INTERFACE BALANCE LAWS, GROWTH CONDITIONS AND EXPLICIT INTERFACE MODELING USING ALGEBRAIC LEVEL SETS FOR MULTIPHASE SOLIDS WITH INHOMOGENEOUS SURFACE STRESS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Pavan Kumar Vaitheeswaran

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Prof. Ganesh Subbarayan, Chair
School of Mechanical Engineering

Prof. Marcial Gonzalez
School of Mechanical Engineering

Prof. Hector Gomez
School of Mechanical Engineering

Prof. John Blendell
School of Materials Engineering

Approved by:

Prof. Nicole Key
Associate Head for Graduate Studies

ACKNOWLEDGMENTS

There is an endless list of people that I am indebted to, for helping and shaping me in this long journey. The foremost among them is Prof. Subbarayan. Not only did he guide me in my research work, he also mentored me to grow as a researcher and as an individual. I thank him for all that he has taught me so far, on how to conduct research, on academic writing and for helping me communicate my ideas better. He has been a role model for us, being very patient and readily approachable, taking a personal interest in us, and for simply enjoying teaching students. I also thank Prof. Gonzalez, Prof. Gomez and Prof. Blendell for serving in my committee and for their valuable inputs to my thesis. My gratitude also extends to Purdue, Texas Instruments (TI), Semiconductor Research Corporation (SRC) and Cooling Technologies Research Center (CTRC) for providing financial support.

This journey would not have been possible without the members of HiDAC, both present and past. I thank Anirudh for giving me a warm welcome to the group and for being ready to help even after his graduation. I would also like to thank Tao for helping me get started with research and for his dedication. To Yuvraj I owe thanks for his hard work, for initiating small talk and for his wealth of stories about our group. I would like to thank Chun-Pei for showing perfection in his work and Travis for his warmth, for hosting the group during Thanksgiving and for not failing to wish us with chocolates for the Christmas holidays. I would also like to thank Chung-Shuo, Sukshitha and Huanyu for our collaborations and for being good friends, Yaxiong for our lively conversations and for teaching me rudimentary Mandarin (xiè xie!) and David for his readiness to help. My thanks also extend to Sudarshan for our discussions not just on research but on everything under the sun. I would also like to thank the new members of the group, Sai Sanjit, Colin and Chetan for our light-hearted talks and I wish them all the best in their graduate studies.

There are many outside of work that I owe gratitude to. I would like to thank Subramanian, Meher, Praveen, Raju, Sesha and Vignesh for being there for me at all times. Last but not the least, I would like to thank my parents and sister for giving me unconditional support throughout my life. This would not have been possible without them. They have been right next to me in this journey, from halfway across the world.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

ABSTRACT

Vaitheeswaran, Pavan Kumar PhD, Purdue University, December 2020. Interface Balance Laws, Growth Conditions and Explicit Interface Modeling Using Algebraic Level Sets for Multiphase Solids with Inhomogeneous Surface Stress. Major Professor: Ganesh Subbarayan, School of Mechanical Engineering.

Many natural and engineering phenomena including cancer cell growth, solidification, and crack propagation may be classified as phase evolution problems. These phenomena may have significant physiological or engineering impact, but are difficult to analyze due to the complexity of their interface geometry evolution and the complexity of their governing equations. The goal of this dissertation is to derive the thermodynamic conditions governing the evolution of the interfaces and an efficient computational procedure for modeling the interface evolution. The first contribution towards the goal is to derive the thermodynamic conditions at a moving interface in a body subject to large deformation with multiple diffusing species and arbitrary surface stress. A pillbox procedure is used to form balance laws at an interface, analogous to conventional balance laws in the bulk. The thermodynamic conditions that result from interface free-energy inequality lead to the analytical form of the configurational force for bodies subject to mechanical loads, heat and multiple diffusing species. The derived second law condition naturally extends the Eshelby energy-momentum tensor to include species diffusion terms. The above second law restriction is then used to derive the condition for the growth of new phases in a body undergoing finite deformation subject to inhomogeneous, anisotropic surface stress. Next, a general, finite-deformation, arbitrary surface stress form of phase nucleation condition is derived by considering uncertainty in growth of a small nucleus. The probability of nucleation is shown to naturally depend on a theoretical estimate of critical volu-

metric energy density, which is directly related to the surface stress. The classical nucleation theory is shown to result as a simplified special case.

Towards the goal of computationally simulating evolving interfaces, an interface tracking approach called enriched isogeometric analysis (EIGA) is adopted in this thesis. The phase boundary is represented explicitly using parametric splines, with the physical fields isogeometrically defined on the interface geometry, and immersed in a non-conforming underlying domain. The behavioral field solution at a point is given by a convex blending of the underlying solution and the interface solution. Signed algebraic level sets are used as a measure of distance to model the weakening influence of a phase interface with distance. These level sets are generated from the implicitization of geometries using resultants, and are smooth, monotonic with distance, and exact on the boundary. Furthermore, the sign of these level sets enable classifying points as lying inside or outside a given closed geometry. The generation of these level sets is found to fail often for even simple three-dimensional surfaces, where the Dixon resultant used for implicitization is either identically zero or unsigned. A maximal-rank submatrix approach is adopted in this thesis to recover the implicitization for surfaces with identically zero resultants. Also, a polynomial square root procedure is developed to extract sign from unsigned resultants. The proposed approach is demonstrated on three-dimensional electrostatic and electromigration problems, and is used to simulate electromigration experiments conducted on Copper-TiN line structures. Since EIGA is an explicit interface method, topological changes that are common in phase evolution pose geometric challenges, such as computing intersection between boundaries of coalescent phases. This is overcome in this thesis by using Boolean compositions on algebraic level sets. These compositions can be done algebraically using R-functions, and provide level sets for merged phases. As analysis in EIGA is based only on signed level sets, the Boolean compositions automatically allow EIGA of systems with coalescent phases. Since the compositions automatically generate appropriate signed level sets for merged and separate phases, topological changes can be modeled without needing explicit surface-surface intersection calculation. The ef-

ficiency of this approach is demonstrated on various electrostatic problems of solids with voids, and on a problem of bubble coalescence subject to the constraint of volume conservation.

1. INTRODUCTION

A large and important class of problems is phase evolution under mechanical and non-mechanical loads. Examples of such problems include cancer tissue growth, solidification, electromigration driven void growth, growth of intermetallic compounds in solder joints and crack propagation. Such phase evolution problems have significant physiological and engineering impact. For example, voids nucleating due to electromigration in a current carrying line can grow large enough to sever the line, causing open failure. Similarly, presence and growth of cracks can severely decrease the failure life of parts and components. Phase evolution phenomena such as electromigration are complex to analyze both due to the complexity of their governing equations and due to the complexity of the evolving interface geometry. For example, electromigration is driven by an electric field, thermal gradients, mechanical stresses as well as diffusion of vacancies. Existing analytical methods in literature often assume constitutive relations between fluxes and corresponding driving forces when modeling phase nucleation and growth. However, these constitutive relations are generally empirical and not derived from thermodynamic principles at a phase interface. Models based on thermodynamic balance laws at an interface either ignore the influence of surface stress, or assume it to be isotropic and homogeneous. Generalized conditions for nucleation and growth of phases in multiphase systems with arbitrary surface stress appear to be largely absent in literature.

At the same time, numerical modeling of phase evolution phenomena faces several challenges. Computationally, such phenomena can be classified under the umbrella of moving boundary problems, where an interface or boundary evolves with time. The motion and evolution of the interface can lead to complex shapes and topologies. For instance, spontaneous solidification of supercooled liquid results in formation of dendritic crystals with dense branching (see Fig. 1.1). Also, when multiple phase

Figure 1.1. : Snow crystals are an example of dendritic solidification in nature. The wide diversity and complexity of snow crystal shapes make it highly unlikely for any two crystals to be identical in shape (Image by Wilson Bentley–in Public Domain).

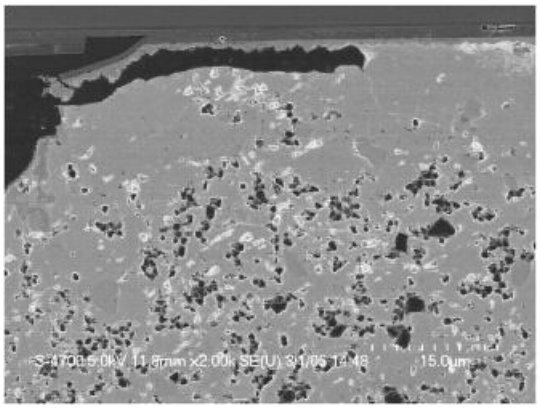


Figure 1.2. : Nucleation and growth of voids in a solder joint due to electromigration. Since a large number of voids are closely packed, voids coalesce with each other to form large macro-voids. [1]

interfaces are present, interactions between them can lead to complex topological changes. This can be observed in Fig. 1.2, where numerous micro-voids, formed due to electromigration, coalesce to form larger voids. As a result, explicit interface representation methods encounter geometric challenges in modeling evolution of phase boundaries. Consequently, such problems are typically modeled using implicit interface methods such as the phase-field method or the level set method. While these techniques naturally capture complex shape and topological changes, they lose geometric information on the interface such as the curvature and normal at a point. Such information can be important to capture the underlying physics. For instance, in dendritic solidification, the temperature at the solid-liquid interface is dictated by the curvature-dependent Gibbs-Thomson condition [2],

$$T = T_m - \epsilon_c(\boldsymbol{n})\kappa - \epsilon_v(\boldsymbol{n})v_n \tag{1.1}$$

where, T_m is the melting point and ϵ_c, ϵ_v are parameters that depend on the normal $\boldsymbol{n}$ to the phase interface at a point. The curvature is denoted by κ, while v_n denotes the normal velocity of the interface. In implicit geometry methods, since the interface is not explicitly defined, the interface boundary conditions imposed indirectly, and possibly approximately. A computational technique that accommodates complex shape and topological changes and imposes interface boundary conditions precisely, while retaining interface geometric information appears to be absent in the literature.

1.1 Literature Survey

A survey of literature on studying nucleation and growth of phases shall now be presented. This shall be followed by a description of various computational techniques found in literature for numerical modeling of moving boundary problems in general, and phase evolution in particular.

1.1.1 Phase Nucleation and Growth

Diffusion-driven phase nucleation and growth under mechanical, thermal or electrical fields represents an important class of problems. Void nucleation and growth under the influence of an electrical current (electromigration) is an example of this class of problems. The growth of a void is typically attributed to the coalescence of vacancies, which is dependent on the magnitude of the bulk diffusion flux of the vacancies [3]. For instance, computational studies of void growth due to electromigration often rely on the flux of vacancies to predict the growth rate [4, 5]. A competing model for void growth is to use the bulk entropy of the system to develop a measure of damage [6]. In these models, a constitutive relation is assumed between the bulk diffusion flux and the various driving forces such as the gradients of chemical potential, hydrostatic stress, temperature or electric potential. This constitutive relation is however not derived using established principles of rational mechanics at a phase interface. In other words, the constitutive relation is empirical and does not directly follow from balance laws governing a moving interface. Abeyaratne and Knowles [7] are among the few that use interface balance laws to derive a thermodynamic force driving the evolution of a phase interface. The thermodynamic force, or the configurational force [8, 9], determines the energy per unit area of the interface consumed as the phase evolves.

In general, the motion of the interface is influenced by the interfacial stress, which may be anisotropic and inhomogeneous. For instance, electromigration experiments have shown a strong relation between void growth rate and surface energy at the interface [10]. The generalization of the configurational force associated with a moving interface, accounting for arbitrary interfacial stress, appears to be largely missing in the literature. The existing models for interfacial growth, while they assume a constutitve relation between the bulk flux and fields, often do not obey interfacial balance laws or incorporate surface stress in their formulation. Gurtin [9] derives the configurational force for a system with homogeneous, isotropic surface stress. However

this does not appear to have been extended to a general system with aniostropic, inhomogeneous surface stress.

In the context of phase nucleation, transformed fraction models based on the Kolmogorov–Johnson–Mehl–Avrami theory and its extensions [11,12] have been used to estimate the nucleation rate. Although the nucleation process has its roots in statistical mechanics, the classical nucleation criterion is developed using continuum theory [13,14]. Specifically in electromigration, a critical vacancy concentration condition [15,16] is often used to describe the onset of void nucleation [4,17]. Among the other criteria that are used to describe eletromigration, Tu et al. proposed that the region of "maximum current crowding" or the region with the highest current density is most susceptible to void nucleation [18,19]. Alternatively, stress-based conditions have also been suggested for nucleation [20–22]. A normal stress condition was used in finite element simulations in [23], while a cohesive zone model was used in [24–26]. Finally, [27,28] have used an entropy based damage criterion to predict void nucleation in finite element simulations. In general, the above models, similar to the growth criteria, do not follow the principles of rational mechanics to derive the nucleation condition from thermodynamic balance laws, and further do not consider the influence of surface stress. Nucleation conditions based on the presence of flaws in free surfaces under homogeneous, isotropic surface stress have been proposed [20,29]. Nevertheless, these models have not been extended to general systems with anisotropic, inhomogeneous surface stress.

1.1.2 Computational Approaches for Phase Evolution Problems

Moving boundary problems in general, and phase evolution problems in particular, involve interfaces evolving with time. In conventional numerical methods such as finite elements, the domain geometry is approximated through a boundary-conforming mesh. Such approaches require re-meshing as the phase interface evolves, resulting in prohibitive computational costs. To avoid re-meshing, several non-conforming

mesh approaches have been proposed in literature, which can be broadly classified into explicit and implicit interface representation methods. When the interface is represented explicitly, geometric information such as the curvature and normal to the interface at a point can be obtained directly. Such geometric information is absent when implicit interface methods such as the level set method or the phase-field method are used. However, implicit interface methods generally model topological changes such as merging and nucleation of phases naturally.

Immersed boundary methods

A common class of non-conforming mesh methods are the Immersed Boundary (IB) methods. Introduced by Peskin [30] to study blood flow in the heart, this class of methods generally involve an interface immersed or embedded in a fixed background mesh and have been used extensively to model fluid-structure interaction problems [31, 32]. Other problems that have been studied include the free-boundary problem and optimal design [33]. Several approaches under this class have been proposed in literature, predominantly with an explicit interface representation. Generally, the immersed body is assumed rigid, and hence only the background fluid domain is solved for with traction and no-slip conditions on the fluid-solid interface. In the immersed finite element method [34], the immersed solid is allowed to deform. A moving Lagrangian mesh is used for the solid while a fixed Eulerian background mesh is used for the fluid. The immersed boundary approach can also be applied to mechanics of complex geometries, by extending the domain to a larger, simpler fictitious region. This is called the fictitious domain method [35, 36], and allows solving problems on complex geometries with simple, structured grids. More recently, the Finite Cell Method [37] has been proposed in a similar vein, wherein a "soft" outer fictitious domain is assumed, with near-zero material properties. Since immersed boundary methods generally use an explicit representation for the interface, geometric information at a point on the interface can be obtained. However, since

the background mesh does not conform with the immersed boundary, imposing interface boundary conditions becomes challenging. Common approaches to impose interface boundary conditions include using Lagrange multipliers, penalty method and Nitsche's method. However, these approaches only enforce interface boundary conditions weakly. The solution structure method was proposed as an alternative for accurate imposition of boundary conditions [38]. Based on Weighted Extended B-splines or WEB-splines [39], this method imposes Dirichlet boundary conditions exactly on the immersed boundary, and uses a distance function to spread this solution to the bulk. However, the generation of the distance function is non-trivial, limiting the application of this method for complex geometries. Furthermore, topological changes such as merging of immersed boundaries pose geometric challenges and require contact detection and computation of intersections between boundaries.

Implicit Interface Methods

Implicit interface methods such as the phase-field method or the level set method describe the phase interface through a state variable such as the phase-field variable or the level set parameter, respectively. The solution to this state variable naturally captures evolution of the interface as well as topological changes such as coalescence and splitting of phases. This however comes at the cost of not explicitly tracking normals and curvatures at the interface, which could be important for phenomena such as solidification.

In the level set method, the evolving interface is implicitly represented as the zero contour of the level set parameter [40, 41]. Evolution of the interface is dictated by the solution of the level set parameter ϕ to the Hamilton-Jacobi equation,

$$\frac{\partial \phi}{\partial t} + v \left\| \nabla \phi \right\| = 0 \tag{1.2}$$

Equation (1.2) is a first-order hyperbolic equation and requires sophisticated techniques to minimize oscillations in the solution [42]. The Generalized Finite Element

Method (GFEM) was introduced by Strouboulis et al. [43], which incorporates a-priori known solution features into the GFEM approximation through enrichments to classical finite element shape functions. Different enrichment functions have been proposed for different behavioral fields [44]. A related approach in the context of crack propagation is the eXtended Finite Element Method (XFEM) proposed by Belytschko [45], where the asymptotic crack tip displacement is used as the enrichment function. The Heaviside function is used to incorporate the jump in displacement across the crack interface [46]. The level set method is used to track the interface and enrichment functions are applied only to nodes close to the interface. Since these enriched nodes are part of the underlying domain, interface boundary conditions cannot be directly imposed on the phase boundary, but only weakly enforced in the background mesh. Apart from crack propagation [47], XFEM has also been applied to the problems of dislocation [48, 49] and phase evolution [50–52].

Another widely used implicit interface method is the phase-field method [53], which has been used to solve moving boundary problems such as crack growth [54, 55] and solidification [56–58]. A phase-field variable is introduced as an additional field. A double-well potential is used to drive the phase-field solution towards equilibrium states corresponding to pure phases. The interface is defined as a diffused region in which the phase-field variable transitions from one equilibrium value to another. A primary challenge with phase-field methods is the need to develop diffused interface equivalents to the governing equations. Common phase-field systems are observed to be highly nonlinear and non-convex, making the solution process challenging. For example, the Cahn-Hilliard system used in phase evolution problems is inherently fourth order in the phase-field variable. Furthermore, since the interface is diffused and not clearly defined, interface boundary conditions are imposed only approximately.

1.1.3 Enriched Field Approximations

In this thesis, an explicit interface method called Enriched Isogeometric Analysis (EIGA) is adopted [59]. Similar to immersed boundary methods, a non-conforming background mesh is used for the underlying domain, and evolving interfaces are represented explicitly as lower-dimensional parametric geometries. However, unlike general immersed boundary methods, the immersed interfaces have separate degrees of freedom associated with them. The behavioral field solution is described as a weighted blending of a continuous approximation associated with the underlying domain, and enrichments representing the influence of immersed interfaces (see Fig. 1.3). Approximations for the behavioral fields are chosen to be isoparametric with the geometric models to facilitate CAD-CAE integration. The construction of behavioral approximations isoparametric with parametric spline geometric models was proposed earlier [60–63]. The use of such approximations for analysis is at present popularly referred as Isogeometric Analysis (IGA, [64]). Since separate degrees of freedom are associated with the immersed interfaces, interface boundary conditions can be accurately and strongly imposed. Further, since the interface is represented explicitly (e.g., through parametric splines), geometric information on the interface such as the normal to and curvature at a point is directly obtained. The governing equations are also simpler as compared to implicit interface formulations. However, topological changes such as coalescence and splitting of phases pose geometric challenges. For example, when two phases coalesce, the merged geometries need to be described. This generally involves computing the surface-surface intersection of the two phase boundaries. Common strategies for contact detection and intersection computation include subdivision, lattice evaluation, marching methods and implicitization [65]. Performing these operations at each simulation time step is computationally expensive.

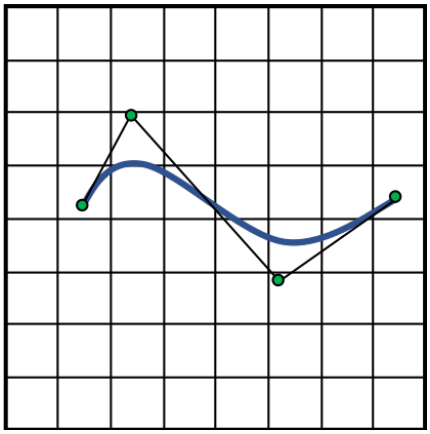


Figure 1.3. : Schematic of enriched isogeometric analysis containing an explicit interface immersed in a non-conforming mesh. Both the explicit interface and the underlying domain contribute to the behavioral field solution.

1.1.4 Algebraic Level Sets

Generally, the influence of an interface weakens with distance. To reflect this, the weighted blending in EIGA is based on distance, so that the interface solution dominates only in the neighborhood of the immersed interface. Also, since immersed interfaces typically separate regions of different phases and thus different material properties, points should be classified based on the side of the interface they lie in. This raises the need for a fast and efficient measure of signed distance, with the sign denoting point classification. Signed distance fields are also important in other applications such as describing the physical behavior of graded materials [66] and fluid-structure interactions [67, 68]. They are also used in geometric applications such as image processing [69], collision detection such as for robotics [70] and animation [71–73], and multi-body contact in finite element analysis [74, 75]. Finally, they are used in volumetric geometric representaion [76] for offset surfaces and in morphing between shapes [77].

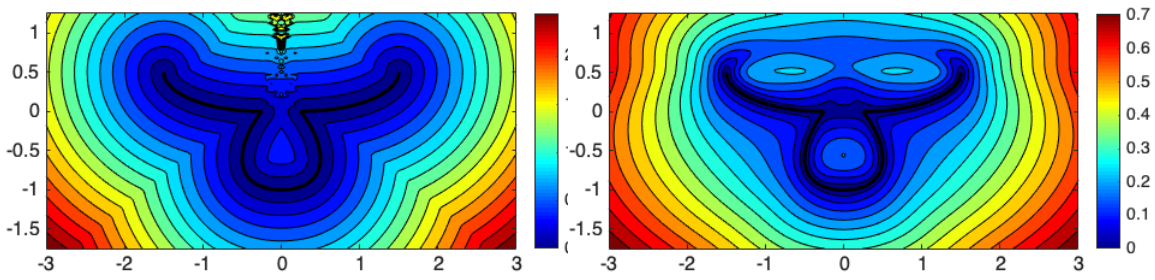
Distance and point classification queries are usually handled using iterative methods such as Newton-Raphson [78–80]. Such iterative methods are not only computationally expensive, the computed distance field is also not sufficiently smooth for

analysis purposes. An alternative approach is to approximate the parametric interface geometry by a polytope (polygon for curves and polyhedron for surfaces), since distance from a planar or linear segment can be easily computed [81, 82]. In order to make the distance field smooth at the vertices and edges, R-functions can be used to combine distance fields from individual segments of the polytope to obtain a smooth distance field for the entire geometry [83]. However, these methods lose the geometric exactness of the interface and are exact only in the limit of refinement. Upreti et al. [84, 85] developed signed algebraic level sets that were generated from implicitization of parametric geometries using resultants [86]. The implicit form of a parametric curve or surface can be expressed as,

$$[\boldsymbol{M_D}(\boldsymbol{x})]\,[\boldsymbol{u}] = 0 \tag{1.3}$$

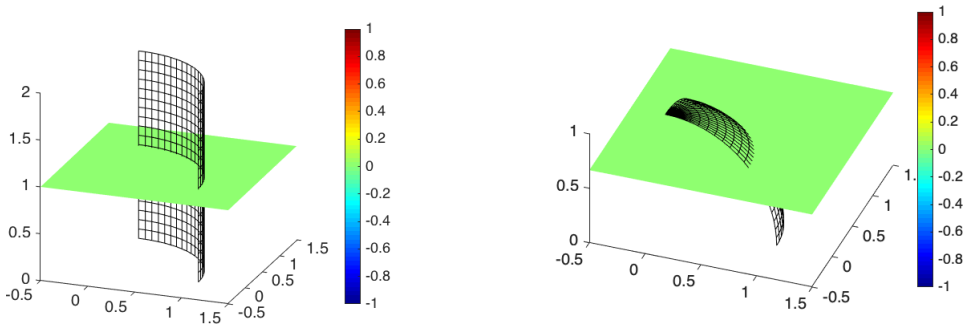
where, the Dixon matrix $\boldsymbol{M_D}$ is linear in the position $\boldsymbol{x}$, and $\boldsymbol{u}$ is a vector of monomials of the parametric coordinates and their powers. For points on the curve or surface, Eq. (1.3) is satisfied for some $\boldsymbol{u}$. Thus, the Dixon matrix $\boldsymbol{M_D}$ is singular for all points on the curve or surface, and $\det(\boldsymbol{M_D}(\boldsymbol{x}))$ generates level sets increasing monotonically with distance. These distance level sets provide a smooth, monotonic distance measure that preserves the exactness of the interface geometry. Since they are generated through a non-iterative, analytical procedure, the distance estimation is computationally efficient. Furthermore, for closed geometries, the algebraic level sets are signed, thereby intrinsically classifying points as lying inside, on or outside the closed geometry. An illustration of the smoothness of these algebraic level sets vis-a-vis an iterative method such as Newton-Raphson is provided in Fig. 1.4. These advantages of the algebraic level sets have recently been taken advantage of to perform smooth, algebraic point projections on to curves and surfaces, including geometries with high local curvature and sharp corners [87].

EIGA with algebraic level sets have been used to study several moving boundary problems such as solidification [88], fracture [84] and optimal design [89]. However,



(a) (b)

Figure 1.4. : Distance level sets for a quadratic NURBS curve obtained from (a) Newton-Raphson iterations and (b) Algebraic level sets. The algebraic level sets are smooth even in the neighborhood of sharp corners and in regions of high curvature. Distance estimates using the Newton-Raphson, on the other hand, show discontinuities near such features.



(a) (b)

Figure 1.5. : Dixon resultant contours for (a) cylinder of unit radius centered at the origin (b) sphere of unit radius centered at the origin. The Dixon matrix is identically singular and hence the resultant is zero everywhere for both geometries.

application to three-dimensional problems has been limited due to issues in generating signed algebraic level sets for general surfaces. For several three-dimensional parametric surfaces (including common surfaces such as spheres and cylinders), the Dixon matrix is trivially singular. For such surfaces, $\det(\boldsymbol{M_D}(\boldsymbol{x}))$ is zero everywhere, even where Eq. (1.3) is not satisfied, thereby failing to generate distance level sets (see Fig. 1.5). The vanishing of the Dixon resultant is thus only a necessary condition to satisfy Eq. (1.3), and not sufficient. A second issue with using the Dixon resultant is the occasional lack of sign for closed surfaces. The sign of the Dixon resultant is used to ascribe the sign of algebraic level sets for points close to the surface. For most surfaces the Dixon resultant is signed, classifying space into inside and outside regions. However, for some solids (see Fig. 5.3), the Dixon resultant could lack an intrinsic sign. Subsequently, the generated algebraic level sets are unsigned and thus cannot classify points as lying inside or outside the closed geometry. Both these issues also warrant rectifications to the algebraic point projection procedure proposed in [87]. An alternate implicitization procedure for parametric surfaces was developed by L. Busé [90] called the Matrix-Representation (M-Rep). In this approach, the parametric geometry is described as the locus of rank drop of a single M-Rep matrix $\boldsymbol{M_R}$; the matrix is singular only at points on the curve or surface. Further, the author also provides a measure of distance from the geometry based on the singular values of the M-Rep matrix. This robust implicitization procedure is observed to be valid even for geometries with trivially singular Dixon resultants. However, since singular values are always non-negative, the generated level sets are unsigned and hence cannot answer point containment queries. Additionally, the M-Rep distance measure requires a singular value decomposition at each point of evaluation. This makes the approach computationally expensive. L. Busé also suggests an alternate distance measure given by the determinant of the matrix product $\boldsymbol{M_R}\boldsymbol{M_R}^{\mathrm{T}}$. This distance measure is also unsigned, and while it may avoid calculating singular values, matrix products are generally not robust and can lead to round-off errors.

1.2 Research Objectives

The objectives of this dissertion are the analytical and numerical modeling of phase evolution problems for systems with multiple diffusing species and inhomogeneous surface stress. As discussed in § 1.1.1, existing models for phase evolution often assume constitutive relations and generally do not follow the principles of rational mechanics to derive nucleation and growth conditions from thermodynamic balance laws. Further, existing models either ignore the influence of surface stress, or assume the surface stress to be homogeneous and isotropic. Thus, there is a need to develop generalized conditions for growth or nucleation of a phase under the influence of mechanical and non-mechanical fields as well as arbitrary surface stress. To this regard, a configurational force associated with a moving interface in a body is developed, allowing inhomogeneous, anisotropic arbitrary surface stresses. Further, general thermodynamic principles governing the motion of an interface are used to derive conditions for growth of a phase as well as the nucleation of a new phase.

The modeling of phase evolution is also numerically challenging, as shown in § 1.1.2. The non-conforming explicit interface method, Enriched Isogeometric Analysis (EIGA), is adopted in this thesis. However, application of EIGA to three-dimensional problems is challenged by the need to generate algebraic level sets and point projection for surfaces whose Dixon resultant is singular. In this thesis, rectification measures are developed to generate signed algebraic level sets and for smooth point projection for surfaces. This allows studying three-dimensional phase evolution problems using EIGA. Electromigration experiments on Blech-like Copper-TiN test structures with different cap layers were conducted in [91]. The copper edge was observed to displace due to electromigration, and the growth rate was measured under different conditions of temperature and current densities. In this thesis, EIGA simulations are used to model these experiments to study the copper edge profile under electromigration. Since an explicit interface representation is used in EIGA, modeling topological changes poses geometric challenges. Taking advantage of the fact that analysis in

EIGA is based on signed algebraic level sets, an implicit interface information, algebraic Boolean compositions on the level sets are used to accommodate topological changes such as merging. This allows modeling coalescence of phases without having to detect contact and compute surface-surface intersections.

1.3 Outline

The rest of this dissertation is organized as follows. The analytical modeling of phase evolution is described in detail in Chapter 2. Balance laws are derived on an interface using a pillbox procedure, to derive a configurational force driving motion of the interface. These are then used to derive generalized conditions for the nucleation and growth of phases for multiphase systems with arbitrary surface stress. Following the analytical modeling, an introduction to enriched isogeometric analysis (EIGA) is provided in Chapter 3, with an example of a two-dimensional electrostatics problem. This approach uses a quick and efficient measure of distance called signed algebraic level sets. A detailed discussion on signed algebraic level sets is provided in Chapter 4. The generation of level sets is shown to fail frequently for three-dimensional surfaces. Rectification measures are developed in Chapter 5 to generate signed level sets for surfaces. This enables modeling three-dimensional phase evolution problems using EIGA. As an illustration, a three-dimensional electrostatics problem is solved. The approach is then applied to solve three-dimensional electromigration problems in Chapter 6. First, the evolution of a void under electromigration in a line is tracked. This is followed by EIGA simulations of a Blech-like Copper-TiN test structure subjected to electromigration, showing edge displacement as observed in the experiments conducted in [91]. Complications in EIGA occur when there are multiple, interacting phase boundaries as these generally lead to topological changes. A computationally efficient approach to model such changes using algebraic Boolean compositions on algebraic level sets is provided in Chapter 7. The developed approach is demonstrated

on electrostatics problems with coalescent voids. The approach is then used to solve a problem of bubble coalescence subject to constraints of conservation of mass.

2. PHYSICS OF PHASE NUCLEATION AND GROWTH

Diffusion-driven phase nucleation and growth under mechanical, thermal or electrical fields represents an important class of problems. Void nucleation and growth under the influence of an electrical current (electromigration) is an example of this class of problems. Other examples of such phenomena include crack propagation due to creep [92], dendritic growth in Li-ion batteries [93, 94], growth of brittle intermetallic compounds in solder joints [95, 96], tissue growth [97], and growth of compounds under chemical-vapor deposition [98]. The interplay between stress and diffusion can change the morphology of the phase-interface from being crack-like to spheroidal [99]. The growth of such heterogeneities can have significant physical as well as physiological effect. For example, growth of cracks at high temperatures due to creep decides the life of pipes, turbines, engines in power-plants and aircrafts, and moderating the growth of malignant tissues can aid in recovery of diseased individuals. As described in § 1.1.1, generalized conditions describing nucleation and growth of phases in multiphase systems with arbitrary surface stress are absent in literature.

In this chapter, a transport theorem for a phase interface is derived following the pioneering work of Truesdell and Toupin [100], and used to form balance laws at the interface. From interface second law conditions, a configurational force associated with a moving interface in a body with multiple diffusing species and inhomogeneous, anisotropic surface stress is developed. This is used to develop generalized conditions for growth or nucleation of a phase under the influence of mechanical and non-mechanical fields as well as arbitrary surface stress. Finally, the developed ideas are demonstrated on a practical problem to deduce the critical energy density associated with electromigration in Al-TiN interface.

2.1 Review of Balance Laws in the Bulk

The procedure described in [8,101] is followed in developing the bulk relations for a multi-species (atoms or molecules) solid. Consider a region Ω of the body $\mathcal{B}$ at an instant in time convecting with the body. $\rho(\boldsymbol{x},t)$ is the instantaneous mass density at any point in the region. Since the region Ω convects with the body, $\frac{d}{dt}\int_\Omega \rho\, d\Omega = 0$. Let the convecting region contain N species labeled $\alpha = 1,2,...,N$, and let $\nu^\alpha(\boldsymbol{x},t)$ denote the instantaneous mass fraction of species α (ratio of species mass density to ρ). The body is assumed to consist entirely of diffusing species such that $\sum_\alpha \nu^\alpha = 1$, and $\rho^\alpha = \rho\nu^\alpha$ represents the mass density of species α. Changes in Ω are brought about by the diffusion of species α across the boundary $\partial\Omega$ as well as instantaneous species supply r^α measured in mass per unit volume per unit time. The mass flux $\boldsymbol{j}^\alpha(\boldsymbol{x},t)$ is measured in mass per unit area, per unit time, and assumed positive in the outward normal direction of the surface across which the species is transported. Letting $\boldsymbol{n}$ denote the outward unit normal to the boundary, the mass balance for the species is,

$$\frac{d}{dt}\int_\Omega \rho\nu^\alpha\, d\Omega = -\int_{\partial\Omega} \boldsymbol{j}^\alpha\cdot\boldsymbol{n}\, d\Gamma_{\mathrm{s}} + \int_\Omega r^\alpha\, d\Omega \tag{2.1}$$

which yields, after the application of the divergence theorem as well as the Reynolds Transport theorem [101], the following local mass balance for any species α,

$$\rho\dot{\nu}^\alpha = -\nabla\cdot\boldsymbol{j}^\alpha + r^\alpha \tag{2.2}$$

Since the mass fractions sum to unity at all times, the above expression also implies

$$\sum_\alpha \rho\dot{\nu}^\alpha = \sum_\alpha (-\nabla\cdot\boldsymbol{j}^\alpha + r^\alpha) = 0 \tag{2.3}$$

The tractions acting on the surface of the body cause stress $\boldsymbol{\sigma}$ within the body at any instant. $\boldsymbol{\sigma n}$ represents the surface traction exerted on Ω across the boundary $\partial\Omega$. The balance of linear and angular momentum requires,

$$\begin{aligned} \frac{d}{dt}\int_{\Omega} \rho \boldsymbol{v}\, d\Omega &= \int_{\partial\Omega} \boldsymbol{\sigma n}\, d\Gamma_{\mathrm{s}} + \int_{\Omega} \rho \boldsymbol{b}\, d\Omega \\ \frac{d}{dt}\int_{\Omega} \boldsymbol{x} \times \rho \boldsymbol{v}\, d\Omega &= \int_{\partial\Omega} \boldsymbol{x} \times (\boldsymbol{\sigma n})\, d\Gamma_{\mathrm{s}} + \int_{\Omega} \boldsymbol{x} \times \rho \boldsymbol{b}\, d\Omega \end{aligned} \tag{2.4}$$

where, $\boldsymbol{v}$ is the velocity and $\boldsymbol{b}$ is the body force per unit mass. These yield the local force and moment balances,

$$\begin{aligned} \rho \boldsymbol{a} &= \nabla \cdot \boldsymbol{\sigma}^{\mathrm{T}} + \rho \boldsymbol{b} \\ \boldsymbol{\sigma} &= \boldsymbol{\sigma}^{\mathrm{T}} \end{aligned} \tag{2.5}$$

with $\boldsymbol{a}$ being the acceleration.

The accumulation of the specific internal energy e and the kinetic energy within the body is:

$$\begin{aligned} \frac{d}{dt}\int_{\Omega} \left(\rho e + \frac{1}{2}\rho \boldsymbol{v} \cdot \boldsymbol{v}\right) d\Omega = &\int_{\partial\Omega} (\boldsymbol{\sigma n}) \cdot \boldsymbol{v}\, d\Gamma_{\mathrm{s}} + \int_{\Omega} \rho \boldsymbol{b} \cdot \boldsymbol{v}\, d\Omega \\ &- \int_{\partial\Omega} \boldsymbol{j}^{q} \cdot \boldsymbol{n}\, d\Gamma_{\mathrm{s}} + \int_{\Omega} r^{q}\, d\Omega \\ &- \sum_{\alpha=1}^{N} \left(\int_{\partial\Omega} \mu^{\alpha} \boldsymbol{j}^{\alpha} \cdot \boldsymbol{n}\, d\Gamma_{\mathrm{s}} - \int_{\Omega} \mu^{\alpha} r^{\alpha}\, d\Omega\right) \end{aligned} \tag{2.6}$$

where, $\boldsymbol{j}^{q}$ is the heat flux assumed positive in the outward normal direction of the surface, r^{q} is the rate of heat supply, and μ^{α} is the chemical potential of species α measured in units of energy per unit mass. As before, applying the Reynolds Transport theorem, using the divergence theorem on the integral over $\partial\Omega$, and utilizing the mass balance relation Equation (2.2), one obtains the local rate of change of internal energy as

$$\rho \dot{e} = \boldsymbol{\sigma} : \mathbf{D} - \nabla \cdot \boldsymbol{j}^{q} + r^{q} - \sum_{\alpha=1}^{N} (\boldsymbol{j}^{\alpha} \cdot \nabla \mu^{\alpha} - \rho \mu^{\alpha} \dot{\nu}^{\alpha}) \tag{2.7}$$

where, $\mathbf{D}$ is the symmetric part of the velocity gradient tensor $\nabla \boldsymbol{v}^{\mathrm{T}}$. The local form of entropy imbalance is defined as:

$$\rho\dot{\eta} \geq -\nabla \cdot \left(\frac{\boldsymbol{j}^q}{T}\right) + \frac{r^q}{T} \tag{2.8}$$

where, η is the specific internal entropy and T is the absolute temperature. Defining the free energy as $\psi = e - \eta T$, using the local form of entropy imbalance Equation (2.8), finally the local form of the free energy imbalance is obtained as:

$$\rho(\dot{\psi} + \eta\dot{T}) \leq \boldsymbol{\sigma} : \mathbf{D} - \frac{1}{T}\boldsymbol{j}^q \cdot \nabla T - \sum_{\alpha=1}^{N} \left(\boldsymbol{j}^\alpha \cdot \nabla \mu^\alpha - \rho\mu^\alpha \dot{\nu^\alpha}\right) \tag{2.9}$$

Thus, all thermodynamically permissible processes in the bulk must be consistent with Equation (2.9). The free energy ψ is a function of the deformation, the species concentration, and the temperature $\psi \equiv \psi\left(\boldsymbol{F}, \nu^\alpha, T\right)$, where $\boldsymbol{F}$ is the deformation gradient. Using the chain rule to compute $\dot{\psi}$ and writing $\boldsymbol{\sigma} : \mathbf{D} = \boldsymbol{\sigma} : \nabla \boldsymbol{v}^{\mathrm{T}} = \boldsymbol{\sigma} : \dot{\boldsymbol{F}}\boldsymbol{F}^{-1}$ gives,

$$\begin{aligned}\left(\rho\frac{\partial\psi}{\partial\boldsymbol{F}} - \boldsymbol{\sigma}\boldsymbol{F}^{-\mathrm{T}}\right) : \dot{\boldsymbol{F}} + \sum_{\alpha=1}^{N}\left(\frac{\partial\psi}{\partial\nu^\alpha} - \mu^\alpha\right)\rho\dot{\nu^\alpha} + \left(\frac{\partial\psi}{\partial T} + \eta\right)\rho\dot{T} \\ + \sum_{\alpha=1}^{N}\boldsymbol{j}^\alpha \cdot \nabla\mu^\alpha + \frac{1}{T}\boldsymbol{j}^q \cdot \nabla T \leq 0\end{aligned} \tag{2.10}$$

Since this inequality has to hold for arbitrary values of $\dot{\boldsymbol{F}}$, $\dot{\nu^\alpha}$, and $\dot{T}$ we get the constitutive equations,

$$\boldsymbol{\sigma} = \rho\frac{\partial\psi}{\partial\boldsymbol{F}}\boldsymbol{F}^{\mathrm{T}} \tag{2.11}$$

$$\mu^\alpha = \frac{\partial\psi}{\partial\nu^\alpha} \quad \forall\alpha \tag{2.12}$$

$$\eta = -\frac{\partial\psi}{\partial T} \tag{2.13}$$

The terms within the parentheses hence vanish. Finally, since temperature is always positive, the free energy inequality is satisfied by constitutive relations of form,

$$\boldsymbol{j}^{\alpha} = -\mathbf{M}^{\alpha}\nabla\mu^{\alpha} \quad \forall\alpha \tag{2.14}$$

$$\boldsymbol{j}^{q} = -\mathbf{k}\nabla T \tag{2.15}$$

where, $\mathbf{M}^{\alpha}$ is a positive definite mobility tensor for species α and $\mathbf{k}$ is a positive definite thermal conductivity tensor. Equation (2.14) describes Fick's first law of diffusion for each species α, while Eq. (2.15) is the Fourier law of heat conduction.

2.2 Derivation of Balance Laws at the Interface

The development in this section is motivated by the pioneering work of Truesdell and Toupin [100]. The approach followed is to make a systematic analogy to the bulk balance laws described in § 2.1, including in the explicit use of an entropic inequality condition on the interface. Additionally, analogous to bulk derivation, surface quantities corresponding to mass, momentum, internal energy, entropy and free energy are introduced as variables and resolved through the derived thermodynamic restrictions. The resulting form of the driving force on the phase interface is similar to that derived in [8,102], but the derivation here allows inhomogeneous and anisotropic surface stress.

2.2.1 Interface Transport Theorem

Prior to deriving the balance laws at the interface, the interface transport theorem for a scalar field on the interface is first derived. This then serves to provide the basis for derivation of mass, momentum and energy balances at the interface. The interface is modeled to convect with the body in the present work. To connect the bulk fields with the interface, an interface "pillbox" (Fig. 2.1) is typically used to derive local balance relations at interfaces.

Consider the domain $\Omega = \Omega^- \cup \Omega^+$ divided by the interface Γ shown in Fig. 2.1 with the subregion Γ_s within the control volume. Ω^- and Ω^+ are bounded otherwise by Γ^- and Γ^+. An arbitrary field $\phi(\boldsymbol{x}, t)$ is considered, with the values in each of the phases specified by $\phi^-(\boldsymbol{x}, t)$ and $\phi^+(\boldsymbol{x}, t)$. The balance law for the domain of each phase can now be written as,

$$\frac{d}{dt}\left(\int_{\Omega^i} \phi^i \, d\Omega\right) = -\int_{\Gamma^i} \boldsymbol{j}^i \cdot \boldsymbol{n}^i \, d\Gamma_s - \int_{\Gamma_s} \boldsymbol{j}^i \cdot \boldsymbol{n}^i \, d\Gamma_s + \int_{\Omega^i} r^i \, d\Omega \qquad i = -, + \quad (2.16)$$

where, $\boldsymbol{j}$ refers to the flux of the field ϕ while r denotes volumetric generation. The interface is now considered as a separate domain, bounded by the curve $\partial\Gamma_s$ (Fig. 2.1). The superficial field on the interface ϕ_s is changed by inflows due to a flux $\boldsymbol{h}$ on the boundary $\partial\Gamma_s$ of the interface, the flux exchanges $\boldsymbol{j}^i$ with the bulk domains on either side, and the spontaneous generation rate r_s. The normal to the curve $\partial\Gamma_s$ in the tangent plane of Γ is denoted $\boldsymbol{m}$ (see Fig. 2.2). The balance law for the interface is now written as,

$$\frac{d}{dt}\int_{\Gamma_s} \phi_s \, d\Gamma_s = -\oint_{\partial\Gamma_s} \boldsymbol{h}.\boldsymbol{m} \, dc + \sum_{i=-,+} \int_{\Gamma_s} \boldsymbol{j}^i \cdot \boldsymbol{n}^i \, d\Gamma_s + \int_{\Gamma_s} r_s \, d\Gamma_s \qquad (2.17)$$

To be able to define a transport theorem for the interface pillbox, Eqs. (2.16) and (2.17) need to be combined together. Utilizing the expressions for the material derivatives on the left hand sides of Eqs. (2.16) and (2.17) (derived in Appendices B and C on the basis of surface identities given in Appendix A), following [100], the combined bulk and surface balances yields,

$$\begin{aligned} \sum_{i=-,+} \left(\int_{\Omega^i} \frac{\partial \phi^i}{\partial t} \, d\Omega + \int_{\Gamma^i} \phi^i v_n^i \, d\Gamma_s + \int_{\Gamma_s} \phi^i v_{s_n}^i \, d\Gamma_s\right) \\ + \int_{\Gamma_s} \left(\dot{\phi}_s + \phi_s \nabla_s \cdot \boldsymbol{v}_s\right) d\Gamma_s = \\ \sum_{i=-,+} \left(-\int_{\Gamma^i} \boldsymbol{j}^i \cdot \boldsymbol{n}^i \, d\Gamma_s + \int_{\Omega^i} r^i \, d\Omega\right) - \oint_{\partial\Gamma_s} \boldsymbol{h}.\boldsymbol{m} \, dc + \int_{\Gamma_s} r_s \, d\Gamma_s \end{aligned} \qquad (2.18)$$

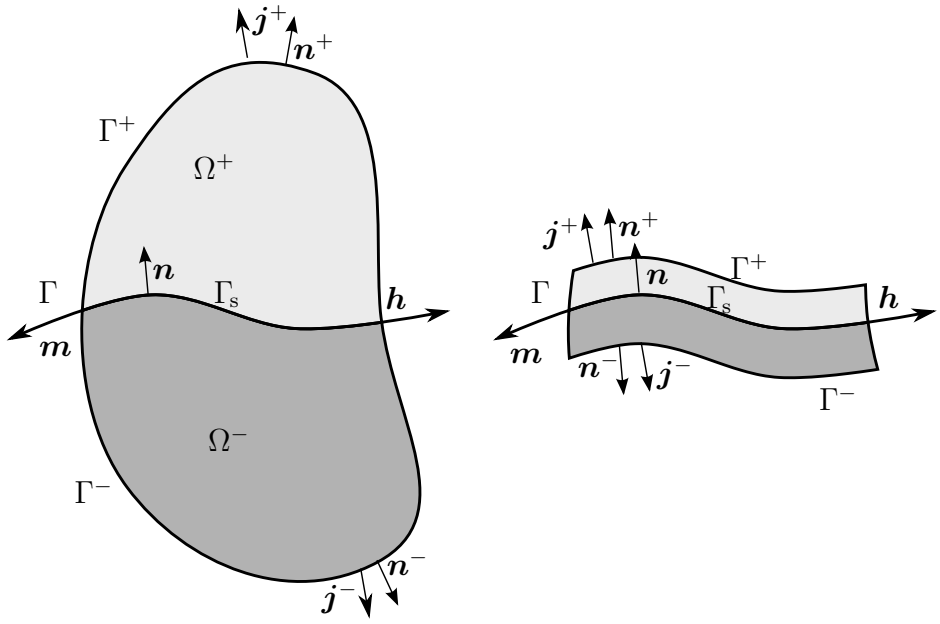


Figure 2.1. : Control Volume (interface "pillbox") with generalized field and fluxes.

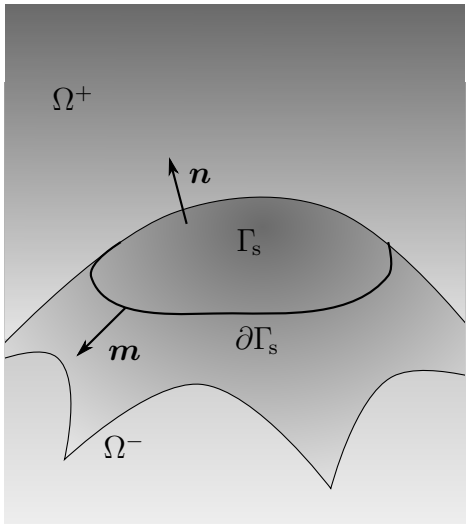


Figure 2.2. : Schematic showing the definition of the normals to the interface and to the boundary of the interface subregion Γ_s.

where, $v_n^i = \boldsymbol{v}^i \cdot \boldsymbol{n}^i$, $v_{\mathrm{s_n}}^i = \boldsymbol{v}_\mathrm{s}^i \cdot \boldsymbol{n}^i$. The following simplifications are made to Eq. (2.18).

1. The outer control surfaces Γ^-, Γ^+ are infinitesimally close to the inner control surface Γ_s. As $\Gamma^-, \Gamma^+ \to \Gamma_\mathrm{s}$, $\Omega \to 0$ and the bulk integral terms drop out.

2. $\boldsymbol{n}^-$ is set equal to $\boldsymbol{n}$ on Γ_s. This implies that $\boldsymbol{n}^+ = -\boldsymbol{n}$ on Γ_s. Thus, as $\Omega \to 0$, $\boldsymbol{n}^- = -\boldsymbol{n}$ on Γ^-, and $\boldsymbol{n}^+ = \boldsymbol{n}$ on Γ^+.

These simplifications along with the use of the surface divergence theorem of Equation (A.9) reduces Eq. (2.18) to,

$$\begin{aligned} \int_{\Gamma_\mathrm{s}} \Big(-\left[\!\left[\phi \left(v_{\mathrm{s_n}} - v_n\right)\right]\!\right] + \dot{\phi}_\mathrm{s} + \phi_\mathrm{s} \nabla_\mathrm{s} \cdot \boldsymbol{v}_\mathrm{s} \Big)\, d\Gamma_\mathrm{s} = \\ \int_{\Gamma_\mathrm{s}} \Big(-\left[\!\left[\boldsymbol{j}^\phi\right]\!\right] \cdot \boldsymbol{n} - \nabla_\mathrm{s} \cdot \boldsymbol{h}_\mathbf{t}^\phi + r_\mathrm{s}^\phi \Big)\, d\Gamma_\mathrm{s} \end{aligned} \tag{2.19}$$

where, the jump terms are defined as $[\![\cdot]\!] = (\cdot^+ - \cdot^-)$ and $(v_{\mathrm{s_n}} - v_n) = (\boldsymbol{v}_\mathrm{s} - \boldsymbol{v}) \cdot \boldsymbol{n}$; $\boldsymbol{h}_\mathbf{t}^\phi = \mathbf{P}\boldsymbol{h}^\phi$ is the tangential component of the flux $\boldsymbol{h}^\phi$ with $\mathbf{P}$ being the surface projection tensor defined in Eq. (A.1). The superscript ϕ serves to remind the quantity

whose flux is on the right hand side. Since Eq. (2.19) must be valid on every arbitrary subregion Γ_s of Γ, we arrive at the local form of the interface transport theorem:

$$-\left[\!\left[\phi\left(v_{s_n}-v_n\right)\right]\!\right]+\dot{\phi}_s+\phi_s\nabla_s\cdot\boldsymbol{v}_s=-\left[\!\left[\boldsymbol{j}^{\phi}\right]\!\right]\cdot\boldsymbol{n}-\nabla_s\cdot\boldsymbol{h}_{\mathbf{t}}^{\phi}+r_s^{\phi}\ \text{ on }\ \Gamma \tag{2.20}$$

where, the left hand side of the above expression represents the accumulation of bulk and interface fields due to bulk normal fluxes and the interface flux.

2.2.2 Balance Laws at the Interface

The general interface transport theorem of Equation (2.20) can be applied to various conserved quantities such as species mass, momentum and energy, by replacing ϕ with the appropriate conserved quantity to obtain the balance law. The species mass balance at the interface is obtained by replacing the bulk field ϕ by $\rho\nu^{\alpha}$, the mass flux $\boldsymbol{j}^{\phi}$ of the diffusing species α by $\boldsymbol{j}^{\alpha}$, and the tangential interface flux $\boldsymbol{h}_{\mathbf{t}}^{\phi}$ by the surface mass flux $\boldsymbol{h}_{\mathbf{t}}^{\alpha}$. Also, the interface is considered massless, and therefore, surface mass excess, ϕ_s is ignored. The rate of spontaneous generation of species α at the surface (as due to a chemical reaction) is denoted r_s^{α}. Substituting in Equation (2.20), we obtain the species balance as,

$$-\left[\!\left[\rho\nu^{\alpha}\left(v_{s_n}-v_n\right)\right]\!\right]=-\left[\!\left[\boldsymbol{j}^{\alpha}\right]\!\right]\cdot\boldsymbol{n}-\nabla_s\cdot\boldsymbol{h}_{\mathbf{t}}^{\alpha}+r_s^{\alpha} \tag{2.21}$$

The momentum balance at the interface is obtained by considering momentum in an arbitrary, but fixed direction $\hat{\boldsymbol{n}}$. By analogy to the bulk relation Eq. (2.4), we choose the field ϕ to be the momentum in the direction $\hat{\boldsymbol{n}}$, that is, $\phi=\rho\boldsymbol{v}\cdot\hat{\boldsymbol{n}}$. Since the negative of the normal component of the bulk flux enters the domain, by analogy of Eq. (2.4), $-\boldsymbol{j}^{i}\cdot\boldsymbol{n}^{i}=\hat{\boldsymbol{n}}\cdot\boldsymbol{\sigma}\boldsymbol{n}^{i}$, and the flux is defined as $\boldsymbol{j}^{\mathbf{v}}=-\boldsymbol{\sigma}\hat{\boldsymbol{n}}$ so that the normal component of this quantity defines the traction component in the direction $\hat{\boldsymbol{n}}$. Similarly, since the component of the surface flux normal to the boundary $\partial\Gamma_s$ enters the surface, the surface flux is defined as $\boldsymbol{h}^{\mathbf{v}}=-\boldsymbol{\sigma}_s\hat{\boldsymbol{n}}$ with a symmetric surface

stress $\boldsymbol{\sigma}_{\mathrm{s}}$ such that traction component $\boldsymbol{h}^{\mathbf{v}} \cdot \boldsymbol{m}$ acts on the boundary $\partial\Gamma_{\mathrm{s}}$ tangent to the plane of the interface. Now, assuming that the interface is massless, and that the spontaneous generation of momentum is negligible, i.e., ϕ_{s} and r_{s} are zero, substituting in Equation (2.20), we obtain the momentum balance as:

$$-\left[\!\left[\rho \boldsymbol{v}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\right] \cdot \hat{\boldsymbol{n}}=\left[\!\left[\boldsymbol{\sigma} \hat{\boldsymbol{n}}\right]\!\right] \cdot \boldsymbol{n}-\nabla_{\mathrm{s}} \cdot \boldsymbol{h}_{\mathbf{t}}^{\mathbf{v}} \tag{2.22}$$

Furthermore, $\boldsymbol{h}_{\mathbf{t}}^{\mathbf{v}}=\mathbf{P} \boldsymbol{h}^{\mathbf{v}}=-\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}} \hat{\boldsymbol{n}}$. Since $\hat{\boldsymbol{n}}$ is arbitrary, constant and non-zero, Eq. (2.22) yields:

$$-\left[\!\left[\rho \boldsymbol{v}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\right]=\left[\!\left[\boldsymbol{\sigma}\right]\!\right] \boldsymbol{n}+\nabla_{\mathrm{s}} \cdot\left(\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}}\right) \tag{2.23}$$

If the surface stress is homogeneous and isotropic, i.e., $\boldsymbol{\sigma}_{\mathrm{s}}=\gamma \mathbf{I}$, bulk inertial effects are small, then the momentum jump can be ignored, and using Equation (A.6), we can obtain the Laplace-Young equation:

$$-\boldsymbol{n} \cdot\left[\!\left[\boldsymbol{\sigma}\right]\!\right] \boldsymbol{n}=\gamma \kappa \tag{2.24}$$

where, κ is the total curvature of the interface.

As with momentum balance, the interface energy balance is obtained by analogy of Eq. (2.6). Thus, the bulk field is $\phi=\rho e+\frac{1}{2} \rho \boldsymbol{v} \cdot \boldsymbol{v}$, bulk energy flux is $\boldsymbol{j}^e=-\boldsymbol{\sigma} \boldsymbol{v}+\boldsymbol{j}^q+\sum_{\alpha=1}^N \mu^\alpha \boldsymbol{j}^\alpha$. The surface field ϕ_{s} is given by the internal energy per unit area, e_{s}. By analogy to interface momentum balance, the tangential component of the surface flux corresponding to internal energy accumulation is given by $\boldsymbol{h}_{\mathbf{t}}^e=-\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}} \boldsymbol{v}_{\mathrm{s}}+\boldsymbol{h}^q+\sum_{\alpha=1}^N \mu_{\mathrm{s}}^\alpha \boldsymbol{h}_{\mathbf{t}}^\alpha$. Thus, the interface energy balance relation is:

$$\begin{aligned}
&-\left[\!\left[\phi\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\right]+\dot{\phi}_{\mathrm{s}}+\phi_{\mathrm{s}} \nabla_{\mathrm{s}} \cdot \boldsymbol{v}_{\mathrm{s}}= \\
&-\left[\!\left[-\boldsymbol{\sigma} \boldsymbol{v}+\boldsymbol{j}^q+\sum_{\alpha=1}^N \mu^\alpha \boldsymbol{j}^\alpha\right]\!\right] \cdot \boldsymbol{n}-\nabla_{\mathrm{s}} \cdot\left(-\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}} \boldsymbol{v}_{\mathrm{s}}+\boldsymbol{h}^q+\sum_{\alpha=1}^N \mu_{\mathrm{s}}^\alpha \boldsymbol{h}_{\mathbf{t}}^\alpha\right) \\
&+r_{\mathrm{s}}^q+\sum_{\alpha=1}^N \mu_{\mathrm{s}}^\alpha r_{\mathrm{s}}^\alpha
\end{aligned} \tag{2.25}$$

Using the interface species mass balance Eq. (2.21) and utilizing the surface identities in Appendix A, the above equation simplifies to:

$$\begin{aligned} -\left[\!\!\left[\left(\rho e+\frac{1}{2}\rho\boldsymbol{v}\cdot\boldsymbol{v}\right)\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\dot{e}_{\mathrm{s}}+e_{\mathrm{s}}\nabla_{\mathrm{s}}\cdot\boldsymbol{v}_{\mathrm{s}}= \\ \left[\!\!\left[\boldsymbol{\sigma v}\right]\!\!\right]\cdot\boldsymbol{n}-\left[\!\!\left[\boldsymbol{j}^q\right]\!\!\right]\cdot\boldsymbol{n}-\nabla_{\mathrm{s}}\cdot\boldsymbol{h}_{\mathbf{t}}^q+r_{\mathrm{s}}^q \\ -\sum_{\alpha=1}^{N}(\mu_{\mathrm{s}}^{\alpha}\left[\!\!\left[\rho\nu^{\alpha}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\boldsymbol{h}_{\mathbf{t}}^{\alpha}\cdot\nabla_{\mathrm{s}}\mu_{\mathrm{s}}^{\alpha}) \\ +\sum_{\alpha=1}^{N}(-\left[\!\!\left[\mu^{\alpha}\boldsymbol{j}^{\alpha}\right]\!\!\right]+\mu_{\mathrm{s}}^{\alpha}\left[\!\!\left[\boldsymbol{j}^{\alpha}\right]\!\!\right])\cdot\boldsymbol{n} \\ +\nabla_{\mathrm{s}}\cdot(\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}}) \end{aligned} \tag{2.26}$$

The interface entropy inequality can be obtained analogous to the bulk entropy inequality Eq. (2.8). Let η denote the bulk specific internal entropy and η_{s} denote the entropy per unit area of the interface. The external sources of entropy are the bulk heat flux $\boldsymbol{j}^q$, surface heat flux $\boldsymbol{h}^q$ and the surface heat generation r_{s}^q. Following the pillbox procedure in § 2.2.1 gives the interface entropy imbalance inequality,

$$-\left[\!\!\left[\rho\eta\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\dot{\eta}_{\mathrm{s}}+\eta_{\mathrm{s}}\nabla_{\mathrm{s}}\cdot\boldsymbol{v}_{\mathrm{s}}\geq-\left[\!\!\left[\frac{\boldsymbol{j}^q}{T}\right]\!\!\right]\cdot\boldsymbol{n}-\nabla_{\mathrm{s}}\cdot\frac{\boldsymbol{h}_{\mathbf{t}}^q}{T_{\mathrm{s}}}+\frac{r_{\mathrm{s}}^q}{T_{\mathrm{s}}} \tag{2.27}$$

where, T_{s} is the surface temperature.

Defining the specific free energy, $\psi=e-\eta T$ and its interface equivalent, the free energy per unit area of the interface, $\psi_{\mathrm{s}}=e_{\mathrm{s}}-\eta_{\mathrm{s}}T_{\mathrm{s}}$, the free energy inequality is obtained by using the first and second laws of thermodynamics (Eqs. (2.26) and (2.27)), assuming continuity of chemical potential across the interface, i.e.,

$$\left[\!\!\left[\mu^{\alpha}\right]\!\!\right]=0,\quad\mu_{\mathrm{s}}^{\alpha}=\mu^{\alpha} \tag{2.28}$$

and under the assumption of continuity of temperature, i.e.,

$$\left[\!\!\left[T\right]\!\!\right]=0,\quad T_{\mathrm{s}}=T \tag{2.29}$$

we get:

$$\begin{aligned}
&-\left[\!\!\left[\left(\rho\psi+\frac{1}{2}\rho\boldsymbol{v}\cdot\boldsymbol{v}\right)\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\dot{\psi}_{\mathrm{s}}+\eta_{\mathrm{s}}\dot{T}_{\mathrm{s}}\\
&\qquad+\psi_{\mathrm{s}}\nabla_{\mathrm{s}}\cdot\boldsymbol{v}_{\mathrm{s}}+\frac{1}{T_{\mathrm{s}}}\boldsymbol{h}_{\mathbf{t}}^{q}\cdot\nabla_{\mathrm{s}}T_{\mathrm{s}}\leq\\
&-\left[\!\!\left[\rho\boldsymbol{v}_{\mathrm{s}}\cdot\boldsymbol{v}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]-\left[\!\!\left[\boldsymbol{\sigma}\left(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v}\right)\right]\!\!\right]\cdot\boldsymbol{n}-\left[\nabla_{\mathrm{s}}\cdot\left(\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}\right)\right]\cdot\boldsymbol{v}_{\mathrm{s}}\\
&\qquad-\sum_{\alpha=1}^{N}\left(\left[\!\!\left[\rho\mu^{\alpha}\nu^{\alpha}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\boldsymbol{h}_{\mathbf{t}}^{\alpha}\cdot\nabla_{\mathrm{s}}\mu_{\mathrm{s}}^{\alpha}\right)\\
&\qquad+\nabla_{\mathrm{s}}\cdot\left(\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}}\right)
\end{aligned}\tag{2.30}$$

where, we have substituted for $[\![\boldsymbol{\sigma}\boldsymbol{v}]\!]\cdot\boldsymbol{n}$ from the expression derived in Eq. (E.3) of Appendix E. The right hand side indicates the work that is being done on the control volume due to surface tractions and diffusion. The free energy increase of the system is given to be less than or equal to the right hand side by the second law. Substituting $\nabla_{\mathrm{s}}\cdot\left(\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}}\right)=\left[\nabla_{\mathrm{s}}\cdot\left(\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}\right)\right]\cdot\boldsymbol{v}_{\mathrm{s}}+\mathbf{P}\boldsymbol{\sigma}_{\mathrm{s}}:\nabla_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}}$, and keeping in mind that $\boldsymbol{\sigma}_{\mathrm{s}}$ and $\mathbf{P}$ are symmetric, we get,

$$\begin{aligned}
&-\left[\!\!\left[\rho\left(\frac{\boldsymbol{v}}{2}-\boldsymbol{v}_{\mathrm{s}}\right)\cdot\boldsymbol{v}\left(v_{\mathrm{s_n}}-v_n\right)\right]\!\!\right]+\dot{\psi}_{\mathrm{s}}+\eta_{\mathrm{s}}\dot{T}_{\mathrm{s}}\\
&\qquad+\psi_{\mathrm{s}}\nabla_{\mathrm{s}}\cdot\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{\sigma}_{\mathrm{s}}\mathbf{P}:\left(\nabla_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}}\right)^{\mathrm{T}}\\
&-\left[\!\!\left[\left(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v}\right)\cdot\left(\rho\psi\mathbf{I}-\boldsymbol{\sigma}-\rho\sum_{\alpha=1}^{N}\mu^{\alpha}\nu^{\alpha}\mathbf{I}\right)\boldsymbol{n}\right]\!\!\right]\\
&\qquad+\sum_{\alpha=1}^{N}\boldsymbol{h}_{\mathbf{t}}^{\alpha}\cdot\nabla_{\mathrm{s}}\mu_{\mathrm{s}}^{\alpha}+\frac{1}{T_{\mathrm{s}}}\boldsymbol{h}_{\mathbf{t}}^{q}\cdot\nabla_{\mathrm{s}}T_{\mathrm{s}}\leq 0
\end{aligned}\tag{2.31}$$

Finally, it is postulated that the surface free energy is a function of temperature and a scalar state variable ξ_{s}. That is, $\psi_{\mathrm{s}}\equiv\psi_{\mathrm{s}}(\xi_{\mathrm{s}},T_{\mathrm{s}})$. Thus one can write,

$$\dot{\psi}_{\mathrm{s}}(\xi_{\mathrm{s}},T_{\mathrm{s}})=\frac{\partial\psi_{\mathrm{s}}}{\partial\xi_{\mathrm{s}}}\dot{\xi}_{\mathrm{s}}+\frac{\partial\psi_{\mathrm{s}}}{\partial T_{\mathrm{s}}}\dot{T}_{\mathrm{s}}\tag{2.32}$$

Ignoring inertial effects or the higher order velocity terms, utilizing the derivations for $(\nabla_{\mathrm{s}}\boldsymbol{v}_{\mathrm{s}})^{\mathrm{T}}$ and $\nabla_{\mathrm{s}}\cdot\boldsymbol{v}_{\mathrm{s}}$ given in Appendix D, Eq. (D.11) and Eq. (D.13), one can now rewrite Eq. (2.31) as

$$\begin{aligned}
&\frac{\partial\psi_{\mathrm{s}}}{\partial\xi_{\mathrm{s}}}\dot{\xi}_{\mathrm{s}}+\left(\frac{\partial\psi_{\mathrm{s}}}{\partial T_{\mathrm{s}}}+\eta_{\mathrm{s}}\right)\dot{T}_{\mathrm{s}}\\
&\qquad+\left(\psi_{\mathrm{s}}\mathbf{I}-\boldsymbol{\sigma}_{\mathrm{s}}\right)\mathbf{P}:\overset{\triangle}{\boldsymbol{F}}\boldsymbol{G}\\
&-\left[\!\!\left[(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v})\cdot\left(\rho\psi\mathbf{I}-\boldsymbol{\sigma}-\rho\sum_{\alpha=1}^{N}\mu^{\alpha}\nu^{\alpha}\mathbf{I}\right)\boldsymbol{n}\right]\!\!\right]\\
&\qquad+\sum_{\alpha=1}^{N}\boldsymbol{h}_{\mathbf{t}}^{\alpha}\cdot\nabla_{\mathrm{s}}\mu_{\mathrm{s}}^{\alpha}+\frac{1}{T_{\mathrm{s}}}\boldsymbol{h}_{\mathbf{t}}^{q}\cdot\nabla_{\mathrm{s}}T_{\mathrm{s}}\leq 0
\end{aligned}\tag{2.33}$$

where, $\overset{\triangle}{\boldsymbol{F}}$ is the convected time derivative of the deformation gradient defined in Eq. (D.12).

Since Eq. (2.33) has to be satisfied for arbitrary values of the rates $\dot{\xi}_{\mathrm{s}},\overset{\triangle}{\boldsymbol{F}},\dot{T}_{\mathrm{s}}$, the following conditions, which are analogous to bulk definitions of Eqs. (2.11) to (2.13), have to hold:

$$\frac{\partial\psi_{\mathrm{s}}}{\partial\xi_{\mathrm{s}}}=0 \tag{2.34}$$

$$\left(\psi_{\mathrm{s}}\mathbf{I}-\boldsymbol{\sigma}_{\mathrm{s}}\right)\mathbf{P}=0 \tag{2.35}$$

$$\eta_{\mathrm{s}}=-\frac{\partial\psi_{\mathrm{s}}}{\partial T_{\mathrm{s}}} \tag{2.36}$$

Furthermore, to satisfy Eq. (2.33), the following inequalities also have to hold on the interface,

$$-\left[\!\!\left[(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v})\cdot\left(\rho\psi\mathbf{I}-\boldsymbol{\sigma}-\rho\sum_{\alpha=1}^{N}\mu^{\alpha}\nu^{\alpha}\mathbf{I}\right)\boldsymbol{n}\right]\!\!\right]\ \leq 0 \tag{2.37}$$

$$\boldsymbol{h}_{\mathbf{t}}^{\alpha}\cdot\nabla_{\mathrm{s}}\mu_{\mathrm{s}}^{\alpha}\ \leq 0\quad\forall\alpha \tag{2.38}$$

$$\boldsymbol{h}_{\mathbf{t}}^{q}\cdot\nabla_{\mathrm{s}}T_{\mathrm{s}}\ \leq 0 \tag{2.39}$$

Eq. (2.35) implies that the rows of $(\psi_{\mathrm{s}}\mathbf{I} - \boldsymbol{\sigma}_{\mathrm{s}})$ are in the surface normal direction $\boldsymbol{n}$. This in turn implies that the quantity has unit rank with a single out of plane eigenvector. Thus, for arbitrary surface normal $\boldsymbol{n}$ and hence arbitrary $\mathbf{P}$, for condition Eq. (2.35) to hold

$$\det\left(\psi_{\mathrm{s}}\mathbf{I} - \boldsymbol{\sigma}_{\mathrm{s}}\right) = 0 \tag{2.40}$$

This condition is trivially achieved if the surface stress is homogeneous and isotropic, i.e., if $\boldsymbol{\sigma}_{\mathrm{s}} = \gamma\mathbf{I}$. The conditions Equations (2.38) and (2.39) are satisfied by constitutive relations of the following form:

$$\boldsymbol{h}_{\mathbf{t}}^{\alpha} = -\mathbf{M}_{\mathrm{s}}^{\alpha}\nabla_{\mathrm{s}}\,\mu_{\mathrm{s}}^{\alpha} \tag{2.41}$$

$$\boldsymbol{h}_{\mathbf{t}}^{q} = -\mathbf{k}_{\mathrm{s}}\nabla_{\mathrm{s}}\,T_{\mathrm{s}} \tag{2.42}$$

where, $\mathbf{M}_{\mathrm{s}}^{\alpha}$ is a positive definite surface mobility tensor for the species α, and $\mathbf{k}_{\mathrm{s}}$ is a positive definite heat conductivity tensor on the surface. The above equations are the surface analogs of the bulk versions of Fick's law and Fourier's Law (Equation (2.14) and Equation (2.15)).

As derived in Appendix F, in the absence of inertial forces, Equation (2.37) can be restated in the reference configuration using Eq. (F.6),

$$\boldsymbol{V}_{\mathrm{S}} \cdot [\![\boldsymbol{\Sigma}_{\nu} - \boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.43}$$

where,

$$\boldsymbol{\Sigma}_{\nu} = \boldsymbol{\Sigma}_0 - \rho_0 \sum_{\alpha=1}^{N} \mu_0^{\alpha}\nu_0^{\alpha}\mathbf{I} \tag{2.44}$$

is an extension to the Eshelby energy momentum tensor [103] $\boldsymbol{\Sigma}_0 = (\rho_0\psi_0\mathbf{I} - \nabla_0\boldsymbol{U}\boldsymbol{\sigma}_{\mathrm{I}})$, with $\boldsymbol{\sigma}_{\mathrm{I}}$ being the first Piola-Kirchoff stress tensor; $\boldsymbol{U}$ is the displacement in reference coordinates and Γ_{s_0} is the subsurface viewed in the reference configuration. Eshelby also suggests $\boldsymbol{\Sigma}_0^{*} = \boldsymbol{\Sigma}_0 - \boldsymbol{\sigma}_{\mathrm{I}}$ as an energy-momentum tensor with greater utility for estimating the force on a defect [103].

The jump $[\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_{\mathrm{I}}]\!]$ represents a conserved quantity that will integrate to zero within a homogeneous domain. In this sense, it is an extension to the path-independent J-integral of fracture mechanics [104] to bodies with multiple diffusing species. The conjugate to the reference interface velocity $\boldsymbol{V}_{\mathrm{s}}$, namely $[\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N}$, is the reference configurational force associated with the motion of the interface.

2.3 Conditions for Phase Growth

During the growth of the phases, Eq. (2.37) should be satisfied at each point on Γ and at all instants of time. It is convenient, however, to derive the growth criterion in the reference configuration using Eq. (2.43), specifically the alternative form of the equation as listed in Eq. (F.5). The surface velocity is first decomposed into tangential and normal components in the reference configuration $\boldsymbol{V}_{\mathrm{S}} = \boldsymbol{V}_{\mathbf{S_T}} + V_{\mathrm{S_N}}\boldsymbol{N}$, where $\boldsymbol{N}$ is the surface normal vector with the convention that $\boldsymbol{N}$ points from Phase $-$ into Phase $+$ (see Fig. 2.1). Thus, the left hand side of Eq. (2.43) can be written as

$$\begin{aligned}(\boldsymbol{V}_{\mathbf{S_T}} + V_{\mathrm{S_N}}\boldsymbol{N}) \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N} =& \boldsymbol{V}_{\mathbf{S_T}} \cdot \left[\!\!\left[\rho_0 \left(\psi_0 - \sum_{\alpha=1}^{N} \mu_0^\alpha \nu_0^\alpha \right) \mathbf{I} - \boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}_{\mathrm{I}} \right]\!\!\right] \boldsymbol{N} \\ &+ V_{\mathrm{S_N}}\boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N} \end{aligned} \tag{2.45}$$

where, $\boldsymbol{F}$ is the deformation gradient with the inverse $\boldsymbol{F}^{-1} = \mathbf{G}$. Simplifying, the second law condition of Eq. (2.45) reduces to requiring:

$$-[\![\boldsymbol{F}\boldsymbol{V}_{\mathbf{S_T}} \cdot \boldsymbol{\sigma}_{\mathrm{I}}\boldsymbol{N}]\!] + V_{\mathrm{S_N}}\boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.46}$$

In the above expression, it was possible to move $\boldsymbol{V}_{\mathbf{S_T}}$ and $\boldsymbol{N}$ inside the jump term since for a coherent interface (see Appendix D), $[\![\boldsymbol{N}]\!] = [\![\boldsymbol{V}_{\mathrm{S}}]\!] = 0$ and hence $[\![V_{\mathrm{S_N}}\boldsymbol{N}]\!] = 0$ leading to $[\![\boldsymbol{V}_{\mathrm{S}}]\!] = [\![V_{\mathrm{S_N}}\boldsymbol{N} + \boldsymbol{V}_{\mathbf{S_T}}]\!] = [\![\boldsymbol{V}_{\mathbf{S_T}}]\!] = 0$.

The relation between the normal in the reference configuration and in the current configuration is obtained using the differential surface relation Eq. (F.2):

$$\boldsymbol{N} = \frac{J_{\mathrm{s}}}{J^i} \boldsymbol{F}^{i^{\mathrm{T}}} \boldsymbol{n} \tag{2.47}$$

where, i represents either $+$ or $-$ phase and,

$$J_{\mathrm{s}} = \frac{d\Gamma_{\mathrm{s}}}{d\Gamma_{\mathrm{s}_0}} = \frac{J^i}{\sqrt{\boldsymbol{n} \cdot \boldsymbol{F}^i \boldsymbol{F}^{i^{\mathrm{T}}} \boldsymbol{n}}} \tag{2.48}$$

It should be noted that $J_{\mathrm{s}} > 0$, $[\![J_{\mathrm{s}}]\!] = 0$ due to coherence of the interface. Also, $\boldsymbol{F}\boldsymbol{V}_{\mathbf{S_T}} \cdot \boldsymbol{n} = 0$ since by its definition (see Eq. (2.47)) $\boldsymbol{n} = \frac{J^i}{J_{\mathrm{s}}} \boldsymbol{G}^{i^{\mathrm{T}}} \boldsymbol{N}$, and since $\boldsymbol{V}_{\mathbf{S_T}} \cdot \boldsymbol{N} = 0$. Thus, the quantity $\boldsymbol{F}\boldsymbol{V}_{\mathbf{S_T}}$ is tangential to the interface. Therefore, using Eq. (D.4), we obtain

$$V_{\mathrm{S_N}} = \frac{J_{\mathrm{s}}}{J^i} (\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}^i) \cdot \boldsymbol{n} \tag{2.49}$$

$$\boldsymbol{F}^i \boldsymbol{V}_{\mathbf{S_T}} = \mathbf{P}(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}^i) \tag{2.50}$$

This last quantity $\mathbf{P}(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}^i)$ is the difference in velocity in the tangential direction between the interface and the neighboring material point. Using the definition of the first Piola-Kirchoff stress (Eq. (F.3)) $\boldsymbol{\sigma}_{\mathrm{I}} \boldsymbol{N} = J_{\mathrm{s}} \boldsymbol{\sigma} \boldsymbol{n}$, we can now write the first term in Eq. (2.46) as

$$\boldsymbol{F}^i \boldsymbol{V}_{\mathbf{S_T}} \cdot \boldsymbol{\sigma}_{\mathrm{I}} \boldsymbol{N} = J_{\mathrm{s}} \mathbf{P}(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}^i) \cdot \boldsymbol{\sigma} \boldsymbol{n} \tag{2.51}$$

The right hand side is zero if there is no slip in the tangential direction between the surface and the adjacent material point, that is, if $\mathbf{P}\left(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}\right) = 0$.

Now, treating the dissipation due to tangential slip independent of the normal velocity causing the growth, the thermodynamic second law conditions for growth, Equation (2.46), may be stated as:

$$V_{\mathrm{S_N}} \boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_\mathrm{I}]\!] \boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.52}$$

$$-\boldsymbol{V}_{\mathbf{S_T}} \cdot \left[\!\!\left[\boldsymbol{F}^{\mathrm{T}} \boldsymbol{\sigma}_\mathrm{I} \right]\!\!\right] \boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.53}$$

with $\boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_\mathrm{I}]\!] \boldsymbol{N}$ representing a jump in volumetric energy density. The second condition is trivially satisfied in the absence of slip.

Recalling that the positive normal velocity points in the direction $\boldsymbol{N}$ of the interface, from the $-$ Phase to the $+$ Phase in the reference configuration, a convention is assumed that Phase $-$ is the growing phase and therefore, $V_{\mathrm{S_N}}$ is positive during the growth. Although it is not necessary to choose *a priori* the observed quantity, the surface normal speed $V_{\mathrm{S_N}}$ is chosen as a macroscopic, positive observed quantity. Therefore, Equations (2.52) and (2.53) can be restated to require that:

$$\boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_\mathrm{I}]\!] \boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.54}$$

$$V_{\mathrm{S_N}} = C_{V_N} \boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_\mathrm{I}]\!] \boldsymbol{N} \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.55}$$

$$\boldsymbol{V}_{\mathbf{S_T}} = -C_{V_T} \left[\!\!\left[\boldsymbol{F}^{\mathrm{T}} \boldsymbol{\sigma}_\mathrm{I} \right]\!\!\right] \boldsymbol{N} \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.56}$$

where, C_{V_N} and C_{V_T} are arbitrary positive quantities.

2.3.1 Growth Conditions in the Current Configuration

In this section, an alternative form of Eq. (2.54) is derived in the current configuration. Using Eq. (F.3) in the interface momentum balance equation Eq. (2.23), ignoring inertial effects, one can write

$$\begin{aligned} [\![\boldsymbol{\sigma}_\mathrm{I}]\!] \boldsymbol{N} d\Gamma_{\mathrm{s}_0} &= -\nabla_\mathrm{s} \cdot (\mathbf{P} \boldsymbol{\sigma}_\mathrm{s}) \, d\Gamma_\mathrm{s} \\ &= -\left(\kappa \boldsymbol{\sigma}_\mathrm{s} \boldsymbol{n} + \nabla_\mathrm{s} \cdot \boldsymbol{\sigma}_\mathrm{s} \right) J_\mathrm{s} d\Gamma_{\mathrm{s}_0} \end{aligned} \tag{2.57}$$

Substituting Eq. (2.57) into Eq. (2.54), and utilizing Equations (2.47) and (2.48), we get

$$\frac{1}{J_{\mathrm{s}}}\boldsymbol{N}\cdot[\![\boldsymbol{\Sigma}_{\nu}]\!]\,\boldsymbol{N}+\frac{\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\cdot(\kappa\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{n}+\nabla_{\mathrm{s}}\cdot\boldsymbol{\sigma}_{\mathrm{s}})\geq 0\quad\text{on}\quad\Gamma_{\mathrm{s}} \tag{2.58}$$

While the above equation is left in mixed form between current and reference configurations for ease of reading, it is possible to express the first term in the current configuration using and Eq. (F.4):

$$\frac{1}{J_{\mathrm{s}}}\boldsymbol{N}\cdot[\![\boldsymbol{\Sigma}_{\nu}]\!]\,\boldsymbol{N}=\left[\!\!\left[\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}\left(\rho\psi-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right)-\frac{\boldsymbol{n}\cdot\boldsymbol{F}\left(\boldsymbol{F}^{\mathrm{T}}-\mathbf{I}\right)\boldsymbol{\sigma}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\right]\!\!\right] \tag{2.59}$$

Similar to the above derivation, one can derive the following condition using Eqs. (2.50) and (2.51):

$$[\![\boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}_{\mathrm{I}}]\!]\,\boldsymbol{N}=J_{\mathrm{s}}\,[\![\boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}]\!]\,\boldsymbol{n}\quad\text{on}\quad\Gamma_{\mathrm{s}_0} \tag{2.60}$$

Thus, utilizing Eq. (2.50), one can write the second law condition of Eq. (2.53) in the current configuration as

$$-\mathbf{P}(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v}^{i})\cdot\boldsymbol{G}^{i\mathrm{T}}\,[\![\boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}]\!]\,\boldsymbol{n}\geq 0\quad\text{on}\quad\Gamma_{\mathrm{s}} \tag{2.61}$$

Substituting Eq. (2.59) into the growth condition into Eq. (2.58), we get the current configurations forms of Eqs. (2.54) and (2.55):

$$\begin{aligned}g_{sn}\equiv&\left[\!\!\left[\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}\left(\rho\psi-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right)-\frac{\boldsymbol{n}\cdot\boldsymbol{F}\left(\boldsymbol{F}^{\mathrm{T}}-\mathbf{I}\right)\boldsymbol{\sigma}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\right]\!\!\right]\\&+\frac{\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\cdot(\kappa\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{n}+\nabla_{\mathrm{s}}\cdot\boldsymbol{\sigma}_{\mathrm{s}})\geq 0\quad\text{on}\quad\Gamma_{\mathrm{s}}\end{aligned} \tag{2.62}$$

$$(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v}^{i})\cdot\boldsymbol{n}=c_{v_n}J^{i}g_{sn}\quad\text{on}\quad\Gamma_{\mathrm{s}} \tag{2.63}$$

$$\mathbf{P}(\boldsymbol{v}_{\mathrm{s}}-\boldsymbol{v}^{i})=-c_{v_t}\boldsymbol{G}^{i\mathrm{T}}\,[\![\boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}]\!]\,\boldsymbol{n}\quad\text{on}\quad\Gamma_{\mathrm{s}} \tag{2.64}$$

where, Eqs. (2.49) and (2.61) have been used to arrive at the above condtions. c_{v_n} and c_{v_t} are arbitrary, positive proportionality constants.

2.3.2 Statistical Growth Condition

In general, the atomistic phenomena that lead to phase nucleation and growth are statistical in nature. Thus, in experimental observations, the growth condition of Equation (2.54) requires a treatment keeping in mind the inherent variability at microscopic length scales. Considering now Equation (2.62), the alternative form of Equation (2.54), the equation is rewritten as

$$g_{sn} \equiv \left(H - \frac{G}{l} \right) \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{2.65}$$

where, H and G are volumetric and surface energy densities respectively, and l is a characteristic length scale in the current configuration as explained below. The definitions of H and G follow from Equation (2.62)

$$H\left(\boldsymbol{F}, \nu^{\alpha}, T\right) = \left[\!\!\left[\sqrt{\boldsymbol{n} \cdot \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}} \left(\rho\psi - \sum_{\alpha=1}^{N} \rho\mu^{\alpha}\nu^{\alpha} \right) - \frac{\boldsymbol{n} \cdot \boldsymbol{F}\left(\boldsymbol{F}^{\mathrm{T}} - \mathbf{I}\right)\boldsymbol{\sigma}\boldsymbol{n}}{\sqrt{\boldsymbol{n} \cdot \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}} \right]\!\!\right] \tag{2.66}$$

$$G\left(\boldsymbol{F}, \boldsymbol{\sigma}_{\mathrm{s}}, T\right) = -l \frac{\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}{\sqrt{\boldsymbol{n} \cdot \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}} \cdot \left(\kappa\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{n} + \nabla_{\mathrm{s}} \cdot \boldsymbol{\sigma}_{\mathrm{s}} \right) \tag{2.67}$$

The arguments in the above expressions explicitly indicate the independent physical variables that influence the quantities. For ease of reading, henceforth, the arguments will not be explicitly included when referring to these quantities. The explicit introduction of the length scale l in the definition of G nullifies the length scale dependence inherent in the expression on the right in Equation (2.62) due to the curvature and surface divergence terms.

Since, for an applied configurational force, the observed velocity in general is influenced by microstructural arrangement influenced by atomic scale uncertainty, a probabilistic treatment of the observed velocity is now considered for a given (de-

terministic) configurational force. A natural choice for the probabilistic distribution that captures the atomic uncertainty in $(v_{\mathrm{s_n}} - v_n)$ is the Boltzmann distribution [105]. Therefore, the probability density function for $(v_{\mathrm{s_n}} - v_n)$ is defined as

$$f(v_{\mathrm{s_n}} - v_n) \equiv f(c_E) = f_0 \exp\left[-\left(\frac{c_E g_{sn} l^3}{k_B T}\right)\right] \tag{2.68}$$

where, k_B is the Boltzmann constant, l is a characteristic length in the current configuration, and c_E is a positive non-dimensional measure that captures the uncertainty in the energy causing the interface motion. The unknown f_0 is evaluated by using the property of the probability density function

$$\int_0^\infty f(c_E)\, dc_E = 1 \rightarrow f_0 = \frac{g_{sn} l^3}{k_B T} \tag{2.69}$$

where, we have used the physical requirement that $g_{sn}, l > 0$. We now define a positive energy quantity E for ease of recognition, which is dimensionally of the form:

$$E = c_E g_{sn} l^3 \tag{2.70}$$

Finally, assuming the existence of a critical value of the energy E_c, the expected surface velocity resulting from the second law condition of Equation (2.63) is:

$$\begin{aligned} \left\langle v_{\mathrm{s_n}} - v_n^i \right\rangle &= \int_{E_c}^\infty (v_{\mathrm{s_n}} - v_n^i) f(E) dE \\ &= c_{v_n} J^i g_{sn} \exp\left[-\left(\frac{E_c}{k_B T}\right)\right] \end{aligned} \tag{2.71}$$

with g_{sn} as defined in Eqs. (2.62) and (2.65). The corresponding form in the reference configuration is

$$\langle V_{\mathrm{S_N}} \rangle = C_{V_N} \left(\boldsymbol{N} \cdot [\![\boldsymbol{\Sigma}_\nu - \boldsymbol{\sigma}_1]\!]\, \boldsymbol{N}\right) \exp\left[-\left(\frac{E_c}{k_B T}\right)\right] \tag{2.72}$$

2.3.3 Application to Small Deformation Diffusive Void Growth

Assuming small deformation from the definition of surface jacobian Eq. (2.48) we have,

$$\begin{aligned}\frac{J}{J_{\mathrm{s}}} &= \sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{T}\boldsymbol{n}} \\ &= \sqrt{\boldsymbol{n}\cdot(\mathbf{I}+\nabla\boldsymbol{u}^{T})(\mathbf{I}+\nabla\boldsymbol{u})\boldsymbol{n}} \\ &\approx \sqrt{\boldsymbol{n}\cdot(\mathbf{I}+\nabla\boldsymbol{u}^{T}+\nabla\boldsymbol{u})\boldsymbol{n}} \\ \frac{J}{J_{\mathrm{s}}} &\approx \sqrt{1+2\varepsilon_{nn}}\end{aligned} \tag{2.73}$$

where, $\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla\boldsymbol{u}+\nabla\boldsymbol{u}^{T}\right)$ is the infinitesimal strain tensor, $\varepsilon_{nn} = \boldsymbol{n}\cdot\boldsymbol{\varepsilon}\boldsymbol{n}$ is the normal strain and the quadratic terms of $\nabla\boldsymbol{u}$ are assumed to be negligible. We can now simplify Eq. (2.59) under assumptions of small deformations:

$$\begin{aligned}\frac{1}{J_{\mathrm{s}}}\boldsymbol{N}\cdot[\![\boldsymbol{\Sigma}_{\nu}]\!]\,\boldsymbol{N} &= \left[\!\!\left[\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}\left(\rho\psi-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right)-\frac{\boldsymbol{n}\cdot\boldsymbol{F}\left(\boldsymbol{F}^{\mathrm{T}}-\mathbf{I}\right)\boldsymbol{\sigma}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\right]\!\!\right] \\ &\approx \left[\!\!\left[\sqrt{1+2\varepsilon_{nn}}\left(\rho\psi-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right)-\frac{1}{\sqrt{1+2\varepsilon_{nn}}}\boldsymbol{n}\cdot\nabla\boldsymbol{u}\boldsymbol{\sigma}\boldsymbol{n}\right]\!\!\right] \\ &\approx \left[\!\!\left[\left(\rho\psi-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right)-\boldsymbol{n}\cdot\nabla\boldsymbol{u}\boldsymbol{\sigma}\boldsymbol{n}\right]\!\!\right] \\ &\approx \left[\!\!\left[\boldsymbol{n}\cdot\boldsymbol{\Sigma}\boldsymbol{n}-\sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right]\!\!\right]\end{aligned} \tag{2.74}$$

where, we have used the simplification $\boldsymbol{F}(\boldsymbol{F}^{T}-\mathbf{I}) = \boldsymbol{F}\boldsymbol{F}^{T}-\boldsymbol{F} \approx \mathbf{I}+\nabla\boldsymbol{u}^{T}+\nabla\boldsymbol{u}-(\mathbf{I}+\nabla\boldsymbol{u}^{T}) = \nabla\boldsymbol{u}$, and the small strain assumption, $\varepsilon_{nn} \ll 1$. $\boldsymbol{\Sigma}$ is the Eshelby Energy-

Momentum tensor defined as $\boldsymbol{\Sigma} = \rho\psi\mathbf{I} - \nabla\boldsymbol{u}\boldsymbol{\sigma}$. This reduces the growth condition of Eq. (2.58) to,

$$\begin{aligned}\left[\!\left[\boldsymbol{n}\cdot\boldsymbol{\Sigma}\boldsymbol{n} - \sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right]\!\right] &\geq -\frac{\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{n}}}\cdot(\kappa\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{n} + \nabla_{\mathrm{s}}\cdot\boldsymbol{\sigma}_{\mathrm{s}}) \\ &\geq -\frac{1}{\sqrt{1+2\varepsilon_{nn}}}(\mathbf{I}+\nabla\boldsymbol{u})\boldsymbol{n}\cdot(\kappa\boldsymbol{\sigma}_{\mathrm{s}}\boldsymbol{n} + \nabla_{\mathrm{s}}\cdot\boldsymbol{\sigma}_{\mathrm{s}})\end{aligned} \tag{2.75}$$

Now, if the surface stress is homogeneous and isotropic ($\boldsymbol{\sigma}_{\mathrm{s}} = \gamma\mathbf{I}$), and if strains are small $\varepsilon_{nn} \ll 1$, the above equation simplifies to,

$$\left[\!\left[\boldsymbol{n}\cdot\boldsymbol{\Sigma}\boldsymbol{n} - \sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha}\right]\!\right] + \kappa\gamma \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}} \tag{2.76}$$

Now, a growing phase is locally required to have negative curvature. Thus, a local radius of curvature $r = -\frac{1}{\kappa} > 0$ is defined, and assuming Phase $-$ to be void (values are zero inside the jump term), we get the condition for diffusive void growth as

$$\boldsymbol{n}\cdot\boldsymbol{\Sigma}\boldsymbol{n} - \sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha} - \frac{\gamma}{r} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}} \tag{2.77}$$

Finally, applying the statistical arguments of § 2.3.2, we arrive at the samll deformation diffusive void normal velocity as:

$$v_{\mathrm{s_n}} = c_{v_n}\left(\boldsymbol{n}\cdot\boldsymbol{\Sigma}\boldsymbol{n} - \sum_{\alpha=1}^{N}\rho\mu^{\alpha}\nu^{\alpha} - \frac{\gamma}{r}\right)\exp\left[-\left(\frac{E_c}{k_B T}\right)\right] \tag{2.78}$$

2.4 Criterion for Phase Nucleation

The nucleation of a new phase is modeled as the growth of an infinitesimally small embryo under the influence of the various thermodynamic forces. The uncertainty, however, is presumed to occur in the lengthscale l in Equation (2.68). Consider an infinitesimal nucleus of $\Omega_n = l^3$, with the bounding surface Γ_n. During nucleation, for a supplied energy density, the resulting length scale l is presumed uncertain and

thus, only some nuclei will continue to grow into the second phase. That is, the configurational force g_{sn} is deterministic, while the characteristic volume is uncertain. Thus,

$$l \geq l_c \tag{2.79}$$

with the critical length defined as $l_c = \frac{G}{H}, G \geq 0, H > 0$. During nucleation, it is expected that the length scale $l \ll 1$. Therefore, the length scale uncertainty that expresses through energy that varies as l^2 dominates over the energy that varies as l^3. Thus, representing the energy that varies as l^2 as $c_G G l^2$ $(G > 0)$ with c_G an unknown positive constant, the uncertainty in l may be expressed using the Boltzmann distribution as

$$f(l) = f_0 \exp\left[-\left(\frac{c_G G l^2}{k_B T}\right)\right] \tag{2.80}$$

The required condition for nucleation is then written as the probability that the length l will exceed the critical value l_c:

$$\begin{aligned} P(l > l_c) &= \int_{l_c}^{\infty} f(l) dl \\ &= 1 - \text{erf}\left[\sqrt{\frac{c_G G^3}{k_B T}} \frac{1}{H}\right] \\ &= 1 - \text{erf}\left[\frac{\hat{H}_c}{H}\right] \end{aligned} \tag{2.81}$$

with the definition $\hat{H}_c = \sqrt{\frac{c_G G^3}{k_B T}}$. $\hat{H}_c$ is a theoretical estimate of the characteristic value of H at which nucleation will accelerate, with $\text{erf}(x)$ being the error function. It is interesting to note the nonlinear dependence of the characteristic value $\hat{H}_c$ on the surface stress on the interface. A plot of the probability of nucleation against normalized H is shown in Figure 2.3. It is clear from the figure that the probability of nucleation is practically zero if $H \leq H_c$. A one percent change in the probability value occurs when $H \approx 0.55\hat{H}_c$.

The existence of a critical value beyond which nucleation occurs rapidly is supported experimentally [106,107]. While the value of H_c can be estimated theoretically

as in the $\hat{H}_c$ value, in general, the theoretical estimate may not correlate well with experimental observation due to errors in estimating the surface energy. The surface energy of a highly curved interface will be different from the bulk surface energy calculated for a flat surface. Thus, the value of H_c would in general be estimated experimentally. Finally, in the event $l_c = G = 0$, then, trivially, the probability $P(l > l_c) = 1$.

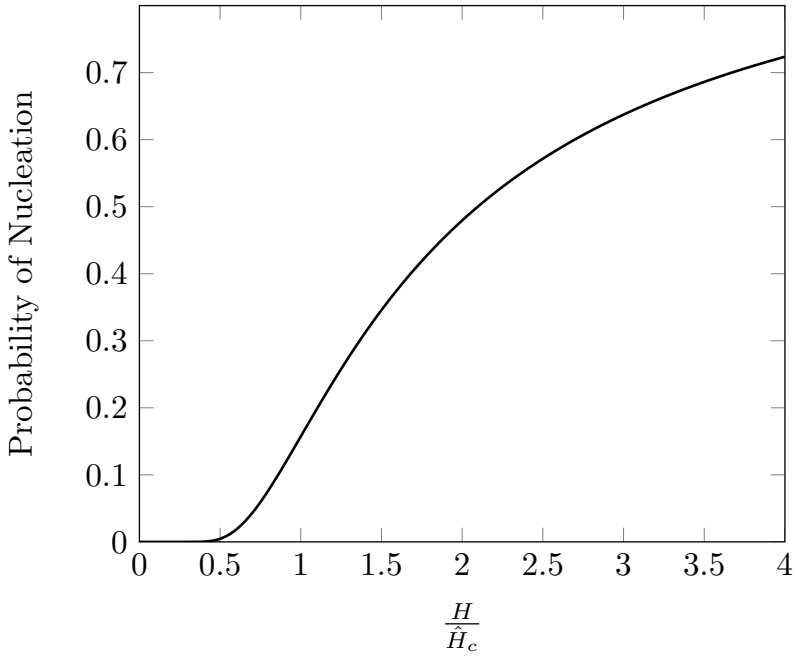


Figure 2.3. : Probabilitiy of nucleation versus $\frac{H}{\hat{H}_c}$. There is a sharp increase in the probability of nucleation beyond a characteristic value $H \geq H_c$. A one percent change in the probability value occurs when $H \approx 0.55\hat{H}_c$.

2.4.1 Reduction to Classical Nucleation Condition

In classical nucleation, the deformation and stress in the bulk are assumed negligible (pure diffusion) and that surface stress is isotropic ($\boldsymbol{\sigma}_{\mathrm{s}} = \gamma\mathbf{I}$), leading to $G = -l\kappa\gamma$. A growing nucleus of Phase $-$ will have a negative total curvature, therefore, in Equation (2.76), we consider $r_n = -\frac{1}{\kappa}$ as a characteristic radius of the growing nucleus in

the current configuration. This gives $G = \gamma$. For a viable nucleus, the right hand side of Eq. (2.76) has to be positive. Thus, Equation (2.65) reduces to

$$r_n \geq \frac{\gamma}{H} \tag{2.82}$$

where,

$$H = \left[\!\!\left[\rho\psi - \sum_{\alpha=1}^{N} \rho\mu^{\alpha}\nu^{\alpha} \right]\!\!\right] \tag{2.83}$$

H is the (deterministic) volumetric energy density for the growth of the nucleus. Specifically for a sphere, the length scale is $r_n = \frac{R}{2}$, where R is the radius of the sphere. The above criterion describes the minimum radius in the current configuration for viable nucleation in the presence of isotropic surface stress.

The classical nucleation theory [108, 109] suggests that when embryos of the new phase are too small to satisfy Eq. (2.82), they are unstable and collapse. Embryos above the critical size undergo stable growth to form viable nuclei. Since the embryos are very small, the energy of formation is predominantly the surface energy, varying as $G_n \approx c_\gamma \gamma r_n^2$, where c_γ is a positive constant relating the defined length scale to the surface area. For a sphere, $c_\gamma = 16\pi$. Following the procedure of the previous section, the probability of nucleation then is

$$P\left(r_n \geq \frac{\gamma}{H}\right) = 1 - \mathrm{erf}\left(\frac{\hat{H}_c}{H}\right) \tag{2.84}$$

where, $\hat{H}_c = \sqrt{\frac{c_\gamma \gamma^3}{k_B T}}$. The classical nucleation theory is thus a special case of the general criterion of Equation (2.81).

2.5 Electromigration Induced Void Nucleation and Growth

In the context of entropic inequality, the applied electrical field provides an external power to each diffusing species of the form $\boldsymbol{j}^\alpha \cdot \boldsymbol{e}^\alpha$ leading to modification of the constitutive equation for the bulk mass flux, Equation (2.14),

$$\boldsymbol{j}^\alpha = -\mathbf{M}^\alpha(\nabla\mu^\alpha + \boldsymbol{e}^\alpha) \quad \forall\alpha \tag{2.85}$$

Since the free energy is a function of only the deformation gradient, temperature and concentration, it is unaffected by the presence of the electromigration force. Similarly, the electrical field may be thought to add an external surface power term $\boldsymbol{h}_{\mathbf{t}}^\alpha \cdot \boldsymbol{e}_{\mathrm{s}}^\alpha$ leading to the modification of the surface constitutive equation, Equation (2.41),

$$\boldsymbol{h}_{\mathbf{t}}^\alpha = -\mathbf{M}_{\mathrm{s}}^\alpha(\nabla\mu_{\mathrm{s}}^\alpha + \boldsymbol{e}_{\mathrm{s}}^\alpha) \quad \forall\alpha \tag{2.86}$$

Since electromigration force causes a contribution only to the external power, the criterion for nucleation, Equation (2.84), remains unchanged. However, since the interface is now between a void phase and a solid phase, the nucleation criterion simplifies to:

$$\rho\psi - \sum_{\alpha=1}^{N}\rho\mu^\alpha\nu^\alpha \geq H_c \tag{2.87}$$

where, all the quantities are evaluated in the solid phase.

The experiments by Lane et al [10] may now be interpreted in light of Equation (2.87) as follows. The dominant path for the diffusion in copper is along the grain-boundaries. Therefore, for a copper line subject to electromigration force, the vacancy concentration at the cathode at a given time instant is independent of the adhesion energy of the interface (between the metal and barrier layer) binding the line. As the vacancy concentration increases at the cathode, the hydrostatic stress will also increase, and consequently there is an increase in the chemical potential and therefore H near the cathode. When H reaches a critical value, H_c, which depends

on the adhesion energy of the interface, a void will nucleate. Once a void nucleates, the growth of the void is governed by the magnitude of the configurational force at the interface. Since κ is negative for a convex void, the force is greater when the adhesion energy is smaller, and the void will grow at a faster rate at an interface with lower adhesion energy compared to one with a stronger adhesion.

2.6 Application: Critical Energy Density of Al-TiN Interface

In this section, the famous Blech's experiment [110] is simulated to estimate the critical value of (H_c) for Al-TiN interface. Figure 2.4 shows a schematic of the geometry used in Blech's experiment. The experiment consisted of a current carrying aluminum line deposited on a titanium nitride (TiN) layer. The ends of the TiN layer were connected to the electrical leads. Thus, the current enters the geometry through the TiN layer, passes through the Al line (due to the lower resistivity) and then exits once again through the TiN layer. From a historical perspective, this experiment is important because it elucidates the mechanism behind electromigration. Specifically, Blech identified a critical length of the interconnect line, for a given current density, now called the Blech length, below which no electromigration was observed. Furthermore, he observed that the product of the critical length and the current density was a constant. He then explained the observations by developing a model for how the vacancy concentration gradient in the line setup by the initial electromigration can balance the forced diffusion due to the electric current. When the vacancy concentration at the cathode is below a critical limit, electromigration voids are not observed.

It shall now be shown that using the notion of the Blech-length, one can estimate the value of H_c. This then provides an easy method to determine void nucleation criterion in metals. Figure 2.6 shows a portion of the geometry that was used to simulate Blech's experiment. The length of the Al line was chosen to be equal to $10\mu m$. The interface between Al and TiN was modeled as a distinct material with

thickness equal to $0.1\mu m$ but with higher diffusivity (see Tab. 2.1). It is known from Blech's experiments that for Al tested in a geometry of this type, the critical value of jL was equal to $(jL)_c = 1260\ A/cm$. In order to simulate electromigration, it is necessary to simultaneously solve three sets of partial differential equations [4, 6]: (1) the electrical charge continuity equation, (2) the vacancy diffusion equation and (3) the stress-equilibrium equation and the coupling between them. The stress problem was solved by modifying the elastic constitutive behavior of Al to account for the dilational strain that is induced due to vacancy concentration variations. Electromigration was modeled by solving an advection diffusion equation over the Al and interface domains. Diffusion was not simulated in the TiN region, as it is known to be resistant to electromigration.

The electrical boundary conditions in the simulations were as shown in Figure 2.4. The electric potential was applied at the ends of TiN layer corresponding to the critical current density, $(jL)_c$. All the boundaries were made impervious to vacancy diffusion during the simulations. The initial vacancy concentration was assumed to be 0.9 mol/m^3 roughly corresponding to a concentration of 1 vacancy for every 10^6 atoms of copper. All the boundaries were held fixed during the elastic stress analysis. The line was assumed to be initially stress-free.

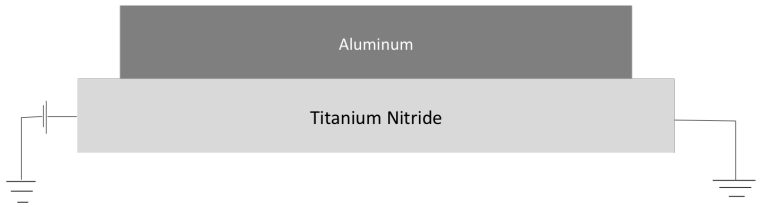


Figure 2.4. : Schematic of the Blech-structure that was used in the simulation.

Figure 2.5 shows the variation of H at the cathode end as a function of time. The onset of electromigration equilibrium occurs when $t \approx 15000s$. Figure 2.7 shows the spatial variation of several physical variables in the Blech structure once vacancy concentration equilibrium was achieved. Fig. 2.7a shows the electrical current density, $\boldsymbol{j} = \sigma \nabla \phi$, where ϕ is the electrical potential and the σ is the electrical conductivity.

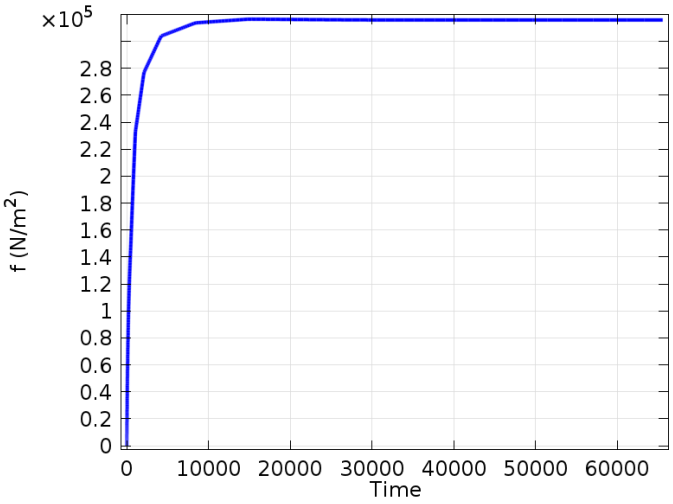


Figure 2.5. : Variation of H at the cathode as a function of time.

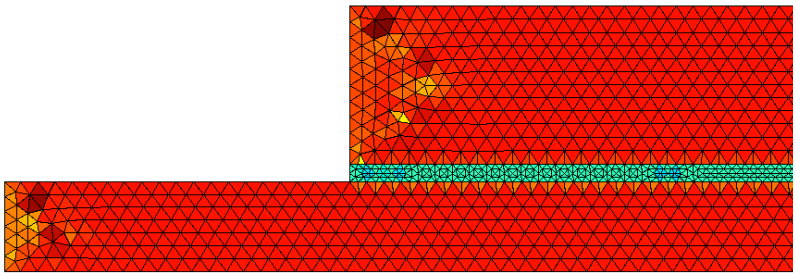


Figure 2.6. : Mesh of the Blech structure with the color representing the size of the elements.

Table 2.1. : Table of Material Properties.

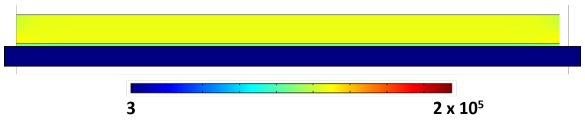
Material Property	**Value**
Aluminum Diffusivity	$1 \times 10^{-14}\ m^2/s$
Interface Diffusivity	$1 \times 10^{-}6\ m^2/s$
Al Young's Modulus	70×10^9 Pa
TiN Young's Modulus	500×10^9 Pa
Al Electrical Conductivity	3.774×10^7 S/m
TiN Electrical Conductivity	2.6×10^4 S/m

Since the conductivity is higher in Al compared to TiN, the current density is correspondingly higher. Fig. 2.7b shows the distribution of vacancy concentration. As might be expected, the vacancy concentration is higher at the cathode than at the anode. Consequently, the hydrostatic stress is tensile at the cathode and compressive at the anode. Due to the higher vacancy concentration and the tensile hydrostatic stress, H is also higher at the cathode. Since it is known that it is at this value of H that voids begin to nucleate at the cathode, this H is equal to H_c. Hence, the Blech experiment may be used to determine the value of H_c to predict void nucleation in a material.

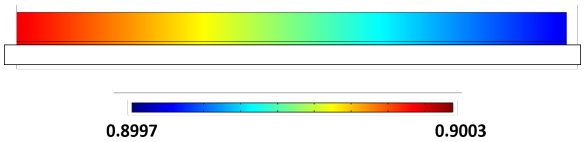
The simulations indicate that the value of $H_c = 5.5$ Joules/cm^3 for the Al-TiN interface. The value of H_c being an intrinsic property of the interface, it does not depend on the diffusivities of the materials in the test structure. But, to accurately predict the time for void nucleation, the diffusivities through the bulk, grain-boundaries and through the metal-dielectric interfaces will need to be known. In general, as in fracture mechanics, in a structure with multiple interfaces, the knowledge of H alone is insufficient to predict void location since the critical value of H_c depends on the specific interface.

2.7 Summary

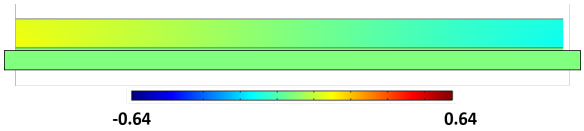
In conclusion, the free energy inequality on a moving interface naturally yields: (1) constitutive equations relating surface free energy to the surface stress, (2) surface



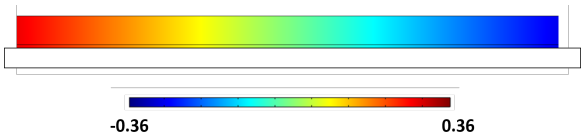
(a) Electrical current density (A/cm^2)



(b) Vacancy concentration (moles/m^3)



(c) Hydrostatic stress (MPa)



(d) H (Joules/cm^3)

Figure 2.7. : Spatial variation of the various physical quantities at equilibrium.

entropy to the surface free energy, (3) constitutive equation relating surface species mass flux to the gradient of surface chemical potential and (4) surface heat flux to the gradient of surface temperature respectively. The derived interface second law condition was shown to lead to a configurational force associated with the interface evolution that naturally extends the Eshelby energy momentum tensor to problems with species diffusion. Also, the dependence on the jump in the free energy and the elastic misfit energy are shown to emerge naturally from the thermodynamic restrictions at the interface. This configurational force was then used to develop the criteria for growth and nucleation of a phase in a body with multiple diffusing species undergoing finite deformation, and with arbitrary surface stress. The interface growth and nucleation criteria developed in this chapter are thermodynamic restrictions and as such do not presuppose any constitutive relation for the bulk or surface diffusional flux. The phase growth condition is statistical to account for the statistical nature of the atomistic phenomena that lead to phase growth. Conditions for small deformation diffusive void growth as well as the classical nucleation theory are obtained as special cases of the developed general criteria for growth and nucleation respectively. It is shown that a critical material-dependent energy density that must be overcome for nucleation (H_c) naturally arises as a consequence of the statistical distribution of nucleus sizes, and the existence of a critical nucleus size. As an application, the developed theory is used to study inherent material resistance to void nucleation at Al-TiN interface. Through a simulation of Blech's experiments, the critical energy density for the interface is estimated as $H_c = 5.5$ Joules/cm^3.

3. ENRICHED ISOGEOMETRIC ANALYSIS

To simulate phase evolution problems numerically, an explicit boundary tracking approach called Enriched Isogeometric Analysis (EIGA) [59] is adopted in this thesis. It shall be shown in later chapters how this method can be used to model topological changes without having to compute surface-surface intersections. In EIGA, each phase interface is explicitly described by a parametric spline. The behavioral field is described as a weighted blending of a continous approximation associated with the underlying domain, and enrichments representing the influence of the interfaces. The influence of an interface is generally expected to decrease with distance from the interface. Thus, the enrichments require a measure of distance from the interface. Approximations for the behavioral fields are usually chosen to be isoparametric with the geometric models to facilitate CAD-CAE integration. In this chapter, the Hierarchical Partition of Unity Field Compositions (HPFC) method will be introduced, from which the EIGA method shall be developed. Finally, as an illustration, the EIGA method is used to model the electrostatic problem of a current carrying metal line with a void.

3.1 Hierarchical Partition of Unity Field Compositions

The HPFC method was developed by Subbarayan et al. [111, 112] based on the Partition of Unity Finite Element Method [113]. The domain of interest is constructed as hierarchical compositions of simpler primitives (see Fig. 3.1). Analogous to the CAD procedure of Constructive Solid Geometry (CSG), the geometry, materials and

behavioral fields over the domain ($\boldsymbol{f}(\boldsymbol{x})$) are given by weighted compositions of fields defined on the constituent primitives (Ω_i),

$$\boldsymbol{f}(\boldsymbol{x}) = \sum_i w_i(\boldsymbol{x}) \boldsymbol{f}_{\Omega_i}(\boldsymbol{x}) \tag{3.1}$$

where, w_i are weight functions. To ensure convergence of the analysis, the weight functions must satisfy,

$$\sum_i w_i(\boldsymbol{x}) = 1 \qquad \forall\ \boldsymbol{x} \tag{3.2a}$$

$$0 \leq w_i(\boldsymbol{x}) \leq 1 \qquad \forall\ i, \boldsymbol{x} \tag{3.2b}$$

$$\|w_i(\boldsymbol{x})\|_\infty \leq C_\infty \qquad \forall\ i, \boldsymbol{x} \tag{3.2c}$$

$$\|\nabla w_i(\boldsymbol{x})\|_\infty \leq \frac{C_G}{\text{diam } \Omega_\text{i}} \quad \forall\ i, \boldsymbol{x} \tag{3.2d}$$

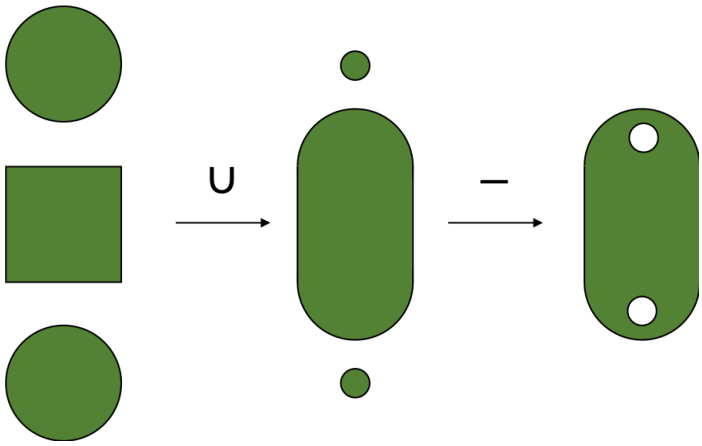


Figure 3.1. : Complex domain constructed as hierarchical compositions of simple primitives in a CSG procedure.

3.2 Enriched Field Approximations

An extension to the HPFC method, called Enriched Isogemetric Analysis (EIGA) was developed by Tambat et al. [59], composing a continuous domain with lower-

dimensional geometries such as evolving interfaces. Consider interfaces Γ_i embedded in an underlying domain Ω, with associated behavioral field approximations $\boldsymbol{f}_{\Gamma_i}$ and $\boldsymbol{f}_{\Omega}$ respectively. The EIGA formulation for any behavioral field $\boldsymbol{f}(\boldsymbol{x})$ can then be written as,

$$\boldsymbol{f}(\boldsymbol{x}) = \left(1 - \sum_i w_i(\boldsymbol{x})\right) \boldsymbol{f}_{\Omega}(\boldsymbol{x}) + \sum_i w_i(\boldsymbol{x}) \boldsymbol{f}_{\Gamma_i}\left(\boldsymbol{x}, \mathcal{P}(\boldsymbol{x})\right) \tag{3.3}$$

where, $w_i(\boldsymbol{x})$ is the weight field associated with the i^{th} interface Γ_i. Since the enrichments are defined on lower-dimensional geometries, their influence at any point $\boldsymbol{x}$ in space is dictated by its projection $\mathcal{P}(\boldsymbol{x})$ on to the lower-dimensional geometry. Interface boundary conditions are directly imposed on the degrees of freedom associated with the enrichments. An illustration is provided in Fig. 3.2.

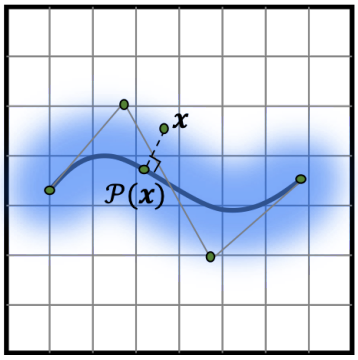


Figure 3.2. : Behavioral field obtained as a composition of a continuous approximation over the underlying domain, and an enrichment associated with the interface. The influence of the interface is limited to a region in its neighborhood.

3.2.1 Choice of Weight Function

Interfaces exert influence on behavioral fields in their neighborhood. This influence generally weakens with distance away from the interfaces. To capture this weakening influence, weight functions based on interface distance are used for blending of enrichments in Enriched IGA. Several weight functions have been proposed

in literature, such as inverse distance weighting [114], spline functions [115, 116] and exponential functions [59].

Consider a single enrichment system as shown in Fig. 3.2. A weight function $w(d)$, varying with distance d from the enrichment is assumed. The influence of the enrichment is restricted to the region within a cutoff distance d_s that depends on the length-scale dictated by the geometry and governing physics. Thus, the behavioral field and its gradient are given by,

$$\boldsymbol{f}(\boldsymbol{x}) = (1 - w(d))\boldsymbol{f}_\Omega(\boldsymbol{x}) \quad + w(d)\boldsymbol{f}_\Gamma\left(\boldsymbol{x}, \mathcal{P}(\boldsymbol{x})\right) \tag{3.4a}$$

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{x}) = (1 - w(d))\nabla\boldsymbol{f}_\Omega(\boldsymbol{x}) + w(d)\nabla\boldsymbol{f}_\Gamma\left(\boldsymbol{x}, \mathcal{P}(\boldsymbol{x})\right) \\ - \nabla w\Big(\boldsymbol{f}_\Omega(\boldsymbol{x}) - \boldsymbol{f}_\Gamma\left(\boldsymbol{x}, \mathcal{P}(\boldsymbol{x})\right)\Big)\end{aligned} \tag{3.4b}$$

In order for the interface solution and its gradient to be exact on an interface, the weight function should be unity and its gradient zero, on the interface (i.e. $d = 0$). Further, the weight function and its gradient should vanish at the cutoff distance d_s, so that the influence of the enrichment falls smoothly to zero. Mathematically,

$$w(0) = 1 \tag{3.5a}$$

$$w(\pm d_s) = 0 \tag{3.5b}$$

$$\nabla w(0) = 0 \tag{3.5c}$$

$$\nabla w(\pm d_s) = 0 \tag{3.5d}$$

where, the sign of the distance is used to differentiate the two sides of an enrichment. In this thesis, cubic and quartic weight functions that satisfy all conditions of Eq. (3.5) are used, shown in Fig. 3.3,

$$w(d) = 2\left|\frac{d}{d_s}\right|^3 - 3\left|\frac{d}{d_s}\right|^2 + 1 \tag{3.6a}$$

$$w(d) = -3\left|\frac{d}{d_s}\right|^4 + 8\left|\frac{d}{d_s}\right|^3 - 6\left|\frac{d}{d_s}\right|^2 + 1 \tag{3.6b}$$

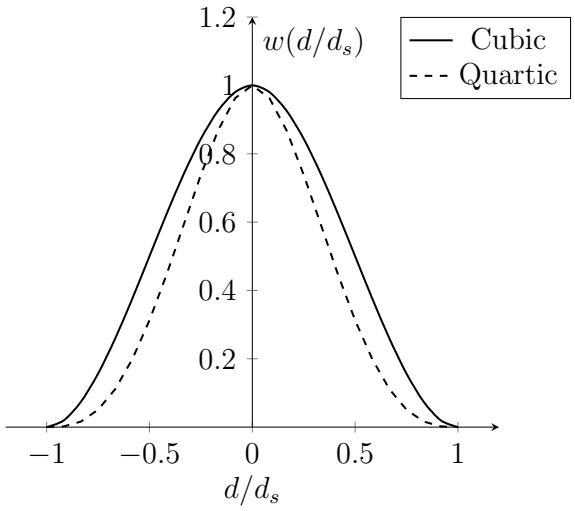


Figure 3.3. : Cubic and quartic weight functions used for weighted blending of continuous approximation and enrichments.

Since the behavioral field approximation at a point depends on its distance from enrichments such as phase interfaces, EIGA requires a computationally efficient measure of distance of a point from an interface. Such a distance measure called signed algebraic level sets shall be introduced in the following chapters. The interface degrees of freedom influencing a particular point in the domain are obtained from its projection on to the interface. Thus, a computationally efficient point projection procedure is necessary, and an algebraic approach for the same shall be introduced in subsequent chapters. Finally, Boolean compositions on signed algebraic level sets are described in Chapter 7 to enable modeling topological changes such as merging without having to compute intersections between phase interfaces.

3.3 Application: Current Through Line with Void

As an application, the electrostatic problem of a current carrying metal line with a void is considered. This is of relevance in studying void growth due to electromigration, which is a failure concern in the semiconductor industry. A formal description of the electromigration problem follows (see Fig. 3.4a for reference). A rectangular

domain Ω, containing an arbitrarily shaped void is considered. To allow irregular shapes, voids are represented using NURBS curves. The system is governed by the Laplace equation on the electric potential ϕ,

$$\nabla^2 \phi = 0 \ \text{ in } \Omega \tag{3.7}$$

Dirichlet boundary conditions are applied at the top and bottom surfaces, and the walls are assumed to not allow electric flux to flow through them. Additionally, there is no flux entering or exiting the surface of a void Γ_e,

$$\frac{\partial \phi}{\partial n} = 0 \ \text{ on } \Gamma_e \tag{3.8}$$

The cubic weight function in Eq. (3.6a) is used to express the electric potential as a weighted blending of a continuous approximation ϕ_c, and an enrichment ϕ_e associated with the void surface,

$$\phi(\boldsymbol{x}) = (1 - w(\boldsymbol{x}))\, \phi_c(\boldsymbol{x}) + w(\boldsymbol{x}) \phi_e(\mathcal{P}(\boldsymbol{x})) \tag{3.9}$$

An advantage of this form for the electric potential is that it can be shown to automatically satisfy Eq. (3.8) on the void boundary. For any point on the void boundary, where the distance $d = 0$, we have from Eq. (3.5) that $w = 1, \nabla w = 0$. Thus, on the boundary,

$$\begin{aligned} \frac{\partial \phi}{\partial n} &= (1 - w) \nabla \phi_c(\boldsymbol{x}) \cdot \boldsymbol{n} + w \nabla \phi_e(\mathcal{P}(\boldsymbol{x})) \cdot \boldsymbol{n} + \nabla w \cdot \boldsymbol{n} \left(-\phi_c + \phi_e \right) \\ &= \nabla \phi_e(\mathcal{P}(\boldsymbol{x})) \cdot \boldsymbol{n} \end{aligned} \tag{3.10}$$

At any point $\boldsymbol{x}$ on the boundary, traveling along the normal direction does not change the projection $\mathcal{P}(\boldsymbol{x})$. This gives $\nabla \mathcal{P}(\boldsymbol{x}) \cdot \boldsymbol{n} = 0$, and by extension $\nabla \phi_e(\mathcal{P}(\boldsymbol{x})) \cdot \boldsymbol{n} = 0$, and Eq. (3.8) is satisfied. As a result, only the governing Laplace equation with Dirichlet boundary conditions need to be solved. The blended form assumed for the electric potential ensures that the interface conditions are obeyed exactly. Some

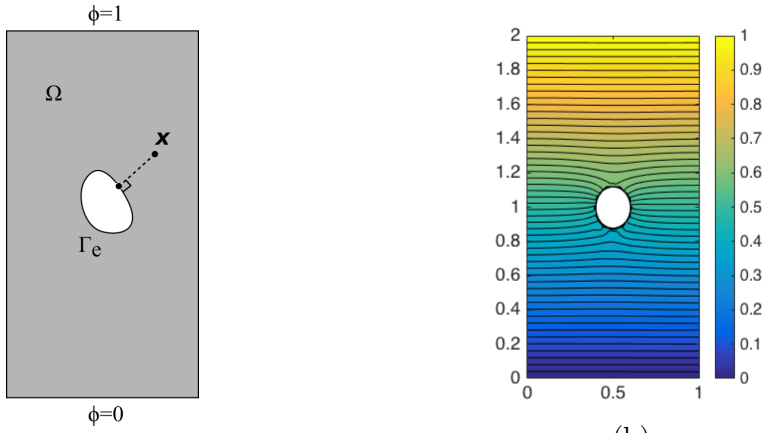
problems also involve non-homogeneous interface conditions, such as passing current through a line with an inclusion. For such problems, an additional gradient degree of freedom is required at the void interface, as used in [89].

$$\phi(\boldsymbol{x}) = (1 - w(\boldsymbol{x}))\phi_c(\boldsymbol{x}) + w(\boldsymbol{x})\left(\phi_e(\mathcal{P}(\boldsymbol{x})) + G_e(\mathcal{P}(\boldsymbol{x}))\right) \tag{3.11}$$

The system is solved using isogeometric analysis for an elliptical void, and the resulting potential solution is shown in Fig. 3.4b. The obtained results can be interpreted as follows. For a system with no voids, the analytical solution at a point $\boldsymbol{x} \equiv (x, y)$ is given by,

$$\phi(\boldsymbol{x}) = \frac{y}{h} \tag{3.12}$$

where, h is the height of the metal line. The potential solution varies linearly from the bottom surface to the top. This satisfies both the governing Laplace equation, as well as the Dirichlet and wall boundary conditions. Such a solution is characterized by straight, horizontal contour lines. As can be seen in Fig. 3.4b, far away from the void, the potential contour lines are unaffected and remain straight and horizontal. However in the neighbourhood of the void, the contours are distorted in order to satisfy Eq. (3.8). The use of an explicit interface representation along with the assumed form for the potential approximation in Eq. (3.9) has allowed imposing the void interface conditions exactly.



(a) (b)

Figure 3.4. : (a) Schematic of the electromigration problem for a rectangular domain. A sample point and its projection on to the void interface are illustrated. (b) Contours of the electric potential solution obtained for a system with a single void.

4. SIGNED ALGEBRAIC LEVEL SETS

For effective application of enriched isogeometric analysis, a fast and monotonic measure of signed distance is necessary, with the sign of the distance distinguishing points lying on either side of the enrichment. In this chapter, a distance measure proposed by Upreti et al. [84, 85], called signed algebraic level sets is introduced. Algebraic level sets are obtained from the implicitization of the enrichment geometry, followed by normalization and trimming. A bounding box procedure was proposed in [85] to ascribe sign and thus classify points as lying inside, on or outside closed geometries. Since the procedure to generate these signed level sets is completely analytical, it is computationally very efficient. Further, these level sets are smooth, increase monotonically with distance, and are exact on the boundary, making it apt for enriched isogeometric analysis.

4.1 Background: Implicitization of Parameteric Geometries

Surfaces and curves may be expressed with an implicit or parametric representation. Most CAD systems use parametric representations such as NURBS which provide an easier and more intuitive control for users. On the other hand, the implicit representation of a surface allows natural generation of level sets that increase monotonically with distance, thereby serving as a measure of distance. It is hence desirable to obtain the equivalent implicit representation for a given parametric curve or surface. Since spline geometries are piece-wise polynomial, they are first decomposed into constituent Bézier polynomial segments through knot insertion [80, 117, 118]. Each Bézier segment is then implicitized using the Dixon resultant [119] from Elimination theory, as shown in [86]. Resultants are polynomial expressions on the coefficients of a given system of polynomial equations. The given system of equations have a common

solution only if their resultant vanishes. For example, for a linear system $\boldsymbol{Ax} = 0$, the determinant of the coefficient matrix, $|\boldsymbol{A}|$ depends only on the coefficients of the system and behaves as a resultant; the system has a non-trivial common solution only if this determinant vanishes. The Dixon resultant is obtained as an n-dimensional extension of the Cayley-Bezout resultant [120], and shall be discussed presently. This is followed by a brief description of an alternative implicitization procedure developed by L. Busé [121]. While the procedure described here is for three-dimensional surfaces, it can be adapted for planar curves in a straight-forward manner.

4.1.1 Dixon Resultant

Rational parametric representations such as Bézier and NURBS have the general form,

$$x(u,v) = \frac{X(u,v)}{W(u,v)}, y(u,v) = \frac{Y(u,v)}{W(u,v)}, z(u,v) = \frac{Z(u,v)}{W(u,v)} \tag{4.1}$$

where, X, Y, Z, W are functions in the parameters (u, v), with degree m in u and n in v. Such representations can be converted into a polynomial system of equations,

$$\begin{aligned} X(u,v) - xW(u,v) &= 0 \\ Y(u,v) - yW(u,v) &= 0 \\ Z(u,v) - zW(u,v) &= 0 \end{aligned} \tag{4.2}$$

Consider the following determinant where α, β are any constants,

$$\begin{vmatrix} xW(u,v) - X(u,v) & yW(u,v) - Y(u,v) & zW(u,v) - Z(u,v) \\ xW(u,\beta) - X(u,\beta) & yW(u,\beta) - Y(u,\beta) & zW(u,\beta) - Z(u,\beta) \\ xW(\alpha,\beta) - X(\alpha,\beta) & yW(\alpha,\beta) - Y(\alpha,\beta) & zW(\alpha,\beta) - Z(\alpha,\beta) \end{vmatrix}$$

This determinant is zero whenever $u = \alpha$ or $v = \beta$, and hence $(u - \alpha)$ and $(v - \beta)$ are factors of the determinant. Define,

$$\delta(\boldsymbol{x}) = \frac{1}{(u-\alpha)(v-\beta)} \begin{vmatrix} xW(u,v) - X(u,v) & yW(u,v) - Y(u,v) & zW(u,v) - Z(u,v) \\ xW(u,\beta) - X(u,\beta) & yW(u,\beta) - Y(u,\beta) & zW(u,\beta) - Z(u,\beta) \\ xW(\alpha,\beta) - X(\alpha,\beta) & yW(\alpha,\beta) - Y(\alpha,\beta) & zW(\alpha,\beta) - Z(\alpha,\beta) \end{vmatrix}$$

For points on the surface, the first-row entries are zero, and the determinant is zero irrespective of α, β. Hence all points on the surface satisfy,

$$\delta(\boldsymbol{x}) = 0 \quad \forall\ \alpha, \beta \in \mathbb{R} \tag{4.3}$$

Now, the quantity δ depends on α, β, u and v, and can be expanded to separate these factors as,

$$\delta = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{m-1}\beta^{2n-1} \end{bmatrix} [\boldsymbol{M_D}(\boldsymbol{x})] \begin{bmatrix} 1 \\ u \\ u^2 \\ \vdots \\ u^{2m-1}v^{n-1} \end{bmatrix} = [\boldsymbol{\alpha}][\boldsymbol{M_D}(\boldsymbol{x})][\boldsymbol{u}] \tag{4.4}$$

For the determinant to vanish for all α, β, we have,

$$[\boldsymbol{M_D}(\boldsymbol{x})][\boldsymbol{u}] = 0 \tag{4.5}$$

$$|\boldsymbol{M_D}(\boldsymbol{x})| = 0 \tag{4.6}$$

This forms a necessary condition for a given point $\boldsymbol{x}$ to lie on the parametric surface and can act as its implicit equation. The $2mn \times 2mn$ determinant in Eq. (4.6) is called the Dixon resultant and was derived by Dixon [119] in an involved procedure

expressing X, Y, Z, W in the power basis of the parameters. The corresponding Dixon matrix is linear in x, y, z and can be written as,

$$\boldsymbol{M_D}(\boldsymbol{x}) = \mathbb{M}_w + \mathbb{M}_x x + \mathbb{M}_y y + \mathbb{M}_z z \tag{4.7}$$

where each of the coefficient matrices are constants and independent of u, v or $\boldsymbol{x}$. In [119], it is shown that the coefficient of $\alpha^i \beta^j u^k v^l$ in the expansion of δ in Eq. (4.4) can be written as,

$$\boldsymbol{M_D}(i, j, k, l) = \sum (d_{IJ} x - a_{IJ}, d_{KL} y - b_{KL}, d_{MN} z - c_{MN}) \tag{4.8}$$

summed over all integers $I, J, K, L', M', N \geq 0$ such that,

$$\begin{aligned} L &= L' + j + 1 \\ M &= M' + k + 1 \\ I + K + M' &= i \\ J + L' + N &= l \end{aligned} \tag{4.9}$$

Here, a, b, c, d are the coefficients in the expansion of X, Y, Z, W, written as,

$$\begin{aligned} X(u, v) &= \sum_{K,L} a_{KL} u^K v^L \\ Y(u, v) &= \sum_{K,L} b_{KL} u^K v^L \\ Z(u, v) &= \sum_{K,L} c_{KL} u^K v^L \\ W(u, v) &= \sum_{K,L} d_{KL} u^K v^L \end{aligned} \tag{4.10}$$

The parentheses in Eq. (4.8) denote a determinant, with the notation as used in [86],

$$(a_{IJ}, b_{KL}, c_{MN}) = \begin{vmatrix} a_{IJ} & b_{IJ} & c_{IJ} \\ a_{KL} & b_{KL} & c_{KL} \\ a_{MN} & b_{MN} & c_{MN} \end{vmatrix}$$

Constructing the Dixon Matrix

From Eq. (4.8), a simple procedure to build the coefficient matrices in Eq. (4.7) and hence the Dixon matrix $\boldsymbol{M_D}$ is to loop over all possible combinations of I, J, K, L, M, N and assign at the appropriate location in $\boldsymbol{M_D}$. Such an algorithm for building the Dixon resultant is provided in [86] and a slightly modified version is presented here (see Algm. 1). The degree of the surface is assumed to be n in both parameters.

4.1.2 Implicitization using M-Rep

An alternative implicitization procedure was proposed by L. Busé [121,122] called the Matrix Representation or the M-Rep. As a generalization of the moving lines implicitization method [123], this approach constructs for a given curve or surface, a matrix with elements varying in space. The matrix is constructed such that its rank drops exactly on the curve or surface. Consider a tensor product Bézier surface $\boldsymbol{f}(u,v) = \begin{bmatrix} f_0 & f_1 & f_2 & f_3 \end{bmatrix}$ with degree (d_1, d_2). Let $B_i^n(u) = \binom{n}{i}(1-u)^i u^{n-i}$ denote the i^{th} Bernstein polynomial of order n. The M-Rep for this surface is constructed as follows (refer [121] for further details). An orthogonal Bézier surface $\boldsymbol{g}(u,v)$ is constructed with degree $\boldsymbol{\nu} \equiv (\nu_1, \nu_2) \geq (d_1 - 1, 2d_2 - 1)$ [or $\boldsymbol{\nu} \geq (2d_1 - 1, d_2 - 1)$] [122, 124, 125],

$$\boldsymbol{g}(u,v) = \sum_{i=0}^{\nu_1} \sum_{j=0}^{\nu_2} \boldsymbol{\alpha}_{ij} B_i^{\nu_1}(u) B_j^{\nu_2}(v) \tag{4.11}$$

$$\boldsymbol{f}(u,v) \cdot \boldsymbol{g}(u,v) = 0 \qquad \forall\ (u,v) \tag{4.12}$$

Algorithm 1 Dixon Resultant Assembly

Require: $n, a_{IJ}, b_{IJ}, c_{IJ}, d_{IJ}$
  for $I, J, K, N = 0$ to n **do**
    for $L, M = 1$ to n **do**
      if ($IJ == KL$ or $KL == MN$ or $MN == IJ$) **then**
        cycle
      end if
      $D_x = (d_{IJ}, -b_{KL}, -c_{MN})$
      $D_y = (-a_{IJ}, d_{KL}, -c_{MN})$
      $D_z = (-a_{IJ}, -b_{KL}, d_{MN})$
      $D_w = (-a_{IJ}, -b_{KL}, -c_{MN})$
      for $M' = 0$ to $\min(M - 1, 2n - 1 - I - K)$ **do**
        for $L' = 0$ to $\min(L - 1, 2n - 1 - J - N)$ **do**
          $i = I + K + M'$
          $j = L - L' - 1$
          $k = M - M' - 1$
          $l = J + L' + N$
          $\mathbb{M}_x(i, j, k, l) = \mathbb{M}_x(i, j, k, l) + D_x$
          $\mathbb{M}_y(i, j, k, l) = \mathbb{M}_y(i, j, k, l) + D_y$
          $\mathbb{M}_z(i, j, k, l) = \mathbb{M}_z(i, j, k, l) + D_z$
          $\mathbb{M}_w(i, j, k, l) = \mathbb{M}_w(i, j, k, l) + D_w$
        end for
      end for
    end for
  end for
  return $\mathbb{M}_x, \mathbb{M}_y, \mathbb{M}_z, \mathbb{M}_w$

In matrix form, this can be written as,

$$\begin{bmatrix} B_0^{\nu_1+d_1}(u)B_0^{\nu_2+d_2}(v) & \dots & B_{\nu_1+d_1}^{\nu_1+d_1}(u)B_{\nu_2+d_2}^{\nu_2+d_2}(v) \end{bmatrix} \mathbb{S}_\nu = \\ \begin{bmatrix} B_0^{\nu_1}(u)B_0^{\nu_2}(v)f_0 & \dots & B_{\nu_1}^{\nu_1}(u)B_{\nu_2}^{\nu_2}(v)f_0 & \dots & B_{\nu_1}^{\nu_1}(u)B_{\nu_2}^{\nu_2}(v)f_3 \end{bmatrix} \tag{4.13}$$

where, $\mathbb{S}_\nu$is a matrix constructed using the following relation between Bernstein polynomials,

$$B_i^{d_1}(u)B_j^{d_2}(v)B_k^{\nu_1}(u)B_l^{\nu_2}(v) = \frac{\binom{d_1}{i}\binom{d_2}{j}\binom{\nu_1}{k}\binom{\nu_2}{l}}{\binom{d_1+\nu_1}{i+k}\binom{d_2+\nu_2}{j+l}} B_{i+k}^{d_1+\nu_1}(u)B_{j+l}^{d_2+\nu_2}(v) \tag{4.14}$$

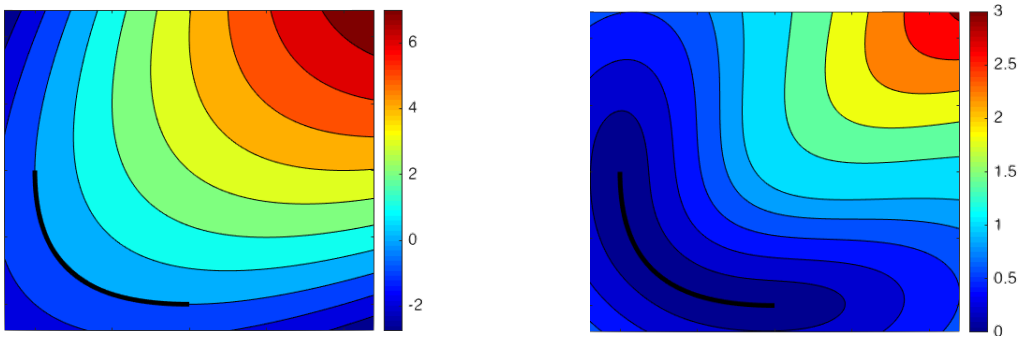
Thus, solutions for $\boldsymbol{\alpha}_{ij}$ in Eqs. (4.11) and (4.12) lie in the null space of $\mathbb{S}_\nu$. Since $\mathbb{S}_\nu$ may not have full row rank, a singular value decomposition is required to compute the null space $\texttt{Null}(\mathbb{S}_\nu)$. Writing a basis for this null space as $\texttt{Null}(\mathbb{S}_\nu) = \begin{bmatrix} \mathbb{M}_0 & \mathbb{M}_1 & \mathbb{M}_2 & \mathbb{M}_3 \end{bmatrix}^{\mathrm{T}}$, the Matrix Representation (M-Rep), $\boldsymbol{M_R}$, of the surface is obtained as,

$$\boldsymbol{M_R}(\boldsymbol{x}) = \mathbb{M}_0 + \mathbb{M}_1 x + \mathbb{M}_2 y + \mathbb{M}_3 z \tag{4.15}$$

where, $\mathbb{M}_i$ are matrices of real constants. For points P on the curve or surface, the M-Rep matrix drops rank since for some u, v, it satisfies,

$$\begin{bmatrix} B_0^{\nu_1}(u)B_0^{\nu_2}(v) & \dots & B_{\nu_1}^{\nu_1}(u)B_{\nu_2}^{\nu_2}(v) & \dots & B_{\nu_1}^{\nu_1}(u)B_{\nu_2}^{\nu_2}(v) \end{bmatrix} \boldsymbol{M_R}(P) = 0 \tag{4.16}$$

A measure of unsigned distance from the curve or surface may be obtained from $\boldsymbol{M_R}(\boldsymbol{x})$ as the product of its singular values, or as $\det\left(\boldsymbol{M_R}^{\mathrm{T}}\boldsymbol{M_R}\right)$. The determinant cannot be used directly since $\boldsymbol{M_R}$ may not be a square matrix.



(a) (b)

Figure 4.1. : Algebraic level sets for a section of a parabola generated from (a) Dixon resultant and (b) after the trimming operation.

4.2 Unsigned Algebraic Level Sets

For each Bézier segment of the spline geometry, the Dixon resultant derived in § 4.1.1 allows us to generate the algebraic level sets [84],

$$\Gamma(\boldsymbol{x}) = |\boldsymbol{M_D}(\boldsymbol{x})| \tag{4.17}$$

Composition of these level sets using the R-conjunction operation [126], corresponding to Boolean intersection, provides a smooth, composite algebraic level set for the complete spline geometry.

The generated level sets for a section of a parabola are shown in Fig. 4.1a. It can be seen that while the parametric curve is restricted to just a section, the resultant generates level sets over the entire parametric range, i.e., for the entire parabola. As a result, as we move away from any end of the curve in the tangential direction, the level sets are not monotonic. This is because the implicit equation for a section of a parabola is the same as that for the whole parabola. It is hence required that the implicitization is restricted to the required parametric domain. This is achieved using a trimming procedure based on R-functions [83, 126].

4.2.1 Trimming Procedure

The convex hull of the parametric surface, defined by the field $\Phi(\boldsymbol{x}) \geq 0$, is used as the trimming region. If $f(\boldsymbol{x})$ is the implicitization of the parametric surface, then the trimmed field, $g(\boldsymbol{x})$, is given by the R-function [127],

$$g(\boldsymbol{x}) = \sqrt{f^2 + \frac{\left(|\Phi| - \Phi\right)^2}{4}} \tag{4.18}$$

Within the trimming region, the original implicitization is recovered, while outside the region a composite field is obtained,

$$g(\boldsymbol{x}) = \begin{cases} |f(\boldsymbol{x})| & \Phi(\boldsymbol{x}) \geq 0 \\ \sqrt{f^2 + \Phi^2} & \Phi(\boldsymbol{x}) < 0 \end{cases} \tag{4.19}$$

Usage of the R-function ensures that the subsequent field is smooth. The level set Φ for the convex hull can be computed as a Boolean union of distance fields of its faces (edges in the case of planar curves). The procedure to obtain the Boolean union is described in § 4.2.2 as the R-disjunction operation. Trimmed level sets generated for the parabola section are shown in Fig. 4.1b; it can be seen that the level sets are globally monotonically increasing.

4.2.2 Boolean Operations Using R-Functions

Parametric spline geometries are piecewise polynomial representations. Consequently, an algebraic distance field measure for splines can be obtained as the Boolean intersection of the level sets of individual polynomial geometries. These Boolean operations are carried out algebraically using R-functions given in [126] to ensure smoothness. The three fundamental R-function Boolean operations are,

1. **Negation**: A positive argument gives a negative result and vice-versa

$$\neg g = -g \tag{4.20}$$

2. **Union**: R-disjunction provides the equivalent to Boolean union, where the result is positive if either argument is positive

$$g_1 \vee g_2 = g_1 + g_2 + \sqrt{g_1^2 + g_2^2} \tag{4.21}$$

3. **Intersection**: R-conjunction provides the functional equivalent, where the result is positive only if both arguments are positive

$$g_1 \wedge g_2 = g_1 + g_2 - \sqrt{g_1^2 + g_2^2} \tag{4.22}$$

Usage of the R-conjunction function for the Boolean intersection of individual level sets ensures a smooth composite field for parametric splines.

4.2.3 Normalization of the Algebraic Level Sets

In order to compose algebraic level sets from different segments of a parametric spline geometry, each constituent field has to be normalized. The level sets generated by the implicitization process in § 4.1.1 do not inherently satisfy the property $\|\nabla d\| = 1$ on the geometry, and hence different segments of the spline could grow at different rates. This could lead to a non-monotonic composite field as shown in Fig. 4.2a. Hence the algebraic level set in § 4.2.1 is normalized before trimming and composition. Consider a point $\boldsymbol{x}$, a distance d away from the geometry. Let $\boldsymbol{x}_f$ be the projection of the point on to the geometry. The Taylor series expansion of the resultant at $\boldsymbol{x}$ is,

$$\Gamma(\boldsymbol{x}_f) = \Gamma(\boldsymbol{x}) - \nabla\Gamma(\boldsymbol{x}) \cdot \boldsymbol{n}d + \frac{d^2}{2}\boldsymbol{n} \cdot \nabla\nabla\Gamma \cdot \boldsymbol{n} + \dots \tag{4.23}$$

where, $\boldsymbol{n}$ is the unit normal at $\boldsymbol{x}_f$. Since $\boldsymbol{x}_f$ lies on the curve or surface, $\Gamma(\boldsymbol{x}_f) = 0$. Ignoring higher order terms, we get the first order approximation for the distance,

$$d(\boldsymbol{x}) = \frac{\Gamma(\boldsymbol{x})}{\|\nabla\Gamma\|} \tag{4.24}$$

The distance measure defined by Eq. (4.24) satisfies the property $\|\nabla d\| = 1$ trivially on the geometry, while providing a first order accuracy for the distance. Additionally, the measure is exactly zero on the curve or surface. Normalizing individual distance measures before composition ensures monotonicity of the composite field of the spline, as shown in Fig. 4.2b. Higher order normalizations can also be obtained in a similar manner.

The gradient of the resultant is obtained using Eq. (4.17). From tensorial calculus, the derivative of the determinant of a matrix $\boldsymbol{M}$ is given by,

$$\nabla|\boldsymbol{M}| = |\boldsymbol{M}|\ tr\left(\boldsymbol{M}^{-1}\nabla\boldsymbol{M}\right) \tag{4.25}$$

where, $tr(\bullet)$ denotes the trace of a matrix. The gradient of the resultant is hence given by,

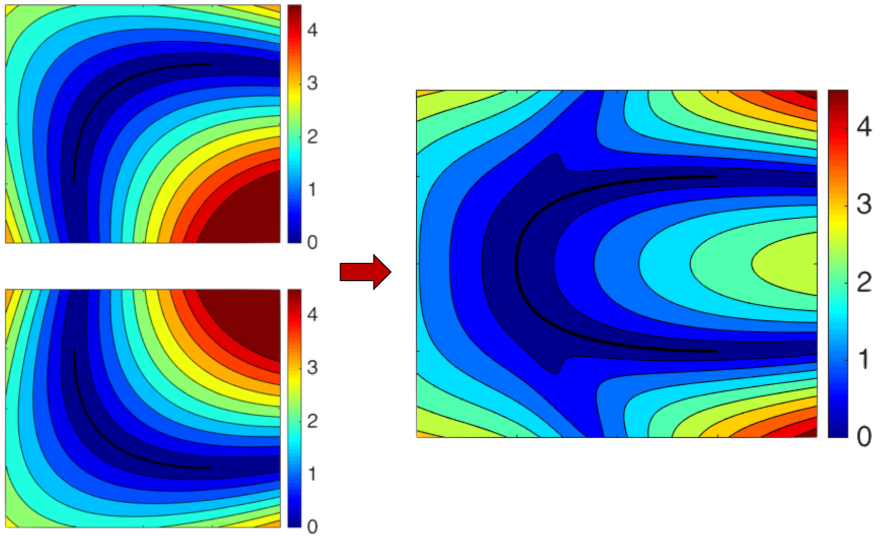
$$\nabla\Gamma = \Gamma\ tr\left(\boldsymbol{M}_{\boldsymbol{D}}^{-1}\nabla\boldsymbol{M}_{\boldsymbol{D}}\right) \tag{4.26}$$

The gradient of the Dixon matrix can be calculated from Eq. (4.7) as,

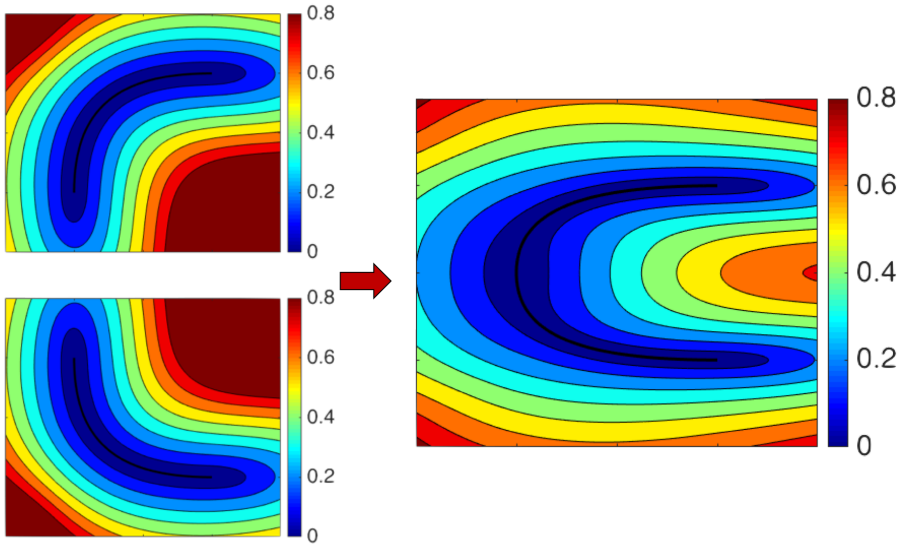
$$\nabla\boldsymbol{M}_{\boldsymbol{D}} = \begin{bmatrix} \mathbb{M}_x \\ \mathbb{M}_y \\ \mathbb{M}_z \end{bmatrix} \tag{4.27}$$

4.2.4 Construction of Unsigned Algebraic Level Sets

The procedure to construct algebraic level sets for parametric spline geometries is now summarized. Splines are piecewise polynomial representations. Common para-



(a)



(b)

Figure 4.2. : Algebraic level sets for a spline obtained from R-function based composition of individual distance fields. (a) Composition of non-normalized fields results in a non-monotonic distance measure (b) Normalization of individual level sets before composition ensures monotonicity of the resultant field

metric representations like composite Bézier and NURBS can be decomposed into individual Bézier components through knot insertion [80, 117, 118] and consequently, through basis conversion to the power basis, into the polynomial constituents.

1. The Dixon resultant Γ is computed for each individual component using Algm. 1
2. To facilitate composition with adjacent algebraic level sets, the individual resultants are normalized using Eq. (4.24)
3. The normalized algebraic level set is restricted to the defined parametric domain by using Eq. (4.18) with the convex hull as the trimming region
4. R-conjunction (Eq. (4.22)) is used to compose individual trimmed level sets to form algebraic level sets for the parametric spline

The algebraic level sets thus obtained are unsigned due to the trimming procedure. Both the trimming procedure and R-conjunction preserve the normalization [84].

4.3 Signed Alegebraic Level Sets

An algebraic procedure to generate level sets was discussed in the previous section § 4.2. Due to the trimming and R-conjunction, the level sets lack sign and are always positive. Open curves and surfaces do not partition space into inside and outside regions, and hence the concept of signed level sets does not apply to them. Signed algebraic level sets can hence be generated only for closed geometries. Attributing sign to these level sets helps classify points as lying inside or outside the geometry.

In [85], the sign of the algebraic level sets is computed separately from the distance computation, through a point by point containment query. The sign convention used here is that of positive distance inside the geometry and negative distance outside. On the curve or surface, the distance is zero, as seen in the previous section § 4.2. A close-fitted bounding polygon is constructed for the closed spline geometry, from the convex hulls of individual Bézier components. For each Bézier component, the

sign of the Dixon resultant Γ is set such that the resultant is negative for control points that lie on the bounding polygon (and hence outside the geometry). This is a one-time process for a given geometry. During sign determination, the point of interest is first classified with respect to the bounding box. If the point is outside the bounding box, then it is also outside the given geometry and its distance can be taken to be negative. Query points that lie inside the bounding box are then classified with respect to the convex hulls of the Bézier components. If the point lies inside any of the hulls, then the sign of the distance is the same as the sign of the Dixon resultant of the corresponding Bézier component, evaluated at the point of interest. If the query point does not lie inside any of the individual convex hulls, but lies inside the bounding box, then it lies inside the closed geometry and its distance can be taken to be positive.

The only non-trivial step in the point containment approach discussed is the construction of the bounding polygon from individual convex hulls; this is described presently.

4.3.1 Bounding Box Construction

A close-fitted bounding polygon can be constructed for a closed spline geometry from the convex hulls of its Bézier components. However the construction approach is different for planar curves and three-dimensional surfaces, and both are discussed separately here.

Bounding Box for Planar Curves

A convention is followed wherein the control points of the parametric spline are ordered in an anti-clockwise sense. Consider the convex hull of any Bézier component with vertices $v_1, v_2, \cdots, v_n$, where v_1 and v_n are the end points of Bézier. Due to the anti-clockwise ordering of the control points, the interior of the spline is to the left of the line segment v_1v_n. Hence all vertices of the hull to the left of this line segment are

discarded, while the rest contribute to the bounding box. This is illustrated in Fig. 4.3. Repeating for all Bézier components gives the required bounding box. An example is provided in Fig. 4.4 where the convex hulls of individual Bézier components are shown dashed. The bounding box constructed using the above procedure is shown with solid lines.

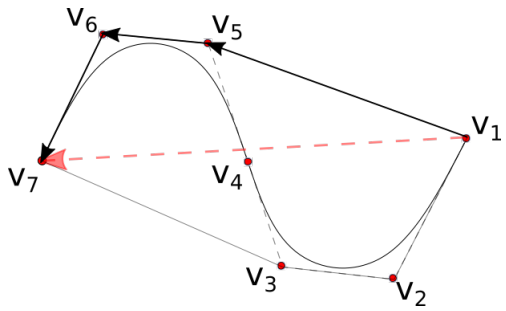


Figure 4.3. : Contribution to the bounding box from convex hull of a Bézier component. Only vertices to the right of the v_1v_7 line contribute to the bounding box. Edges of the convex hull contributing to the bounding box are shown as arrows, while the remaining edges are shown in grey.

In some cases the bounding box constructed could contain self-intersections, one such example shown in Fig. 4.5. Such a bounding polygon does not have clearly defined inside and outside regions, making point classification and subsequent sign determination infeasible. However a corner cutting procedure can be employed based

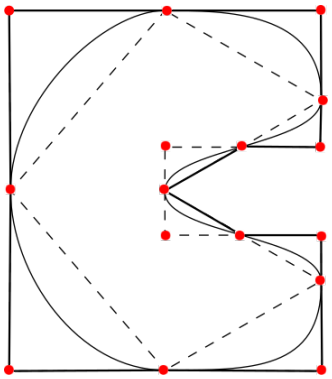


Figure 4.4. : Illustration of construction of a bounding polygon for a NURBS curve from the convex hulls of individual Bézier components. Edges contributing to the the bounding box are shown solid, while the remaining edges are shown dashed.

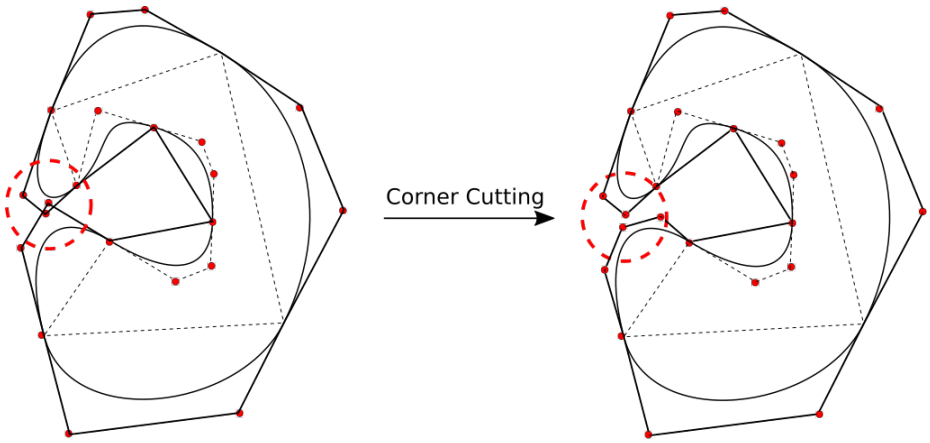


Figure 4.5. : Corner cutting to handle self-intersections in the bounding box.

on degree elevation of the intersecting Bézier components [85]. One of the intersecting Bézier components is chosen. Let the control points of this curve be $\boldsymbol{P}_0, \boldsymbol{P}_1, \cdots, \boldsymbol{P}_p$, where p is the degree of the Bézier curve. The curve can be equivalently represented as a $p+1$ degree Bézier curve, with $p+2$ control points given by [80],

$$\boldsymbol{Q}_i = \left(1 - \frac{i}{p+1}\right)\boldsymbol{P}_i + \frac{i}{p+1}\boldsymbol{P}_{i-1} \quad \forall\ i = 0, 1, \cdots, p+1 \tag{4.28}$$

Due to the nature of Bézier curves, as the degree increases, the convex hull approaches the curve. Hence the degree elevation procedure can be repeated until the bounding box has no self-intersections. This is illustrated in Fig. 4.5, where the bounding box has self-intersections. The intersecting cubic Bézier curve with 4 control points is degree elevated to a quartic curve with 5 control points, *without altering the original curve*. The bounding box no longer has self-intersections.

Bounding Box for Three-Dimensional Surfaces

The control points of the Bézier components of the spline surface (NURBS or composite Bézier) are assumed to be ordered such that the normal to the patch is pointed outwards to the enclosed volume. The vertices of the faces of their convex hulls are also ordered in an anti-clockwise sense so that the face normal points away from the Bézier patch. Consider the convex hull of the i^{th} Bézier component. Let f_k^i denote the k^{th} face of this convex hull and let the face normal be $\boldsymbol{n}_k^i$. Compute the centroid of f_k^i and project it on to the Bézier component under consideration. Let $\boldsymbol{N}_k^i$ be the normal to the Bézier at the projection point. If both $\boldsymbol{n}_k^i$ and $\boldsymbol{N}_k^i$ point in the same direction, i.e., $\boldsymbol{n}_k^i \cdot \boldsymbol{N}_k^i > 0$, then the face contributes to the bounding box. This process is then repeated for each face of the convex hulls of each Bézier component. An illustration is provided in Fig. 4.6. Self-intersections in the bounding box are handled similar to the case of 2D curves; one of the intersecting patches undergoes degree elevation until there are no more self-intersections. In case of overlap between

faces, the overlapping faces are replaced by their union. The process is detailed in Algm. 2

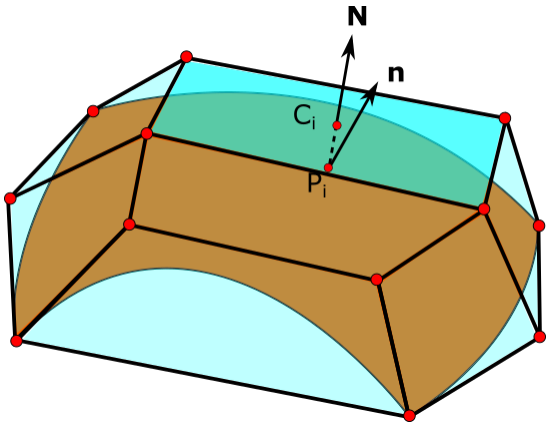


Figure 4.6. : Construction of the bounding box for a parametric surface from convex hull faces of Bézier components. C_i is the centroid of the face under interest (highlighted), and P_i is the projection of the centroid on to the Bézier surface. Since the face normal ($\boldsymbol{N}$) and surface normal ($\boldsymbol{n}$) point in the same direction, the face is added to the bounding box.

Algorithm 2 Bounding Box Construction for Spline Surfaces

Require: Bézier decomposition of spline surface
  $B \leftarrow$ empty polyhedron
  for all Bézier patch S^i **do**
    $H^i \leftarrow$ convex hull of S^i
    for all face f_k^i of H^i **do**
      $\boldsymbol{n}_k^i \leftarrow$ face normal of f_k^i
      $C_k^i \leftarrow$ centroid of f_k^i
      $P_k^i \leftarrow$ projection of C_k^i on to S^i
      $\boldsymbol{N}_k^i \leftarrow$ normal to S^i at P_k^i
      if $\boldsymbol{n}_k^i \cdot \boldsymbol{N}_k^i > 0$ **then**
        $B \leftarrow B \cup \{f_k^i\}$
      end if
    end for
  end for
  while B contains self-intersections **do**
    degree elevation of intersecting patches
  end while
  return B

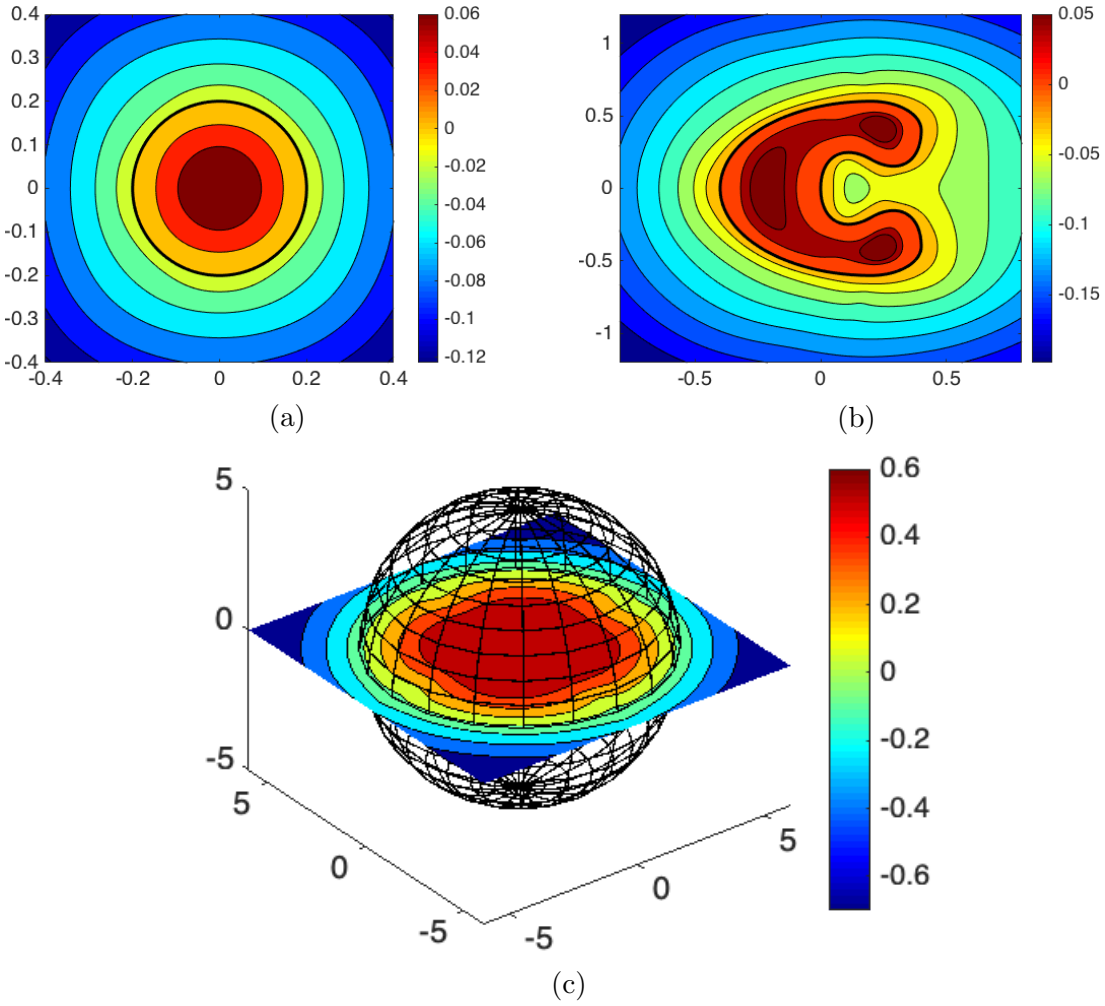


Figure 4.7. : Examples of 2D and 3D signed algebraic level sets: (a) Circle (b) Arbitrary bean-like shape (c) Sphere

4.3.2 Examples

Some examples of signed algebraic level sets, for both planar curves and three-dimensional surfaces are shown in Fig. 4.7. All three examples have been generated using quadratic NURBS representation. It can be observed that the generated algebraic level sets are exactly zero on the boundary. Also, they increase in magnitude as we move away from the boundary. Finally, as per the convention followed, the level sets are positive inside the closed geometries and negative outside.

4.4 Point Projection

Point projection and inversion are required to determine which control points of the enrichment influence behavior at a given point in the underlying domain. A smooth algebraic point projection procedure proposed in [87, 89] is used in this paper and shall now be described in brief. The foot point $\boldsymbol{x}_f$ of a point $\boldsymbol{x}$ in space is given by,

$$\boldsymbol{x}_f = \boldsymbol{x} - d\boldsymbol{n} \tag{4.29}$$

where, $d = \frac{\Gamma}{\|\nabla\Gamma\|}$ is a first order distance estimate obtained from a first order Taylor series expansion of the Dixon resultant $\Gamma(\boldsymbol{x})$. A more accurate point projection may be obtained by considering a second order Taylor series expansion,

$$\Gamma(\boldsymbol{x}_f = \boldsymbol{x} - d\boldsymbol{n}) = \Gamma(\boldsymbol{x}) - \nabla\Gamma \cdot \boldsymbol{n}d + \frac{d^2}{2}\boldsymbol{n} \cdot \nabla\nabla\Gamma\boldsymbol{n} = 0 \tag{4.30}$$

$$d = \frac{1}{\Gamma_{,nn}}\left(\Gamma_{,n} - \sqrt{\Gamma_{,n}^2 - 2\Gamma\ \Gamma_{,nn}}\right) \tag{4.31}$$

where, $(\bullet)_{,n}$ and $(\bullet)_{,nn}$ denote $\nabla(\bullet)\cdot\boldsymbol{n}$ and $\boldsymbol{n}\cdot\nabla\nabla(\bullet)\boldsymbol{n}$, respectively. The foot points obtained are in the physical space and have to be converted to parametric coordinates on the surface. This point inversion procedure can be carried out using Eq. (4.5). For

foot points $\boldsymbol{x}_f$ on the curve or surface, the Dixon matrix loses rank and satisfies this equation. Partitioning the Dixon matrix gives,

$$\begin{bmatrix} \boldsymbol{m} & \boldsymbol{M}_{\boldsymbol{D}}^{c} \end{bmatrix} \begin{bmatrix} 1 & u & u^2 & \dots & u^{2m-1}v^{2n-1} \end{bmatrix}^{\mathrm{T}} = 0 \tag{4.32}$$

$$\Rightarrow \boldsymbol{M}_{\boldsymbol{D}}^{c} \begin{bmatrix} u & u^2 & \dots & u^{2m-1}v^{2n-1} \end{bmatrix}^{\mathrm{T}} = -\boldsymbol{m} \tag{4.33}$$

where, $\boldsymbol{m}$ is a column vector and $\boldsymbol{M}_{\boldsymbol{D}}^{c}$is a matrix with appropriate dimensions. Now, $\boldsymbol{M}_{\boldsymbol{D}}^{c}$is of full rank and thus Eq. (4.33) can be solved to obtained the required parametric coordinates u, v.

4.5 Issues with Signed Algebraic Level Sets

As seen in § 4.1.1, it is only a necessary condition, not necessarily sufficient, that $\Gamma = |\boldsymbol{M}_{\boldsymbol{D}}| = 0$ on the geometry. It has been observed that for many three-dimensional parametric solids (including spheroids and cylinders), the Dixon resultant is identically zero everywhere [128, 129]. For such geometries, the parametric form cannot be implicitized by the method suggested, causing the level set generation to fail. A second issue is that the resultant may not be signed. The signed algebraic level set procedure uses the sign of the resultant to ascribe the sign of level sets for points close to the surface (within the convex hull). While the Dixon resultant generally switches from positive to negative across the boundary, there are some parametric solids where the resultant is either non-negative or non-positive everywhere. This causes the generation of signed algebraic level sets based on the resultant to fail, and consequently, the algebraic point projection procedure detailed in § 4.4. These issues shall be discussed separately in the following chapter.

5. EXTENDING SIGNED ALGEBRAIC LEVEL SETS TO THREE-DIMENSIONAL SURFACES

While the generation of algebraic level sets for curves using the Dixon resultant is robust, there are two issues in generating these level sets for surfaces. The first issue is the insufficiency of the Dixon resultant. It can be observed that for several three-dimensional parametric solids (including spheres and cylinders), the Dixon resultant is identically singular [128, 129]. Such geometries cannot directly be impliticized using the Dixon resultant and thus cannot generate algebraic level sets. The second issue with using the Dixon resultant is the occasional lack of sign for closed surfaces. The sign of the Dixon resultant is used to ascribe the sign of algebraic level sets for points close to the surface. For most surfaces the Dixon resultant is signed, classifying space into inside and outside regions. However, for some geometries (described later in Fig. 5.3), the Dixon resultant could lack an intrinsic sign. Subsequently, the generated algebraic level sets are unsigned and thus cannot classify points as lying inside or outside the closed geometry. In this chapter, procedures are developed to address wide rank-deficiency and loss of sign of the Dixon resultant. The maximal-rank submatrix approach developed in [128, 129] is used to correct for wide rank-deficiency of the resultant. A multivariate polynomial square root method is developed to recover the sign for unsigned resultants. Rectification measures for algebraic point projection developed in [87] are also proposed for surfaces with identically singular or unsigned Dixon resultants. As an application, the electrostatic problem is solved on a three-dimensional system with voids using Enriched Isogeometric Analysis.

5.1 Insufficiency of the Dixon Resultant

Consider a cylinder and a sphere, both of unit radius centered at the origin. As shown in Fig. 5.1, for both geometries, the resultant is zero everywhere. We do not have the natural generation of level sets required for the algebraic level sets. A similar issue was reported in [128] in the context of elimination theory, where the authors studied multivariate systems of equations. The Dixon resultant, used to detect the existence of a common solution to a multivariate system, was found to be identically zero for some systems. The authors attributed the cause to the presence of extraneous factors in the resultant. They also compared the Dixon resultant with other resultants such as the Sylvester resultant and concluded that the Dixon resultant generally had the least number of extraneous factors [129]. To recover the resultant, the authors suggested using the maximal-rank submatrix of the Dixon matrix $\boldsymbol{M_D}$ in Eq. (4.17). The same approach is adopted here in the context of implicitization of a parametric geometry.

A point that does not lie on the surface is chosen (e.g. one of the control points) and the Dixon matrix $\boldsymbol{M_D}$ is computed at this point. Since the resultant is zero everywhere, the Dixon matrix must be singular. The LU decomposition of the matrix with complete pivoting is performed. Complete pivoting is required since it is applicable even for singular matrices and is rank-revealing.

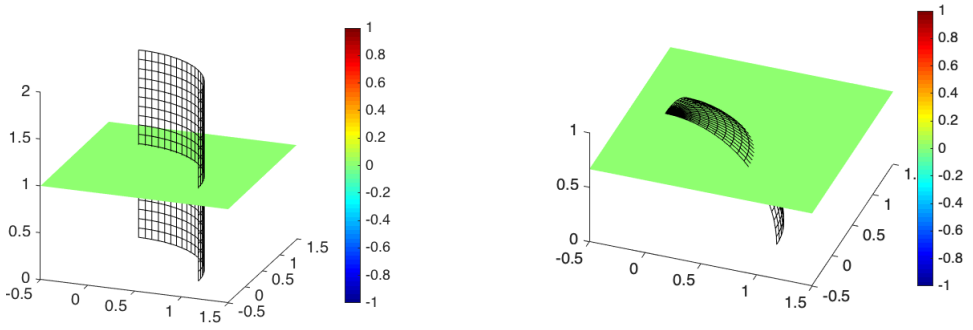
$$\boldsymbol{P}\boldsymbol{M_D}\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{U} = \boldsymbol{L}\begin{bmatrix}\tilde{\boldsymbol{U}}_{p\times 2mn}\\ \boldsymbol{0}\end{bmatrix} \tag{5.1}$$

where $\boldsymbol{P}$ and $\boldsymbol{Q}$ are permutation matrices, $\boldsymbol{L}$ is lower triangular, and $\boldsymbol{U}$ and $\tilde{\boldsymbol{U}}$ are upper triangular matrices. For identically singular Dixon matrices, the upper triangular matrix has rows of zeros as shown. The number of non-zero rows $p \leq 2mn$ in $\boldsymbol{U}$ is the row rank of the Dixon matrix. The maximal-rank submatrix is obtained by taking the first p rows and first p columns from the permuted Dixon matrix. The permutation matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ can be represented as an ordering of row indices

p_i and column indices q_j, respectively. Then, the maximal-rank submatrix may be denoted as,

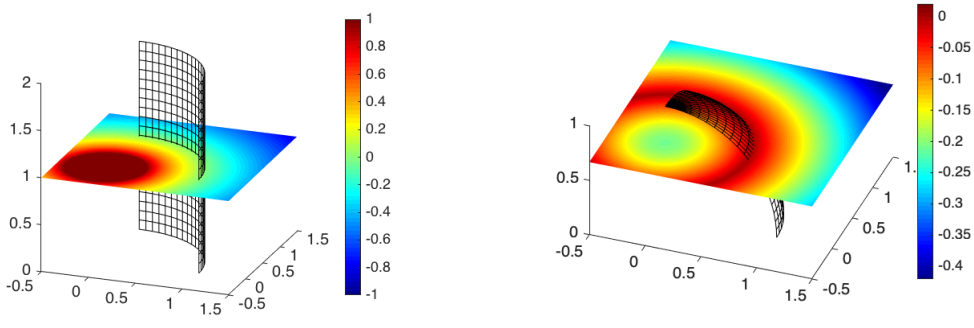
$$\boldsymbol{M}_{\boldsymbol{D}}^{sub} = \boldsymbol{M}_{\boldsymbol{D}}[p_1 \dots p_p;\ q_1 \dots q_p] \tag{5.2}$$

This submatrix now behaves like a resultant. This operation is only performed once for the geometry; the corresponding indices are stored and used in distance computation for all points. For points on the surface this submatrix becomes singular, and the corresponding resultant is zero.



(a) (b)

Figure 5.1. : Dixon resultant contours for (a) cylinder of unit radius centered at the origin (b) sphere of unit radius centered at the origin. The Dixon matrix is identically singular and hence the resultant is zero everywhere for both geometries.



(a) (b)

Figure 5.2. : Using the submatrix recovers the implicitization of the (a) cylinder and (b) sphere, thereby generating algebraic level sets.

Example 1. An octant of a unit sphere centered at origin, shown in Fig. 5.1b, is now considered. A random point $(2.00, 1.57, 1.97)$, not lying on the sphere, is chosen. One of the control points, excluding the end points, may also be chosen. At the selected point the Dixon matrix has the form,

$$\boldsymbol{M_D} = \begin{bmatrix} -2.00 & -1.63 & -1.03 & 2.25 & 2.11 & -0.08 & 0.91 & -0.54 \\ -2.20 & 1.29 & 5.74 & -1.14 & -1.70 & -0.31 & -1.85 & 0.16 \\ -0.46 & 5.08 & 2.85 & -6.57 & -1.32 & 0.00 & -1.07 & 1.49 \\ 2.58 & -3.59 & -4.51 & 2.75 & 0.69 & 1.11 & 1.24 & -0.27 \\ -0.21 & -3.37 & -1.92 & 4.39 & 1.29 & -0.02 & 0.85 & -1.00 \\ -2.04 & 2.42 & 4.03 & -1.89 & -0.82 & -0.73 & -1.17 & 0.20 \\ -0.96 & 2.42 & 1.32 & -3.08 & -0.05 & -0.03 & -0.31 & 0.69 \\ 0.76 & -1.66 & -0.67 & 1.22 & -0.18 & 0.55 & 0.09 & -0.10 \end{bmatrix} \tag{5.3}$$

This matrix is singular even though the chosen point does not lie on the sphere. Performing LU decomposition with complete pivoting gives the upper triangular matrix,

$$\boldsymbol{U} = \begin{bmatrix} -6.57 & 2.85 & -0.46 & 5.08 & 0 & 1.49 & -1.32 & -1.07 \\ 0 & 5.25 & -2.12 & 0.41 & -0.31 & -0.1 & -1.47 & -1.66 \\ 0 & 0 & -2.17 & 0.11 & -0.08 & -0.03 & 1.65 & 0.53 \\ 0 & 0 & 0 & -1.16 & 0.88 & 0.28 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.4}$$

By observation, it is clear that the Dixon matrix is of rank 4. Using the permutation matrices of the complete pivoting, the maximal-rank submatrix of the Dixon matrix $\boldsymbol{M_D}$ is obtained as,

$$\boldsymbol{M_D^{sub}} = \begin{bmatrix} 2.85 & -6.57 & -0.46 & 5.08 \\ 5.74 & -1.14 & -2.2 & 1.29 \\ -1.03 & 2.25 & -2.0 & -1.63 \\ -4.51 & 2.75 & 2.58 & -3.59 \end{bmatrix} \tag{5.5}$$

This LU decomposition is performed only once. For subsequent resultant computations, the same indices are used for the submatrix. Figure 5.2 shows the recovery of the algebraic level sets on using the submatrix approach for a quarter-cylinder and an octant of a sphere.

5.1.1 Point Inversion

While the earlier described procedure for point projection continues to be valid for surfaces with identically singular Dixon matrices, point inversion to parametric coordinates is not valid due to rank loss exceeding unity. Unlike for general surfaces as described in § 4.4, Eq. (4.5) cannot be used for point inversion since the null space of the Dixon matrix has dimension greater than 1, thereby allowing multiple solutions for this equation. Instead, point inversion may be carried out using the M-Rep matrix equation Eq. (4.16) since $\boldsymbol{M_R^{\mathrm{T}}}$ has a null space dimension of 1 even for surfaces with identically singular Dixon matrices. The null space of the M-Rep matrix may be computed from the singular value decomposition (SVD) to determine the corresponding parametric coordinates [121]. Alternatively, here, a column deletion

procedure, akin to § 4.4, is used to avoid performing the computationally expensive SVD operation. Expanding the Bernstein polynomials in Eq. (4.16), we get,

$$\boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}} \begin{bmatrix} (1-u)^{\nu_1}(1-v)^{\nu_2} \\ \nu_1 u(1-u)^{\nu_1-1}(1-v)^{\nu_2} \\ \vdots \\ u^{\nu_1}v^{\nu_2} \end{bmatrix} = 0 \tag{5.6}$$

First, it is verified that $u = 1$ or $v = 1$ are not feasible. Now assuming $u \neq 1, v \neq 1$, we define new variables $t = \frac{u}{1-u}, s = \frac{v}{1-v}$ having a one-one correspondence with u, v respectively. For feasible parameters $0 \leq u, v \leq 1$, we have $s, t \geq 0$. Thus, Eq. (5.6) can be expressed in terms of s, t as,

$$\begin{bmatrix} \boldsymbol{m} & \boldsymbol{M}_{\boldsymbol{R}}^{c} \end{bmatrix} \begin{bmatrix} 1 & \nu_1 t & \dots & t^{\nu_1}s^{\nu_2} \end{bmatrix}^{\mathrm{T}} = 0 \tag{5.7}$$

$$\Rightarrow [\boldsymbol{M}_{\boldsymbol{R}}^{c}] \begin{bmatrix} \nu_1 t & \binom{\nu_1}{2} t^2 & \dots & t^{\nu_1}s^{\nu_2} \end{bmatrix}^{\mathrm{T}} = -\boldsymbol{m} \tag{5.8}$$

where $\boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}}$ has been partitioned into a column vector $\boldsymbol{m}$ and a matrix $\boldsymbol{M}_{\boldsymbol{R}}^{c}$. Since $\boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}}$ has a nullity of 1, $\boldsymbol{M}_{\boldsymbol{R}}^{c}$is of full-rank and thus, Eq. (5.8) may be solved directly to obtain s, t and subsequently $u = \frac{t}{1+t}, v = \frac{s}{1+s}$.

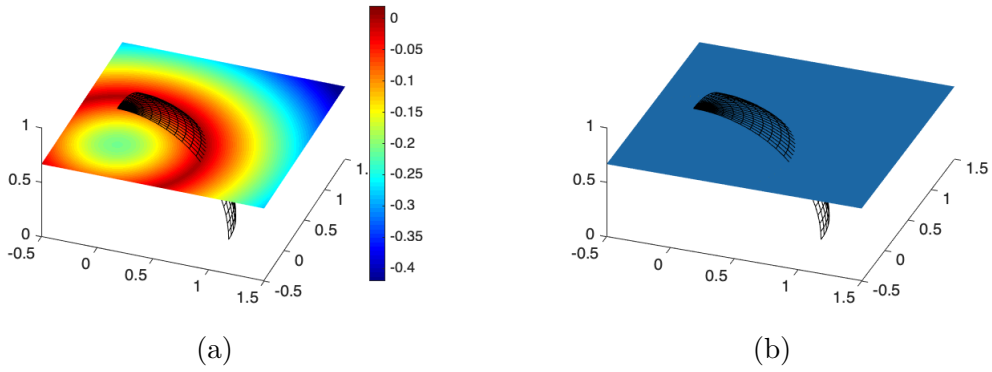
5.2 Lack of Signed Resultant

As described in § 4.3, for points inside the convex hull of a Bézier component, the sign of the algebraic level set is taken to be the same as that of the Dixon resultant. This relies on the fact that the implicit form of a parametric curve or surface generally switches from positive to negative across the boundary of the geometry. For example the implicit form of a unit circle, $(1 - x^2 - y^2)$, transitions from being positive inside the circle, zero on the circumference and negative outside. However, this need not be true in all cases. For the same example of a unit circle, an equally valid implicit form is $(1 - x^2 - y^2)^2$. This is zero on the circle and positive everywhere else. While still

capturing the geometry accurately, such implicitizations do not allow signed algebraic level sets and by extension, point containment determination.

The Dixon resultant being an implicit form is also not immune to being unsigned. Consider an octant of a unit sphere centered at the origin. The sphere is defined as a quadratic rational Bézier surface. The Dixon resultant was found to be trivially singular and hence the maximal-rank submatrix was used. The resultant and its sign are shown in Fig. 5.3. It can be seen that the resultant is zero on the sphere and negative everywhere else. In such cases, the resultant can be expressed as $\Gamma = -\rho^2$, where ρ is the desired resultant. This can be verified analytically for the unit sphere example, where the Dixon resultant can be expanded as,

$$\Gamma(\boldsymbol{x}) = -1.56(x^2 + y^2 + z^2 - 1)^2 \tag{5.9}$$



(a) (b)

Figure 5.3. : (a) Maximal-rank submatrix based resultant and its (b) sign, for an octant of a unit sphere centered at the origin. The resultant is negative (blue) everywhere.

5.2.1 Possible Strategies for Sign Extraction

A general procedure is hence required to extract the sign of the fundamental resultant for cases similar to Eq. (5.9). A couple of possible approaches to do the same are discussed in brief below.

Matrix Square Root

From Eq. (4.17) and writing $\Gamma = -\rho^2$, we can extract the fundamental resultant as,

$$\Gamma' = \left|\sqrt{\boldsymbol{M_D}}\right| = i\rho \tag{5.10}$$

Using Schur decomposition, one can compute the matrix square root of the Dixon matrix using the Bjorck-Hammarling recursion [130], whose determinant would be the required resultant. However the sign of the square root was found to be inconsistent (see Fig. 5.4), possibly depending on the nature of the eigenvalues of $\boldsymbol{M_D}$. If $e_1, e_2, \ldots, e_n$ are the eigenvalues of the Dixon matrix then,

$$\Gamma' = \sqrt{e_1} \times \sqrt{e_2} \times \cdots \times \sqrt{e_n} \tag{5.11}$$

The nature of Eq. (5.11) depends on whether the eigenvalues are all real or include complex conjugate pairs. For example, if the Dixon matrix has 3 eigenvalues, all real, with $\Gamma = -1$, then two possible eigenvalue triplets are $(1, 1, -1)$ and $(-1, -1, -1)$. The former gives $\Gamma' = i$ while the latter gives $\Gamma' = -i$. The square root hence allows both positive and negative values. However, if the eigenvalues contain a complex conjugate pair, then the third eigenvalue must be negative (e.g. $(-1, i, -i)$). In such cases the imaginary part of Γ' is always positive.

Normal Gradient Method

While two points on either side of the curve of surface can have the same sign, the gradients at these points would point in opposite directions, as shown in Fig. 5.5. To compute the sign of the resultant, the gradient $\nabla\Gamma$ is computed at the point of interest. The point is then projected on to the curve or surface and the normal $\boldsymbol{n}$ computed at the projection point. The sign of the resultant is then given by the sign of $\nabla\Gamma \cdot \boldsymbol{n}$.

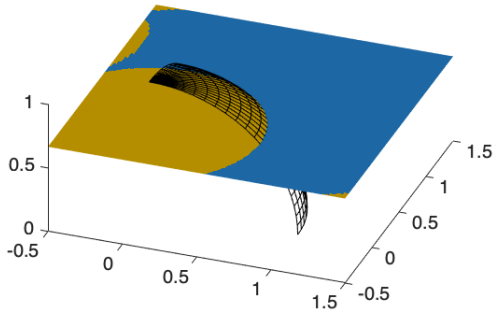


Figure 5.4. : Sign of the resultant obtained from the matrix square root method. Regions in yellowish brown denote positive distance and those in blue denote negative distance. The sign is observed to contain inconsistencies.

However, this method requires accurate point projection. Algebraic point projection used in [87, 89] is not exact and its accuracy is found to be wanting, especially for points far away from the geometry. Alternatively one can use iterative methods such as the Newton-Raphson method. Such methods lack robustness [84] and also lose the computational advantage of signed algebraic level sets.

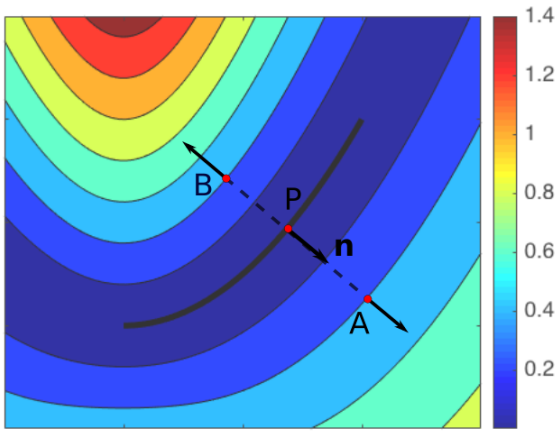


Figure 5.5. : Points A and B on opposite sides of the cubic curve both have positive resultants, but the gradient of the resultant (marked by arrows) at these points point in opposite directions. P is the projection of A and B on to the curve while **n** is the normal to the curve at P. Thus, A has a positive normal gradient and is hence outside, while B has a negative normal gradient and is hence inside.

5.2.2 Algebraic Square Root

A method is hence required to extract the sign when the Dixon resultant lacks sign. A robust, consistent and algebraic method is now developed to extract the fundamental resultant directly from the unsigned Dixon resultant. From the form of the Dixon matrix in Eq. (4.7), the resultant can be expressed as a polynomial in x, y, z. Using Eq. (4.17), we can write,

$$\Gamma = |\boldsymbol{M_D}| = \sum_i \sum_j \sum_k M_{ijk} x^i y^j z^k \tag{5.12}$$

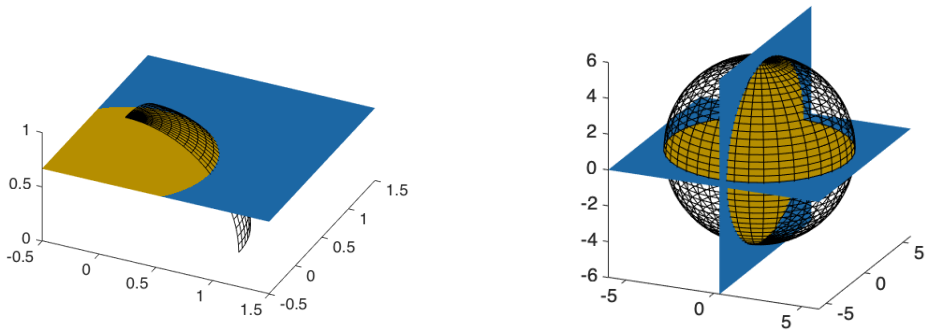
where, M_{ijk} is the sum of determinants of *all* matrices formed from any i rows of $\mathbb{M}_x$, j rows of $\mathbb{M}_y$, k rows of $\mathbb{M}_z$ and the rest from $\mathbb{M}_w$, as explained in detail below. The problem of determining the fundamental resultant ρ then reduces to determining the square root of a multivariate polynomial. The sign of the algebraic level set is then obtained from the sign of the fundamental resultant. Since ρ is obtained directly as the polynomial square root of the Dixon resultant, this procedure is required only once for the curve or surface. Once computed, the fundamental resultant can be used directly at all points. This is illustrated in Fig. 5.6 for the unit sphere example. It can be seen that this method is able to successfully classify regions as inside and outside. The problem can hence be divided into two steps,

1. Expansion of the Dixon resultant into the polynomail form
2. Obtaining the polynomial square root of the resultant

Both steps will now be discussed in brief.

Polynomial Expansion of the Resultant

The coefficients M_{ijk} in the polynomial expansion in Eq. (5.12) can be determined by decomposing the determinant into a sum of several determinants such that each



(a) (b)

Figure 5.6. : Sign of the algebraic level sets determined using the algebraic square root method. Positive level sets are coloured yellowish brown, while blue regions correspond to negative level sets. (a) Sign of resultant for an octant of a unit sphere (b) Sign of algebraic level sets for a complete sphere

determinant contains only monomials elements, with the variable being the same in each row. In other words, the resultant can be expressed in the form,

$$\Gamma = |\mathbb{M}_w + \mathbb{M}_x x + \mathbb{M}_y y + \mathbb{M}_z z| = \sum_{(i_1,i_2,\dots)} \begin{vmatrix} \boldsymbol{M}^1_{x_{i1}} x_{i1} \\ \boldsymbol{M}^2_{x_{i2}} x_{i2} \\ \vdots \end{vmatrix} = \sum_i \sum_j \sum_k M_{ijk} x^i y^j z^k \quad (5.13)$$

Here, $\boldsymbol{M}^p_{x_{ip}}$ denotes the p^{th} row of the coefficient matrix $\mathbb{M}_{x_{ip}}$. The subscript ip takes the values 0,1,2 and 3 denoting constant, x, y, z terms respectively. The summation is over all possible combinations of $(i1, i2, \dots)$, allowing repetition of variables. An illustration for a 2×2 determinant is provided in Ex. 2. Each determinant in Eq. (5.13) that consists of i rows from $\mathbb{M}_x$, j rows from $\mathbb{M}_y$, k rows from $\mathbb{M}_z$ and the rest from $\mathbb{M}_w$, evaluates into a $x^i y^j z^k$ term. Hence the cofficient M_{ijk} in Eq. (5.12) is simply the sum of *all* such determinants. This gives the polynomial expansion of the resultant, term by term.

Example 2. Consider the determinant,

$$\Delta = \begin{vmatrix} x+2 & y+2 \\ x+2y & x+y+1 \end{vmatrix} \quad (5.14)$$

This determinant can be split as,

$$\Delta = \begin{vmatrix} x & 0 \\ x & x \end{vmatrix} + \begin{vmatrix} x & 0 \\ 2y & y \end{vmatrix} + \begin{vmatrix} x & 0 \\ 0 & 1 \end{vmatrix} + \begin{vmatrix} 0 & y \\ x & x \end{vmatrix} + \begin{vmatrix} 0 & y \\ 2y & y \end{vmatrix} + \begin{vmatrix} 0 & y \\ 0 & 1 \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ x & x \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ 2y & y \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ 0 & 1 \end{vmatrix} \quad (5.15)$$

Each determinant in the decomposition only consists of monomials. Additionally each row has the same variable.

- The constant term in the expansion is the determinant with only constant elements, $M_{000} = \begin{vmatrix} 2 & 2 \\ 0 & 1 \end{vmatrix}$

- The term linear in x is the sum of determinants containing one row of x and one constant row, $M_{100} = \begin{vmatrix} x & 0 \\ 0 & 1 \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ x & x \end{vmatrix}$
- The term linear in y is the sum of determinants containing one row of y and one constant row, $M_{010} = \begin{vmatrix} 0 & y \\ 0 & 1 \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ 2y & y \end{vmatrix}$

Thus the polynomial expansion of Δ can be constructed term by term.

Computation of the Polynomial Square Root

Once the polynomial expansion of the resultant is known, the polynomial square root can be computed analytically for small-degree surfaces. For example, for a linear surface we have,

$$\begin{aligned} \Gamma = \rho^2 &= (ax + by + cz + d)^2 \\ &= a^2x^2 + b^2y^2 + c^2z^2 + 2(abxy + bcyz + acxz) + 2d(ax + by + cz) + d^2 \end{aligned} \tag{5.16}$$

For this case, the fundamental resultant can be extracted as,

$$\begin{aligned} d &= \sqrt{M_{000}} \\ a = \frac{M_{100}}{2d}; b &= \frac{M_{010}}{2d}; c = \frac{M_{001}}{2d} \end{aligned} \tag{5.17}$$

where, M_{ijk} denotes the coefficient of $x^i y^j z^k$ in the polynomial expansion of Γ. It may seem simpler to obtain a directly as $a = \sqrt{M_{200}}$. However, this does not extend naturally to higher degree surfaces. On the other hand, Eq. (5.17) gives the coefficients for the linear and constant terms for surfaces of any degree. This is because quadratic and higher order terms in ρ cannot contribute to the linear and constant terms in Γ. This suggests an approach to generalize the procedure for any surface.

Let the polynomial expansion of the fundamental resultant be,

$$\rho = \sum_i \sum_j \sum_k R_{ijk} x^i y^j z^k \tag{5.18}$$

Expanding Γ in terms of the fundamental resultant, we get,

$$\Gamma = \rho^2 = \sum_p \sum_q \sum_r M_{pqr} x^p y^q z^r \tag{5.19}$$

$$= \left(\sum_i \sum_j \sum_k R_{ijk} x^i y^j z^k \right) \left(\sum_l \sum_m \sum_n R_{lmn} x^l y^m z^n \right) \tag{5.20}$$

$$\Gamma = \sum_i \sum_j \sum_k \sum_l \sum_m \sum_n R_{ijk} R_{lmn} x^{i+l} y^{j+m} z^{k+n} \tag{5.21}$$

Now, higher order terms in ρ cannot contribute to the coefficient M_{pqr} in Γ. This gives, for $\{pqr\} \neq \{000\}$,

$$M_{pqr} = \sum_{i=0}^{p} \sum_{j=0}^{q} \sum_{k=0}^{r} R_{ijk} R_{p-i,q-j,r-k} \tag{5.22}$$

$$= 2R_{000} R_{pqr} + \sum_{\substack{(i,j,k)\neq(0,0,0) \\ (i,j,k)\neq(p,q,r)}} R_{ijk} R_{p-i,q-j,r-k} \tag{5.23}$$

$$\Rightarrow R_{ijk} = \frac{1}{2R_{000}} \left[M_{pqr} - \sum_{\substack{(i,j,k)\neq(0,0,0) \\ (i,j,k)\neq(p,q,r)}} R_{ijk} R_{p-i,q-j,r-k} \right] \tag{5.24}$$

This provides a recursive procedure to calculate the polynomial expansion of the fundamental resultant, described in Algm. 3. An illustration is provided in Ex. 3. This method is valid for surfaces of any degree, but Eq. (5.24) assumes that $R_{000} \neq 0$, i.e., the curve or surface does not contain the origin. It is easy to deal with curves or surfaces with $R_{000} = 0$, by choosing a coordinate system with a translated origin for the purposes of computing the resultant. This new coordinate system is used only for distance computations of the particular curve or surface, and is chosen such that its origin does not lie on the geometry of interest (say, a control point). Such an origin

shift is valid since the distance of a point from a curve or surface is invariant under translation of the origin. Once the fundamental resultant is known, it can be evaluated at the point of interest and its sign used in the algebraic level sets directly. This allows the generation of signed distance level sets completely algebraically, without resorting to iterative and non-robust Newton-Raphson method.

Algorithm 3 Computation of the Polynomial Square Root

Require: Polynomial coefficients M_{pqr} of the given polynomial
  $R_{000} = \sqrt{M_{000}}$
  $N \leftarrow$ degree of given polynomial
  for $p, q, r = 0$ to $N/2$ **do**
    if $((i, j, k) = (0, 0, 0))$ **or** $(i + j + k > N/2)$ **then**
      cycle
    end if
    $R_{pqr} \leftarrow M_{pqr}$
    for $i = 0$ to p **do**
      for $j = 0$ to q **do**
        for $k = 0$ to r **do**
          if $(i, j, k) = (0, 0, 0)$ **or** $(i, j, k) = (p, q, r)$ **then**
            cycle
          end if
          $R_{pqr} \leftarrow R_{pqr} - R_{ijk} R_{p-i,q-j,r-k}$
        end for
      end for
    end for
    $R_{pqr} \leftarrow R_{pqr}/(2R_{000})$
  end for
  return Polynomial coefficients R_{ijk} of the square root

Example 3. Consider the example of a unit sphere with the resultant given by,

$$\Gamma = x^4 + y^4 + z^4 + 2(x^2y^2 + y^2z^2 + x^2z^2) - 2(x^2 + y^2 + z^2) + 1 \tag{5.25}$$

The coefficients of the square root are then obtained as,

$$R_{000} = \sqrt{M_{000}} = \sqrt{1} = 1 \tag{5.26}$$

$$R_{001} = \frac{M_{001}}{2R_{000}} = 0; R_{010} = \frac{M_{010}}{2R_{000}} = 0; R_{100} = \frac{M_{100}}{2R_{000}} = 0 \tag{5.27}$$

$$R_{011} = \frac{1}{2R_{000}} \left[M_{011} - 2R_{010}R_{001}\right] = 0; R_{101} = R_{110} = 0 \tag{5.28}$$

$$R_{002} = \frac{1}{2R_{000}} \left[M_{002} - R_{001}^2\right] = -\frac{2}{2} = -1; R_{020} = R_{200} = -1 \tag{5.29}$$

This gives the signed polynomial square root,

$$\rho = \sqrt{\Gamma} = (1 - x^2 - y^2 - z^2) \tag{5.30}$$

The polynomial square root method is used to generate the algebraic level sets shown in Fig. 5.6, as well as for the sphere in Fig. 4.7c.

5.2.3 Point Projection and Inversion

For a resultant in the squared form $\Gamma = \pm\rho^2$, the degree is twice that of the fundamental resultant ρ. This leads to poor point projection using the Taylor series expansion described in § 4.4. In order to improve the accuracy of point projection, the Taylor series expansion in Eq. (4.30) is expressed in terms of the fundamental resultant. For first order algebraic point projection, the distance estimate used in Eq. (4.29) is modified as,

$$d = \frac{|\rho|}{\|\nabla\rho\|} = 2\frac{|\Gamma|}{\|\nabla\Gamma\|} \tag{5.31}$$

Thus, using the original squared resultant underestimates the distance of a point from the surface. For second order algebraic point projection, we rewrite Eq. (4.31) in terms of the fundamental resultant and express the result in terms of Γ,

$$\begin{aligned} d &= \frac{1}{\rho_{,nn}} \left(\rho_{,n} - \sqrt{\rho_{,n}^2 - 2\rho\ \rho_{,nn}} \right) \\ &= \frac{2\Gamma}{\left(2\Gamma\ \Gamma_{,nn} - \Gamma_{,n}^2\right)} \left(\Gamma_{,n} - \sqrt{3\Gamma_{,n}^2 - 4\Gamma\ \Gamma_{,nn}} \right) \end{aligned} \tag{5.32}$$

where $(\bullet)_{,n}$ and $(\bullet)_{,nn}$ denote $\frac{\partial \bullet}{\partial n}$ and $\frac{\partial^2 \bullet}{\partial n^2}$, respectively.

As a consequence of the squared form of the resultant, each point on the physical surface corresponds to two points in the parametric space. For example, in the octant example shown in Fig. 5.3, the point $(0.36, 0.48, 0.8)$ lies on the surface. In the parametric space, this point corresponds to the two points $(0.5858, 0.5858)$ and $(-16.4853, 1.5469)$. Out of these, only one point lies within the parametric range of interest $[0, 1]$. Thus, unlike general surfaces, there are two solutions to Eq. (5.6), and the M-Rep matrix drops rank by two for points on such a surface. As a result, surfaces with squared-form resultants require an alternative procedure for point inversion from physical to parametric space. Such a procedure is now developed.

Beginning with Eq. (5.6), it is first verified that $u = 1$ or $v = 1$ are not feasible. Now, assuming $u \neq 1, v \neq 1$, we define new variables $t = \frac{u}{1-u}, s = \frac{v}{1-v}$ having a one-one correspondence with u, v respectively. For feasible parameters $0 \leq u, v \leq 1$, we have $s, t \geq 0$. Expressing Eq. (5.6) in terms of the new variables and assuming the existence of two solutions $(t_1, s_1), (t_2, s_2)$, we have,

$$\boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}} \begin{bmatrix} 1 \\ \nu_1 t_1 \\ \binom{\nu_1}{2} t_1^2 \\ \vdots \\ t_1^{\nu_1} s_1^{\nu_2} \end{bmatrix} = \boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}} \begin{bmatrix} 1 \\ \nu_1 t_2 \\ \binom{\nu_1}{2} t_2^2 \\ \vdots \\ t_2^{\nu_1} s_2^{\nu_2} \end{bmatrix} = \boldsymbol{M}_{\boldsymbol{R}}^{\mathrm{T}} \begin{bmatrix} 0 \\ \nu_1 \Delta t \\ \binom{\nu_1}{2} \Delta t^2 \\ \vdots \\ \Delta \left(t^{\nu_1} s^{\nu_2} \right) \end{bmatrix} = 0 \tag{5.33}$$

where, $\Delta(\bullet) = (\bullet)_1 - (\bullet)_2$. Now, dividing by Δt in the new homogeneous equation and partitioning the coefficient matrix gives,

$$\begin{bmatrix} \boldsymbol{m}_1 & \boldsymbol{m}_2 & \boldsymbol{M}^c_{\boldsymbol{R}} \end{bmatrix} \begin{bmatrix} 0 & \nu_1 & \binom{\nu_1}{2}\Delta t^2/\Delta t & \dots & \Delta\left(t^{\nu_1} s^{\nu_2}\right)/\Delta t \end{bmatrix}^{\mathrm{T}} = 0 \tag{5.34}$$

$$\Rightarrow \left[\boldsymbol{M}^c_{\boldsymbol{R}}\right] \begin{bmatrix} \binom{\nu_1}{2}\Delta t^2/\Delta t & \dots & \Delta\left(t^{\nu_1} s^{\nu_2}\right)/\Delta t \end{bmatrix} = -\nu_1 \boldsymbol{m}_2 \tag{5.35}$$

where, $\boldsymbol{m}_1, \boldsymbol{m}_2$ are column vectors and $\boldsymbol{M}^c_{\boldsymbol{R}}$ is a matrix of appropriate dimensions. Since $\boldsymbol{M}^{\mathrm{T}}_{\boldsymbol{R}}$ has a nullity of 2, $\boldsymbol{M}^c_{\boldsymbol{R}}$ is of full rank and Eq. (5.35) can be solved directly to obtain (t_1, s_1) and (t_2, s_2). Taking only the positive (feasible) solution, the required parametric point can be obtained as $u = \frac{t}{1+t}, v = \frac{s}{1+s}$. The result of the procedure is illustrated on a sphere in Fig. 5.7.

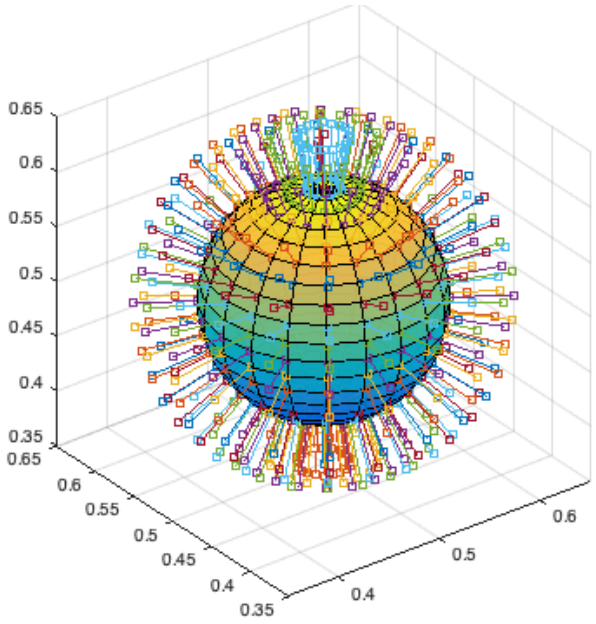


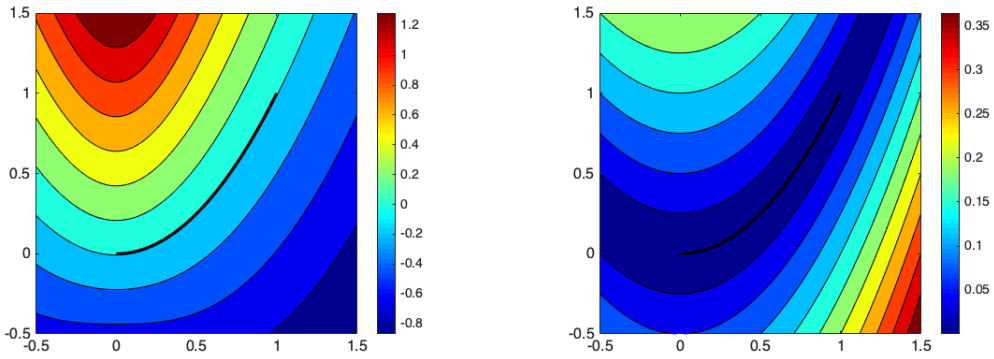
Figure 5.7. : Algebraic point projection on to a sphere of radius 0.1 units. The test points were sampled from a concentric sphere of radius 0.15 units.

5.2.4 Study of Computational Efficiency

Four different geometries (Figs. 5.3, 5.8 and 5.9 and Fig. 5.2a) were considered to study the computation time of algebraic level sets. For comparison, the computation times for the same geometries using the M-Rep method developed by Busé [121] was also calculated. This analysis has two parts. The first is a comparison of the pre-processing times. For algebraic level sets, pre-processing requires robustness checks for sufficiency and sign of the Dixon resultant. Depending on the geometry, this may require computing a maximal-rank submatrix and/or a polynomial square root. On the other hand, for the M-Rep method, pre-processing requires Singular-Value Decomposition (SVD) to compute the M-Rep matrix $\boldsymbol{M_R}$. The second part of the computational analysis is the estimation of distance at sample points. 10,000 points were chosen and the distance estimated at these points. In case of the M-Rep method, the product of singular values was chosen as the distance estimate. Busé [90] also suggests using the determinant of $\boldsymbol{M_R}\boldsymbol{M_R}^{\mathrm{T}}$ as an alternate distance measure. This was avoided in this paper as matrix products can lead to round-off errors.

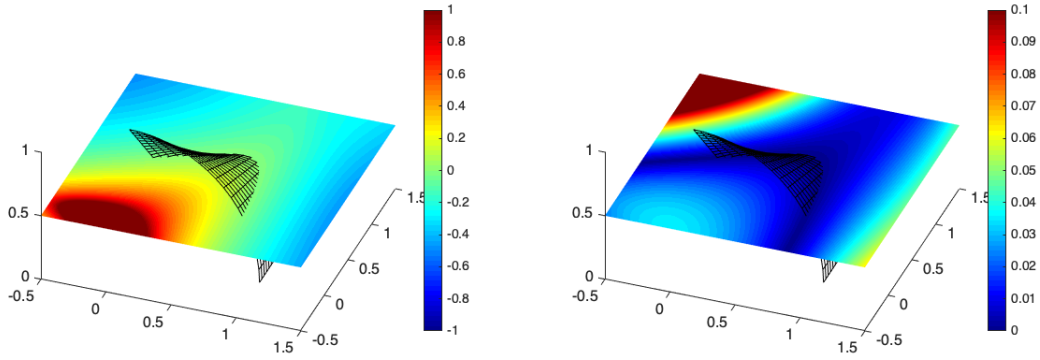
Computation times for both parts are listed in Tabs. 5.1 and 5.2, respectively. In Tab. 5.1, 25 different instances were run and the mean time tabulated. The M-Rep method is found to take much less time for pre-processing. This is due to the different robustness measures that must be taken for algebraic level sets. These measures are unnecessary for the M-Rep method, making it faster despite having to perform the computationally expensive singular value decomposition (SVD). It should be noted that this pre-processing step needs to be performed only once for a given geometry in either method. Table 5.2 lists the computation times for distance estimation, calculated as the mean of 10,000 points. As can be observed from the table, the proposed algebraic level sets method takes much less time for distance estimation as compared to the M-Rep. This is because the proposed method requires a simple determinant computation for the distance estimation while the M-Rep requires a SVD operation at each point. Further, for geometries like the quarter-cylinder and spherical

octant that had higher pre-processing times, using algebraic level sets is even faster since the maximal-rank submatrix method reduces the size of the resultant matrix.



(a) (b)

Figure 5.8. : Algebraic level sets generated for a cubic curve using the (a) normalized Dixon resultant and (b) M-Rep method. The level sets generated by the M-Rep method are non-negative everywhere. The algebraic level sets generated by the normalized Dixon resultant differ in sign on either side of the curve. No robustness measures were required for the normalized Dixon resultant.



(a) (b)

Figure 5.9. : Algebraic level sets generated for a bi-quadratic surface using the (a) normalized Dixon resultant and (b) M-Rep method. The level sets generated by the M-Rep method are non-negative everywhere. The algebraic level sets generated by the normalized Dixon resultant differ in sign on either side of the surface. No robustness measures were required for the normalized Dixon resultant.

Table 5.1. : Pre-processing times for algebraic level sets and the M-Rep method for different geometries. A total of 25 different instances were considered for each geometry, and the mean time computed. Pre-processing refers to computing a maximal-rank submatrix and/or a polynomial square root.

Geometry	Algebraic Level Sets (ms)	M-Rep (ms)	Remarks
2D Bézier curve (Fig. 5.8)	0.261	0.123	–
3D Bézier curve (Fig. 5.9)	0.810	0.256	–
Quarter Cylinder (Fig. 5.2a)	1.700	0.249	Requires maximal-rank submatrix computation
Spherical Octant (Fig. 5.3)	1.950	0.274	Requires maximal-rank submatrix and polynomial square root computation

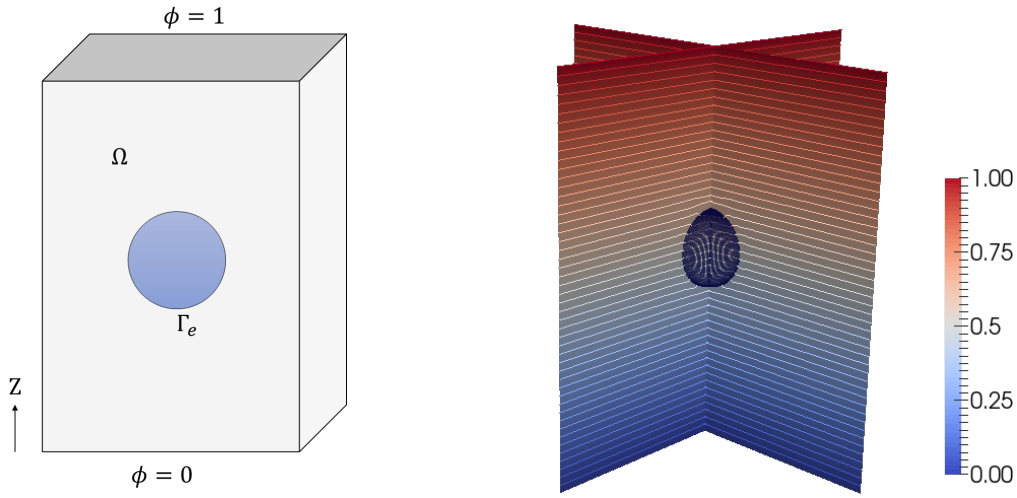
Table 5.2. : Computation time for distance estimation using algebraic level sets and the M-Rep method for different geometries. The distance was estimated at 10,000 points and the mean time computed.

Geometry	Algebraic Level Sets (μs)	M-Rep (μs)
2D Bézier curve (Fig. 5.8)	9.70	10.64
3D Bézier curve (Fig. 5.9)	8.28	13.03
Quarter Cylinder (Fig. 5.2a)	5.20	12.64
Spherical Octant (Fig. 5.3)	6.66	12.32

5.3 Application to EIGA: Current Through Line with Void

The application of Enriched Isogeometric Analysis (EIGA) to three-dimensional problems has been limited due to the issues with the Dixon resultant underscored in the previous chapter. As a demonstration, the robustness enhancements developed in this chapter are now applied to a three-dimensional electrostatics problem of passing current through a line with a void. An EIGA formulation similar to § 3.3 is used for the analysis. A schematic of the problem is shown in Fig. 5.10a. The electric potential is governed by the Laplace equation in the domain. Dirichlet boundary conditions are prescribed on the top and bottom surfaces while the walls are assumed adiabatic. No electric flux is allowed to enter or exit the void. The approximation form used for the electric potential solution is analogous to that in § 3.3, with a quartic weight function as defined in Eq. (3.6b),

$$\phi(\boldsymbol{x}) = (1 - w)\phi_c(\boldsymbol{x}) + w\phi_e(\mathcal{P}(\boldsymbol{x})) \tag{5.36}$$



(a) (b)

Figure 5.10. : (a) Schematic of the three-dimensional electrostatic problem with a void. (b) Electric potential solution contours on the mid-section planes in the case of a single spherical void.

If the void interface is represented as a parametric spline such as NURBS, then signed algebraic level sets can be used as the distance measure $d(\boldsymbol{x})$. As in § 3.3, this form of the electric potential automatically satisfies the void boundary condition Eq. (3.8). As a result, only the Laplace equation has to be solved for in the domain, along with the Dirichlet boundary conditions. The electric potential solution for a system with a single spheroidal void is shown in Fig. 5.10b. The figure shows the electric potential contours on the two mid-section planes. For the spherical void, the Dixon resultant is found to be identically zero. Hence, the maximal-rank submatrix method was used to recover the resultant. However, the recovered resultant was found to be negative everywhere. Consequently, the polynomial square root procedure was employed to generate signed algebraic level sets. These level sets were then used as the distance measure for the weighted blending in Eq. (5.36).

An approximate analytical solution for the electric potential near the void is now used to validate the EIGA solution. A spherical coordinate system (r, θ, ϕ) with origin at the center of the spherical void is chosen. In the absence of the void, the analytical solution to the problem is $\phi(\boldsymbol{x}) = z$. The presence of the void is assumed to perturb this solution as $\phi(\boldsymbol{x}) = z + \phi'(\boldsymbol{x})$, where $\phi'(\boldsymbol{x})$ satisfies the Laplace equation and vanishes on moving away from the void. Further, the perturbed solution must satisfy the interface condition Eq. (3.8).

$$\nabla^2 \phi'(\boldsymbol{x}) = 0 \tag{5.37}$$

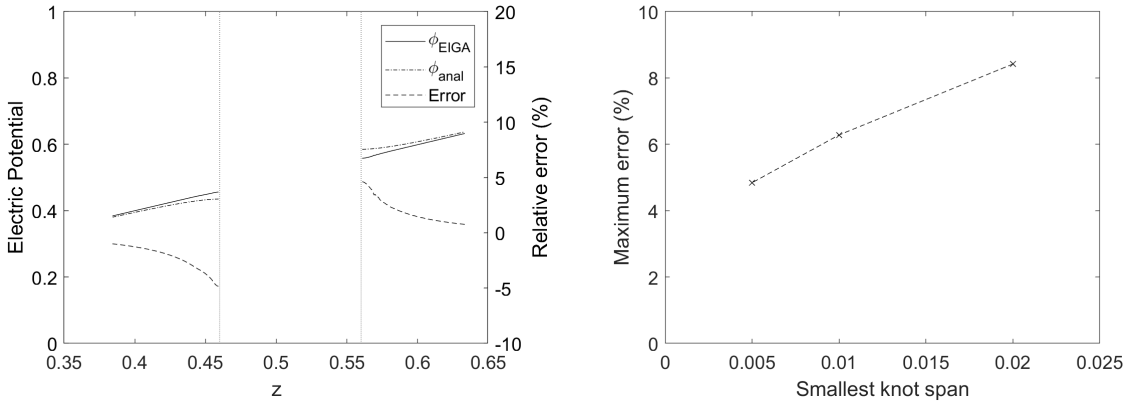
$$\lim_{r \to \infty} \phi'(\boldsymbol{x}) = 0 \tag{5.38}$$

$$\frac{\partial \phi'}{\partial r} = -\cos\theta \quad \text{at } r = R \tag{5.39}$$

where, R is the radius of the spherical void. This gives the perturbation as $\phi'(\boldsymbol{x}) = \frac{R^3}{2r^2}\cos\theta$, and the perturbed solution as,

$$\phi(\boldsymbol{x}) = z + \frac{R^3}{2r^2}\cos\theta \tag{5.40}$$

This analytical solution is accurate only near the void. The electric potential solution obtained from enriched isogeometric analysis is compared with Eq. (5.40) for points along a vertical line passing through the center of the void, and the results shown in Fig. 5.11. Points inside the void were ignored for the comparison. Discretization size is measured in the parametric space in the form of the smallest knot span. The EIGA solution shows good agreement with the analytical solution, with a maximum relative error magnitude of 4.84%. The maximum error shows decrease with decrease in the smallest knot span.



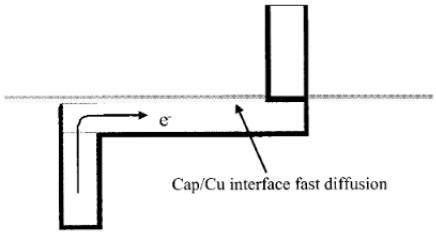
(a) (b)

Figure 5.11. : (a) Comparison between the analytical electric potential and the solution obtained from enriched isogeometric analysis, along the vertical line passing through the center of the spherical void. Points inside the sphere have been ignored. The smallest knot span was 0.005. The maximum relative error magnitude is 4.84% (b) Maximum relative error magnitude reduces with decrease in the smallest knot span.

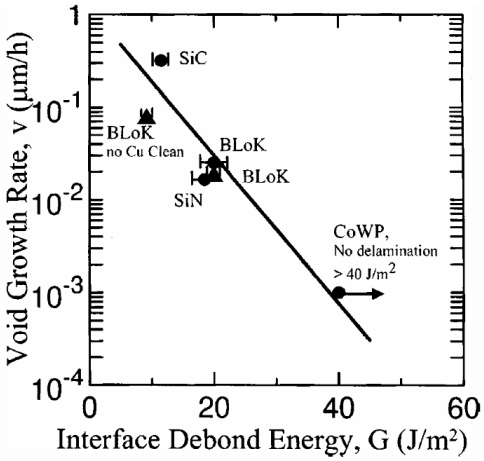
6. ELECTROMIGRATION DRIVEN VOID GROWTH

Electromigration is an important failure concern in the semiconductor industry. It occurs in lines carrying high current densities, where the electrons impart sufficient momentum to metal ions to displace them. This leads to formation of voids near the cathode and accumulation of material, called hillocks, near the anode. The developed voids could evolve and grow large enough under the imposed electric field so as to sever the line and cause open circuit failure. Electromigration experiments on line structires have shown a strong relation between void growth rate and the presence of cap layers. In [10], copper lines with different cap layers were subjected to electromigration to measure void growth rate. Separately, mechanical interfacial fracture experiments were performed to measure interfacial debond energy of the cap layers. It was observed that cap layers with higher debond energy induced slower void growth (see Fig. 6.1). In [91], Blech-like test structures of Cu-TiN lines with TiN, SiN and Ta cap layers were considered. On passing large currents, voiding was observed near the cathode and found to cause displacement of the copper edge. The edge motion was tracked optically under a scanning electron microscope and used to measure the void growth rate under different conditions of temperature and current densities. The results obtained are shown in Fig. 6.2. As the copper edge moved towards the anode, the ends were observed to move faster than the center, causing the edge to develop curvature.

In this chapter, EIGA and signed algebraic level sets are used to model electromigration driven void growth. Firstly, the velocity at a void interface is derived from first priciples, using the interfacial balance laws and growth conditions developed in Chapter 2. This is followed by solving a three-dimensional electromigration problem of void evolution in a current line. Enriched IGA and algebraic level sets are used, along with the recitification measures proposed in the previous chapter.



(a)



(b)

Figure 6.1. : (a) Schematic of the line-via-line structure used in [10] (b) Void growth rate as a function of interfacial debond energy [10]

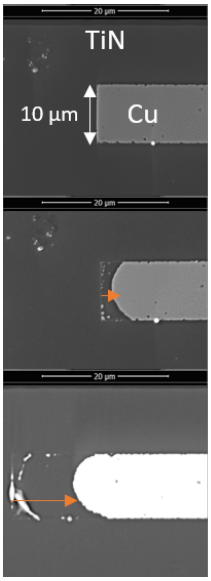


Figure 6.2. : Observed edge displacement in a $10\mu m$ width Cu-TiN line. The flat copper edge becomes curved as it moves towards the anode.

The problem formulation for electromigration is very similar to the electrostatics problem discussed in § 5.3, with the schematic shown in Fig. 5.10a. The electric potential is governed by the Laplace equation, with Dirichlet boundary conditions at the top and bottom. No electric flux is allowed to enter or exit the void. Mathematically, we have,

$$\Delta\phi = 0 \quad \text{in } \Omega \tag{6.1a}$$

$$\nabla\phi \cdot \boldsymbol{n} = 0 \quad \text{on } \Gamma_e \tag{6.1b}$$

$$\phi = 0 \quad \text{on } z = 0 \tag{6.1c}$$

$$\phi = 1 \quad \text{on } z = 1 \tag{6.1d}$$

6.1 Velocity at Void Interface

Void growth is driven by the external electric field, as well as by gradients in stress and temperature. For void growth to be thermodynamically feasible, any point on the void interface should satisfy the interfacial second law conditions derived in Chapter 2 at all times. Assuming small deformation diffusive void growth, we have the second law condition Eq. (2.77) give,

$$\boldsymbol{n} \cdot \boldsymbol{\Sigma}\boldsymbol{n} - \sum_{\alpha=1}^{N} \rho\mu^{\alpha}\nu^{\alpha} + \gamma\kappa \geq 0 \tag{6.2}$$

For a solid with a void, there is only one material phase (denoted sol). Writing $\rho^{\text{sol}} = \rho\nu^{\text{sol}}$, the above equation gives $\rho^{\text{sol}}\mu^{\text{sol}} \leq \boldsymbol{n} \cdot \boldsymbol{\Sigma}\boldsymbol{n} + \gamma\kappa$, allowing us to express the chemical potential as,

$$\mu^{\text{sol}} = \frac{c_0}{\rho^{\text{sol}}}(\boldsymbol{n} \cdot \boldsymbol{\Sigma}\boldsymbol{n} + \gamma\kappa) \tag{6.3}$$

where, $c_0 \leq 1$ is some constant.

An expression for the velocity at the void interface may be obtained by reducing the interface species balance condition Eq. (2.21) for a solid with a single material phase and a void,

$$v_{\mathrm{s_n}} = \frac{1}{\rho^{\mathrm{sol}}} \left(\boldsymbol{j}_{sol} \cdot \boldsymbol{n} + \nabla_{\mathrm{s}} \cdot \boldsymbol{h}_{\mathbf{t}} - r_s \right) \tag{6.4}$$

For void growth, we can assume the bulk mass flux and species generation to be negligible. A constitutive relation is now assumed for the surface mass flux. Since void growth is driven by the external electric field as well as gradients in the stress and concentration, we assume the constitutive relation,

$$\boldsymbol{h}_{\mathbf{t}} = -M_s \rho^{\mathrm{sol}} \Omega_{\mathrm{sol}} \left(\nabla_{\mathrm{s}} \left(\rho^{\mathrm{sol}} \Omega_{\mathrm{sol}} \mu^{\mathrm{sol}} \right) + Z^* e \nabla_{\mathrm{s}} \phi_e \right) \tag{6.5}$$

where, Ω_{sol} is the atomic volume for the solid, M_s is the surface mobility of the solid, Z^* is the effective charge and e is the charge on an electron. Thermal gradients are assumed to be negligible and are ignored here. Using Eq. (6.3) with this constitutive relation gives the void growth velocity as,

$$v_{\mathrm{s_n}} = -M_s c_0 \Omega_{\mathrm{sol}}^2 \nabla_s^2 (\boldsymbol{n} \cdot \boldsymbol{\Sigma} \boldsymbol{n} + \gamma \kappa) - M_s Z^* e \Omega_{\mathrm{sol}} \nabla_s^2 \phi_e \tag{6.6}$$

Neglecting mechanical contributions and introducing constants c_1, c_2, we can write,

$$v_{\mathrm{s_n}} = c_1 \nabla_s^2 \kappa + c_2 \nabla_s^2 \phi_e \tag{6.7}$$

where, $c_1 = -M_s c_0 \gamma \Omega_{\mathrm{sol}}^2, c_2 = -M_s Z^* e \Omega_{\mathrm{sol}}$. Since Enriched IGA includes an explicit representation of the void interface, the curvature and surface Laplacians may be computed directly and accurately. The procedure for computing the curvature and the surface Laplacians is described in Appendix G. This expression for the velocity can now be used to evolve the static void described in § 5.3. Forward Euler time stepping

is used in the simulations, and the step length at each time step is determined through a CFL condition,

$$\Delta t = C_t \frac{h}{\max_i |\boldsymbol{v}_i|} \tag{6.8}$$

where, C_t is the CFL constant and h is the discretization size.

6.2 Electromigration Driven Void Evolution in Line

A parametric representation using NURBS is used for the void interface. Enriched IGA with Eq. (5.36) as the form for the electric potential is used to solve the electromigration system of equations (Eq. (6.1)) at each time step. Velocities are determined for the control points of the void surface such as to satisfy Eq. (6.7) in a least squares sense. A time step is chosen based on the CFL condition, and the void is evolved by moving the control points with the solved velocities.

Table 6.1. : Values chosen for the constants for the void evolution simulations.

c_1	1.8×10^{-4}
c_2	6.0×10^{-2}

Since adaptive time stepping is used, the absolute values of the constants c_1, c_2 in Eq. (6.7) are not as important as their relative magnitudes, with a large c_1 value describing a material with large surface energy. The values chosen for the constants are given in Tab. 6.1. Two cases were considered. The first case was the limiting case of low or negligible surface stress. In this case, the $\nabla_s^2\kappa$ term in Eq. (6.7) was neglected to describe a material with very low surface energies. Snapshots of the void evolving with time, with electric potential contours, are shown in Fig. 6.3. The void moves towards the cathode, as is observed in electromigration experiments. The void also shows significant shape distortion as it moves. It took about 1.9 hrs for the simulations to run for 60 time steps on a 8GB RAM computer, at an average of about 2 mins per time step. The second case includes the surface stress ($\nabla_s^2\kappa$ term) in the evolution of the void. Snapshots of the void for this case, with electric potential

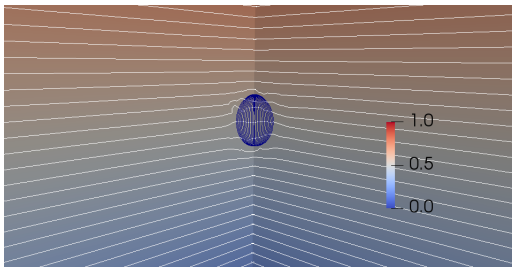
contours, are shown in Fig. 6.4. Though the void moves towards the cathode in this case as well, it shows less distortion in its shape. This is because $\nabla_s^2\kappa$ is a stabilising term that drives the void shape to the minimum surface energy configuration of a sphere. It took about 1.37 hrs for the simulations to run for 60 time steps on a 8GB RAM computer, at an average of about 82s per time step.

(a) n=1

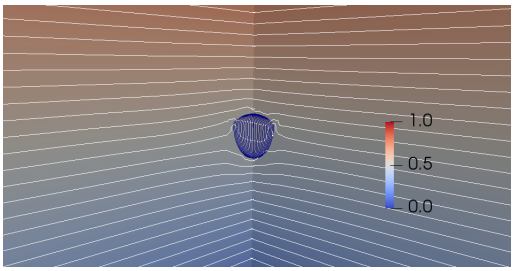
(b) n=20

(c) n=40

(d) n=60



(e) n=6



(f) n=80

Figure 6.3. : (a)-(d) Snapshots of the void at different instances, for a material with negligible surface stress. The electric potential solution contours on the mid-section planes at (e) n=6 time steps and (f) n=80 time steps. As the void moves towards the cathode, it shows significant shape distortion as well.

6.3 Void Growth in Cu-TiN Line

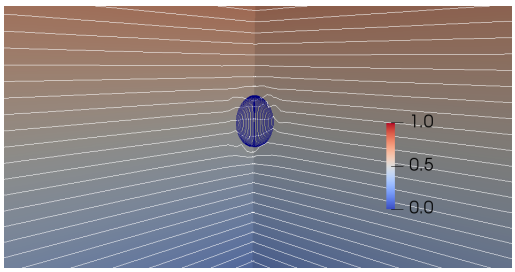
The electromigration experiments of [91] shall now be modeled using EIGA. As shown in Fig. 6.2, the copper edge is observed to curve as it moves towards the anode. This behavior shall be shown to be captured in the simulations as well.

(a) n=1

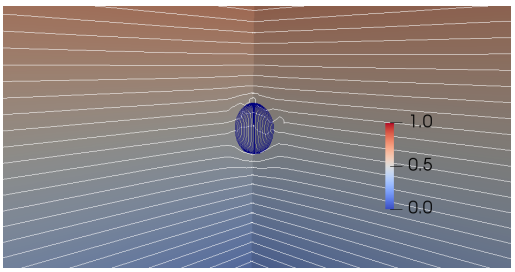
(b) n=20

(c) n=40

(d) n=60



(e) n=6



(f) n=50

Figure 6.4. : (a)-(d) Snapshots of the void at different instances, for a material with negligible surface stress. The electric potential solution contours on the mid-section planes at (e) n=6 time steps and (f) n=50 time steps. While the void moves towards the cathode, it does not show much distortion in its shape.

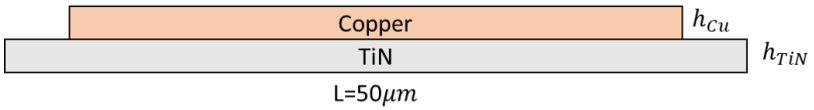


Figure 6.5. : Side view of the geometry used for the simulations. Both layers have a width of $10\mu m$ into the plane of the paper.

6.3.1 Model Formulation

Blech-like test structures of Cu-TiN lines are used for the simulations, as depicted in Fig. 6.5. Dirichlet boundary conditions of zero electric potential are imposed on the left TiN interface, while Neumann boundary conditions are applied on the right TiN interface to impose required current densities. Current densities considered here are $1 - 3 \times 10^6 A/cm^2$.

Table 6.2. : Material properties of copper used in the simulations.

Property	Value	Units
Electric conductivity	5.998×10^7	S/m
Density	8700	kg/m^3
Z^*	0.5	—
Atomic radius	0.128	nm
Adhesion energy	10	J/m^2

Table 6.3. : Material properties of copper in terms of the scaled units. These values were directly used in the simulations.

Property	Value	Units
Electric conductivity	59.98	$S/\mu m$
Density	8.7	$pg/\mu m^3$
Z^*	0.5	—
Atomic radius	1.28×10^{-4}	μm
Adhesion energy	10^{-5}	$\mu J/\mu m^2$

The material properties of copper used in the simulations are provided in Tab. 6.2. The electrical conductivity of TiN is taken to be 1.33×10^7 S/m. For better numerical stability, scaled units are used. The length scale chosen for analysis is the μm, while

the scale chosen for the mass is pg, i.e., $10^{-15}kg$. Electric current and potential are denoted in mA and mV, respectively. The scaled material properties of copper in terms of these units are provided in Tab. 6.3. The electrical conductivity of TiN scales to 1.33 $S/\mu m$, while the imposed current density is $10 - 30$ $mA/\mu m^2$. The velocity for void evolution is given by Eq. (6.7). In terms of the scaled units, it can be seen that the constants c_1, c_2 for the Cu-TiN line simulations should be of comparable magnitudes,

$$\frac{c_2}{c_1} = \frac{Z^* e}{\Omega_{\text{sol}} \gamma c_0} \sim 1 \ (mV)^{-1} (\mu m)^{-1} \tag{6.9}$$

To analyze each coefficient in Eq. (6.7) separately, a scaling for $M_s \Omega_{\text{sol}}$ is to be determined. This quantity can be written as [131],

$$M_s \Omega_{\text{sol}} = \frac{D_s h_{\text{int}}}{k_B T}$$

where, D_s is the surface diffusivity and h_{int} is the interface thickness. The surface diffusivity for copper is given as $10^{-8} cm^2/s$ or 1 $\mu m^2/s$ [132]. Based on the other chosen scaled units, the compatible scaled unit for time is $\overline{\tau} = 0.1 \mu s$ which gives $D_s = 10^{-7} \mu m^2/\overline{\tau}$. An interface thickness of 10 A^0 is assumed, while the temperature is chosen to be $275^{\circ} C$, based on the experiments in [91]. Thus,

$$M_s \Omega_{\text{sol}} = \frac{D_s h_{\text{int}}}{k_B T} = 1.32 \times 10^{-4} \mu m \ \overline{\tau}/pg \tag{6.10}$$

The constants c_1, c_2 in Eq. (6.7) can now be evaluated in terms of the scaled quantities as,

$$c_1 = (M_s \Omega_{\text{sol}})(c_0 \gamma \Omega_{\text{sol}}) = 1.16 \times 10^{-13} \mu m^4/\overline{\tau} \tag{6.11}$$

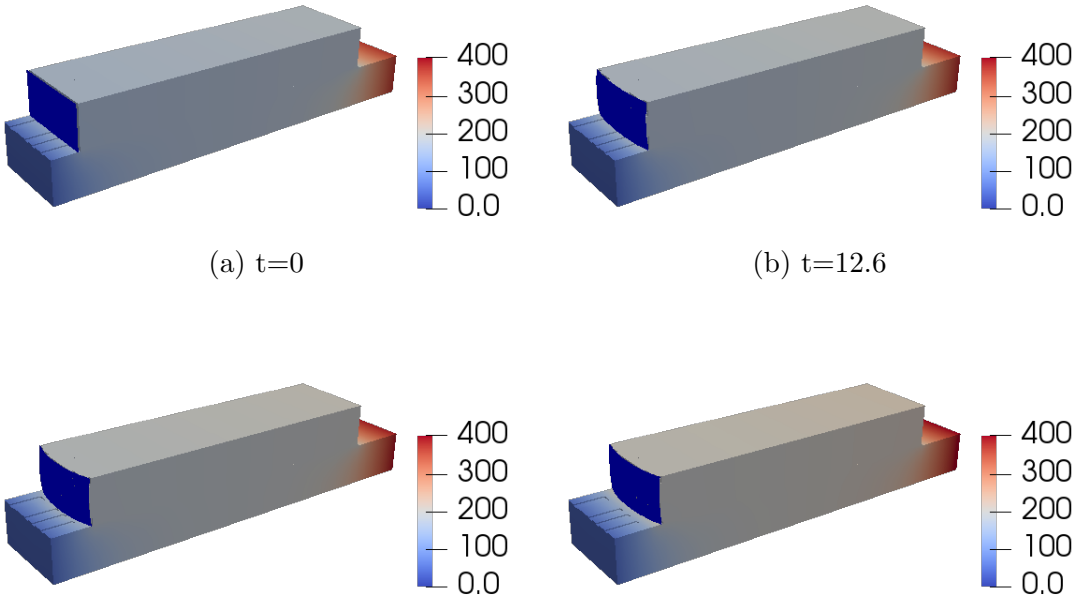
$$c_2 = (M_s \Omega_{\text{sol}})(Z^* e) = 1.056 \times 10^{-13} \mu m\text{-}\overline{\tau}^2\text{-}mA/pg \tag{6.12}$$

This expresses the void velocity in $\mu m/\overline{\tau}$. To obtain the velocity in $\mu m/hr$, both constants have to be multiplied by 3600×10^7. Thus, the void velocity in $\mu m/hr$ for Cu-TiN line structures under electromigration is given by,

$$v_{\mathrm{s_n}} = -4.17 \times 10^{-2} \nabla_s^2 \kappa - 3.80 \times 10^{-2} \nabla_s^2 \phi_e \tag{6.13}$$

Three-dimensional electromigration simulations were conducted on a Cu-TiN line structure. A line length of $L = 50\mu m$ and width $w = 10\mu m$ were used. The copper and TiN layers had a height of $h = 5\mu m$ each, and a current density of $3 \times 10^6 A/cm^2 = 30mA/\mu m^2$ was applied as a Neumann boundary condition. Laplace equation was solved for in the domain, and Eq. (6.13) was used to evolve the copper-void interface. A parametric representation was used for the void interface using NURBS. The simulations took about 2.5 hrs for 300 time steps on an 8GB RAM desktop computer with a 2.60GHz processor, at an average of about half a minute per time step. The results are shown in Fig. 6.6. As expected, most of the potential drop is across the TiN extensions, with very little potential drop across the copper layer. This is because the electric conductivity of copper is 60 times that of TiN. While the copper-void interface is initially nearly planar, it begins to curve as the void interface moves. However, the curvature is restricted to the width direction of the line, and there is very little curvature along the thickness direction. This suggests that a two-dimensional model would be sufficient to capture the void growth in Cu-TiN lines.

A two-dimensional model was chosen to capture behaviour at the void interface region. A rectangular domain was used for the line, with a total length of $L = 100\mu m$ and width of $w = 20\mu m$. A planar interface was assumed at the initial time step (see Fig. 6.7a), dividing the line into a TiN segment of length $25\mu m$ and a Cu-TiN segment of length $75\mu m$. Since copper has much higher electrical conductivity than TiN (~ 60 times), current can be assumed to flow predominantly through the copper layer. Thus, material below the void interface was assumed to be pure copper,



(a) t=0 (b) t=12.6

(c) t=25 (d) t=32

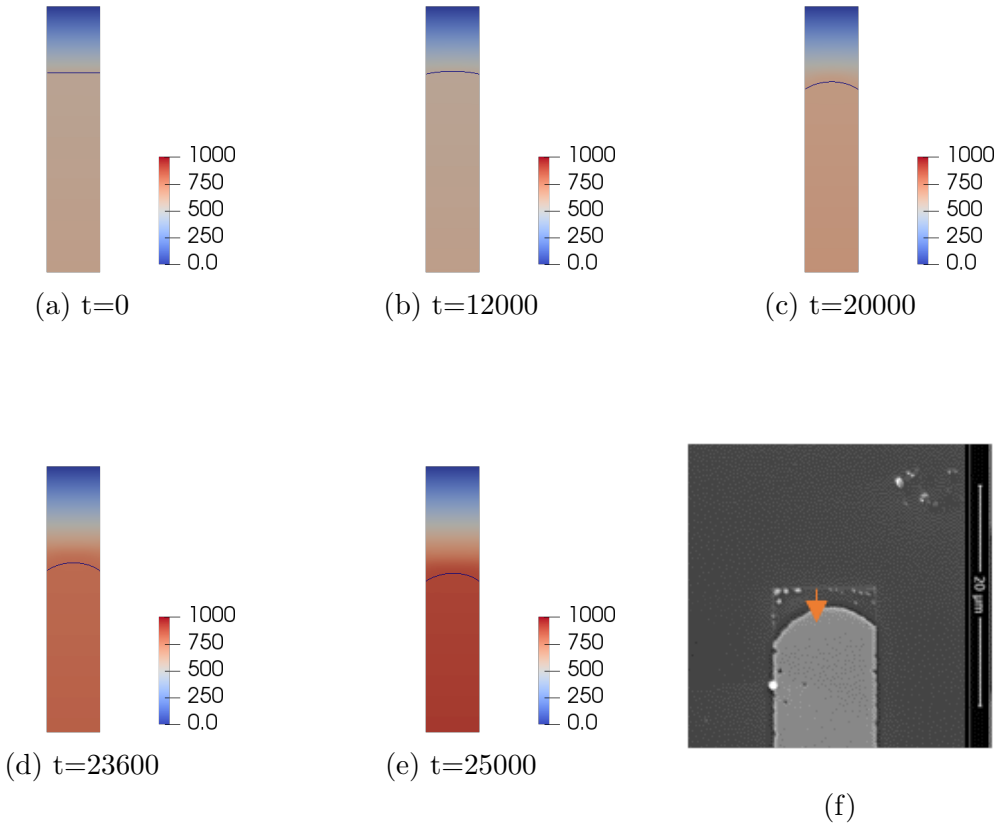
Figure 6.6. : Electric potential solution in the Cu-TiN line at different instances of time. While the void interface is initially planar, it begins to turn cylindrical as line shrinks.

carrying a current density of $3 \times 10^6 A/cm^2$ applied as a Neumann boundary condition at the bottom interface. Material above the void interface was taken to be TiN, with Dirichlet boundary conditions of zero potential at the top interface. At the void, no electric flux is assumed to enter or exit the interface as in the previous simulations. The simulations took about 27 minutes for 5000 time steps on a desktop computer, at an average of 0.3 s per time step. The results are shown in Fig. 6.7. As expected, there is very little potential drop across the copper layer. The line slowly shrinks with time. As the line shrinks, the resistance of the Cu-TiN structure is expected to increase. This can be observed here indirectly in the form of an increase in the total potential difference across the line. Further, while the void interface is initially planar, the interface develops a curvature as the line shrinks. This is in direct agreement with the observations in [91].

6.3.2 Incorporating Cap Layers

The presence of cap layers inhibits void growth since additional energy is required to debond the copper-cap layer interface. Let the interface debond energy for copper with the cap layer be $\overline{\gamma}^{\text{cap}}$. Assuming no slip at the interface, if v_{s_n} is the velocity of the void surface and v_n is the velocity due to deformation in copper, then $v_{\text{s}_\text{n}} - v_n$ denotes the rate of debond at the cap layer. This requires additional power, dissipated at the rate $\dot{E}_{\text{diss}} = \overline{\gamma}^{\text{cap}}\ p(v_{\text{s}_\text{n}} - v_n)$ where $p \approx w$ is the perimeter of the cap layer interface. Similarly, the Cu-TiN interface also dissipates energy at the rate $\overline{\gamma}^{\text{TiN}}\ L_c(v_{\text{s}_\text{n}} - v_n)$ where $L_c \approx w$ is the width of the Cu-void interface.

The resistance provided by the cap layer can be translated into a slow-down in species generation through the chemical potential. Since the chemical potential μ^{sol} denotes the energy per unit mass of the solid, $\dot{E}_{\text{diss}}/\mu^{\text{sol}}$ corresponds to the reduction in rate of species generation. Averaging this out over the total cross-section area A,



(a) t=0 (b) t=12000 (c) t=20000

(d) t=23600 (e) t=25000 (f)

Figure 6.7. : (a)-(e) Electric potential solution in the Cu-TiN line at different instances of time (f) Curved profile of copper edge observed in the experiments of [91]. While the void interface in the simulations is initially planar, it begins to curve as the edge moves. This shows good agreement with the profile observed in experiments.

and using the approximation $p/A \approx 1/h, L_c/A \approx 1/h$, we get the modified species balance equation,

$$-\left[\!\left[\rho^{\mathrm{sol}}(v_{\mathrm{s_n}} - v_n)\right]\!\right] = -\left[\!\left[\boldsymbol{j}_{\mathrm{sol}}\right]\!\right] \cdot \boldsymbol{n} - \nabla_{\mathrm{s}} \cdot \boldsymbol{h_t} + r_s + \left[\!\left[\frac{\overline{\gamma}^{\mathrm{cap}} + \overline{\gamma}^{\mathrm{TiN}}}{\mu^{\mathrm{sol}} h}(v_{\mathrm{s_n}} - v_n)\right]\!\right] \tag{6.14}$$

For a solid with a single material phase and a void under negligible bulk motion, this reduces to,

$$\rho^{\mathrm{sol}} v_{\mathrm{s_n}} = \boldsymbol{j}_{sol} \cdot \boldsymbol{n} + \nabla_{\mathrm{s}} \cdot \boldsymbol{h_t} - r_s - \frac{(\overline{\gamma}^{\mathrm{cap}} + \overline{\gamma}^{\mathrm{TiN}})}{\mu^{\mathrm{sol}} h} v_{\mathrm{s_n}} \tag{6.15}$$

$$\Rightarrow v_{\mathrm{s_n}} = \frac{1}{\rho^{\mathrm{sol}}\left(1 + \frac{\overline{\gamma}^{\mathrm{cap}} + \overline{\gamma}^{\mathrm{TiN}}}{\mu^{\mathrm{sol}} \rho^{\mathrm{sol}} h}\right)} \left(\boldsymbol{j}_{sol} \cdot \boldsymbol{n} + \nabla_{\mathrm{s}} \cdot \boldsymbol{h_t} - r_s\right) \tag{6.16}$$

Following the approach in § 6.1 and writing it in the form of Eq. (6.7) gives,

$$v_{\mathrm{s_n}} = \frac{c_1 \nabla_s^2 \kappa + c_2 \nabla_s^2 \phi}{\left(1 + \frac{\overline{\gamma}^{\mathrm{cap}} + \overline{\gamma}^{\mathrm{TiN}}}{\mu^{\mathrm{sol}} \rho^{\mathrm{sol}} h}\right)} \tag{6.17}$$

When there is no cap layer, $\overline{\gamma}^{\mathrm{cap}}$=0 and Eq. (6.7) is recovered. The addition of the cap layer slows the void growth. For very thick lines, the influence of the cap layer reduces as expected. For two copper lines with different cap layers (say SiN and Ta), the ratio of velocities is,

$$\frac{v_{\mathrm{s_n}}^{SiN}}{v_{\mathrm{s_n}}^{Ta}} = \frac{(\rho^{\mathrm{sol}} \mu^{\mathrm{sol}} h + \overline{\gamma}^{Ta} + \overline{\gamma}^{\mathrm{TiN}})}{(\rho^{\mathrm{sol}} \mu^{\mathrm{sol}} h + \overline{\gamma}^{SiN} + \overline{\gamma}^{\mathrm{TiN}})} = \frac{c_0 \kappa h + (\overline{\gamma}^{Ta} + \overline{\gamma}^{TiN})/\gamma}{c_0 \kappa h + (\overline{\gamma}^{SiN} + \overline{\gamma}^{TiN})/\gamma} \tag{6.18}$$

where, we have taken $\rho^{\mathrm{sol}} \mu^{\mathrm{sol}} = c_0 \gamma \kappa$. Typically, $\kappa \sim 1/w \sim 0.1 \mu m^{-1}, h \sim 0.1 \mu m, c_0 \sim 1$, while $\overline{\gamma}^{\mathrm{cap}}/\gamma \sim 1$. Thus, the $c_0 \kappa h$ terms can be neglected in favour of the adhesion energy terms. This provides the simplified expression for the ratio of velocities,

$$\frac{v_{\mathrm{s_n}}^{SiN}}{v_{\mathrm{s_n}}^{Ta}} = \frac{\overline{\gamma}^{Ta} + \overline{\gamma}^{TiN}}{\overline{\gamma}^{SiN} + \overline{\gamma}^{TiN}} \tag{6.19}$$

The above expression shows good agreement with the behaviour observed in [91].

7. MODELING TOPOLOGICAL CHANGES USING ALGEBRAIC LEVEL SETS

Problems with evolving phase interfaces generally involve topological changes namely merging, splitting, nucleation and dissolution of phases. For example in electromigration driven voiding in solder shown in Fig. 1.2, numerous microscopic voids coalesce to form larger voids. When an explicit representation is used for evolving interfaces, such topological changes introduce geometric challenges. They generally require contact detection and computation of surface-surface intersections at each time step. Common strategies for such geometric operations include subdivision, marching, lattice evaluation and implicitisation [65]. Performing these operations at each and every time step is computationally prohibitive.

As seen in Chapter 3, the Enriched IGA method uses an explicit parametric representation for evolving phase interfaces. However, analysis in EIGA is based on distance based weighted blending, and consequently on signed algebraic level sets. Topological changes on the geometry can be translated into algebraic Boolean compositions on the signed algebraic level sets. This allows modeling topological changes using EIGA, without detecting or computing surface-surface intersections.

7.1 Boolean Compositions on Algebraic Level Sets

Signed algebraic level sets can be used to perform Boolean operations on simple geometries to obtain level sets for composite geometries [85]. First, algebraic level sets are generated for each constituent geometry. These algebraic level sets are composed using appropriate R-functions defined in Eqs. (7.1) to (7.3). The result is a signed algebraic level set corresponding to the composite geometry.

Common Boolean operations performed on primitives are negation, union, intersection and difference. Consider any two primitive geometries, say A and B. If d_1

and d_2 are the algebraic level sets for these primitives, then the result of Boolean operations on these primitives can be described by the algebraic level set d given by,

- **Negation** $(\neg A)$:

$$d = \neg d_1 = -d_1 \tag{7.1}$$

- **Union** $(A \cup B)$:

$$d = d_1 \vee d_2 = d_1 + d_2 + \sqrt{d_1^2 + d_2^2} \tag{7.2}$$

- **Intersection** $(A \cap B)$:

$$d = d_1 \wedge d_2 = d_1 + d_2 - \sqrt{d_1^2 + d_2^2} \tag{7.3}$$

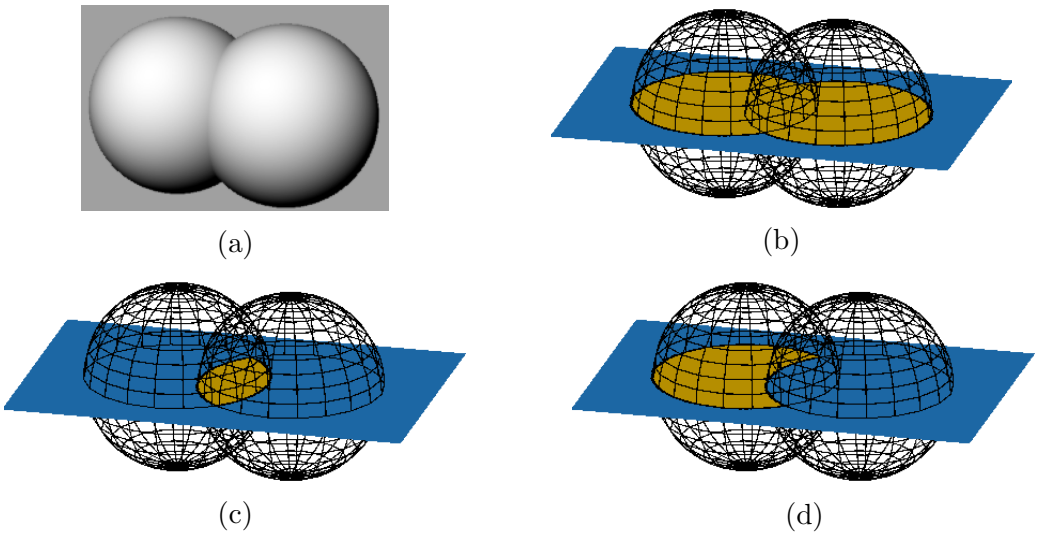
- **Difference** $(A - B = A \cap B')$:

$$d = d_1 \wedge \neg d_2 = d_1 - d_2 - \sqrt{d_1^2 + d_2^2} \tag{7.4}$$

These operations were performed for two spheres to obtain algebraic level sets for the different composite geometries, and the results are shown in Fig. 7.1. Generating level sets for the spheres required finding a maximal rank submatrix for their Dixon resultants, followed by computing their algebraic square roots as per the procedure described in § 5.2.2.

7.2 Modeling Topological Changes

Topological changes on the geometry can be translated into algebraic Boolean compositions on the signed algebraic level sets. This idea is illustrated in Fig. 7.2 for a system with two spherical voids. The figure shows the Boolean union (from Eq. (7.2)) of the signed algebraic level sets of the two voids, both before and after coalescence. It can be observed that the union operation automatically generates signed level sets for the merged void upon coalescence. Since the analysis depends only on the



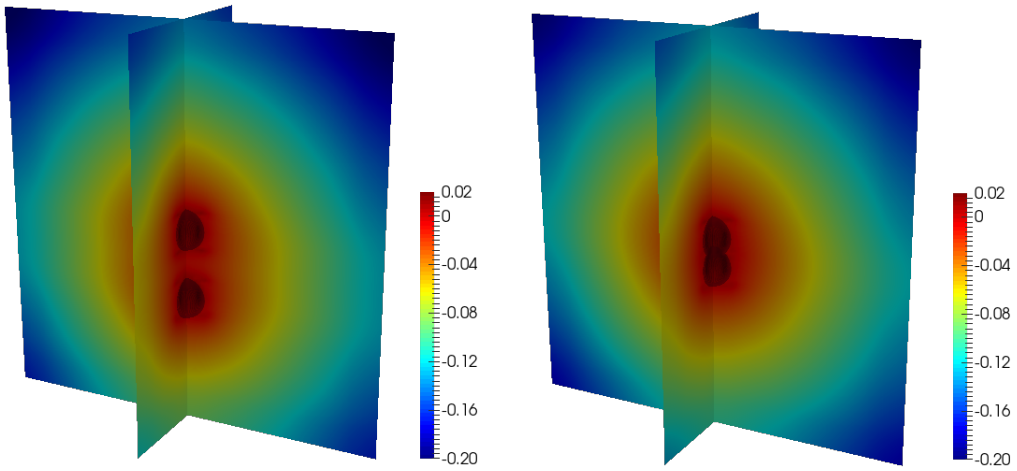
(a) (b) (c) (d)

Figure 7.1. : Sign of algebraic level sets obtained for Boolean compositions of two spheres. Yellowish brown denotes positive level sets (inside) while blue indicates negative level sets. (a) CAD model of the two spheres. (b) Union, (c) intersection and (d) set difference of the two spheres.

Table 7.1. : Computation times for electromigration simulations of the different systems considered. The systems are referred by the index of the corresponding figure. There is no significant overhead on handling systems with coalescent voids.

Electromigration System	Degrees of Freedom	Time (in s)	
		Pre-merging	Post-merging
Two bean-shaped voids (Fig. 7.4)	259	0.99	0.73
Three bean-shaped voids (Fig. 7.5)	273	2.15	1.61
One spherical void (Fig. 5.10b)	2465	29.64	N/A
Two spherical voids (Fig. 7.3)	2510	75.0	72.62

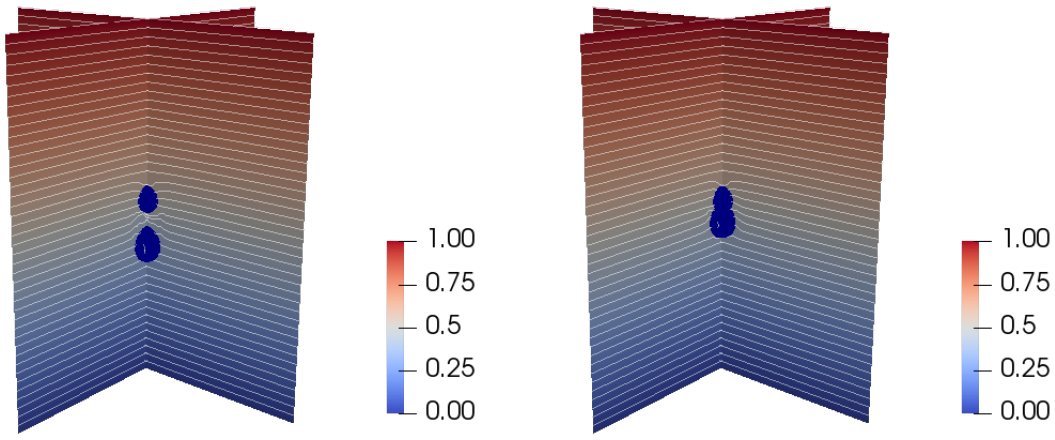
weight field and hence the algebraic level sets, coalescent voids can be accommodated without resorting to collision detection and intersection computation. Solving the electromigration problem as in § 5.3 for a solid with two spherical voids, we get electric potential solutions as shown in Fig. 7.3.



(a) (b)

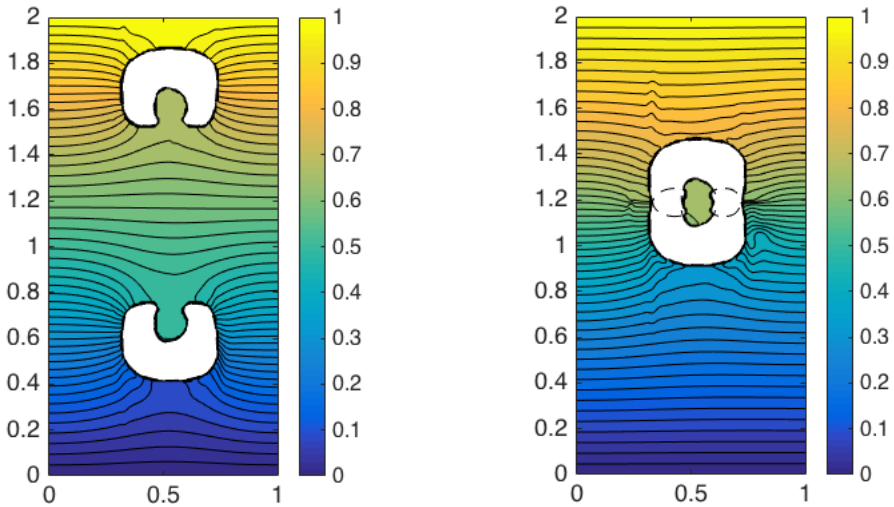
Figure 7.2. : Signed algebraic level sets for a system with two interacting voids (a) without coalescence (b) with coalescence, generated as the Boolean union of the individual level sets in both cases.

Further illustrations are provided in Figs. 7.4 to 7.6 for two-dimensional and three-dimensional systems with free-form voids with and without coalescence, forming complex topologies. The three-dimensional free-form voids required both the maximal-rank submatrix approach, as well as the polynomial square root method to generate



(a) (b)

Figure 7.3. : Contours of the electric potential solution for a system with two interacting spherical voids (a) without coalescence (b) with coalescence.



(a) (b)

Figure 7.4. : Contours of the electric potential solution for a system with two free-form voids (a) without coalescence (b) with coalescence.

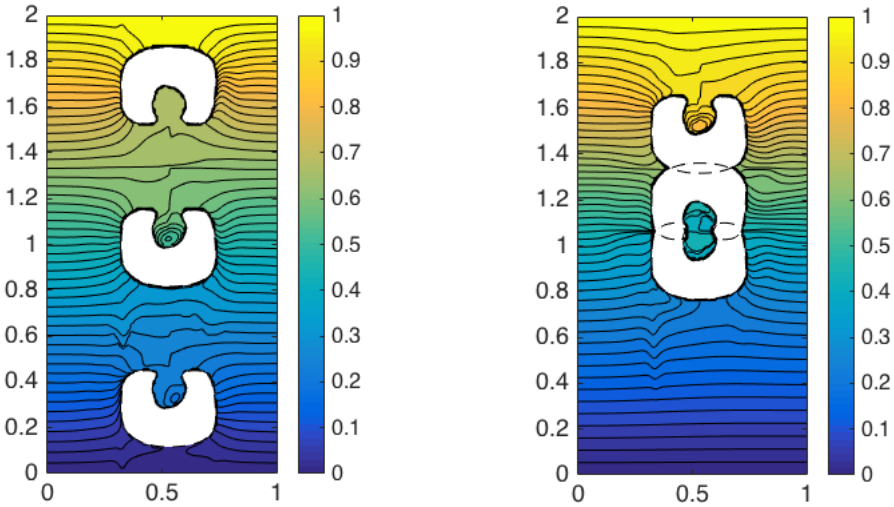


Figure 7.5. : Contours of the electric potential solution for a system with multiple free-form voids (a) without coalescence (b) with coalescence.

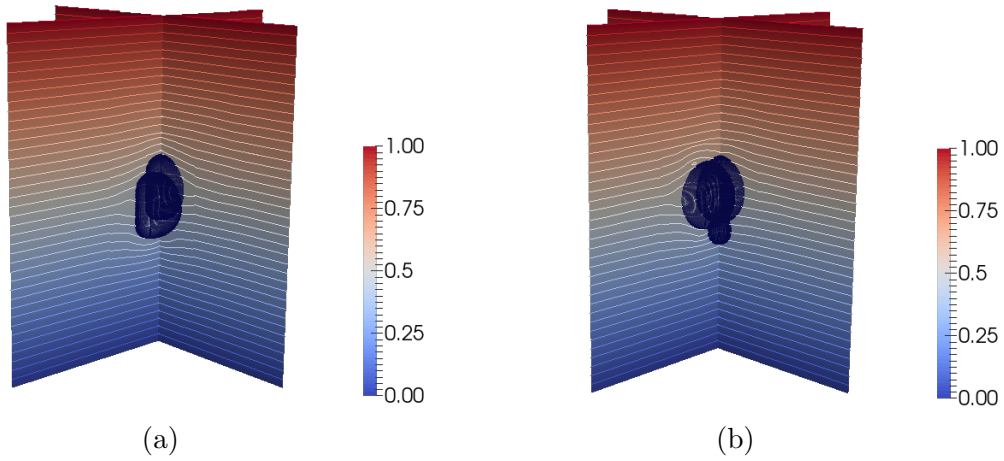


Figure 7.6. : Contours of the electric potential solution at the mid-section planes for a solid with (a) two coalescent free-form voids (b) three coalescent free-form voids.

individual signed algebraic level sets. The efficiency of the proposed approach is demonstrated in Tab. 7.1, where it is observed that there is no overhead when coalescence occurs. The solution time is unaffected by topological changes such as merging of phases.

7.3 Application to Bubble Evolution

To demonstrate the capabilities of modeling topological changes using Boolean operations, a bubble coalescence problem shall now be considered. A collection of bubbles are introduced, and each bubble is prescribed with an upward veloctiy, inversely proportional to its size. That is, larger bubbles "float" slower than smaller ones, causing different bubbles to coalesce. To conserve mass, assuming constant density, the total volume of all bubbles should be conserved. Mathematically, for a collection of N bubbles, the conservation of volume can be written as,

$$\frac{dV}{dt} = \frac{d}{dt}\left(\sum_{i=1}^{N}\int_{\Omega} d\Omega\right) = \sum_{i=1}^{N}\int_{\Omega}(\nabla\cdot\boldsymbol{v})d\Omega = \sum_{i=1}^{N}\int_{\Gamma}\boldsymbol{v}\cdot\boldsymbol{n}\; d\Gamma = 0 \tag{7.5}$$

Let the prescribed velocity for the i^{th} bubble be denoted as $\boldsymbol{v}_0^i(\boldsymbol{x})$. A parametric spline representation, namely NURBS, is used to represent the bubbles. Motion of a bubble corresponds to motion of its control points. The bubble coalescence problem can thus be described as a least-squares fitting of the prescribed velocities subject to the volume constraint, Eq. (7.5).

7.3.1 Problem Formulation

Mathematically, the bubble evolution problem can be written as,

$$\min\ I = \sum_{i=1}^{N} \left(\frac{1}{2} \int_{\Gamma_i} \left\| \boldsymbol{v} - \boldsymbol{v}_0^i \right\|^2 \ d\Gamma \right) \tag{7.6}$$

$$s.t. \quad \sum_{i=1}^{N} \left(\int_{\Gamma_i} \boldsymbol{v} \cdot \boldsymbol{n} \ d\Gamma \right) = 0 \tag{7.7}$$

The volume constraint can be moved to the objective function by introducing Lagrange variables λ. This gives the Lagrangian function,

$$\mathcal{L}(\boldsymbol{v}, \lambda) = \frac{1}{2} \sum_{i=1}^{N} \int_{\Gamma_i} \left\| \boldsymbol{v} - \boldsymbol{v}_0^i \right\|^2 \ d\Gamma - \lambda \sum_{i=1}^{N} \int_{\Gamma_i} \boldsymbol{v} \cdot \boldsymbol{n} \ d\Gamma \tag{7.8}$$

The solution occurs at a stationary point of the Lagrangian function $\mathcal{L}$. Taking the variation of $\mathcal{L}(\boldsymbol{v}, \lambda)$ gives,

$$\delta\mathcal{L} = \sum_{i=1}^{N} \int_{\Gamma_i} \left[(\boldsymbol{v} - \boldsymbol{v}_0) \cdot \delta\boldsymbol{v} - \lambda \boldsymbol{n} \cdot \delta\boldsymbol{v} \right] d\Gamma - \delta\lambda \left(\sum_{i=1}^{N} \int_{\Gamma_i} \boldsymbol{v} \cdot \boldsymbol{n} \ d\Gamma \right) = 0 \tag{7.9}$$

The above equation should be satisfied for all variations of $\boldsymbol{v}, \lambda$. Further, the variations in these quantities are independent of each other. For this to be true, the following system of equations must be satisfied,

$$\sum_{i=1}^{N} \int_{\Gamma_i} (\boldsymbol{v} - \lambda \boldsymbol{n}) \cdot \delta\boldsymbol{v} \ d\Gamma - \sum_{i=1}^{N} \int_{\Gamma_i} \boldsymbol{v}_0 \cdot \delta\boldsymbol{v} \ d\Gamma = 0 \tag{7.10}$$

$$\sum_{i=1}^{N} \int_{\Gamma_i} \boldsymbol{v} \cdot \boldsymbol{n} \ d\Gamma = 0 \tag{7.11}$$

A parametric representation is used for the enrichments to discretize this system. For any particular enrichment Γ_i, each control point is associated with a nodal velocity. The velocity at any point on the enrichment can be written in terms of the shape functions, $[\boldsymbol{N}]$ as, $\boldsymbol{v} = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix}^{\mathrm{T}}$, with $v_x = \begin{bmatrix} \boldsymbol{N} \end{bmatrix} \begin{Bmatrix} \boldsymbol{v}_x \end{Bmatrix}$. Similarly,

$\delta \boldsymbol{v} = \begin{bmatrix} \delta v_x & \delta v_y & \delta v_z \end{bmatrix}^{\mathrm{T}}$, with $\delta v_x = \begin{bmatrix} \boldsymbol{N} \end{bmatrix} \begin{Bmatrix} \delta \boldsymbol{v}_x \end{Bmatrix}$. Here, $\begin{Bmatrix} \boldsymbol{v}_x \end{Bmatrix}$ and $\begin{Bmatrix} \delta \boldsymbol{v}_x \end{Bmatrix}$ denote the velocities and variations, repsectively, at the control points of the enrichment Γ_i. $[\boldsymbol{N}]$ is a row matrix of shape functions, and has been chosen to be NURBS in this thesis. This gives the dot product,

$$\boldsymbol{v} \cdot \delta \boldsymbol{v} = \begin{bmatrix} \delta \boldsymbol{v}_x^T & \delta \boldsymbol{v}_y^T & \delta \boldsymbol{v}_z^T \end{bmatrix} \begin{bmatrix} \boldsymbol{N}^T \boldsymbol{N} & 0 & 0 \\ 0 & \boldsymbol{N}^T \boldsymbol{N} & 0 \\ 0 & 0 & \boldsymbol{N}^T \boldsymbol{N} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{bmatrix} \tag{7.12}$$

We define symmetric positive definite matrices K_i for each enrichment, and a global matrix K as,

$$K_i = \int_{\Gamma_i} [\boldsymbol{N}]^T [\boldsymbol{N}] \; d\Gamma$$

$$K = \begin{bmatrix} K_1 & 0 & \dots & 0 \\ 0 & K_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & K_N \end{bmatrix}$$

Using this with Eq. (7.12) gives,

$$\sum_{i=1}^{N} \int_{\Gamma_i} \boldsymbol{v} \cdot \delta \boldsymbol{v} \; d\Gamma = \begin{bmatrix} \delta \boldsymbol{v}_x^T & \delta \boldsymbol{v}_y^T & \delta \boldsymbol{v}_z^T \end{bmatrix} \begin{bmatrix} K & 0 & 0 \\ 0 & K & 0 \\ 0 & 0 & K \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{bmatrix} \tag{7.13}$$

Similarly, we introduce quantities representing the normal and $\boldsymbol{v}_0$ terms in Eq. (7.10) for each enrichment,

$$\boldsymbol{s}_{x_i} = \int_{\Gamma_i} [\boldsymbol{N}]^T n_x \; d\Gamma \tag{7.14}$$

$$\boldsymbol{f}_{x_i} = \int_{\Gamma_i} [\boldsymbol{N}]^T v_{x_0}^i \; d\Gamma \tag{7.15}$$

The introduction of these quantities allows us to write Eq. (7.10) in a matrix form as,

$$\begin{aligned}\begin{bmatrix}\delta\boldsymbol{v}_x^T & \delta\boldsymbol{v}_y^T & \delta\boldsymbol{v}_z^T\end{bmatrix}\left(\begin{bmatrix}K & 0 & 0\\ 0 & K & 0\\ 0 & 0 & K\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_x\\ \boldsymbol{v}_y\\ \boldsymbol{v}_z\end{bmatrix}-\lambda\begin{bmatrix}\boldsymbol{s}_x\\ \boldsymbol{s}_y\\ \boldsymbol{s}_z\end{bmatrix}\right)\\ =\begin{bmatrix}\delta\boldsymbol{v}_x^T & \delta\boldsymbol{v}_y^T & \delta\boldsymbol{v}_z^T\end{bmatrix}\begin{bmatrix}\boldsymbol{f}_x\\ \boldsymbol{f}_y\\ \boldsymbol{f}_z\end{bmatrix}\end{aligned} \tag{7.16}$$

Taking into account the fact that the variations $\delta\boldsymbol{v}$ are arbitrary gives the linear system of equations,

$$\begin{bmatrix}K & 0 & 0 & -\boldsymbol{s}_x\\ 0 & K & 0 & -\boldsymbol{s}_y\\ 0 & 0 & K & -\boldsymbol{s}_z\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_x\\ \boldsymbol{v}_y\\ \boldsymbol{v}_z\\ \lambda\end{bmatrix}=\begin{bmatrix}\boldsymbol{f}_x\\ \boldsymbol{f}_y\\ \boldsymbol{f}_z\end{bmatrix} \tag{7.17}$$

The volume conservation equation, Eq. (7.11) has not been included yet. This constraint can be written in terms of the newly defined quantities as,

$$\begin{bmatrix}\boldsymbol{s}_x^T & \boldsymbol{s}_y^T & \boldsymbol{s}_z^T\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_x\\ \boldsymbol{v}_y\\ \boldsymbol{v}_z\end{bmatrix}=\begin{bmatrix}-\boldsymbol{s}_x^T & -\boldsymbol{s}_y^T & -\boldsymbol{s}_z^T\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_x\\ \boldsymbol{v}_y\\ \boldsymbol{v}_z\end{bmatrix}=0 \tag{7.18}$$

Adding this to Eq. (7.17) gives the final symmetric, positive semi-definite system to be solved at each time step,

$$\begin{bmatrix}K & 0 & 0 & -\boldsymbol{s}_x\\ 0 & K & 0 & -\boldsymbol{s}_y\\ 0 & 0 & K & -\boldsymbol{s}_z\\ -\boldsymbol{s}_x^T & -\boldsymbol{s}_y^T & -\boldsymbol{s}_z^T & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_x\\ \boldsymbol{v}_y\\ \boldsymbol{v}_z\\ \lambda\end{bmatrix}=\begin{bmatrix}\boldsymbol{f}_x\\ \boldsymbol{f}_y\\ \boldsymbol{f}_z\\ 0\end{bmatrix} \tag{7.19}$$

At each time step, this linear system is solved to obtain the control point velocities to evolve the bubbles. Since the integrals involved in the assembly process are all surface integrals, there is no underlying domain required. The integration is performed numerically using 3-point Gauss quadrature for each enrichment. For coalescent bubbles, only the external regions of the bubble surface contribute to the integrals. In other words, quadrature points in an enrichment that lie inside other bubbles do not contribute towards the integrals. For each enrichment, the Boolean union of level sets from all other bubbles is used to determine and ignore such quadrature points. Due to the use of the Boolean union, the coalescence is handled automatically without the need to detect and compute surface-surface intersections. Standard quadrature is however not sufficiently accurate in the regions where the bubble surfaces intersect. To improve the accuracy adapative quadrature based on k-D trees, proposed in [89], is used near such intersection regions. Quadrature cells that are intersected by bubble boundaries are refined into smaller cells iteratively, with signed algebraic level sets used to detect such quadrature cells. Forward Euler time stepping is used in the simulations, and the total volume of the bubbles computed at each time step. The step length at each time step is determined through the CFL condition Eq. (6.8).

The total volume of the bubbles is computed as the sum of individual volumes. Volume of each bubble is computed by the surface integral,

$$V_i = \int_\Omega d\Omega = \int_{\Gamma_i} \boldsymbol{x} \cdot \boldsymbol{n} \; d\Gamma \tag{7.20}$$

Quadrature points that are inside other bubbles are neglected during numerical integration. When a bubble is completely absorbed inside another bubble, then its volume contribution is zero. When this contribution is below a specified tolerance, the bubble is assumed to have been absorbed and is removed from the system.

7.3.2 Results

A system with two bubbles is first considered. The bubbles are initially both spherical with radii 0.0375 units and 0.075 units, respectively. The smaller bubble is given a constant upward velocity of 0.014 units, while the larger bubble is slower with an upward velocity of 0.002 units. The bubbles are initially separated by a gap of 0.01 units. The simulations took about 53 mins for 800 time steps on a desktop computer, at an average of about 4s per time step. Snapshots of the bubbles at different time steps are shown in Fig. 7.7, while the variation of the total volume and corresponding error are shown in Fig. 7.8. The error is initially very low and rises up to 17.3% during coalescence. Once the smaller bubble has been completely absorbed, the error falls back to very low values.

(a) (b) (c) (d)

Figure 7.7. : Snapshots of the bubbles as both bubbles float at (a) initial time step (b) $t = 2.66$ (c) $t = 5.36$ (d) $t = 17.44$. The smaller bubble has been completely absorbed into the larger bubble.

Next, a system with 7 bubbles of various sizes is considered. The bubbles are initially spherical with radii ranging from 0.03 units to 0.085 units. They are provided a size-dependent upward velocity given by,

$$v_i = 0.0009/r_i - 0.01 \tag{7.21}$$

where, $2r_i$ is the maximum dimension of the i^{th}bubble along the Cartesian axes. The simulations took about 9hrs 40 mins for 1200 time steps on a desktop computer,

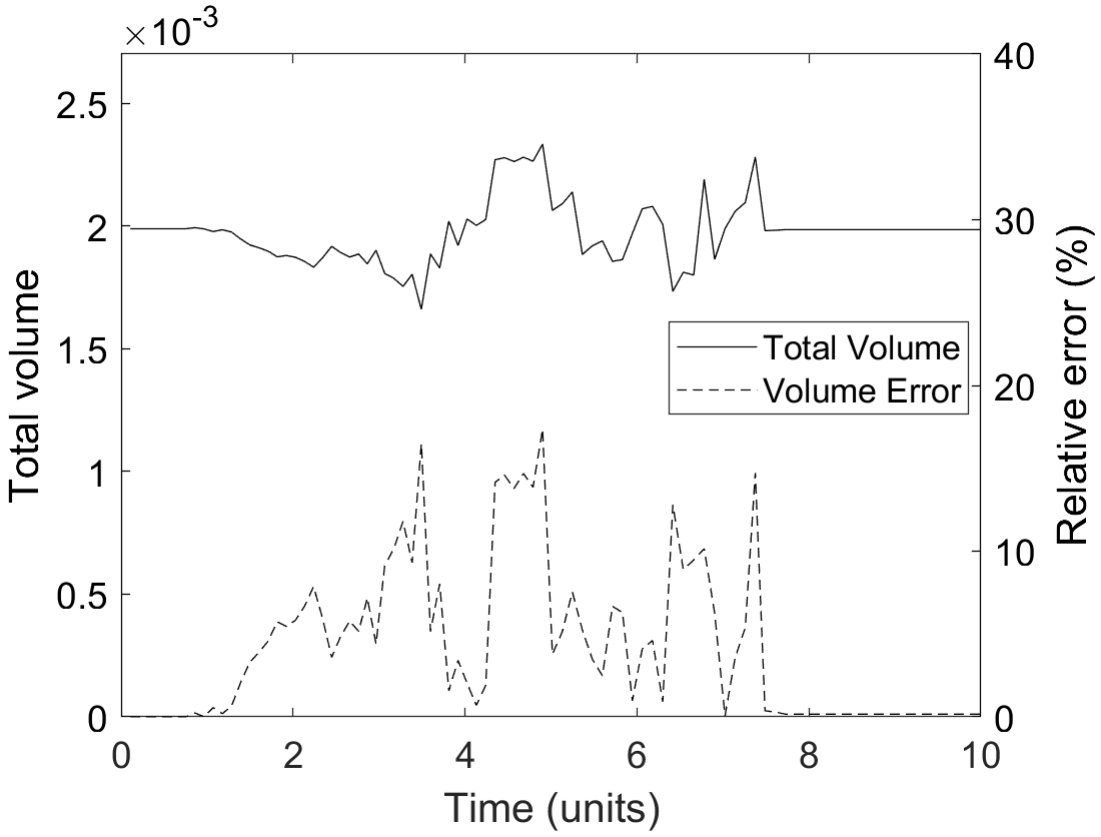
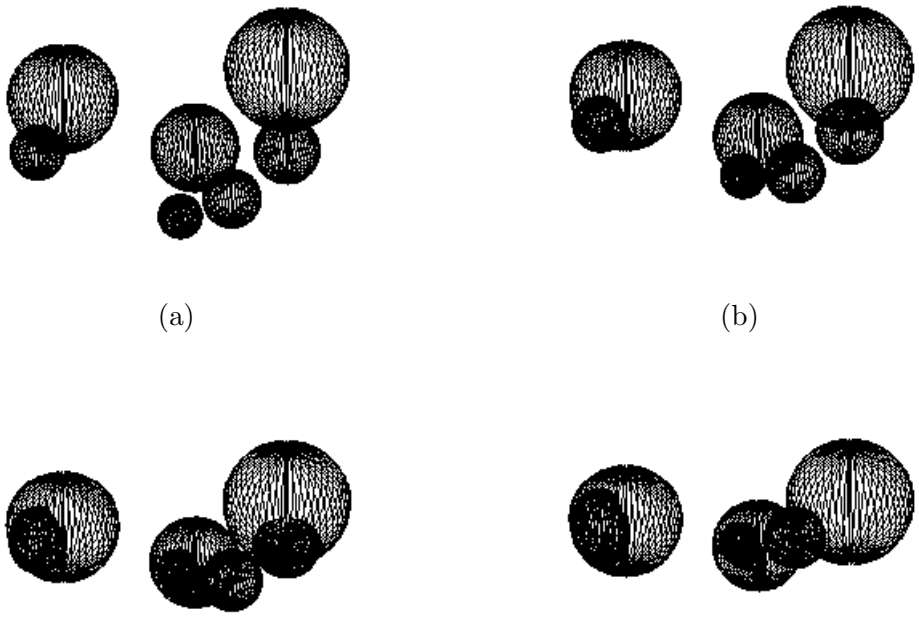


Figure 7.8. : Variation of volume with each time step for the system with two bubbles. The total volume error is initially low, before rising to 17.35% during coalescence, and falling back to very low values after complete absorption of the smaller bubble.

at an average of about half a minute per time step. Snapshots of the bubbles at different time steps are shown in Fig. 7.9, while the variation of the total volume and corresponding are shown in Fig. 7.10. The maximum error in the total volume is 13.06%.



(a) (b)

(c) (d)

Figure 7.9. : Snapshots of the bubbles at (a) initial time step (b) $t = 3.39$ (c) $t = 7.27$ (d) $t = 13.15$. One of the bubbles have been completely absorbed, while some have nearly been.

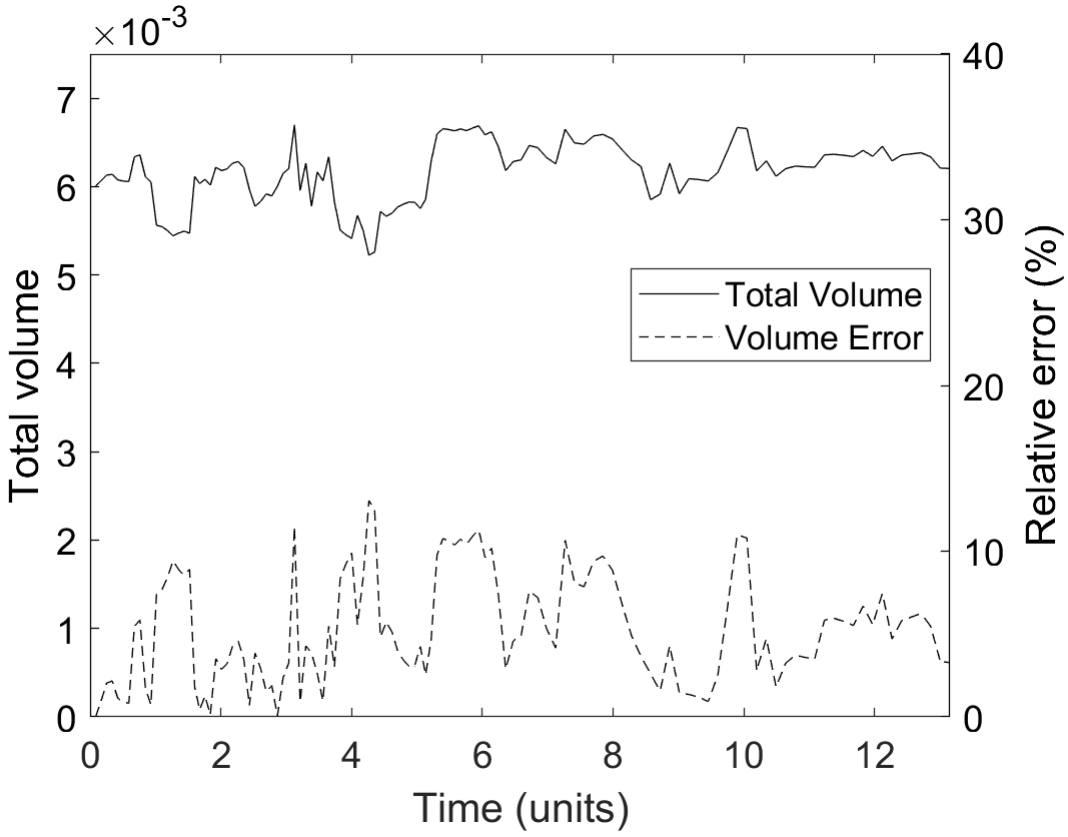


Figure 7.10. : Variation of volume with each time step for the system with 7 bubbles. The total volume error is 13.06%

8. CLOSURE

This work explores the analytical and numerical modeling of phase evolution problems for systems with multiple diffusing species. Generalized criteria for phase nucleation and growth were derived from first principles of thermodynamics. A pillbox procedure was followed to form balance laws of mass, momentum and energy as well as the entropy inequality, at a phase interface. The interface second law conditions provided a configurational force associated with the motion of the interface. The derived configurational force naturally extends the Eshelby energy momentum tensor to problems with multiple diffusing species and arbitrary surface stress. This configurational force was used to develop generalized conditions that dictate the growth and nucleation of a phase in a body with multiple diffusing species. These conditions are also applicable to bodies with finite deformation and arbitrary surface stress. Notably, no constitutive relations for the surface and bulk diffusional flux were assumed in deriving these generalized conditions. A critical material-dependent energy density was developed, which must be overcome for phase nucleation to occur. As an illustration, simulations of Blech's experiments were used to estimate this critical energy density for a Al-TiN interface as 5.5 Joules/cm^3.

Numerical modeling of phase evolution was performed using an explicit interface technique called enriched isogeometric analysis. In this analysis approach, the behavioral field solution is given as a weighted blending of the underlying solution and the interface solution. In order to restrict the influence of an interface to its neighborhood, a distance-based weight field was used. Signed algebraic level sets were used as a fast and efficient measure of distance for the weight field. The sign of these level sets helped classify points based on the side of the phase interface they lie in. These level sets were generated from the implicitization of the interface geometry using the Dixon resultant. While being robust for curves, the generation of these level sets was

found to fail for common three-dimensional surfaces. For several parametric surfaces including spheroids and cylinders, the Dixon resultant was identically zero. Such geometries could be directly implicitized using the Dixon resultant. In this work, a maximal-rank submatrix procedure was adopted to recover the resultant and thus the implicitization, for trivially singular Dixon matrices. Further, the Dixon resultant was found to be unsigned for some closed surfaces such as spheres. The sign of the Dixon resultant is important to classify points as lying inside or outside a given closed geometry. A polynomial square root procedure was developed in this work to extract the sign from unsigned Dixon resultants, completely algebraically. Rectification measures to [87] were also proposed for smooth algebraic point projection for surfaces with trivially singular or unsigned Dixon resultants. The developed procedure is compared with the M-Rep method developed by Busé [90]. Since the M-Rep method does not require measures for robustness, it was observed to have lower pre-processing times than algebraic level sets. However, distance estimation using the M-Rep method requires performing a SVD at each point of evaluation, and consequently, the M-Rep method was observed to be much slower than algebraic level sets in providing a measure of distance at a point. Furthermore, algebraic level sets are signed, unlike the M-Rep distance measure, and thus can trivially answer point containment queries. As an illustration, different three-dimensional electromigration problems were solved, including simulations of electromigration experiments on Copper-TiN line structures.

Moving boundary problems in general, and phase evolution problems in particular, can involve complex topologies. Since enriched isogeometric analysis (EIGA) is an explicit interface method, topological changes in phases pose geometric challenges. Conventionally, these would warrant computing intersections between phase boundaries, which could be computationally prohibitive to perform at each time step. Taking advantage of the fact that analysis in EIGA is based on algebraic level sets, a simple and efficient approach to model topological changes without introducing computational overhead was proposed in this work. Algebraic Boolean operations on signed level sets using R-functions were used to model coalescence of phases. Appli-

cation to various problems with multiple, interacting voids showed that the proposed method does not introduce any overhead when modeling systems with or without coalescence.

8.1 Future Work

The major avenue to extend the work in this thesis is to capture general topological changes. In addition, this thesis may be enhanced by using local refinement techniques and by improving the bounding box construction implementation. These shall now be discussed in brief.

8.1.1 Modeling Topological Changes

In physical problems, there are four topological changes that phases may undergo, viz. coalescence, splitting, nucleation and dissolution. Algebraic Boolean compositions on signed level sets were introduced in this work to model coalescence of phases efficiently. As future work, methods and techniques shall have to be developed to accommodate the other topological changes. Nucleation and dissolution may be approximated by assuming a minimum size condition for a phase, possibly derived from Eq. (2.82). Another possibility is to use a statistical approach, where the probability of nucleation Eq. (2.84) can be used to introduce nuclei stochastically. The splitting of a phase into two would require detection of self-intersection of the phase boundary and using the self-intersection to split the phase into multiple geometries.

8.1.2 Local Refinement

In this thesis, all simulations were conducted with a Non-Uniform Rational B-Spline (NURBS) representation for the void or phase interface. A disadvantage of using a NURBS representation is that it does not allow local refinement. Since NURBS is a tensor-product representation, knot insertion is a global operation and thus re-

finement cannot be confined to a region of interest. This leads to an unnecessarily large number of degrees of freedom and consequently longer run times. To alleviate this alternative representations such as THB-splines, which support local refinement, may be used. Using such representations will reduce the number of degrees of freedom and thus the computation time.

8.1.3 Native Implementation of Convex Hull

The bounding box procedure used to generate signed algebraic level sets requires computing the convex hull of the control points of the parametric curve of surface. In this thesis a C++ library, QHull [133], is used for this operation. It is observed that communicating the control point coordinates with the library functions introduces numerical error. The coordinates obtained from the computed convex hulls do not match exactly with the original coordinates. This causes the constructed bounding box to contain open slivers. To avoid this, a convex hull algorithm should be implemented natively. This would ensure that the constructed bounding box is perfectly closed and without gaps.

REFERENCES

REFERENCES

[1] Lingyun Zhang, Shengquan Ou, Joanne Huang, K. N. Tu, Stephen Gee, and Luu Nguyen. Effect of Current Crowding on Void Propagation at the Interface between Intermetallic Compound and Solder in Flip Chip Solder joints. *Applied Physics Letters*, 88(1):012106, 2006.

[2] V Alexiades and AD Solomon. Mathematical modeling of melting and freezing processes, 323 pp. *Hemisphere Publishing Corporation, Washington. JAPEX/JNOC/GSC Mallik 2L-38 Gas Hydrate Research Well, Mackenzie Delta, Northwest Territories, Canada, edited by SR Dallimore, T. Uchida, and TS Collett, Bulletin*, 544:357–376, 1993.

[3] RJ Gleixner and WD Nix. A physically based model of electromigration and stress-induced void formation in microelectronic interconnects. *Journal of Applied Physics*, 86(4):1932–1944, 1999.

[4] Valeriy Sukharev and Ehrenfried Zschech. A model for electromigration-induced degradation mechanisms in dual-inlaid copper interconnects: Effect of interface bonding strength. *Journal of Applied Physics*, 96(11):6337–6343, 2004.

[5] L Xia, Allan F Bower, Z Suo, and CF Shih. A finite element analysis of the motion and evolution of voids due to strain and electromigration induced surface diffusion. *Journal of the Mechanics and Physics of Solids*, 45(9):1473–1493, 1997.

[6] Cemal Basaran, Minghui Lin, and Hua Ye. A thermodynamic model for electrical current induced damage. *International Journal of Solids and Structures*, 40(26):7315–7327, 2003.

[7] Rohan Abeyaratne and James K Knowles. On the driving traction acting on a surface of strain discontinuity in a continuum. *Journal of the Mechanics and Physics of Solids*, 38(3):345–360, 1990.

[8] Eliot Fried and Morton E Gurtin. Coherent solid-state phase transitions with atomic diffusion: A thermomechanical treatment. *Journal of Statistical Physics*, 95(5-6):1361–1427, 1999.

[9] Morton E Gurtin. *Configurational forces as basic concepts of continuum physics*, volume 137. Springer Science & Business Media, 1999.

[10] MW Lane, EG Liniger, and JR Lloyd. Relationship between interfacial adhesion and electromigration in cu metallization. *Journal of Applied Physics*, 93(3):1417–1421, 2003.

[11] Melvin Avrami. Kinetics of phase change. ii transformation-time relations for random distribution of nuclei. *The Journal of chemical physics*, 8(2):212–224, 1940.

[12] John W Cahn. The kinetics of grain boundary nucleated reactions. *Acta metallurgica*, 4(5):449–459, 1956.

[13] John Wyrill Christian. *The theory of transformations in metals and alloys*. Newnes, 2002.

[14] Ken Kelton and Alan Greer. *Nucleation in Condensed Matter: Applications in Materials and Biology*, volume 15, Pergamon Materials Series. Pergamon, 2010.

[15] M Shatzkes and JR Lloyd. A model for conductor failure considering diffusion concurrently with electromigration resulting in a current exponent of 2. *Journal of Applied Physics*, 59(11):3890–3893, 1986.

[16] JR Lloyd and J Kitchin. The electromigration failure distribution: The fine-line case. *Journal of applied physics*, 69(4):2117–2127, 1991.

[17] Brook Chao, Seung-Hyun Chae, Xuefeng Zhang, Kuan-Hsun Lu, Min Ding, Jay Im, and Paul S Ho. Electromigration enhanced intermetallic growth and void formation in pb-free solder joints. *Journal of applied physics*, 100(8):084909, 2006.

[18] KN Tu, CC Yeh, CY Liu, and Chih Chen. Effect of current crowding on vacancy diffusion and void formation in electromigration. *Applied Physics Letters*, 76(8):988–990, 2000.

[19] Everett CC Yeh, WJ Choi, KN Tu, Peter Elenius, and Haluk Balkan. Current-crowding-induced electromigration failure in flip chip solder joints. *Applied Physics Letters*, 80(4):580–582, 2002.

[20] RJ Gleixner, BM Clemens, and WD Nix. Void nucleation in passivated interconnect lines: Effects of site geometries, interfaces, and interface flaws. *Journal of Materials Research*, 12(08):2081–2090, 1997.

[21] JJ Clement and CV Thompson. Modeling electromigration-induced stress evolution in confined metal lines. *Journal of applied physics*, 78(2):900–904, 1995.

[22] R Kirchheim. Stress and electromigration in al-lines of integrated circuits. *Acta Metallurgica et Materialia*, 40(2):309–323, 1992.

[23] Allan F Bower and Sadasivan Shankar. A finite element model of electromigration induced void nucleation, growth and evolution in interconnects. *Modelling and Simulation in Materials Science and Engineering*, 15(8):923, 2007.

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

[25] Tanmay K Bhandakkar and Huajian Gao. Cohesive modeling of crack nucleation under diffusion induced stresses in a thin strip: Implications on the critical size for flaw tolerant battery electrodes. *International Journal of Solids and Structures*, 47(10):1424–1434, 2010.

[26] N Singh, AF Bower, and S Shankar. A three-dimensional model of electromigration and stress induced void nucleation in interconnect structures. *Modelling and Simulation in Materials Science and Engineering*, 18(6):065006, 2010.

[27] Cemal Basaran and Shihua Nie. An irreversible thermodynamics theory for damage mechanics of solids. *International Journal of Damage Mechanics*, 13(3):205–223, 2004.

[28] Wei Yao and Cemal Basaran. Computational damage mechanics of electromigration and thermomigration. *Journal of Applied Physics*, 114(10):103708, 2013.

[29] BM Clemens, WD Nix, and RJ Gleixner. Void nucleation on a contaminated patch. *Journal of materials research*, 12(8):2038–2042, 1997.

[30] Charles S Peskin. Flow patterns around heart valves: a numerical method. *Journal of computational physics*, 10(2):252–271, 1972.

[31] Rajat Mittal and Gianluca Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[32] Rainald Löhner, Juan R. Cebral, Fernando E. Camelli, S. Appanaboyina, Joseph D. Baum, Eric L. Mestreau, and Orlando A. Soto. Adaptive embedded and immersed unstructured grid techniques. *Computer Methods in Applied Mechanics and Engineering*, 197(25-28):2173–2197, 2008.

[33] P Neittaanmäki and D Tiba. An embedding of domains approach in free boundary problems and optimal design[J]. *SIAM journal on control and optimization*, 33(5):1587–1602, 1995.

[34] Lucy Zhang, Axel Gerstenberger, Xiaodong Wang, and Wing Kam Liu. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering*, 193(21-22):2051–2067, 2004.

[35] Isabelle Ramière, Philippe Angot, and Michel Belliard. A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 196(4-6):766–781, 2007.

[36] R. Glowinski and Yu Kuznetsov. Distributed Lagrange multipliers based on fictitious domain method for second order elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1498–1506, 2007.

[37] Jamshid Parvizian, Alexander Düster, and Ernst Rank. Finite cell method : hh- and p-extension for embedded domain problems in solid mechanics. *Computational Mechanics*, 41(1):121–133, 2007.

[38] Michael Freytag, Vadim Shapiro, and Igor Tsukanov. Finite element analysis in situ. *Finite Elements in Analysis and Design*, 47(9):957–972, 2011.

[39] Klaus Höllig, Christian Apprich, and Anja Streit. Introduction to the Web-method and its applications. *Advances in Computational Mathematics*, 23(1-2):215–237, 2005.

[40] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[41] James Albert Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, volume 3. Cambridge university press, 1999.

[42] T. J. R. Hughes, L. P. Fraca, and G. M. Hulbert. A new finite element formulation for computational fluid dynamics/viii, the galerkin least-sqaures method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73:173–189, 1989.

[43] Theofanis Strouboulis, Ivo Babuška, and Kevin Copps. The design and analysis of the generalized finite element method. *Computer methods in applied mechanics and engineering*, 181(1-3):43–69, 2000.

[44] Theofanis Strouboulis, Kevin Copps, and Ivo Babuška. The generalized finite element method. *Computer methods in applied mechanics and engineering*, 190(32-33):4081–4193, 2001.

[45] Ted Belytschko and Tom Black. Elastic crack growth in finite elements with minimal remeshing. *International journal for numerical methods in engineering*, 45(5):601–620, 1999.

[46] John Dolbow, Nicolas Moës, and Ted Belytschko. Discontinuous enrichment in finite elements with a partition of unity method. *Finite elements in analysis and design*, 36(3-4):235–260, 2000.

[47] N Sukumar, David 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, 2003.

[48] Ted Belytschko and Robert Gracie. On xfem applications to dislocations and interfaces. *International journal of plasticity*, 23(10-11):1721–1738, 2007.

[49] Jay Oswald, Robert Gracie, Roopam Khare, and Ted Belytschko. An extended finite element method for dislocations in complex geometries: Thin films and nanotubes. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1872–1886, 2009.

[50] Jack Chessa, Patrick Smolinski, and Ted Belytschko. The extended finite element method (xfem) for solidification problems. *International Journal for Numerical Methods in Engineering*, 53(8):1959–1977, 2002.

[51] John Dolbow, Eliot Fried, and Huidi Ji. A numerical strategy for investigating the kinetic response of stimulus-responsive hydrogels. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4447–4480, 2005.

[52] Ravindra Duddu, Stéphane Bordas, David Chopp, and Brian Moran. A combined extended finite element and level set method for biofilm growth. *International Journal for Numerical Methods in Engineering*, 74(5):848–870, 2008.

[53] William J Boettinger, James A Warren, Christoph Beckermann, and Alain Karma. Phase-field simulation of solidification. *Annual review of materials research*, 32(1):163–194, 2002.

[54] Michael J Borden, Clemens V Verhoosel, Michael A Scott, Thomas JR Hughes, and Chad M Landis. A phase-field description of dynamic brittle fracture. *Computer Methods in Applied Mechanics and Engineering*, 217:77–95, 2012.

[55] Michael J Borden, Thomas JR Hughes, Chad M Landis, and Clemens V Verhoosel. A higher-order phase-field model for brittle fracture: Formulation and analysis within the isogeometric analysis framework. *Computer Methods in Applied Mechanics and Engineering*, 273:100–118, 2014.

[56] Gunduz Caginalp. Solidification Problems as Systems of Nonlinear Differential Equations. *Lectures in Applied Mathematics*, 23:347–369, 1986.

[57] Alain Karma and Wouter-Jan Rappel. Phase-field method for computationally efficient modeling of solidification with arbitrary interface kinetics. *Physical review E*, 53(4):R3017, 1996.

[58] Alain Karma and Wouter-Jan Rappel. Quantitative phase-field modeling of dendritic growth in two and three dimensions. *Physical review E*, 57(4):4323, 1998.

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

[60] M. Cassale. *Integration of Geometric Analysis and Structural Analysis using Trimmed Patches*. PhD thesis, University of California, Irvine, 1989.

[61] P. Kagan and A. Fischer. Integrated mechanically based CAE systems using B-spline based finite elements. *Computer Aided Design*, 32, 2000.

[62] 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.

[63] F.P. Renken and G. Subbarayan. NURBS based solution to inverse boundary problems in droplet shape prediction. *Computer Methods in Applied Mechanics and Engineering*, 190:1391–1406, 2000.

[64] 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.

[65] Shankar Krishnan and Dinesh Manocha. An efficient surface intersection algorithm based on lower-dimensional formulation. *ACM Transactions on Graphics (TOG)*, 16(1):74–106, 1997.

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

[67] Antoine Legay, Jack Chessa, and Ted Belytschko. An eulerian–lagrangian method for fluid–structure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2070–2087, 2006.

[68] Thomas Rüberg and Fehmi Cirak. An immersed finite element method with integral equation correction. *International journal for numerical methods in engineering*, 86(1):93–114, 2011.

[69] Azriel Rosenfeld and John L. Pfaltz. Sequential operations in digital picture processing. *J. ACM*, 13(4):471–494, 1966.

[70] Elon Rimon and Stephen P. Boyd. Obstacle collision detection using best ellipsoid fit. *Journal of Intelligent and Robotic Systems*, 18(2):105–126, Feb 1997.

[71] Sebastian Wolff and Christian 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, 2013.

[72] Arnulph Fuhrmann, Gerrit Sobotka, and Clemens Groß. Distance fields for rapid collision detection in physically based modeling. In *Proceedings of GraphiCon 2003*, pages 58–65, 2003.

[73] 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, Nov 2001.

[74] T Belytschko, WJT 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, 2002.

[75] Julien Bruchon, Hugues Digonnet, and Thierry 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, 2009.

[76] Sarah F Frisken, Ronald N Perry, Alyn P Rockwood, and Thouis R Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 249–254. ACM Press/Addison-Wesley Publishing Co., 2000.

[77] Daniel Cohen-Or, Amira Solomovic, and David Levin. Three-dimensional distance field metamorphosis. *ACM Transactions on Graphics (TOG)*, 17(2):116–141, 1998.

[78] Xiao-Diao Chen, Jun-Hai Yong, Guozhao Wang, Jean-Claude Paul, and Gang Xu. Computing the minimum distance between a point and a NURBS curve. *Computer-Aided Design*, 40(10-11):1051–1054, 2008.

[79] Ying Liang Ma and W Terry Hewitt. Point inversion and projection for NURBS curve and surface: control polygon approach. *Computer Aided Geometric Design*, 20(2):79–99, 2003.

[80] Les Piegl and Wayne Tiller. *The NURBS Book*. Springer Science & Business Media, 2012.

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

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

[83] Arpan Biswas and Vadim Shapiro. Approximate distance fields with non-vanishing gradients. *Graphical Models*, 66(3):133–159, 2004.

[84] 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, 2014.

[85] K Upreti and G Subbarayan. Signed algebraic level sets on NURBS surfaces and implicit Boolean compositions for isogeometric CAD–CAE integration. *Computer-Aided Design*, 82:112–126, 2017.

[86] T. W. Sederberg. *Implicit and Parametric Curves and Surfaces for Computer Aided Geometric Design*. PhD thesis, Purdue University, 1983.

[87] Huanyu Liao, Pavan Kumar Vaitheeswaran, Tao Song, and Ganesh Subbarayan. Algebraic point projection for immersed boundary analysis on low degree nurbs curves and surfaces. *Algorithms*, 13(4):82, 2020.

[88] T Song, K Upreti, and G Subbarayan. A sharp interface isogeometric solution to the Stefan problem. *Computer Methods in Applied Mechanics and Engineering*, 284:556–582, 2015.

[89] Tao Song. *A sharp interface isogeometric strategy for moving boundary problems*. PhD thesis, Purdue University, 2016.

[90] Busé Laurent. Implicit matrix representations of rational Bézier curves and surfaces. *Computer-Aided Design*, 46:14–24, 2014.

[91] Yuvraj Singh. *Characterization of Interfacial Energy of Thin Films Through Current Induced Diffusive Interfacial Voiding*. PhD thesis, PhD Thesis submitted to Purdue University, 2019.

[92] David RH Jones and Michael F Ashby. *Engineering materials 2: an introduction to microstructures, processing and design*. Butterworth-Heinemann, 2005.

[93] J-M Tarascon and Michel Armand. Issues and challenges facing rechargeable lithium batteries. *Nature*, 414(6861):359–367, 2001.

[94] G Girishkumar, B McCloskey, AC Luntz, S Swanson, and W Wilcke. Lithium-air battery: promise and challenges. *The Journal of Physical Chemistry Letters*, 1(14):2193–2203, 2010.

[95] AJ Sunwoo, JW Morris, and GK Lucey. The growth of cu-sn intermetallics at a pretinned copper-solder interface. *Metallurgical Transactions A*, 23(4):1323–1332, 1992.

[96] Anirudh Udupa, Subramanya Sadasiva, and Ganesh Subbarayan. A framework for studying dynamics and stability of diffusive-reactive interfaces with application to cu6sn5 intermetallic compound growth. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 472(20160134), 2016.

[97] Martine Ben Amar and Alain Goriely. Growth and instability in elastic tissues. *Journal of the Mechanics and Physics of Solids*, 53(10):2284–2319, 2005.

[98] Xuesong Li, Carl W Magnuson, Archana Venugopal, Rudolf M Tromp, James B Hannon, Eric M Vogel, Luigi Colombo, and Rodney S Ruoff. Large-area graphene single crystals grown by low-pressure chemical vapor deposition of methane on copper. *Journal of the American Chemical Society*, 133(9):2816–2819, 2011.

[99] Weiqing Wang and Z Suo. Shape change of a pore in a stressed solid via surface diffusion motivated by surface and elastic energy variation. *Journal of the Mechanics and Physics of Solids*, 45(5):709–729, 1997.

[100] Clifford Truesdell and Richard Toupin. The classical field theories. In *Principles of classical mechanics and field theory/Prinzipien der Klassischen Mechanik und Feldtheorie*, pages 226–858. Springer, 1960.

[101] Morton E. Gurtin, Eliot Fried, and Lallit Anand. *The Mechanics and Thermodynamics of Continua*. Cambridge University Press, New York, NY, 2010.

[102] Morton E Gurtin and Michel E Jabbour. Interface evolution in three dimensions with curvature-dependent energy and surface diffusion: Interface-controlled evolution, phase transitions, epitaxial growth of elastic films. *Archive for Rational Mechanics and Analysis*, 163(3):171–208, 2002.

[103] JD Eshelby. The elastic energy-momentum tensor. *Journal of elasticity*, 5(3-4):321–335, 1975.

[104] James R Rice. A path independent integral and the approximate analysis of strain concentration by notches and cracks. *Journal of Applied Mechanics*, 35(2):379–386, 1968.

[105] L. D. Landau and E. M. Lifshitz. *Statistical physics 2nd ed*. Addison-Wesley Pub. Co., Reading, MA, 1969.

[106] IS Servi and D Turnbull. Thermodynamics and kinetics of precipitation in the copper-cobalt system. *Acta Metallurgica*, 14(2):161–169, 1966.

[107] FK LeGoues and HI Aaronson. Influence of crystallography upon critical nucleus shapes and kinetics of homogeneous fcc-fcc nucleation. comparisons between theory and experiment in cu-co alloys. *Acta Metallurgica*, 32(10):1855–1864, 1984.

[108] M Volmer and A Weber. Keimbildung in übersättigten gebilden. *Z. phys. Chem*, 119:277–301, 1926.

[109] D Turnbull and J Co Fisher. Rate of nucleation in condensed systems. *Journal of Chemical Physics*, 17(1):71–73, 1949.

[110] Illan A Blech. Electromigration in thin aluminum films on titanium nitride. *Journal of Applied Physics*, 47(4):1203–1208, 1976.

[111] 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, 2007.

[112] 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, 2010.

[113] Jens M Melenk and Ivo Babuška. The partition of unity finite element method: basic theory and applications. *Computer methods in applied mechanics and engineering*, 139(1-4):289–314, 1996.

[114] Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference*, pages 517–524. ACM, 1968.

[115] Joe J Monaghan. Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, 30(1):543–574, 1992.

[116] Ted Belytschko, Yury Krongauz, Daniel Organ, Mark Fleming, and Petr Krysl. Meshless methods: an overview and recent developments. *Computer methods in applied mechanics and engineering*, 139(1-4):3–47, 1996.

[117] Elaine Cohen, Tom Lyche, and Richard Riesenfeld. Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics. *Computer graphics and image processing*, 14(2):87–111, 1980.

[118] Wolfgang Boehm and Hartmut Prautzsch. The insertion algorithm. *Computer-Aided Design*, 17(2):58–59, 1985.

[119] AL Dixon. On a form of the eliminant of two quantics. *Proceedings of the London Mathematical Society*, 2(1):468–478, 1908.

[120] Arthur Cayley. On the theory of elimination. *Cambridge and Dublin Math. J*, 3:116–120, 1848.

[121] Busé Laurent. Implicit matrix representations of rational Bézier curves and surfaces. *Computer-Aided Design*, 46:14–24, 2014.

[122] Laurent Busé and Thang Luu Ba. Matrix-based Implicit Representations of Rational Algebraic Curves and Applications. *Computer Aided Geometric Design*, 27(9):681–699, 2010.

[123] Thomas W Sederberg and Falai Chen. Implicitization using moving curves and surfaces. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 301–308, 1995.

[124] Laurent Busé and Marc Chardin. Implicitizing rational hypersurfaces using approximation complexes. *Journal of Symbolic Computation*, 40(4-5):1150–1168, 2005.

[125] Nicolás Botbol. The implicit equation of a multigraded hypersurface. *Journal of Algebra*, 348(1):381–401, 2011.

[126] Vadim Shapiro. Theory of R-functions and applications: A Primer. Technical report, Cornell University, 1991.

[127] Vadim Shapiro and Igor Tsukanov. Implicit functions with guaranteed differential properties. In *Proceedings of the Fifth ACM symposium on Solid Modeling and Applications*, pages 258–269. ACM, 1999.

[128] Deepak Kapur, Tushar Saxena, and Lu Yang. Algebraic and geometric reasoning using Dixon resultants. In *Proceedings of the international symposium on Symbolic and algebraic computation*, pages 99–107. ACM, 1994.

[129] Deepak Kapur and Tushar Saxena. Comparison of various multivariate resultant formulations. In *Proceedings of the 1995 international symposium on Symbolic and algebraic computation*, pages 187–194. ACM, 1995.

[130] Åke Björck and Sven Hammarling. A Schur method for the square root of a matrix. *Linear algebra and its applications*, 52:127–140, 1983.

[131] R W. Balluffi, S. Allen, and W C. Carter. *Kinetics of Materials*. Wiley, 2005.

[132] KN Tu. Recent advances on electromigration in very-large-scale-integration of interconnects. *Journal of Applied Physics*, 94(9):5451–5473, 2003.

[133] C Bradford Barber, David P Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.

[134] ME Gurtin, A Struthers, and WO Williams. A transport theorem for moving interfaces. *Quarterly of Applied Mathematics*, 47(4):773–777, 1989.

APPENDICES

A. SURFACE IDENTITIES

Following the definitions given in [102], the surface gradient is defined using the projection tensor:

$$\mathbf{P} = \mathbf{I} - \boldsymbol{n} \otimes \boldsymbol{n} \tag{A.1}$$

where, $\mathbf{I}$ is the identity tensor and $\boldsymbol{n}$ is the normal to the surface at the point of interest. Referring to Figure 2.2, for a scalar field $\phi_{\mathrm{s}}(\boldsymbol{x}_{\mathrm{s}})$ defined on the subregion Γ_{s} of Γ, the surface gradient operator is defined to relate the gradient of the field in its extension into the body as

$$\nabla_{\mathrm{s}}\, \phi_{\mathrm{s}} = \mathbf{P} \nabla \phi_{\mathrm{s}} \tag{A.2}$$

where, the quantity $\nabla_{\mathrm{s}}\, \phi_{\mathrm{s}}$ is defined on the tangent plane at the point of interest. Similarly, for a vector field $\boldsymbol{a}(\boldsymbol{x}_{\mathrm{s}})$ defined on the surface Γ_{s}, the surface gradient operator and the surface divergence operators are defined as

$$\nabla_{\mathrm{s}}\, \boldsymbol{a} = \mathbf{P} \nabla \boldsymbol{a}, \quad \nabla_{\mathrm{s}} \cdot \boldsymbol{a} = \mathrm{tr}\, (\nabla_{\mathrm{s}}\, \boldsymbol{a}) = \mathbf{P} : \nabla \boldsymbol{a} \tag{A.3}$$

Finally, for a second-order tensor field $\mathbf{A}(\boldsymbol{x}_{\mathrm{s}})$ defined on Γ_{s}, the two surface operators are similarly defined as,

$$\nabla_{\mathrm{s}}\, \mathbf{A} = \mathbf{P} \nabla \mathbf{A}, \quad \nabla_{\mathrm{s}} \cdot \mathbf{A} = \mathrm{tr}\, (\nabla_{\mathrm{s}}\, \mathbf{A}) = \mathbf{P} : \nabla \mathbf{A} \tag{A.4}$$

The curvature tensor at any point on the surface and the total curvature are defined using the surface gradient as

$$\mathbf{L} = -\nabla_{\mathrm{s}}\, \boldsymbol{n}, \quad \kappa = \mathrm{tr}\, (\mathbf{L}) = -\nabla_{\mathrm{s}} \cdot \boldsymbol{n} \tag{A.5}$$

where, κ is the total curvature, or twice the mean value. The curvature tensor is fully tangential and symmetric. Further, from the above relationship, it is easy to show:

$$\nabla_{\mathrm{s}} \cdot \mathbf{P} = \kappa \boldsymbol{n} \tag{A.6}$$

This gives a useful surface divergence product rule for any second order tensor field $\mathbf{A}(\boldsymbol{x}_{\mathrm{s}})$,

$$\begin{aligned} \nabla_{\mathrm{s}} \cdot (\mathbf{P}\mathbf{A}) &= (\nabla_{\mathrm{s}} \cdot \mathbf{P}) \cdot \mathbf{A} + \mathbf{P} : \nabla \mathbf{A} \\ &= \kappa \boldsymbol{n} \cdot \mathbf{A} + \nabla_{\mathrm{s}} \cdot \mathbf{A} \end{aligned} \tag{A.7}$$

We list next an identity that is useful for simplifications carried out in this thesis. For a smooth scalar and vector fields $\phi(\boldsymbol{x}_{\mathrm{s}})$ and $\boldsymbol{a}(\boldsymbol{x}_{\mathrm{s}})$ respectively,

$$\nabla_{\mathrm{s}} \cdot \phi_{\mathrm{s}} \boldsymbol{g} = \phi_{\mathrm{s}} \nabla_{\mathrm{s}} \cdot \boldsymbol{g} + \boldsymbol{g} \cdot \nabla_{\mathrm{s}} \phi_{\mathrm{s}} \tag{A.8}$$

The surface divergence theorem for a *tangential* vector field $\boldsymbol{a_t}(\boldsymbol{x}_{\mathrm{s}})$ defined on a sub-region Γ_{s} is:

$$\int_{\Gamma_{\mathrm{s}}} \nabla_{\mathrm{s}} \cdot \boldsymbol{a_t} \, d\Gamma_{\mathrm{s}} = \oint_{\partial\Gamma_{\mathrm{s}}} \boldsymbol{m} \cdot \boldsymbol{a_t} \, dc \tag{A.9}$$

where, $\boldsymbol{m}$ is tangent to the surface, but normal to the bounding curve $\partial\Gamma_{\mathrm{s}}$. For a *superficial* vector field with both normal and tangential components of the form $\boldsymbol{a} = a_n \boldsymbol{n} + \boldsymbol{a_t}$ such that $a_n = \boldsymbol{a} \cdot \boldsymbol{n}$, the above surface divergence theorem can be generalized using Equation (A.8) and Equation (A.5) as

$$\int_{\Gamma_{\mathrm{s}}} \nabla_{\mathrm{s}} \cdot \boldsymbol{a} \, d\Gamma_{\mathrm{s}} = -\int_{\Gamma_{\mathrm{s}}} \kappa \boldsymbol{n} \cdot \boldsymbol{a} \, d\Gamma_{\mathrm{s}} + \oint_{\partial\Gamma_{\mathrm{s}}} \boldsymbol{m} \cdot \boldsymbol{a} \, dc \tag{A.10}$$

Similarly for a *superficial* tensor field $\mathbf{A}(\boldsymbol{x}_{\mathrm{s}})$, the divergence theorem has the form,

$$\int_{\Gamma_{\mathrm{s}}} \nabla_{\mathrm{s}} \cdot \mathbf{A} \, d\Gamma_{\mathrm{s}} = -\int_{\Gamma_{\mathrm{s}}} \kappa \boldsymbol{n} \cdot \mathbf{A} \, d\Gamma_{\mathrm{s}} + \oint_{\partial\Gamma_{\mathrm{s}}} \boldsymbol{m} \cdot \mathbf{A} \, dc \tag{A.11}$$

B. MATERIAL TIME DERIVATIVE OF A BULK FIELD

Given a field $\phi(\boldsymbol{x}, t)$, the material time derivative of the field is defined as [101]

$$\dot{\phi} = \frac{\partial \phi}{\partial t} + \boldsymbol{v} \cdot \nabla \phi \tag{B.1}$$

Thus,

$$\begin{aligned} \frac{d}{dt} \int_\Omega \phi \, d\Omega &= \int_\Omega \dot{\phi} \, d\Omega + \int_\Omega \phi \, \dot{\overline{d\Omega}} \\ &= \int_\Omega \left(\frac{\partial \phi}{\partial t} + \boldsymbol{v} \cdot \nabla \phi \right) \, d\Omega + \int_\Omega \phi \nabla \cdot \boldsymbol{v} \, d\Omega \\ &= \int_\Omega \frac{\partial \phi}{\partial t} \, d\Omega + \int_\Omega \nabla \cdot \phi \boldsymbol{v} \, d\Omega \\ &= \int_\Omega \frac{\partial \phi}{\partial t} \, d\Omega + \int_{\partial \Omega} \phi \boldsymbol{v} \cdot \boldsymbol{n} \, d\Gamma_{\mathrm{s}} \end{aligned} \tag{B.2}$$

C. SURFACE TRANSPORT THEOREM

The surface transport theorem then gives the rate of change of a superficial scalar field, $\phi_{\mathrm{s}}(\boldsymbol{x}_{\mathrm{s}}(t), t)$, defined on the interface [134]. The surface transport theorem can be derived using the concept of material time derivative:

$$\frac{d}{dt}\int_{\Gamma_{\mathrm{s}}} \phi_{\mathrm{s}}\, d\Gamma_{\mathrm{s}} = \int_{\Gamma_{\mathrm{s}}} \dot{\phi}_{\mathrm{s}}\, d\Gamma_{\mathrm{s}} + \int_{\Gamma_{\mathrm{s}}} \phi_{\mathrm{s}}\, \dot{\overline{d\Gamma_{\mathrm{s}}}} \tag{C.1}$$

By definition of material time derivative:

$$\dot{\phi}_{\mathrm{s}}\left(\boldsymbol{x}_{\mathrm{s}}\left(t\right), t\right) = \frac{\partial \phi_{\mathrm{s}}}{\partial t} + \frac{\partial \phi_{\mathrm{s}}}{\partial \boldsymbol{x}_{\mathrm{s}}} \cdot \frac{d\boldsymbol{x}_{\mathrm{s}}}{dt} \tag{C.2}$$

$$= \frac{\partial \phi_{\mathrm{s}}}{\partial t} + \boldsymbol{v}_{\mathrm{s}} \cdot \nabla \phi_{\mathrm{s}} \tag{C.3}$$

Now, the material time derivative of the differential surface element is [101]

$$\dot{\overline{d\Gamma_{\mathrm{s}}}} = \left(\nabla \cdot \boldsymbol{v}_{\mathrm{s}} - \boldsymbol{n} \cdot \nabla \boldsymbol{v}_{\mathrm{s}} \boldsymbol{n}\right) d\Gamma_{\mathrm{s}} = \mathbf{P} : \nabla \boldsymbol{v}_{\mathrm{s}} d\Gamma_{\mathrm{s}} = \nabla_{\mathrm{s}} \cdot \boldsymbol{v}_{\mathrm{s}} d\Gamma_{\mathrm{s}} \tag{C.4}$$

Thus,

$$\frac{d}{dt}\int_{\Gamma_{\mathrm{s}}} \phi_{\mathrm{s}}\, d\Gamma_{\mathrm{s}} = \int_{\Gamma_{\mathrm{s}}} \left(\dot{\phi}_{\mathrm{s}} + \phi_{\mathrm{s}} \nabla_{\mathrm{s}} \cdot \boldsymbol{v}_{\mathrm{s}}\right)\, d\Gamma_{\mathrm{s}} \tag{C.5}$$

D. KINEMATICS OF A COHERENT INTERFACE

The velocity of a particle at a spatial location $\boldsymbol{x}(\boldsymbol{X}, t)$ obtained by holding position in the reference configuration $\boldsymbol{X}$ fixed, is defined as,

$$\boldsymbol{v} = \frac{\partial}{\partial t}\boldsymbol{x}(\boldsymbol{X}, t) \tag{D.1}$$

Now, it is assumed that the interface convects with the body, and that different material particles come to occupy the interface at different instants of time. Hence, the interface can be viewed as evolving with time in both the reference and current configurations. Let $\boldsymbol{X}_{\mathrm{S}}(t)$ denote the reference coordinate of particles on the interface at time t. Then, the interface velocity as viewed in the reference configuration is given by [102],

$$\boldsymbol{V}_{\mathrm{S}} = \frac{d}{dt}\boldsymbol{X}_{\mathrm{S}}(t) \tag{D.2}$$

Similarly, the velocity of the interface in the current configuration is,

$$\begin{aligned} \boldsymbol{v}_{\mathrm{s}} &= \frac{d}{dt}\boldsymbol{x}_{\mathrm{s}}(\boldsymbol{X}_{\mathrm{S}}(t), t) \\ &= \frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial t} + \frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial \boldsymbol{X}_{\mathrm{S}}} \cdot \frac{d\boldsymbol{X}_{\mathrm{S}}}{dt} \end{aligned} \tag{D.3}$$

The first term $\boldsymbol{v} = \frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial t}$ represents the velocity of a material point currently at the interface, $\boldsymbol{F} = \frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial \boldsymbol{X}_{\mathrm{S}}}$ is the instantaneous deformation gradient at the spatial location $\boldsymbol{x}_{\mathrm{s}}$ with its corresponding reference location $\boldsymbol{X}_{\mathrm{S}}$. The subscripts on $\boldsymbol{x}_{\mathrm{s}}$ and $\boldsymbol{X}_{\mathrm{S}}$ in the definition of $\boldsymbol{F}$ serve to remind the fact that these material points currently reside on the interface, but are free to change in any direction. The second term represents the contribution due to interfacial motion in the reference configuration, since by Eq. (D.2), $\boldsymbol{V}_{\mathrm{S}} = \frac{d\boldsymbol{X}_{\mathrm{S}}}{dt}$ is the velocity of the interface in the reference configuration.

Thus, the difference between the velocity of the interface and the velocity of a particle currently at the interface is,

$$\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v} = \boldsymbol{F}\boldsymbol{V}_{\mathrm{S}} \tag{D.4}$$

The inverse relationship is thus,

$$\boldsymbol{V}_{\mathrm{S}} = \boldsymbol{G}\left(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}\right) \tag{D.5}$$

where, $\boldsymbol{G} = \boldsymbol{F}^{-1} = \frac{\partial \boldsymbol{X}_{\mathrm{S}}}{\partial \boldsymbol{x}_{\mathrm{s}}}$ is the inverse of the deformation gradient at a point on the interface. Furthermore, for a coherent interface, in both the reference and current configurations, at all times, the following conditions must be satisfied

$$\begin{aligned} [\![\boldsymbol{v}_{\mathrm{s}}]\!] &= 0 \\ [\![\boldsymbol{V}_{\mathrm{S}}]\!] &= 0 \end{aligned} \tag{D.6}$$

This gives a relation for the jump in the bulk velocity across the interface,

$$[\![\boldsymbol{v}]\!] = -\,[\![\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}]\!] = -\,[\![\boldsymbol{F}]\!]\,\boldsymbol{V}_{\mathrm{S}} \tag{D.7}$$

where, we have used $[\![\boldsymbol{V}_{\mathrm{S}}]\!] = 0$. Although not done here, the above condition is sometimes further reduced by assuming that the deformation gradient jump is non-zero only in the normal direction [7]. That is,

$$[\![\boldsymbol{v}]\!] = -\,[\![\boldsymbol{F}]\!]\,\boldsymbol{N}V_{\mathrm{S_N}} \tag{D.8}$$

$$0 = -\,[\![\boldsymbol{F}]\!]\,\boldsymbol{V}_{\mathbf{S_T}} \tag{D.9}$$

where, $V_{\mathrm{S_N}}$ is the normal component of the reference interface velocity, $V_{\mathrm{S_N}} = \boldsymbol{N} \cdot \boldsymbol{V}_{\mathrm{S}}$ and $\boldsymbol{V}_{\mathbf{S_T}}$ is the tangential component of the reference interface velocity.

The surface gradient and surface divergence of the interface velocity are next derived. Applying the surface gradient operation on Eq. (D.4), we get

$$\nabla_{\mathrm{s}}\, \boldsymbol{v}_{\mathrm{s}} = \mathbf{P}\nabla \boldsymbol{v} + \mathbf{P}\boldsymbol{G}^{\mathrm{T}} \left[\nabla_0 \boldsymbol{F}\ \boldsymbol{V}_{\mathrm{S}} + \nabla_0 \boldsymbol{V}_{\mathrm{S}}\ \boldsymbol{F}^{\mathrm{T}}\right] \tag{D.10}$$

Observing that $\frac{\partial}{\partial \boldsymbol{X}_{\mathrm{S}}}\left(\frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial t}\right) = \frac{\partial}{\partial t}\left(\frac{\partial \boldsymbol{x}_{\mathrm{s}}}{\partial \boldsymbol{X}_{\mathrm{S}}}\right)$, the above expression can be rewritten as

$$\left(\nabla_{\mathrm{s}}\, \boldsymbol{v}_{\mathrm{s}}\right)^{\mathrm{T}} = \overset{\triangle}{\boldsymbol{F}}\boldsymbol{G}\mathbf{P} \tag{D.11}$$

where, $\overset{\triangle}{\boldsymbol{F}}$ is the convected time derivative of the deformation gradient

$$\overset{\triangle}{\boldsymbol{F}} = \frac{\partial \boldsymbol{F}}{\partial t} + \left(\nabla_0 \boldsymbol{F}\ \boldsymbol{V}_{\mathrm{S}}\right)^{\mathrm{T}} + \boldsymbol{F}\ \left(\nabla_0 \boldsymbol{V}_{\mathrm{S}}\right)^{\mathrm{T}} \tag{D.12}$$

Finally, using the above derivation, it is easy to show that

$$\left(\nabla_{\mathrm{s}}\ \cdot \boldsymbol{v}_{\mathrm{s}}\right) = \overset{\triangle}{\boldsymbol{F}}\boldsymbol{G} : \mathbf{P} \tag{D.13}$$

E. DERIVATION FOR THE JUMP IN STRESS-VELOCITY TERM

Using Equation (D.4), the stress-velocity jump term is written as

$$[\![\boldsymbol{\sigma v}]\!] \cdot \boldsymbol{n} = [\![\boldsymbol{\sigma} \boldsymbol{v}_{\mathrm{s}} - \boldsymbol{\sigma} \left(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}\right)]\!] \cdot \boldsymbol{n} \tag{E.1}$$

Denoting the average $\langle\!\langle \cdot \rangle\!\rangle = \frac{1}{2}(\cdot_{-} + \cdot_{+})$, the following product relationship may be derived: $[\![\mathbf{ab}]\!] = [\![\mathbf{a}]\!] \langle\!\langle \mathbf{b} \rangle\!\rangle + \langle\!\langle \mathbf{a} \rangle\!\rangle [\![\mathbf{b}]\!]$. Thus,

$$\begin{aligned} [\![\boldsymbol{\sigma} \boldsymbol{v}_{\mathrm{s}}]\!] \cdot \boldsymbol{n} &= \left([\![\boldsymbol{\sigma}]\!] \langle\!\langle \boldsymbol{v}_{\mathrm{s}} \rangle\!\rangle + \langle\!\langle \boldsymbol{\sigma} \rangle\!\rangle [\![\boldsymbol{v}_{\mathrm{s}}]\!]\right) \cdot \boldsymbol{n} \\ &= [\![\boldsymbol{\sigma n}]\!] \cdot \boldsymbol{v}_{\mathrm{s}} \\ &= \left(- [\![\rho \boldsymbol{v} \left(v_{\mathrm{s_n}} - v_n\right)]\!] - \nabla_{\mathrm{s}} \cdot \left(\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}}\right)\right) \cdot \boldsymbol{v}_{\mathrm{s}} \end{aligned} \tag{E.2}$$

where, we have used Equation (2.23) and Equation (D.6). Thus, using Eq. (A.6), and substituting Equation (E.2) into Equation (E.1), we get:

$$[\![\boldsymbol{\sigma v}]\!] \cdot \boldsymbol{n} = - [\![\rho \boldsymbol{v}_{\mathrm{s}} \cdot \boldsymbol{v} \left(v_{\mathrm{s_n}} - v_n\right)]\!] - [\![\boldsymbol{\sigma} \left(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}\right)]\!] \cdot \boldsymbol{n} - \left(\nabla_{\mathrm{s}} \cdot \left(\mathbf{P} \boldsymbol{\sigma}_{\mathrm{s}}\right)\right) \cdot \boldsymbol{v}_{\mathrm{s}} \tag{E.3}$$

F. DERIVATION OF THE SECOND LAW CONDITION IN THE REFERENCE CONFIGURATION

Writing $(\boldsymbol{v}_{\mathrm{s}} - \boldsymbol{v}) = \boldsymbol{F}\boldsymbol{V}_{\mathrm{S}}$, the second law condition Eq. (2.37) can be rewritten as,

$$\int_{\Gamma_{\mathrm{s}}} \boldsymbol{V}_{\mathrm{S}} \cdot \left[\!\!\left[\boldsymbol{F}^{\mathrm{T}} \left(\rho\psi\mathbf{I} - \rho\sum_{\alpha=1}^{N} \mu^{\alpha}\nu^{\alpha}\mathbf{I} - \boldsymbol{\sigma} \right) \right]\!\!\right] \boldsymbol{n}\, d\Gamma_{\mathrm{s}} \geq 0 \tag{F.1}$$

Nanson's formula [101] is now used to relate the differential surfaces in current and reference configurations,

$$\boldsymbol{n}\, d\Gamma_{\mathrm{s}} = J\boldsymbol{G}^{\mathrm{T}}\boldsymbol{N}\, d\Gamma_{\mathrm{s}_0} \tag{F.2}$$

where, $J = det(\boldsymbol{F}) > 0$ is the Jacobian or the determinant of the deformation gradient $\boldsymbol{F}$, $\boldsymbol{G} = \boldsymbol{F}^{-1} = \frac{\partial \boldsymbol{X}}{\partial \boldsymbol{x}}$ is the inverse of the deformation gradient, and $\boldsymbol{N}$ is the normal to the interface in the reference configuration. By the definition of the first Piola-Kirchoff stress tensor,

$$\int_{\Gamma_{\mathrm{s}}} [\![\boldsymbol{\sigma}]\!]\, \boldsymbol{n}\, d\Gamma_{\mathrm{s}} = \int_{\Gamma_{\mathrm{s}_0}} [\![\boldsymbol{\sigma}_{\mathrm{I}}]\!]\, \boldsymbol{N}\, d\Gamma_{\mathrm{s}_0} \tag{F.3}$$

leading to the expression

$$\boldsymbol{\sigma}_{\mathrm{I}} = J\boldsymbol{\sigma}\boldsymbol{G}^{\mathrm{T}} \tag{F.4}$$

Thus, expressing the integral in Eq. (F.1) in the reference configuration, and using the fact that Γ_{s_0} is arbitrary, we get the second law condition in reference configuration as

$$\boldsymbol{V}_{\mathrm{S}} \cdot \left[\!\!\left[\rho_0 \left(\psi_0 - \sum_{\alpha=1}^{N} \mu_0^{\alpha}\nu_0^{\alpha} \right) \mathbf{I} - \boldsymbol{F}^{\mathrm{T}}\boldsymbol{\sigma}_{\mathrm{I}} \right]\!\!\right] \boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{F.5}$$

Using the fact that the deformation gradient $\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \mathbf{I} + \nabla_0 \boldsymbol{U}^{\mathrm{T}}$ the above condition can be expressed in the following alternative form,

$$\boldsymbol{V}_{\mathrm{S}} \cdot \left[\!\!\left[\boldsymbol{\Sigma}_0 - \rho_0 \sum_{\alpha=1}^{N} \mu_0^{\alpha} \nu_0^{\alpha} \mathbf{I} - \boldsymbol{\sigma}_{\mathrm{I}} \right]\!\!\right] \boldsymbol{N} \geq 0 \quad \text{on} \quad \Gamma_{\mathrm{s}_0} \tag{F.6}$$

where, we have used the fact that $\rho_0 = J\rho$. $\boldsymbol{\Sigma}_0 = \rho_0 \psi_0 \mathbf{I} - \nabla_0 \boldsymbol{U} \boldsymbol{\sigma}_{\mathrm{I}}$ is the Eshelby energy-momentum tensor [103] in the reference configuration.

G. EVALUATING SURFACE LAPLACIANS FOR PARAMETRIC GEOMETRIES

The procedure to determine the surface Laplacian $\nabla_s^2 f$ for any scalar field f, as well as the curvature at a point, $\kappa = -\nabla_\mathrm{s} \cdot \boldsymbol{n}$ for curves is straightforward, while the procedure for surfaces is involved. They are hence described separately here.

G.1 Curves

Let s denote the arc length of the curve and $g = \frac{ds}{du}$ denote its metric. For scalar and vector fields f and $\boldsymbol{g}$, respectively, we have

$$\nabla f = \frac{\partial f}{\partial n}\boldsymbol{n} + \frac{\partial f}{\partial s}\hat{e}_s \tag{G.1}$$

$$\nabla \boldsymbol{g} = \boldsymbol{n} \otimes \frac{\partial \boldsymbol{g}}{\partial n} + \hat{e}_s \otimes \frac{\partial \boldsymbol{g}}{\partial s} \tag{G.2}$$

The surface gradient is obtained using the projection operator $\mathcal{P}$ as,

$$\nabla_\mathrm{s} f = \mathcal{P}\nabla f = \nabla f - \boldsymbol{n}\frac{\partial f}{\partial n} = \frac{\partial f}{\partial s}\hat{e}_s = \frac{1}{g}\frac{\partial f}{\partial u}\hat{e}_s \tag{G.3}$$

For a vector field $\boldsymbol{g}$, we have,

$$\nabla_\mathrm{s} \boldsymbol{g} = \mathcal{P}\nabla \boldsymbol{g} = \mathcal{P}\left(\boldsymbol{n} \otimes \frac{\partial \boldsymbol{g}}{\partial n} + \hat{e}_s \otimes \frac{\partial \boldsymbol{g}}{\partial s}\right) = \hat{e}_s \otimes \frac{\partial \boldsymbol{g}}{\partial s} \tag{G.4}$$

$$\nabla_\mathrm{s} \cdot \boldsymbol{g} = \hat{e}_s \cdot \frac{\partial \boldsymbol{g}}{\partial s} \tag{G.5}$$

Thus, the surface Laplacian is given by,

$$\nabla_s^2 f = \hat{e}_s \cdot \left(\frac{\partial}{\partial s} \left(\frac{\partial f}{\partial s} \hat{e}_s \right) \right) \tag{G.6}$$

$$= \frac{\partial}{\partial s} \left(\frac{\partial f}{\partial s} \right) \tag{G.7}$$

$$= \frac{1}{g} \frac{\partial}{\partial u} \left(\frac{1}{g} \frac{\partial f}{\partial u} \right) \tag{G.8}$$

$$= \frac{1}{g} \left(\frac{-1}{g^2} g_{,u} f_{,u} + \frac{1}{g} f_{,uu} \right) \tag{G.9}$$

$$\nabla_s^2 f = \frac{1}{g^3} (-g_{,u} f_{,u} + g f_{,uu}) \tag{G.10}$$

where, we have used the fact that $\frac{\partial \hat{e}_s}{\partial s}$ is in the normal direction, i.e., $\hat{e}_s \cdot \frac{\partial \hat{e}_s}{\partial s} = 0$. The metric is given by $g = \sqrt{x_{,u}^2 + y_{,u}^2}$ which also allows computing $g_{,u}$ in a straightforward manner. The curvature is given by the well-known expression,

$$\kappa = \frac{\frac{d^2 y}{dx^2}}{\left(1 + \frac{dy}{dx}\right)^{3/2}} \tag{G.11}$$

$$\kappa = \frac{1}{g^3} (x_{,u} y_{,uu} - y_{,u} x_{,uu}) \tag{G.12}$$

The Laplacian of the curvature can be computed analytically by finding parametric derivatives through Eq.(G.12) and using Eq.(G.10).

G.2 Surfaces

Let $\boldsymbol{r}$ denote the position of any point on the surface, and $\boldsymbol{r}_u$ and $\boldsymbol{r}_v$ denote the parametric derivatives $\frac{\partial \boldsymbol{r}}{\partial u}$ and $\frac{\partial \boldsymbol{r}}{\partial v}$, respectively. Henceforth, a subscript with u or v shall denote the respective partial derivative unless otherwise specified. The curvature and surface Laplacian for a parametric surface are given in terms of quantities called

fundamental forms. The first and second fundamental forms are defined respectively as,

$$\boldsymbol{I} = \begin{bmatrix} \boldsymbol{r}_u \cdot \boldsymbol{r}_u & \boldsymbol{r}_u \cdot \boldsymbol{r}_v \\ \boldsymbol{r}_u \cdot \boldsymbol{r}_v & \boldsymbol{r}_v \cdot \boldsymbol{r}_v \end{bmatrix} = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \tag{G.13}$$

$$\boldsymbol{II} = \begin{bmatrix} \boldsymbol{n} \cdot \boldsymbol{r}_{uu} & \boldsymbol{n} \cdot \boldsymbol{r}_{uv} \\ \boldsymbol{n} \cdot \boldsymbol{r}_{uv} & \boldsymbol{n} \cdot \boldsymbol{r}_{vv} \end{bmatrix} = \begin{bmatrix} L & M \\ M & N \end{bmatrix} \tag{G.14}$$

Another surface quantity that is commonly referred to is the shape operator $S = \nabla_{\mathrm{s}} \boldsymbol{n}$. The Weingarten equation gives the curvature of the surface in terms of the fundamental forms as, $\kappa = \mathrm{tr}\left([\boldsymbol{II}][\boldsymbol{I}^{-1}]\right)$. The final results for the surface gradient, divergence and Laplacian shall be presented here, with detailed derivation provided in a subsequent section.

For any scalar field f, the surface gradient is given by,

$$\nabla_{\mathrm{s}} f = \boldsymbol{R}_1 f_u + \boldsymbol{R}_2 f_v \tag{G.15}$$

where,

$$\boldsymbol{R}_1 = \frac{1}{(EG - F^2)}[G\boldsymbol{r}_u - F\boldsymbol{r}_v] \tag{G.16}$$

$$\boldsymbol{R}_2 = \frac{1}{(EG - F^2)}[-F\boldsymbol{r}_u + E\boldsymbol{r}_v] \tag{G.17}$$

We also have the relation $EG - F^2 = \det(\boldsymbol{I}) = \|\boldsymbol{r}_u \times \boldsymbol{r}_v\|^2$. For a vector field $\boldsymbol{g}$, the surface gradient and divergence are similarly given by,

$$\nabla_{\mathrm{s}} \boldsymbol{g} = \boldsymbol{R}_1 \otimes \boldsymbol{g}_u + \boldsymbol{R}_2 \otimes \boldsymbol{g}_v \tag{G.18}$$

$$\nabla_{\mathrm{s}} \cdot \boldsymbol{g} = \boldsymbol{R}_1 \cdot \boldsymbol{g}_u + \boldsymbol{R}_2 \cdot \boldsymbol{g}_v \tag{G.19}$$

Finally, the surface Laplacian for any scalar field f is given by,

$$\begin{aligned}\nabla_s^2 f = &\left\|\boldsymbol{R}_1\right\|^2 f_{uu} + \left(\boldsymbol{R}_1 \cdot \boldsymbol{R}_{1,u} + \boldsymbol{R}_2 \cdot \boldsymbol{R}_{1,v}\right) f_u + 2\boldsymbol{R}_1 \cdot \boldsymbol{R}_2 f_{uv} \\ &+ \left\|\boldsymbol{R}_2\right\|^2 f_{vv} + \left(\boldsymbol{R}_1 \cdot \boldsymbol{R}_{2,u} + \boldsymbol{R}_2 \cdot \boldsymbol{R}_{2,v}\right) f_v\end{aligned} \tag{G.20}$$

Thus, to compute the surface Laplacian of any field f, one needs to compute the parametric derivatives f_u, f_v, f_{uu}, f_{uv} and f_{vv}. For solved quantities like the electric potential which are represented through shape functions, these derivatives are easily obtained.

Using Eq.(G.19) on the normal at a point, we get the relation for the curvature at a point as,

$$\kappa = -\nabla_{\mathrm{s}} \cdot \boldsymbol{n} = \frac{1}{(EG - F^2)}[GL - 2FM + EN] \tag{G.21}$$

Since the curvature depends on L, M, N, it is by itself a second order quantity. To compute the surface Laplacian of the curvature, one needs to compute its parametric derivatives up to the second order. This is not easily computable since these quantities are fourth order derivatives. One workaround is to do a NURBS fit for the curvature and use the NURBS basis functions to approximate the parametric differentiation.

VITA

VITA

Pavan Kumar was born and brought up in Chennai, India. He graduated with Bachelors in Mechanical Engineering from the Indian Institute of Technology Madras in 2015. The same year, he joined the Ph.D. program in Purdue University under Prof. Ganesh Subbarayan in the school of Mechanical Engineering. His research interests lie in the theoretical and numerical modeling of moving boundary problems, particularly phase evolution problems. He has also worked on optimizing thermal and mechanical properties of particulate Thermal Interface Materials (TIMs), and on the reliability of the back end of line to current pulsing. His interests outside of work include reading, chess and music. He will be joining Intel post graduation.