0,1.5), (2,6,0,2).

2.3.2 Implicitization of Parametric Surfaces

Implicitization of parametric surfaces is carried out using Dixon's resultant [73]. Given a rational parametric surface $S(X(u, v), Y(u, v), Z(u, v), W(u, v))$ of degree $r_1 \times r_2$, three auxiliary polynomials are formed as:

$$\begin{aligned} q_1(x, u, v) &= W(u, v)x - X(u, v) = 0 \\ q_2(y, u, v) &= W(u, v)y - Y(u, v) = 0 \\ q_3(z, u, v) &= W(u, v)z - Z(u, v) = 0 \end{aligned} \tag{2.8}$$

As before, holding x, y and z to be constants, if some (u', v') simultaneously satisfy $q_1 = q_2 = q_3 = 0$, then for $(u, v) = (u', v')$ and any (α, β), the following determinant always vanishes [74]:

$$Det(u, v, \alpha, \beta) = \begin{vmatrix} q_1(u, v) & q_2(u, v) & q_3(u, v) \\ q_1(\alpha, v) & q_2(\alpha, v) & q_3(\alpha, v) \\ q_1(\alpha, \beta) & q_2(\alpha, \beta) & q_3(\alpha, \beta) \end{vmatrix} \tag{2.9}$$

Also, the determinant vanishes for $u = \alpha$ or $v = \beta$. Hence, $(u - \alpha)$ and $(v - \beta)$ are factors of the determinant. Now, these factors may be eliminated to obtain the following determinant:

$$\delta(u, v, \alpha, \beta) = \frac{Det(u, v, \alpha, \beta)}{(u - \alpha)(v - \beta)} = \sum_{i=0}^{2r_1-1} \sum_{j=0}^{r_2-1} Q_{i,j}(u, v)\alpha^i \beta^j \tag{2.10}$$

The polynomial δ is further expressed as a polynomial in α and β whose coefficients $Q_{i,j}$ are polynomials in u and v. The polynomial δ vanishes for $(u, v) = (u', v')$ with any choice of (α, β), which is possible if and only if every $Q_{i,j}(u', v') = 0$. This generates $2r_1r_2$ polynomials and gives rise to the following system of equations [73]:

$$\begin{array}{c} \alpha^0\beta^0 \\ \vdots \\ \alpha^i\beta^j \\ \cdots \\ \alpha^{2r_1-1}\beta^{r_2-1} \end{array} \begin{bmatrix} A(0,0,0,0) & \cdots & A(0,0,k,l) & \cdots & A(0,0,r_1-1,2r_2-1) \\ \vdots & & \vdots & & \vdots \\ A(i,j,0,0) & \cdots & A(i,j,k,l) & \cdots & A(i,j,r_1-1,2r_2-1) \\ \vdots & & \vdots & & \vdots \\ A(2r_1-1,r_2-1,0,0) & \cdots & A(2r_1-1,r_2-1,k,l) & \cdots & A(2r_1-1,r_2-1,r_1-1,2r_2-1) \end{bmatrix} \begin{Bmatrix} u^0v^0 \\ \vdots \\ u^kv^l \\ \cdots \\ u^{r_1-1}v^{2r_2-1} \end{Bmatrix} = 0 \tag{2.11}$$

where terms $A(i, j, k, l)$ are linear functions of x, y and z. The determinant of the above $2r_1r_2 \times 2r_1r_2$ matrix is Dixon's resultant.

2.4 Challenges to Implicit Representation

The implicit equation obtained using Bezout's resultant gives distance measure (level sets) from the implicit representation. Fig. 2.4 shows the level curves of the implicit representation of a parametric cubic curve. These level curves form an approximate distance field.

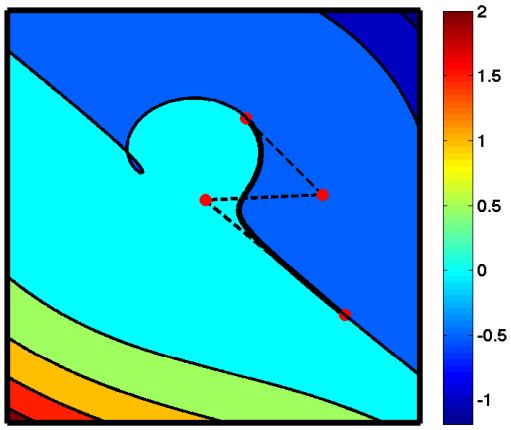


Figure 2.4. Level set as distance measure. Level curves of implicit representation of a degree-3 Bezier curve.

However, the implicit representation of a parametric curve is global, meaning that it extends beyond the parametric range of the original curve. The distance measure formed by the level sets of the implicit representation essentially corresponds to the shortest distance from the implicit curve. But the shortest distance to implicit curve may not always be the shortest distance to the parametric curve. This is illustrated in Fig. 2.5. The parametric curve is a segment of the circle and the implicit curve is the circle (Fig. 2.5(a)). The implicitized distance function is plotted against exact distance from the Bezier curve in Fig. 2.5(d). Implicit distance function is not monotonic with respect to exact distance from the parametric curve. Hence, the implicitized distance field does not qualify as a valid distance measure for the purposes of analysis.

2.5 Boolean Operations using R-functions

In order to obtain a valid distance measure, it is essential to obtain an implicit or function representation of the parametric geometry that is defined only

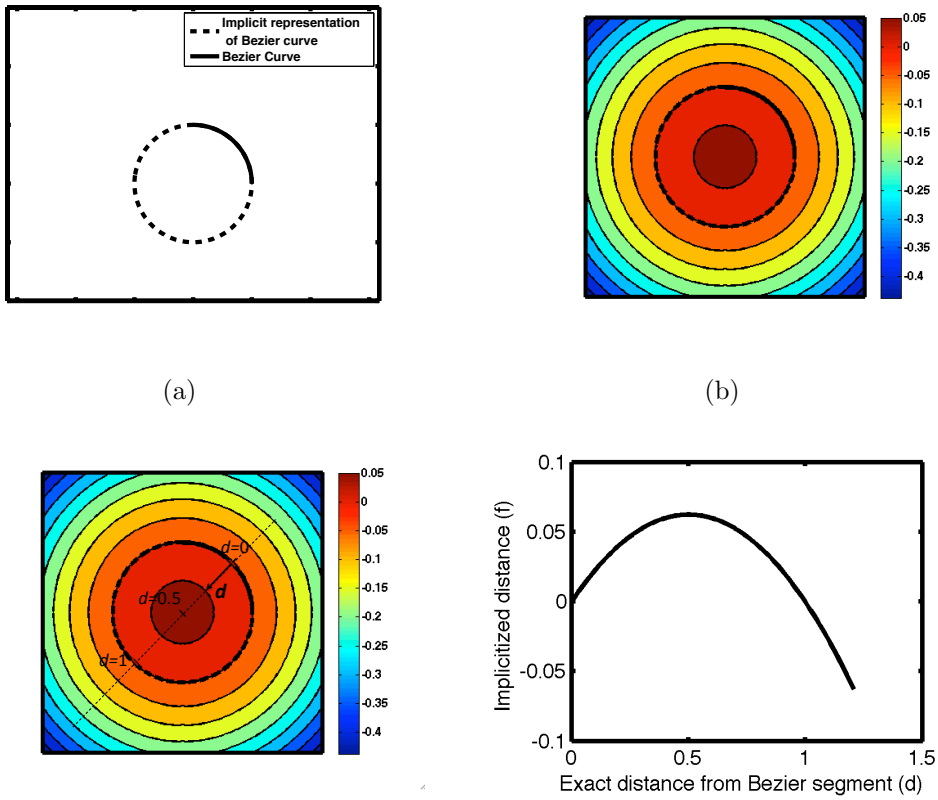


Figure 2.5. Implicitized distance function. (a) An example of a Bezier curve and the corresponding implicit representation, (b) Level curves of implicit representation as distance field, (c) Axis with markings of exact distance from the Bezier curve, (d) Plot of implicitized distance field with respect to exact distance from Bezier curve.

within the parametric range of the geometry. For instance, given a rational curve $C(X(t), Y(t), W(t))$ with the parametric range $t \in [t_0, t_n]$, the implicit representation must be of the form $\mathcal{F}(X(t), Y(t)) = 0 \ \forall \ t \in [t_0, t_n]$. The level curves of such an implicit function ($\mathcal{F}(X(t), Y(t)) = c$) will provide a valid distance measure for the given parametric curve.

In this work, an algebraic distance measure is proposed based on the following trimming procedure [75]. A *convex* trimming region $\phi \geq 0$ is defined such that its

intersection with the implicit curve $f = 0$ is the parametric curve segment $C(t)$ as shown in Fig. 2.6. The algebraic representation of this trimmed parametric curve segment $C(t)$ is obtained by boolean operations on the fields of trimming region ϕ and implicit representation f. R-functions [75] are chosen here to enable boolean operations on functional description of domains.

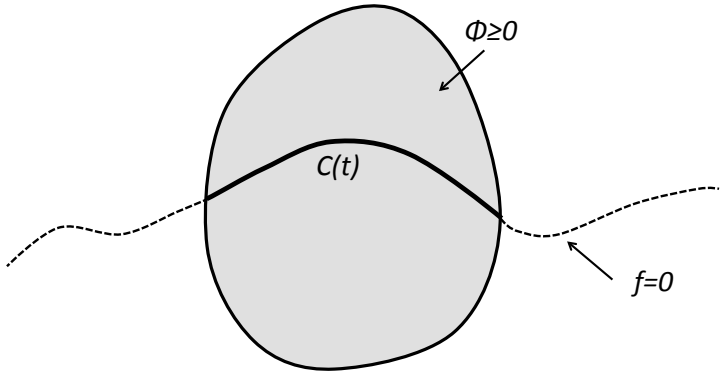


Figure 2.6. Parametric curve $C(t)$ as an intersection of region $\phi \geq 0$ and implicitized curve $f = 0$.

The theory of R-functions [75] provides smooth functional equivalents of boolean operations and is therefore appropriate in an algebraic procedure. Some basic R-functions and their companion boolean operations are as follows

1. R-conjunction is the functional equivalent to boolean intersection operation. The R-conjunction function value essentially corresponds to the minimum of functions g_1 and g_2 such that the resulting function has desired smoothness.

$$g_1 \wedge g_2 = g_1 + g_2 - \sqrt{g_1^2 + g_2^2} \tag{2.12}$$

2. R-disjunction is the functional equivalent to boolean union operation. The R-disjunction function corresponds to the maximum of functions g_1 and g_2 with desired smoothness.

$$g_1 \vee g_2 = g_1 + g_2 + \sqrt{g_1^2 + g_2^2} \tag{2.13}$$

In this work, the R-function for the trimming procedure is adopted from reference [72]. This R-function as proposed by Rvachev [76] is of the form

$$g = \sqrt{f_1^2 + f_2^2} \tag{2.14}$$

where, f_1 is a distance measure to the infinite curve and f_2 is a distance measure to the trimming region such that f_2 is identically zero in the interior of the trimming region. Then, the function g defines the distance measure from any point to the trimmed curve.

Let $f(x, y)$ be the implicit representation for the given parametric boundary and ϕ be the algebraic representation of the convex trimming region. Then, functions f_1 and f_2 in Equation (2.14) are constructed as follows.

$$f_1 = f$$

$$f_2 = \frac{(|\phi| - \phi)}{2}$$

Substituting the functions f_1 and f_2 constructed from the function representation of implicit curve and trimming region respectively, a valid distance field $g(d)$ is constructed by the following R-function.

$$g = \sqrt{f^2 + \frac{(|\phi| - \phi)^2}{4}} \tag{2.15}$$

This trimming operation enables construction of a valid approximate distance field $g(x, y)$ for a parametric curve with implicit representation $f(x, y) = 0$. Inside region ϕ, the distance field corresponds to the implicit representation such that

$$g = f \quad \forall\, x \in \phi \geq 0 \tag{2.16}$$

For points outside the trimming region ϕ, distance field is obtained as a composition of the fields ϕ and f.

$$g = \sqrt{f^2 + \phi^2} \quad \forall\, x \in \phi < 0 \tag{2.17}$$

2.6 Normalization and Composition of Algebraic Distance Fields

The parametric representation of a spline geometry is a piecewise polynomial representation. Hence, the algebraic distance measure of the spline is constructed through union of the algebraic distance fields of the constituent parametric geometries. However, boolean composition of algebraic distance fields may result in a non-monotonic distance measure as illustrated in Fig. 2.7(a). This is because the gradient of algebraic distance measure may not always satisfy the property, $\| \nabla d \| = 1$. Hence, the adjoining distance fields of piecewise polynomial curve may grow at different rates and exhibit non-monotonic behavior when composed. Also, an algebraic function that gives an approximate measure of distance from a boundary could be locally converted into a proper distance function by scaling. For both of these reasons, normalization is of importance when constructing distance fields. The normalization process is described below.

Let $d(x_1, ..., x_n)$ be an exact distance function from a boundary and $s(x_1, ..., x_n)$ be an approximate distance field to the boundary. Then, s is a function of the exact distance d from the boundary. Let $\mathbf{n}$ be the variable in the normal direction at the boundary. Taylor series expansion of $s(d)$ with respect to d near the boundary can be written as

$$s(d) = s(0) + \frac{\partial s}{\partial \mathbf{n}}(0)\, d + \frac{1}{2!}\frac{\partial^2 s}{\partial \mathbf{n^2}}(0)\, d^2 + \frac{1}{3!}\frac{\partial^3 s}{\partial \mathbf{n^3}}(0)\, d^3 + ... \tag{2.18}$$

Ignoring the higher order terms in the expansion, $s(d)$ can be assumed to be a linear approximation locally such that scaling the function by $\frac{\partial s}{\partial \mathbf{n}}(0)$ gives a first order normalized distance function.

$$\hat{s}(d) = \frac{s(d)}{\frac{\partial s}{\partial \mathbf{n}}(0)} \tag{2.19}$$

Higher order normalization can be obtained by assuming a better approximation of the function $s(d)$ [19]. In general, the first order normalized distance fields possess the properties that make them suitable for analysis [72]. Thus, normalization enables the construction of smooth boolean compositions of the algebraic distance fields. The extent of smooth blending depends on the order of normalization (Fig. 2.7(b)).

R-function operations such as R-conjunction are used to compose the normalized algebraic distance fields into a single algebraic distance measure. R-function operations preserve normalization of the constituent fields almost everywhere. Consider two normalized functions $\hat{s}_1$ and $\hat{s}_2$ and let $\hat{s}$ be obtained as an R-conjunction composition of $\hat{s}_1$ and $\hat{s}_2$. Then,

$$\hat{s} = \hat{s}_1 + \hat{s}_2 - \sqrt{\hat{s}_1^2 + \hat{s}_2^2}$$

$$\frac{\partial \hat{s}}{\partial x} = \frac{\partial \hat{s}_1}{\partial x} + \frac{\partial \hat{s}_2}{\partial x} - \frac{\hat{s}_1 \frac{\partial \hat{s}_1}{\partial x} + \hat{s}_2 \frac{\partial \hat{s}_2}{\partial x}}{\sqrt{\hat{s}_1^2 + \hat{s}_2^2}} \quad , \quad \frac{\partial \hat{s}}{\partial y} = \frac{\partial \hat{s}_1}{\partial y} + \frac{\partial \hat{s}_2}{\partial y} - \frac{\hat{s}_1 \frac{\partial \hat{s}_1}{\partial y} + \hat{s}_2 \frac{\partial \hat{s}_2}{\partial y}}{\sqrt{\hat{s}_1^2 + \hat{s}_2^2}}$$

It can be seen that when $\hat{s}_1 = 0$ and $\hat{s}_2 \neq 0$, $\frac{\partial \hat{s}}{\partial x} = \frac{\partial \hat{s}_1}{\partial x}$ and $\frac{\partial \hat{s}}{\partial y} = \frac{\partial \hat{s}_1}{\partial y}$. Thus,

$$|\nabla \hat{s}|^2 = |\nabla \hat{s}_1|^2 = 1$$

Similarly, $|\nabla \hat{s}|^2 = 1$ when $\hat{s}_1 \neq 0$ and $\hat{s}_2 = 0$. Thus, R-conjunction preserves normalization near the boundary except when $\hat{s}_1 = 0$ and $\hat{s}_2 = 0$. At such points, the gradient is bounded and the bounds are derived in Section 4.2 as

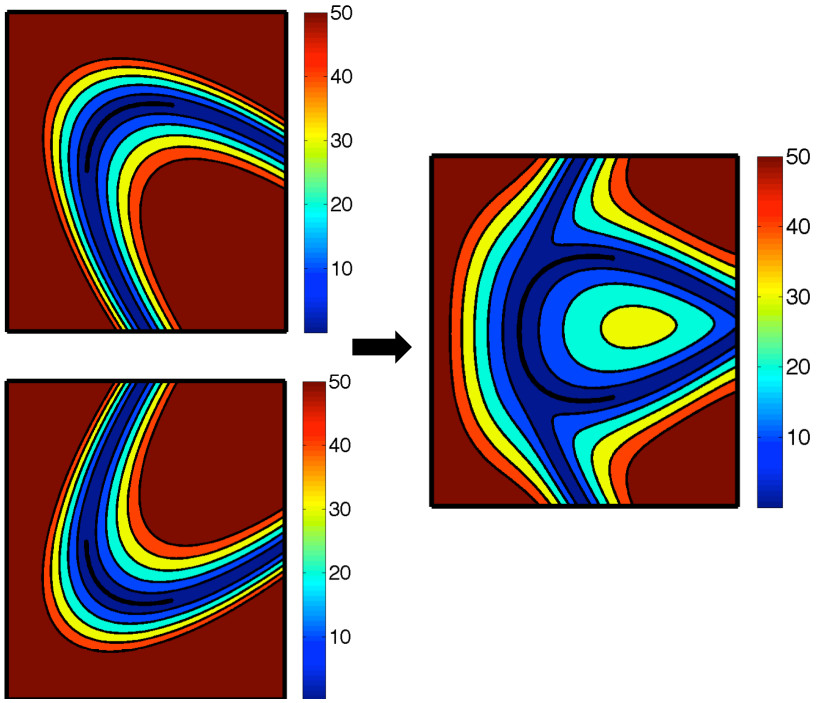
$$|\frac{m-1}{\sqrt{1+m^2}}| \leq |\nabla \hat{s}| \leq |2 - \frac{m+1}{\sqrt{1+m^2}}| \tag{2.20}$$

where

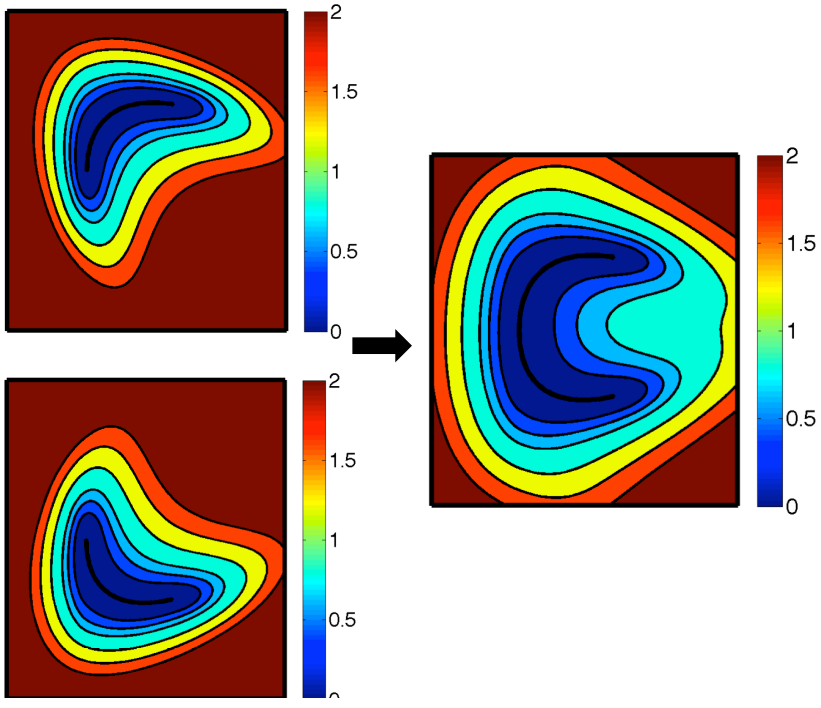
$$m = \lim_{\hat{s}_1 \to 0} \frac{\hat{s}_2}{\hat{s}_1}$$

The maximum value of the upper bound is 3 and the minimum value of lower bound is 0 (Section 4.2).

Hence, once the constituent algebraic fields are normalized, the normalization is carried forward in further boolean compositions except at points where the parametric curves intersect. However, the gradient is bounded at such points.



(a)



(b)

Figure 2.7. Normalization of distance field. (a) The composition of non-normalized distance fields can lead to a non-monotonic approximate distance field, (b) First order normalization yields a monotonic distance field.

3. UNSIGNED DISTANCE FIELD CONSTRUCTION

Based on the theory described in the previous chapter, detailed algorithms are proposed for constructing algebraic distance fields from NURBS and Bezier curves and surfaces.

3.1 Algebraic Distance Field Construction Algorithm for a NURBS Curve

The flowchart in Fig. 3.1 illustrates the steps for the algebraic distance computation from a NURBS curve. These steps are explained in detail below. The algorithm performs an initial check for repeated control points of the NURBS curve, in which case the NURBS curve reduces to a degenerate point. In such a case the distance is computed to the degenerate point and the execution of the algorithm is stopped. For a non-degenerate case, the algorithm proceeds as described below.

3.1.1 Bezier Decomposition

Since NURBS geometries are piecewise parametric representations, they are not suitable for direct implicitization. Therefore, the NURBS curve is first decomposed into its Bezier segments. Given a NURBS curve of degree p, the Bezier segments can be obtained by inserting additional knots at interior knot locations until the multiplicity becomes p. Established subdivision techniques using knot insertion described in [64, 77, 78] were used. The decomposition process is illustrated in Fig. 3.2 using a degree two NURBS curve.

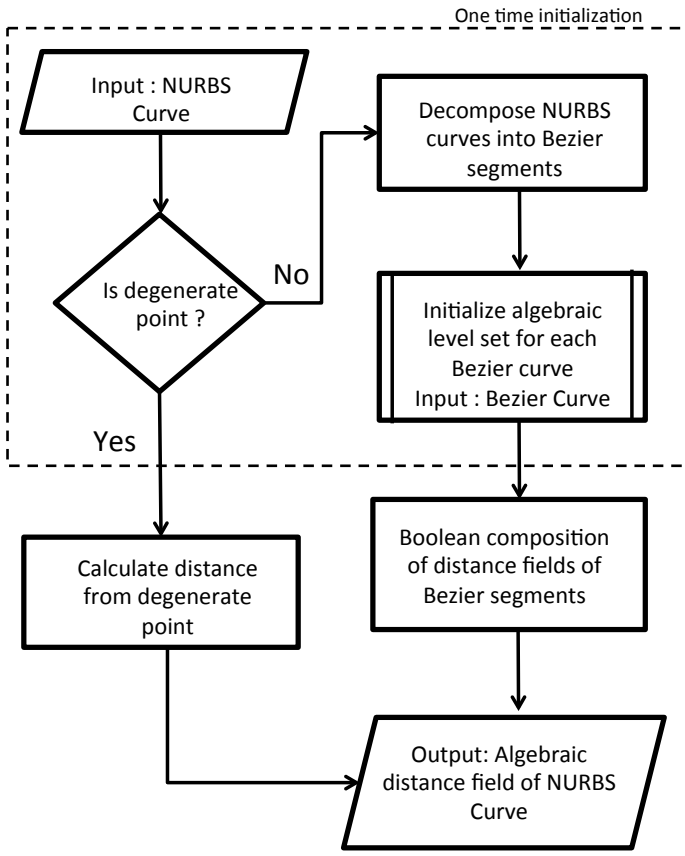
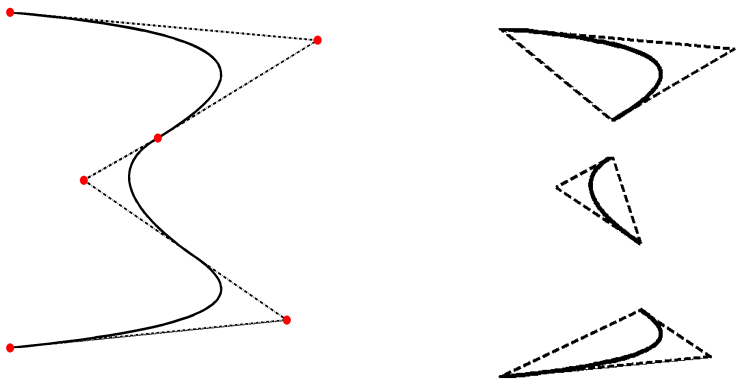


Figure 3.1. Flowchart describing the algebraic distance field construction for a NURBS curve.

3.1.2 Algebraic Distance Field Construction for Bezier Curve

After Bezier decomposition, the algebraic distance field is constructed for each Bezier segment. The flow of control for the construction of distance fields from a Bezier curve is shown in Fig. 3.3. This algorithm also begins with a check for the case of repeated control points that reduces the Bezier curve to a degenerate point. In such a case distance between the given point and the degenerate point is



(a) (b)

Figure 3.2. Decomposition of a NURBS curve into its Bezier segments. (a) Degree two NURBS curve with control polygon, (b) Bezier curves with respective convex hulls.

computed and the algorithm is terminated. For a non-degenerate case, the convex hull is constructed for the Bezier curve using its control points.

Normalized Distance Field of Convex Hull

The algebraic distance field is based on the trimming procedure described in Section 2.5. The trimming procedure requires the following condition to be met - a convex trimming region $\phi \geq 0$ (Fig. 2.6) that encloses the parametric curve segment such that the end points of the curve segment lie on the trimming boundary. In this work, the convex hull of the Bezier segments are proposed as the trimming region since the convex hull satisfies the requirement on convexity and the condition on curve endpoints. The convex hull is also straightforward to compute from the Bezier control points.

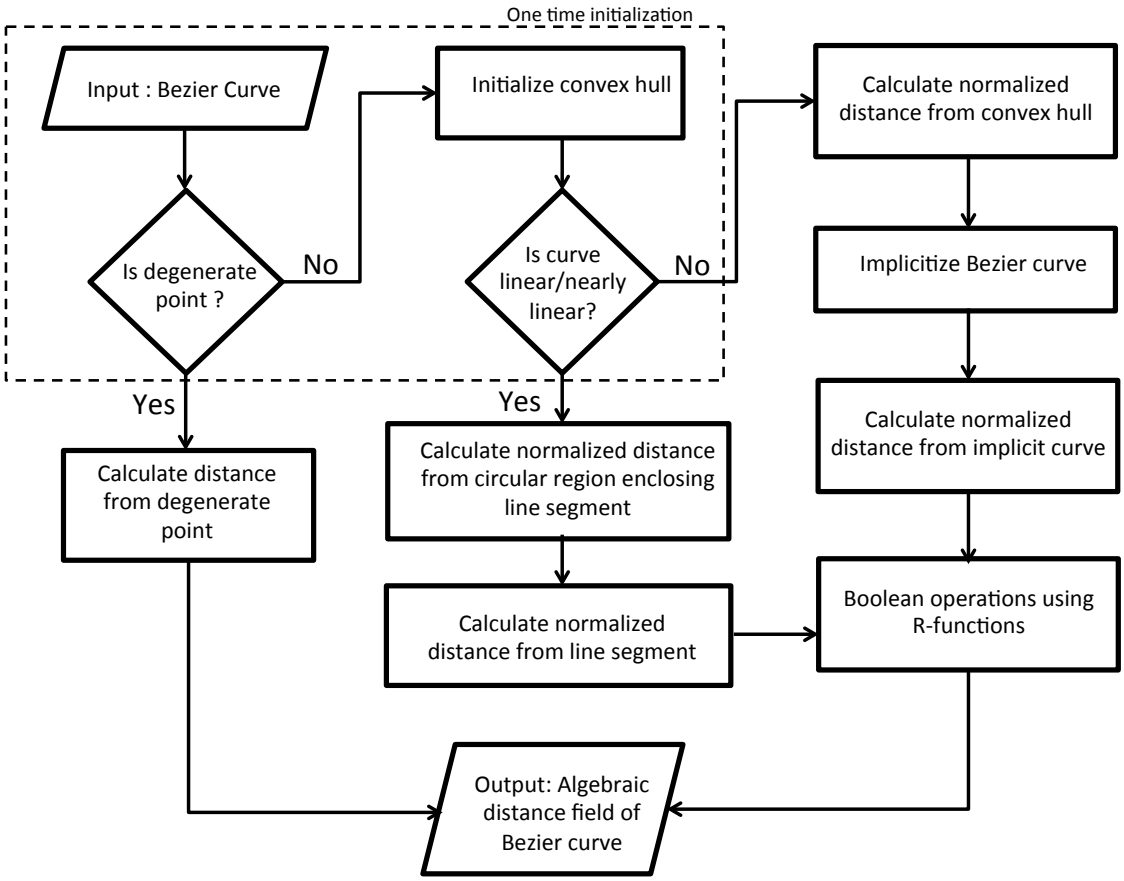


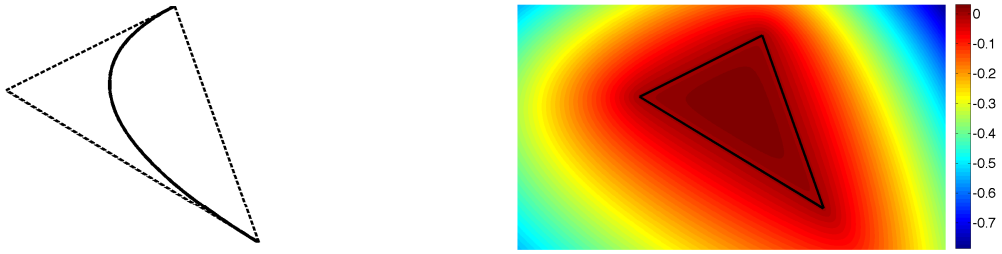
Figure 3.3. Flowchart describing the algebraic distance field construction for a Bezier curve.

Let (x_i, y_i) and (x_{i+1}, y_{i+1}) be the end points of the $i-$th edge of the convex hull, then the algebraic representation of that edge is given by its normalized implicit equation

$$h_i(x,y) = \pm \frac{(x-x_i)(y_{i+1}-y_i) - (y-y_i)(x_{i+1}-x_i)}{\sqrt{(x_{i+1}-x_i)^2 + (y_{i+1}-y_i)^2}} \tag{3.1}$$

The implicit equation $h_i(x, y)$ gives signed distance to an infinite line passing through the edge. The sign of the implicit field is chosen so as to ensure $h_i(x, y) \geq 0$ inside the convex hull. Then, the distance field of the convex polygon is obtained using the R-conjunction operation of Eq. (2.12). Such a process is equivalent to determining

the minimum distance of the implicit fields of the individual edges. The resulting field is a signed distance measure to the convex hull. The field for the convex hull of a Bezier curve is shown in Fig. 3.4.



(a) (b)

Figure 3.4. Function representation of convex hull. (a) Convex hull as trimming region $\phi \geq 0$ for the Bezier curve, (b) Implicit field of convex hull of the Bezier curve.

In the case of linear or nearly linear Bezier curve, it is not possible to construct a convex hull from the control points. For such a case, a circle with the line segment joining the end points of the Bezier curve as diameter is chosen as the trimming region $\phi \geq 0$.

Normalized Distance Field of Implicit Curve

An implicit representation f of each Bezier curve is obtained by the implicitization process using Bezout resultant. The implicit function is normalized locally near the curve by scaling it by the gradient value at a point on the curve. The gradient value at one of the end control points of the Bezier curve was chosen here for scaling the implicit function. The derivative of the implicit function for scaling is calculated by

exploiting the Bezout resultant matrix A described earlier in Eq. (2.7). The derivative is obtained using Jacobi's formula as:

$$\frac{\partial |A|}{\partial x_i} = tr\left[adj(A)\frac{\partial A}{\partial x_i}\right] \tag{3.2}$$

where, $\frac{\partial}{\partial x_i}$ is partial derivative with respect to the ith coordinate variable, tr is the trace operator, and adj is the adjugate or the transpose of the cofactor matrix. Since each term in the Bezout matrix A is linear in x and y, the derivative of Bezout resultant matrix in Eq. (3.2) is a constant. In the case of linear or nearly linear Bezier curve, implicit representation f is the line passing through the end points of the Bezier curve. Then, the normalized distance representation of the line is given by Eq. (3.1).

Trimming Operation

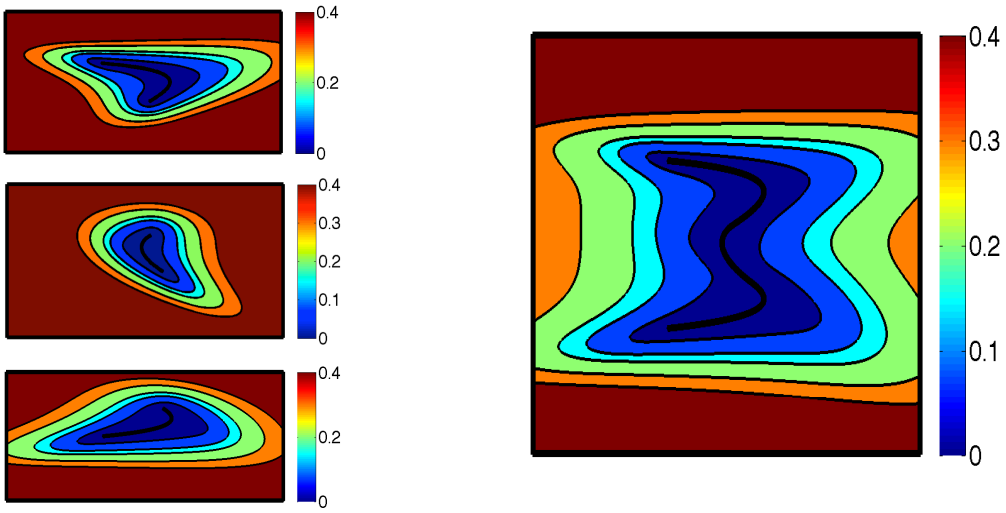
After the normalized fields are computed for the trimming region as well as the implicit curve, the algebraic distance field for Bezier curve is calculated by the trimming operation on the normalized fields using Eq. (2.15). Here, the convex hull of the Bezier curve is used as the trimming region ϕ for the implicitized representation f of the Bezier curve. An example of the calculated algebraic distance field for a Bezier curve is shown in Fig. 3.5(a).

3.1.3 Compositions using R-conjunction

Further compositions are then carried out between adjoining Bezier curve fields using the R-conjunction operation given in Eq. (2.12). The resulting distance field, given below, corresponds to the nearest Bezier segment:

$$g_1 \wedge g_2 = g_1 + g_2 - \sqrt{g_1^2 + g_2^2} \tag{3.3}$$

where, g_1 and g_2 are algebraic distance fields of adjoining Bezier curves. The composed distance field for the NURBS curve is shown in Fig. 3.5(b).



(a) (b)

Figure 3.5. Composition of distance field of Bezier curves. (a) Distance field of each Bezier curve, (b) Composed distance field of the NURBS curve.

3.2 Extension to NURBS Surfaces

The algebraic distance field construction technique for planar NURBS and Bezier curves is extended to three-dimensional NURBS and Bezier surfaces in this section. The basic theory and algorithm remain the same, only difference being the underlying technique used for implicitization and the technique used for constructing the trimming region. Bezier surfaces are implicitized using Dixon's resultant described in Section 2.3.2.

3.2.1 Trimming Region for Bezier Surface

For the case of a three-dimensional parametric surface, the convex hull is constructed from the control point grid of the Bezier patch (Fig. 3.6). Each face of the convex hull is described by the equation of a plane. Let (x_1, y_1, z_1), (x_2, y_2, z_2) and

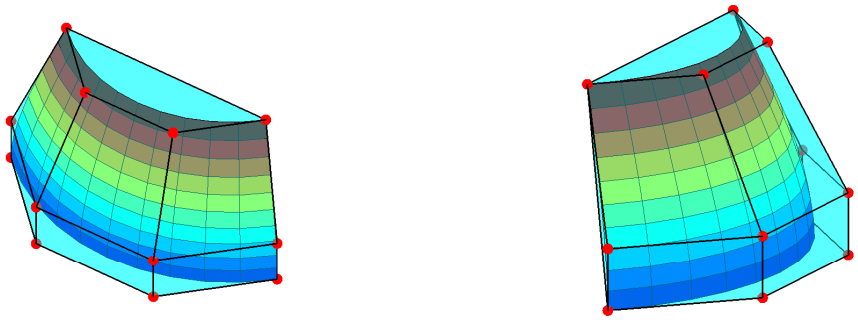
(x_3, y_3, z_3) be three non-collinear vertices of the $i-$th face of the convex hull. Then, the normalized field of the face is constructed using the following implicit equation

$$h_i(x, y, z) = \pm \frac{Ax + By + Cz + D}{\sqrt{A^2 + B^2 + C^2}} \tag{3.4}$$

where,

$$A = \begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix}, B = \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}, C = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}, D = - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}$$

The distance measure corresponding to the convex hull is obtained by R-conjunction operation on the normalized fields of the faces. In the case of planar, or nearly planar



(a) (b)

Figure 3.6. Three-dimensional convex hull as the trimming region for a Bezier patch.

Bezier patch, since its control point grid degenerates to a plane, it is not possible to construct a three-dimensional convex hull from the control points. In such a planar case, the implicit representation of the patch is the plane containing the patch. Hence, a trimming region has to be constructed such that its intersection with the plane gives the planar Bezier patch. This implies that the boundary of the patch must lie on the the trimming region. Using this insight, the trimming region is chosen to be the boundary Γ of the planar patch as shown in Fig. 3.7.

Now, the trimming region ϕ corresponds to the planar region bounded by Γ and its implicit representation is obtained by using the algebraic distance field computation procedure for planar curves. The R-function in Eq. (2.14) is used to construct the algebraic distance field of the planar patch. First, the normalized implicit field f_1 is obtained as the implicit equation of the plane containing the patch. The implicit function f_2 is constructed from the implicit representation of trimming region ϕ such that it is identically zero in the interior of Γ. At any point in the domain, f_2 is computed at its corresponding projected point on the plane of the Bezier patch.

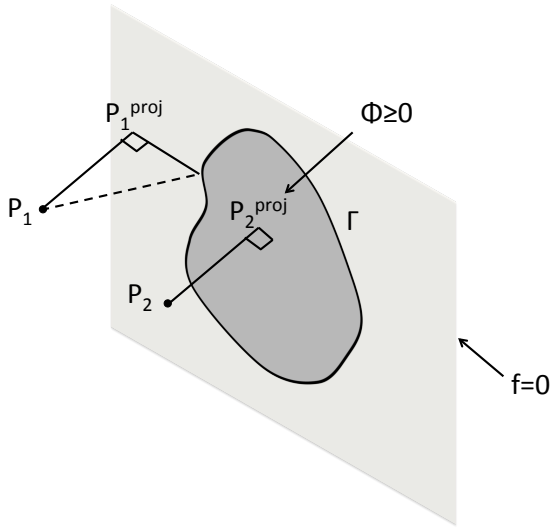


Figure 3.7. Trimming region for planar Bezier patch.

This process is illustrated in Fig. 3.7 using two test points P_1 and P_2. In each case, the test point is projected onto the plane of the patch and then algebraic distance is calculated to the bounding curve Γ. If the projected point P_1^{proj} lies outside the bounding curve then the algebraic distance field of the planar patch is computed as

$$g = \sqrt{f_1^2 + f_2^2}$$

where, f_2 is the distance from the projected point P_1^{proj} to Γ. If the projected point P_2^{proj} is on or inside the bounding curve then Eq. (2.14) reduces to normalized

distance from plane, i.e., $g = f_1$. The distance field for a planar patch is illustrated in Fig. 3.8.

(a) (b)

Figure 3.8. Planar patch. (a) Planar NURBS patch, (b) Algebraic distance field of the planar patch on a plane slicing the patch in three-dimensional space.

3.2.2 Normalization of Three-Dimensional Distance Fields

In general, the distance fields computed on the three-dimensional convex hull as well as the implicitized Bezier patch needed greater care in normalization to ensure that the resulting distance fields resembled the exact ones.

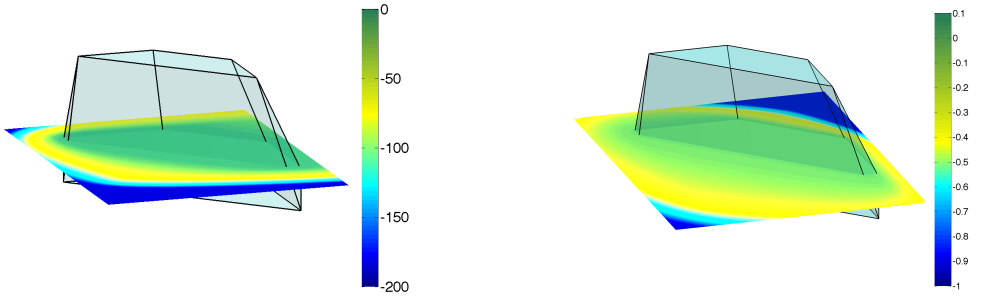
Normalized Distance Field of Three-Dimensional Convex Hull

The distance field of the three-dimensional convex hull was constructed by R-conjunction operations on normalized implicit field of its planar faces. As mentioned in Section 2.6, the R-conjunction operation preserves normalization at almost all points except at points on the intersection edge between any two faces. Hence, after each R-conjunction operation, the resulting distance field was normalized prior to further compositions. The field ϕ_k, obtained after each pairwise R-conjunction oper-

ation on the $\hat{\phi}_{k-1}$ normalized field and kth face of the convex hull, was normalized by its gradient as follows:

$$\hat{\phi}_k = \frac{\phi_k}{|\nabla \phi_k|} \tag{3.5}$$

For normalized fields of the convex hull faces, $|\nabla \phi_k| = 1$ near the faces except at their intersection. In Section 2.6, the bounds on the gradient of field obtained using R-conjunction operation near the intersection region was derived as $0 < |\nabla \phi_k| \leq 3$. Here, normalization of the R-conjunction fields is carried out by a scaling operation using a representative value of $\nabla \phi_k$ such that $1 < |\nabla \phi_k| \leq 3$. In this work, $\frac{\partial \phi_k}{\partial x_i}$ is chosen to be 1 such that $|\nabla \phi_k| = \sqrt{3}$. The algebraic distance field of the convex hull with and without the proposed scaling are compared in Fig. 3.9.



(a) (b)

Figure 3.9. Normalization of field of trimming region. (a) Field of trimming region before scaling, (b) Field of trimming region after scaling.

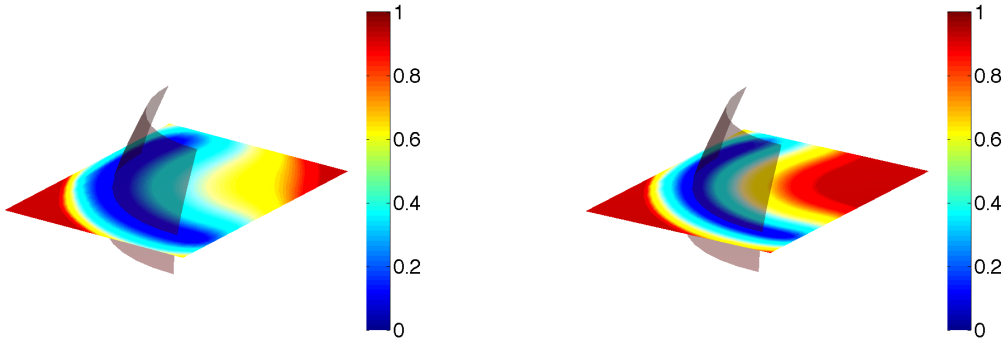
Normalized Distance Field of Implicit Surface

The implicit function f of the Bezier patch obtained using Dixon's resultant is normalized by scaling it using the ratio of the implicitized distance $s(\mathbf{x}_r)$ and the exact distance $d(\mathbf{x}_r)$ at a representative point $\mathbf{x}_r$. Here, a control point that does not

lie on the Bezier patch is chosen as the representative point $\mathbf{x}_r$. The scaled implicit function $\hat{f}$ is then given by

$$\hat{f} = \frac{f}{(\frac{s(\mathbf{x}_r)}{d(\mathbf{x}_r)})} \tag{3.6}$$

This strategy was used to avoid computing the gradient $|\nabla f|$ since that is computationally more expensive to carryout. The algebraic distance field for a Bezier patch with non-normalized and normalized implicit function are compared in Fig. 3.10.



(a) (b)

Figure 3.10. Algebraic distance field of a bicubic Bezier patch. (a) Distance field of Bezier patch before scaling of implicit field, (b) Distance field of Bezier patch after scaling of implicit field.

3.3 Time Complexity of Algebraic Distance Field Algorithm

Consider a NURBS curve of degree p. Further, assume that the NURBS curve decomposes into n Bezier curves. Then, the time complexity of this decomposition process is $O(n)$. Each Bezier curve comprises of $p+1$ control points. Hence, for each step in constructing algebraic distance field of the Bezier curve, the time complexity is a function of its degree p as follows -

- Time complexity of convex hull construction is $O(p \log(p))$
- Time complexity of computing normalized distance field of convex hull is $O(p)$
- Time complexity of computing the resultant depends on the determinant calculation and is $O(p^3)$
- Time complexity of trimming operation is $O(1)$

Thus, the time complexity of constructing algebraic distance field of Bezier curve is $O(p^3)$ and that for the NURBS curve is $O(np^3)$.

Given a NURBS surface of degree $p \times p$ that decomposes into n Bezier patches, the time complexity can be computed as follows -

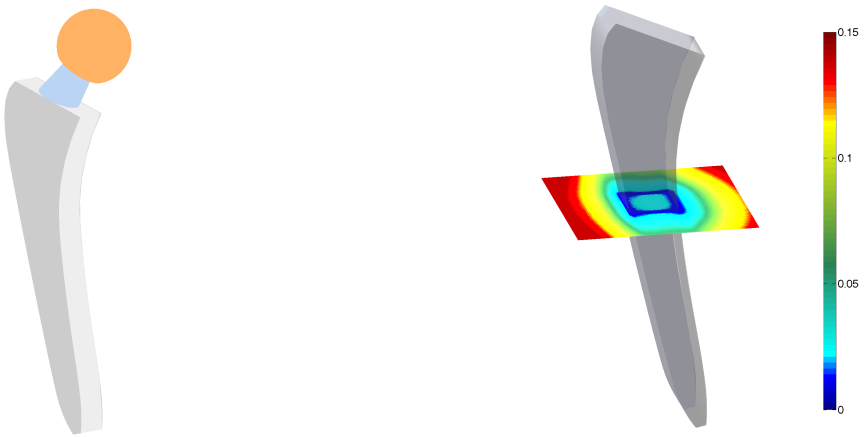
- Time complexity of convex hull construction is $O(p^2 \log(p))$
- Time complexity of computing normalized distance field of convex hull is $O(p^2)$
- Time complexity of computing the resultant depends on the determinant calculation and is $O(p^6)$
- Time complexity of trimming operation is $O(1)$

Thus, the time complexity of constructing algebraic distance field of Bezier surface is $O(p^6)$ and that for the NURBS surface is $O(np^6)$.

3.4 Algebraic Distance Field for Complex Three-Dimensional Geometries

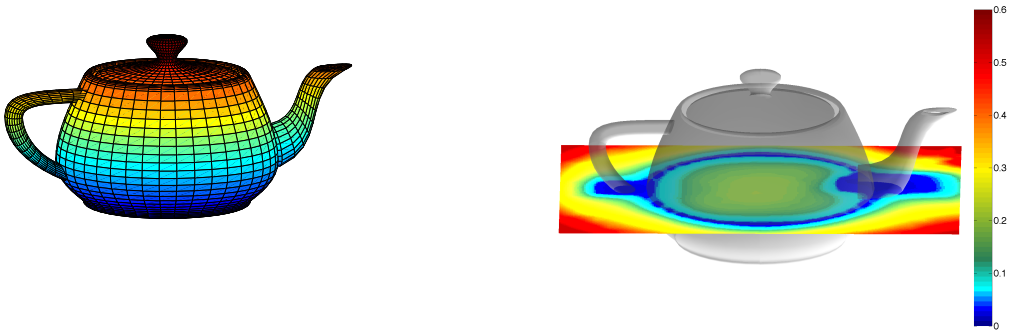
In this example, the algebraic distance field is demonstrated on the three-dimensional geometry of a hip prosthesis. A representative model of the hip implant (Fig. 3.11(a)) was constructed using Bezier patches. The algebraic distance field over a cross-section of the prosthesis is shown in Fig. 3.11(b).

A more complex geometry, of that of the Utah teapot [79], is demonstrated next. The teapot model consisted of 32 bicubic Bezier patches (Fig. 3.12(a)). The algebraic



(a) (b)

Figure 3.11. Hip prosthesis. (a) Model of hip implant constructed in this study for demonstration, (b) Algebraic distance field over a cross-section of the hip implant.



(a) (b)

Figure 3.12. The utah teapot. (a) The teapot modeled using 32 bicubic Bezier patches, (b) Algebraic distance field on a plane slicing the teapot.

distance field for Utah teapot is demonstrated on a plane slicing the geometry (Fig. 3.12(b)).

4. PROPERTIES OF THE ALGEBRAIC DISTANCE FIELD

The algebraic distance field proposed in this work is a non-iterative and efficient approximation to the exact distance function obtained through Newton-Raphson iterations. In this chapter the properties of the algebraic distance field are examined to determine the quality of its approximation to exact distance. The properties of the algebraic distance are detailed below, and the performance and robustness of the algebraic measure are compared against Newton-Raphson iterations that is used to determine the orthogonal distance.

4.1 Exact Value at Zero Level Set

An important property of distance function satisfied by the algebraic distance field is zero value on the respective curve or in other words exact value on the zero level set. This implies that the zero level set of the algebraic distance field will exactly represent the curve geometry in implicit representation. Thus, the algebraic distance field preserves the geometric exactness of the explicit boundary.

4.2 Differential Properties

The differential properties of the algebraic distance field depend on the differential properties of the implicit representation of Bezier curve and its convex hull, and also on the R-functions used for performing boolean operations on the implicit functions. During the process of distance field construction, all implicit functions are normalized to first order near the boundary. This implies that these functions behave like exact distance function and satisfy an important property of distance function (Eq. (2.2)) near the boundary. Such a normalization is preserved by R-function operations almost

everywhere. The effect of R-function operations on the differential properties of the algebraic distance field is discussed next.

The following two-dimensional analysis, motivated by the analysis in [72], enables the estimation of the bounds for gradient of approximate distance field in the neighbourhood of a boundary. Let $\hat{u}$ be the approximate distance function obtained by R-function operation F on normalized fields ϕ_1 and ϕ_2.

$$\hat{u} = F(\phi_1, \phi_2)$$

The partial derivatives and the magnitude of gradient of $\hat{u}$ are

$$\frac{\partial \hat{u}}{\partial x} = \frac{\partial F}{\partial \phi_1}\frac{\partial \phi_1}{\partial x} + \frac{\partial F}{\partial \phi_2}\frac{\partial \phi_2}{\partial x} \ , \ \frac{\partial \hat{u}}{\partial y} = \frac{\partial F}{\partial \phi_1}\frac{\partial \phi_1}{\partial y} + \frac{\partial F}{\partial \phi_2}\frac{\partial \phi_2}{\partial y}$$

$$|\nabla \hat{u}|^2 = \left(\frac{\partial F}{\partial \phi_1}\frac{\partial \phi_1}{\partial x} + \frac{\partial F}{\partial \phi_2}\frac{\partial \phi_2}{\partial x}\right)^2 + \left(\frac{\partial F}{\partial \phi_1}\frac{\partial \phi_1}{\partial y} + \frac{\partial F}{\partial \phi_2}\frac{\partial \phi_2}{\partial y}\right)^2$$

If ϕ_1 and ϕ_2 are normalized, i.e., $|\nabla \phi_1| = 1$ and $|\nabla \phi_2| = 1$,

$$|\nabla \hat{u}|^2 = \left(\frac{\partial F}{\partial \phi_1}\right)^2 + \left(\frac{\partial F}{\partial \phi_2}\right)^2 + 2\frac{\partial F}{\partial \phi_1}\frac{\partial F}{\partial \phi_2}\left(\frac{\partial \phi_1}{\partial x}\frac{\partial \phi_2}{\partial x} + \frac{\partial \phi_1}{\partial y}\frac{\partial \phi_2}{\partial y}\right)$$

$$|\nabla \hat{u}|^2 = \left(\frac{\partial F}{\partial \phi_1}\right)^2 + \left(\frac{\partial F}{\partial \phi_2}\right)^2 + 2\frac{\partial F}{\partial \phi_1}\frac{\partial F}{\partial \phi_2}\left(\nabla \phi_1 . \nabla \phi_2\right) \tag{4.1}$$

Since, ϕ_1 and ϕ_2 are normalized, $\nabla \phi_1 . \nabla \phi_2$ varies between -1 and +1. Hence, the theoretical bounds on $|\nabla \hat{u}|$ are

$$\left(\frac{\partial F}{\partial \phi_1} - \frac{\partial F}{\partial \phi_2}\right)^2 \leq |\nabla \hat{u}|^2 \leq \left(\frac{\partial F}{\partial \phi_1} + \frac{\partial F}{\partial \phi_2}\right)^2 \tag{4.2}$$

Using R-conjunction Eq. (2.12), $F(\phi_1, \phi_2)$ is given by

$$F = \phi_1 + \phi_2 - \sqrt{\phi_1^2 + \phi_2^2}$$

Now, the partial derivatives of F with respect to ϕ_1 and ϕ_2 are

$$\frac{\partial F}{\partial \phi_1} = 1 - \frac{\phi_1}{\sqrt{\phi_1^2 + \phi_2^2}} \ , \ \frac{\partial F}{\partial \phi_2} = 1 - \frac{\phi_2}{\sqrt{\phi_1^2 + \phi_2^2}}$$

$$\frac{\partial F}{\partial \phi_1} - \frac{\partial F}{\partial \phi_2} = \frac{\phi_2 - \phi_1}{\sqrt{\phi_1^2 + \phi_2^2}} \ , \ \frac{\partial F}{\partial \phi_1} + \frac{\partial F}{\partial \phi_2} = 2 - \frac{\phi_2 + \phi_1}{\sqrt{\phi_1^2 + \phi_2^2}} \tag{4.3}$$

Near the boundary, if $\phi_1 = 0$ and $\phi_2 \neq 0$ or $\phi_1 \neq 0$ and $\phi_2 = 0$, $|\nabla \hat{u}| = 1$. However, for points where $\phi_1 = 0$ and $\phi_2 = 0$, $|\nabla \hat{u}| \neq 1$. Thus, R-conjunction preserves normalization near the boundary except when $\phi_1 = 0$ and $\phi_2 = 0$. It can be shown that at such points the gradient is bounded as follows.

In the $\phi_1 - \phi_2$ coordinate plane, the point of interest is the origin, where $\phi_1 = \phi_2 = 0$. A quadratic transformation [80] of the $\phi_1 - \phi_2$ plane to $\phi_1' - \phi_2'$ plane is defined as $\phi_1' = \phi_1$ and $\phi_2' = \phi_2/\phi_1$ as shown in Fig. 4.1. Under this transformation the origin is mapped onto the entire ϕ_2'-axis. At the origin in the $\phi_1 - \phi_2$ plane, an indeterminate form $\phi_2' = \frac{0}{0}$ exists.

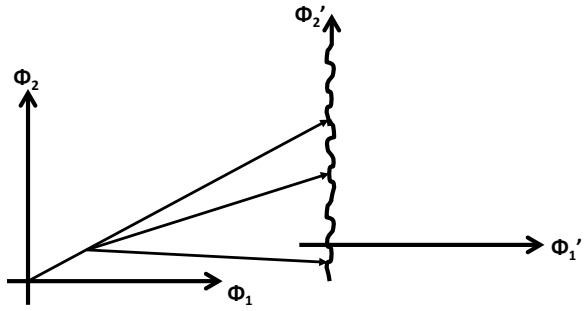


Figure 4.1. Quadratic transformation of $\phi_1 - \phi_2$ plane to $\phi_1' - \phi_2'$ plane.

By approaching the origin along the line of slope m (line $\phi_2 = m\phi_1$),

$$\phi_2' = \frac{\phi_2}{\phi_1} = \frac{m\phi_1}{\phi_1} = m, \quad \forall \ \phi_1 \neq 0$$

hence,

$$\lim_{\phi_1 \to 0} \phi_2' = m \ \Rightarrow \ \lim_{\phi_1 \to 0} \frac{\phi_2}{\phi_1} = m$$

Substituting in Eq. (4.3),

$$\frac{\partial F}{\partial \phi_1} - \frac{\partial F}{\partial \phi_2} = \frac{m-1}{\sqrt{1+m^2}} \ , \ \frac{\partial F}{\partial \phi_1} + \frac{\partial F}{\partial \phi_2} = 2 - \frac{m+1}{\sqrt{1+m^2}} \tag{4.4}$$

Thus the bounds for gradient of distance function in the neighbourhood of the boundary are given by

$$|\frac{m-1}{\sqrt{1+m^2}}| \leq |\nabla \hat{u}| \leq |2 - \frac{m+1}{\sqrt{1+m^2}}| \tag{4.5}$$

Thus, the maximum value of the upper bound is 3 and the minimum value of lower bound is 0 (Fig. 4.2).

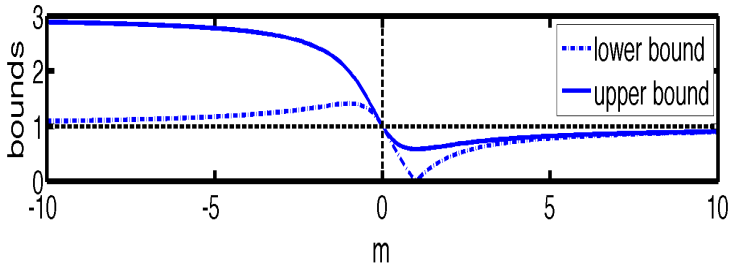


Figure 4.2. Bounds on the gradient of R-conjunction composition of normalized algebraic distance fields.

The bounds on the gradient of algebraic distance field in the neighbourhood of the boundary completely characterize rate of change of distance in this region.

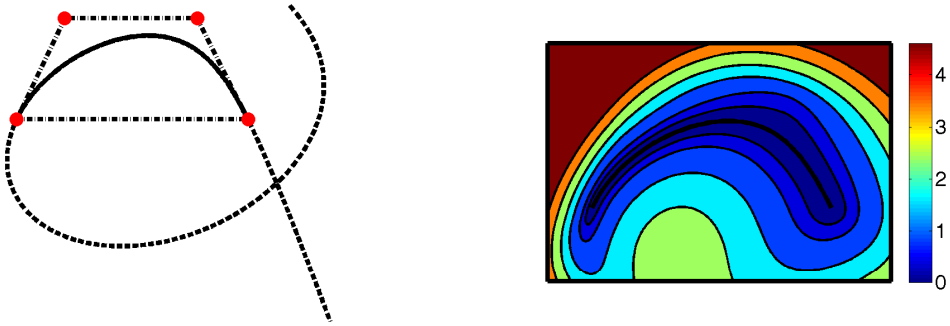
4.3 Double Points and their Elimination

The methodology described in this work exploits algebraic geometry concepts that are briefly elaborated here. Given an irreducible plane algebraic curve C_p of degree p, by Bezout's theorem [80], it can be shown that the curve of algebraic degree p may possess as many as $\frac{(p-1)(p-2)}{2}$ self-intersections or double points (see Fig. 4.3). The critical question of concern here is whether the implicitized version of the rational parametric curve will possess self-intersections. This possibility depends on the inverse mapping, given an algebraic curve, to its equivalent rational parametric form. The existence of the rational parametric form depends on the genus G of the curve defined as

$$G = \frac{(p-1)(p-2)}{2} - DP \tag{4.6}$$

where, DP is the number of double points present in the algebraic curve. In order for a curve to have a rational parametrization, its genus needs to be zero. Thus, an implicitized curve obtained from a rational parametric curve of degree three will

posses a single double point when implicitized. However, if the singular point lies



(a) (b)

Figure 4.3. Illustration of singular points on a degree three Bezier curve. The homogeneous control points are (-10,-8,0,1), (-8,-1.2,0,1.2), (5,-2,0,2), and (6,-8,0,1). (a) The lone singular point lies outside the convex hull of Bezier curve allowing one to construct, (b) The algebraic distance field of Bezier curve.

outside the convex hull region (see Fig. 4.3), then the algebraic distance field is not affected by the singular point as the convex hull is used to trim the distance field. But, if the singular point falls inside the convex hull, then it affects the algebraic distance construction. In such a case, the parametric curve is subdivided at the singular point to force the singular point to lie outside the trimming region (Fig. 4.4). The parameter t corresponding to the singular point is calculated using moving line techniques [81–83] discussed briefly in the next section. Subdivision is performed by using the knot insertion algorithm at the parameter t on the curve. Algebraic distance field is then constructed for each sub-curve and composed using R-functions.

The above technique can be extended to higher degree curves with multiple singularities. However, higher the degree of the curve, the greater the number of singular points per Eq. (4.6). Therefore, in this work, the methodology is restricted to low degree NURBS curves ($d \leq 3$).

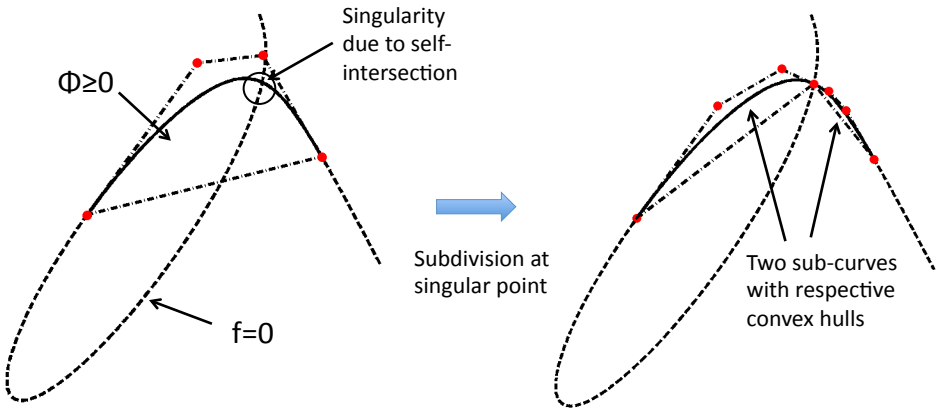


Figure 4.4. Subdivision of curve to remove the singular point from the convex hull.

4.3.1 Computing Singular Points of Plane Rational Curves

Computing singular points of a planar rational curve is non- trivial. Given an algebraic curve $f(x, y, w) = 0$ in its homogeneous form, a point $P_0 = (x_0, y_0, w_0)$ is a singular point on the curve if and only if

$$f_x(x_0, y_0, w_0) = 0, \qquad f_y(x_0, y_0, w_0) = 0, \qquad f_w(x_0, y_0, w_0) = 0 \tag{4.7}$$

P_0 is a singular point of $f = 0$ with multiplicity r if all derivatives of order $r - 1$ are zero at P_0 and atleast one rth derivative of f does not vanish at P_0. The standard way to obtain singular points is to solve the above system of equations [84]. This method involves implicitizing the parametric curve symbolically to obtain $f(x, y, w) = 0$, then differentiating f to find its partial derivatives and then solving the system of equations. Hence, it is not very convenient to be used as a part of geometric algorithms. Alternately, the parametric curve can be directly used to determine the singular points [81, 82].

Let $P(t) = (x(t), y(t), w(t))$ be a rational curve and $L(t)$ denote the moving line $a(t)x + b(t)y + c(t) = 0$, then $L(t)$ follows $P(t)$ if [85]

$$P(t).L(t) = 0 \qquad or \qquad a(t)x(t) + b(t)y(t) + c(t)w(t) = 0$$

For any rational cubic bezier curve, there exist two moving lines of degrees 1 and 2 respectively, which follow the curve and are referred to as μ-basis [83]. The degree 1 moving line is an axial moving line with axis at the singular point of the curve. This means that all the lines in the family will pass through the axis point. Hence, the singular point is obtained as the intersection of any two lines belonging to this family. In Fig. 4.5, the solid lines represent the degree 1 moving line with axis at the singular point of the curve. The dotted lines represent the degree 2 moving line. The intersection of the two moving lines gives the points on the curve. The parameter t corresponding to the singular point is calculated from the degree 2 moving line equation.

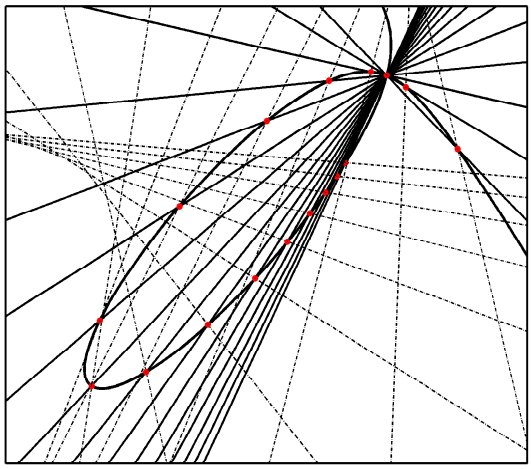


Figure 4.5. Moving lines following the cubic curve.

4.4 Numerical Efficiency Relative to Newton-Raphson Iterations

The time complexity of the algebraic distance construction algorithm is derived in 3.3. In general, the cost of computing the algebraic distance field from a NURBS curve is $O(np^3)$, where n is the number of Bezier segments in the NURBS curve and p is the degree of the curve. With the assumption of low algebraic degree, p may be

assumed to be fixed. Hence, in practice, the computational cost is dependent on n alone.

Next, the computational efficiency of the algebraic distance field is compared to the Newton-Raphson iterations through the following examples. Three NURBS curves (Fig. 4.6) of degrees two, three and four respectively are chosen for the test. The distance field is constructed both algebraically and using Newton-Raphson iterations. The corresponding computational expense is plotted in Fig. 4.7. As can be seen, the algebraic distance estimation costs approximately $12 - 14\%$ of the Newton-Raphson iterations.

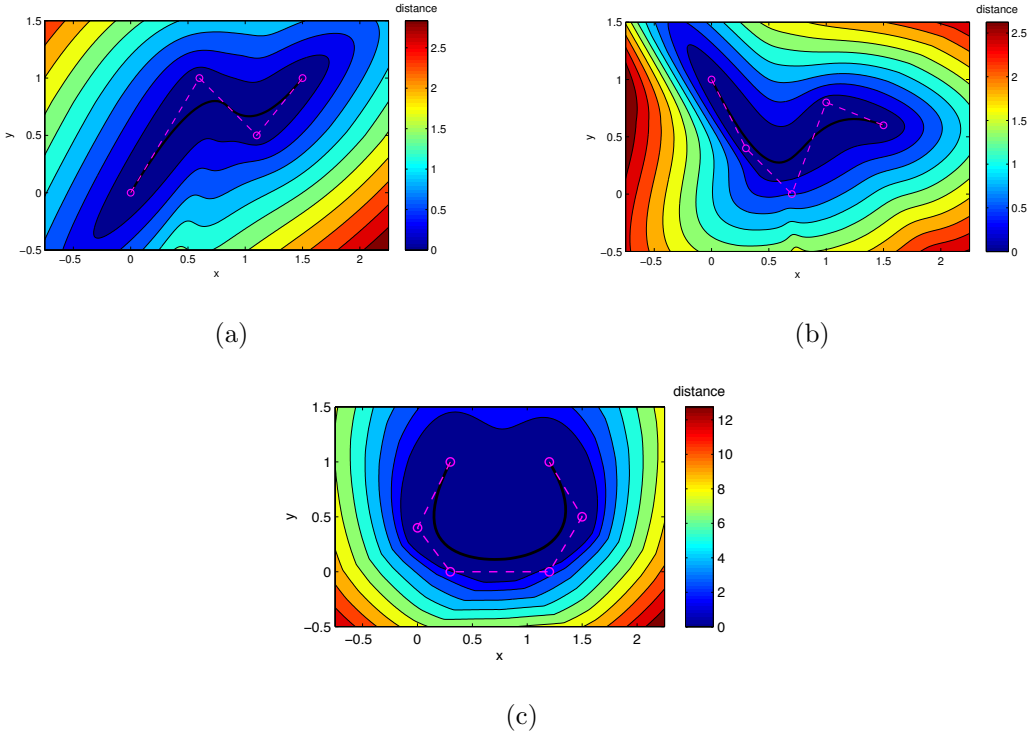


Figure 4.6. Test curves with corresponding control net and algebraic distance field. (a) A quadratic NURBS curve, (b) A cubic curve and (c) A quartic curve.

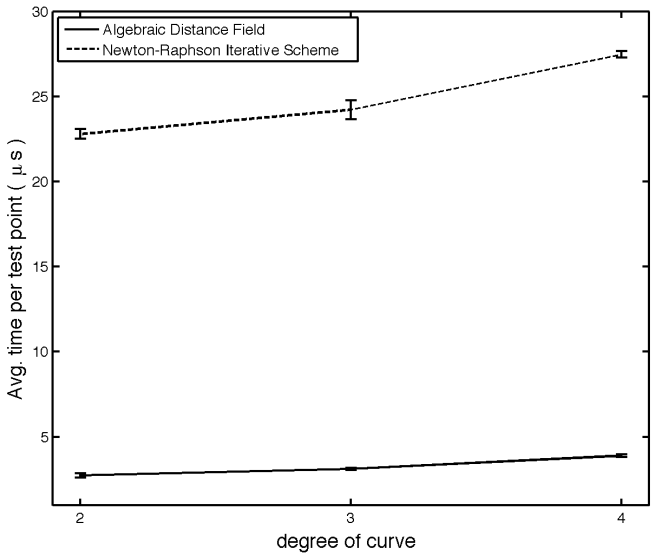
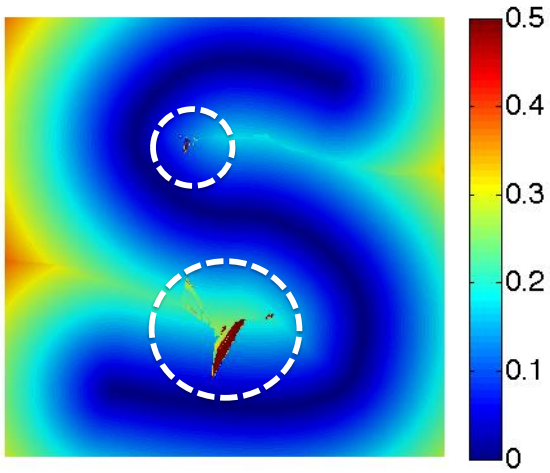


Figure 4.7. Comparison of algebraic distance method and Newton-Raphson iterative scheme for curves of varying degree in Fig. 4.6 (tested using computer with single Intel i5 processor and 4G memory).

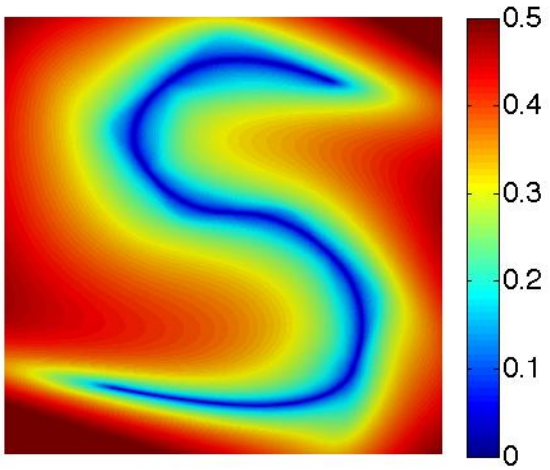
4.5 Uniqueness of Solution and Robustness

The exact distance field and the algebraic distance field for an S-shaped curve are compared in Fig. 4.8. The circular regions marked in the figure indicate points that are equidistant from the curve. At these points, the exact distance calculations are non-unique leading to non-smooth distance fields overall. The algebraic distance field, on the other hand, gives a smoother estimate of the distance to the curve while preserving exactness to the parametric curve. Thus, the algebraic procedure provides a more robust estimate of the distance.

Unlike the exact distance function that is often not uniquely defined near points of high curvature, the algebraic distance field is unique and sufficiently smooth for use during analysis.



(a)



(b)

Figure 4.8. Comparison of distance fields. (a) Exact distance field with the marked regions corresponding to points that are equidistant to two different points on the curve, and (b) Algebraic distance field.

5. SIGNED DISTANCE FIELD CONSTRUCTION

The algebraic distance field discussed in the previous chapters is unsigned by construction. This is due to the composition of constituent algebraic distance field of Bezier patches to obtain the resulting distance measure of the bounded solid. In general, the constituent Bezier curves or patches are not bounded and do not divide the space into inside and outside regions. Hence, the concept of sign does not apply in this case. Thus, the constituent algebraic distance fields of Bezier patches are unsigned by construction and lead to unsigned field for the bounded solid. The only case when a signed distance field can be constructed from the algebraic distance technique is if the constituent Bezier geometry is closed or bounded.

While an unsigned distance field is sufficient to model the influence of known behavior on boundaries over the underlying domain, sign of the distance field for bounded solids further enable the point membership query in CAD/CAE applications. Signed distance function finds application in contact problems for constructing the gap function for contact condition between two bodies [59, 86]. Signed distance function finds application in variety of analysis problems solved using level set methods such as topology optimization [87, 88], stefan problem for solidification [89, 90], fluid-interface problems [56, 91], crack and fracture [92–94]. The signed distance to the interface is used for initialization of the level set function in such applications. Further, signed distance field has also found applications in collision detection [60,95]. It is also applied for volumetric geometry representation [96] and morphing [97] in CAD applications. The applications of signed distance function are summarized in Table 5.1.

In this chapter, algorithms are developed for computing the sign associated with the distance field of a closed NURBS curve or NURBS solid boundary. It will be seen

Table 5.1. Classification of problems in CAD/CAE applications that require signed distance function.

Problem Characteristic	**Type**	**Application**	**Example Problems**
Geometry Modeling	CAD	Geometry and Volume Representation	Offset surfaces and medial axis problems
Blending	CAD	Geometry Modification	Morphing between shapes, filleting/chamfering
	CAD	Heterogeneous Material Modeling	Functionally graded materials
	CAE	Enriched Field Approximations	Crack, material interface, boundary condition
Multi-Body Interaction	CAD	Collision Detection	Robotics and automation, animation
	CAD/CAM	Interference Detection	Verification of manufacturing assemblies, machining toolpath generation
	CAE	Contact Mechanics	Impact and metal forming simulations

later in the chapter that this sign is calculated as by-product of the algebraic distance field calculation technique.

5.1 Signed Distance Function

Given a domain Ω, bounded by the region $\partial\Omega$, for any point $x \in \mathbb{R}^n$ a signed distance representation $d_\Omega(x)$ of the domain is given by the following expression

$$d_\Omega(\mathbf{x}) = \mathbf{sign}(\mathbf{x}) \inf_{\mathbf{p}\in\partial\mathbf{\Omega}} \parallel \mathbf{x} - \mathbf{p} \parallel \tag{5.1}$$

where

$$\mathbf{sign}(\mathbf{x}) = \begin{cases} 1 & \text{if } x \in \Omega, \\ -1 & \text{otherwise} \end{cases} \tag{5.2}$$

While the sign of the distance function enables classification of points as inside or outside the solid; the distance from the boundary captures the influence of fields defined on the boundaries over the interior and hence enables queries of the field values (material or behavior) and their derivatives in the interior. Thus, given only the boundary of a bounded solid, it is possible to infer volumetric data for the solid by constructing its signed distance representation.

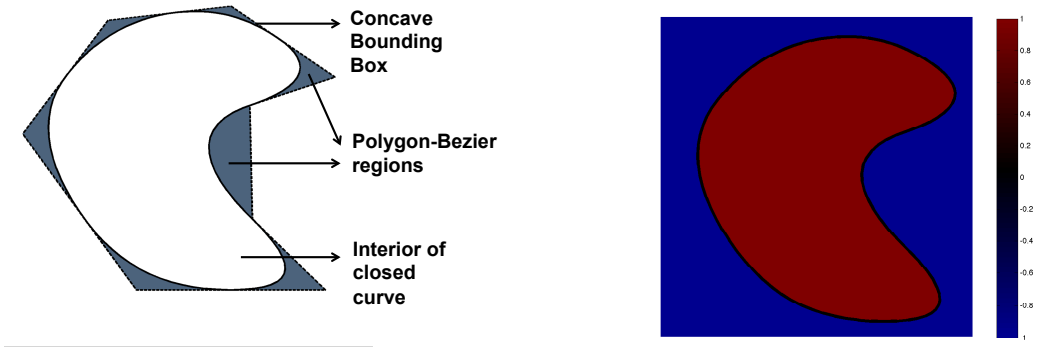
In general, sign of the distance field is computed by projecting the query point to the surface of the solid and evaluating the dot product of the normal at the projected point and the direction vector from the query point to the projected point. For piecewise planar approximations, point projection operation is computationally inexpensive. However, the challenge is that normal is not defined on edges and vertices. Hence, sign calculation techniques for such geometries focus on algorithms for computing approximate normals at such points. For non-linear geometries, the point projection problem is computationally expensive as it involves iterative solution procedure. Hence, a computationally inexpensive non-iterative solution is desired for the signed distance problem. In this work, an algebraic measure of signed distance is proposed. Since, this is an approximation to the exact signed distance function,

it is referred to as Algebraic Level Set or Algebraic Distance Field instead of Signed distance function.

5.2 Algorithm for Algebraic Sign Calculation

The algorithm for algebraic sign calculation is based on case by case point containment queries. First, a concave/convex bounding box is constructed for the closed boundary using the convex hull of each Bezier patch (Fig. 5.1(a)). The point containment is queried in this bounding box. If the point is outside the bounding box, then the algorithm stops and point is outside the bounded solid. However, if it is inside the box then the next case is queried. In the next step, point containment queries are performed in the Polygon-Bezier regions shown in Fig. 5.1. If query point lies inside one of these regions, then the algorithm stops and point is outside the bounded solid, else the query point lies within the solid.

Positive value of the algebraic sign function implies point containment in the enclosed region while zero implies a boundary point. Let P_i be a point in the plane



(a) (b)

Figure 5.1. Construction of signed distance field of closed NURBS curve. (a) Illustration of sign calculation technique, (b) Sign of the field of the closed NURBS curve.

and $g(x, y)$ be the distance field of the closed NURBS curve with bounding box field

ϕ and m curved Bezier polygons (shaded regions in Fig. 5.1(a)) with distance field $\psi_j, j \in 1, \ldots, m$. The sign associated with the distance field $g(x, y)$ at point P_i is given by the following scheme

$$\text{sign}(g) = \begin{cases} 0 & \text{if } g = 0 \ , \\ \text{-1} & \text{if } (\phi < 0) \text{ or } (\phi \geq 0 \text{ and any } \psi_j \geq 0, j \in 1, \ldots, m \text{ and } g \neq 0) \ , \\ 1 & \text{otherwise} \end{cases}$$

This scheme applies to bounded NURBS surfaces as well. In the following sections, algorithmic details of constructing the bounding box and polygon-Bezier regions will be discussed. Further, algorithms for point containment queries in the bounding box and the algebraic sign calculation in polygon-Bezier regions will be detailed.

5.3 Construction of Convex/Concave Bounding Box from Bezier Control Polygons

5.3.1 Bounding Box for Closed Planar Curves

The first step in the sign calculation is to construct a tight bounding box for the closed curve. It is assumed that the control points are always stored in a counter-clockwise order. Consider the convex hull of the ith Bezier curve with vertices $v_1, v_2, \ldots, v_k, \ldots v_n$, where v_1 and v_n are the end points of the bezier curve. Then, any vertex v_k which is on the left of the line segment formed by v_1 and v_n is removed and the remaining vertices belong to the bounding box as illustrated in Fig. 5.2. An example of bounding box construction is shown in Fig. 5.3. A closed NURBS curve with its control polygon is shown in Fig. 5.3(a). Fig. 5.3(b) shows the bounding box constructed from the convex hull of the bezier segments of the NURBS curve. The solid edges of the convex hulls contribute towards constructing the bounding box while the dashed edges are removed.

The degenerate case when the bounding box is a complex polygon due to self-intersection is handled as shown in Fig. 5.4. One of the two intersecting convex hulls is updated by a corner cutting process. The corner cutting process essentially utilizes the

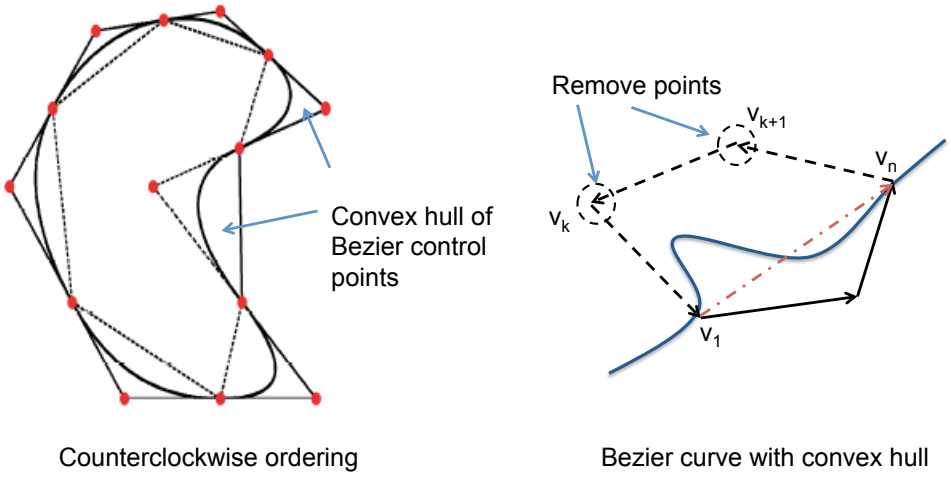
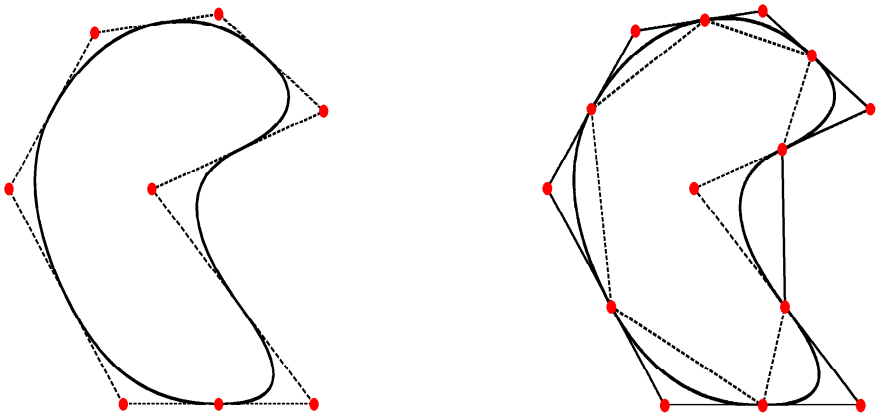


Figure 5.2. Bounding box construction details.



(a) (b)

Figure 5.3. Construction of bounding box of closed NURBS curve. (a) Closed NURBS curve with control polygon, (b) Bounding box constructed from Bezier control polygons.

property of Bezier curves that by adding control points to control polygon, the control polygon approaches the Bezier curve. Thus, the control points of the Bezier curve

are updated by adding new control points such that the corresponding Bezier curve is unaltered and the control points move closer to the curve eliminating the intersection of the convex hulls. Let the Bezier curve be $C_p(u) = \sum_{i=0}^{p} B_{i,p}(u)\mathbf{P_i}$ where $B_{i,p}$ are

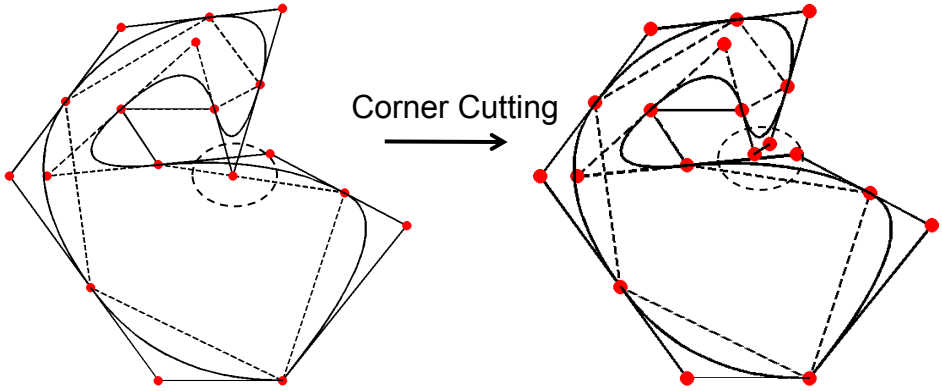


Figure 5.4. Handling degenerate case in bounding box construction.

the basis functions and P_i are the original control points. Now, using corner cutting process, an additional control point is added such that $C_{p+1}(u) = \sum_{i=0}^{p+1} B_{i,p}(u)\mathbf{Q_i}$, where Q_i are the new control points. For C_{p+1} to be same as C_p, the following relation must hold

$$Q_i = (1 - \alpha_i)P_i + \alpha_i P_{i-1} \tag{5.3}$$

where $\alpha_i = \frac{i}{p+1}$ and $i = 0, \ldots, p+1$

5.3.2 Bounding Box for Bounded Solids

The bounding surface is assumed to comprise of either NURBS patches or Bezier patches. In the algebraic distance field construction process, NURBS patches are decomposed into Bezier patches and convex polyhedron is constructed for each Bezier patch. Similar to the two-dimensional case, an assumption on ordering of control points is enforced such that normal of each patch points to outside of the solid.

An assumption of ordering is also enforced on each face of the convex hull such that the vertices are counter-clockwise. Conceptually, bounding box for solids is an extension of the two-dimensional case. Thus, the faces of each convex hull that lies inside the bounding surface are removed and a bounding box is constructed out of all the faces lying on or in the exterior of the bounding surface. However, in terms of implementation, it is non-trivial to extend the algorithm from two-dimensions to three-dimensions. The bounding box algorithm for bounded solids is illustrated in Fig. 5.5. It is assumed that the Bezier patches and their respective convex hulls are precomputed during one time initialization of Algebraic Level Set.

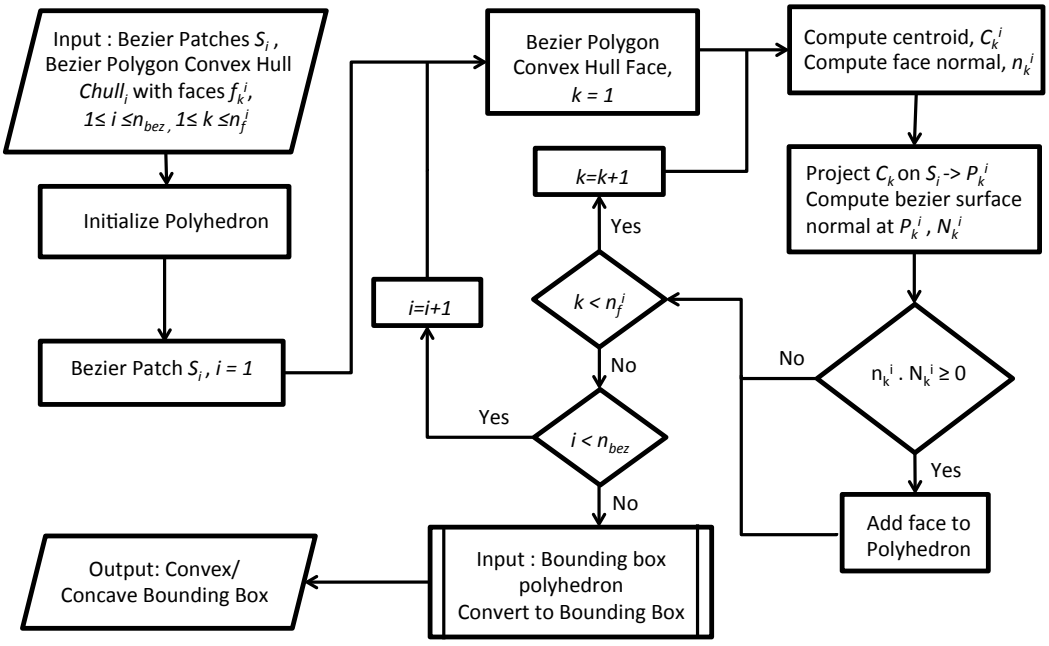


Figure 5.5. Bounding box Algorithm.

Populate Bounding Box Polyhedron from Bezier Convex Hulls

The first step in constructing the bounding box is to identify faces of convex hulls of Bezier patches that lie outside the parametric geometry and then, populate the bounding box polyhedron with these faces. Consider the convex hull of the ith Bezier patch with faces $f_1^i, f_2^i, \cdots, f_k^i, \cdots, f_{n_f^i}^i$. For each face f_k^i, the centroid C_k^i is projected onto the patch. At the projected point P_k^i, the surface normal N_k^i is computed. Let n_k^i be the normal of the face f_k^i. Then, the face lies outside the bounding surface if $n_k^i.N_k^i \geq 0$. This face is added to the bounding box polyhedron.

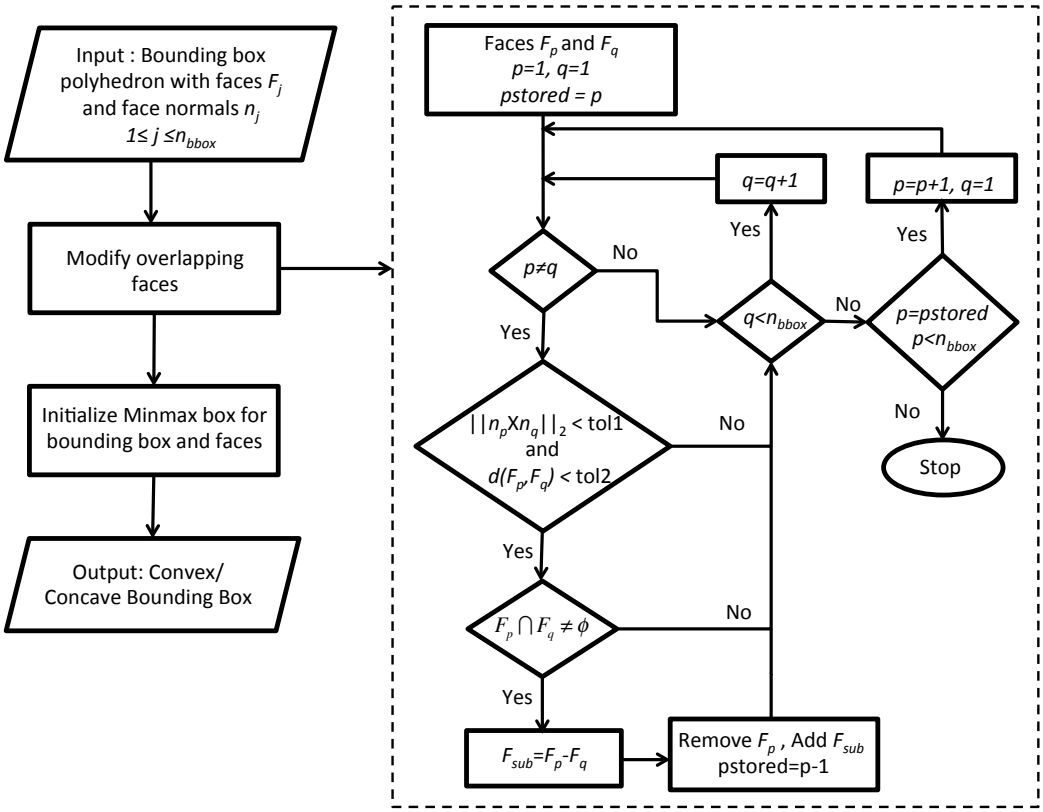


Figure 5.6. Algorithm to convert bounding box polyhedron into a valid bounding box.

Convert Bounding Box Polyhedron into a Valid Bounding Box

The polyhedron populated from convex hull faces of Bezier patch may contain overlapping or intersecting faces. In this step, such faces are identified and modified to get a valid polyhedron. A valid polyhedron, in this work, refers to a polyhedron with faces that only share vertices and edges with other faces of the polyhedron. The algorithm for this process is outlined in Fig. 5.6. Faces are queried pairwise (F_p, F_q) for intersection checks. Intersection checks on pair of non-coplanar faces are eliminated by checking for coplanarity of the face pair. This elimination is accomplished by checking the following condition

$$\| n_p \times n_q \|_2 < \text{tol1} \text{ and } |d(F_p, F_q)| < \text{tol2} \tag{5.4}$$

where n_p and n_q are normals of faces F_p and F_q respectively and $d(F_p, F_q)$ is the distance between the two faces. Tolerances tol1 and tol2 are chosen to check for the coplanarity conditions. The intersection checks on the faces are done using a geometry library in Java - JavaGeom [98]. Once the intersecting face pairs are identified, a subtraction operation is performed $F_{sub} = F_p - F_q$ using the same library. A set of subtracted faces F_{sub} is obtained. Face F_p is then removed from the polyhedron and the new faces in set F_{sub} are added to the polyhedron. This process is continued till all intersecting faces are modified to get a valid polyhedron. Further, minmax boxes made out of the minimium and maximum (x, y, z) coordinates are constructed for the bounding box and its faces for operations downstream. Bounding box construction is illustrated for a torus geometry in Fig. 5.7.

5.4 Signed Distance Field in the Bounding Box of NURBS Curve

The sign in the bounding box is calculated by one of the standard point in polygon tests [99,100]. These include but are not limited to Parity test, Winding number test, Orientation based test.

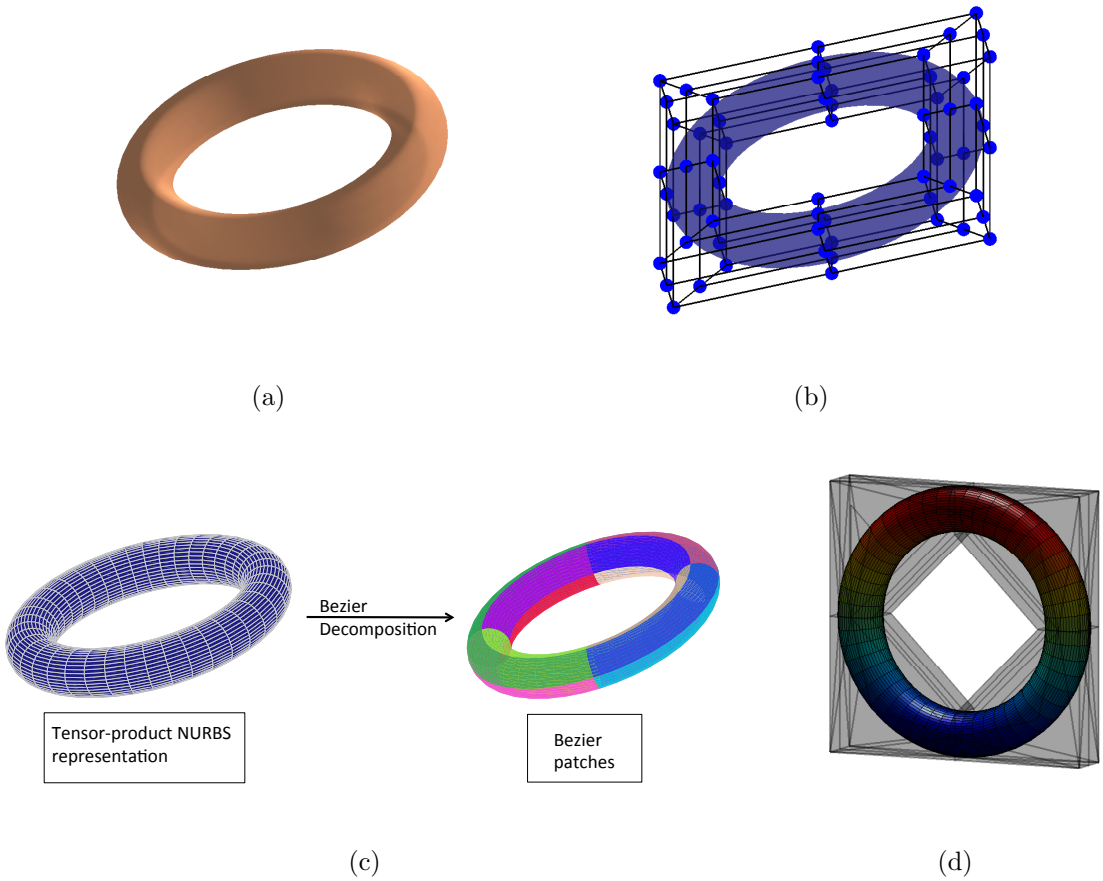


Figure 5.7. Construction of bounding box of bounded solid. (a) Torus NURBS bounded surface, (b) Torus control polygon, (c) Bezier decomposed torus surface, (d) Bounding box for torus.

5.5 Signed Distance Field in Curved Polygon-Bezier Regions

Given a Bezier curve $f = 0$, a convex hull is obtained from the associated control point polygon of the Bezier curve. The Bezier curve divides the convex hull into two regions, each of which is referred to as Polygon-Bezier region in this work. The polygon-Bezier region is defined as a region bounded by a Bezier curve and a set of linear edges (see Fig. 5.8). Let $f \geq 0$ and $-f \geq 0$ be the two half spaces formed by the Bezier curve. Let $\phi \geq 0$ be the region inside the control polygon. Then, $\phi_1 \geq 0$

and $\phi_2 \geq 0$ are the two polygon-Bezier regions obtained by the set-theoretic boolean operations, $\phi_1 = \phi \bigcap f$ and $\phi_2 = \phi \bigcap -f$. The distance fields $\phi_1 \geq 0$ and $\phi_2 \geq 0$ are constructed during the process of the distance field construction of the Bezier curve. In order to obtain the sign of the distance field of NURBS curve, it is required to store the signed field of the polygon-Bezier region whose linear edges form a part of the bounding box of the NURBS curve. The easiest way to do this is to check which of the two polygon-Bezier regions has zero field value for a point in the convex hull belonging to the bounding box. In other words, a test vertex of the convex hull ϕ that does not lie on the Bezier curve is used to choose the right polygon-Bezier region. The test vertex must lie on the bounding box. Then, the correct polygon-Bezier region (ϕ_1 or ϕ_2) is the one for which the corresponding field value ($\phi_1(x, y)$ or $\phi_2(x, y)$) is zero at this test point. The methodology is same in three-dimensions for computing sign in Polyhedron-Bezier regions of the Bezier surfaces.

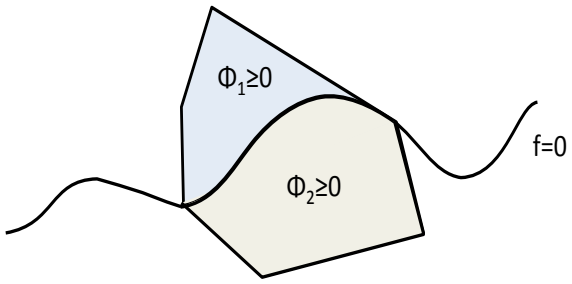


Figure 5.8. Polygon-Bezier regions.

5.6 Special Cases

A discussion of sign calculation for general three-dimensional NURBS geometries was covered in the previous sections. A discussion of techniques to handle special surfaces of three-dimensional CAD geometries is presented in this section.

5.6.1 Planar NURBS Surfaces

The algebraic distance field construction for a planar NURBS surface is discussed in Section 3.2.1. In the case of sign calculation of a geometry with a planar face, the underlying steps are modified as follows

- Bounding Box Construction - The planar face is added to the bounding box polyhedron of the solid.
- Signed Field of Polyhedron-Bezier Regions - This step is skipped for the planar face as a Polyhedron-Bezier region does not exist.
- Sign Calculation - This step proceeds as before without any modification.

5.6.2 Trimmed Parametric Geometries

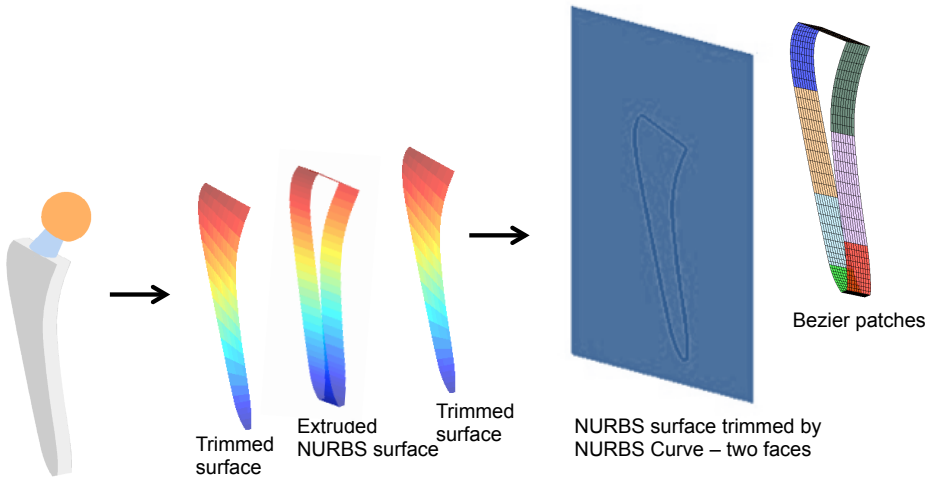
The trimmed NURBS geometries are a special case of B-rep geometries where a parent NURBS surface is trimmed by a NURBS curve resulting in a trimmed surface. The information that can be imported from IGES CAD files about such trimmed geometries comprise of the parent NURBS surface definition and the trimming curve defined in the parametric space of the parent NURBS surface. In this work, planar trimmed surfaces are assumed and handled as follows

- Bounding Box Construction - The bounding box polygon of the trimming curve is initialized. The planar face bounded by this polygon is added to the bounding box polyhedron of the solid.
- Signed Field of Curved Bezier Polyhedron - This step is skipped for the planar face as a Polyhedron-Bezier region does not exist
- Sign Calculation - This step proceeds as before without any modification

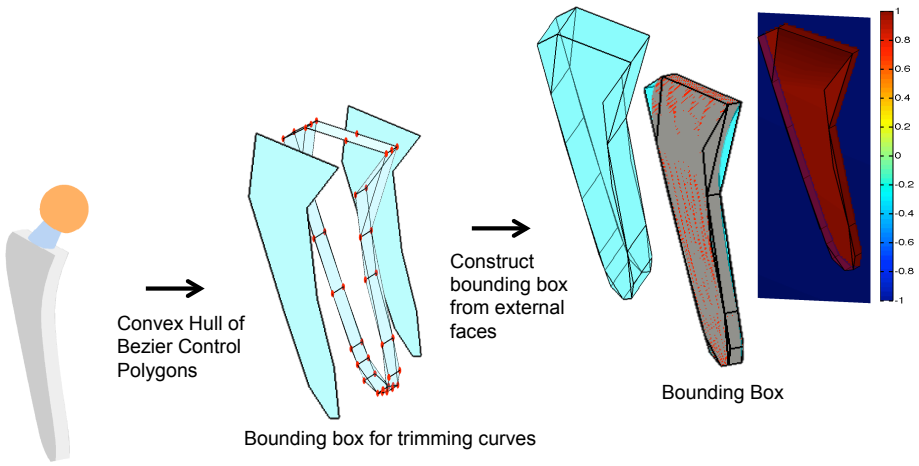
Currently, non-planar trimmed surfaces cannot be handled by the algorithm.

5.7 Illustrative Examples

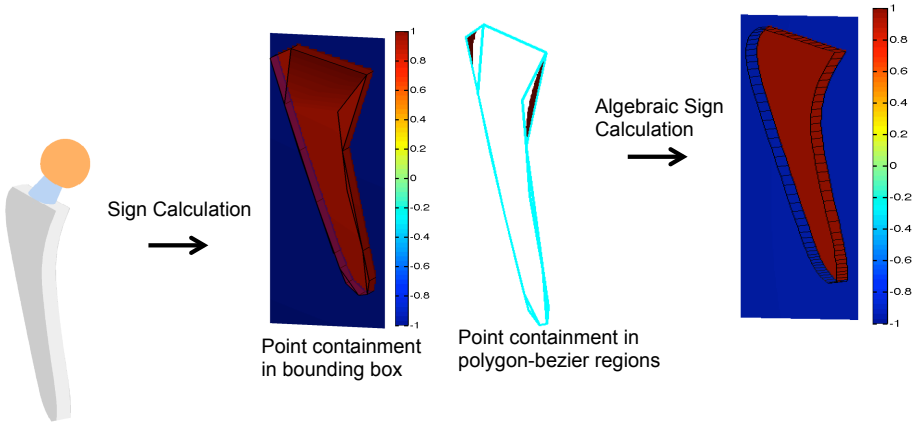
Consider the example of hip prosthesis from Section 3.4. In this section the sign calculation for the hip implant is illustrated through a step-by-step procedure. The hip-implant consists of two trimmed NURBS surfaces and one extruded NURBS surface. First, the extruded NURBS boundary is decomposed into Bezier patches as shown in Fig. 5.9(a). In this process, convex hull is initialized for each Bezier patch. Further, bounding boxes are initialized for the trimming curves of the planar trimmed NURBS surfaces. Next, the bounding box is constructed using the convex hulls of Bezier patches and the planar polygon faces formed by bounding box of the trimming curves (see Fig. 5.9(b)). Finally, point-containment queries in bounding box and polyhedron-Bezier region enable algebraic sign calculation as shown in Fig. 5.9(c). Algebraic sign calculation is also demonstrated for the torus geometry from Fig. 5.7 and the Utah teapot from Section 3.4 (see Fig. 5.10).



(a)

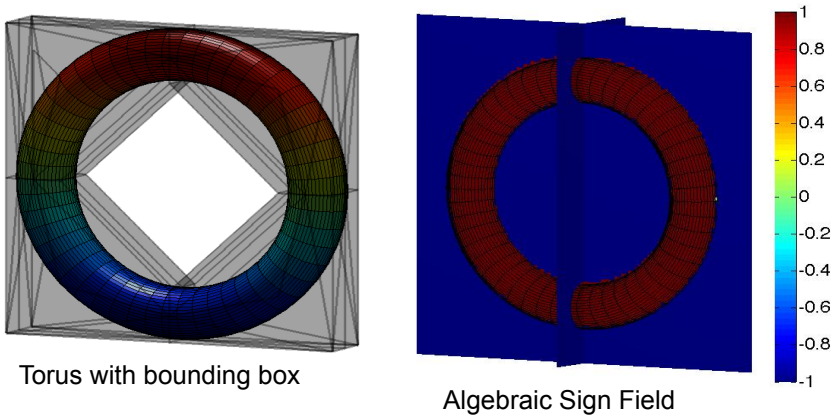


(b)

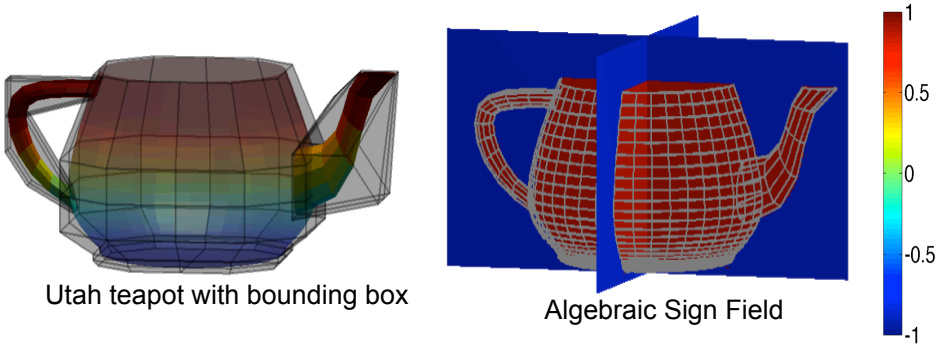


(c)

Figure 5.9. Sign calculation for hip implant. (a) Bezier decomposition, (b) Construction of bounding box, (c) Algebraic sign calculation.



(a)



(b)

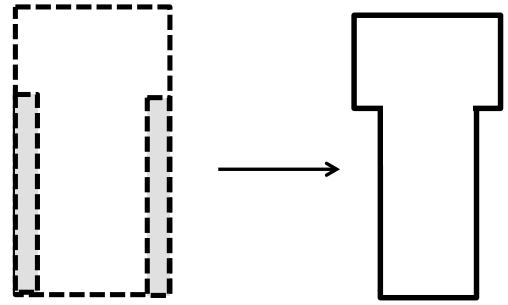
Figure 5.10. Examples of algebraic sign field. (a) Torus, (b) Utah teapot.

6. ALGEBRAIC SOLUTION TO GEOMETRIC COMPOSITIONS

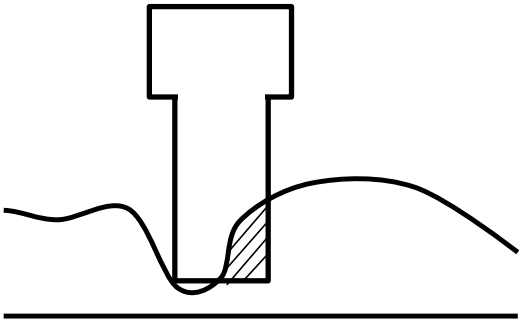
B-rep geometries obtained from solid modeling kernels form the geometrical foundation in a variety of applications in the areas of computer-aided simulation of machining, robotics and automation, computer graphics, animation, computer simulated environments and engineering analysis. Boolean operations on B-rep geometries are fundamental to such applications [8, 101–104].

In existing CAD systems, B-rep CAD models are represented by a collection of trimmed spline patches such as NURBS. Boolean operations on such a representation involves numerical surface-surface intersection operation between the trimmed NURBS patches. Surface-surface intersection algorithms attempt to determine the explicit geometric representation for the intersecting boundary during boolean operations involving surfaces (see Fig. 6.1(a)). Surface intersection calculation is a challenging problem as it is very difficult to satisfy the conflicting goals of accuracy, robustness and efficiency simultaneously using the existing methods.

However, in many applications concerning the analysis of multi-body interactions as in analysis of contact between two bodies, often it is not required to explicitly compute the intersection between interacting B-rep CAD models (see Fig. 6.1(b)); an implicit knowledge of the intersection may be sufficient. For example, multi-body interactions between parts of an assembly occur in manufacturing applications. In order to avoid design errors at the manufacturing or assembly stages, the CAD models of these assemblies are inspected using visualization tools for any interference between the parts [105–108]. Such an application only requires knowledge of whether an intersection between the B-rep CAD models occurs. Further, interference detection is also common in machining applications such as interference free toolpath generation and verification of generated toolpath through computer-aided simulation of machining operations [109–112]. Multi-body interaction problem is also solved in contact



(a)



(b)

Figure 6.1. Classification of boolean operations based on application. (a) Explicit boolean operation : geometry construction through boolean operations require surface-surface intersection computation,(b) Implicit boolean operation: multi-body interaction between tool and part requires only interference detection.

mechanics applications such as simulation of impact or metal forming process [59,86]. In such problems, the aim is to impose the non-penetration condition between the bodies. Such a contact condition is often modeled using a gap function based on signed distance function.

In this chapter, a purely algebraic solution to the problem of implicit boolean operations between NURBS B-rep solids is presented. In the proposed technique, there is no numerical computation of intersection region or the exact signed distance; all implicit boolean compositions in CAD and CAE are addressed algebraically using the algebraic level sets.

6.1 Surface Intersection Problem

Rational polynomial parametric surfaces such as Bezier, B-spline and NURBS surface patches are common in CAD and isogeometric analysis [20, 21, 42]. Given two parametric surfaces $P(u,v) : R^2 \rightarrow R^3$ and $Q(s,t) : R^2 \rightarrow R^3$, the intersection problem is posed as the solution to the equation

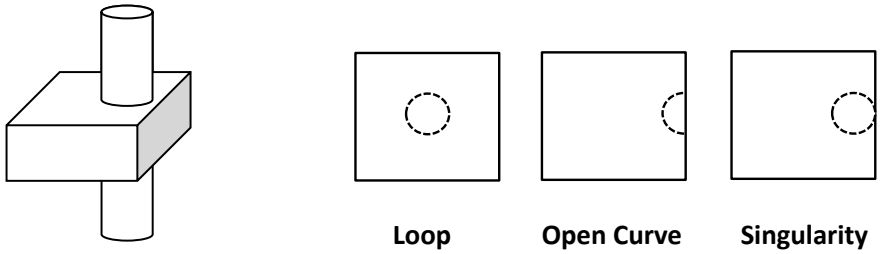
$$P(u,v) = Q(s,t) \tag{6.1}$$

or

$$\begin{aligned} X(u,v) &= X(s,t) \\ Y(u,v) &= Y(s,t) \\ Z(u,v) &= Z(s,t) \end{aligned} \tag{6.2}$$

This is an underconstrained system of three non-linear equations in four unknowns u, v, s, t. Hence, numerical methods are often utilized to solve this problem. The surface intersection problem is further complicated by the presence of open segments, closed loops and self-intersections (singularities) (Fig. 6.2). Surface intersection solution techniques can be broadly classified into four categories: algebraic, subdivision, lattice and marching methods. Hybrid methods have also been developed recently combining features of the above methods.

Algebraic methods : The intersection problem becomes easier to solve if one of the surfaces has an implicit representation of the form $f(x,y,z) = 0$ and the second



(a) (b)

Figure 6.2. Various components of the intersection curve. (a) Simple intersection between a cylinder and a cuboid, (b) Possible intersection curves based on position of cylinder.

has a parametric representation $Q(u, v) : R^2 \rightarrow R^3$. In this case, the intersection problem reduces to finding a solution to the equation

$$f(x(u, v), y(u, v), z(u, v)) = 0 \tag{6.3}$$

This is a non-linear root finding problem which can be solved by using the Newton's method. In general, the CAD geometric representations are parametric in nature and hence the algebraic geometric technique of *implicitization* is used to obtain an implicit representation of one of the parametric surfaces [73, 113–115]. However, the implicitization process for a tensor-product surface of degree (m, n) leads to an implicit representation of degree $2mn$. For instance, a bicubic surface patch results in an equation of degree 18 with 1330 terms! As a result, during numerical root determination, this algebraic method not only lead to large scale computation, but also to significant loss of accuracy, making the method unattractive for practical applications. Therefore, the algebraic approach is generally combined with other methods such as subdivision or marching methods into a hybrid scheme [43, 44]. These approaches will be discussed in the following sections.

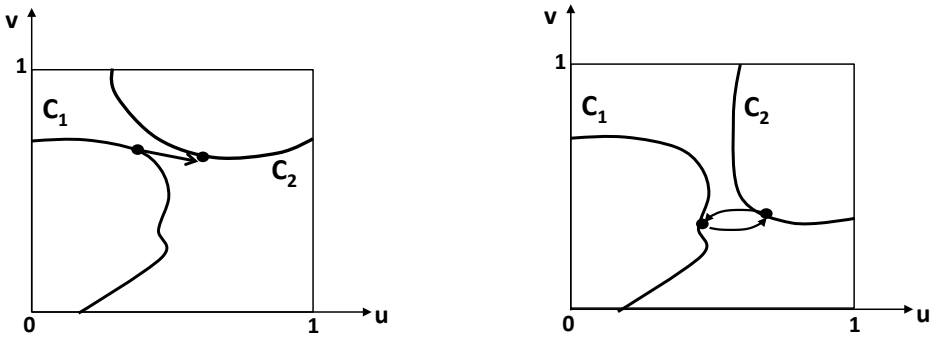
Subdivision methods : These methods are based on a *divide and conquer* strategy [11, 45–48]. The strategy involves recursive subdivision of the geometry until one obtains a simple direct solution (e.g. plane/plane intersection). Accuracy depends on the number of subdivision steps. While these methods converge in the limit, a need for high-precision can lead to data proliferation making the computation slow. Also, correct connectivity of the solution branches in the vicinity of singular or non-singular branches is difficult to guarantee, and small loops maybe missed or extraneous loops may be present in the approximation of the solution [116].

Lattice methods : In this technique, the problem of surface-surface intersection is simplified into a series of low geometric complexity curve-surface intersection problems. This is done by approximating one of the surfaces by a grid of finite number of curves [49]. Intersection of each of these curves with the other surface yields discrete points in space. These points are then connected to get the intersection curve. An initial decision of grid resolution may lead to incorrect connectivity or missing features of the solution. Hence, these techniques suffer from robustness problems in getting the correct intersection topology.

Marching methods : These methods trace each branch of the intersection curve using information of the local geometry [50–52]. They require starting points for each branch of the curve. Most of the algorithms are based on local geometry coupled with quasi-Newton methods for tracing the curve. This technique suffers from convergence issues as well as choice of step size to prevent component jumping and looping (Fig. 6.3). Most algorithms use conservative step size that makes them slow.

6.1.1 Challenges with Current Approaches

In general, in the above described surface intersection techniques, it is challenging to *simultaneously* achieve goals of accuracy, robustness and efficiency. As discussed above, greater accuracy in general is associated with greater data proliferation and



(a) (b)

Figure 6.3. Issues with currently popular marching method illustrated by two branches of intersection curve in paramteric space of one of the surfaces. (a) Component jumping or straying, (b) Looping.

computational cost. The impact of the goals of accuracy, robustness and efficiency on CAD as well as CAE phases and a summary of the extent to which these goals are met by the current methods is discussed below.

Accuracy : Accuracy of intersections is important for engineering applications. Numerical inaccuracies may lead to detection of wrong topology of solution (Fig. 6.4). The algebraic complexity of the intersection curve as described in algebraic methods makes it very difficult if not impossible to describe the curve exactly. Due to the high polynomial degree of the equation to be solved (Eq. (6.3)), a closed form expression for the intersection curve is, in general, not possible. Consider two bicubic surfaces $P(u, v)$ and $Q(s, t)$. Let the implicitized equation of $P(u, v)$ be $f(x, y, z)$, which will be of degree 18 consisting of 1330 monomials. Then, the intersection of the two bicubic patches is given by

$$f(Q(s, t)) = 0 \tag{6.4}$$

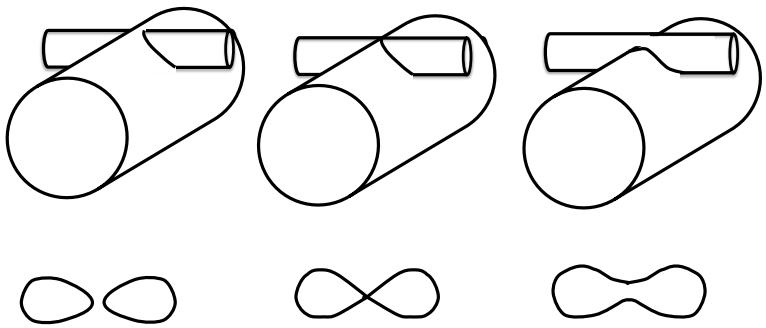


Figure 6.4. Three possibilities for topology of intersection curve between two cylinders with different diameters [117].

This is a polynomial of degree $(54, 54)$ in the variables (s, t) consisting of 3025 monomials and describing an algebraic curve of total degree 108. Hence, the intersection of simple surfaces can lead to a very complex intersection curve. Nearly all surface intersection techniques are based on approximation schemes such as subdivision or polytope approximation, or grid approximation in lattice methods, and hence, lead to inexact geometry. Even algebraic methods need to solve the algebraic equation numerically, and that leads to inaccuracy in the solution. Hence, the current methods do not yield an exact geometry corresponding to the intersection.

Robustness : The intersection techniques need to detect all curve branches, loops and singularities to ensure robustness. However, robustness is affected by step size in marching methods or number of subdivision steps or other numerical choices. Thus, in general, robustness is not automatic but dependent on heuristic numerical parameter choices.

Efficiency : The intersection determination need to be efficient as they are recurring operations in CAD/CAE. However, in most numerical schemes, the goal of efficiency is often in conflict with the goal of accuracy. For instance, very high-precision subdivi-

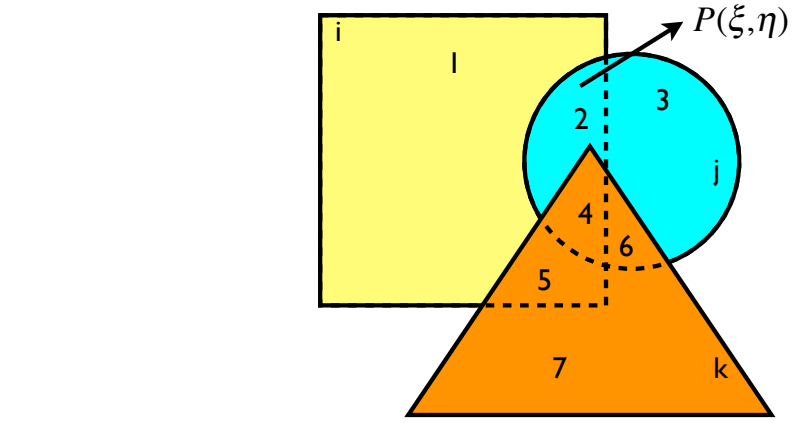
sion leads to data proliferation making the scheme inefficient. Efficiency of marching methods is governed by step size that also controls the accuracy of the solution. Since the algebraic complexity of the intersection curve determines the required step size, the nature of the intersection geometry (and therefore the required robustness) has a strong impact on the efficiency; sacrificing accuracy to improve efficiency may also sacrifice robustness. Hence, it is very difficult to satisfy the conflicting goals of accuracy, robustness and efficiency simultaneously using existing methods.

6.2 Algebraic Boolean Operations

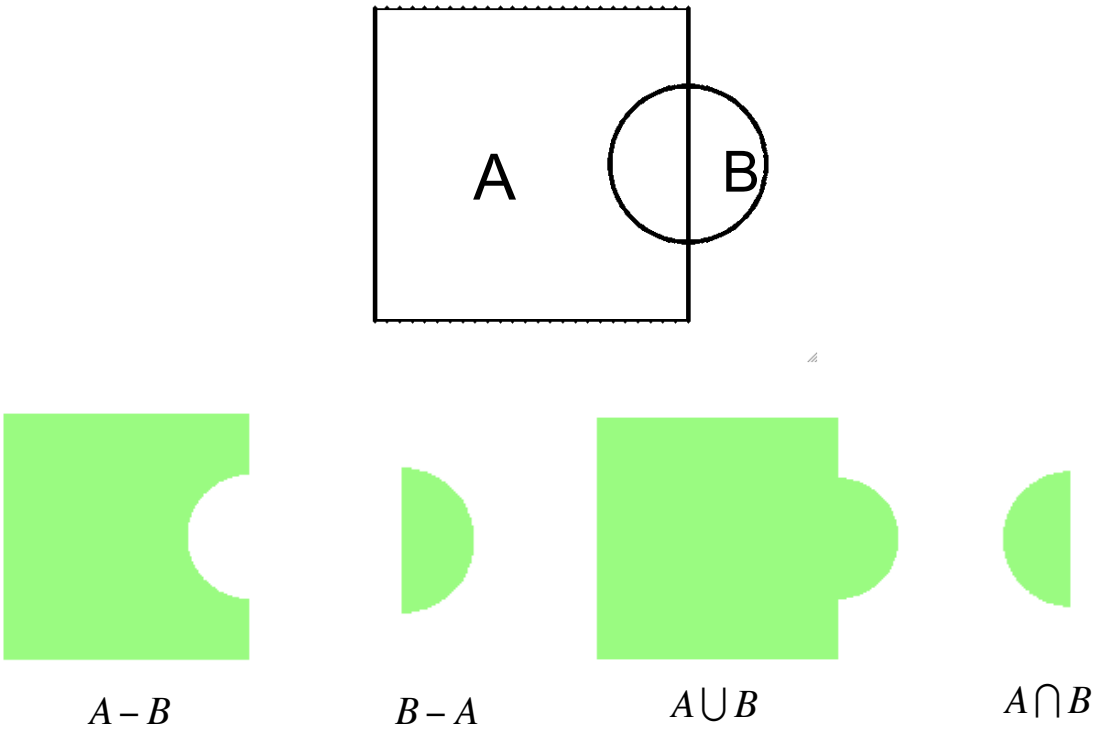
An algebraic point containment strategy is developed to perform implicit boolean operations without explicit surface intersection operations. Consider n B-rep geometries $\Gamma_1, \ldots, \Gamma_i, \ldots, \Gamma_n$ undergoing boolean operations. Let $s_1, \ldots, s_i, \ldots, s_n$ be their respective algebraic sign field. The interior domain Ω_i of the ith B-rep geometry Γ_i is defined as the region $s_i \geq 0$. The algebraic sign of each B-rep solid is used to perform point classification queries. Pairwise boolean operations are performed by using the point classification queries. Examples of pairwise boolean operations between B-rep geometries Γ_i and Γ_j are as follows

- Boolean Union $(\Omega_i \cup \Omega_j)$: $s_i \geq 0$ or $s_j \geq 0$
- Boolean Intersection $(\Omega_i \cap \Omega_j)$: $s_i \geq 0$ and $s_j \geq 0$
- Boolean Subtraction $(\Omega_i - \Omega_j)$: $s_i \geq 0$ and $s_j \leq 0$

In Fig. 6.5(a), three intersecting B-rep solids are shown. Further, seven regions formed by boolean operations between the three solids are highlighted. At any query point $P(\xi, \eta)$, the algebraic sign field s_i, s_j, s_k of each solid is evaluated. Using these sign fields, the region containing the query point can be identified. Further, boolean operations using the sign fields will give an implicit sign field corresponding to the region of interest (see Fig. 6.5(b)).



(a)



(b)

Figure 6.5. Algebraic boolean operations. (a) Algebraic point classification, (b) Illustrative example of boolean operations between rectangle and circle using algebraic level sets.

6.3 Comparison with Surface Intersection Technique

The algebraic implicit boolean operations are performed using point-wise queries of the non-iterative algebraic level set function. The algebraic nature of the query makes it efficient compared to the complex numerical surface intersection operation. Further, at a query point, the technique gives an accurate point containment check as opposed to the numerical inaccuracies associated with the surface intersection operation. Further, robustness of intersection technique in the presence of singularities is affected by numerical choices of the algorithm. However, the algebraic boolean operation technique is independent of such singularity issues as the query is not computed using the intersection geometry. It is a function of the interacting geometries (see Fig. 6.6).

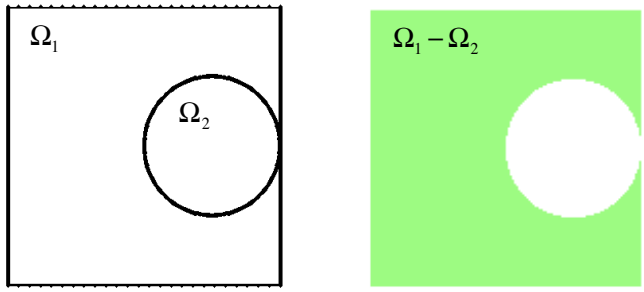
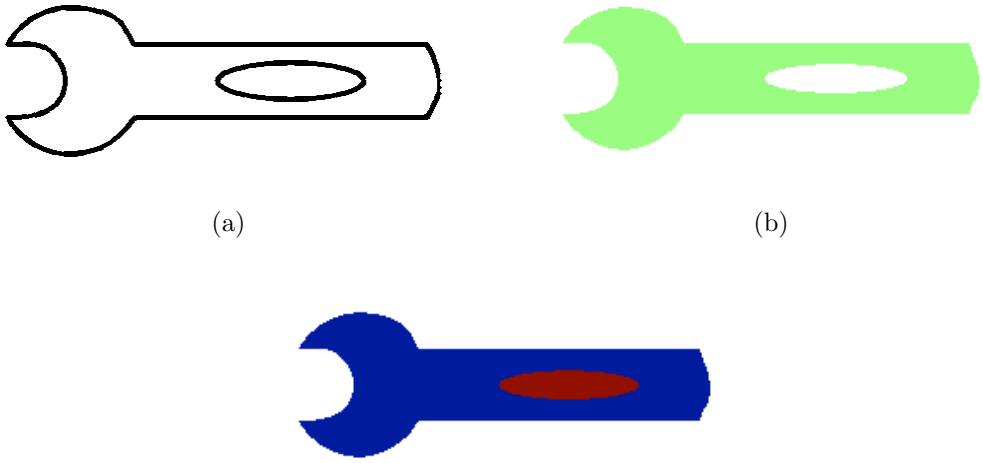


Figure 6.6. Algebraic boolean operation is not affected by the presence of singular intersection.

6.4 Illustrative Examples

Algebraic boolean operation between a wrench and an elliptical hole is shown in Fig. 6.7(a). In this example, the algebraic sign query is used for heterogeneous material modeling. In Fig. 6.7(b), the wrench has an elliptical hole with no material property, while in Fig. 6.7(c), an elliptical inclusion is modeled inside the wrench.



(a) (b)

(c)

Figure 6.7. Heterogeneous material modeling. (a) Wrench and elliptical primitive geometries, (b) Wrench with elliptical hole, (c) Wrench with elliptical inclusion.

A more complex example of multiple algebraic boolean operations is illustrated in Fig. 6.8. These operations result in the geometry of wheel with spokes.

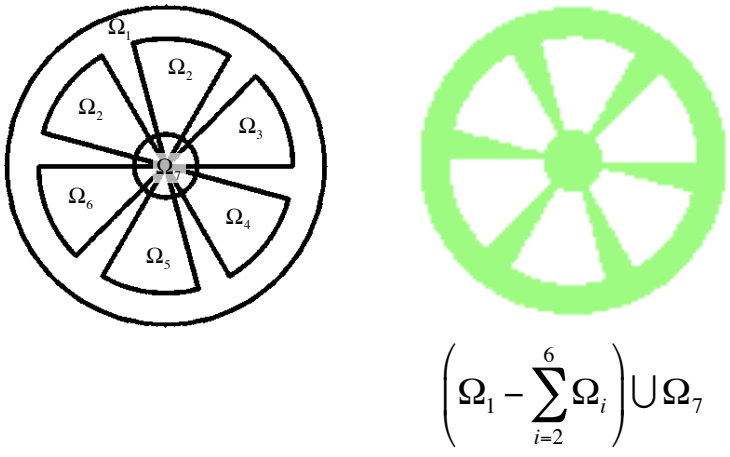


Figure 6.8. Multiple algebraic boolean operations.

Algebraic boolean operations between a cube and cylinder are shown in Fig. 6.9. The topology of the intersection curves with different cylinder positions is captured here.

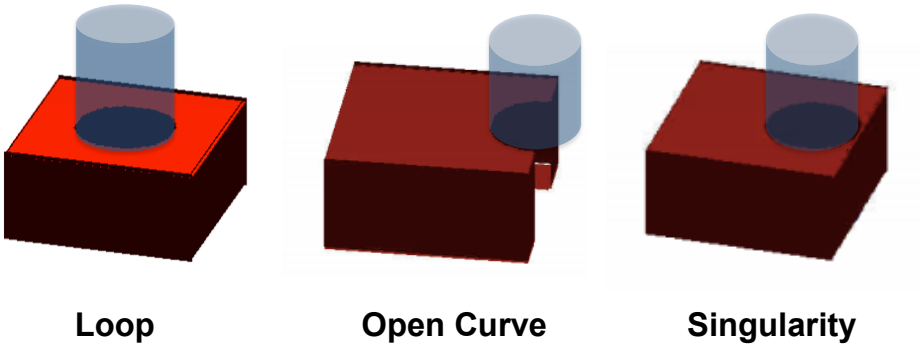


Figure 6.9. Topology of intersection curve as captured by Algebraic boolean operations.

Further, algebraic implicit boolean operations are shown to enable sign calculation in a utah teapot constructed from sign field of three primitive geometries (see Fig. 6.10). No surface intersection operations or stitching is performed to construct the geometry, hence, there is no challenge with maintaining water tightness.

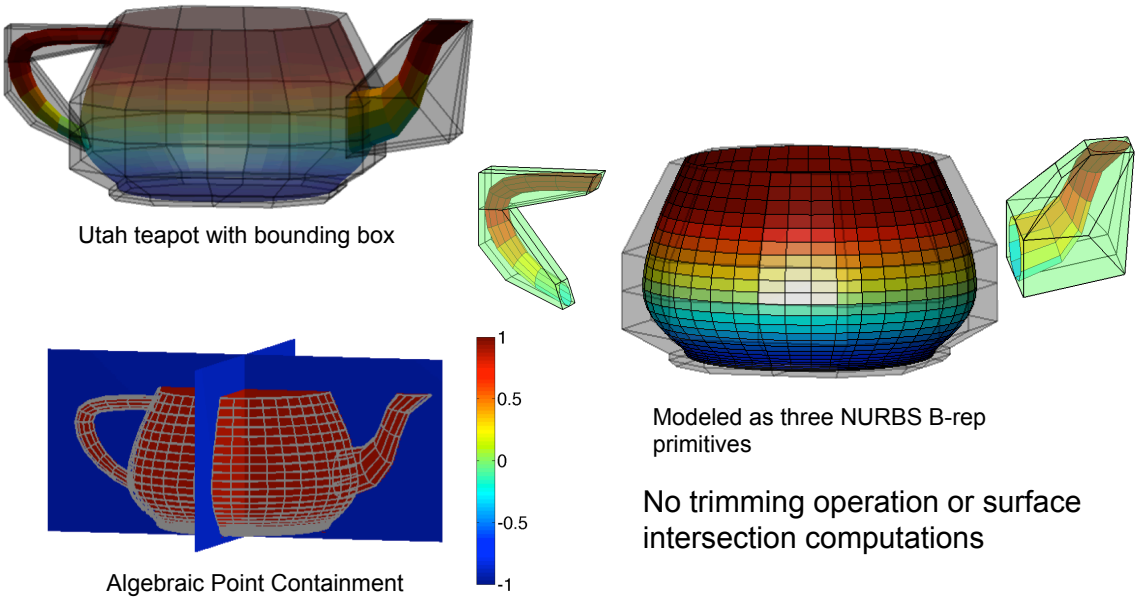


Figure 6.10. Algebraic boolean operations to construct sign field of utah teapot.

7. CAD/CAE INTEGRATION

The problem of CAD/CAE integration faces the challenges of mesh generation problem, capturing geometric features exactly, point containment problem and surface intersection problem. The mesh generation problem is eliminated by meshless approximations such as Kantarovich method and the class of isoparamteric approximations (IGA, HPFC) with trivariate or volumetric CAD geometry representations. However, meshless approximations have difficulty in capturing the geometric features exactly as well as the numerical challenge of point containment and surface intersection problem. On the other hand, isoparametric volumetric approximations can ensure exact geometry and enable point containment. But the major challenge with the latter class of analysis techniques is the need for trivariate analysis ready CAD models. The existing CAD systems cannot provide such a representation, hence, such geometries need to be constructed from scratch and new CAD modeling tools need to be developed.

In order to utilize the current CAD modeling systems, while solving the challenges with CAD/CAE integration, a new analysis technique is proposed that brings together the advantages of the meshless methods and the isoparametric volumetric schemes. The analysis technique developed in this work is based on the HPFC theory as well as its generalization to enriched field modeling developed in the references [1, 2, 118]. Through these techniques, geometry changes in iterative design problems can be localized during analysis by procedurally mimicking the CSG procedure of CAD during analysis [1, 42, 119]. Further, the proposed technique utilizes *Algebraic Level Sets* to enable algebraic solutions to the problems of point containment and surface intersection during CAD/CAE integration. In the proposed strategy, surface intersection operations are eliminated by modeling the primitive geometries or individual components of a multi-component CAD model as enrichments on a trivariate regular

domain. Algebraic level sets of the enrichments are used to enable efficient point containment checks to infer boolean compositions. Algebraic level sets ensure exactness of geometry while eliminating iterative numerical computations. This technique enables direct analysis of complex physical problems without meshing, thus, integrating CAD and CAE. The idea of the proposed technique is constrasted against traditional FEA through a two-dimensional example in Fig. 7.1.

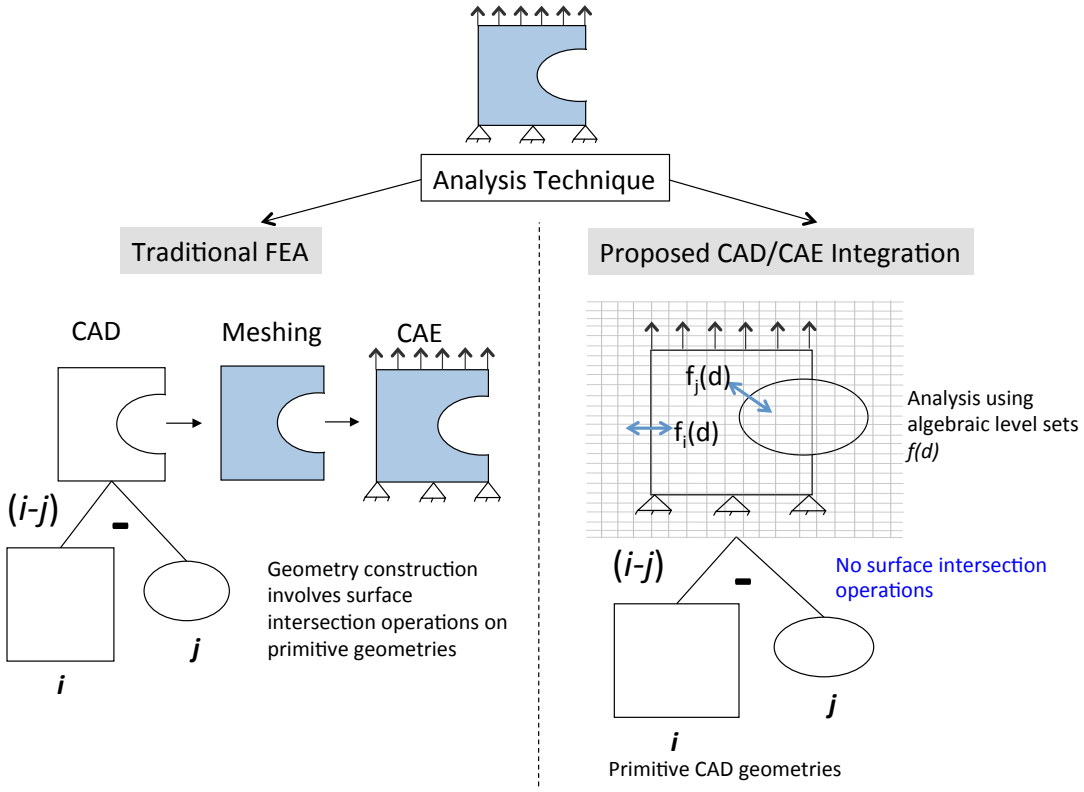


Figure 7.1. Algebraic level sets based isogeometric analysis contrasted against traditional FEA.

7.1 Background

The HPFC theory and enriched field modeling are integral to the analysis technique proposed in this work. Hence, these techniques are briefly reviewed in the following discussion.

7.1.1 Hierarchical Partition of Unity Field Compositions (HPFC) Theory

The HPFC theory [1] describes a design state $\mathscr{D}$ as the product space $\mathscr{D} = (\mathscr{G}, \mathscr{M})$ of geometry $\mathscr{G} \subset \mathbb{H}^1$, material $\mathscr{M} \subset \mathbb{H}^0$ and an appropriate mapping $A : \mathscr{G} \times \mathscr{M} \rightarrow \mathscr{U}$ to behavior $\mathscr{U}$ for analyzing the design state. The global design state is constructed by a composition of the primitive states in a manner that is analogous to CSG procedure (Fig. 7.2). At any point $\mathbf{x}$ in the design domain Ω, the following form of the constructed fields is proposed

$$f(\mathbf{x}) = \sum_i w^i_\Omega(\mathbf{x}) f^i(\mathbf{x}) \tag{7.1}$$

such that weights obey partition of unity as follows

$$\begin{aligned} &\sum_i w^i(\mathbf{x}) = \mathbf{1} \\ &0 \leq w^i \leq 1 \end{aligned} \tag{7.2}$$

$$\|w^i\|_{L^\infty(R^d)} \leq C_\infty \tag{7.3}$$

$$\|\nabla w^i\|_{L^\infty(R^d)} \leq \frac{C_G}{diam\Omega^i} \tag{7.4}$$

Such partitions of unity ensure convergence of the global approximations when errors in local approximations are bounded.

7.1.2 Enriched Field Approximations

The ideas developed in [1, 118], were further extended by Tambat and Subbarayan [2] to enrich the approximation constructed in the previous section with

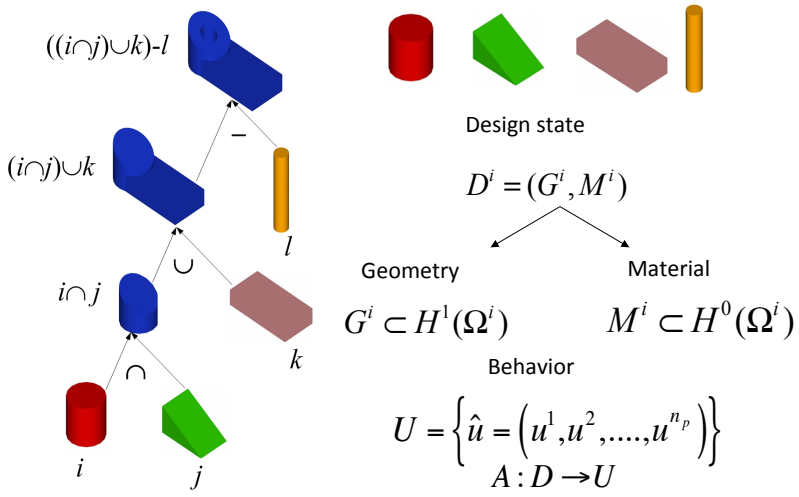


Figure 7.2. The Hierarchical Partition of Unity Field Compositions theory describes a complex design state consisting of a triad of functions (belonging to an appropriate function space and) approximating geometry, material and behavior through compositions of functions defined on the primitive entities in a manner analogous to the constructive solid geometry procedure.

known behavior or material property on surfaces (or curves or even vertices) within the domain.

Let the geometry of the underlying domain Ω be defined as $S_\Omega(p,q,r)$ and the behavioral field as $f_\Omega(\mathbf{x}(p,q,r))$. Let the geometry of the lower-dimensional enriching geometry be parametrically defined as $C_\Gamma(s,t)$ and the behavioral field as $f^b_\Gamma(\mathbf{x}(p,q,r) \rightarrow (s,t))$. Then, the global approximation field $f(\mathbf{x})$ at any point $\mathbf{x}$ in the domain is given by the following weighted composition (see Fig. 7.3):

$$f(\mathbf{x}) = w_\Omega(\mathbf{x}) f_\Omega(\mathbf{x}\,(p,q,r)) + w^b_\Omega(\mathbf{x}) f^b_\Gamma(\mathbf{x}(p,q,r) \rightarrow (s,t)) \tag{7.5}$$

Here, the weights w_Ω and w^b_Ω obey partition of unity property [53] such that $w_\Omega + w^b_\Omega = 1$. Since the influence of the enriching field must decay with distance from the enriching geometrical entity, the weight fields are required to be monotonically

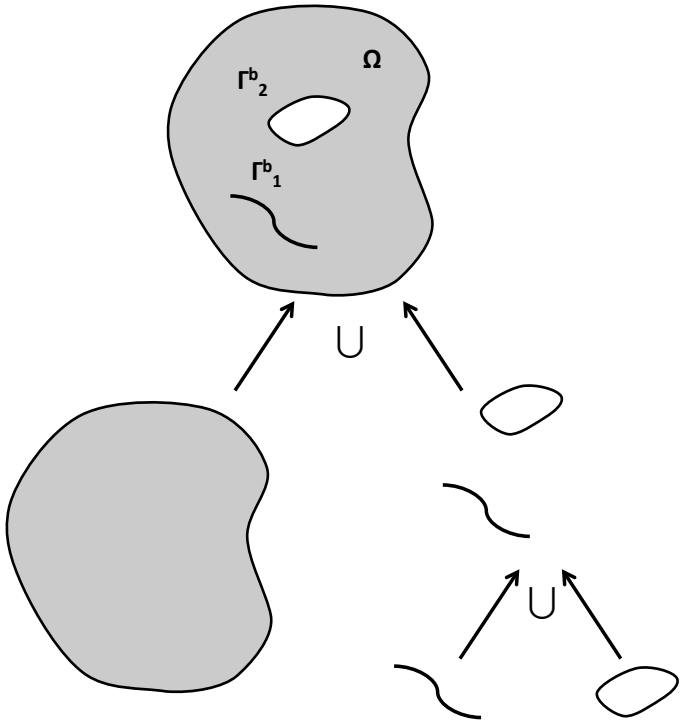


Figure 7.3. Global approximation is constructed by hierarchical composition of lower order primitives with the higher order primitive.

decreasing functions of distance. Some possible functions for weight field w^b_Ω include the exponential function (Eq. (7.6)) and the spline function (Eq. (7.7)) [25]

$$w^b_\Omega = e^{-\left(\frac{d}{d_e}\right)^2} w_\Gamma \tag{7.6}$$

$$w^b_\Omega = \begin{cases} w_\Gamma(1 - \frac{6}{8}(\frac{d}{d_s})^2 + \frac{1}{8}(\frac{d}{d_s})^3) & 0 \leq \frac{d}{d_s} < 1 \\ w_\Gamma(\frac{3}{8}(2 - \frac{d}{d_s})^3) & 1 \leq \frac{d}{d_s} < 2 \\ 0 & \frac{d}{d_s} \geq 2 \end{cases} \tag{7.7}$$

where d is the algebraic distance measure from the enriching geometry Γ, d_e and d_s are scaling factors and w_Γ is the assigned weight value on the enriching geometry Γ. The weight field thus constructed limits the influence of enriching feature over a finite distance.

The approximations both over the domain and on the enriching entities are constructed using NURBS, which are popular in CAD for modeling geometrical surfaces [64]. Also, NURBS basis functions exhibit properties such as partition of unity, local support, domain of influence, smoothness, convex hull and non-negativity that are critical to ensuring convergence of approximations to a known solution. Thus, the behavioral field f_Ω is isoparametrically defined as

$$f_\Omega(\mathbf{x}(p,q,r)) = \sum_I N_I(p,q,r)\hat{\mathbf{u}}_I \tag{7.8}$$

where, $N_I(p,q,r)$ are the NURBS basis functions over the underlying domain and $\hat{\mathbf{u}}_I$ is the field value at the I-th control point of the NURBS geometric domain. The behavioral field f^b_Γ is defined as

$$f^b_\Gamma(\mathbf{x}(p,q,r) \to (s,t)) = \psi(\mathbf{x}) \left(\sum_J N_J(s,t)\hat{\mathbf{v}}_J \right) \tag{7.9}$$

where, $N_J(s,t)$ are the NURBS basis functions over the lower order primitive and $\hat{\mathbf{v}}_J$ is the field value at the J-th control point of the enriching geometry, ψ is an enrichment function that provides the required spatial modulation of the field u to together achieve the desired enrichment and $\mathbf{x}(p,q,r) \to (s,t)$ is a projection from the point $\mathbf{x}$ on the underlying domain to the closest point $P(s,t)$ on the lower order enriching primitive.

7.2 Overview of Numerical Scheme

In this work, direct analysis on B-rep CAD models is enabled using the HPFC theory and enriched field approximations. Approximations are constructed on an underlying domain, that does not conform with the geometry of the B-rep CAD model, similar to the meshless approximations distinct from geometry. The HPFC theory is applied to construct analysis approximations by blending behavior on the boundaries of free form CAD models with the approximations on the underlying domain. The boundary conditions on the geometric model are represented as lower order primitives

(Fig. 7.4). The fields on these lower order primitives are composed with the approximations built on the underlying domain through weight fields constructed using the algebraic level sets. Further, the algebraic level sets also enable algebriac point containment queries and compositions during analysis. Geometry-based analysis of free form CAD models is demonstrated using linear elasticity problems.

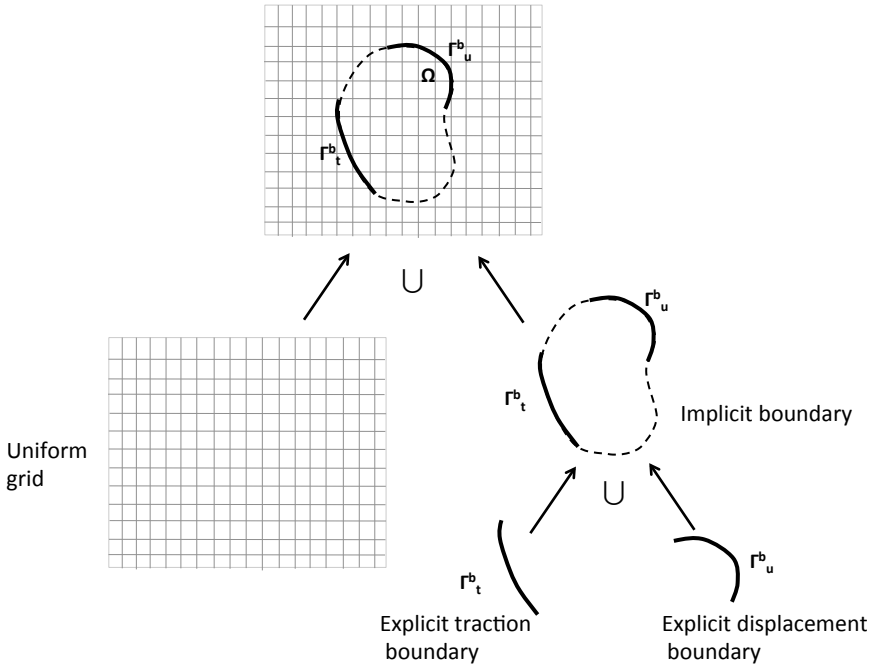


Figure 7.4. Global approximation is constructed by hierarchical composition of explicit boundary conditions with approximations on a uniform grid.

7.3 Formulation of Discretized Solution System

Consider the domain Ω with boundary Γ. The boundary Γ consists of the displacement and traction boundary conditions. The corresponding dirichlet boundary Γ_u, or traction boundary Γ_t is modeled explicitly as lower order NURBS primitive. In general, the total potential energy for linear elastic systems is given by

$$\Pi^*(\mathbf{u}) = \int_\Omega \frac{1}{2}\sigma_{ij}\varepsilon_{ij}\, d\Omega - \int_\Omega \hat{b}_i u_i\, d\Omega - \int_{\Gamma_t} \hat{t}_i u_i\, d\Gamma \tag{7.10}$$

where $\sigma_{ij} = D_{ijkl}\varepsilon_{kl}$ is the generalized Hookes law relating stress σ_{ij} to strain ε_{kl}, and D_{ijkl} is the fourth order elasticity tensor. The displacement field u_i is approximated by the weighted composition between the domain and the boundary as described in Eq. (7.5).

$$u_i = w_i^d u_i^d + w_i^b u_i^b = \tilde{u}_i^d + \tilde{u}_i^b \tag{7.11}$$

such that $w_i^d + w_i^b = 1$. Superscripts d and b correspond to domain and boundary respectively. The influence of the boundary on the underlying domain is modeled using weight field w_i^b constructed as a monotonically decreasing function of the algebraic distance field. The strain and stress components in terms of the domain and boundary displacements are as follows

$$\varepsilon_{ij} = \frac{u_{i,j} + u_{j,i}}{2} = \frac{(w^d u^d)_{i,j} + (w^d u^d)_{j,i}}{2} + \frac{(w^b u^b)_{i,j} + (w^b u^b)_{j,i}}{2} = \tilde{\varepsilon}_{ij}^d + \tilde{\varepsilon}_{ij}^b \tag{7.12}$$

$$\sigma_{ij} = D_{ijkl}\varepsilon_{kl} = D_{ijkl}\tilde{\varepsilon}_{ij}^d + D_{ijkl}\tilde{\varepsilon}_{ij}^b \tag{7.13}$$

The total potential energy is then written as a function of u^d and u^b.

$$\Pi^*(\mathbf{u}^d, \mathbf{u}^b) = \int_\Omega \frac{1}{2}\sigma_{ij}\tilde{\varepsilon}_{ij}^d \, d\Omega + \int_\Omega \frac{1}{2}\sigma_{ij}\tilde{\varepsilon}_{ij}^b \, d\Omega - \int_\Omega \hat{b}_i\tilde{u}_i^d \, d\Omega - \int_\Omega \hat{b}_i\tilde{u}_i^b \, d\Omega - \int_{\Gamma_t} \hat{t}_i\tilde{u}_i^d \, d\Gamma - \int_{\Gamma_t} \hat{t}_i\tilde{u}_i^b \, d\Gamma \tag{7.14}$$

The stationarity of the total potential energy with respect to u^d and u^b yields the following equilibrium conditions.

$$\delta\Pi^*_{\mathbf{u}^d} = \int_\Omega \sigma_{ij}\tilde{\varepsilon}_{ij}^d \, d\Omega - \int_\Omega \hat{b}_i\tilde{u}_i^d \, d\Omega - \int_{\Gamma_t} \hat{t}_i\tilde{u}_i^d \, d\Gamma$$

$$\delta\Pi^*_{\mathbf{u}^b} = \int_\Omega \sigma_{ij}\tilde{\varepsilon}_{ij}^b \, d\Omega - \int_\Omega \hat{b}_i\tilde{u}_i^b \, d\Omega - \int_{\Gamma_t} \hat{t}_i\tilde{u}_i^b \, d\Gamma \tag{7.15}$$

The displacement fields of the higher and lower order primitives are next discretized using NURBS basis functions as shown in Eqs.

$$\tilde{\mathbf{u}}^d = \mathbf{w}^d \sum_I N_I(p,q,r)\mathbf{u}_I^d = \sum_I \tilde{N}_I(p,q,r)\mathbf{u}_I^d \tag{7.16}$$

$$\tilde{\mathbf{u}}^b(\mathbf{x}(p,q,r) \to (s,t)) = \mathbf{w}^b \sum_J N_J(s,t)\mathbf{u}_J^b = \sum_J \tilde{N}_J(s,t)\mathbf{u}_J^d \tag{7.17}$$

After substituting the approximating fields into the equilibrium equations, the following matrix system is obtained:

$$\begin{bmatrix} K^{dd} & K^{db} \\ K^{db^T} & K^{bb} \end{bmatrix} \begin{Bmatrix} \hat{u}^d \\ \hat{u}^b \end{Bmatrix} = \begin{Bmatrix} g^d \\ g^b \end{Bmatrix} + \begin{Bmatrix} h^d \\ h^b \end{Bmatrix} \tag{7.18}$$

where, the stiffness matrices are constructed as

$$\begin{aligned} [K^{dd}] &= \int_\Omega (\tilde{B}^d)^T D \tilde{B}^d \, d\Omega \\ [K^{bb}] &= \int_\Omega (\tilde{B}^b)^T D \tilde{B}^b \, d\Omega \\ [K^{db}] &= \int_\Omega (\tilde{B}^d)^T D \tilde{B}^b \, d\Omega \end{aligned} \tag{7.19}$$

,the work equivalent forces are determined by integration of tractions on primitive boundaries

$$\begin{aligned} \{g^d\} &= \int_{\Gamma_t^d} (\tilde{N}^d)^T \hat{t} \, d\Gamma \\ \{g^b\} &= \int_{\Gamma_t^b} (\tilde{N}^b)^T \hat{t} \, d\Gamma \end{aligned} \tag{7.20}$$

and the body forces are determined as follows

$$\begin{aligned} \{h^d\} &= \int_\Omega (\tilde{N}^d)^T \hat{d} \, d\Omega \\ \{h^b\} &= \int_\Omega (\tilde{N}^d)^T \hat{b} \, d\Omega \end{aligned} \tag{7.21}$$

If Γ_b is a traction boundary, then the matrix system reduces to

$$\begin{bmatrix} K^{dd} & K^{db} \\ K^{db^T} & K^{bb} \end{bmatrix} \begin{Bmatrix} \hat{u}^d \\ \hat{u}^b \end{Bmatrix} = \begin{Bmatrix} 0 \\ g^b \end{Bmatrix} + \begin{Bmatrix} h^d \\ h^b \end{Bmatrix} \tag{7.22}$$

since $w^d = 0$ on Γ_t^d

If Γ_b is a dirichlet boundary, then the matrix system reduces to

$$\begin{bmatrix} K^{dd} \end{bmatrix} \begin{Bmatrix} \hat{u}^d \end{Bmatrix} = \begin{Bmatrix} g^d \end{Bmatrix} + \begin{Bmatrix} h^d \end{Bmatrix} - \begin{bmatrix} K^{db} \end{bmatrix} \begin{Bmatrix} \hat{u}^b \end{Bmatrix} \tag{7.23}$$

since $\hat{u}^b$ are known displacements on the boundary.

For more than two primitives, the solution system is formed by a pairwise interaction between the primitives following an approach similar to that described in reference [1].

7.4 Algebraic Level Sets in Analysis

The information required for engineering analysis and implicit boolean operations comprises of the following point-wise queries:

- Classification as inside or outside
- Field values (material or behavior), f^i
- Field derivatives, $\frac{\partial f^i}{\partial x_j}$

The B-rep CAD representation is insufficient to provide such a volumetric information as its information exists only on the boundaries. However, a signed distance representation of a B-rep solid is sufficient to evaluate the above mentioned queries.

The algebraic level sets proposed in this work are non-iterative approximations to signed distance function. Thus, algebraic level sets enable the following operations in the proposed analysis technique (see Fig. 7.5).

- *Enable construction of weight fields* - Capture influence of boundaries on underlying domain through monotonic distance measure.
- *Enable point classification* - To identify quadrature points inside primitive, assign material properties and enable algebraic boolean operations.

7.5 Numerical Examples

In this section, the analysis technique described in the previous section is applied to elasticity problems with known analytical solutions to validate the scheme. In all of the examples, meshless analysis of B-rep CAD models is demonstrated. It is assumed that the geometric models are represented only by NURBS boundaries and that a parametric description of the domain of analysis is not available. A background grid is constructed in each problem using a rectangular NURBS primitive. The background grid does not conform to the domain of the geometric model. Boundary conditions

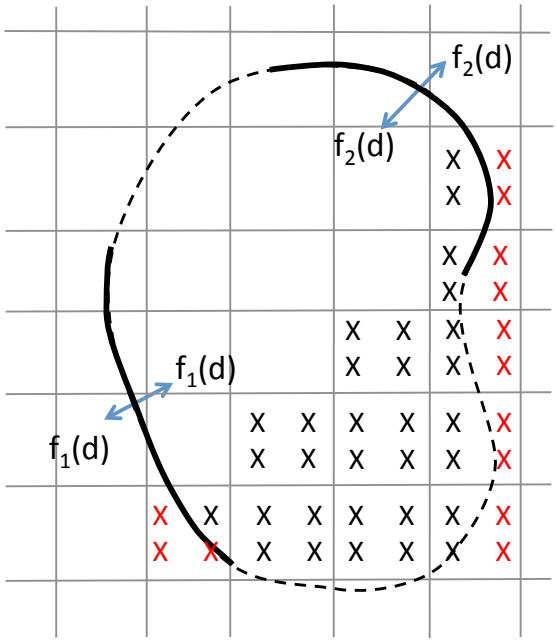


Figure 7.5. Role of algebraic level sets in CAD/CAE integration.

are defined on the explicitly modeled lower order NURBS primitives. Weight fields are constructed using the exponential function given in Eq. (7.6). Algebraic distances are computed from the boundaries of the lower order primitives. Material assignment is based on point containment checks using the signed field from the boundaries of the geometric model. Thus, the proposed technique allows direct analysis on B-rep CAD models without tedious mesh generation.

7.5.1 Plate with an Elliptical Hole

An elastic plate with an elliptical hole under uniform tension is known to have an analytical stress concentration factor that is dependent on the ratio of the major to minor diameter of the ellipse. The geometry and boundary conditions for this problem are shown in Fig. 7.6(a). The modulus of elasticity of the plate is assumed to be 100 units and the Poissons ratio is 0.3. A background uniform grid is constructed as a rectangular ($2lX2w$) NURBS primitive with the dimensions indicated in Fig. 7.6.

The two traction boundaries are modeled explicitly as lower order NURBS primitives. The geometric model and the background mesh are independent of each other and are composed hierarchically as shown in Fig. 7.6(c). Point containment checks in the B-rep CAD model of the plate with elliptical hole were used to eliminate quadrature points on the mesh lying outside the material domain.

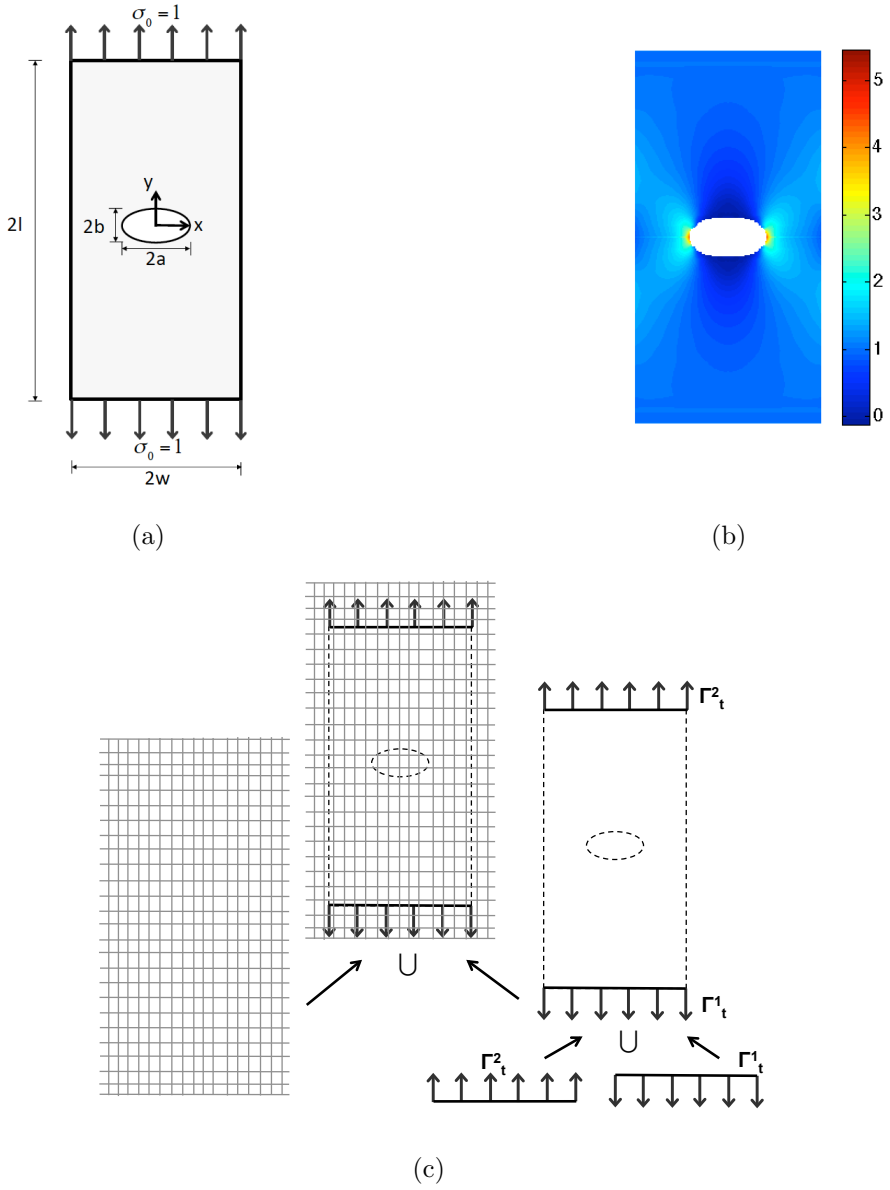
The analytical stress concentration factor for elliptical hole with $\frac{a}{b} = 2$ accounting for the finite plate width is 5.9 [120]. The stress plot (σ_{yy}) is shown in Fig. 7.6(b) for a uniform control point grid spacing of h = 0.03 units and local refinement around the hole. The stress concentration factor obtained was 5.5292. Further local refinement resulted in an improved stress concentration factor of 5.7348.

7.5.2 Curved Cantilever under End Loading

In this example, a curved cantilever beam is subjected to tip loading as shown in Fig. 7.7(a). The modulus of elasticity of the beam was assumed to be 100 units and the Poissons ratio was 0.3. The analytical stress field solution is given by the following equations [121] :

$$\begin{aligned} \sigma_r &= \tfrac{P}{N}(r + \tfrac{a^2b^2}{r^3} - \tfrac{a^2+b^2}{r})sin\theta \\ \sigma_\theta &= \tfrac{P}{N}(3r - \tfrac{a^2b^2}{r^3} - \tfrac{a^2+b^2}{r})sin\theta \\ \tau_{r\theta} &= -\tfrac{P}{N}(r + \tfrac{a^2b^2}{r^3} - \tfrac{a^2+b^2}{r})cos\theta \\ &\text{where } N = a^2 - b^2 + (a^2 + b^2)log(\tfrac{b}{a}) \end{aligned} \tag{7.24}$$

A background mesh was constructed as a rectangular (2b X2b) NURBS primitive with the specific numerical values as indicated in Fig. 7.7(a). The displacement and traction boundary conditions were modeled explicitly as lower order NURBS primitives. The geometric model and the background mesh are independent of each other as shown in Fig. 7.7(b). The behavior on the explicit primitive boundaries are composed hierarchically as shown in Fig. 7.7(c). The variation of the numerical solution of stress field σ_θ is plotted with radial distance at $\theta = \frac{\pi}{2}$ and compared with the analytical solution in Fig. 7.8.



(a) (b)

(c)

Figure 7.6. Plate with elliptical hole. (a) Geometry ($a = 0.2, b = 0.1, w = 0.5, l = 1$) and boundary conditions, (b) Stress plot in y-direction (σ_{yy}), (c) Hierachical compositions of behavior on explicit geometries.

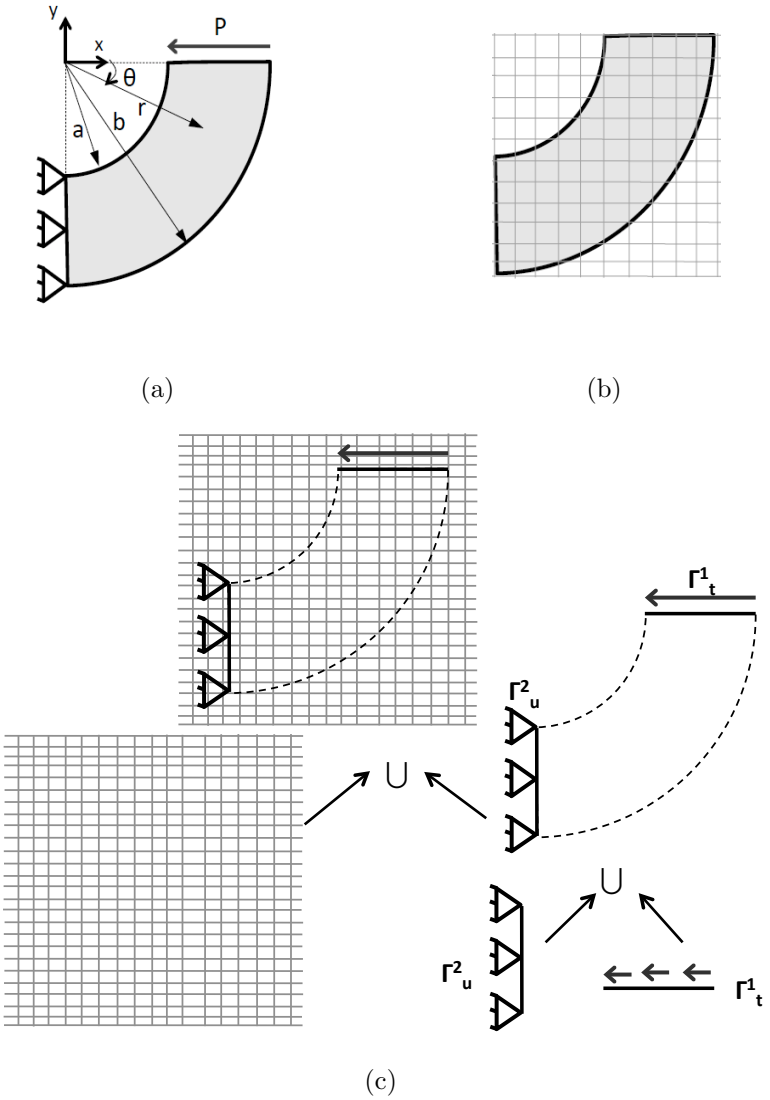


Figure 7.7. Curved cantilever beam. (a) Geometry (a = 0.2, b = 0.4) and boundary conditions (P = 0.2 units), (b) Underlying control point grid is independent of the geometry, (c) Hierachical compositions of behavior on explicit geometries.

7.5.3 Analysis of a Wrench under Use Load

A problem with relatively complex two-dimensional geometry is considered next; a wrench is analyzed under the loading and boundary conditions shown in Fig. 7.9(a).

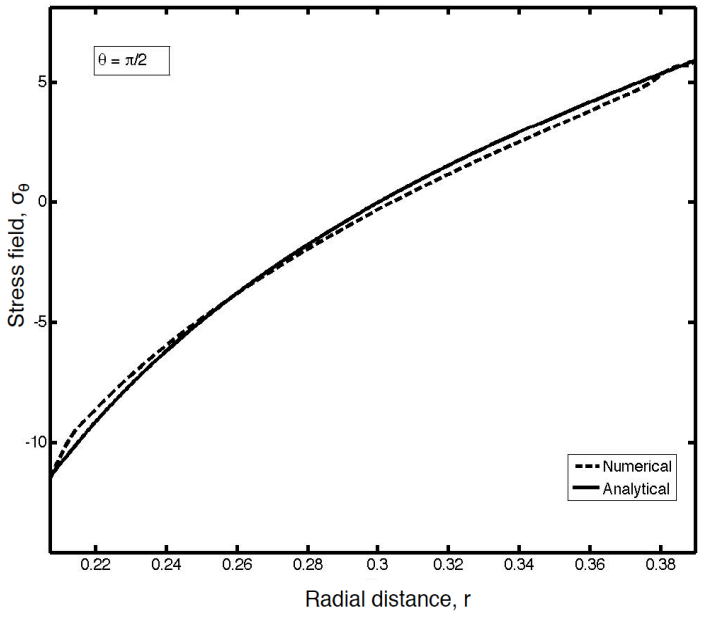


Figure 7.8. Variation of stress field σ_θ with radial distance r at $\theta = \frac{\pi}{2}$.

A rectangular background grid was constructed independent of the wrench geometry. The behavior on the explicit boundaries are hierarchically composed as shown in Fig. 7.9(c). The modulus of elasticity of the wrench was assumed to be 100 units and the Poissons ratio was 0.3. The displacement and traction boundary conditions were modeled explicitly as lower order NURBS primitives. Weight fields were constructed from each of these primitives using algebraic distance field. The effective weight field corresponding to the rectangular domain NURBS primitive in the analysis problem is shown in Fig. 7.9(b). Deformed shape is plotted in Fig. 7.10.

7.5.4 Analysis of a Hip Implant

The analysis of a hip implant under prescribed loading and boundary conditions is demonstrated. The implant is fixed at its lower end while traction boundary condition is applied on the top surface. The implant material is assumed to be elastic with

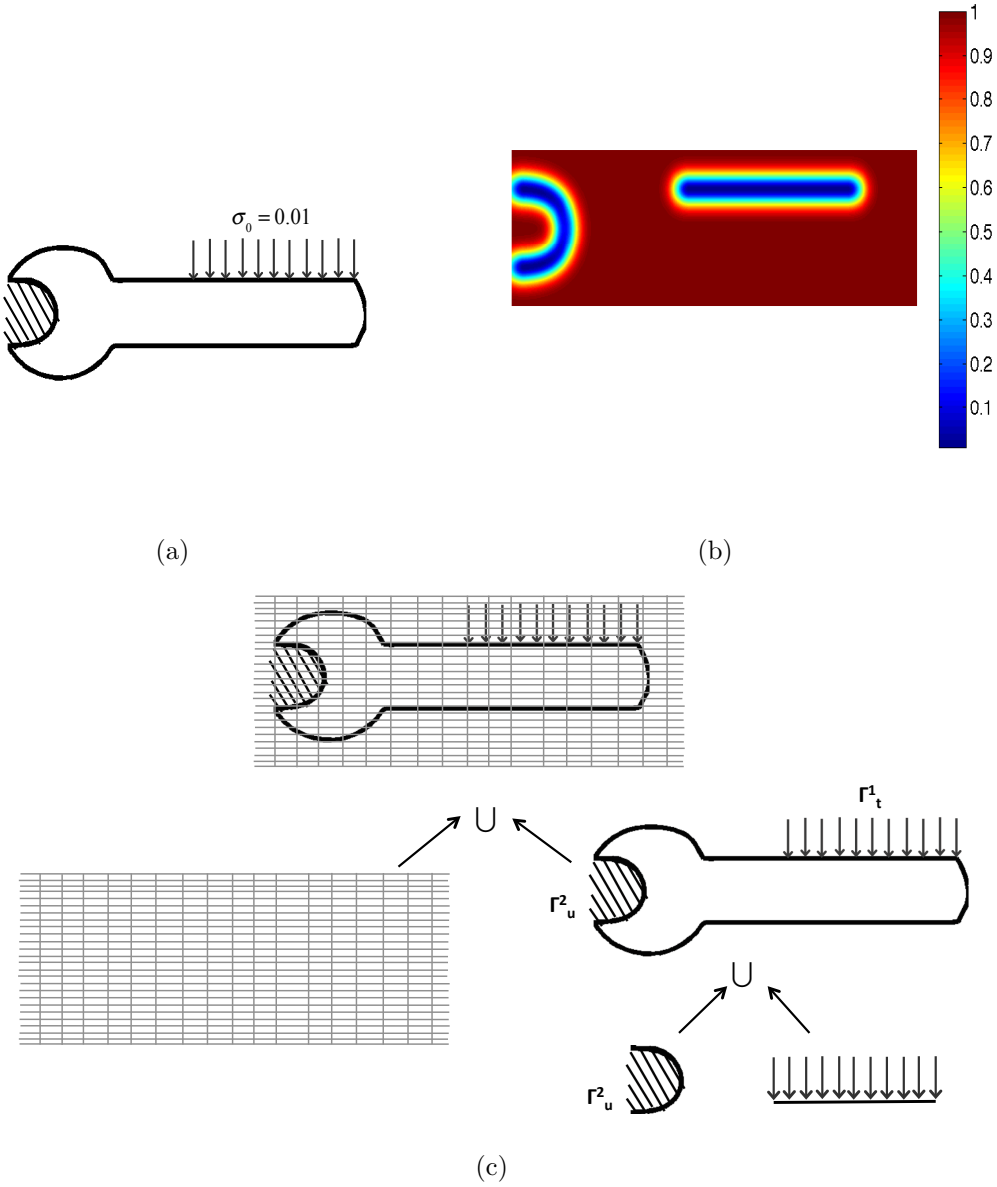


Figure 7.9. Analysis using boundary representation of a wrench. (a) Loading and boundary condition on the wrench, (b) Weight field of the rectangular domain NURBS primitive constructed using algebraic distance from the displacement and traction boundaries, (c) Hierachical compositions of behavior on explicit geometries.

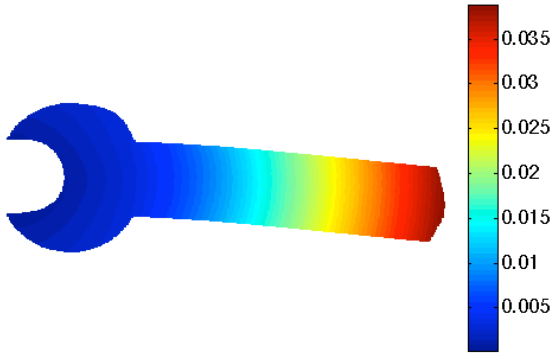


Figure 7.10. Analysis results. The resultant displacement as well as the deformed shape shown magnified five times.

elastic modulus $E = 100$ units and poisson's ratio $\nu = 0$. Loading and boundary conditions are shown in Fig. 7.11(a).

A uniform background NURBS approximation of 20X20X20 control point grid is constructed independent of the implant geometry. The explicit traction and displacement boundaries are modeled as NURBS surfaces. Weight field was constructed from each boundary primitive using algebraic level sets. The effective weight field corresponding to the background domain NURBS primitive in the analysis problem is shown in Fig. 7.11(b). Displacement field in z-direction is plotted in Fig. 7.11(c).

7.5.5 Heat Conduction in Utah Teapot

A heat conduction analysis problem is demonstrated for the utah teapot geometry discussed in the previous chapter. The thermal conductivity of water in the teapot was assumed to be 1 unit. A temperature boundary condition of 100 °C was applied to the bottom surface of the teapot while the top surface was exposed to air at 25 °C. Convection and radiation effects were neglected. The loading and boundary conditions are shown in Fig. 7.12(a).

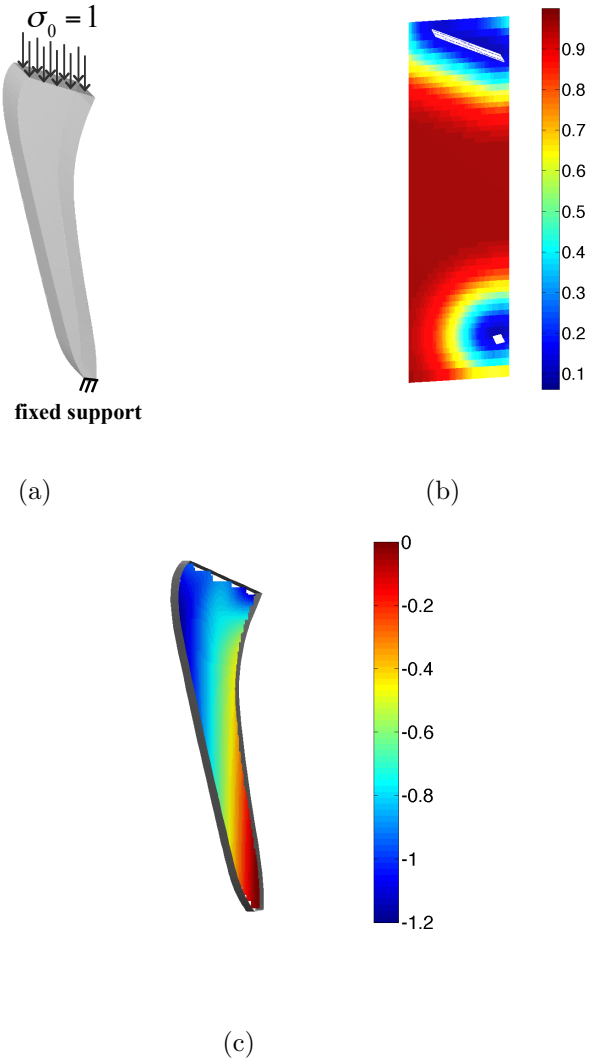
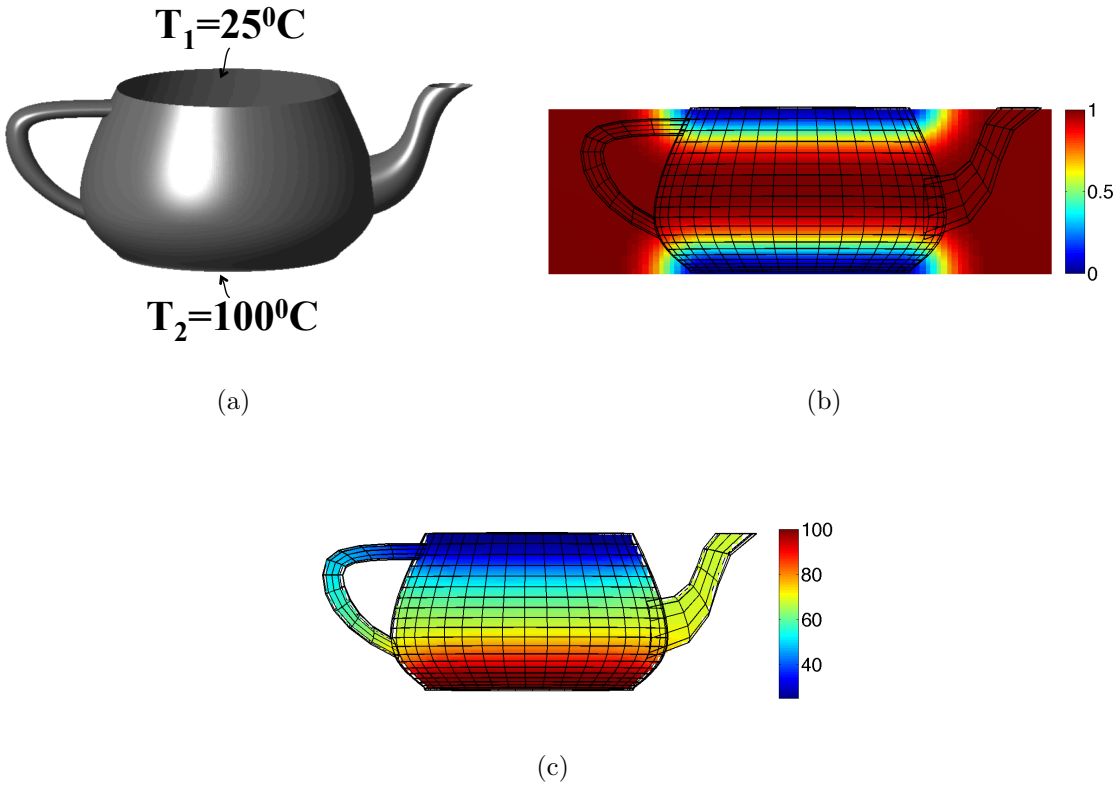


Figure 7.11. Analysis of a hip implant. (a) Geometry and boundary conditions, (b) Weight field on the background domain goes to zero on explicit boundaries,(c) z-displacement plotted on a cross-section of the implant.

A uniform background NURBS approximation of 25X25X25 control point grid was constructed independent of the utah teapot geometry. The behavior on the explicit dirichlet boundaries were hierarchically composed with the background domain. The temperature boundary conditions were modeled explicitly as lower order NURBS primitives. Weight fields were constructed from each of these primitives using alge-

braic level sets. The effective weight field corresponding to the rectangular domain NURBS primitive in the analysis problem is shown in Fig. 7.12(b). Temperature distribution in the teapot is plotted in Fig. 7.12(c).



(a) (b)

(c)

Figure 7.12. Thermal analysis of utah teapot. (a) Geometry and boundary conditions, (b) Weight field on the background domain goes to zero on temperature boundaries,(c) Temperature field plotted on a cross-section of the teapot.

8. CRACK GROWTH ANALYSIS USING UNSIGNED DISTANCE FIELD

The potential applications of the algebraic distance field in modeling fracture and damage are demonstrated using the enriched field modeling technique [2] described in Chapter 7. The choice of enrichment function depends on the *apriori* knowledge of the behavior. For example, to model the discontinuity in the solution field across the crack surface, the Heaviside step function is used as enrichment such that $\psi = 1$ above the crack surface and $\psi = -1$ below it [2, 25, 118, 122]. The continuous and discontinuous fields are modeled completely independent of each other and composed to obey partition of unity. The weight field is constructed using the algebraic distance field developed in this thesis. The significant computational advantage of such an enrichment strategy is that the changes are localized to crack geometry during crack propagation. Also, note that the work described in this chapter is in collaboration with the authors of [123].

8.1 Dirichlet Boundary Conditions

In this example , a two-dimensional square domain of size 2×2 units is considered. The domain is centered at the origin with the $x-$axis in the horizontal direction and the $y-$axis in the vertical direction. The boundary Γ_1 is restrained in $y-$direction and a displacement of $u_y = 0.015(1-x^2)$ units is applied on the boundary Γ_2 (see Fig. 8.1). The material is isotropic linear elastic with a modulus of elasticity of $E = 200$ units and Poisson's ratio of $\nu = 0$.

In this example, the enriched field approximation theory is used to satisfy prescribed non-homogenous boundary conditions. Realizing the fact that boundaries of a domain are lower-dimensional geometrical entities, an enriching field can be constructed on them using the procedure described in Chapter 7. Let the boundary where dirichlet

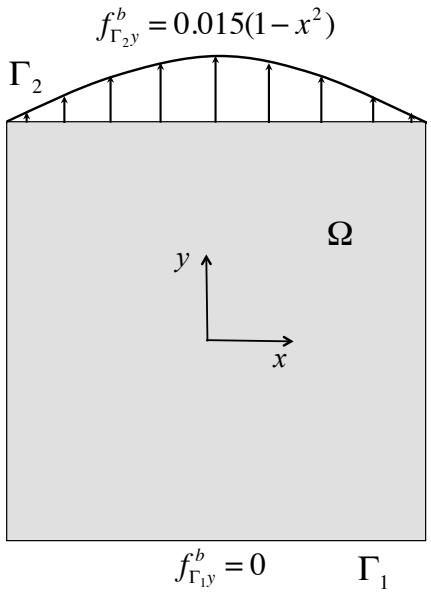


Figure 8.1. Plate with specified boundary conditions : Geometry and loading.

boundary conditions are specified be denoted by Γ_u and let the specified boundary conditions be f^b_Γ. The behavioral approximation to the solution of the boundary value problem is then constructed as described before

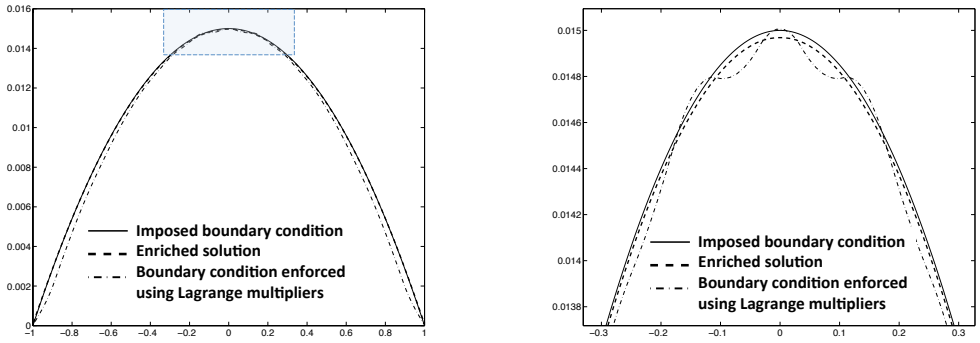
$$f(\mathbf{x}) = \left(1 - w^b_\Omega\right) f^c_\Omega(\mathbf{x}(\xi,\eta)) + w^b_\Omega f^b_\Gamma \left(P\left(\mathbf{x}\right) \rightarrow t\right) \tag{8.1}$$

where, f^c_Ω is the underlying continuous approximation and w^b_Ω is the weight function. In this example, the square geometry is modeled as a bi-quadratic NURBS surface with uniform, 10×10 control point distribution, and the boundaries (Γ_1, Γ_2) are modeled as quadratic NURBS curves with 10 control points each. The exponential blending function (Eq. (8.2)) is used which in turn is a function of the algebraic distance field (d).

$$w^b_\Omega = e^{-\left(\frac{d}{d_0}\right)^2} \tag{8.2}$$

$f(\mathbf{x})$ smoothly approximates the specified boundary conditions as the weight field w^b_Ω goes to 1 on the boundary Γ_u and decays exponentially away from the boundary.

Note that in the proposed approach, the boundary approximation f^b_Γ is constructed independent of the underlying continuous approximation f^c_Ω. Also, the prescribed



(a) (b)

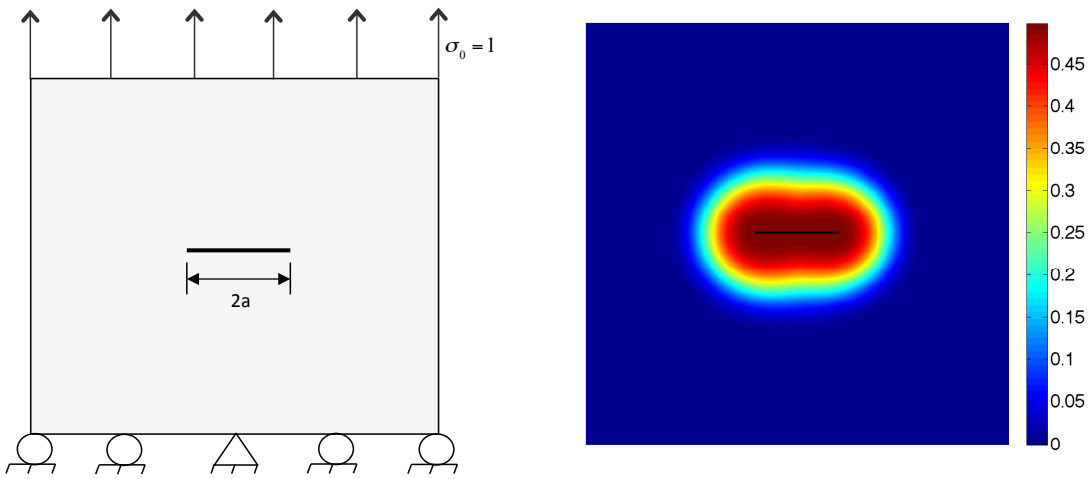
Figure 8.2. Comparison of Dirichlet boundary conditions. (a) Displacement field u_y obtained by enrichment technique is plotted and compared with imposed boundary condition and boundary condition enforced using lagrange multipliers, (b) The same plot is now magnified in the shaded block region shown in the previous figure.

boundary conditions are satisfied exactly. The resulting y-directional displacement field is shown in Fig. 8.2. A comparison with the imposed boundary condition and the boundary condition enforced using lagrange multipliers is also shown. It is observed that using enrichment technique, the Dirichlet boundary condition is enforced fairly well when compared with the oscillatory displacements obtained by lagrange multipliers.

8.2 Plate with Crack

In this example, we consider an isotropic, elastic square plate 2x2 units centered on the origin with x-axis in the horizontal direction and the y-axis in the vertical direction with a horizontal center crack of length 0.2 units. Unit traction is applied in the y direction on the top edge of the plate with the bottom edge restrained in the y direction except at the bottom center where the displacement was restrained in both x and y directions. The modulus of the plate was assumed to be 100 units and

the Poissons ratio was 0.3. The geometry and boundary conditions of the problem are shown in Fig. 8.3(a). The algebraic distance field is used to determine the weight field of the crack over the rectangular domain as shown in Fig. 8.3(b).



(a) (b)

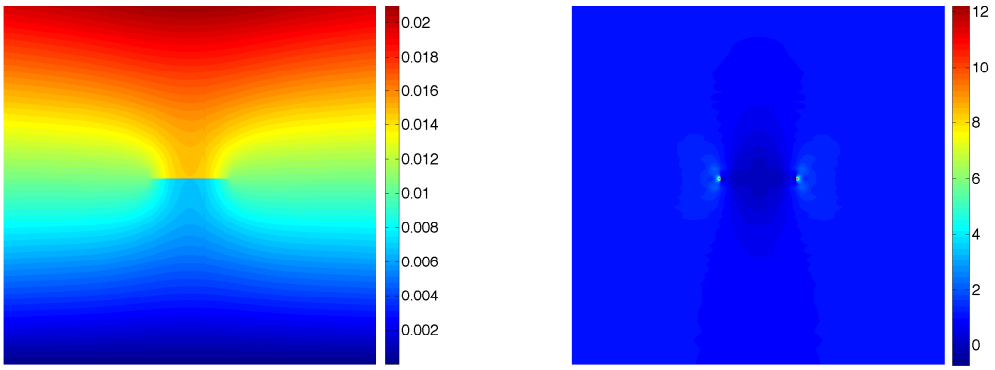
Figure 8.3. Plate with crack under tension. (a) Geometry and boundary conditions, (b) Algebraic distance field based weight field of the crack geometry.

The stress intensity factor (SIF) is obtained using the displacement correlation. The expression for the stress intensity factors obtained using displacement correlation under plain strain conditions (and under plane stress conditions if ν is replaced by $\nu = \nu/(1+\nu)$) are:

$$K_I = \frac{\mu\sqrt{2\pi}\,(v_b - v_a)}{\sqrt{r}\,(2-2\nu)} \tag{8.3}$$

$$K_{II} = \frac{\mu\sqrt{2\pi}\,(u_b - u_a)}{\sqrt{r}\,(2-2\nu)} \tag{8.4}$$

where μ is the shear modulus, ν is Poisson's ratio, r is the distance from the crack tip to the correlation point, (u_a, v_a) and (u_b, v_b) are the orthogonal displacements in x and y directions at correlation points a and b respectively.



(a) (b)

Figure 8.4. Plate with crack under tension. (a) Displacement fields in y-direction, (b) Normal stress fields in y-direction.

As a validation exercise, the SIF was first determined. The displacement field and stress field in the y-direction in the plate corresponding to such a crack are shown in Fig. 8.4. The SIF determined using displacement correlation was 0.7797 units, which compares very well with the analytical solution of 0.785 units.

8.3 Modeling Cracks through Material Enrichment

A heat conduction application is considered here in which heat flows from one face of the solid into the opposite face that is cooled. In the presence of a thermal defect, normal heat flow is blocked and therefore causes a temperature rise behind the defect.

In this example, the defect is modeled implicitly as a material enrichment [2] of zero thermal conductivity over the underlying domain. The enriched thermal conductivity of the domain is modeled as:

$$k = (1 - w)k_0 + wk_e \tag{8.5}$$

where, k_0 is the thermal conductivity of the underlying material, k_e is the thermal conductivity of the defect. The weight function w smears the loss in thermal conduc-

tivity over a finite distance. The weight function is constructed as a monotonically decreasing function of distance from the defect. A Gaussian form of weight field is used in this example:

$$w = e^{-\left(\frac{s}{s_0}\right)^2} \tag{8.6}$$

where s is the algebraic distance field from the defect and s_0 is a scaling factor.

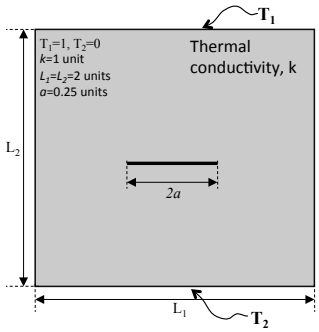
An infinite strip defect was modeled as an equivalent two-dimensional problem with a line defect. A two-dimensional square domain of size 2×2 units was considered (see Fig. 8.5(a)). Constant temperature boundary conditions were enforced on top and bottom edges of the domain as shown in Fig. 8.5(a). The enriched thermal conductivity of the domain is shown in Fig. 8.5(b). The obtained numerical solution for rise in temperature behind the defect is plotted against the analytical solution [124] in Fig. 8.5(c).

The accuracy of the material enrichment model is a function of the distance scaling factor s_0. As $s_0 \to 0$, the smeared region approaches a sharp crack configuration. Refinement studies were performed for three scaling factors $s_0 = 0.2, 0.1, 0.05$ (Fig. 8.5(c)). In the limit, as $s_0 \to 0$, the solution approaches the analytical one with refinement of the control point spacing h. The convergence in error as a function of the distance scaling factor, s_0 for three points on the crack at distance x from the crack center is shown in Fig. 8.5(d).

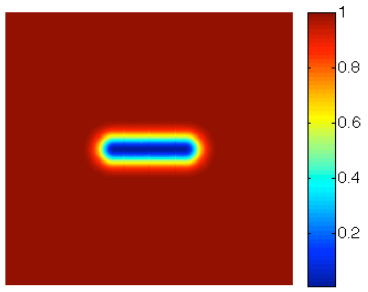
8.4 Analysis of a Curved Crack

The problem of an infinite plate with a curved center crack under uniaxial tensile stress is modeled here and the numerical convergence of the stress intensity factors is demonstrated. This problem was adapted from the reference [125]. The analytical stress intensity factors are [126]:

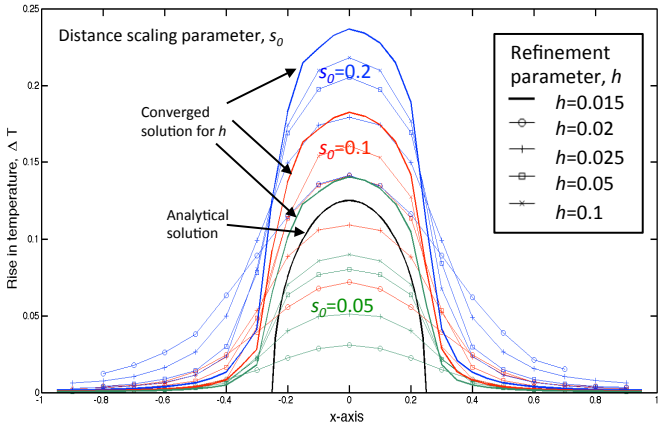
$$K_I = \frac{\sigma}{2}\sqrt{\pi R sin(\beta)}\left[\frac{(1 - sin^2(\beta/2)\, cos^2(\beta/2))\, cos(\beta/2)}{1 + sin^2(\beta/2)} + cos(3\beta/2)\right] \tag{8.7}$$



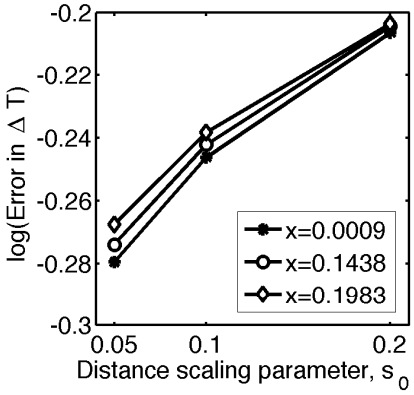
(a)



(b)



(c)



(d)

Figure 8.5. Thermal conduction in the presence of infinite strip defect. (a) Infinite strip defect modeled as a two-dimensional domain with a line crack, (b) Enriched thermal conductivity over the domain, (c) Validation for rise in temperature behind defect, (d) Error in rise in temperature as a function of distance scaling factor.

$$K_{II} = \frac{\sigma}{2}\sqrt{\pi R sin(\beta)}\left[\frac{(1 - sin^2(\beta/2)\, cos^2(\beta/2))\, sin(\beta/2)}{1 + sin^2(\beta/2)} + sin(3\beta/2)\right] \tag{8.8}$$

where σ is the uniaxial tensile stress, R is the radius of the circular arc, and 2β is the angle subtended by the arc at the center as shown in Fig. 8.6(a).

The problem is modeled using a finite square domain of size 2×2 units and a circular crack of radius $R = 1.005$ units, center angle $2\beta = 11.42°$ such that the ratio of the square side length to crack length is reasonably large (≥ 10) (see Fig. 8.6(a)). Only one half of the plate with the crack was modeled due to symmetry across the y-axis. The loading and boundary conditions are described in Fig. 8.6(b). Elastic material properties with modulus $E = 100$ and poisson's ratio $\nu = 0.3$ were chosen.

Previously developed method for enriched isogeometric approximations [2] was adopted to model the crack. A general enriched approximation for modeling fracture is of the the form:

$$f = (1 - w_e) f_\Omega + w_e f_e \tag{8.9}$$

where, the continuous field (f_Ω) and discontinuous enriching field (f_e) are approximated independent of each other and composed to obey partition of unity. The weight field w_e is defined as an exponentially decaying function of algebraic distance. The discontinuity due to the crack is modeled using Heaviside step function $H(\mathbf{x})$ as:

$$f_e(P(\mathbf{x}) \to t) = \sum_I N_I(t) \hat{u}_I H(\mathbf{x}) \tag{8.10}$$

Uniform control point grid discretizations of $20 \times 20, 25 \times 25, 31 \times 31$ and 34×34 were chosen for the convergence study. The analytical solution for Mode I and Mode II stress intensity factors (for the chosen parameters) are $K_I = 0.5556$ and $K_{II} = 0.0556$ respectively. The numerical stress intensity factors were calculated using the displacement correlation method. The resulting expression for estimating the stress intensity factors under plane strain conditions (and under plane stress conditions if ν is replaced by $\nu = \nu/(1 + \nu)$) are:

$$K_I = \frac{\mu\sqrt{2\pi}\,(v_b - v_a)}{\sqrt{r}\,(2 - 2\nu)} \tag{8.11}$$

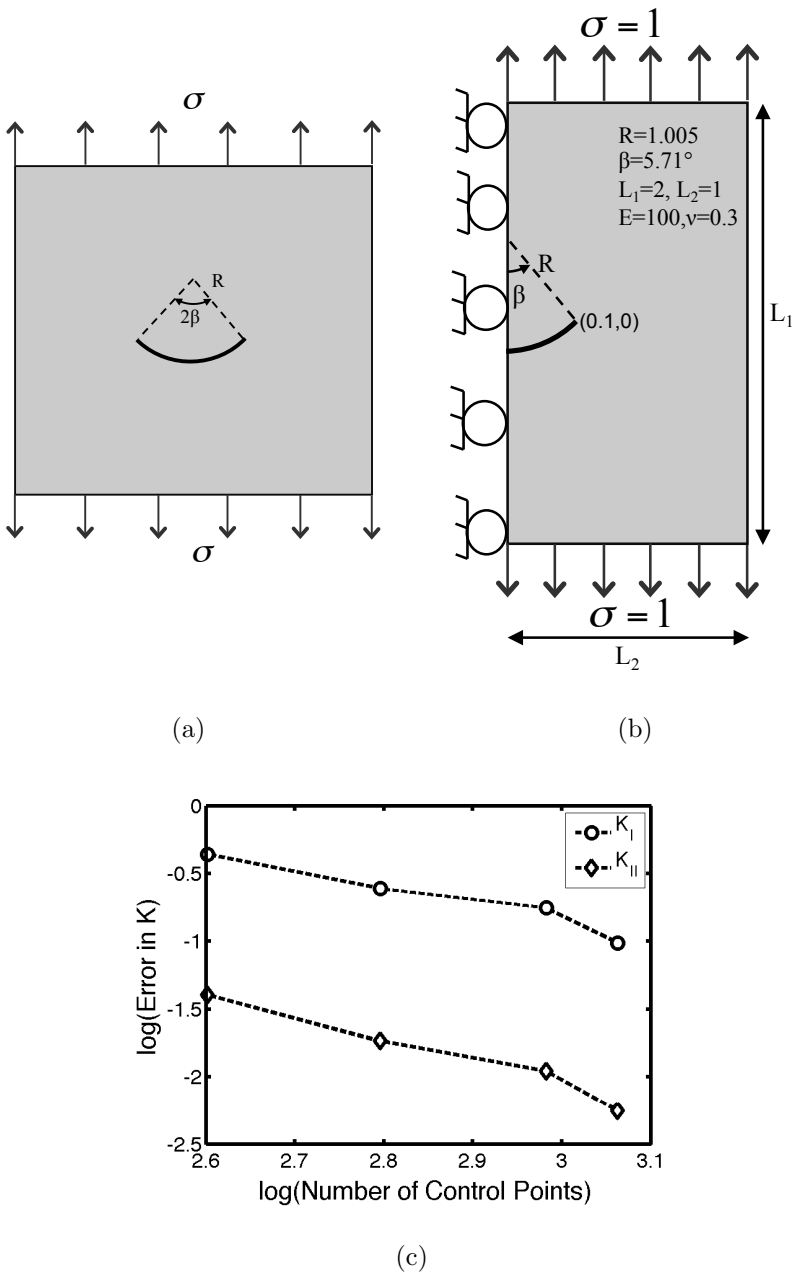


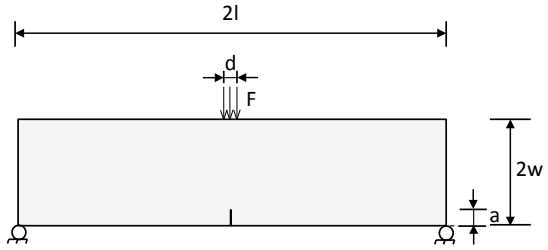
Figure 8.6. Plate with a curved crack. (a) Infinite plate modeled as a two-dimensional finite domain with a center curved crack, (b) Geometry, loading and boundary conditions, (c) Error in numerical stress intensity factors as a function of number of control points.

$$K_{II} = \frac{\mu\sqrt{2\pi}\,(u_b - u_a)}{\sqrt{r}\,(2 - 2\nu)} \tag{8.12}$$

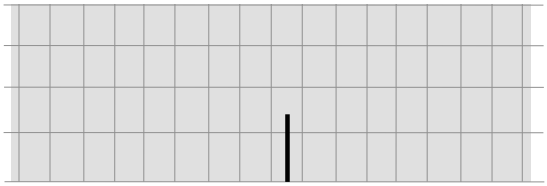
where, μ is the shear modulus, ν is Poisson's ratio, r is the distance from the crack tip to the correlation point, (u_a, v_a) and (u_b, v_b) are the orthogonal displacements in x and y directions at correlation points a and b respectively. The convergence in the stress intensity factors is shown in Fig. 8.6(c).

8.5 Three-Point Bending Test

In this example, a beam in three point bending is considered (Fig. 8.7). This example has been adapted from [127]. The crack growth is studied for the following geometrical parameters, $l = 0.3$ units, $w = 0.075$ units, $a = 0.025$ units and the following material parameters, $E = 36.5$ GPa, $\nu = 0.1$. The boundary conditions are as shown in Fig. 8.7(a) with $F = 1$ unit.



(a)



(b)

Figure 8.7. Three-point bending test (a) Geometry and boundary conditions. (b) Underlying regular control point grid independent of internal boundary.

The phenomenon of fracture is characterized by a sharp jump in the solution field across the crack faces. Cracks are modeled using a behavioral approximation of the form:

$$f_{\Gamma}^{d}(P(\mathbf{x}) \rightarrow t) = H(\mathbf{x}) \sum_{I} N_I(t) \hat{u}_I \tag{8.13}$$

where, $H(\mathbf{x})$ models the discontinuity across the crack geometry. The Heaviside step function $H(\mathbf{x})$ is defined here as

$$H(\mathbf{x}) = \frac{d(\mathbf{x})}{|d(\mathbf{x})|} \tag{8.14}$$

where, $d(\mathbf{x})$ is the signed distance to the crack geometry from the spatial location $\mathbf{x}$. The corresponding weight field w_{Ω}^{e} for the enriching behavioral field is defined as an exponentially decaying distance field as in previous example. The continuous and discontinuous fields are modeled completely independent of each other and composed to obey partition of unity (see Fig. 8.7(b)). Hence, the changes are localized to crack geometry during crack propagation. In this example, the underlying continuous geometry is modeled using a uniform bi-quadratic NURBS discretization with a spacing of $h = 0.05$ units over the domain, and the initial crack geometry is modeled using a quadratic NURBS curve with 10 control points. At the i^{th} step of crack propagation, the fracture criterion is evaluated and the angle of crack propagation θ is computed [2]. A linear NURBS segment of length $\Delta a = 0.005$ units is created between crack tip at the i^{th} step and the new crack tip. The original crack geometry and the new NURBS segment are joined into a single uni-variate NURBS curve for the $(i+1)^{th}$ crack propagation step. Snapshots of crack propagation steps are shown in Fig. 8.8.

8.6 Simulations of Inclined Crack Propagation

Crack propgation in a 45° inclined edge-cracked specimen under uniform tension is modeled next. Geometry and loading conditions were as shown in Fig. 8.9. The elastic modulus was assumed to be 100 units and the Poisson's ratio was 0.3.

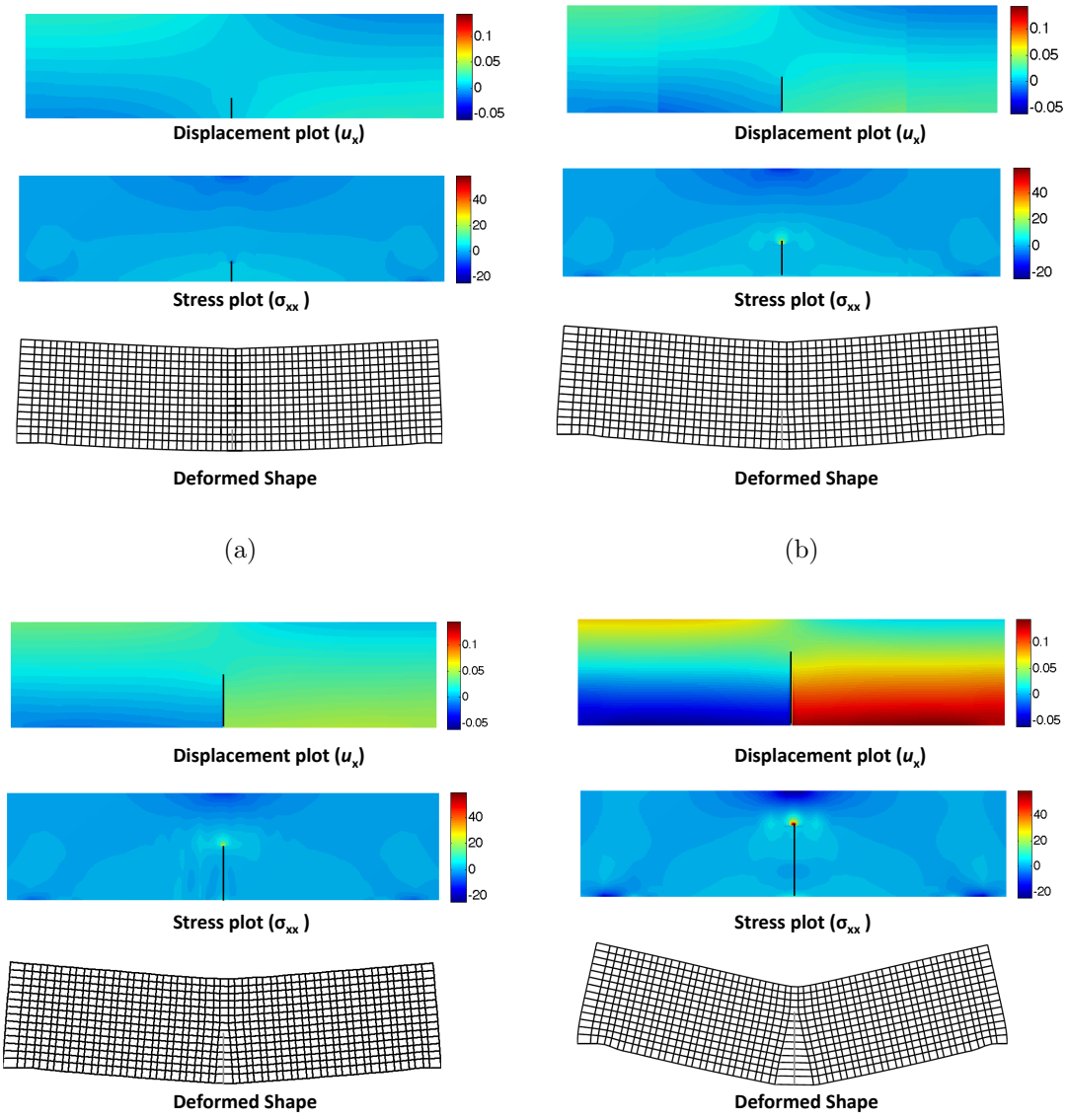
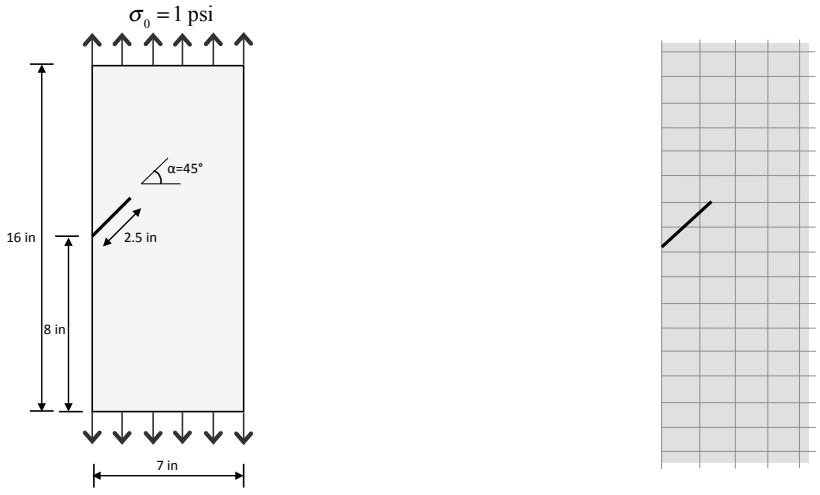


Figure 8.8. Snapshots of crack propagation steps in the three-point bending test.



(a) (b)

Figure 8.9. Plate with inclined crack under tension. (a) Geometry and boundary conditions, (b) Underlying regular control point grid independent of inclined crack.

The crack was modeled as a material enrichment [2] with a modulus value of zero. The elastic modulus of the underlying material was enriched with a cohesive damage description to nucleate and propagate the crack. Thus, the enriched elastic modulus of the domain was of the form:

$$\begin{aligned} E &= (1-w)E_0 + w(1-D)E_0 \\ 0 &\leq D \leq 1 \end{aligned} \tag{8.15}$$

where, E_0 is the elastic modulus of the undamaged material, D is the measure of damage with $D = 0$ being the pristine state and $D = 1$ being the fully damaged state of the material at which it does not bear any load (fractured state). The weight function w smears the loss in modulus over a finite spatial region. The algebraic distance field was used to determine the weight field of the crack over the rectangular domain.

The crack propagation was based on maximum damage at points on a circular path ahead of the crack tip. In general, the damage measure D is defined as the ratio:

$$D = \frac{G}{\Gamma} \tag{8.16}$$

where G is the energy release rate and Γ is the fracture toughness. In this example, a bilinear irreversible cohesive damage description was used [2]. Snapshots of crack propagation steps are shown in Fig. 8.10. It can be observed that the crack eventually becomes horizontal, which is expected due to the symmetric tension loading on the specimen.

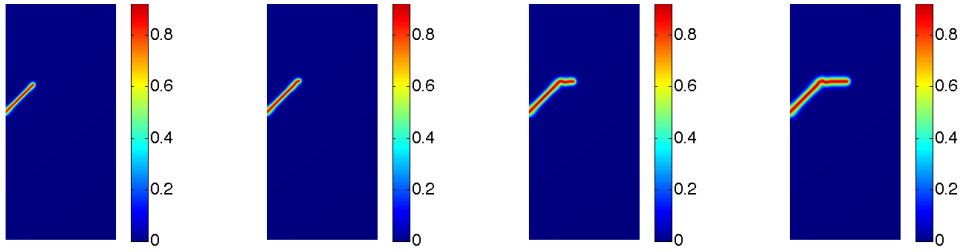
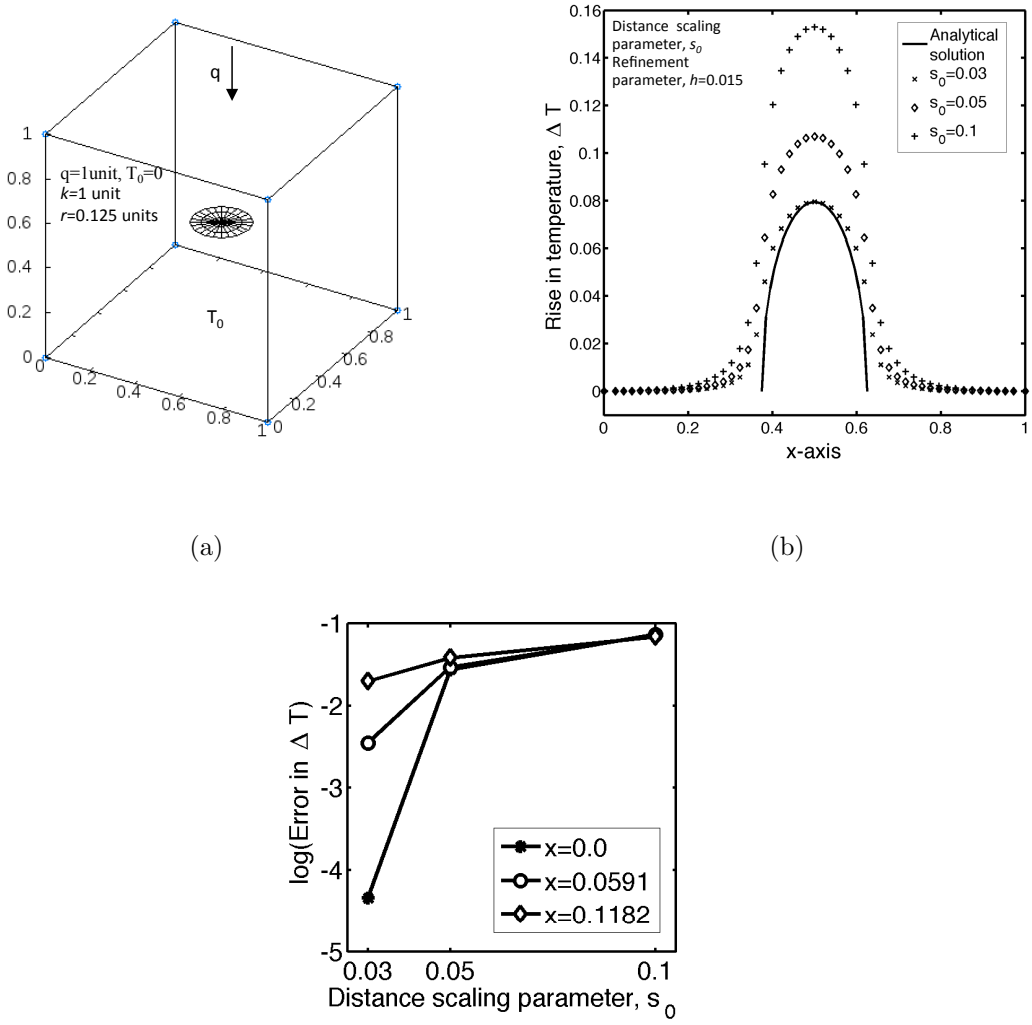


Figure 8.10. Snapshots of crack propagation steps in order: Step 2 $\longrightarrow$ Step 6 $\longrightarrow$ Step 10 $\longrightarrow$ Step 15.

8.7 Analysis of Three-Dimensional Cracks

The heat conduction problem considered in Section 8.3 is reconsidered here. Specifically, a three-dimensional domain with a planar circular defect is analyzed. Geometry and boundary conditions are shown in Fig. 8.11(a). The converged numerical solution for rise in temperature behind the defect is compared to the analytical solution for distance scaling factors $s_0 = 0.1, 0.05, 0.03$ in Fig. 8.11(b). As before, as $s_0 \to 0$, the numerical solution approaches the analytical solution [124]. Again, the convergence

in error as a function of the distance scaling factor, s_0 for three points on the crack at distance x from the crack center is shown in Fig. 8.11(c).

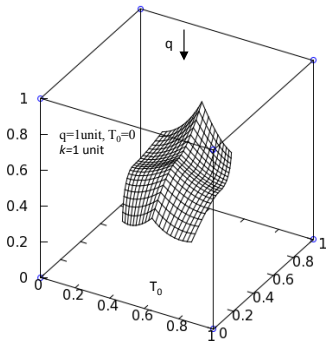


(a) (b)

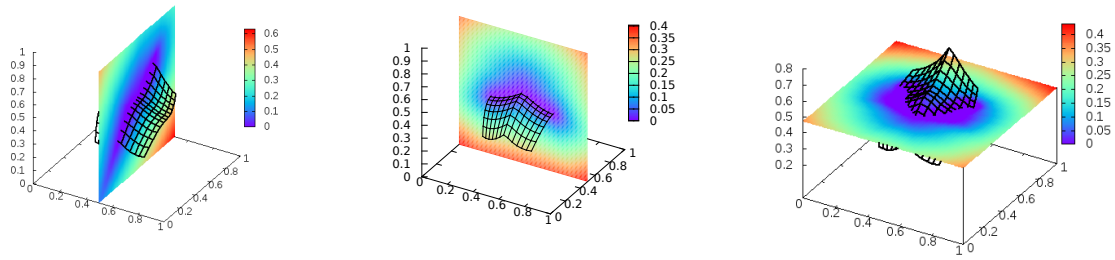
(c)

Figure 8.11. Thermal conduction in the presence of a cirular defect. (a) Three-dimensional domain with a circular defect: geometry and boundary conditions, (b) Validation for rise in temperature behind defect,(c) Error in rise in temperature as a function of distance scaling factor.

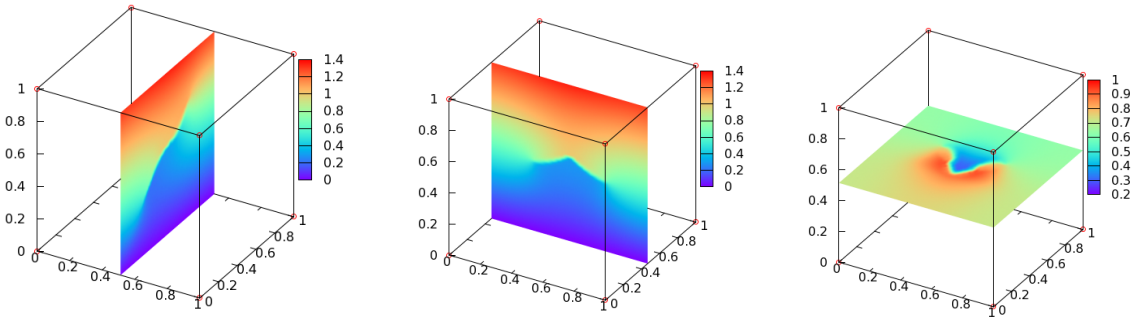
Finally, a complex, three-dimensional freeform NURBS defect geometry was chosen to illustrate the application of algebraic distance field for modeling an implicit crack. Geometry and boundary conditions are shown in Fig. 8.12(a). The algebraic distance field for the complex defect geometry is plotted in Fig. 8.12(b) over three planes slicing the geometry in x, y and z directions respectively. The numerical solution for temperature in the domain is plotted over the same three slices in Fig. 8.12(c). Through this example, it is reinforced that an approximate distance function as developed in this study is sufficient to enrich underlying domain with known discontinuous behavior.



(a)



(b)



(c)

Figure 8.12. Thermal conduction in the presence of complex freeform defect geometry. (a) Geometry and boundary conditions, (b) Algebraic distance field plotted over planes slicing the defect geometry in the three principal directions, (c) Temperature solution plotted over the same planes as the algebraic distance field.

9. SIMULATIONS OF DAMAGE AND FRACTURE IN ULK UNDER PAD STRUCTURES DURING CU WIREBOND PROCESS

9.1 Introduction

The fragility of ULK dielectrics has made Back End of Line (BEOL) integration and packaging a significant challenge for the sub-32 nm technology node and beyond [128, 129]. The ULK dielectrics have made the ILD layer more fragile with weaker mechanical stiffness and fracture-resistance [129, 130]. In the case of wire bonded packages, the reliability concern associated with the fracture of ULK dielectrics while bonding over the active circuits is a significant challenge due to the impact load and the high ultrasonic energy transmitted to the ILD stack. In general, the die sizes in (wirebonded) fine feature technology are pad limited since they are dictated by I/O rather than active areas. Thus, there is a need to place bond pads over active circuitry since that minimizes die size and improves wafer utilization [131]. This is a significant reliability challenge in wirebonded low-k and ULK dies due to the impact load and the high ultrasonic energy transmitted to the ILD stack [132] which makes the ULK dielectric stacks vulnerable to wirebond-induced fracture. This challenge is further exacerbated by the fact that copper wire bonding is set to replace gold wire bonding in high I/O devices because of its lower cost [133, 134]. Since copper is a stiffer material, it requires greater force to insure good bonding to pad, and a larger bond force in turn increases the risk of ULK fracture during the bonding process. This implies that wirebonding is being performed using stiffer wire on a weaker support structure.

Hence, Cu wirebonding on fine pitch low-k (and by extension to ULK) devices is challenged by two facts: (1) higher impact force as well as higher ultrasonic energy is typically necessary for comparable bond strength (as devices with all SiO_2 dielectric

stack) [130] and (2) bonding over active circuits induces mechanical damage leading to fracture during probing and wirebonding [135, 136]. Multi-material corners in ILD stacks that are locations of singular stresses [137] and hence, potential crack nucleation sites, further increase the risk of ULK fracture during the bonding process. Thus, from a design or process optimization point of view, it is essential to develop insights into bonding mechanism and the risk of ULK fracture during the wirebonding process. The research on wire bonding using copper wire is being conducted extensively and numerous experimental investigations of the bonding mechanism and optimization of process parameters were seen in literature [138–143]. Further, few numerical investigations of this problem were also found, but were limited to the study of deformation and stresses induced by Cu wirebonding process in the Cu free air ball (FAB), pad and die regions [144–149]. A review of experimental and numerical findings and recent advances in Cu wirebonding is presented in [132].

However, studies investigating ULK fracture/damage due to wirebonding appear to be very few in literature and are mostly limited to gold wirebonding. An experimental investigation of gold wire bonding impact on low-k dielectric material and bond pad failure is presented in [150]. Numerical analysis of delamination in Cu/low-k bond pads during wire pull test for gold wires uses energy methods to determine the damage sensitivity of different three-dimensional bond pad structures [151]. Futher, brittle delamination failure mode in low-k dielectric stack interfaces was studied in gold wirebonding using an energy based failure criterion [152]. Similar investigations into the impact of Cu wirebond on ULK fracture appear to be nearly non-existant in existing literature. In an experimental study of the underpad damage risk during Cu ball bonding, damage is evaluated by relating the ultrasonic force measured using a microsensor method to the mechanical stress acting on the pad [153]. In another experimental investigation, the impact of Cu wirebonding process parameters on pad damage was studied [143]. Further, in a numerical study of pull test for Cu wires, the damage patterns at the end of pull test were seen to be affecting the metal layers [154].

However, a detailed investigation of ULK fracture/damage due to Cu wirebonding process was not found in these studies.

In this work, a detailed numerical investigation of the risk of fracture in the ULK dies due to Cu wirebonding process is presented. Maximum damage induced in the ILD stack during the process steps is proposed as an indicator of the reliability risk. A multi-level modeling technique is presented to simulate damage accumulation in ILD structure during the wirebonding process. A dynamic finite element (FE) model is constructed in ABAQUS/Explicit to model the deformation and stress fields in the under pad region during the wirebonding process steps of impact and ultrsonic vibration [155]. Further, sophisticated isogeometric computational techniques [2,156] capable of nucleating and propagating cracks at arbitrary locations are used for modeling damage in ILD stack. A damage analysis framework is proposed to develop insights on risk of fracture in the dielectric layers due to Cu wirebonding process. Damage accumlation studies are conducted to identify weak interfaces and potential sites for crack nucleation as well as damage nucleation patterns. Further, the critical process condition is identified by analyzing the damage induced during the impact and ultrasonic excitation stages. A number of experimental studies in literature indicated that adding ultrasonic excitation during impact stage (prebleed) softens the Cu FAB and potentially reduces the risk of die fracture [143, 157, 158]. This finding was numerically investigated in this work. Finally, representative ILD stack designs with varying Cu percentage were compared for risk of fracture.

9.2 Demonstration of Simulation Procedure

A two-dimensional plane strain model of the ILD stack is assumed for damage analysis. The baseline interconnect model configuration used for analysis in this work is shown in Fig. 9.1. The 28μm$\times 4.2\mu$m ILD stack consists of 8 metal layers (M1-M8) consisting of Cu lines and vias and three types of ILD materials SiO_2 (M7-M8), ULK (M3-M6) and SiCOH (M1-M2). The details of Cu geometry in M1-M2

layers are not modeled due to the small length scale of these layers. Instead, the two layers are homogenized by volume averaged material properties of Cu and SiCOH. Further, interfaces between the material layers of ILD stack comprise of SiC etch stop layers and Ta barrier layers as shown in Fig. 9.1. These interfaces are modeled as material enrichments. The material properties of the ILD model are given in Tables 9.1 and 9.2.

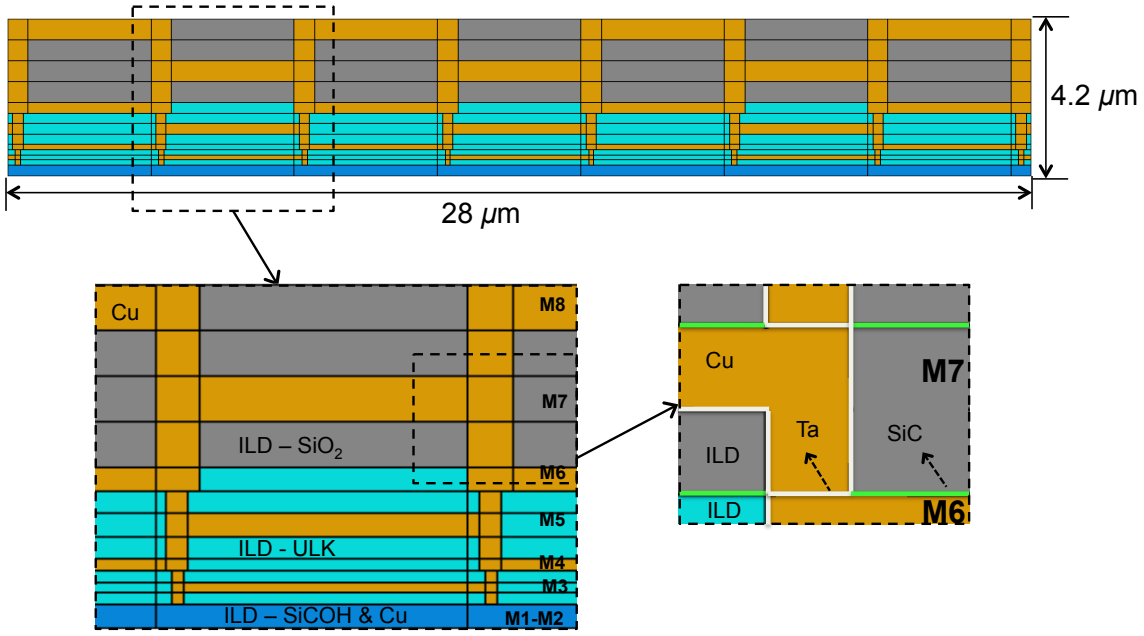


Figure 9.1. A schematic illustration of the baseline interconnect structure with material layers and interfaces.

The simulation procedure is demonstrated by applying a constant temperature field to the ILD stack model. The temperature was increased from 25°C to 100°C. Due to difference in the coefficients of thermal expansion (CTE) of the material domains, thermal strains were induced leading to damage accumulation in the ILD stack as shown in Fig. 9.2.

Table 9.1. Material properties for the ILD stack.

Material	E (GPa)	ν -	α (ppm/K)
Cu	122	0.34	16.7
SiC [159]	200	0.16	4.5
Ta	186	0.35	6.5
ILD-SiO_2	73	0.17	0.55
ILD-SiCOH [160]	9.5	0.2	12
ILD-ULK [161]	3.3	0.2	12

Table 9.2. Interfacial fracture toughness values used in the present work [162].

Interface	G (J/m^2)
SiC-Cu	8
ILD-SiC	3
Ta-ILD	5

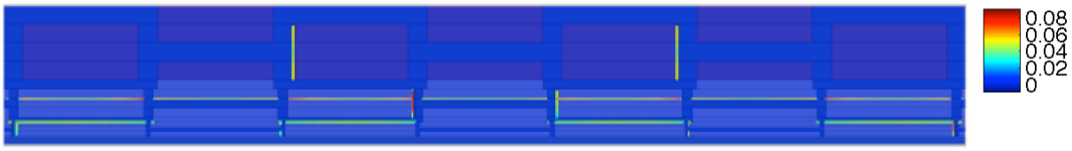


Figure 9.2. Damage accumulated due to thermal strains in ILD stack.

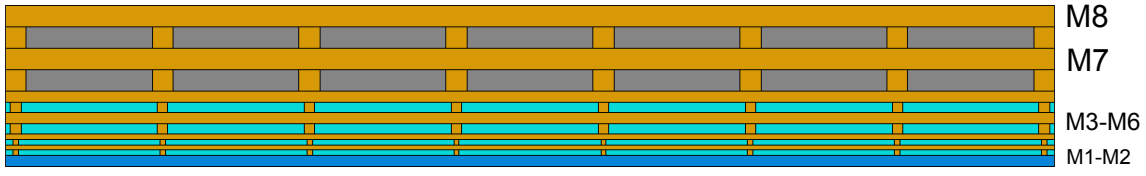
9.3 Geometrical Configurations and Load Cases

9.3.1 Candidate Structures Under Pad

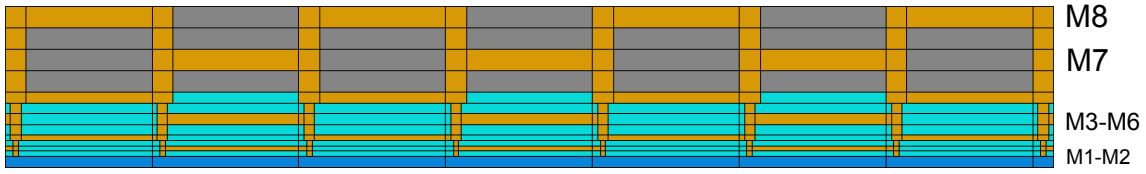
The baseline ILD stack configuration in Fig. 9.1 consists of 34% Cu in M3-M8 layers which is the typical Cu volume fraction used in industry and is referred to as the *Nominal Cu Stack* in subsequent discussion. Two other alternative load-bearing local interconnect structures underneath pad (SUP) are evaluated in this study (see Figs. 9.3(a) and 9.3(c)). The line/via configuration chosen for the three representative structures is similar to those in existing literature [151]. The *High Cu Stack* configuration in Fig. 9.3(a) consists of 56% Cu volume fraction while the *Low Cu Stack* configuration in Fig. 9.3(c) consists of 13% Cu volume fraction. The three representative configurations have same number of Cu vias. They differ with respect to the Cu line configuration, ranging from no Cu lines in *Low Cu Stack* to Cu lines in every layer in the *High Cu Stack*. A detailed damage analysis is carried out on each representative structure to compare the relative fracture resistance of the designs.

9.3.2 Load Cases for Damage Modeling

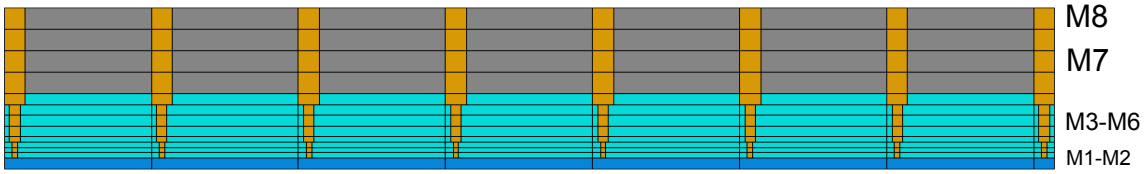
The load imposed on the static ILD stack is extracted from the global finite element model of the wire bond package [155]. The wirebond process typically involves impact and ultrasonic stages. Three impact load cases are modeled here to study the effect of the prebleed magnitude on damage accumulation in ILD stack. These load cases capture increasing magnitude of ultrasonic excitation during impact stage - no prebleed, prebleed of 25% (0.25 μm) and 50% (0.5 μm) amplitude of the ultrasonic excitation (1 μm amplitude). Further, three load cases for the ultrasonic excitation process correspond to the excitation cycles with coefficient of friction (COF) 0.5, 10 and ∞ respectively. Hence, six load cases are identified from the global model for the damage simulation. These are summarized in Table 9.3. For each load case,



(a)



(b)



(c)

Figure 9.3. Representative ILD stack designs with varying Cu volume fraction. (a) High Cu stack (56%), (b) Nominal Cu stack (34%),(c) Low Cu stack (13%).

traction/displacement boundary conditions were extracted from the global model at the time step corresponding to the critical state of stress for that load case.

Table 9.3. Load cases considered for damage simulation in ILD structures.

Impact Stage		
No Prebleed	25% Prebleed	50% Prebleed
Ultrasonic Excitation Cycle		
COF 0.5	COF 10	COF ∞

9.4 Damage Analysis Procedure

A damage analysis framework is proposed to develop insights on risk of fracture in the low-k dielectric layers due to the wirebonding process. Specifically, the damage analysis framework aims at the following reliability studies:

- Identify weak interfaces and potential sites for crack nucleation in the ILD layers
- Identify critical process condition leading to damage and fracture
- Study effect of prebleed on low-k damage
- Evaluate ILD stack designs for relative risk of damage and fracture

9.4.1 Damage Analysis Technique for Impact Stage

Damage is modeled in the baseline *Nominal Cu Stack* for the three impact stage load cases. In this step, weak interfaces and potential sites for crack nucleation during impact process, with and without prebleed, are identified and effect of prebleeding on damage accumulation is studied. Following this, damage evaluated in the impact with 50% prebleed load case is accumulated to the next (ultrasonic excitation) stage through an irreversible damage law (see Fig. 9.4) for all the three ILD stack configurations. Let D_{impact} be the damage accumulated at the end of unloading path of impact stage. The loading path for the new load case will now follow the unloading curve of the previous load case as shown in Fig. 9.4. The damage accumulated after reloading for ultrasonic excitation is computed as

$$D = D_{impact} + \frac{\delta_{max} - \delta_p}{\delta_c - \delta_p} \tag{9.1}$$

Thus, for each unloading and reloading step, permanent damage is accumulated from the previous step. Further, the cohesive damage law parameters (σ_{max} and δ_0) for the reloading step are updated as the new loading path is different from the loading path in the previous step. Such a damage modeling scheme is used in this study for damage accumulation to subsequent process steps.

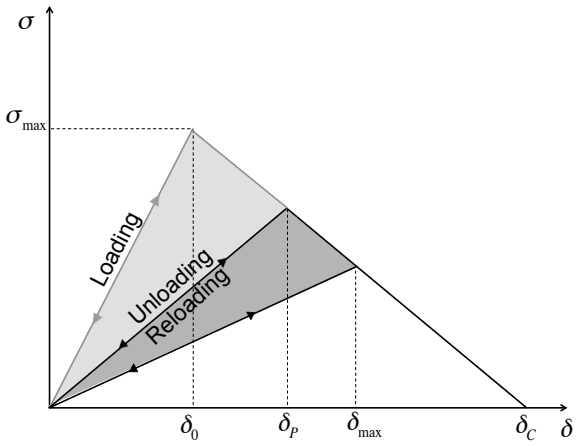


Figure 9.4. Irreversible damage law to accumulate damage to the next process step.

9.4.2 Damage Analysis Technique for Ultrsonic Excitation Stage

The bonding process in the ultrasonic excitation stage is captured through excitation cycles with increasing COF from 0 to ∞. Thus, damage accumulation in the ultrasonic excitation stage, $\Delta D_{ultrasonic}$ is defined as

$$\Delta D_{ultrasonic} = \sum_{COF} \Delta D_{COF} \tag{9.2}$$

where ΔD_{COF} is the damage accumulated during the excitation cycle with COF coefficient of friction.

In this study, three ultrasonic excitation cycles with varying friction coefficients were modeled to simulate the bonding process in the global model. The damage accumulated during n ultrasonic excitation cycles is then approximated as the linear combination of the damage accumulation during n_{COF} cycles of the three load cases simulated in this study with $COF = 0.5, 10, \infty$ and damage accumulation $\Delta D_{0.5}, \Delta D_{10}, \Delta D_{\infty}$ respectively. Thus, Eq. (9.2) reduces to

$$\Delta D_{ultrasonic} = \sum_{COF} \Delta D_{COF} \approx \Delta D_{0.5} + \Delta D_{10} + \Delta D_{\infty} \tag{9.3}$$

For each load case, damage is accumulated through two cycles and then extrapolated to compute damage accumulation in n_{COF} cycles. Assume damage during the

first two cycles to be D_1 and D_2 respectively. Let D_{impact} be the initial damage accumulated from the impact stage. Then, damage accumulation ΔD_1 and ΔD_2 during the two cycles is given by

$$\Delta D_1 = D_1 - D_{impact}$$
$$\Delta D_2 = D_2 - D_1$$

The rate of damage accumulation per cycle $\frac{d(\Delta D)}{dN}$ is approximated as follows

$$\frac{d(\Delta D)}{dN} = \Delta D_2 - \Delta D_1$$

Further, the damage accumulated for the load cases in corresponding n_{COF} cycles is extrapolated as

$$\Delta D_{COF} = \sum_i \Delta D_i = n_{COF} \left[\Delta D_1 + \frac{(n_{COF} - 1)}{2} \frac{d(\Delta D)}{dN} \right] \tag{9.4}$$

In this step, weak interfaces and potential sites for crack nucleation during ultrasonic excitation cycles are identified. Further, comparative severity of impact against ultrasonic excitation process steps is evaluated. Moreover, the total damage accumulation through the process steps for each ILD stack design is evaluated to compare the designs for resistance to fracture.

9.5 Results and Discussion

In this section, the results of the studies performed using the damage analysis framework are discussed.

9.5.1 Critical Stresses in SUP for the Load Cases

In reference [156], the authors showed that the out-of-plane stress component σ_{zz} (with x-y plane representing the die plane) is the main driving force for the fracture in ILD stacks during flip-chip assembly (i.e., for chip-package interaction failures). Based on this physical insight as well as the observation that the ILD stack would experience a dominant compressive stress perpendicular to die plane (in z-direction)

when the capillary tip impacts the bond pad, the σ_{zz} component was chosen as the criterion for identifying the critical step during impact stage. The critical state of σ_{zz} corresponding to each load case is plotted in Fig. 9.5 (obtained from the global FE model [155]). The displacements on nodes along the external boundary of the x-z plane (see Fig. 9.5(a)) were extracted and converted into the traction (on top side) and displacement (on left, right and bottom sides) boundary conditions on the local 2-D ILD stack model. Using these boundary conditions, the damage analysis study was carried out.

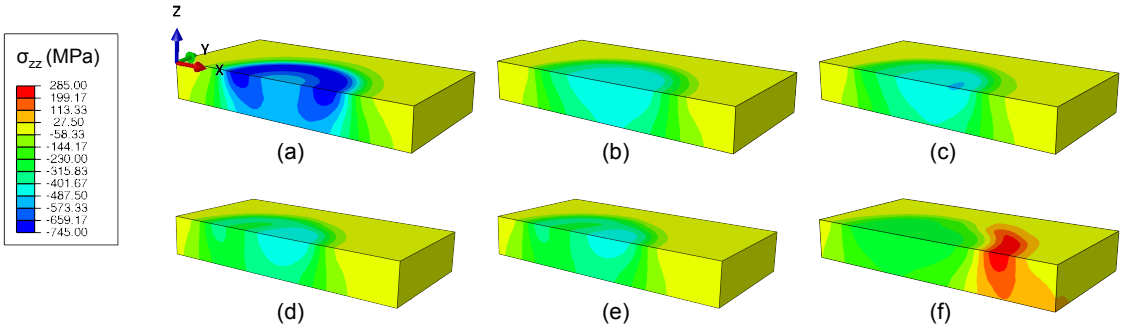


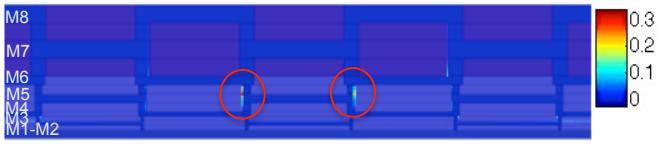
Figure 9.5. Critical states of out-of-plane stress σ_{zz} in the under pad ILD stack corresponding to the simulated wirebond process conditions: (a) Impact without prebleed (b) Impact with prebleed of 25% excitation amplitude (c) Impact with prebleed of 50% excitation amplitude (d) First cycle of ultrasonic excitation (e) Second cycle of ultrasonic excitation (f) Last cycle of ultrasonic excitation.

9.5.2 Damage Nucleation Patterns and Critical Process Step

For the damage model, the value of separation parameters δ_0 and δ_c in the bilinear cohesive damage description were chosen to be 0.1 nm and 1 nm respectively. In the absence of experimental values for the parameters, the damage values in the work give a relative measure of risk of fracture in ILD layer and potential crack initiation sites.

Impact Stage

Damage nucleation patterns during the impact stage for the baseline interconnect structure are shown in Fig. 9.6. It is observed that the vertical interfaces through SiC-Ta-ILD tri-junction in ULK layers are susceptible to damage for impact load with no prebleed (see Fig. 9.6(a)). Further, for the impact load cases with prebleed condition, horizontal interfaces through SiC-Ta-ILD tri-junction in ULK layers are more susceptible to damage than the vertical interfaces (see Fig. 9.6(b)). Maximum damage indeed occurs at the tri-material corners SiC-Ta-ILD as predicted by the asymptotic analysis of strength of singularities [156]. The damage in impact stage is



(a)

(b)

Figure 9.6. Predicted damage in ILD stacks during impact load cases. (a) Impact only, (b) Impact with prebleed.

accumulated to the ultrasonic excitation stage using the irreversible damage law as discussed in Section 9.4.1.

Ultrasonic Excitation

Damage nucleation patterns during ultrasonic vibration process for the baseline interconnect structure are shown in Fig. 9.7. It is observed that the horizontal

interfaces in ULK layers through SiC-Ta-ILD tri-junction are susceptible to damage. It is assumed that an initial damage D_{impact} is accumulated from impact stage at the beginning of this step. It is observed that the maximum damage at the end of the ultrasonic excitation step is the same as that in the impact stage. The interfaces damaged during impact stage are not damaged further in the ultrasonic excitation stage. However, some damage is accumulated during the ultrasonic excitation stage at other interfaces as shown in Fig. 9.7. Hence, it can be concluded that the impact loading is more severe than the ultrasonic excitation loading. The maximum damage indeed occurs at the tri-material corners SiC-Ta-ILD as predicted by the asymptotic analysis of strength of singularities [156].

(a)

(b)

Figure 9.7. Damage accumulation in ILD stack during ultrasonic excitation cycles. (a) Cycle 1, (b) Cycle 2.

9.5.3 Effect of Prebleed

In this study, damage is modeled in the baseline ILD stack for the three impact load cases for the same constitutive behavior of Cu FAB. The maximum damage results are shown in Table 9.4. It is observed that the maximum damage increases by

Table 9.4. Maximum damage during impact load cases.

Load Case	Maximum Damage
Impact only	0.3427
Impact + 25% Prebleed	0.4290
Impact + 50% Prebleed	0.4412

adding the prebleed excitation and increasing the amplitude of excitation. However, the prebleed ultrasonic excitation during impact stage is known to soften the Cu FAB, thus, reducing the impact force needed for bonding it to the pad. Hence, intuitively the damage in the ILD stack should reduce with prebleed excitation which is contrary to the observation. This implies that the current constitutive model is unable to capture the softening behavior of Cu FAB. Further, experimental characterization of the softening behavior of Cu FAB do not seem to appear in literature. In the absense of such a constitutive model of Cu FAB that captures the softening behavior, there is a need for a modeling strategy to induce the softening behavior in the model. Such a modeling strategy is studied here. In an alternate study, damage is again modeled in the baseline ILD stack for the three impact load cases but with reduced yield strength of Cu to model the softening behavior. The % reduction in yield strength of Cu and corresponding maximum damage results are shown in Table 9.5. It is observed

Table 9.5. Effect of softening on maximum damage.

Prebleed Load Case	% Reduction in Yield Strength of Cu to Model Softening	Maximum Damage
25% Prebleed	10	0.3442
50% Prebleed	20	0.2360

that such a modeling strategy for softening behavior does result in a decrease in the

maximum damage in ILD stack. Such an insight is useful for developing constitutive model of Cu FAB.

9.5.4 Comparision of Representative Dielectric Layer Configurations

The damage accumulation in the three ILD stack configurations are compared in this section. The damage accumulated in impact stage for impact with 50% load case is computed for each stack and accumulated to the ultrasonic excitation stage through the irreversible damage law.

Damage accumulation is calculated for two cycles of each load case of the ultrasonic excitation stage with varying friction coefficients. The rate of change of maximum damage accumulation $\frac{d(\Delta D)}{dN}$ is obtained from the two cycles and used to extrapolate damage to n_{COF} cycles. It is observed that damage accumulation reduces in each cycle and stops after some cycles. First, the total number of cycles n_{COF} upto which damage will be accumulated is calculated and then using Eq. (9.4), the total damage accumulation is calculated for each load case. Finally, the damage accumulation for the ultrasonic excitation cycles is computed using Eq. (9.3). The damage accumulation during the ultrasonic excitation stage for the three ILD stack configurations is shown in Table 9.6.

Based on the damage analysis for impact and ultrasonic excitation stages, it is concluded that the damage accumulation reduces with increase in % Cu in the ILD stack designs. Further, for each design, damage accumulation is higher in impact stage as compared to ultrasonic excitation stage. The comparison is shown in Table 9.7.

Table 9.6. Incremental damage accumulation during ultrasonic excitation cycles.

COF	ΔD_1	ΔD_2	$\frac{d(\Delta D)}{dN}$	ΔD_{COF}
Low Cu Stack				
0.5	0.1833	$6.5e^{-04}$	-0.1826	0.184
10	0.1793	$8.1e^{-04}$	-0.1785	0.18
∞	0.2101	$8.7e^{-04}$	-0.2092	0.2110
Nominal Cu Stack				
0.5	0.0963	$3.2e^{-04}$	-0.096	0.096
10	0.0877	0.0046	-0.0831	0.093
∞	.0989	0.001	-0.0979	.0999
High Cu Stack				
0.5	$3.2e^{-04}$	0	$-3.2e^{-04}$	$3.2e^{-04}$
10	$4.9e^{-08}$	0	$-.4.9e^{-08}$	$.4.9e^{-08}$
∞	$.4.9e^{-08}$	0	$-.4.9e^{-08}$	$.4.9e^{-08}$

Table 9.7. Comparison of damage in candidate structures.

	Maximum damage accumulated in Impact Stage	Maximum damage accumulated in Ultrasonic Excitation Cycles
Low Cu Stack	1	0.5751
Nominal Cu Stack	0.4412	0.2888
High Cu Stack	0.0212	$3.2e^{-4}$

10. CLOSURE

10.1 Summary and Novel Contributions

The main theme of this work was to develop a CAD/CAE integration technique that enables direct analysis on B-rep CAD models obtained from solid modeling kernels. To begin with, a survey of computational techniques attempting to address the issue of CAD/CAE was presented. One or more of the geometric challenges of mesh generation, exactness of geometric representation in analysis, point containment queries and surface-surface intersection operations were recognized as the bottlenecks in current CAD/CAE integration techniques. In this work, a computational technique that enables CAD/CAE integration while addressing the above mentioned challenges was proposed.

First, a purely algebraic technique for computing unsigned distance measures from complex non-linear boundaries was proposed. The procedure was based on the fact that the resultant of a parametric entity obtained by algebraic implicitization possesses the properties of distance. The theoretical development of the resultant based implicit function to construct distance measures for the parametric geometry was shown using normalization techniques and boolean operations. Algorithms were developed for algebraic distance field construction for NURBS and Bezier geometries. The proposed technique overcomes the need for iterative numerical distance computations at every quadrature point during isogeometric analysis while enabling greater smoothness of the field and robustness in the distance estimation. Next, the theoretical concepts behind the unsigned algebraic distance field were applied to develop algorithms for sign calculation for bounded B-rep solids. This algebraic signed distance measure was termed as algebraic level set. The algebraic level set was illustrated on complex geometries such as the utah teapot. The algebraic level set was further

shown to enable implicit boolean operations as opposed to numerical surface intersection computations.

Further, a geometry-based analysis scheme was presented that enables tighter integration between CAD and CAE processes in the design cycle. This scheme relies on the HPFC theory and enriched field modeling technique to explicitly model the B-rep boundaries with known behavior and construct global compositions by blending the approximations on B-rep boundary with those on an underlying domain. In this technique, the weight functions were typically constructed using algebraic level sets which facilitates computationally efficient analysis of complex geometries while maintaining the geometric exactness of the boundaries to capture the behavior correctly. Further, all point classification queries and boolean compositions were carried out using the algebraic level sets. The analysis scheme was demonstrated using linear elasticity problems.

The applications of algebraic level sets in modeling fracture propagation was demonstrated using the enriched field modeling technique. Further, the proposed techniques were applied in a study to assess the risk of fracture of ULK dielectrics in ILD stacks during wire bonding process.

10.2 Future Research Directions

10.2.1 Algebraic Level Set based Geometry Evolution

Geometry evolution using the algebraic level set is a natural extension to this thesis. Geometry evolution is fundamental to moving boundary problems such as crack propagation, phase evolution and evolving geometries in design problems. In most of the existing techniques, such moving boundaries are represented implicitly and evolved using level set methods or phase field methods. Such techniques are both geometrically and behaviorally implicit. However, the work in this thesis is based on an explicit description of boundaries. A detailed comparison of implicit versus explicit techniques is discussed in [2]. For evolution of explicit geometry, development of

robust geometry evolution algorithms that can detect and handle topological changes is essential. Further, while evolving a NURBS boundary, there might also be a need for adding or removing the control points in the evolved boundary to capture the intricate details of the new geometry [163, 164].

10.2.2 Modeling Three-Dimensional(3D) Crack Propagation using Algebraic Level Set

In this work, fracture examples are demonstrated both in two and three dimensions using the proposed algebraic level sets applied within an enriched isogeometric analysis scheme. The current work can be further extended to 3D implementation of crack propagation using algebraic level sets. One of the difficulties in modeling 3D crack propagation is how to describe and track the crack surface such that crack path continuity is maintained. Further, most of the current algorithms assume C_0 continuous crack surface and are not able to model higher order continuity. Another difficulty is a suitable 3D fracture criterion to be used for crack propagation [165,166].

Majority of the current 3D crack propagation algorithms rely on an implicit description of crack surface and propagation using level set methods [92, 127, 167, 168]. Recently, an explicit representation of crack as polyhedron and explicit propagation was proposed in order to overcome the challenges associated with level set evolution methods [169]. However, this technique assumes a linear approximation of the crack surface and is exact only in the limit of refinement. Instead, an explicit description of crack using a NURBS representation allows to capture and propagate the geometry exactly as demonstrated in 2D examples. For extension in three-dimensions, there are two possible strategies for modeling crack propagation-

- Choose n discrete points on the boundary of the current crack surface. For each point, find the corresponding point on the boundary of the new crack surface using local crack propagation criterion. Construct a NURBS fit to the new

points. Create an interpolation between the old boundary and new boundary to extend the original crack surface.

- The second method involves either solving for the unknown control points of the new crack boundary or solving for the crack propagation direction and step length at each control point of the original crack surface boundary using global crack propagation criteria. This approach gets its inspiration from global energy minimization techniques used for crack propagation [170].

Other characteristics of geometry evolution algorithms discussed in the previous section need to be considered as well.

10.2.3 Quadrature Schemes for Analysis using Algebraic Level Sets

The analysis techniques proposed in Chapter 7 are based on analysis approximations distinct from geometry. In such a case, the sign of algebraic level set of the B-rep solid enables identification of quadrature points outside the solid which are ignored during the analysis. However, since control points near the boundary of the solid do not exactly conform to the boundary, they have local support that extends beyond the boundary. That is, for some interior control points their support will extend outside the boundary and vice-versa for exterior control points. Hence, the field values obtained from the analysis have inaccuracies near the boundary and the error reduces with refinement. However, instead of refining to capture these boundaries, quadrature schemes can be developed to enable accurate analysis near boundaries. Some explorations with respect to quadrature schemes are presented in [31, 171].

LIST OF REFERENCES

LIST OF REFERENCES

[1] M. Rayasam, V. Srinivasan, and G. Subbarayan. CAD inspired hierarchical partition of unity constructions for NURBS-based, meshless design, analysis and optimization. *International Journal for Numerical Methods in Engineering*, 72(12):1452–1489, December 2007.

[2] A. Tambat and G. Subbarayan. Isogeometric enriched field approximations. *Computer Methods in Applied Mechanics and Engineering*, 245-246:1–21, October 2012.

[3] S.H. Lee. A CADCAE integration approach using feature-based multi-resolution and multi-abstraction modelling techniques. *Computer-Aided Design*, 37(9):941–955, August 2005.

[4] I. Ocana, J. Molinaaldareguia, D. Gonzalez, M. Elizalde, J. Sanchez, J. Martinezesnaola, J. Gilsevillano, T. Scherban, D. Pantuso, and B. Sun. Fracture characterization in patterned thin films by cross-sectional nanoindentation. *Acta Materialia*, 54(13):3453–3462, August 2006.

[5] C.S. Tan, R. Reif, N.D. Theodore, and S. Pozder. Observation of interfacial void formation in bonded copper layers. *Applied Physics Letters*, 87(20):201909, 2005.

[6] M.A. Yerry and M.S. Shephard. Automatic mesh generation for three-dimensional solids. *Computers & Structures*, 20(1-3):31–39, January 1985.

[7] D.A. Field. The legacy of automatic mesh generation from solid modeling. *Computer Aided Geometric Design*, 12(7):651–673, November 1995.

[8] J. Hoschek, D. Lasser, and L.L. Schumaker. *Fundamentals of computer aided geometric design*. AK Peters, Ltd., 1993.

[9] A.A.G. Requicha and J.R. Rossignac. Solid modeling and beyond. *IEEE Computer Graphics and Applications*, 12(5):31–44, September 1992.

[10] C.M. Hoffmann. *Geometric and solid modeling: an introduction*. Morgan Kaufmann Publishers Inc., 1989.

[11] M.J. Pratt and A.D. Geisow. Surface/surface intersection problems. *The mathematics of surfaces*, 6:117–142, 1986.

[12] V. Shapiro, I. Tsukanov, and A. Grishin. Geometric Issues in Computer Aided Design/Computer Aided Engineering Integration. *Journal of Computing and Information Science in Engineering*, 11(2):021005, 2011.

[13] C. Provatidis. A Review on Attempts towards CAD / CAE Integration Using Macroelements. 1(3):61–84, 2013.

[14] O.C. Zienkiewicz, R.L. Taylor, and R.L. Taylor. *The Finite Element Method: The basis*. The Finite Element Method. Butterworth-Heinemann, 2000.

[15] T.J.R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Civil and Mechanical Engineering. Dover Publications, 2000.

[16] B.W. Clark. *Proceedings of the 18th International Meshing Roundtable*. Springer-Verlag Berlin Heidelberg, 2009.

[17] C.M. Hoffmann. Robustness in Geometric Computations. *Journal of Computing and Information Science in Engineering*, 1(2):143–155, 2001.

[18] P. Kagan and A. Fischer. Integrated mechanically based CAE system using B-Spline finite elements. *Computer-Aided Design*, 32(8-9):539–552, August 2000.

[19] M.C. Sabin. *Spline Finite Elements*. PhD thesis, University of Leeds, September 1997.

[20] F.P. Renken and G. Subbarayan. NURBS-based solutions to inverse boundary problems in droplet shape prediction. *Computer Methods in Applied Mechanics and Engineering*, 190(11-12):1391–1406, December 2000.

[21] 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(39-41):4135–4195, October 2005.

[22] L.B. Lucy. A Numerical Approach to Testing the Fission Hypothesis. *The Astronomical Journal*, 82(12):1013–1924, 1977.

[23] M.B. Liu and G.R. Liu. Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments. *Archives of Computational Methods in Engineering*, 17(1):25–76, February 2010.

[24] C.A. Duarte and J.T. Oden. An h-p adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):237–262, December 1996.

[25] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):3–47, December 1996.

[26] L.V. Kantorovich and V.I. Krylov. *Approximate Methods of Higher Analysis*. Interscience, New York, 1958.

[27] V.L. Rvachev, T.I. Sheiko, V. Shapiro, and I. Tsukanov. On completeness of RFM solution structures. *Computational Mechanics*, 25(2-3):305–317, March 2000.

[28] K. Höllig. *Finite Element Methods with B-Splines*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.

[29] M. Freytag, V. Shapiro, and I. Tsukanov. Field modeling with sampled distances. *Computer-Aided Design*, 38(2):87–100, February 2006.

[30] A.V. Kumar, S. Padmanabhan, and R. Burla. Implicit boundary method for finite element analysis using non-conforming mesh or grid. *International Journal for Numerical Methods in Engineering*, 74(9):1421–1447, May 2008.

[31] K. Höllig, U. Reif, and J. Wipper. Weighted Extended B-Spline Approximation of Dirichlet Problems. *SIAM Journal on Numerical Analysis*, 39(2):442–462, January 2001.

[32] C. Hoffmann, V. Shapiro, and V. Srinivasan. Geometric interoperability for resilient manufacturing. Technical Report 11-015, Purdue University,Computer Science Dept.,Cornell Univ., Ithaca, New York, 2011.

[33] C.A. Brebbia. *Boundary element methods in engineering: proceedings of the fourth international seminar, Southampton, England, September 1982*. Boundary Elements Series. Springer, 1982.

[34] M.S. Casale. Free-Form Solid Modeling with Trimmed Surface Patches. *IEEE Computer Graphics and Applications*, 7(1):33–43, January 1987.

[35] M.S. Casale, J.E. Bobrow, and R. Underwood. Trimmed-patch boundary elements: bridging the gap between solid modeling and engineering analysis. *Computer-Aided Design*, 24(4):193–199, April 1992.

[36] K. Li and X. Qian. Isogeometric analysis and shape optimization via boundary integral. *Computer-Aided Design*, 43(11):1427–1437, November 2011.

[37] R.N. Simpson, S.P.a. Bordas, J. Trevelyan, and T. Rabczuk. A two-dimensional Isogeometric Boundary Element Method for elastostatic analysis. *Computer Methods in Applied Mechanics and Engineering*, 209-212:87–100, February 2012.

[38] 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, January 2010.

[39] O. T. Morgan. *HIGEOM: A Symbolic Framework for a Unified Function Space Representation of Geometry and Attributes of Solids*. PhD thesis, Purdue University, April 2011.

[40] T. Martin, E. Cohen, and R.M. Kirby. Mixed-element volume completion from NURBS surfaces. *Computers & Graphics*, 36(5):548–554, August 2012.

[41] Y. Zhang, W. Wang, and T.J.R. Hughes. Conformal solid T-spline construction from boundary T-spline representations. *Computational Mechanics*, 51(6):1051–1059, September 2012.

[42] D. Natekar, X. Zhang, and G. Subbarayan. Constructive solid analysis: a hierarchical, geometry-based meshless analysis procedure for integrated design and analysis. *Computer-Aided Design*, 36(5):473–486, April 2004.

[43] G.A. Kriezis, P.V. Prakash, and N.M. Patrikalakis. Method for intersecting algebraic surfaces with rational polynomial patches. *Computer-Aided Design*, 22(10):645–654, December 1990.

[44] S. Krishnan and D. Manocha. An efficient surface intersection algorithm based on lower-dimensional formulation. *ACM Transactions on Graphics*, 16(1):74–106, January 1997.

[45] J.M. Lane and R.F. Riesenfeld. Bounds on a polynomial. *BIT*, 21(1):112–117, March 1981.

[46] T. Dokken. Finding intersections of B-spline represented geometries using recursive subdivision techniques. *Computer Aided Geometric Design*, 2(1-3):189–195, September 1985.

[47] E.G. Houghton, R.F. Emnett, J.D. Factor, and C.L. Sabharwal. Implementation of a divide-and-conquer method for intersection of parametric surfaces. *Computer Aided Geometric Design*, 2(1-3):173–183, September 1985.

[48] P. Sinha, E. Klassen, and K.K. Wang. Exploiting topological and geometric properties for selective subdivision. In *Proceedings of the first annual symposium on Computational geometry - SCG '85*, pages 39–45, New York, New York, USA, 1985. ACM Press.

[49] Jaroslaw R. Rossignac and Aristides A G Requicha. Piecewise-circular curves for geometric modeling. *IBM Journal of Research and Development*, 31(3):296–313, May 1987.

[50] C.L. Bajaj, C.M. Hoffmann, R.E. Lynch, and J.E.H. Hopcroft. Tracing surface intersections. *Computer Aided Geometric Design*, 5(4):285–307, November 1988.

[51] G.A. Kriezis, N.M. Patrikalakis, and F.-E. Wolter. Topological and differential-equation methods for surface intersections. *Computer-Aided Design*, 24(1):41–55, January 1992.

[52] R.E. Barnhill and S.N. Kersey. A marching method for parametric surface/surface intersection. *Computer Aided Geometric Design*, 7(1-4):257–280, June 1990.

[53] I. Babuska and J.M. Melenk. the Partition of Unity Method. *International Journal for Numerical Methods in Engineering*, 40(4):727–758, February 1997.

[54] J.A. Sethian. Fast Marching Methods. *SIAM Review*, 41(2):199–235, 1999.

[55] A. Rosenfeld and J.L. Pfaltz. Sequential Operations in Digital Picture Processing. *Journal of the ACM*, 13(4):471–494, October 1966.

[56] A. Legay, J. Chessa, and T. Belytschko. An EulerianLagrangian method for fluidstructure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2070–2087, March 2006.

[57] T. Rüberg and F. Cirak. An immersed finite element method with integral equation correction. *International Journal for Numerical Methods in Engineering*, 86(1):93–114, April 2011.

[58] G. Hirota, S. Fisher, A. State, C. Lee, and H. Fuchs. An implicit finite element method for elastic solids in contact. In *Proceedings Computer Animation 2001. Fourteenth Conference on Computer Animation (Cat. No.01TH8596)*, pages 136–254. IEEE Comput. Soc, 2001.

[59] J. Bruchon, H. Digonnet, and T. Coupez. Using a signed distance function for the simulation of metal forming processes: Formulation of the contact condition and mesh adaptation. From a Lagrangian approach to an Eulerian approach. *International Journal for Numerical Methods in Engineering*, 78(8):980–1008, May 2009.

[60] S. Wolff and C. Bucher. Distance fields on unstructured grids: Stable interpolation, assumed gradients, collision detection and gap function. *Computer Methods in Applied Mechanics and Engineering*, 259:77–92, June 2013.

[61] A. Biswas, V. Shapiro, and I. Tsukanov. Heterogeneous material modeling with distance fields. *Computer Aided Geometric Design*, 21(3):215–242, March 2004.

[62] X.Y. Kou and S.T. Tan. Heterogeneous object modeling: A review. *Computer-Aided Design*, 39(4):284–301, April 2007.

[63] M.E. Mortenson. *Geometric Modeling*. Wiley, New York, 1985.

[64] L.A. Piegl and W. Tiller. *The NURBS Book*. Springer, 1997.

[65] Y.L. Ma and W.T. Hewitt. Point inversion and projection for NURBS curve and surface: Control polygon approach. *Computer Aided Geometric Design*, 20(2):79–99, May 2003.

[66] X.-D. Chen, J.-H. Yong, G. Wang, J.-C. Paul, and G. Xu. Computing the minimum distance between a point and a NURBS curve. *Computer-Aided Design*, 40(10-11):1051–1054, October 2008.

[67] M.W. Jones, J.A. Baerentzen, and M. Sramek. 3D distance fields: a survey of techniques and applications. *IEEE transactions on visualization and computer graphics*, 12(4):581–99, 2006.

[68] B.A. Payne and A.W. Toga. Distance field manipulation of surface models. *IEEE Computer Graphics and Applications*, 12(1):65–71, January 1992.

[69] A. Guezlec. "Meshsweeper": dynamic point-to-polygonal mesh distance and applications. *IEEE Transactions on Visualization and Computer Graphics*, 7(1):47–61, 2001.

[70] M. Sean. A fast algorithm for computing the closest point and distance transform. Technical Report caltechASCI/2000.077, Applied and Computational Math., Calif. Inst. of Technology, 2000.

[71] R. Satherley and M.W. Jones. Vector-City Vector Distance Transform. *Computer Vision and Image Understanding*, 82(3):238–254, June 2001.

[72] A. Biswas and V. Shapiro. Approximate distance fields with non-vanishing gradients. *Graphical Models*, 66(3):133–159, May 2004.

[73] T.W. Sederberg. *Implicit and parametric curves and surfaces for computer aided geometric design*. PhD thesis, Purdue, 1983.

[74] A.L. Dixon. The Eliminant of Three Quantics in two Independent Variables. *Proceedings of the London Mathematical Society*, s2-7(1):49–69, January 1909.

[75] V. Shapiro. Theory of r-functions and applications: A primer. Technical Report TR91-1219, Computer Science Dept.,Cornell Univ., Ithaca, New York, 1991.

[76] V. Shapiro and I. Tsukanov. Implicit functions with guaranteed differential properties. In *Proceedings of the fifth ACM symposium on Solid modeling and applications - SMA '99*, number 1, pages 258–269, New York, New York, USA, 1999. ACM Press.

[77] E. Cohen, T. Lyche, and R. Riesenfeld. Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics. *Computer Graphics and Image Processing*, 14(2):87–111, October 1980.

[78] W. Boehm and H. Prautzsch. The insertion algorithm. *Computer-Aided Design*, 17(2):58–59, March 1985.

[79] R. Holmes. The Utah Teapot. `http://www.holmes3d.net/graphics/teapot/`. [Online; accessed June 2012].

[80] S. Abhyankar. *Algebraic geometry for scientists and engineers*. American Mathematical Society, 1990.

[81] K.M. Chen, K.H. Tang, and J.S. Liu. Reliability evaluation of BOAC and normal pad stacked-chip packaging using low-K wafers. *Microelectronics Reliability*, 48(3):408–415, March 2008.

[82] X. Jia and R. Goldman. μ-Bases and singularities of rational planar curves. *Computer Aided Geometric Design*, 26(9):970–988, December 2009.

[83] J. Zheng and T.W. Sederberg. A Direct Approach to Computing theμ-basis of Planar Rational Curves. *Journal of Symbolic Computation*, 31(5):619–629, May 2001.

[84] R.J. Walker. *Algebraic curves*. Princeton University Press, 1950.

[85] T.W. Sederberg and F. Chen. Implicitization using moving curves and surfaces. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques - SIGGRAPH '95*, pages 301–308, New York, New York, USA, 1995. ACM Press.

[86] T. Belytschko, W.J.T. Daniel, and G. Ventura. A monolithic smoothing-gap algorithm for contact-impact based on the signed distance function. *International Journal for Numerical Methods in Engineering*, 55(1):101–125, September 2002.

[87] M.Y. Wang, X. Wang, and D. Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1-2):227–246, January 2003.

[88] M.Y. Wang and X. Wang. Color level sets: a multi-phase method for structural topology optimization with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 193(6-8):469–496, February 2004.

[89] S. Chen, B. Merriman, S. Osher, and P. Smereka. A simple level set method for solving stefan problems. *Journal of Computational Physics*, 135(1):8–29, July 1997.

[90] N. Zabaras, B. Ganapathysubramanian, and L. Tan. Modelling dendritic solidification with melt convection using the extended finite element method. *Journal of Computational Physics*, 218(1):200–227, October 2006.

[91] J.A. Sethian and P. Smereka. Level Set Methods for Fluid Interfaces. *Annual Review of Fluid Mechanics*, 35(1):341–372, January 2003.

[92] A. Gravouil, N. Moes, and T. Belytschko. Non-planar 3D crack growth by the extended finite element and level sets - Part II: Level set update. *International Journal for Numerical Methods in Engineering*, 53(11):2569–2586, April 2002.

[93] N. Sukumar, D.L. Chopp, and B. Moran. Extended finite element method and fast marching method for three-dimensional fatigue crack propagation. *Engineering Fracture Mechanics*, 70(1):29–48, January 2003.

[94] M. Duflot. A study of the representation of cracks with level sets. *International Journal for Numerical Methods in Engineering*, 70(11):1261–1302, June 2007.

[95] A. Fuhrmann, G. Sobotka, and C. Groß. Distance fields for rapid collision detection in physically based modeling. In *Proceedings of GraphiCon 2003*, 2003.

[96] S.F. Frisken, R.N. Perry, A.P. Rockwood, and T.R. Jones. Adaptively sampled distance fields. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH '00*, pages 249–254, New York, New York, USA, 2000. ACM Press.

[97] D. Cohen-Or, A. Solomovic, and D. Levin. Three-dimensional distance field metamorphosis. *ACM Transactions on Graphics*, 17(2):116–141, April 1998.

[98] javaGeom - A geometry library for Java. `http://sourceforge.net/projects/geom-java/`.

[99] F. Feito, J.C. Torres, and A. Ureña. Orientation, simplicity, and inclusion test for planar polygons. *Computers & Graphics*, 19(4):595–600, July 1995.

[100] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics: Principles and Practice in C*. Pearson, 2nd edition, 1995.

[101] R.E. Barnhill, G. Farin, M. Jordan, and B.R. Piper. Surface/surface intersection. *Computer Aided Geometric Design*, 4(1-2):3–16, July 1987.

[102] N.M. Patrikalakis. Surface-to-surface intersections. *IEEE Computer Graphics and Applications*, 13(1):89–95, January 1993.

[103] M.E. Mortenson. *Geometric Modeling*. John Wiley & Sons, Inc., New York, NY, USA, 1985.

[104] N.M. Patrikalakis and T. Maekawa. Intersection problems. In *Shape Interrogation for Computer Aided Design and Manufacturing*, pages 109–160. Springer, 2010.

[105] H.B. Voelcker and A.A.G. Requicha. Geometric Modeling of Mechanical Parts and Processes. *Computer*, 10(12):48–57, December 1977.

[106] A.A.G. Requicha and H.B. Voelcker. Solid Modeling: Current Status and Research Directions. *IEEE Computer Graphics and Applications*, 3(7):25–37, October 1983.

[107] A.A.G. Requicha and H.B. Voelcker. Boolean operations in solid modeling: Boundary evaluation and merging algorithms. *Proceedings of the IEEE*, 73(1):30–44, 1985.

[108] J. Rossignac, A. Megahed, and B.-O. Schneider. Interactive inspection of solids: Cross-sections and Interferences. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques - SIGGRAPH '92*, number July, pages 353–360, New York, New York, USA, 1992. ACM Press.

[109] B.K. Choi and C.S. Jun. Ball-end cutter interference avoidance in NC machining of sculptured surfaces. *Computer-Aided Design*, 21(6):371–378, July 1989.

[110] J.S. Hwang. Interference-free tool-path generation in the NC machining of parametric compound surfaces. *Computer-Aided Design*, 24(12):667–676, December 1992.

[111] D.C.H. Yang and Z. Han. Interference detection and optimal tool selection in 3-axis NC machining of free-form surfaces. *Computer-Aided Design*, 31(5):303–315, April 1999.

[112] S. Ding, M.A. Mannan, and A.N. Poo. Oriented bounding box and octree based global interference detection in 5-axis machining of free-form surfaces. *Computer-Aided Design*, 36(13):1281–1294, November 2004.

[113] T.W. Sederberg, D.C. Anderson, and R.N. Goldman. Implicitization, inversion, and intersection of planar rational cubic curves. *Computer Vision, Graphics, and Image Processing*, 31(1):89–102, 1985.

[114] R.N. Goldman. The method of resolvents: A technique for the implicitization, inversion, and intersection of non-planar, parametric, rational cubic curves. *Computer Aided Geometric Design*, 2(4):237–255, December 1985.

[115] R.N. Goldman and T.W. Sederberg. Analytic approach to intersection of all piecewise parametric rational cubic curves. *Computer-Aided Design*, 19(6):282–292, July 1987.

[116] G.E. Farin, J. Hoschek, and M.-S. Kim. *Handbook of computer aided geometric design [electronic resource]*. North Holland, 2002.

[117] L.A. Piegl. *Fundamental developments of computer-aided geometric modeling*. Academic Pr, 1993.

[118] V. Srinivasan, G. Subbarayan, S. Radhakrishnan, D. Pantuso, and S. Shankar. Hierarchical partition of unity field compositions (HPFC) for optimal design in the presence of cracks. *Mechanics of Advanced Materials and Structures*, 17(7):467–480, October 2010.

[119] X. Zhang, M. Rayasam, and G. Subbarayan. A meshless, compositional approach to shape optimal design. *Computer Methods in Applied Mechanics and Engineering*, 196(17-20):2130–2146, March 2007.

[120] W. Pilkey. *Peterson's stress concentration factors, second ed.* New York: Wiley-Interscience, 1997.

[121] M.H. Sadd. *Elasticity: Theory, Applications, and Numerics*. Elsevier/AP, 2009.

[122] J. Dolbow and T. Belytschko. Numerical integration of the Galerkin weak form in meshfree methods. *Computational Mechanics*, 23(3):219–230, April 1999.

[123] K. Upreti, T. Song, A. Tambat, and G. Subbarayan. Algebraic distance estimations for enriched isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 280:28–56, October 2014.

[124] W. Karush and G. Young. Temperature Rise in a Heat-Producing Solid behind a Surface Defect. *Journal of Applied Physics*, 23(11):1191, 1952.

[125] F.L. Stazi, E. Budyn, J. Chessa, and T. Belytschko. An extended finite element method with higher-order elements for curved cracks. *Computational Mechanics*, 31(1-2):38–48, May 2003.

[126] E.E. Gdoutos. *Fracture Mechanics: An Introduction*. Solid Mechanics and Its Applications. Springer, 2006.

[127] N. Moës, A. Gravouil, and T. Belytschko. Non-planar 3D crack growth by the extended finite element and level sets-Part I: Mechanical model. *International Journal for Numerical Methods in Engineering*, 53(11):2549–2568, April 2002.

[128] G. Wang, C. Merrill, J.-H. Zhao, S.K. Groothuis, and P.S. Ho. Packaging effects on reliability of Cu/low-¡IT¿k¡/IT¿ interconnects. *IEEE Transactions on Device and Materials Reliability*, 3(4):119–128, December 2003.

[129] S. Narasimalu. Wire bond challenges in low-¡IT¿k¡/IT¿ devices. *Microelectronics International*, 25(1):34–40, 2008.

[130] Z. Wang, V. Huang, S. Yao, Z. He, Y.W. Jiang, C.L. Zhang, and P. Cao. Cu low k device wire bonding process modeling. In *8th International Conference on Electronic Packaging Technology, ICEPT 2007.*, volume 00, pages 1–6. IEEE, August 2007.

[131] Y. Liu, S. Irving, and T. Luk. Thermosonic wire bonding process simulation and bond pad over active stress analysis. In *2004 Proceedings. 54th Electronic Components and Technology Conference (IEEE Cat. No.04CH37546)*, volume 1, pages 383–391. IEEE, 2004.

[132] Z.W. Zhong. Wire bonding using copper wire. *Microelectronics International*, 26(1):10–16, 2009.

[133] I. Singh, J.Y. On, and L. Levine. Enhancing fine pitch, high I/O devices with copper ball bonding. In *Proceedings Electronic Components and Technology, 2005. ECTC '05.*, volume 2, pages 843–847. IEEE, 2005.

[134] A. Shah, M. Mayer, Y. Zhou, S.J. Hong, and J.T. Moon. In situ ultrasonic force signals during low-temperature thermosonic copper wire bonding. *Microelectronic Engineering*, 85(9):1851–1857, September 2008.

[135] G. Heinen, R.J. Stierman, D. Edwards, and L. Nye. Wire bonds over active circuits. In *1994 Proceedings. 44th Electronic Components and Technology Conference*, pages 922–928. IEEE, 1994.

[136] K.M. Chen, B.C. Wu, K.H. Tang, F.Y. Cheng, N.H. Kao, and J.Y. Lai. An investigation into the effects of probing and wire bonding stress on the reliability of BOAC. *Microelectronics and Reliability*, 46(2-4):335–342, February 2006.

[137] Y. Luo and G. Subbarayan. A study of multiple singularities in multi-material wedges and their use in analysis of microelectronic interconnect structures. *Engineering Fracture Mechanics*, 74(3):416–430, February 2007.

[138] H. Shouyu, H. Chunjing, and W. Chunqing. Experimental Research of Copper Wire Ball Bonding. *2005 6th International Conference on Electronic Packaging Technology*, pages 1–5, 2005.

[139] A. Shah, M. Mayer, Y. Zhou, S. J. Hong, and J. T. Moon. Reduction of underpad stress in thermosonic copper ball bonding. In *2008 58th Electronic Components and Technology Conference*, pages 2123–2130. IEEE, May 2008.

[140] A. Shah, M. Mayer, Y.N. Zhou, S.J. Hong, and J.T. Moon. Low-Stress Thermosonic Copper Ball Bonding. *IEEE Transactions on Electronics Packaging Manufacturing*, 32(3):176–184, July 2009.

[141] A. Shah, A. Rezvani, M. Mayer, Y. Zhou, J. Persic, and J.T. Moon. Reduction of ultrasonic pad stress and aluminum splash in copper ball bonding. *Microelectronics Reliability*, 51(1):67–74, January 2011.

[142] T.A. Tran, C.-C. Lee, V. Mathew, and L. Higgins. Copper wire bonding on low-k/copper wafers with Bond Over Active (BOA) structures for automotive customers. *2011 IEEE 61st Electronic Components and Technology Conference (ECTC)*, pages 1508–1515, May 2011.

[143] I. Qin, A. Shah, C. Huynh, M. Meyer, M. Mayer, and Y. Zhou. Role of process parameters on bondability and pad damage indicators in copper ball bonding. *Microelectronics Reliability*, 51(1):60–66, January 2011.

[144] J. Chen, D. Degryse, P. Ratchev, and I. DeWolf. Mechanical Issues of Cu-to-Cu Wire Bonding. *IEEE Transactions on Components and Packaging Technologies*, 27(3):539–545, September 2004.

[145] D. Degryse, B. Vandevelde, and E. Beyne. Mechanical FEM Simulation of Bonding Process on Cu Lowk Wafers. *IEEE Transactions on Components and Packaging Technologies*, 27(4):643–650, December 2004.

[146] D. Degryse, B. Vandevelde, and E. Beyne. FEM study of deformation and stresses in copper wire bonds on Cu lowK structures during processing. In *2004 Proceedings. 54th Electronic Components and Technology Conference (IEEE Cat. No.04CH37546)*, pages 906–912. IEEE, 2004.

[147] D. Degryse, B. Vandevelde, and E. Beyne. Cu bonding to Cu lowK wafers: a systematic study of the mechanical bonding process. In *EuroSimE 2005. Proceedings of the 6th International Conference on Thermal, Mechanial and Multi-Physics Simulation and Experiments in Micro-Electronics and Micro-Systems, 2005.*, pages 41–48. IEEE, 2005.

[148] C. Dresbach, G. Lorenz, M. Petzold, and H. Altenbach. Analysis of Chip Damage Risk in Thermosonic Wire Bonding. *Key Engineering Materials*, 478:75–80, April 2011.

[149] H.-C. Hsu, W.-Y. Chang, C.-L. Yeh, and Y.-S. Lai. Characteristic of copper wire and transient analysis on wirebonding process. *Microelectronics Reliability*, 51(1):179–186, January 2011.

[150] V. Kripesh, M. Sivakumar, R. Kumar, and M.K. Iyer. Wire bonding process impact on low-k dielectric material in damascene copper integrated circuits. In *52nd Electronic Components and Technology Conference 2002. (Cat. No.02CH37345)*, pages 873–880. IEEE, 2002.

[151] O. van der Sluis, R.A.B. Engelen, R.B.R. van Silfhout, W.D. van Driel, and M.A.J. van Gils. Efficient damage sensitivity analysis of advanced Cu/low-k bond pad structures by means of the area release energy criterion. *Microelectronics Reliability*, 47(12):1975–1982, December 2007.

[152] V. Fiori, L.T. Beng, S. Downey, S. Gallois-Garreignot, and S. Orain. 3D multi scale modeling of wire bonding induced peeling in Cu/Low-k interconnects: application of an energy based criteria and correlations with experiments. In *Proceedings of IEEE 2007 Electronic Components and Technology Conference (ECTC)*, pages 256–263, 2007.

[153] A. Shah, M. Mayer, Y.N. Zhou, S.J. Hong, and J.T. Moon. Low-stress thermosonic copper ball bonding. *IEEE Transactions on Electronics Packaging Manufacturing*, 32(3):176–184, July 2009.

[154] E. Spaan, E. Ooms, W.D. van Driel, C.A. Yuan, D.G. Yang, and G.Q. Zhang. Wire bonding the future: a combined experimental and numerical approach to improve the Cu-wire bonding quality. In *2010 11th International Thermal, Mechanical & Multi-Physics Simulation, and Experiments in Microelectronics and Microsystems (EuroSimE)*, pages 1–4. IEEE, April 2010.

[155] H.-Y. Lin. *Configurational Optimization for Optimal Topological and Fracture-Resistant Designs of Solids*. PhD thesis, Purdue University, May 2014.

[156] A. Tambat, H.-Y. Lin, G. Subbarayan, D.Y. Jung, and B. Sammakia. Simulations of damage, crack initiation, and propagation in interlayer dielectric structures: understanding assembly-induced fracture in dies. *IEEE Transactions on Device and Materials Reliability*, 12(2):241–254, June 2012.

[157] J. He, Y. Guo, and Z. Lin. Numerical and experimental analysis of thermosonic bond strength considering interfacial contact phenomena. *Journal of Physics D: Applied Physics*, 41(16):165304, August 2008.

[158] H. Huang, B.H. Chang, and D. Du. Effect of superimposed ultrasound on mechanical properties of copper. *Materials Science and Technology*, 27(7):1117–1122, July 2011.

[159] S.W. King and J. Bielefeld. Rigidity percolation in plasma enhanced chemical vapor deposited a-SiC:H thin films. *ECS Transactions*, 33(8):185–194, 2010.

[160] F. Iacopi, Y. Travaly, B. Eyckens, C. Waldfried, T. Abell, E. P. Guyer, D.M. Gage, R.H. Dauskardt, T. Sajavaara, K. Houthoofd, P. Grobet, P. Jacobs, and K. Maex. Short-ranged structural rearrangement and enhancement of mechanical properties of organosilicate glasses induced by ultraviolet radiation. *Journal of Applied Physics*, 99(5):053511, 2006.

[161] Y. Oku, K. Yamada, T. Goto, Y. Seino, A. Ishikawa, T. Ogatal, K. Kohmura, N. Fujii, N. Hata, R. Ichikawa, T. Yoshino, C. Negoro, A. Nakano, Y. Sonoda, S. Takada, H. Miyoshi, S. Oike, H. Tanaka, H. Matsuo, K. Kinoshita, and T. Kikkawa. Novel self-assembled ultra-low-k porous silica films with high mechanical strength for 45 nm BEOL technology. In *IEEE International Electron Devices Meeting 2003*, pages 6.1.1–6.1.4. IEEE, 2003.

[162] R Dauskardt. Adhesion and debonding of multi-layer thin film structures. *Engineering Fracture Mechanics*, 61(1):141–162, August 1998.

[163] Y.-D. Seo, H.-J. Kim, and S.-K. Youn. Isogeometric topology optimization using trimmed spline surfaces. *Computer Methods in Applied Mechanics and Engineering*, 199(49-52):3270–3296, December 2010.

[164] K. Mysore, O.T. Morgan, and G. Subbarayan. NURBS representational strategies for tracking moving boundaries and topological changes during phase evolution. *Computer Methods in Applied Mechanics and Engineering*, 200(33-36):2594–2610, August 2011.

[165] B. Moran and C.F. Shih. Crack tip and associated domain integrals from momentum and energy balance. *Engineering Fracture Mechanics*, 27(6):615–642, January 1987.

[166] M. Gosz and B. Moran. An interaction energy integral method for computation of mixed-mode stress intensity factors along non-planar crack fronts in three dimensions. *Engineering Fracture Mechanics*, 69(3):299–319, February 2002.

[167] T.C. Gasser and G.A. Holzapfel. 3D Crack propagation in unreinforced concrete. *Computer Methods in Applied Mechanics and Engineering*, 195(37-40):5198–5219, July 2006.

[168] T. Rabczuk, S. Bordas, and G. Zi. On three-dimensional modelling of crack growth using partition of unity methods. *Computers & Structures*, 88(23-24):1391–1411, December 2010.

[169] T.-P. Fries and M. Baydoun. Crack propagation with the extended finite element method and a hybrid explicitimplicit crack description. *International Journal for Numerical Methods in Engineering*, 89(12):1527–1558, March 2012.

[170] G. Meschke and P. Dumstorff. Energy-based modeling of cohesive and cohesionless cracks via X-FEM. *Computer Methods in Applied Mechanics and Engineering*, 196(21-24):2338–2357, April 2007.

[171] D. Schillinger, L. Dedè, M.A. Scott, J.A. Evans, M.J. Borden, E. Rank, and T.J.R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-252:116–150, December 2012.

VITA

VITA

Kritika Upreti received her Bachelor of Technology degree in Mechanical Engineering from Indian Institute of Technology, Delhi in Aug 2009. She joined Purdue University for Direct Ph.D. program in Mechanical Engineering in Fall 2009. She was awarded the Ross Fellowship by Purdue University for the academic year 2009-10. She completed her Master of Science in Mechanical Engineering from Purdue University in Aug 2012.

Her research interests are in Computational Mechanics and Computer-Aided Design and Engineering. Her thesis research focused on developing novel algebraic computational techniques for modeling moving boundary problems and reliability of microelectronics packaging.

During the course of her Ph.D. at Purdue, she also got the opportunity to gain industry experience through summer internships at Qualcomm Technologies Inc. in Summer 2013 and Bosch Research in Summer 2014. She will be joining the computational lithography group of Intel Corporation after completing her Ph.D. in 2014.