# Automation & Expert System Framework for Coupled Shell-Solid Finite Element Modeling of Complex Structures

Manasi P. Palwankar

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Aerospace Engineering

Rakesh K. Kapania, Chair
Joseph A. Schetz
Mayuresh J. Patil
William N. Alexander
Daniel C. Hammerand

Feb 18, 2022
Blacksburg, Virginia

Keywords: Finite Element Modeling, Shell-Solid Coupling, Expert System, Machine Learning, Fuzzy Logic, Decision Trees, Stiffened Shells, Fuzzy Inference System

# Automation & Expert System Framework for Coupled Shell-Solid Finite Element Modeling of Complex Structures

Manasi P. Palwankar

(ABSTRACT)

Finite Element (FE) analysis is a powerful numerical technique widely utilized to simulate the real-world response of complex engineering structures. With the advancements in adaptive optimization frameworks, multi-fidelity (coupled shell-solid) FE models are increasingly sought during the early design stages where a large design space is being explored. This is because multi-fidelity models have the potential to provide accurate solutions at a much lower computational cost. However, the time and effort required to create accurate and optimal multi-fidelity models with acceptable meshes for highly complex structures is still significant and is a major bottleneck in the FE modeling process. Additionally, there is a significant level of subjectivity involved in the decision-making about the multi-fidelity element topology due to a high dependence on the analyst's experience and expertise, which often leads to disagreements between analysts regarding the optimal modeling approach and heavy losses due to schedule delays. Moreover, this analyst-to-analyst variability can also result in significantly different final engineering designs. Thus, there is a greater need to accelerate the FE modeling process by automating the development of robust and adaptable multi-fidelity models as well as eliminating the subjectivity and art involved in the development of multi-fidelity models. This dissertation presents techniques and frameworks for accelerating the finite element modeling process of multi-fidelity models. A framework for the automated development of multi-fidelity models with adaptable 2-D/3-D topology using the parameterized full-fidelity and structural fidelity models is presented. Additionally,

issues related to the automated meshing of highly complex assemblies is discussed and a strategic volume decomposition technique blueprint is proposed for achieving robust hexahedral meshes in complicated assembly models. A comparison of the full-solid, full-shell, and different multi-fidelity models of a highly complex stiffened thin-walled pressure vessel under external and internal tank pressure is presented. Results reveal that automation of multi-fidelity model generation in an integrated fashion including the geometry creation, meshing and post-processing can result in considerable reduction in cost and efforts. Secondly, the issue of analyst-to-analyst variability is addressed using a Decision Tree (DT) based Fuzzy Inference System (FIS) for recommending optimal 2D-3D element topology for a multi-fidelity model. Specifically, the FIS takes the structural geometry and desired accuracy as inputs (for a range of load cases) and infers the optimal 2D-3D topology distribution. Once developed, the FIS can provide real-time optimal choices along with interpretability that provides confidence to the analyst regarding the modeling choices. The proposed techniques and frameworks can be generalized to more complex problems including non-linear finite element models and as well as adaptable mesh generation schemes.

# Automation & Expert System Framework for Coupled Shell-Solid Finite Element Modeling of Complex Structures

Manasi P. Palwankar

(GENERAL AUDIENCE ABSTRACT)

Structural analysis is the process of determining the response (mainly, deformation and stresses) of a structure under specified loads and external conditions. This is often performed using computational modeling of the structure to approximate its response in real-life conditions. The Finite Element Method (FEM) is a powerful and widely used numerical technique utilized in engineering applications to evaluate the physical performance of structures in several engineering disciplines, including aerospace and ocean engineering. As optimum designs are increasing sought in industries, the need to develop computationally efficient models becomes necessary to explore a large design space. As such, optimal multi-fidelity models are preferred that utilize higher fidelity computational domain in the critical areas and a lower fidelity domain in less critical areas to provide an optimal trade-off between accuracy and efficiency. However, the development of such optimal models involves a high level of expertise in making *a-priori* and *a-posteriori* optimal modeling decisions. Such experience based variability between analysts is often a major cause of schedule delays and considerable differences in final engineering designs. A combination of automated model development and optimization along with an expert system that relieves the analyst of the need for experience and expertise in making software and theoretical assumptions for the model can result in a powerful and cost-effective computational modeling process that accelerates technological advancements. This dissertation proposes techniques for automating robust development

of complex multi-fidelity models. Along with these techniques, a data-driven expert system framework is proposed that makes optimal multi-fidelity modeling choices based on the structural configuration and desired accuracy level.

# Dedication

*For my parents,*

*Prafulla & Namrata Palwankar,*

*&*

*my grandmother,*

*Kusum Palwankar.*

# Acknowledgments

First of all, I would like to express my deepest gratitude to Dr. Rakesh K. Kapania for believing in me and providing me with the honor of completing my Ph.D. studies under his supervision. His support and motivation during the tough times has helped me tremendously to continue working consistently and achieve my targets. He provided me with the autonomy to come up with my own solutions all the while being attentive and giving directions to make sure I do not lose sight of my goals. This helped me immensely to improve my own problem solving skills and become an independent researcher with a solid foundation. One of my favorite interactions with him will be during his Elastic Stability course when he would make sure all the students comprehend the material completely. I still recall scoring a test grade close to 100; and he called out a completely avoidable mistake I had made on the test. Dr. Kapania's encouragement to strive for perfection has been instrumental to my path within the PhD program. It's been an honor to have an advisor who is passionate about helping his students learn and achieve goals. I will carry along his pursuit of excellence and passion for structural mechanics in my future professional endeavours.

During my PhD studies, I had the privilege of collaborating with M4 Engineering and working under the guidance of Dr. Daniel Hammerand, whose invaluable insights and expertise have significantly shaped this dissertation work. I am extremely grateful for his guidance, timely constructive feedback and our discussions; technical and non-technical, that improved my work and also trained me to be a better professional. I must also thank him for being patient and sharing his decades of wisdom and experience with us, even with the fast pace of the project.

I would also like to express my sincere gratitude for my Ph.D. advisory committee members, Dr. Mayuresh Patil, Dr. Joseph Schetz and Dr. Nathan Alexander for their willingness to serve on my committee, reviewing my dissertation and providing insightful suggestions during the committee meetings. I would like to specially thank Dr. Patil for providing guidance and encouragement during a time of uncertainty as I navigated through different research tracks in the Ph.D. program.

I gratefully acknowledge the funding provided by US Navy for sponsoring this research and my Ph.D. studies under contract number: N68335-19c-0117 and Topic Number: N171-059. I would like to specially thank Nathan Leong, Navy's Technical Monitor, whose experience and practical suggestions during the tri-weekly meetings provided me with the knowledge and direction for this research work. I am also thankful to the many technical discussions and help provided by Edward Swabowski, Dr. Mohamed Jrad and Steven Doyle from M4 Engineering, with whom I had the opportunity to collaborate during the Navy sponsored project. I would also like to extend my sincere thanks to Junhyeon Seo, Dr. Karanpreet Singh and Mohcine Harrach from Virginia Tech for their contributions and discussions during the project.

I wish to express my deep gratitude to my Master's advisor, Dr. Satchi Venkataraman at San Diego State University, whose guidance and encouragement motivated me to pursue a Ph.D. degree.

I have had the priviledge of becoming friends with many brilliant minds from around the world at Virginia Tech. I would like to thank all my colleagues in Dr. Kapania's research group, specially Junhyeon Seo, Dr. Karanpreet Singh, Dr. Berkan Alanbay, Dr. Rikin Gupta, Dr. Jitish Miglani and Siddhant Desai for their friendship. I am incredibly thankful for my

close friends who became my family in Blacksburg; Special thanks to Nandita Hari for being an amazing friend and for always having my back. I will also forever cherish all the fun outings with Neslihan Genckal and Varakini Sanmugadas that made my stay in Blacksburg so much worthwhile. Lastly, I also want to take a moment to thank my childhood friends from India, Sayali and Kanaka for their unconditional friendship through thick and thin during my graduate studies.

My most heartfelt thanks goes to my family. My father has been the biggest supporter of my dreams; he has instilled in me a sense of adventure, courage to dream big and never to give up attitude. It's my mother's love for books that I read about space exploration at a very young age only to determine that I want to design and build rockets once I grow up and may be go to Jupiter someday! :D. Thank you for listening to my everyday stress and giving your never-ending love and support that made it possible for me keep going. Thank you for *Everything*. Finally, very importantly (despite what she might think), I thank my younger sister, Manali, for bearing with a grumpy grad student for all these years! I promise you won't have to ask again why I am this cranky all the time!

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background

*"You're starting to see the era of digital... for the first time ever we have full-up, from the start, digital ship designs."* [1]

Hon. James F. Geurts

Assistant Secretary of the Navy (Research, Development, & Acquisition)

WEST Conference, San Diego Convention Center, San Diego, CA, February 2018

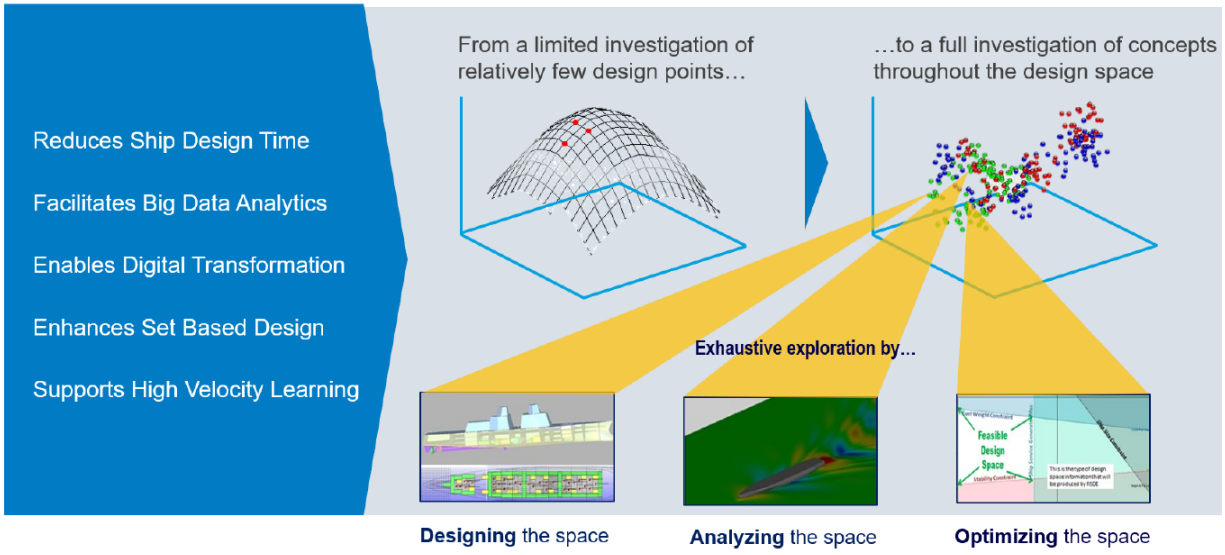



Figure 1.1: A part of digital engineering vision of Department of Defense as presented in the work by Voth *et al.* [1]

Figure 1.2: Illustration of a large and complex real-life structure: Columbia Class Submarine [2].

Aerospace and marine industries are making technological advancements at an exponential rate with the rise in computational power. A range of industries are implementing a digital transformation in order to 'meet new threats, maintain overmatch, and leverage technology advancements'. Work by Zimmerman [3] presents insights into the shift in progress within the DOD towards achieving transformation of traditional engineering practices into digital engineering. As such, there is a growing need to rapidly develop total structural model developments and analysis software suites. Figure 1.1 shows an example of the envisioned platform and framework that highlights the digital transformation being a core strategic initiative of the U.S. Department of Defense Digital Engineering Strategy.

As industries are increasingly turning towards qualifying their new structural configurations through computational analyses, the popularity of the Finite Element Method (FEM) has escalated during the last few decades and is a preferred option for analyzing complex structural geometries. In simple terms, FEM makes *a priori* assumption about the approximation to be utilized for the solution, divides the problem domain into discrete regions, called 'elements' which are chosen appropriate to the problem. These elements are then assembled together and solved using linear algebra methods [4]. This technique has demonstrated its superiority in solving complicated problem domains for which analytical solutions do not exist.

Developing full computational models of real-life large and complex structures using a single model allows engineers to capture the global as well as critical local structural responses in a single analysis step. An example of such a real-life structure is a complex submarine hull as shown in Fig. 1.2. However, FE models of such large and complex structures quickly become computationally expensive if only solid FE elements are used. As adaptive optimized designs are being sought by the industry, there is still a need for computationally efficient models at the conceptual design stages. During these preliminary stages, several designs are explored and a smaller set is finalized for detailed analyses. In such cases, it is often more desirable to use reduced order models which capture crucial structural responses while maintaining lower computational costs. This way, large design spaces can be analyzed and effective designs can be selected in the initial stage itself.

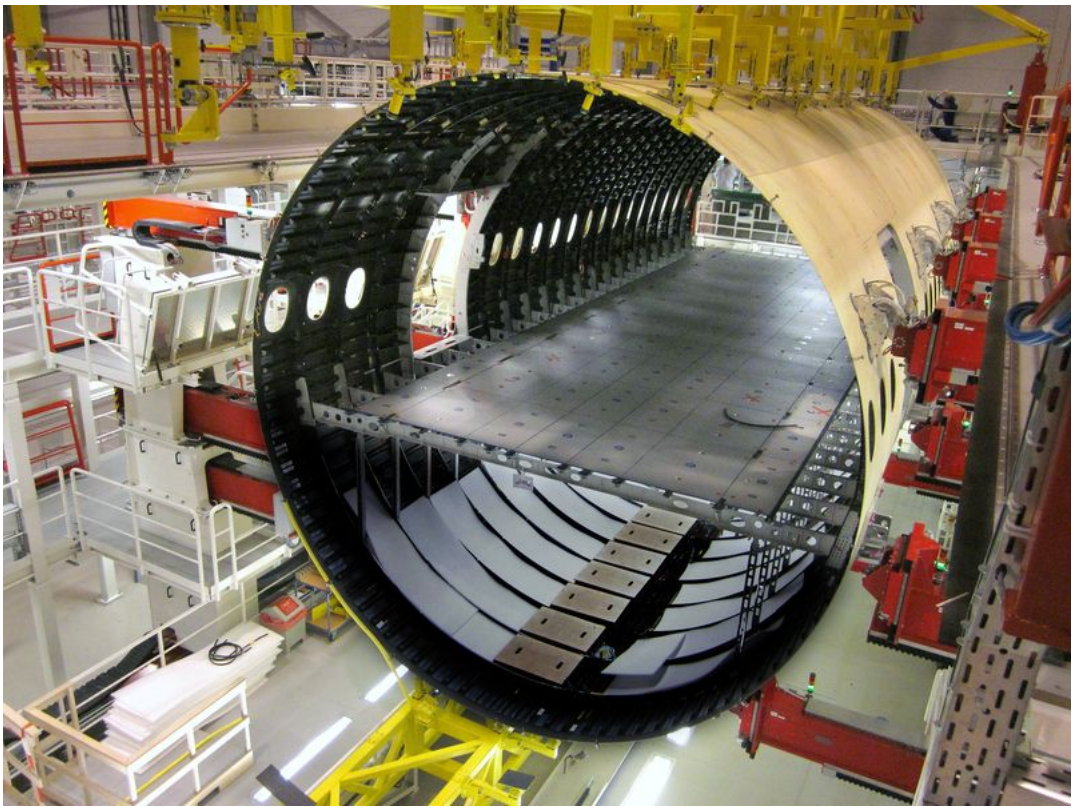


Figure 1.3: Illustration of the stiffened thin-walled cylindrical shell structure of an aircraft fuselage [5].

Real-life thin-walled structures like submarines, aircraft fuselages (Fig. 1.3) and spacecraft can be adequately modeled using reduced-order multi-fidelity models that utilize a combination of 2-D shell and 3-D solid finite elements. Such models prove to be efficient where critical local regions require a relatively fine through-the-thickness solid mesh and a coarser shell mesh away from the high-priority regions. Generally, thin-walled regions like the hull and stiffeners are modeled using shell elements based on thin-shell theory, which have been shown to accurately predict the stresses in thin-walled structures. These elements can be easily modeled with less computational cost. The more complicated regions like intersections and tapers between the thick and thin portions of the pressure vessel need 3-D solid elements to accurately capture the three-dimensional response. A shell-solid finite element

coupling is utilized to ensure correct transfer of loads and displacements at the intersection of two different domain topologies. A typical example of a shell-to-solid coupling is shown in Fig. 1.4).

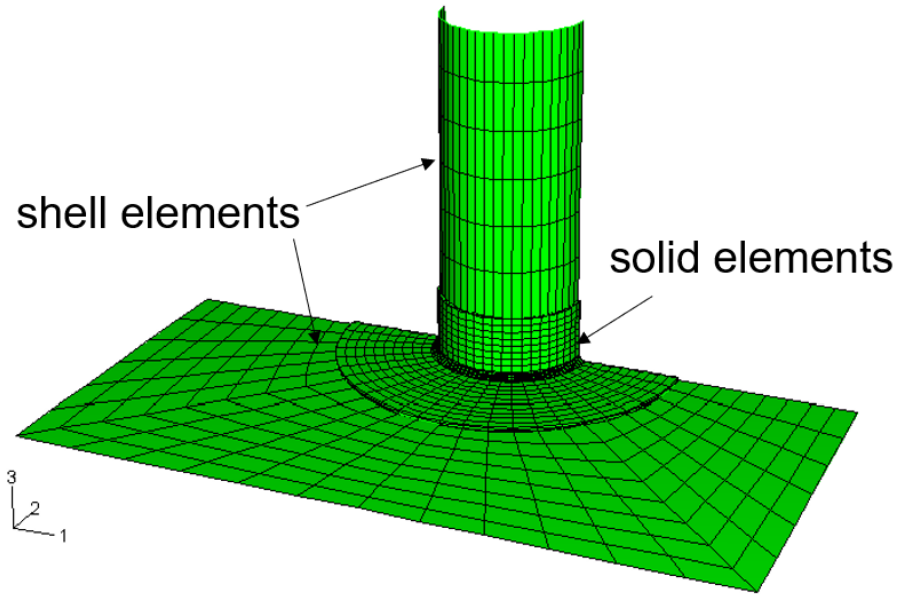
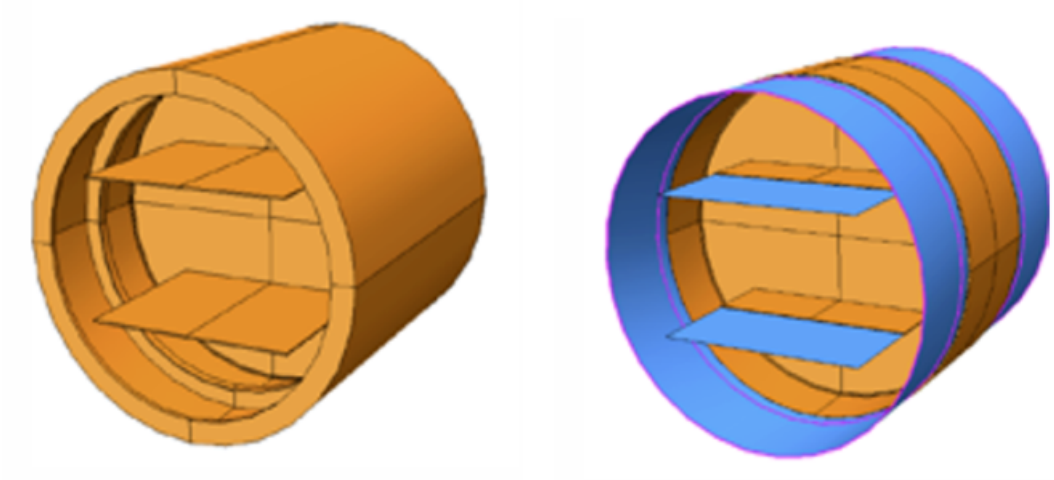



Figure 1.4: Example of a multi-fidelity model, solid elements at the intersection and shell elements away from the joint [6].

Figure 1.5 demonstrates an example of a full-fidelity submarine pressure vessel FE model (Fig. 1.5(a)) converted into a multi-fidelity model (Fig. 1.5(b)) using shell elements away from the intersections (shown in blue) and the stiffeners modeled as 1-D beam elements (shown in magenta) for computational savings.



(a) (b)

Figure 1.5: Computational models of a submarine hull structure: (a) full fidelity model; (b) multi-fidelity model.

Development of an appropriate multi-fidelity model is vital to making sure the structural response is predicted with sufficient accuracy. As several different structural configurations are evaluated, these different modeling techniques need to be adapted to the geometry and loading changes to avoid missing critical structural responses. The decision about the use of a particular approximation in a region becomes increasing vague as the complexity of the structural geometry increases. Often, the decision about selecting an optimal element topology in a mixed model is subjective and relies heavily on the analyst's experience. An analyst usually makes modeling decisions like the choice of the finite element type, analysis type, mesh density based on their experience and expertise. This leads to considerable variability in the modeling choices which eventually gets propagated in the final engineering designs. Furthermore, this variability leads to disagreements between analysts regarding the technical adequacy of a FE model. Generally, multiple modeling configurations are explored and an optimal one is selected. Such an on-going adjustment is a major bottleneck in the

industry. The delays caused in the schedule due to this on-going modeling process forms a significant portion of the non-recurring budget of industries.

As the world enters an era of high speed computing and data-driven decision making systems, there is a growing trend in the field of engineering applications to relieve the end user of the need to have expert knowledge necessary (and thereby the involved subjectivity) for the use of FE modeling and analysis. In order to achieve this goal, an expert system along with automated data generation and optimal model development can make the entire analysis process powerful, efficient and effective for diagnosis of complicated structures [7]. For this purpose, this dissertation explores solutions to eliminate the bottlenecks in the FE modeling process of coupled shell-solid multi-fidelity models and presents solutions to accelerate the utilization of these advanced FE techniques.

## 1.2 Summary of Contributions

The major contributions of this dissertation can be summarized as follows:

1) Techniques related to automatic creation and post-processing of adaptable multi-fidelity FE models using parameterization and scripting within FE packages.
2) A blueprint for strategic volume decomposition of complex assembly models for robust hexahedral mesh generation.
3) Assessment of the multi-fidelity modeling and automation technique for a realistic, highly complex stiffened tank structure subjected to static external hydrostatic pressure as well as internal tank pressure with a parametric study of variations in geometric and multi-fidelity parameters.
4) Discussion of major issues and insights gained related to the development of automation scripts of multi-fidelity models.

5) Utilization and assessment of domain-based local error metrics for comparing FE models with different fidelities

6) Utilization of interpretable machine learning (Decision Trees) to extract and generate data-driven rules for modeling recommendations which lead to an automated knowledge base generation related to structural behavior.

7) Novel application of Decision Trees (DT) combined with Fuzzy Inference System (FIS) in the field of expert systems for providing optimal modeling recommendations related to multi-fidelity modeling element topology.

8) Consideration for imprecise data for modeling recommendations using fuzzy logic.

## 1.3 Dissertation Structure

This dissertation is structured as follows:

**Chapter 1** outlines the motivation, background and the major contributions of the dissertation in the field of multi-fidelity modeling for structural analyses. It introduces the major concepts and approaches utilized in the dissertation.

**Chapter 2** presents background on finite elements, and a detailed literature review of the existing state-of-art techniques for developing automated multi-fidelity models and expert system frameworks for FE modeling decisions. The need for an integrated design process with automated model generation, meshing and post-processing along with an expert system framework for making optimal multi-fidelity modeling choices is proposed. The frameworks presented in this work are demonstrated using two different application geometries, a component-level T-joint structure and a global submarine pressure vessel with a tank under internal pressure.

**Chapter 3** presents a journal manuscript that will be submitted to the Computers and Structures Journal after approval for public disclosure from the US NAVY. The Journal manuscript entitled as 'Automated FE Analysis of a Stiffened Tank Pressure Vessel using Shell-Solid Multi-Fidelity Modeling', presents techniques and discusses insights related to the automation of multi-fidelity models using parameterization and scripting. Additionally, a strategic volume decomposition scheme for robust hex-mesh generation is proposed. The proposed approaches are demonstrated using a highly complex stiffened thin-walled pressure vessel under external and internal pressure loads.

**Chapter 4** presents the parametric studies performed on the submarine stiffened thin-walled pressure vessel under different multi-fidelity element topologies. This chapter assesses the coupled shell-solid multi-fidelity models for different structural and mixed-fidelity parametric variations for the global model of the complex submarine tank structure.

**Chapter 5** summarizes a manuscript that is submitted for publication in the AIAA Journal entitled 'Making Finite Element Modeling Choices using Decision Tree based Fuzzy Inference System'. This study presents a Decision Tree (DT) based Fuzzy Inference System (FIS) for making optimal element topology distribution in the development of multi-fidelity Finite Element (FE) models. The proposed approach is presented using a T-joint structure as an application problem.

**Chapter 6** summarizes the major conclusions and contributions of this dissertation.

**Chapter 7** concludes the dissertation with a discussion about potential directions for future research.

## 1.4 Publications and Conference Proceedings

This section lists papers that have or will be submitted for publication. Content from these manuscripts has been used throughout this dissertation.

1) **Palwankar, M.**, Kapania, R.K., Hammerand,D., "Automated FE Analysis of a Stiffened Tank Pressure Vessel using Shell-Solid Multi-Fidelity Modeling," To be submitted to Computers and Structures Journal, 2022 (submitted to US Navy for approval for public disclosure).

2) **Palwankar, M.**, Kapania, R.K., Hammerand,D., "Making Finite Element Modeling Choices using Decision Tree based Fuzzy Inference System," AIAA Journal, 2022 (submitted).

3) **Palwankar, M.**, Kapania, R.K., Hammerand,D., "Intelligent Support System for Mixed Fidelity Finite Element Modeling Error Prediction using Decision Trees and Fuzzy Logic Classifier," 2022 AIAA SCITECH Forum, 2022, p. 0101. doi:10.2514/6.2022-0101

4) Singh K., **Palwankar, M.**, Kapania, R.K., Hammerand, D., "Machine Learning Approaches for Finite Element Modeling Recommendations," To be submitted to Advances in Computational Design Journal(manuscript nearly complete)

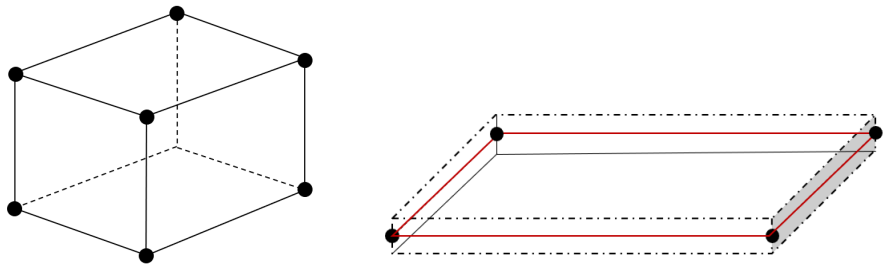
*Presented at ASME V&V Verification and Validation Symposium, 2019.

# Chapter 2

# Theory and Review of Literature

## 2.1 Multi-Fidelity Modeling in FEM

### 2.1.1 Element Types



(a) (b)

Figure 2.1: Illustration of two and three dimensional element types in FEM.: (a) a linear 3-D hexahedral 8-node element; (b) a 4-node 2-D shell element .

In general, 3-D solid finite elements (Fig. 2.1 (a)) have no *a priori* assumptions about the structural response and hence provide higher accuracy with adequate discretization. However, using them can quickly become computationally expensive. When the thickness of structures is very small as compared to the other two dimensions, a complete 3-D analysis is very expensive and often leads to serious numerical ill-conditioning of the problem. In order to avoid such issues, several assumptions are made regarding thin-walled structures. For these structures, shell elements (Fig. 2.1 (b)) can be utilized. Typically, shell elements work very well in predicting the bending behavior of a structure, but they are not capable of accurately predicting the localized 3-D response in the presence of discontinuities or other complexities. Shell elements are defined using certain kinematic assumptions about the through thickness behavior *a priori*. It is to be noted that shell elements specifically formulated for thick shells can capture through thickness responses still under kinematic assumptions made *a priori*. Shell elements have multiple integration

points through the thickness along with rotational degrees of freedom to include bending effects, whereas solid elements only have translational degrees of freedom. Hence, under bending loads, multiple solid elements are needed through the thickness of the structure to capture the bending response. In situations like these, it is more convenient to utilize 2-D surface elements like shells, which neglect both normal and transverse shear stresses in the thickness direction. However, utilization of only shell elements in the presence of complex structures (having connections, joints etc.) is problematic [8].

## 2.2 Coupled Shell-Solid Models

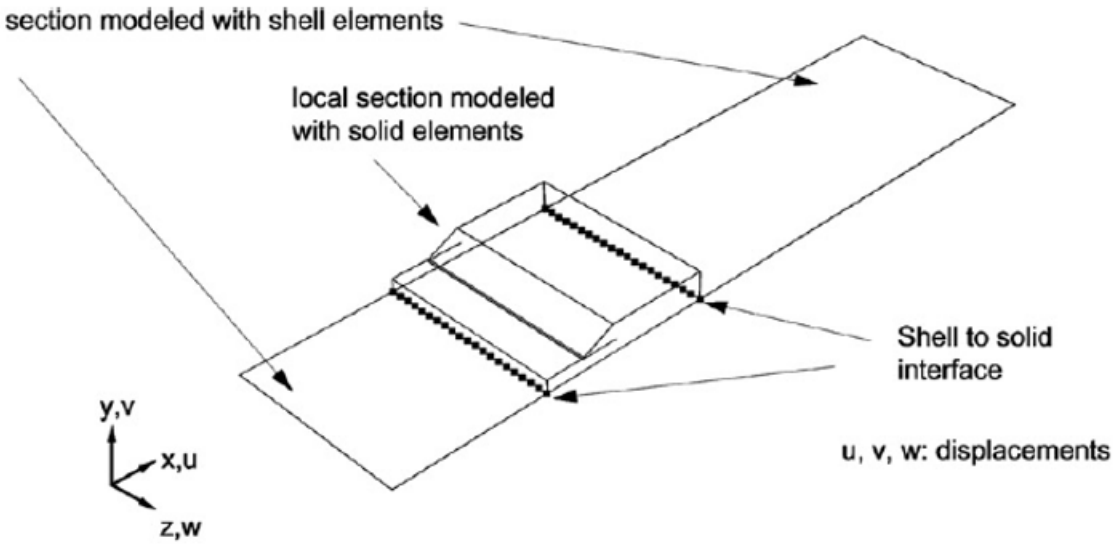



Figure 2.2: Example of a shell/3D model utilized by Krueger *et al.* [9] for debond specimens.

Many real world engineering structures utilize thin-walled stiffened pressure vessels, *e.g.*, aircraft fuselages, ship vessels, submarines etc. Thus, a large part of these structures can be modeled using reduced dimensional elements like beams, plates and shells. However, the

real-life structures are more complicated and have regions with complexities in geometry, loading and material behavior ([10]). In order to combine the benefits of reduced order elements along with the accuracy of higher order elements in a single model, a careful consideration of the 3D-2D coupling is required to avoid errors in the complex regions of interest. The shell elements and solid elements are coupled together by enforcing a set of distributing constraint equations between the shell edge nodes and the corresponding solid surface. An illustration of a solid surface connected to a shell edge is shown in Fig. 2.3. Because solid elements only have translational degrees of freedom (DOFs), the rotational DOFs of the shell elements need to be transferred correctly to the solid surface. This work utilized the multi-point constraint functions available in Abaqus/Standard to achieve the coupling between shell and solid elements [11] .

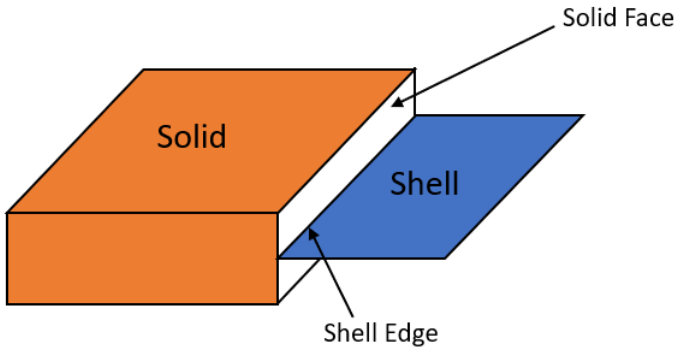



Figure 2.3: Shell-Solid-Coupling using a set of multi-point constraints that maintain the displacement and rotation compatibility between the two element types.

Multiple studies in the past have focused on improving coupling techniques for appropriate constraints at solid-shell transition of multi-fidelity models [10, 12]. The successful demonstration of the benefit of utilizing coupled shell-solid models have made these models popular in recent times. Such multi-fidelity models are common in the initial design stages rather than during the final steps, wherein full fidelity models are employed for highest

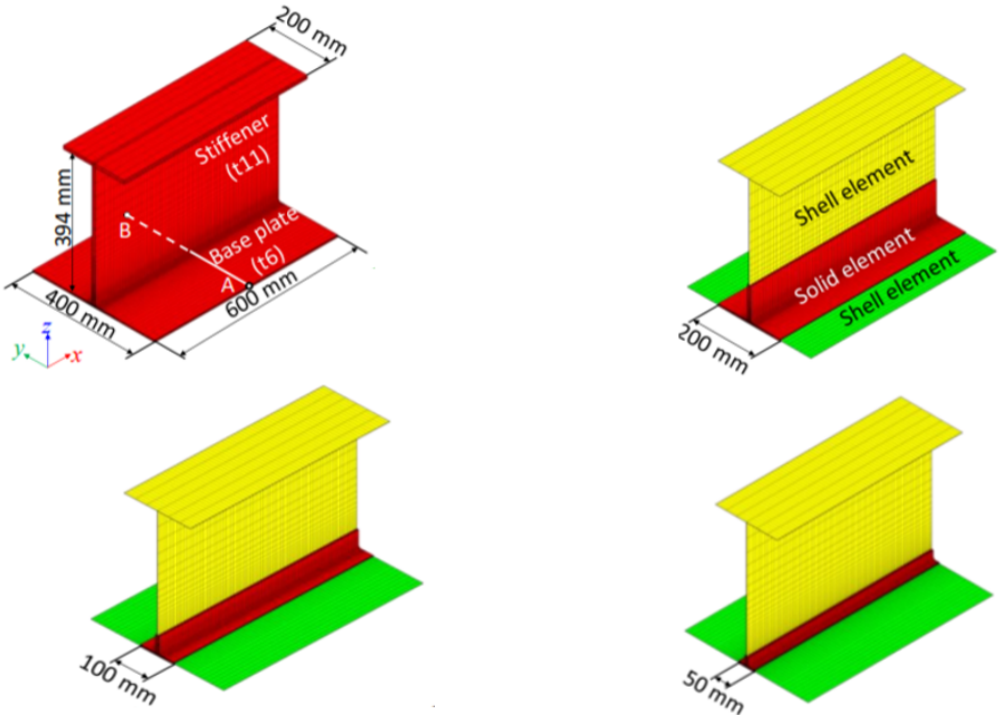



Figure 2.4: Multiple multi-fidelity models investigated by Ikushima *et al.* [16] for understanding the effect of solid region dimension $B_s$ on the analysis accuracy; (a) full solid model; (b) 200 mm solid region at the joint; (c) 100 mm solid region at the joint; (d) 50 mm solid region at the joint.

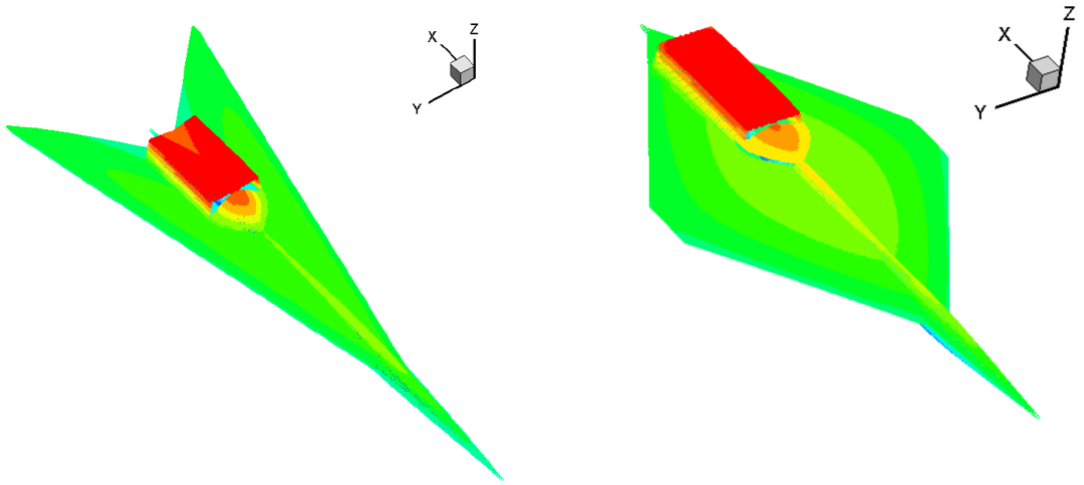
accuracy. As compared to a sub-model, which requires a two step analysis, including an initial global analysis followed by a sub-model analysis, the coupled shell-solid model uses a single step analysis. The use of such models is also gaining interest in the modeling of complex composite structures. Krueger *et al.* [13] presented a shell/solid modeling technique for analyzing delaminated composite laminates. Other works by Krueger *et al.* [9, 14], used shell/3D modeling technique for analyzing skin-stiffener debond specimens and performing debonding. Figure 2.2 shows a typical multi-fidelity shell/3D model developed by Krueger and Minguet [9] to analyze a debond-panel specimen. Another recent work by Salerno *et al.* [15] assessed the potential of a coupled shell-solid model for composite laminates undergoing low-velocity impact loadings, where delamination is expected to take place.

Work by Ikushima *et al.* [16] studied the nonlinear behavior of thin-plate complex structures using a shell-solid mixed-dimensional finite element model. Although, mixed-fidelity models have shown to be useful, a number of models are typically evaluated before a final model is selected as shown in Fig. 2.4. In another work, Reinosa *et al.* [17] assessed the potential of a shell-solid coupling to model the response of a composite specimen with stringers. Their work demonstrated importance of the local 3-D region being large enough to cover the damage/critical regions completely.

## 2.3 State-of-the-Art Techniques for Automation of Multi-fidelity FE Models

As complex computational simulations with adaptive optimization are on the rise, it is becoming crucial to automate the finite element model development [18]. For instance, studies [19] in the past have demonstrated the dramatic variations during a multi-disciplinary design process in the planform shape of a supersonic aircraft design depending on the fidelity level of optimization. Figure 2.5 shows the (a) highly swept wing configuration yielded by a low-fidelity optimization as compared to (b) a trapezoidal wing shape yielded by a higher fidelity optimization process. This demonstrates the importance of using higher fidelity simulations during optimization processes and the need for an integrated automated tool for building different finite element models at a faster rate. Additionally, recent work by Xie *et al.* [20] presents a certification-driven platform for air-frame preliminary design which emphasizes on the need to perform a large design space exploration when numerous tradeoffs between conflicting objectives are involved. In such situations, an automated development of multi-fidelity models, intended to be used in a multi-disciplinary optimization process can provide the high-fidelity design concepts at a

significantly lower computational cost. Such an automated process could also help in rapid design space exploration and development of accurate data-driven surrogate models.




(a) (b)

Figure 2.5: Varying design concept results for an efficient supersonic air vehicle based on different levels of optimization fidelity; (a) low-fidelity optimization and (b) higher fidelity optimization [19].

In the past, medial axis transforms [21, 22] and prominent cross-sections [23] have been utilized to automatically generate mixed dimensional models. However, the methods utilized in all these works are still under research and not applicable for highly complex geometries. More recently, parameterization coupled with in-built scripting capabilities of the commercial FE packages has been utilized to automate generation of FE models. Work done by Sohaib [24] presented a parameterization based model development of a generic wing model using CATIA V5 software. Lai *et al.* [25] performed parametric studies on different types of composite and cylindrical shells using parameterized FE models coded by Patran Command Language (PCL). In another work, Zhang *et al.* [26] performed optimization of aircraft mid-fuselage structure based on parametric modeling design.

Benaouli *et al.* [27] automated the integration of CAD/CAE systems for efficient parametric designs of aircraft wing structures using the built-in programming languages and model exchange capacity of modern computer aided tools. Work by Love *et al.* [28] focused on integrating aircraft parametric structural and the outer mold line geometry modeling to enhance the design process at the critical early conceptual design. In another recent work, Liu *et al.* [29] utilized a parameterized approach to establish a high fidelity 3-D finite element model of involute helical gears by performing all the procedures in finite element software. This greatly simplified the software environment and avoided data loss due to model delivery.

## 2.4 Robust Hexahedral Mesh Generation-A Major Bottleneck

Mesh generation is a major task in the FE model development procedure that needs to be automated to achieve a smooth design exploration process. Most of the times, developing an optimal mesh is considered an art that comes with experience and expertise. Out of the many available mesh types, hexahedral meshing is the most popular type due to its superior performance owing to the additional shape functions in the hexahedral meshes [8]. This is also supported by studies performed in the past [30, 31] on assessing the performance of hexahedral meshes versus tetrahedral meshes showing that tetrahedral meshes can be mathematically stiffer for high deformation analysis with linear elements. Additionally, the use of a hexahedral mesh decreases the overall element count as compared to a tetrahedral mesh. However, there are limitations in achieving a fully automated hexahedral mesh for complicated parts due to additional constraints that come into play due to the hexahedral shape. One of the critical limitations of hex-mesh generation is that for any face of the hex,

there must be an opposite face. The mesh can be viewed as an interlaced stack of elements (Fig. 2.6) such that each stack must begin and end at a boundary or be a closed loop of elements [32]. As it follows, any refinement of one element continues to propagate to its neighbors. This makes the auto-generation of a conformal hex mesh challenging.

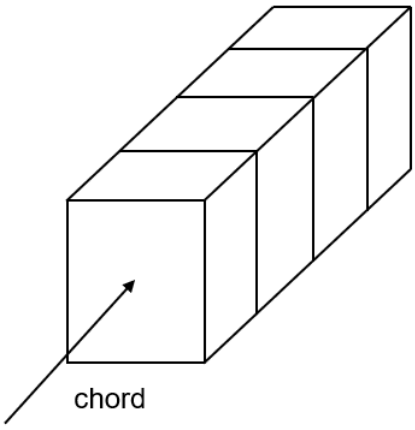



Figure 2.6: Schematic of an hex-mesh as an interlaced stack of elements [33].

Often times, such requirements are overcome by using a combination of hexahedral and tetrahedral with tie constraints at the interface. This technique, although it makes the meshing possible, it increases the local error and rigidity at least locally [32]. It is also found that such non-conformal meshes usually fail at the interface during mesh seed propagation for highly complex structures. Additionally, the order in which the regions are being meshed also affect the final success or failure of mesh generation. Due to the many complexities and factors involved, the generation of an all-hex mesh is still in research stages, and at present, no universal algorithm exists for automatic hex-mesh generation that works robustly for all finite volumes. Currently, a 'divide and conquer' approach to hex meshing is widely utilized, where a part is decomposed into smaller pieces, each of which is easier to mesh than the original part. Since most hex-mesh algorithms are still limited in scope, decomposition of the geometry such that it conforms to the requirements

of the available meshing tools remains a fundamental step.

Bohm *et al.* [34] propose a 5-step strategy modular hex meshing strategy that involves: identification subassemblies in a model, classification of subassemblies in terms of their potential for automation, simplification of features of the geometry, decomposition of the geometry and then hex-meshing. This strategy relies on human expertise and automation of simpler steps like mesh seeding on relevant geometry entities. However, once determined by an human expert, the geometry simplification, volume decomposition scheme can be automated for a particular complex structural geometry. Their work fails to describe an efficient strategy for performing volume decomposition of geometries as well as its automation. Lu *et al.* [35] proposed an approach for classifying edges of the model into different loops (Fig. 2.7) in order to determine cutting surfaces. These loops, although advantageous, cannot guarantee the best decomposition [8]. In another work, Tam *et al.* [36] utilized mid-surface calculation to decompose a target 3-D domain into several types of simple sub-regions. Work by Sun *et al.* [8] describes an automatic identification of thin-sheet regions and cutting faces based on interrogation of face pairs which sets of opposing faces bounding a thin-walled region. This work although reduces the effort significantly, it is still partially automated and needs development for robustness. This problem gets more challenging when multiple parts are connected to form a large complicated assembly. This is because the shared surfaces now need to have a conformal mesh on both sides [33].

The computational modeling efforts are increasingly moving towards significant use of computational analyses and there is an urgent need to accelerate the design process. Thus, an integrated design process is needed which constructs parameterized FE models and allows modifications to the underlying geometric models without requiring a completely new effort to generate a quality mesh [33]. This work presents a blueprint for a robust

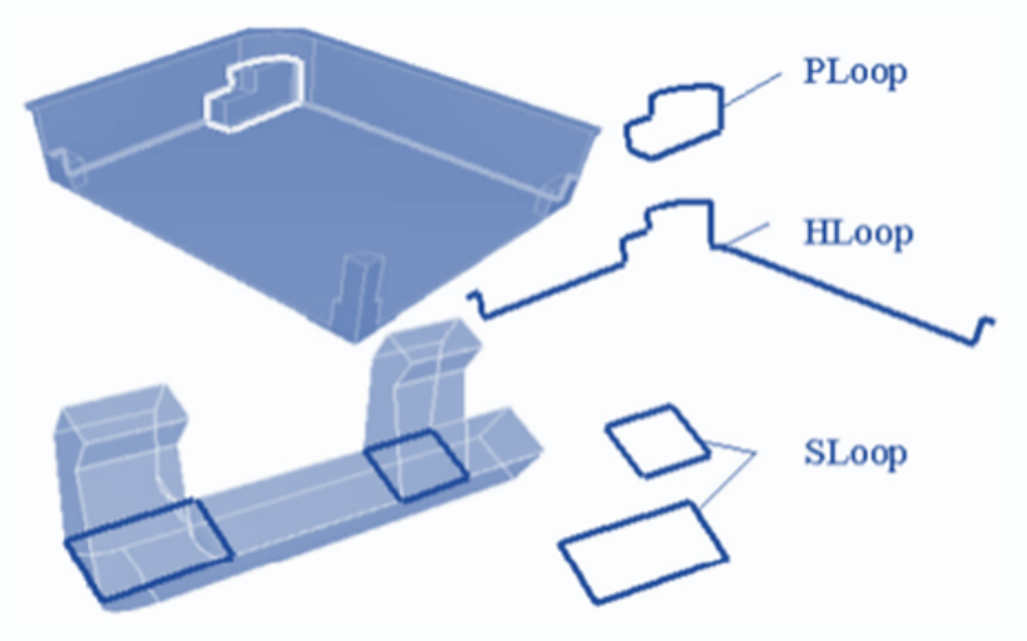



Figure 2.7: Example of decomposed primitives using different loops [35].

hexahedral mesh generation using strategic volume partitioning for complex assembly models. This partitioning scheme can be utilized to developed a fully-functional algorithm to automatically identify partitions and decompose complex geometries.

## 2.5 Expert Systems for Finite Element Modeling

Development of an optimal multi-fidelity model also requires crucial *a-priori* modeling choices to be made regarding its domain-wise fidelity. A theoretical reasoning is needed to develop accurate and cost-effective models. Since the cost of a FE analysis is dependent on the continuity requirements and the total degrees of freedom, an intelligent decision about whether 3-D, 2-D or 1-D elements are appropriate is needed [7]. Many times, these modeling decisions can be quite subjective in nature and lead to significant analyst-to-analyst variability in critical modeling choices, which in turn leads to

considerable differences in obtaining reliable engineering solutions. Glickman and Romero [37] demonstrated the significant effect of analyst-to-analyst variations in the development of computational models, and the need for efficient ways to eliminate these variations. This is especially important in complex structures with multiple connections, where a shell model would be insufficient but a full solid model is computationally expensive. The ideal procedure is to only use 3-D solid elements where it is critical to capture the details of the transverse structural response and to model the remaining thin-walled areas (often the largest portion) with shell elements [38]. However, the problem of determining element topologies in a multi-fidelity model is itself fuzzy in nature and depends strongly on the context of the problem [39]. Moreover, this choice becomes additionally vague during the preliminary design of complex engineering structures[40]. This leads to considerable iterations and adjustment of model development in industry and is still a major bottleneck. In the review work by Naganarayana and Prathap [7], the need for an expert system in modeling complex structures is emphasized. A decision support system that recommends optimal modeling choices can aid in removing this subjectivity and thereby help analysts make confident choices.

The need for an expert system to make finite element modeling decisions has been identified in the literature. Novak *et al.* [41] presented a summary of the state-of-the-art structural analysis based design improvements into a collective advisory intelligent support system. This work demonstrated the usefulness of such expert systems in the design of new products or as an educational tool. Breitfeld *et al.* [42] developed an expert system using a knowledge base and object oriented programming for the verification of finite element analysis. In another work, Dolsak *et al.* [43], proposed an intelligent decision support for structural design analysis and emphasised the usefulness of such an intelligent advisory tool. Another advisory system was developed by Hetey *et al.* [44], for reliable FEM

modeling in aerospace. This tool involved global as well as specific knowledge rules in the form of dialogue boxes for selecting element types, boundary conditions etc. A knowledge database was prepared by subject matter experts for these works. For complicated geometries, it is not straightforward to develop a knowledge base.

Some researchers have focused on building data-driven expert systems using machine learning techniques like Artificial Neural Networks (ANNs). Yeh *et al.* [45], presented an expert system for debugging FEM input data using ANNs and showed that ANNs can work sufficiently as a knowledge acquisition tool. However, a major limitation of Yeh *et al.*'s work was that the ANNs that they employed lack the ability to explain their reasoning, which is critical for establishing user confidence. Another work by Li *et al.* [46] presented a hybrid expert system for finite element modeling of fuselage frames. Although they utilized ANNs for classifying different frame structure types, the knowledge base was developed manually using expert knowledge.

There is a significant demand for an expert system framework which can learn rules automatically and also be consistent with physics of the problem to provide intelligent support for complex FE multi-fidelity modeling decision making. This work presents an expert system framework using Fuzzy Logic and Decision Trees (DTs) for recommending the optimal modeling choices for developing multi-fidelity models. Fuzzy logic, first introduced by Zadeh [47], has gained popularity due to its successful implementation in application areas of automatic control, expert systems, fault diagnostics, decision analysis etc. Fuzzy systems have shown success with treating imprecise and subjective information into the decision making of engineering systems[40, 48]. There are several instances when the available information for structural analysis using FEM is uncertain. Examples include vagueness regarding structural dimensions, loading conditions, external environments, and strength of an inter-laminar bond in composites [7]. Work by Manetviz *et al.* [49] on

utilizing soft computing for automating FEM is worth mentioning. The authors argued that the overall intelligent FEM package is a 'test-bed' for many fuzzy computing techniques.

## 2.6 Fuzzy Logic

Fuzzy logic, as opposed to binary logic, deals with approximate reasoning, which is the most common mode of human thinking. In classical theory, a *crisp* set $A$ is defined as a collection of objects $x \in X$ that is finite and each object can either belong to or not belong to the crisp set $A$. On the other hand, a *fuzzy set* $\tilde{A}$ in $X$, where $X$ is a collection of objects can be denoted as

$$\tilde{A} = (x, \mu_{\tilde{A}}(x) | x \in X)$$

where $\mu_{\tilde{A}}(x)$ is the membership grade of $x$ in $\tilde{A}$ ([50]).

Thus, using a notion of degree, fuzzy logic provides flexibility in terms of making decisions by taking into account uncertainties and vagueness. For example, as shown in Fig. 2.8., a temperature point can belong to multiple sets, 'cold', 'warm', and 'hot' in varying degrees between 0 and 1. Thus, the variables are assigned membership to different sets and utilize a linguistic style which makes it easier to interpret and follow human thinking.

### 2.6.1 Fuzzy Inference System

Fuzzy logic converts the fuzzy sets into linguistic variables such that the rule base can be represented in the form of 'IF..THEN..' rules. This enables the development of a Fuzzy

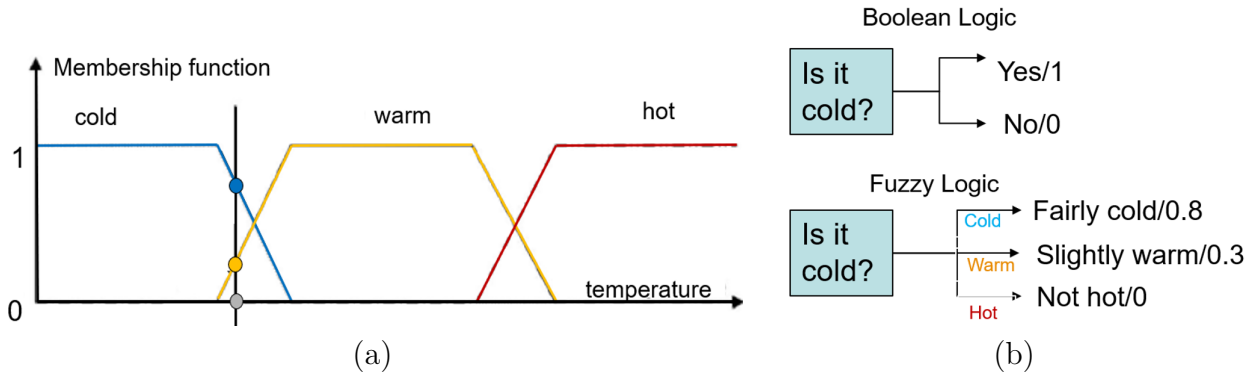



Figure 2.8: Example of fuzzy logic: (a) a temperature point can simultaneously be cold, warm or hot in varying degrees; (b) comparison of Boolean logic versus Fuzzy logic.

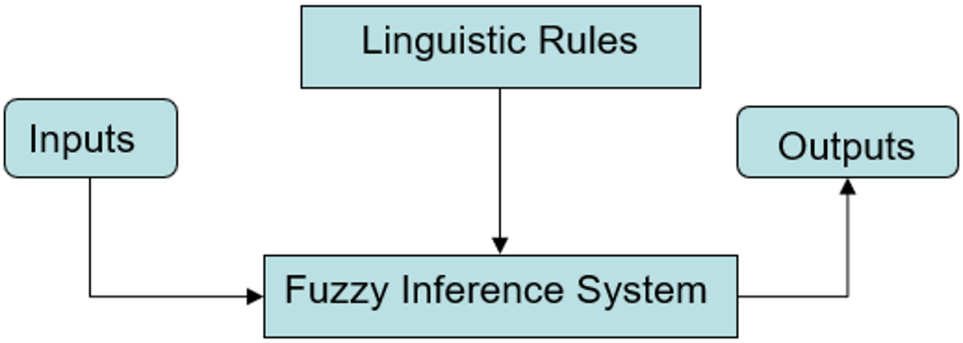



Figure 2.9: Schematic representation of a Fuzzy Inference System (FIS).

Inference System (FIS) that has been shown to demonstrate success in the field of expert systems. Fuzzy Inference systems are one of the most popular applications of fuzzy logic in the field of decision support tools, control systems etc. These systems are powerful because of their ability to handle linguistic and fuzzy rules as well as their efficiency in performing complex mapping between inputs and outputs. Out of the different FIS types available, Mamdani Fuzzy Inference System (FIS) is a control system that can make conclusions blending together a set of linguistic control rules obtained from human experts. These systems become very popular in dealing with uncertain and imprecise data which is commonly encountered in real-life engineering applications. Imprecision herein implies

vagueness instead of the lack of knowledge about the value of a parameter. The applications range from industrial processes to consumer products to decision support systems. In this work, a *Mamdani* fuzzy inference system is utilized. Development of fuzzy inference system has three main features: (a) input/output membership functions and (b) rule-base and (c) fuzzy logic operations. A schematic of the inference process for a *Mamdani-type* is shown in Fig. 2.9.

The inference process of a *Mamdani* FIS consists of the following steps:

(a) The crisp input values are fuzzified using the membership functions of each linguistic label.
(b) Identify the rules in the knowledge base that are activated based on the input value and its strength of membership in each membership function.
(c) Activate the corresponding rules and determine the consequence of each rules.
(d) Aggregate the qualified consequent to produce a crisp output.
Rao *et al.* [51] developed a fuzzy logic-based expert system to predict the results of finite element analysis by using element size and shape ratio as inputs. Their work showed that fuzzy logic can predict FE results within reasonable accuracy. Their approaches utilized manual knowledge base construction as well as Genetic Algorithm (GA) based rule extraction. The GA knowledge base although data-driven, is both dependent on the encoding of the rules and is computationally expensive. Another work by Sangiovanni *et al.* [52] presents a fuzzy logic based finite element mesh design. The authors emphasized the benefit of learning using learning algorithms such that the rules can be adapted to different problems, as compared to the classical mesh adaptive methods which are sensitive to changes in the geometry.

## 2.7 Decision Trees

Decision Trees (DTs) are a supervised Machine Learning (ML) technique that have gained popularity for classification tasks, *i.e.*, assigning labels to unlabeled data instances. This classification technique predicts a label for a new data point by following a series of 'IF-ELSE' statements about the values of the features of the instances. These statements split the design space iteratively into smaller units until a satisfactory decision can be made. Such a simple approach of dividing the design space successively is very powerful in representing linear as well as non-linear decision boundaries. Because of the inherent algorithm of DTs, there is transparency in the decision-making process as opposed to the 'black-box' nature of most other ML techniques. More importantly, this transparent tree-like structure can be readily converted into interpretable data-driven rules in the form of 'IF-ELSE' statements by following the tree-like structure from top to bottom. These resulting rules follow human-like thinking which makes the decision process easy to visualize as well as interpret. Moreover, decision trees also aid in identifying the dominant features in a prediction problem based on the sequence in which the features appear in the tree.

### 2.7.1 Training a Decision Tree

Consider a simple design space as shown in Fig. 2.10(a) with two attributes $x_1$ and $x_2$, which needs to be classified into 'red' and 'blue' labels. An example of a trained decision tree for this data-set is shown in Fig. 2.10(b). A decision tree has three different types of nodes: a)Root node: at which an attribute split is made; b) Internal nodes: which are derived from the splits made at previous/top nodes; c) Leaf/Terminal nodes: which perform the final classification task. The tree performs the splits at the numerical values,

namely at $x_1 = 3.35$ and $x_2 = 3.2$. The decision about where to make a split is based on an impurity measure which is discussed in the subsequent paragraphs. It can be observed that the tree arrived at a prediction by dividing the design space using cascading 'IF-ELSE' block conditions from the root node to a terminal node. Note that the tree makes the predictions based on the most dominant class and ignores the less frequent values as shown in Fig. 2.10(c).

### 2.7.2 Attribute Splitting Criteria

A decision tree algorithm divides the decision space successively into purer subsets. This is done by making a split at an attribute value to create divisions in the design space. The choice of the best criterion to split on is made by using the concept of impurity measures, which provide the goodness of a split.

#### Gini Index

Gini index is a popular impurity measure, which is calculated as follows:

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i|t)]^2 \tag{2.1}$$

where, $c$ is the number of labels or classes and $p(i|t)$ is the fraction of records belonging to class $i$ at a given node $t$.

From Fig. 2.10(b), we can see that at the root node, the fraction of the blue class is 17/29 and that of the red class is 12/29. Thus, Gini index at the root node is

$$Gini(t) = 1 - \frac{17}{29}^2 - \frac{12}{29}^2 = 0.485 \tag{2.2}$$

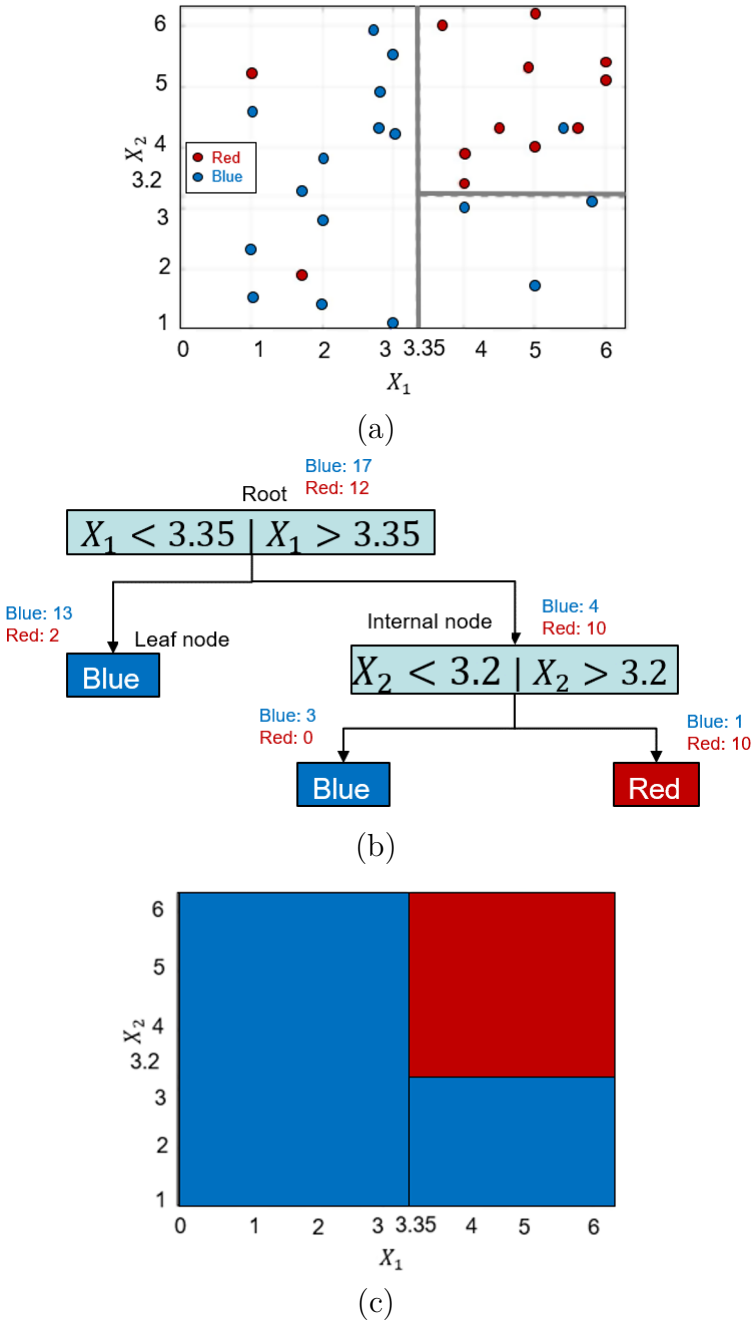



Figure 2.10: Example of two dimensional design space classified into labels using DTs: (a) two-label design space; (b) trained DT; (c) final outcome using DT.

A value of 0.5 implies that the class distribution is very much impure and a value of 0 and 1 implies a pure class. The goal of the decision tree algorithm is to perform splits that

result in purer sets. After the split, the Gini index becomes 0.23 and 0.41 for the new nodes formed. This process is continued until all/most the instances belong to the same class.

### Interaction-Curvature Test

The standard Classification and Regression Tree (CART) algorithm uses the Gini index. However, it has a tendency to develop bias in the presence of numerical/continuous variables. As such, other important predictors could be missed. In order to reduce this bias, Matlab [53] recommends using 'Interaction-curvature' based split predictor selection technique in the presence of continuous or a combination of continuous and categorical features exist. The 'Interaction-curvature' test performs Chi-square test of independence between each predictor and the response. If the *p-value* of this test is less than a threshold, the feature is considered to have an effect on the response and is selected to be split further. The most important feature interacting with the response variable is split first. This technique has the capability to reduce the bias and enhance the tree interpretation by also providing the level of predictor importance. In this work, the 'Interaction-Curvature' based split criterion is used, as there are continuous-valued and categorical features to be used and tree interpretation is important.

### Advantages and Disadvantages

Decision tree classifiers are computationally inexpensive as well as easy to interpret and visualize for small sized trees. This converts a classification problem into a simple graph and is particularly useful where interpretation of a decision is important. DTs can also handle a combination of categorical as well as numerical features. These tools are also robust to outliers and usually make decisions based on dominant features in the design

space.

Decision trees are, however, prone to over-fitting if allowed to grow very deep. it can be mitigated using different available techniques like pruning etc. Moreover, if the tree grows very deep, it becomes very hard to interpret and thus loses its usefulness. Typical decision trees create decision boundaries involving only a single attribute. However, there are advanced decision trees like oblique decision trees that can perform splits involving multiple attributes.

## 2.8 Proposed Methodologies

### 2.8.1 Expert System Framework: DT-based FIS

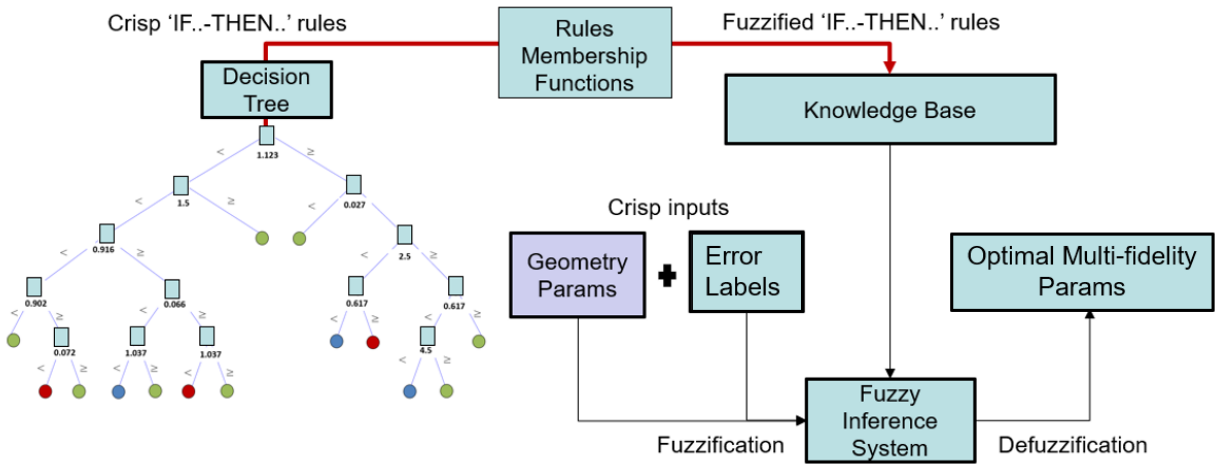



Figure 2.11: Schematic representation the proposed DT based FIS for making optimal multi-fidelity modeling recommendations.

Decision trees have been widely used for extracting rules and performing classification for different applications. As compared to black-box classification and prediction models like ANNs and Support Vector Machines (SVMs), decision trees have the advantage of being

easy to interpret and also visualize in a tree structure. More importantly, decision trees directly learn rules from data in the form IF-THEN blocks which can then be easily utilized to develop a fuzzy inference system. The learning ability of decision trees to provide rules for fuzzy systems, which utilize high-level human-like reasoning, has been proved to be effective and easier to interpret.

The concept of utilizing DTs for generating a knowledge base for a FIS has been proposed and effectively used in the past [54, 55, 56, 57] for different applications. Until now, significant work has not been performed to develop an expert system that provides optimal element topology recommendations for multi-fidelity models. A schematic of the decision tree based fuzzy inference system for making optimal modeling domain-fidelity recommendations is demonstrated in Fig. 2.11.

### 2.8.2 Proposed Overall Framework

An integrated design process with generation of adaptable multi-fidelity topology creation, meshing and post-processing using parameterization and FE package scripting is proposed for automated data generation. Recognizing that complex models need strategic volume decomposition to enable a conformal mesh possible, a foundational blueprint based on heuristic rules is presented. Furthermore, a data-driven expert system based on Decision Trees (DT) and Fuzzy Inference System (FIS) is proposed as a tool to automatically generate an interpretable knowledge base that provides optimal modeling choices as recommendations. A broad summary chart of the proposed techniques is shown in Fig. 2.12. During the preliminary stages, a preliminary dataset for multi-fidelity models can be developed in an automated sense using the presented techniques in this dissertation. The generated data can be utilized for training the proposed expert system. This expert

system can then be implemented to develop optimal multi-fidelity models that could then be employed for accelerated design space explorations.

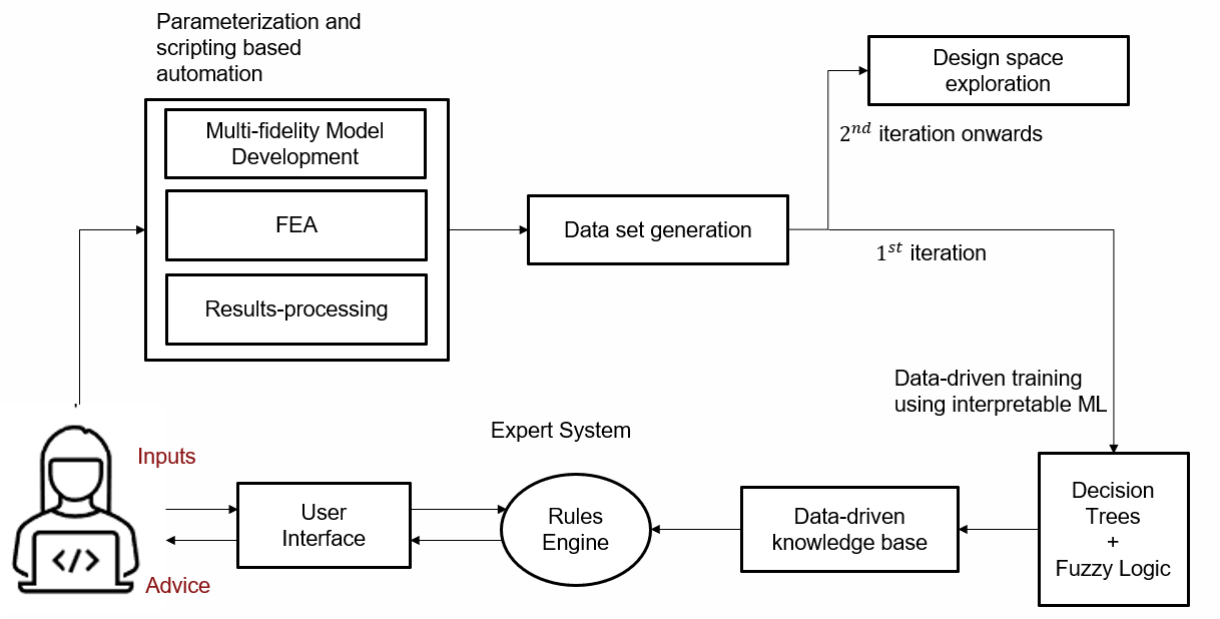



Figure 2.12: Block diagram representation of the proposed framework for accelerated analyses of preliminary design spaces using multi-fidelity FE modeling.

# Chapter 3

# Automated FE Analysis of a Stiffened Tank Pressure Vessel using Shell-Solid Multi-Fidelity Modeling

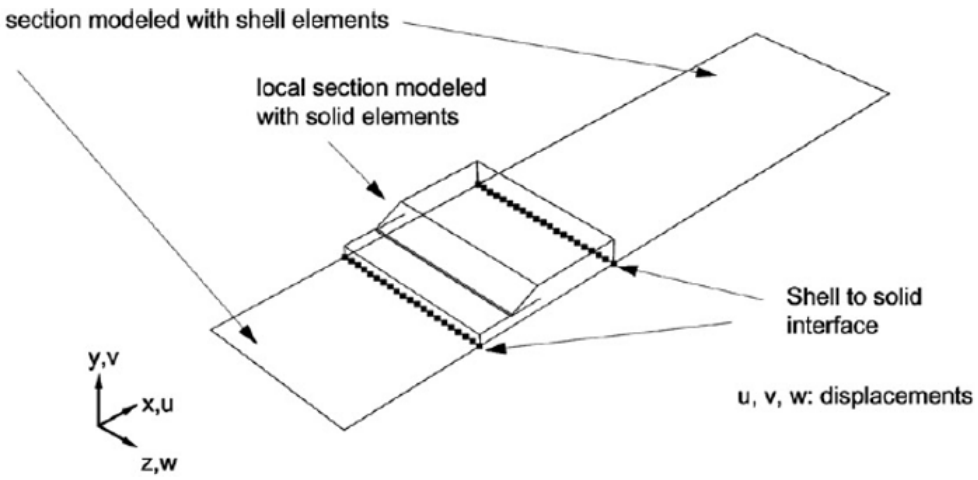
This chapter presents a manuscript that will be submitted to the Computers and Structures Journal and is currently under review by the US NAVY for public disclosure.

[1]

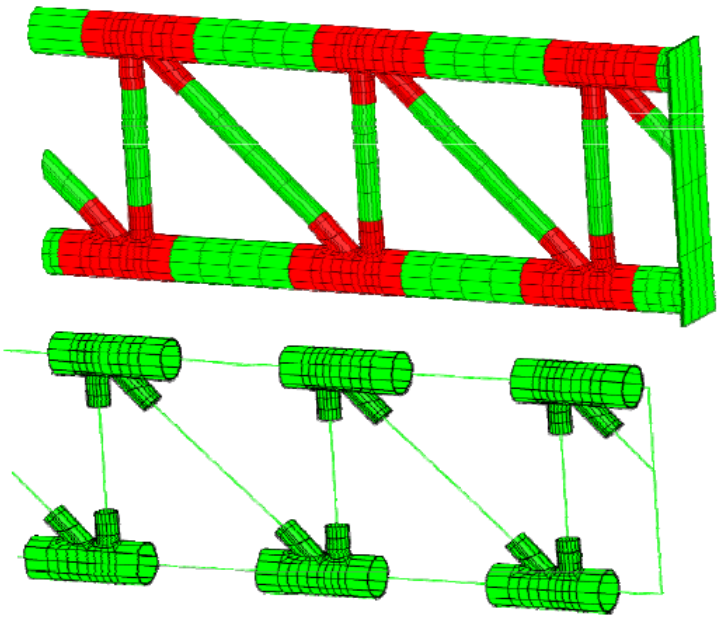
[1] **Palwankar, M.**, Kapania, R.K., Hammerand, D., "Automated FE Analysis of a Stiffened Tank Pressure Vessel using Shell-Solid Multi-fidelity Model," Computers and Structures Journal (submission in progress).

## 3.1 Abstract

Stiffened thin-walled pressure vessels are widely used in the industry, especially in aerospace, marine, automobile and chemical industry. Accurate structural analysis of such complex structures under external and internal pressure loads is key to making critical design decisions during the conceptual and preliminary stages. With advancements in adaptive optimization frameworks, multi-fidelity finite element (FE) models (coupled shell-solid) are increasingly sought in the early stages of design development for accurate and efficient analysis. However, the time and effort required to create such multi-fidelity FE models with acceptable meshes for highly complex structures is still significant and is a major bottleneck in the FE modelling. Thus, automation of FE model creation and mesh generation is critical for advancements in technology. A major hurdle in automating highly complex structural FE models is the generation of robust meshing. This paper presents a framework for the automation of multi-fidelity model development from the solid and shell models using parameterization and scripting within the commercial FE packages. Additionally, issues related to the automated meshing of highly complex assemblies is discussed, and a volume decomposition scheme is proposed based on face partitions. A comparison of the all-solid, all-shell and different multi-fidelity models of a highly complex stiffened thin-walled pressure vessel under external sea pressure and internal tank pressure is presented. Results reveal that automated generation of mixed models in an integrated fashion including the geometry creation, meshing and post-processing is capable of significant computational savings and reducing considerable interactive effort involved in analyzing complex structures.




(a)



(b)

Figure 3.1: Examples of multi-fidelity models: (a) a shell/3D model utilized by Krueger *et al.* [9] for analyzing debond specimens.; (b) a typical offshore platform tubular structure model with 3D elements near the critical joint regions and beam elements elsewhere [21].

## 3.2 Introduction

Real world engineering structures often utilize thin-walled stiffened pressure vessels, *e.g.*, aircraft fuselages, ship vessels, submarines etc. These structures are often analyzed using the Finite Element Method (FEM), which has become an indispensable tool in the aerospace and ocean industry. Due to advancements in computational speeds over the last decade and with the growing need for adaptive optimization, FEM is being increasingly sought during the preliminary and conceptual design phases. Generally, at these stages, a variety of configurations are studied and the goal is to come up with a smaller set of best possible designs which can be studied in some detail to finalize the design. In such cases, it is beneficial to utilize reduced-order models that provide high computational efficiency along with sufficient accuracy. For instance, multi-fidelity finite element models (coupled shell-solid), a form of a reduced-order model, can be employed to utilize the accuracy of solid elements in the regions of interest and efficiency of shells everywhere else in the model. This way, accurate solutions can be obtained at a lower computational cost. Alternate techniques like sub-modeling or a global-local approach need the conversion of rotation/moments from the global analysis to the local sub-model analysis. This necessitates a two-step process which can get difficult and time consuming during the design stages where several configurations need to be studied. On the other hand, a shell-solid coupling method requires a single step analysis and is thus often suitable at the design stages. In particular, large structures, such as underwater submarine hulls, are simultaneously acted upon by external sea pressure and internal loads which make the determination of boundary conditions for the local FE model difficult. This makes single-step solid-shell models preferable [12]. Hence, the use of multi-fidelity modeling is

being increasing sought in recent times for different structural applications.

Multiple studies in the past have focused on proposing coupling techniques for appropriate constraints at solid-shell transition of multi-fidelity models [10, 12]. Ikushima *et al.* [16], utilized a shell-solid mixed finite element model to study the nonlinear mechanical behavior of thin-plate complex structures. Krueger *et al.* [9, 14] proposed a mixed finite element model (Fig. 3.1(a)) for the analysis of composite skin-stiffener debond specimens. Lee *et al.* [21] demonstrated an example (see Fig. 3.1(b)) of a multi-fidelity model for a tubular joint structure commonly modeled using a combination of 1-D and 3-D elements. In other works, Krueger *et al.* [13] showed the effectiveness of a shell/3-D technique for the analysis of delaminated composite laminates. Peric *et al.* [58] investigated residual stresses and distortions due to welded T-joints using shell/3-D modeling. These works proposed novel shell-solid coupling techniques and/or assessed mixed-fidelity models for a particular application which involved relatively simpler models. Often, the decision about selecting the element topology in a mixed model is subjective and relies heavily on the analyst's experience. Many times, the analyst makes multiple mixed model configurations to choose the most optimal mixed model. This in turn leads to significant on-going adjustment of model development in industry and is still a major bottleneck. Figure 3.2 shows an example of the development process of a multi-fidelity model where the effect of the solid region was investigated.

Such trial and error investigation of multi-fidelity modeling topology is time consuming and is one of the leading reasons for schedule delays in the industry. As complex computational simulations with adaptive optimization are on the rise, it is becoming crucial to automate the finite element model development [18]. For instance, studies [19] in the past have demonstrated the dramatic variations during a multi-disciplinary design process in the planform shape of a supersonic aircraft design depending on the fidelity level

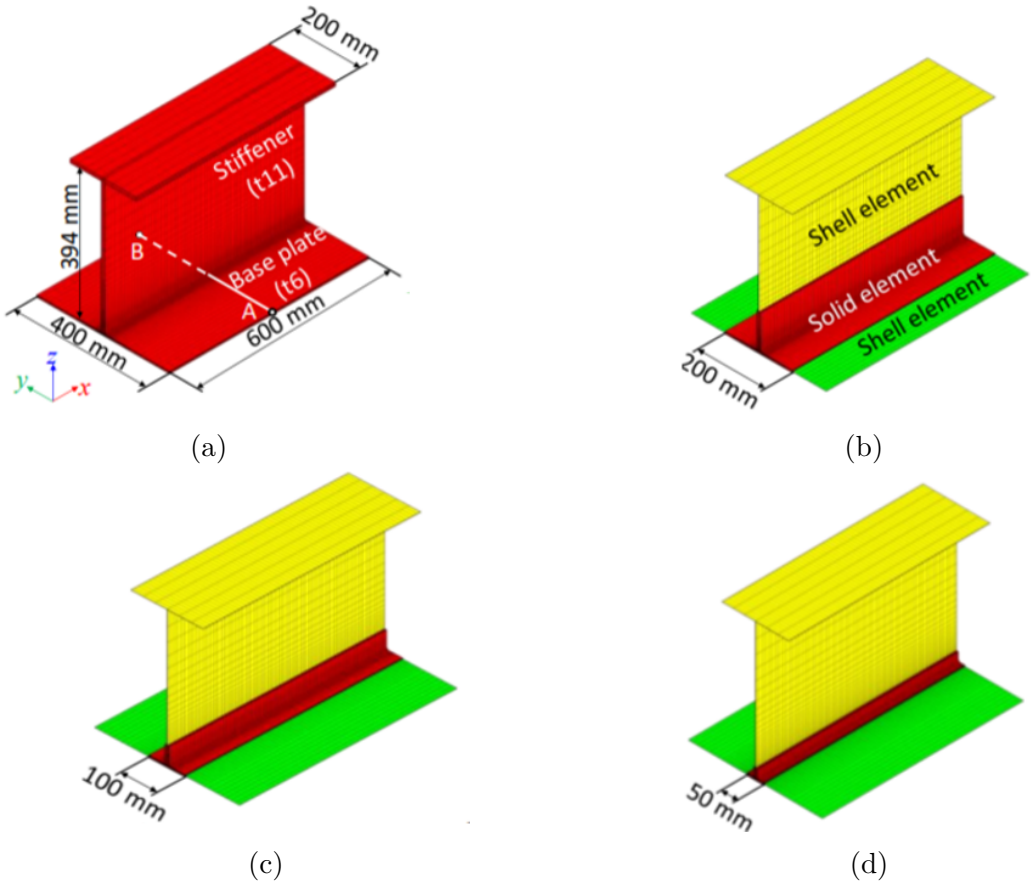
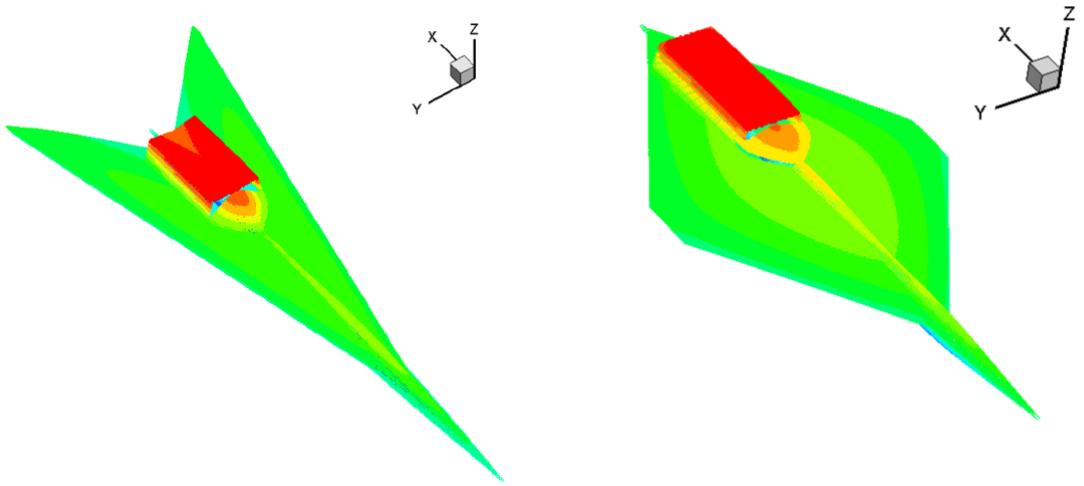



Figure 3.2: Multiple multi-fidelity models investigated by Ikushima *et al.* [16] for understanding the effect of solid region dimension $B_s$ on the analysis accuracy; (a) full solid model; (b) 200 mm solid region at the joint; (c) 100 mm solid region at the joint; (d) 50 mm solid region at the joint.

of optimization. Fig. 3.3 shows the (a) highly swept wing configuration yielded by a low-fidelity optimization as compared to (b) a trapezoidal wing shape yielded by a higher fidelity optimization process. This emphasizes the importance of using higher fidelity simulations during optimization processes and the need for an integrated and automated tool for building different finite element models. Additionally, recent work by Xie *et al.* [20] presents a certification-driven platform for air frame early preliminary design which emphasizes on the need to perform a large design space exploration when numerous tradeoffs between conflicting objectives are involved. In such situations, an automated

development of multi-fidelity models, intended to be used in a multi-disciplinary optimization process can provide the high-fidelity design concepts at a significantly lower computational cost. Such an automated process could also help in rapid design space exploration and development of accurate data-driven surrogate models.




(a) (b)

Figure 3.3: Varying design concept results for an efficient supersonic air vehicle based on different levels of optimization fidelity; (a) low-fidelity optimization and (b) higher fidelity optimization [19].

In the past, medial axis transforms [21, 22] and prominent cross-sections [23] have been utilized to automatically generate mixed dimensional models. However, the methods utilized in all these works are still under research and not applicable for highly complex geometries. More recently, parameterization coupled with in-built scripting capabilities of the commercial FE packages has been utilized to automate generation of FE models. Work done by Sohaib [24] presented a parameterization based model development of a generic wing model using CATIA V5 software. Lai *et al.* [25] performed parametric studies on different types of composite and cylindrical shells using parameterized FE models coded by

Patran Command Language (PCL). In another work, Zhang *et al.* [26] performed optimization of aircraft mid-fuselage structure based on parametric modeling design. Benaouli *et al.* [27] automated the integration of CAD/CAE systems for efficient parametric designs of aircraft wing structures using the built-in programming languages and model exchange capacity of modern computer aided tools. Work by Love *et al.* [28] focused on integrating aircraft parametric structural and outer mold line geometry modeling to enhance the design process at the critical early conceptual design. In another recent work, Liu *et al.* [29] utilized a parameterized approach to establish a high fidelity 3-D finite element model of involute helical gears by performing all the procedures in finite element software. This greatly simplified the software environment and avoided data loss due to model delivery.

In addition to the geometry creation, mesh generation is a major task in the FE model development procedure that needs to be automated to achieve a smooth design exploration process. Hexahedral meshing remains widely accepted due to its superior performance [8]. This is because hexahedral meshes provide shape functions with additional terms that may increase the accuracy. Studies performed in the past [30, 31] on assessing the performance of hexahedral meshes versus tetrahedral meshes showed that tetrahedral meshes can be mathematically stiffer for high deformation analysis with linear elements. Additionally, the use of a hexahedral mesh decreases the overall element count as compared to a tetrahedral mesh. However, there are limitations in achieving a fully automated hexahedral mesh for complicated parts due to additional constraints that come into play due to the hexahedral shape. One of the critical limitations of hex-mesh generation is that for any face of the hex, there must be an opposite face. The mesh can be viewed as an interlaced stack of elements (Fig. 3.4) such that each stack must begin and end at a boundary or be a closed loop of elements [32]. This makes the auto-generation of a conformal hex mesh challenging. As it

follows, any refinement of one element continues to propagate to its neighbors.

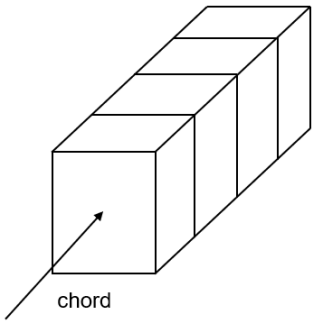



Figure 3.4: Hex-mesh is an interlaced stack of elements [33].

Alternatives exist to overcome this challenge by using a mixed mesh (a combination of hexahedral and tetrahedral mesh), or definition of tie constraints in the regions. However, although such non-conformal mesh interfaces may decrease the meshing complexity, they increase the error and rigidity at least locally [32]. Moreover, such non-conformal meshing often fails at the interface during mesh seeds propagation for highly complex structures. Additionally, the success and failure of a region in terms of meshing also depends on the order of meshing regions. The generation of an all-hex mesh is still in research stages, and at present, no universal algorithm exists for automatic hex-mesh generation that works robustly for all finite volumes. Currently, a divide and conquer approach to hex meshing is widely utilized, where a part is decomposed into smaller pieces, each of which is easier to mesh than the original part. Since most hex-mesh algorithms are still limited in scope, decomposition of the geometry such that it conforms to the requirements of the available meshing tools remains a fundamental step. Bohm *et al.* [34] propose a 5-step strategy to semi-automate the meshign process: (1) identify subassemblies in a model, (2) classify them in terms of their potential for automation, (3) simplify features of the geometry, (4) perform decomposition and then (5) perform hex-meshing. This strategy relies on human expertise and automation of simpler steps like mesh seeding on relevant geometry entities.

However, once determined by an human expert, the geometry simplification, volume decomposition scheme can be automated for a particular complex structural geometry. Their work fails to describe an efficient strategy for performing volume decomposition of geometries as well as its automation. *et al.* [35] proposed an approach for classifying edges of the model into different loops (Fig. 3.5) in order to determine cutting surfaces. These loops, although advantageous, cannot guarantee the best decomposition [8]. In another work, Tam *et al.* [36] utilized mid-surface calculation to decompose a target 3D into several types of simple sub regions. Work by Sun *et al.* [8] describes an automatic identification of thin-sheet regions and cutting faces based on interrogation of face pairs which sets of opposing faces bounding a thin-walled region. This work although reduces the effort significantly, it is still partially automated and needs development for robustness. This problem gets more challenging when multiple parts are connected to form a large complicated assembly. This is because the shared surfaces now need to have a conformal mesh on both sides [33].

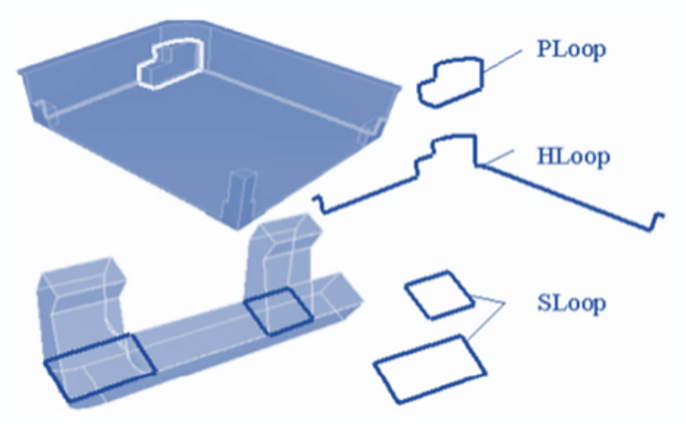



Figure 3.5: Example of decomposed primitives using different loops [35].

The computational modeling efforts are increasingly moving towards significant use of analyses and there is an urgent need to accelerate the design process. Thus, an integrated design process is needed which constructs parameterized FE models and allows

modifications to the underlying geometric models without requiring a completely new effort to generate a quality mesh [33]. However, to the authors knowledge, there is limited work in the literature demonstrating the automation of a complex structure with different fidelity models, especially a multi-fidelity model with variable element topology, including integrated model generation, meshing and post-processing of the results. Recognizing that complex models need strategic volume decomposition to enable a conformal mesh possible, a foundational blueprint based on heuristics-based rules is presented. This flowchart is presented with the aim to provide a useful base for what are some effective versus ineffective types of volume decompositions to perform. This could then be used to complement a general partitioning scheme, such as the CLoops to obtain a robust volume decomposition scheme.

The paper demonstrates the automated development of full solid, full shell and mixed (coupled shell-solid) models with variable topology. A highly complex assembly of a tank structure within a stiffened hull is chosen as an example, assessed for the use of mixed fidelity model and various insights gained through the automation of this complex assembly are discussed.

In particular, the paper makes the following contributions:
1) Automatic creation and post-processing of multi-fidelity models using parameterization and scripting within FE packages.
2) A blueprint for strategic volume decomposition of assembly models for robust hexahedral mesh generation.
3) Assessment of the multi-fidelity modeling technique for a realistic, highly complex stiffened tank structure subjected to static external hydrostatic pressure as well as internal tank pressure.
4) Discussion of major issues arising and insights gained related to development of

automation scripts.

## 3.3 Problem Description

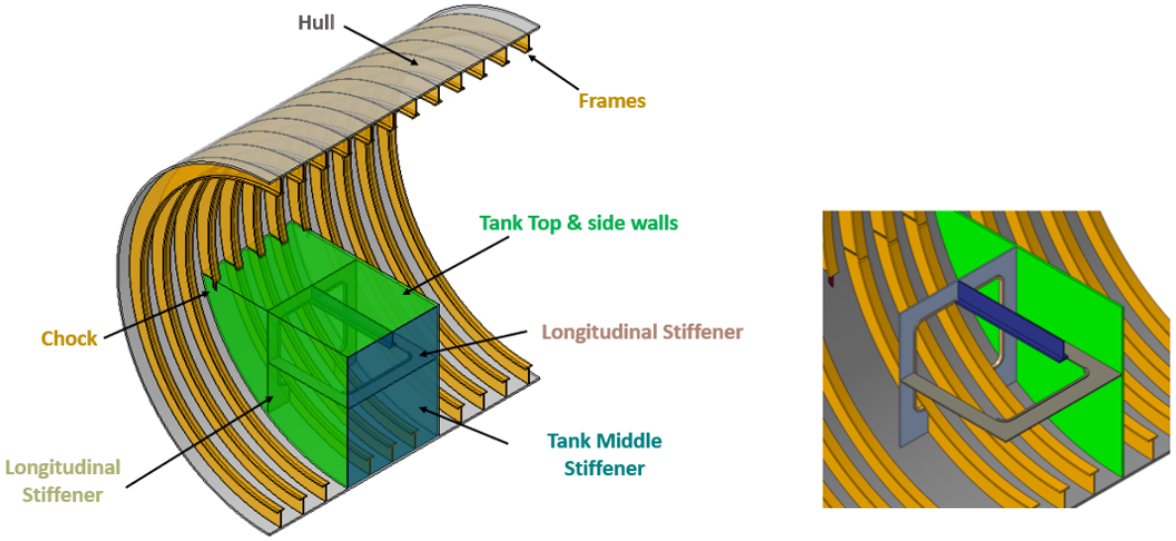


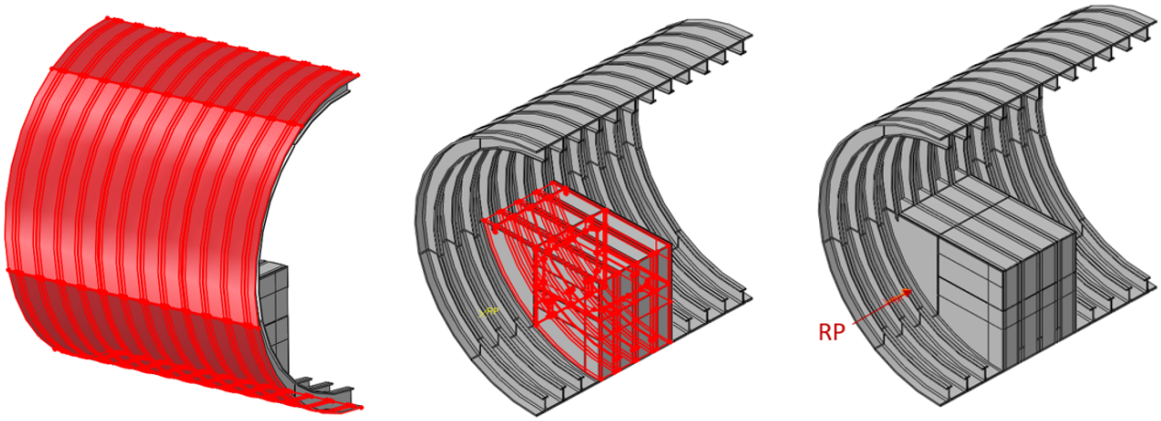

Figure 3.6: Stiffened pressure hull tank with internal stiffeners; (a) assembly, (b) zoomed-in view of the internal stiffener connections inside the tank.

A large stiffened tank pressure vessel (Fig. 3.6) with internal tank stiffeners is considered for this work as a reference. A standard steel material commonly employed for submarine structures is utilized. The properties are given in Table 3.1. It is to be noted that all the material response was considered linear elastic and below yielding. Secondly, all the welds in the component connections were assumed to be perfect.

This structure is subjected to an external sea pressure and an internal tank pressure (Fig. 3.7). A sea depth of 240 m and an internal tank pressure of $689,500$ Pa (100 psi) is applied. Since the submarine hull is truncated axially, the longitudinal stress in the hull is

Table 3.1: Generic steel material properties.

| Item | Value |
|---|---|
| $E$ | 200 GPa |
| $\nu$ | 0.30 |
| $\rho$ | 7,850 kg/m$^3$ |




(a) (b) (c)

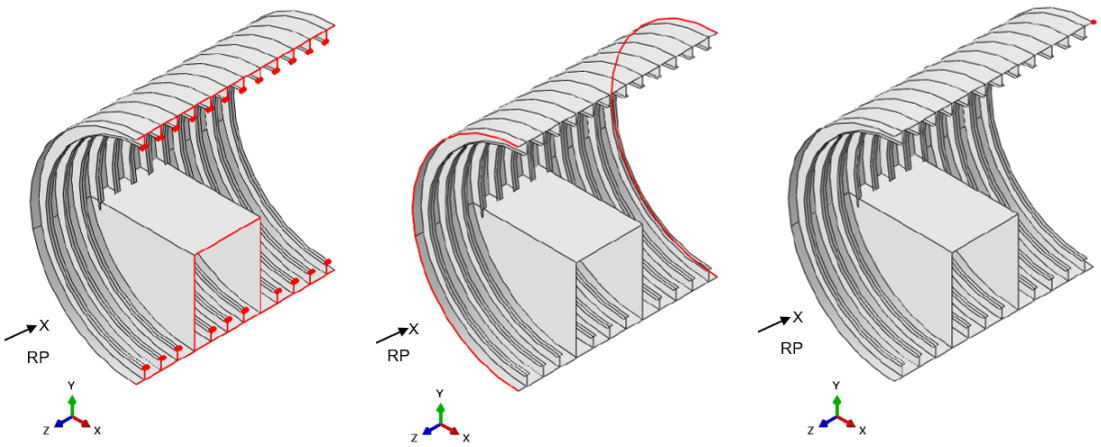
Figure 3.7: Applied loading conditions on the tank geometry: (a) external Sea Pressure; (b) internal tank pressure; (c) axial load to account for sea pressure on unmodeled submarine ends.

applied using a reference point that controls the relative axial displacement between the two Z-face ends in the model (Fig. 3.7) (c). The magnitude of the axial load in units of force is given by:

$$F_{axial} = -\frac{1}{2} p_{\text{axial}} \pi R_o^2 \tag{3.1}$$

where $R_o$ is the outer radii of the hull and $p_{axial}$ is the axial pressure

Taking advantage of symmetry, two symmetry conditions are applied to the tank model. A symmetry condition corresponding to the $X = 0$ plane is applied to the tank half model




(a) (b) (c)

Figure 3.8: Applied boundary conditions on the tank geometry: (a) x-symmetry; (b) periodic boundary condition in Z; (c) no displacement in Y and Z direction for model verification (reveals non-equilibrium).

(Equation 3.2).

$$\begin{aligned} U_x &= 0 \\ \Phi_y &= 0 \\ \Phi_z &= 0 \end{aligned} \tag{3.2}$$

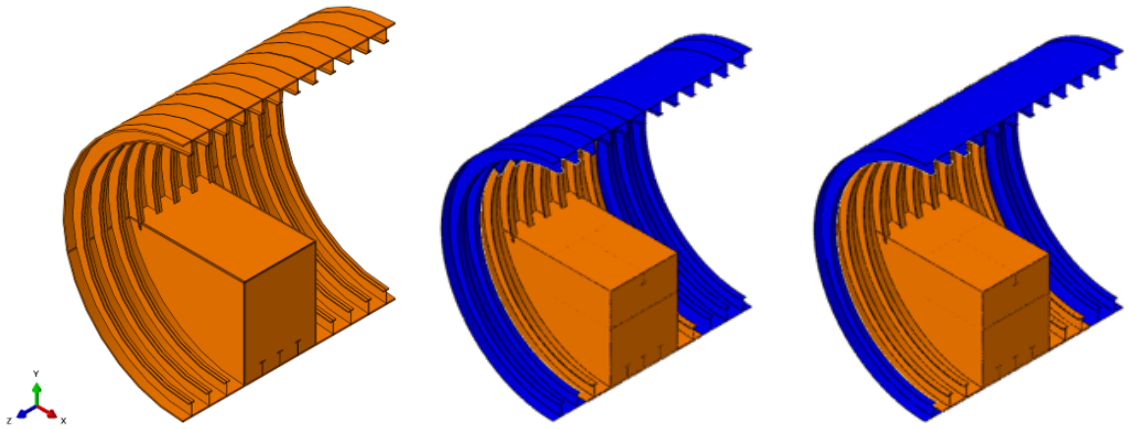
Secondly, periodic boundary conditions to the axial ends of the model coupled with the axial load from Eq. 3.3 is applied as shown in Fig. 3.8(b). Application of a periodic boundary condition avoids stress singularities resulting from the free edges of the section model. The equation constraining displacements in the Z-direction contains a term with the reference point. This enables the application of the axial load corresponding to the sea gauge pressure.

$$
\begin{aligned}
U_x^{Z^+} &= U_x^{Z^-} \\
U_y^{Z^+} &= U_y^{Z^-} \\
U_z^{Z^+} &= U_z^{Z^-} + U_z^{RP} \\
\Phi_x^{Z^+} &= \Phi_x^{Z^-} \\
\Phi_y^{Z^+} &= \Phi_y^{Z^-} \\
\Phi_z^{Z^+} &= \Phi_z^{Z^-}
\end{aligned}
\tag{3.3}
$$

An additional boundary condition is applied to verify the correct application of loads (Fig. 3.8 (c)). A non-equilibrium of loads would result in an unusual corner behavior when one node at the hull corner is constrained in the Y and Z direction (Equation 3.4). This corner boundary condition also removes any rigid body motion and has the side effect of revealing any improper load specification.

$$
\begin{aligned}
U_y &= 0 \\
U_z &= 0
\end{aligned}
\tag{3.4}
$$

The tank structure is a complex structure with multiple connections and components, which when modeled with solid elements, can quickly become computationally expensive. Hence, such structures are often modeled as multi-fidelity models with the critical regions near the tank as 3-D elements and shell elements everywhere. Figure 3.9 shows a representative tank problem geometry and the corresponding High-Fidelity and and Multi-Fidelity models.



(a) (b) (c)

Figure 3.9: Representative tank models (shell regions shown in blue color and solid regions shown in orange color): (a) full solid; (b) multi-fidelity with smaller solid region; (c) multi-fidelity with larger solid region.

## 3.4 Automation using Python Scripting

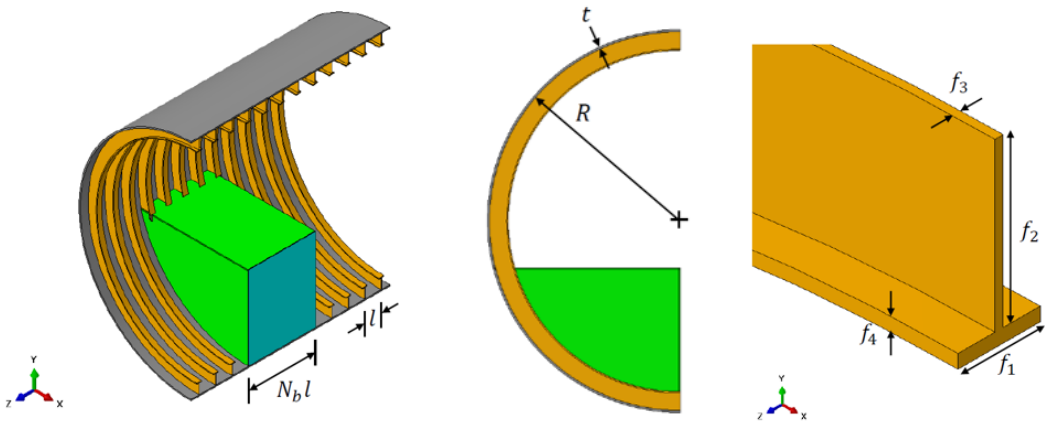
The following sections present the methodology utilized in automating the stiffened tank model development. Major steps as well as insights gained in the development of robust automation scripts from this experience are presented.

### 3.4.1 Geometric Parameterization

The parameterization of the geometry is performed such that there are minimum number of independent parameters needed for the assembly construction. Other dependent parameters are defined in terms of the independent ones. The overbar denotes the non-dimensional quantities which are expressed in terms of the hull radius, $R$, or other independent parameters, and the hat denotes the dependent variables that are derived using other variables.

### Hull and Stiffener Frame Parameters

For scripting purposes, the number of frame bays is fixed to twelve, while the frame spacing and number of tank bays is variable based on user-preference. Fig. 3.10 (a) illustrates the number of tank bays ($N_b$) and the spacing between bays $l$. The radius and thickness of the hull is defined as shown in Fig. 3.10 (b). The frames are defined using the cross-sectional parameters as shown in Fig. 3.10 (c). The parameters are non-dimensionalized as given below:




(a) (b) (c)

Figure 3.10: Essential geometry parameters defining the hull and its frames: (a) hull bays; (b) hull radius and thickness; (c) frame cross-section.

$$\bar{l} = \frac{l}{R},\ \bar{t} = \frac{t}{R},\ \bar{f}_1 = \frac{f_1}{R},\ \bar{f}_2 = \frac{f_2}{R},\ \bar{f}_3 = \frac{f_3}{f_2},\ \bar{f}_4 = \frac{f_4}{f_1}$$

### Chock Parameters

The chock is the small part connected to the two frames at the end of the tank top wall (Fig. 3.6 which transfers load from frame flanges to tank walls. The parameters for the chock are as defined in Fig.3.11 and given explicitly as:

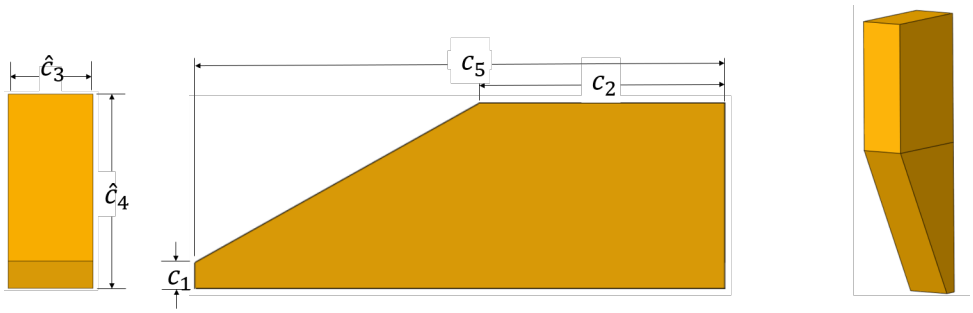



Figure 3.11: Chock parameters.

$$\bar{c}_1 = \frac{c_1}{\hat{c}_4},\ \bar{c}_2 = \frac{c_2}{c_5},\ \hat{c}_3 = \sqrt{(R-f_2)^2-(R-w_2-t_1)^2} - \sqrt{(R-f_2-f_4)^2-(R-w_2-t_1)^2},\ \bar{c}_5 = \frac{c_5}{\hat{c}_4}$$

### Tank Wall Parameters

The tank side walls, top walls and the middle stiffener in the tank component hold internal pressure inside the tank. The parameters for the tank walls are as defined in Fig. 3.12 and given explicitly as:

$$\hat{m}_2 = \hat{t}_3 = N_b l,\ \hat{m}_3 = w_2 + t_1,\ \hat{w}_1 = t_1,\ \bar{m}_1 = \frac{m_1}{\hat{m}_2},\ \bar{w}_2 = \frac{w_2}{R},\ \bar{t}_1 = \frac{t_1}{\hat{t}_3}$$

### Internal Stiffener Parameters

The placement of the longitudinal stiffener is defined to be as general as possible. The placement variable along the x-direction is defined as $x_l$. Using the geometric information of the assembly (radius of the hull and the intersection of the tank top, and the location $x_l$ of the longitudinal stiffener, other parameters are derived, such as the height of the stiffener ($\hat{s}_1$), and height of the foot of the stiffener ($\hat{s}_6$).

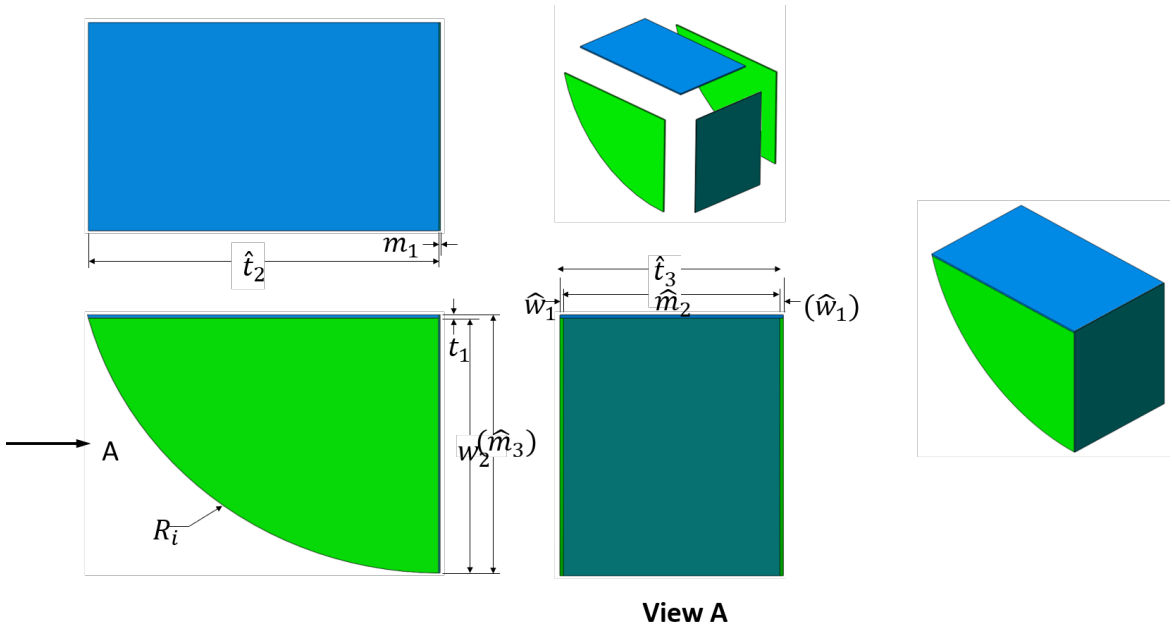



Figure 3.12: Tank wall parameters.

$$\hat{s}_1 = \sqrt{R^2 - {x_l}^2} - R + w_2,\ \hat{s}_2 = lN_b - \hat{w}_1,\ \hat{s}_5 = \hat{s}_2 - 2l + \hat{w}_1 + f_3$$

$$\hat{s}_6 = \sqrt{R^2 - {x_l}^2} - \sqrt{(R - f_2)^2 - {x_l}^2}$$

$$\bar{s}_3 = \frac{s_3}{\hat{s}_1},\ \bar{s}_4 = \frac{s_4}{\hat{s}_1},\ \bar{s}_7 = \frac{s_7}{\hat{s}_1},\ \bar{s}_8 = \frac{s_8}{\hat{s}_1},\ \bar{s}_9 = \frac{s_9}{\hat{s}_1},\ \bar{x}_l = \frac{x_l}{R}$$

$$\hat{b}_3 = s_3,\ \hat{b}_4 = \hat{s}_9,\ \hat{b}_1 = \frac{b_1}{\hat{b}_3},\ \bar{b}_2 = \frac{b_2}{\hat{b}_3}$$

$$\bar{d}_1 = x_l - m_1,\ \hat{d}_2 = \hat{s}_2,\ \hat{d}_3 = s_3,\ \hat{d}_4 = s_4,\ \hat{d}_5 = \hat{s}_2 - 2s_3,\ \hat{d}_7 = s_7,\ \hat{d}_8 = s_8,\ \hat{d}_9 = \hat{s}_9$$

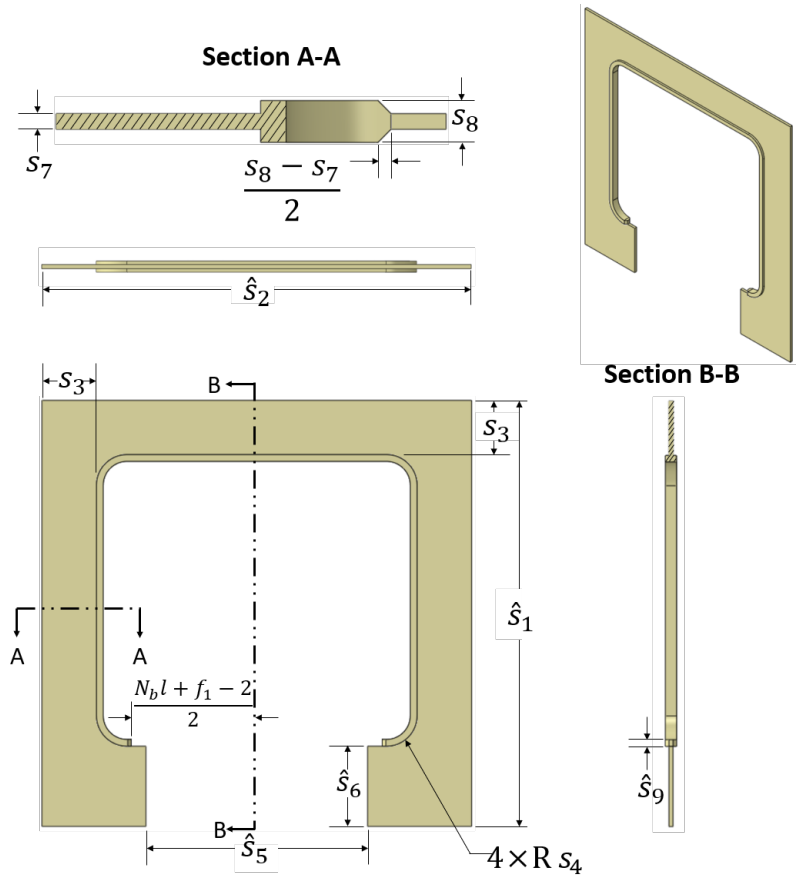



Figure 3.13: Essential geometry parameters defining the internal longitudinal stiffener in the tank.

### 3.4.2 Geometry Creation

Python scripting for Abaqus/CAE is used to construct parameterized models automatically. In order for the scripts to be robust, the sequence of geometry creation and model development needs to be done in a very structured manner. For instance, all the parts for the assembly are created first. Next, partitioning of the part is performed, if needed. Following that, the sets and surfaces are defined, where loads and boundary conditions are to be applied. Finally, materials and sections for the parts are assigned. The reason for a standard sequence for geometry generation is that as the geometric entities are being created, Abaqus/CAE and Abaqus/Standard assigns IDs to them in real time and

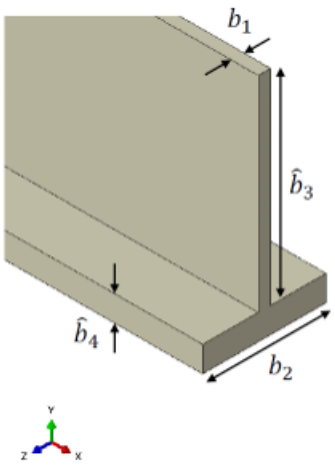



Figure 3.14: Essential geometry parameters defining the internal top lateral stiffener's cross-section inside the tank.

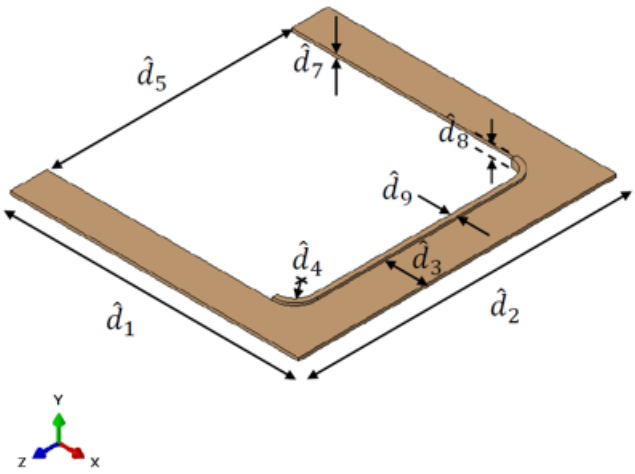
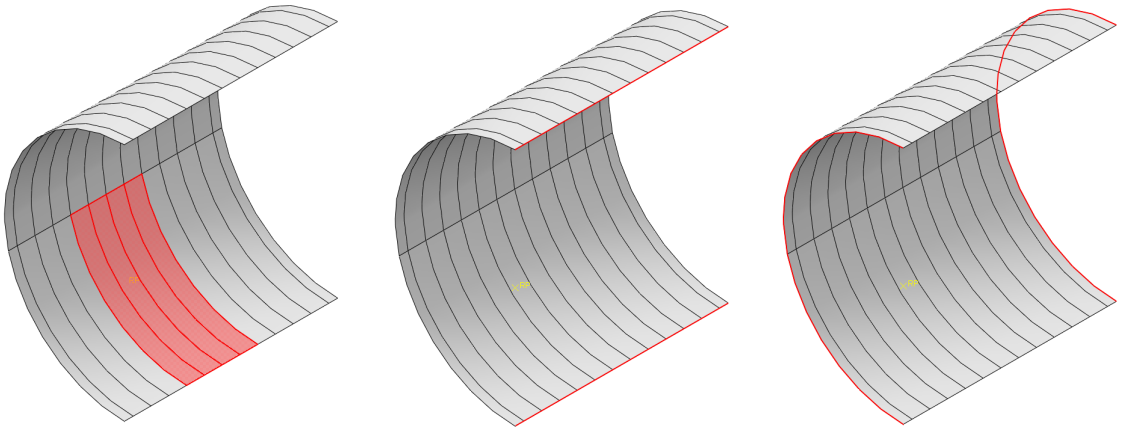



Figure 3.15: Essential geometry parameters defining the internal side lateral stiffener inside the tank.

these IDs can be used to identify the geometric entities later.

### Parts and Sets Creation



(a) (b) (c)

Figure 3.16: Examples of sets defined at the part level to be used later for assemble generation or application of loads and boundary conditions: (a) internal tank pressure application; (b) x-symmetry region; (c) periodic boundary condition region.

Separate programming functions are created for each base parts. These scripts include generalized location of specific points based on the defined parameters such that they could be joined into lines to create a part. Once the part is generated, different sets (a collection of faces and edges) in the part are identified. These sets, once defined, can be recalled in the script multiple times. For instance, the end edge faces of the hull are saved into a set for the application of periodic boundary conditions (Fig. 3.16(c)). Similarly, faces on the hull where is the tank is located is defined into a set for easier selection of faces for the internal tank pressure application. Since the goal is to make the automation script as general as possible, all different cases and scenarios are considered before defining part level sets and partitions. For instance, in order for the tank structure to be able to adapt from two bays to even six bays, the hull structure is partitioned into twelve uniform bays. Partitions are created on the hull at the part level using parameterized datum planes

(shown in dotted yellow) in Fig. 3.16(a).

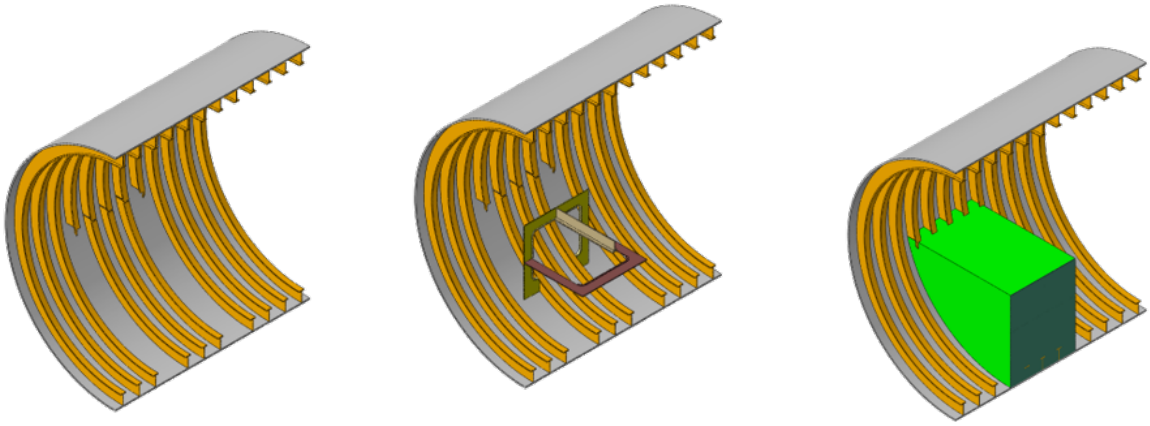
Each bay is defined into a separate set and a IF-ELSE loop is developed to vary the selection of sets as per the change in number of bays over which the tank exists. Partitions are made for the easier placement of frames that run longitudinally along the hull. Additionally, partitions are made in the Z-direction where the tank top wall is supposed to be present. All of these partitions are performed using parameterized datum planes.

Next, sets are defined for the regions where the X-symmetry boundary conditions are applied on the hull (Fig. 3.16(b)). Similarly, other parts including frames, tank walls, chock and internal stiffeners are generated and sets are defined as appropriate. Note that if different sets from different parts have the same set names, these sets are merged in the assembly as a one whole set when geometry merging is performed. This feature is utilized to name all the sets in different parts. For example hull, frames and tank middle wall have to be defined with X-symmetry condition. These different regions are named the same, 'X-symm' in the different parts and are merged in the assembly to obtain a single set called 'X-symm'. Moreover, in order to place different parts correctly into the single assembly, it is important to set certain part edges and points coincident, parallel or coaxial to each other. These specific edges and points are determined at the corresponding part level to be defined into complete assembly. Once the sets are defined, these can be used to make assembly placements for different parts.

### Assembly Creation

Once all the parts and sets have been created, the assembly can be generated by placing the parts appropriately. Assembly tools and constraints available in the FE package like translate, pattern, coincident point, parallel edges, etc. are employed to place all the

different parts at expected locations (Fig. 3.17).



(a) (b) (c)

Figure 3.17: Assembly building: (a) hull with stiffeners; (b) internal tank stiffeners; and (c) tank walls and middle stiffeners

### 3.4.3 Corresponding Shell Model Creation

A corresponding full shell model is developed as shown in Fig. 3.18(a). For this model, it is important to carefully make corrections to the parameters used in the full solid model such that the full shell surfaces refer to the mid-surface in the solid model, unless shell offset modeling is utilized. Geometry parameters like the hull radius, tank walls, tank top and internal tank stiffeners had geometric corrections applied to place the shell in the middle of the corresponding solid wall. Additionally, heights of the frame webs, mid-lateral tank stiffeners and tank-top lateral stiffener web are adjusted appropriately.

However, because the model has a curved topology (hull) intersecting with a horizontal one (tank top), certain entities cannot be adjusted to ensure shell matching at the solid mid-surface. In these situations, these differences are retained in the shell model. For

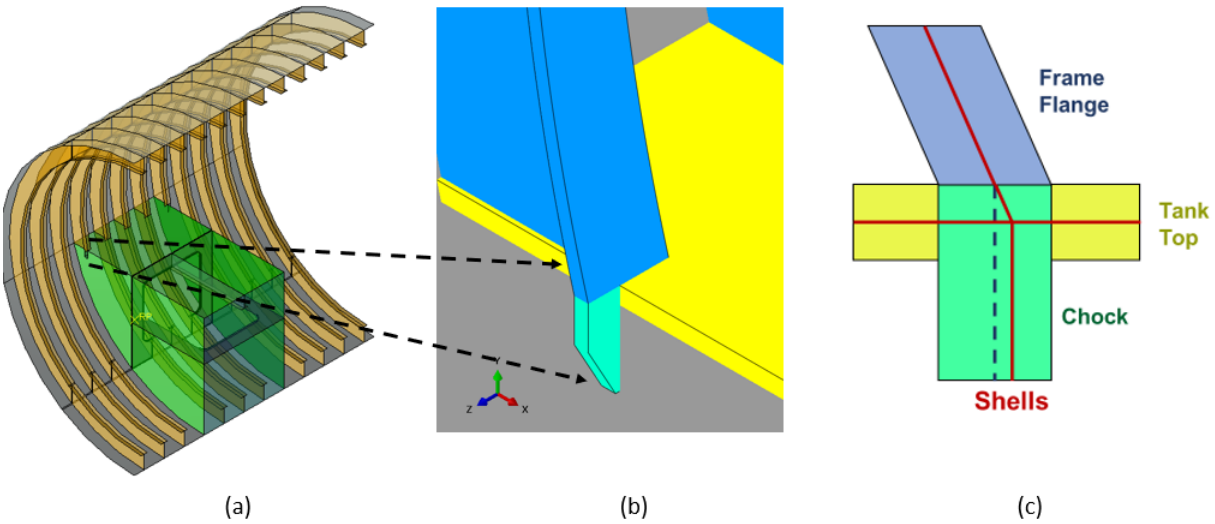



Figure 3.18: Full Shell Model; (a) global view; (b) zoomed in view of the chock and tank top interface in the full solid model; (c) corresponding shell-mid-surfaces in the shell model shown as red lines).

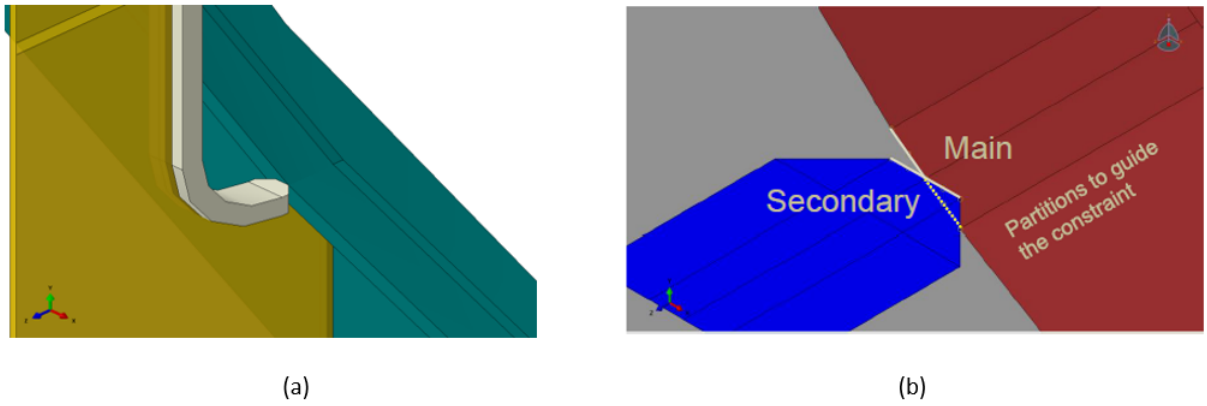
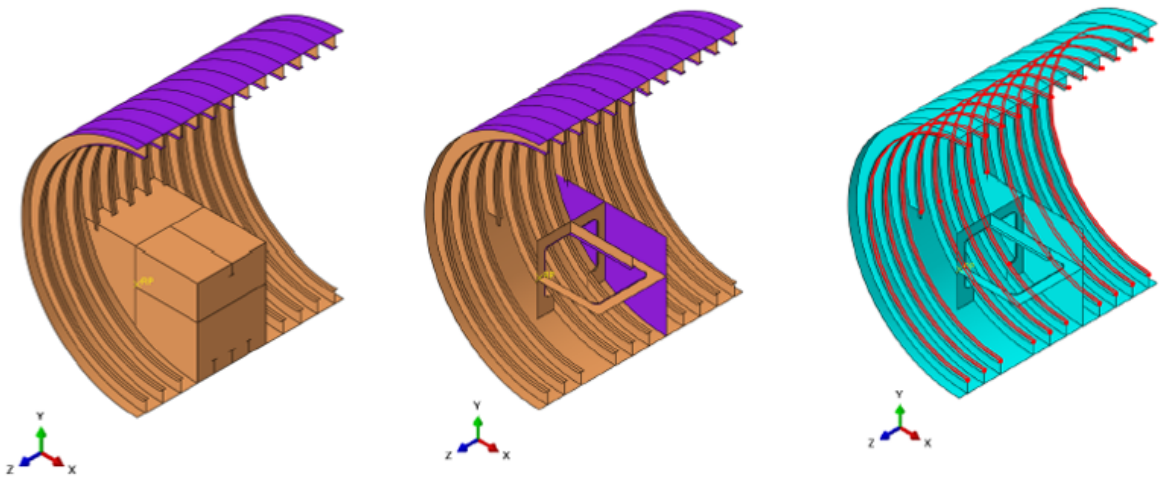



Figure 3.19: Longitudinal stiffener to frame connection; (a) solid model, (b) shell model: flat chamfer connected to frames using tie constraint).

example, Fig. 3.18(b) and Fig. 3.18(c) show a close-up view of the chock to tank top connection in the full solid model and a corresponding shell mid-surface in the all shell model. Another example includes the connection at the longitudinal stiffener foot and the frame as shown in Fig. 3.19. In this case, a flat chamfer with tied contact and adequate

node selection tolerance is used in the full shell model. It is to be noted that although there exists a mid-surface extraction capability in Abaqus/CAE to generate shell models from 3-D geometry, it is not robust for complicated assemblies and hence, a separate script is developed to generate a full shell model.



(a) (b) (c)

Figure 3.20: Full Shell model: (a) shell normals shown in brown and purple, (b) shell normals inside the tank, (c) sets defined for defining shell normal directions.

Moreover, the shell normal direction in the full shell model needs to be verified such that it is consistent with the pressure load application in the corresponding model. In order to automate the surface selections and normal direction definitions, geometry sets are defined such that these can be used in the scripts multiple times. An example of such sets used for consistent shell normals is shown in Fig. 3.20.

### 3.4.4 Corresponding Multi-fidelity Shell-Solid Model

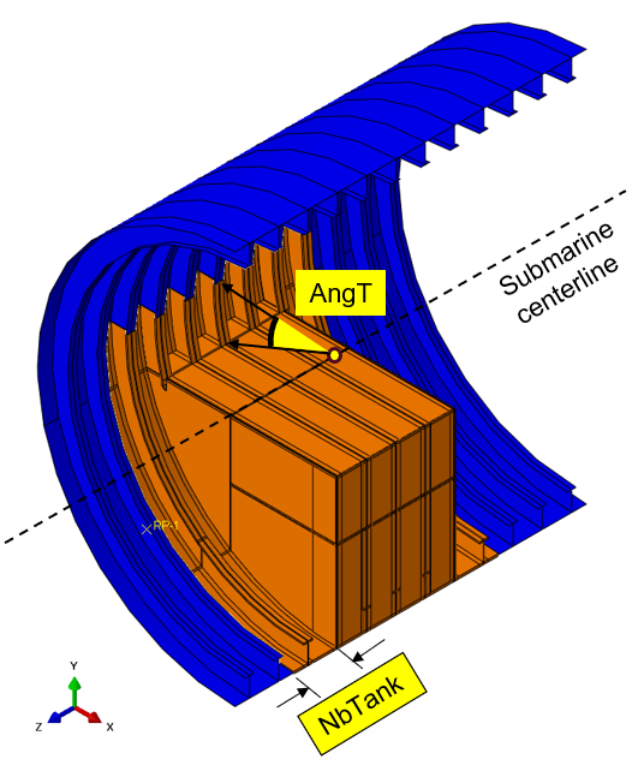



Figure 3.21: Multi-fidelity parameters: Angular approximate solid region above the tank (AngT) and solid side bays (NbTank).

After investigating different mixed-fidelity topologies, a practical model considering solid region near the entire tank (with internal stiffeners) and the remaining hull structure as shell is selected as shown shown in Fig. 3.9. The parameters of the multi-fidelity topology are defined so that the multi-fidelity topology, *i.e.*, the level of fidelity of the model can be changed in an automated and parameterized way. Figures 3.21 demonstrate the two multi-fidelity parameters, $AngT$ and $NbTank$ which correspond to the angular solid region above the tank and solid region in number of bays next to the tank respectively. A side-view of the definition of $AngT$ that subtends above the tank top measured from the submarine center-line is shown in Fig. 3.22. Note that the parameterization was performed

such that the value of the angle subtended above the tank top from the submarine center-line achieves the value of $AngT$ approximately.

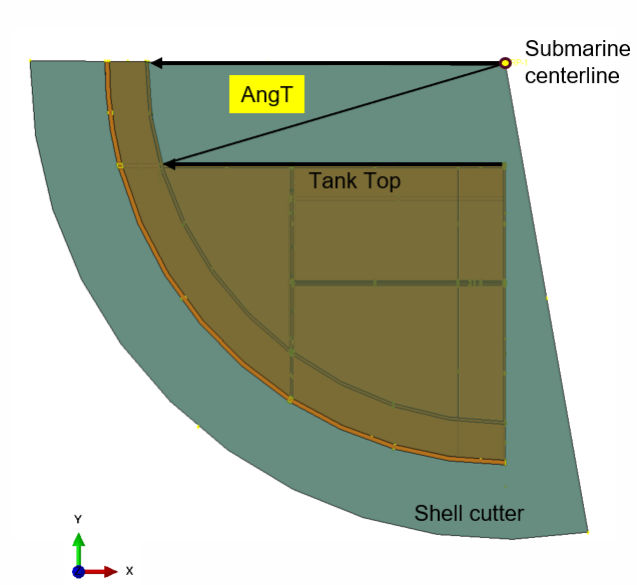



Figure 3.22: Multi-fidelity parameter (side view): Angular approximate solid region ($AngT$) subtended above the tank from the submarine center-line.

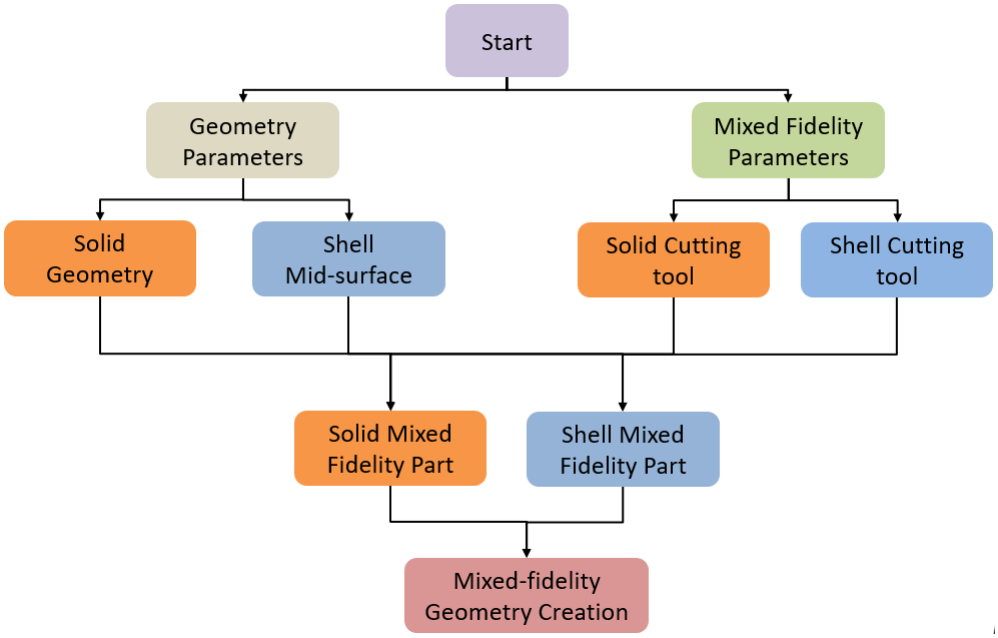



Figure 3.23: Flowchart depicting the methodology used for automatic generation of parametric mixed-fidelity FE model.

A multi-fidelity model leads to interface regions where the solid element's surface needs to be numerically compatible with the shell edges. In this work, Abaqus/Standard was utilized that employs coupling constraint equations to ensure that the bending rotations and moments are correctly transferred between the two domains in a self-equilibrating manner. Scripting a multi-fidelity model with variable shell-solid topology can get very challenging and hard to generalize, especially when further geometric parameters are introduced (*e.g.*, parameters for the size of the different mixed fidelity shell and solid regions). A novel approach is proposed that uses Abaqus/CAE's cutting tools to combine and merge the already developed solid and shell models (or separately scripted shell regions). This is an efficient method since it uses part names and no geometric IDs which are typically encountered while scripting geometry creation for the solid and shell models, thereby reducing considerable effort. The creation of these cutters is automated using scripts and is found to be an effective method in the lieu of developing a separate script for the creation of a mixed model similar to the full solid and full shell models. Figure 3.23 presents a flowchart depicting the cutter methodology utilized to generate adaptable fidelity models. These cutters are parameterized such that the 3-D topology can be changed based on user-preference. The cutting tool methodology is shown in Fig. 3.24 and Fig. 3.25. These cutting tools act as negative spaces to remove geometry from overlapped regions when placed on top of the baseline structural model. These cutting tools are parameterized based on the desired mixed fidelity parameters (Fig. 3.21).

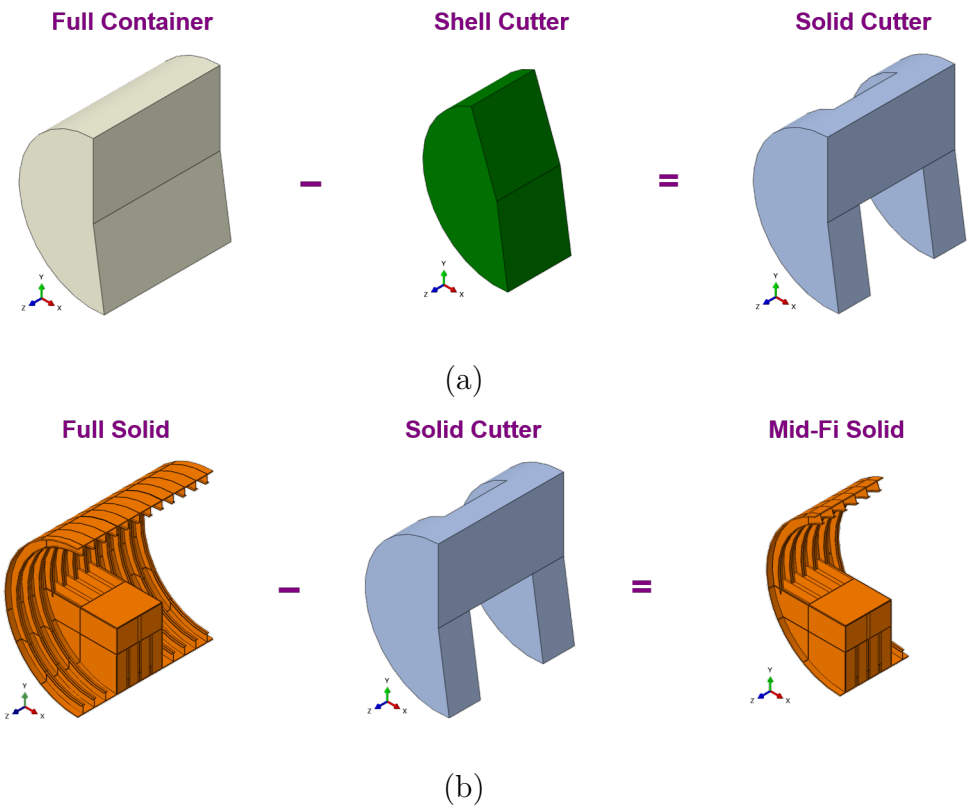
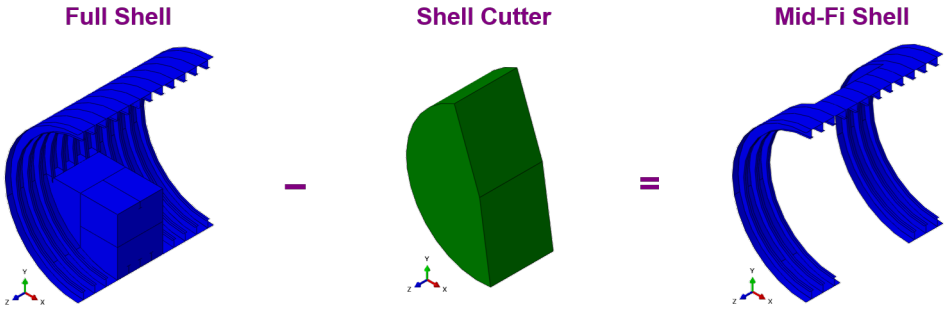


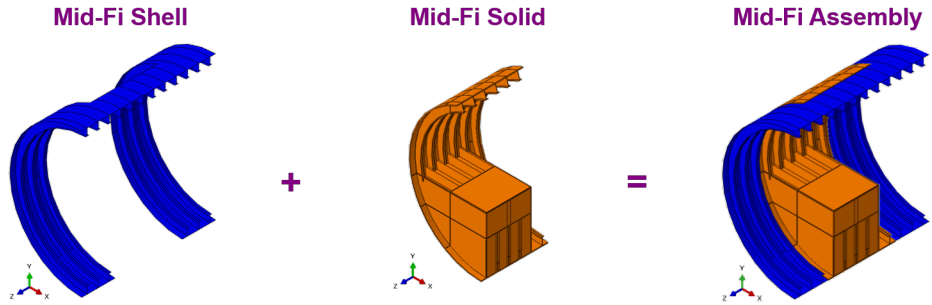

Figure 3.24: Multi-fidelity model creation using complementary cutting tools: (a) creation of a solid cutter; (b) creation of the solid region in the multi-fidelity model using the solid cutter.

Once the cut solid and shell parts are obtained, these need to be constrained at the interface for accurate transfer of forces and moments in the model. It is found that Abaqus/Standard automatically detects the shell-solid interfaces when a geometry merge is performed on these parts. However, this implicit specification leads to a mesh that must be congruent at the interface. Moreover, the shell edge creates a partition on the solid surface which leads to additional constraints in the mesh development of the model. An example of such a congruent mesh at the interface is shown in Fig. 3.26(a). On the contrary, it is possible to generate an incongruent mesh at the shell-solid interface when the separate solid and shell parts are assembled without merging the entire assembly. Such an explicit

shell-solid coupling at the assembly level does not partition the solid part and enables a computationally efficient mesh generation (Fig. 3.26(b)). In order to apply explicit shell-solid coupling in an automated way, the shell surface edges and solid surfaces are selected in a parameterized way. This is performed by assigning a parameterized reference point for each solid surface and shell edge. This point (and consequently the surface) is then selected in the model generation scripting using geometric tolerancing based entity selection. Commands like findAt() are used for selecting the points available in Abaqus/CAE software. The reference points and the surfaces in the shell-solid coupling for the tank problem are shown in Fig. 3.27. The final assembly is developed using explicit shell-solid coupling (Fig. 3.28) at the interface.

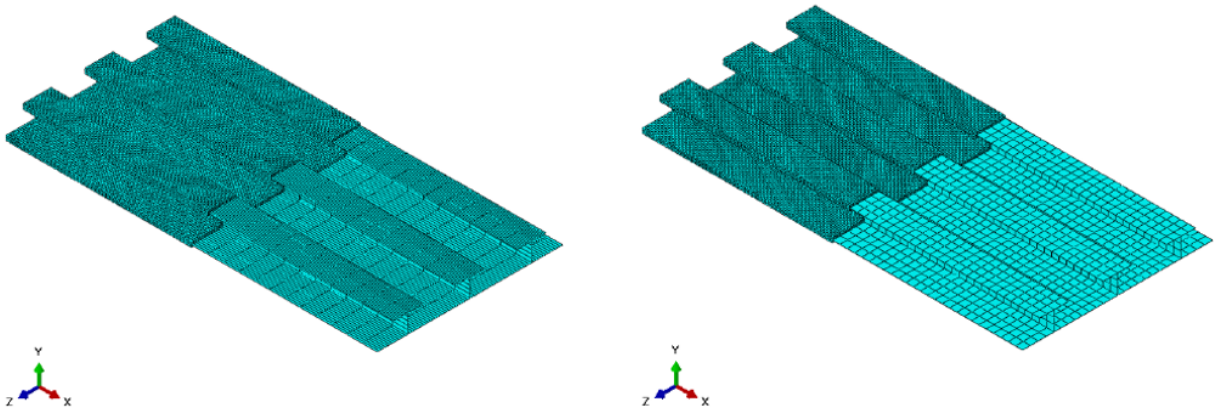



(a)




(b)

Figure 3.25: Multi-fidelity model creation using complementary cutting tools: (a)creation of shell region in the multi-fidelity model using shell a cutter; (b) assembling the solid and shell regions to create the multi-fidelity model assembly.



(a) (b)

Figure 3.26: Solid/shell coupling mesh development: (a) implicit coupling from geometry merging resulting in a congruent mesh at the solid/shell boundary; (b) explicitly specified coupling allowing non-congruent meshing at the boundary.

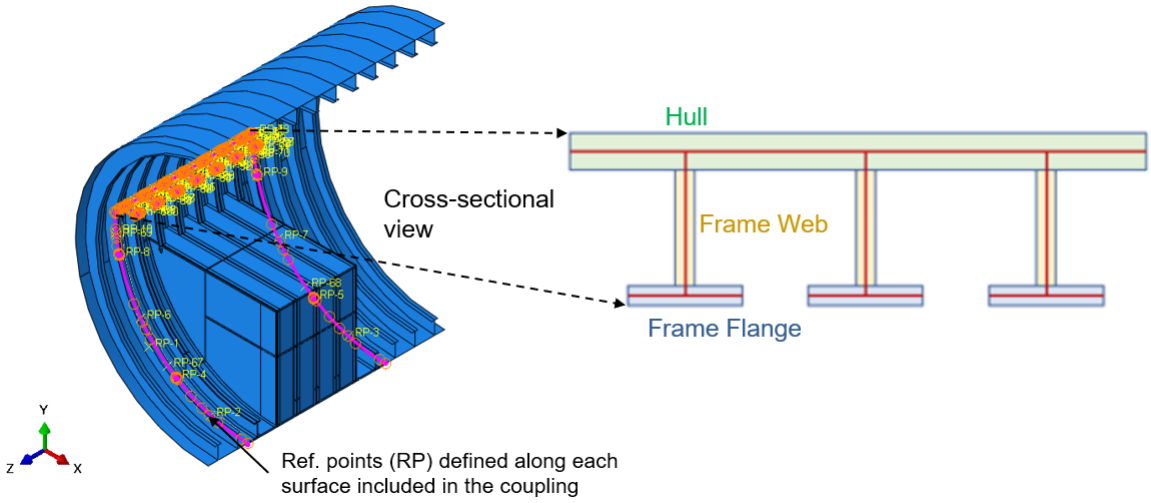
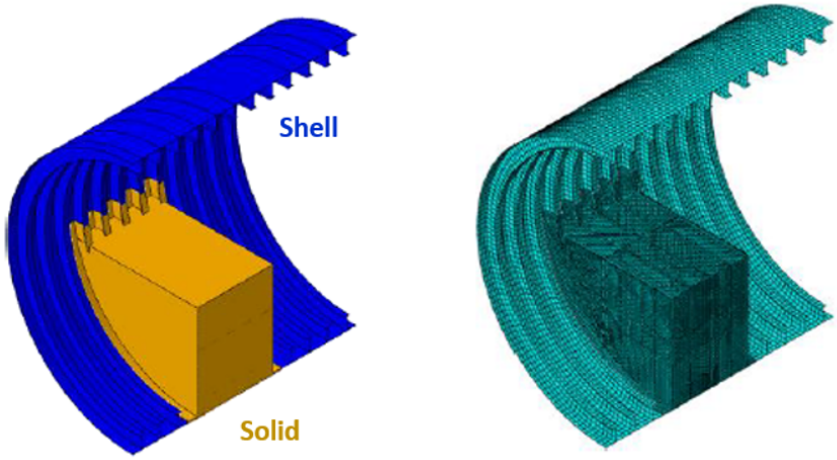



Figure 3.27: Explicit Shell-Solid constraints defined for the tank model.



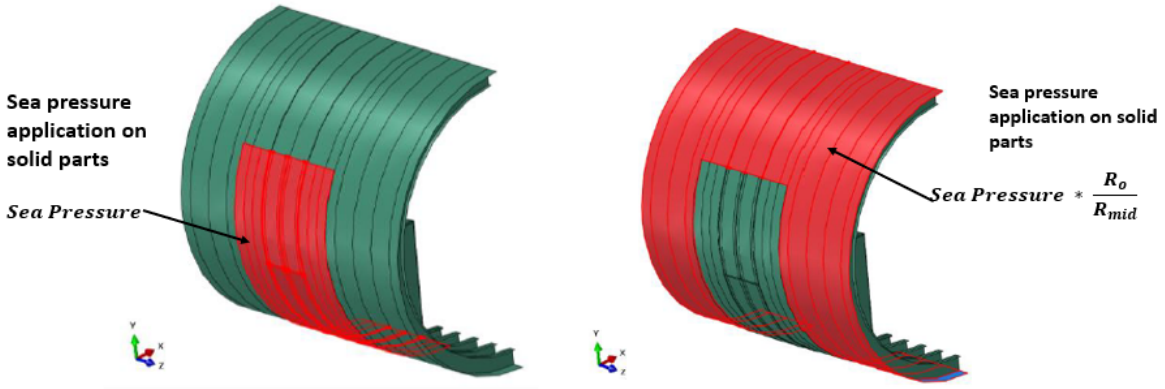

(a) (b)

Figure 3.28: Multi-fidelity tank model: (a) element topology; (b) computationally efficient meshing scheme.

### 3.4.5 Load and Boundary Condition Application

#### Application of Loads

The load and boundary condition application is performed using the previously defined part-level geometric sets which are propagated to the assembly-level. It is also worth mentioning that set names which are exactly the same, used in different parts get merged into a single set at the assembly level, when geometry merging is used. This property of the geometry sets makes it possible to define part level geometry sets which can then be selected as one whole set in the assembly for boundary condition or load application. It was found that many times using scripting commands based on geometric tolerancing available in Abaqus/CAE to make edge and surface selection can become very time consuming for a complicated geometry. In such cases, it can be quite easy to write IF-ELSE blocks to




(a) (b)

Figure 3.29: Sea Pressure Application: (a) solid surfaces (tank and surrounding tank regions) in the mixed model; (b) shell surfaces (away from the tank) in the mixed model.

consider different cases of the expected variation and store the IDs generated by Abaqus/CAE in the correct sequence. For instance, the number of surfaces required to be selected for the internal tank pressure change only for certain cases where the number of tank bays changes. Thus, an IF-ELSE block considering the surface selection for these cases proved to be useful with the utilization of Abaqus/CAE generated IDs. The load application for the solid model is straightforward as shown in Fig. 3.7. When mixed models are used, it is important to apply a correction factor to the sea pressure application. The outer solid surface can use sea pressure value directly. However, the shell surface at mid-plane needs the applied pressure to be adjusted slightly by a factor of ratio of the hull's outer radius to its mid-surface (Fig. 3.29) to account for its surface area difference.

### Application of Boundary Conditions

Application of the boundary conditions is performed using the pre-defined sets at the part level. Special consideration is needed for the application of the periodic boundary

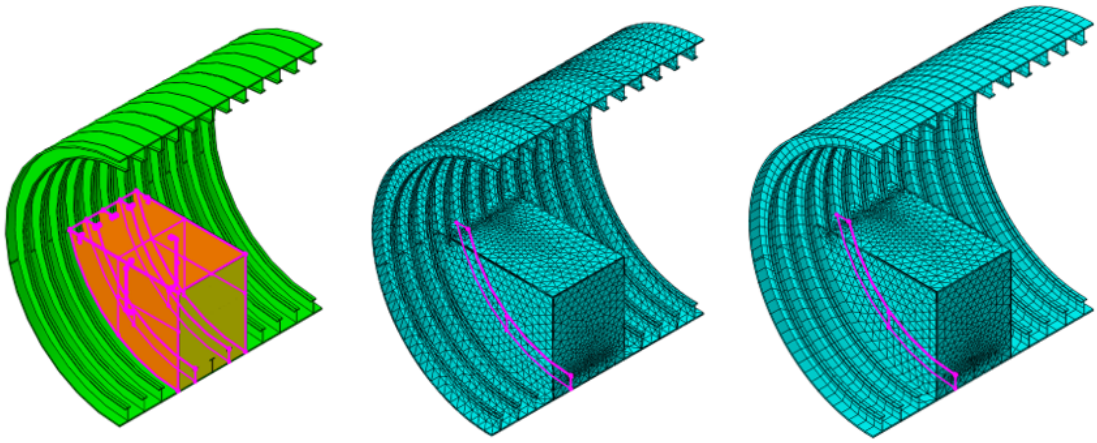
conditions in an automated way. The application of periodic boundary condition needs to be done once the assembly has been meshed. The matching sides in the periodic boundary condition have to be seeded consistently. A separate Python function is written to perform the periodic boundary application on the models depending on the fidelity levels. This subroutine first ensures that the number of nodes on the Z+ and Z- face/edge of the models and their corresponding position is consistent. Following the confirmation, the co-ordinates ($x$, $y$, $z$) of these nodes are saved in different arrays. Once saved, these arrays are sorted using a single coordinate at a time and then saved in a single text file. This procedure is repeated for the second side. The equations are created in the input files using the $x$, $y$ and $z$ co-ordinates along with the node number of the reference point (if applicable). Once sorted, the corresponding nodes on the Z+ side and Z- side are then constrained in *Equation card in the input file of the FE model.

### 3.4.6 Automated Mesh Generation

#### Challenges Associated with Hex-Meshing Complex Assemblies

Commercial FE packages like Abaqus/CAE allow for the creation of a complex assembly by geometric merging and cutting of multiple parts. It is observed that the assembly of multiple parts necessitates conformal meshing on faces of the neighboring components. This amplifies the complexity involved in meshing complicated geometries with hexahedral meshes, especially solid regions. The realistic tank geometry is complicated with several connections of radial and rectangular regions. Fig. 3.30 demonstrates the challenges encountered in meshing the geometry. It can be seen that the problematic regions are occurring at the regions of interactions between radial hull/frame with the rectangular tank walls and stiffeners. Fig. 3.30 (a) shows the regions which fail to be meshed unless the

assembly is partitioned into simpler parts or tetrahedral elements are used. However, the alternative of using tetrahedral mesh is not feasible since some regions still fail, shown in magenta color in Fig. 3.30 (b). Another mitigation step of using a combination of tetrahedral and hexahedral meshes leads to tied contact at the incompatible surfaces. This option, although not preferred, also breaks down for complex geometries (Fig. 3.30 (c)).



(a) (b) (c)

Figure 3.30: Meshing the tank model: (a) regions shown in magenta cannot be meshed automatically unless partitioned or assigned Tet elements; (b) regions shown in magenta failed to mesh even with tetrahedral element type; (c) alternative of a combination of structured (hull region) and tetrahedral mesh (tank region) also fails to generate a mesh automatically.

More specifically, these component interactions create partitions on the faces at the connection location. These contact trace edges, although seemingly trivial, need careful consideration during meshing. An example of such a face partition in the tank model due to the connection of the internal longitudinal stiffener and the internal frames, is shown in Fig. 3.31. The tank model contains several connections similar to this and have to be addressed strategically to obtain a conformal mesh. Volume decomposition is an indispensable technique that has been used to divide complicated geometries into simpler regions such that the available meshing algorithms are able to produce a mesh

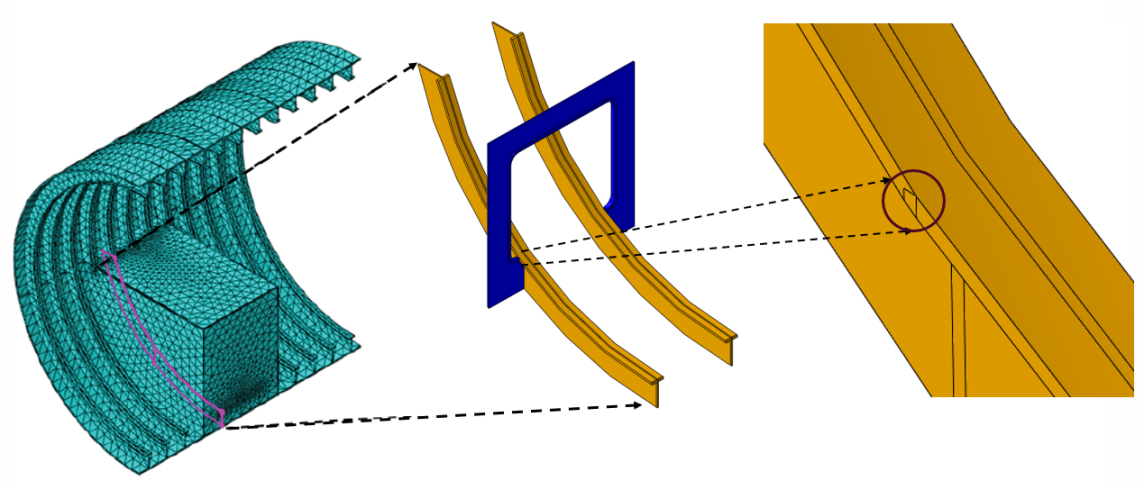


Figure 3.31: Retained boundaries on the assembled components leading to meshing complexities.

automatically. This is also effective in avoiding less accurate alternate methods like tie constraints and tetrahedral elements. Moreover, a well partitioned volume ensures high quality hexahedral meshes. Hence, it is convenient to partition the geometry such that a structured scheme can be utilized. The following section presents a heuristics-based blueprint that has proved to be effective in decomposing the tank model to ensure hex-dominant meshes.

### Framework for Effective Partitions

The problematic region in the tank assembly at the foot of the longitudinal stiffener in connection with the frames is resolved using this face propagation technique. The face partitions induced due to the connection (Fig. 3.32) are propagated around the longitudinal stiffener such that the elements are conformal throughout the assembly (Fig. 3.33). Based on the interactions between the frames with tank walls and internal stiffeners connected to the side tank walls, partitions were made along the connection edges as shown in Fig. 3.34. Consequently, most parts of the model are flagged as structured mesh by Abaqus/CAE.

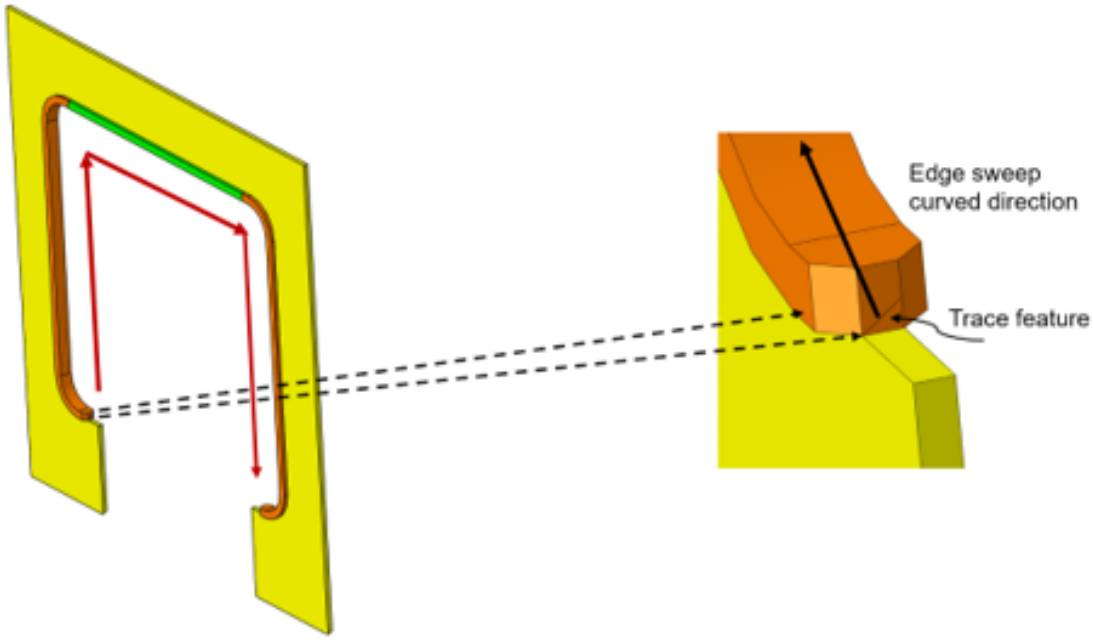
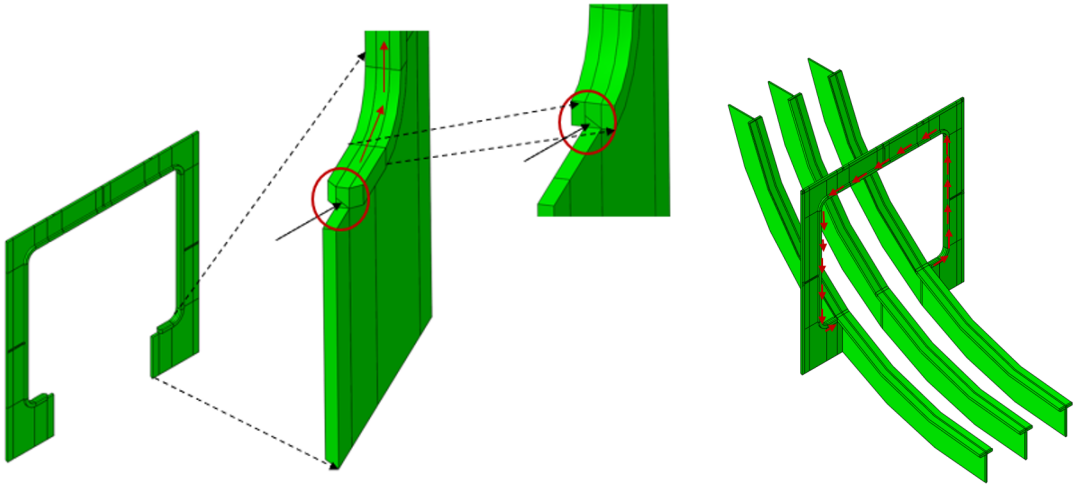
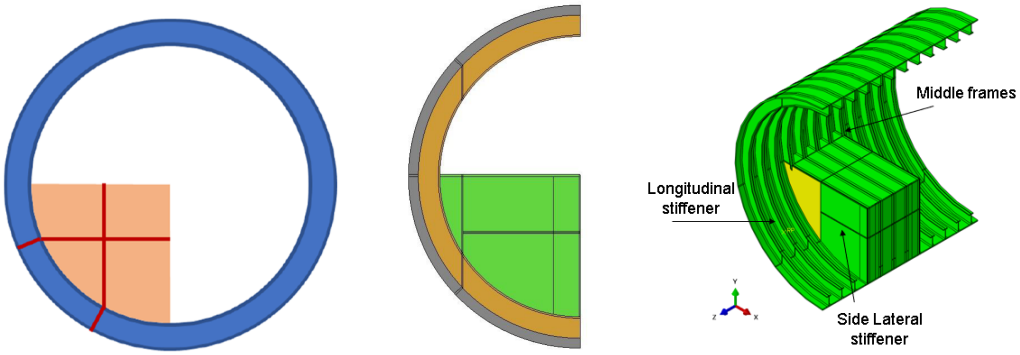



Figure 3.32: Retained boundary at the longitudinal stiffener flange due to component assemblies that need to be propagated throughout the assembly.



(a) (b)

Figure 3.33: Face propagation technique applied to the tank model: (a) zoomed-in view of the face partition present at the foot of the longitudinal stiffener connected to the frames; (b) face partitions propagated along the longitudinal stiffener flange until a closed loop is achieved.




(a) (b) (c)

Figure 3.34: Partitions conforming to the geometry lead to better quality meshes: (a) radial cut on hull avoids skewed meshes; (b) conformal partitions utilized on the tank assembly (side view); (c) conformal partitions utilized on the tank assembly (isometric view).

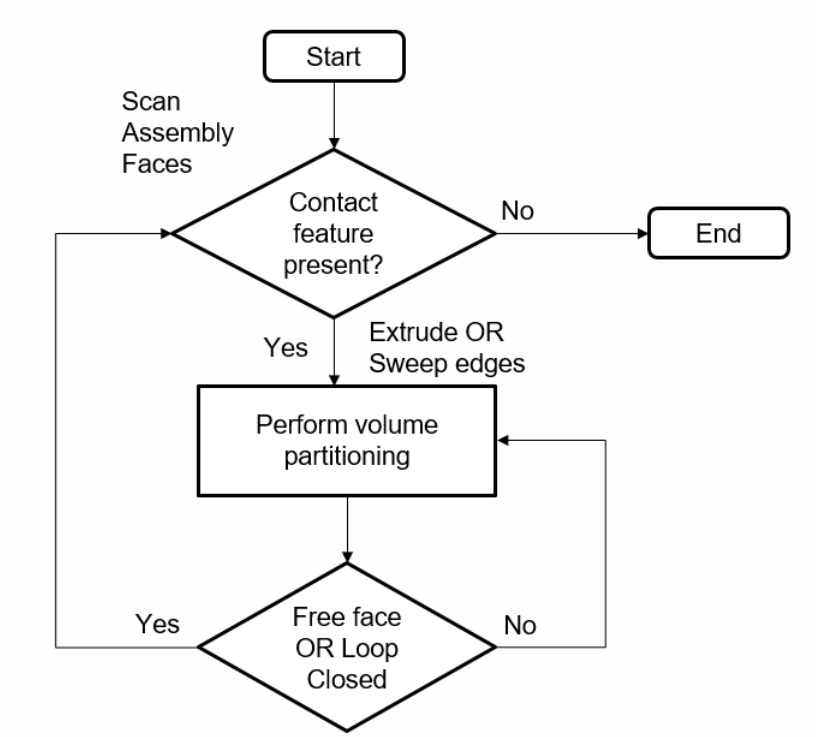



Figure 3.35: Foundational framework for a volume decomposition algorithm for generating hex-mesh dominant regions.

Using the insights gained from effectively parameterizing the tank model, a face partition propagation based volume decomposition technique is proposed. This algorithm is

presented at a foundational level with the aim to provide insights for future research in this area. The flowchart is presented in Fig. 3.35. The decomposition technique aims at propagating a face partition, caused by connections in the assembly, until a free surface is reached or until a curvature loop is closed in an assembly. This way, the trace feature is propagated all the way throughout the assembly, and it becomes easier for the existing algorithms to create a hexahedral mesh. In particular, making partitions in the assembly guided by the the face partitions (created by the connections) is sought. This approach to volume decomposition directly aligns with the idea that the hexahedral elements are like an interlaced stack such that each stack must begin and end at a boundary or be a closed loop of elements, as mentioned by Blacker *et al.* [32].

As a starting point for the algorithm shown in Fig. 3.35, all the faces in the assembly can be identified. Next, face partitions can be recognized by identifying the common edge between two adjacent faces using the existing algorithms like the CLoops identification. Once this face partition has been identified, the next step would be to determine the cell associated with this face partition. This cell needs the face partition to be propagated throughout its volume until the opposite face of the cell has been reached. This can be done by sweeping/extrude the face partition along with direction in which the opposite face exists. The extrude tool is useful when the two opposite faces of a cell are parallel. If the cell has a curved shape, then the sweep tool is appropriate to propagate the face partition. Once this partition is propagated on the other side, a check could be performed to determine if a free end has been reached or the face partition already exists (case where it is a closed loop). If not, the procedure can be repeated until a free end or a close loop is achieved as shown in Fig. 3.33.

### Creating Parameterized Partitions and Automated Meshing

Once a scheme leading to the successful generation of hex-dominant mesh is achieved, partitions are parameterized using the geometry parameters of the tank model and scripted using Abaqus/Python interface. Specifically, partitions are performed using datum planes or face extensions. The creation of a datum plane or face extension is performed by picking vertices based on parameterized coordinates, ID tracking or by estimated geometric information. Once the partitioning tool is defined, the faces and cells to be partitioned are also picked using estimated geometric information or geometric ID tracking using commands like findAt(), getByBoundingBox(), getClosest() in Abaqus/CAE. It is worth mentioning that this process of utilizing geometric parameters for entity selection is tedious but enables robust scripts for large design exploration. Partitions are created using the several volume decomposition tools available in Abaqus/CAE. Most of the partitions are performed at the part level in Abaqus/CAE before creating the assembly. This makes it easier to locate the geometric identities required to perform partitions at a local level.

The next step is to determine if certain regions need specific meshing seeds in the model. The defined geometric sets are utilized to perform mesh seeding. For instance, the thickness edges for the hull, tank walls and frame flanges are constrained to have at least three elements through the thickness. This is done to capture the accurate bending response of these walls due to the applied external and internal pressure loads. The solid structures are analyzed using 3-D continuum elements (Abaqus/Standard C3D8R element with reduced integration and hourglass control). In order to select the correct mesh size for comparison, a mesh convergence study was performed for the solid and shell model based on the maximum displacement and the maximum principal stress at the location of the maximum displacement. For the all solid modeling, three elements through the thickness of the hull and through the thickness of the frame flanges near the tank top, are utilized. The

full shell model was meshed using S4R shell elements which employ reduced integration with hourglass control. The mixed model utilized a combination of C3D8R and S4R elements with shell-solid coupling at the interface. The developed automation scripts can develop parameterized full solid and mixed models. Figure 3.36 shows some instances of the solid models generated using the automation script.

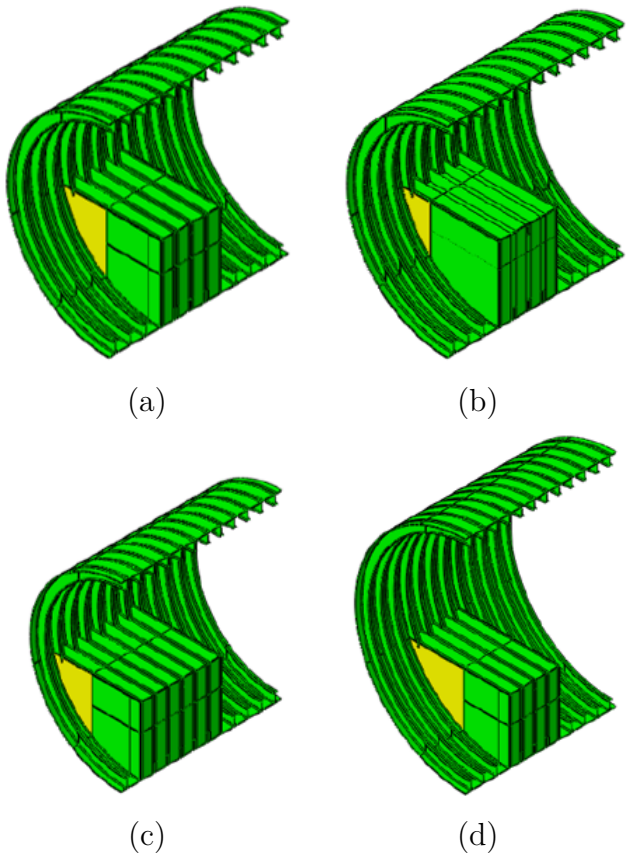



Figure 3.36: The automation script is capable to changing all the different parameters of the tank model (green indicates regions selected automatically for structured meshing, while yellow indicates regions selected automatically for swept meshing). Some examples include (a) baseline geometry; (b) variable longitudinal stiffener location; (c) variable tank length (d) variable hull radius.

## 3.5 Comparing Modeling Fidelities and Results Post-Processing

A comparison between different modeling fidelities is performed for a reference geometry with the following parameters (defined in section IV. A.): $R = 6.7\,\text{m}$, $l = 1.0\,\text{m}$, $t = 0.0762\,\text{m}$, Nb $= 4$, $w_2 = 4.97\,\text{m}$, $t_1 = 0.0635\,\text{m}$, $m_1 = 0.03\,\text{m}$, $f_1 = 0.3\,\text{m}$, $f_3 = 0.03\,\text{m}$, $f_2 = 0.61\,\text{m}$, $f_4 = 0.05\,\text{m}$, $c_1 = 0.01\,\text{m}$, $c_2 = 0.16\,\text{m}$, $c_5 = 0.4\,\text{m}$, $s_3 = 0.5\,\text{m}$, $s_4 = 0.28\,\text{m}$, $s_7 = 0.0381\,\text{m}$, $s_8 = 0.1016\,\text{m}$, $x_l = -3.6\,\text{m}$, $b_1 = 0.03\,\text{m}$, $b_2 = 0.3\,\text{m}$. The material property used is as follows: $E = 200\,\text{GPa}$, $\nu = 0.3$, $\rho = 7850\,\text{kg/m}^3$.

Five different fidelity levels are compared starting with (a) a full shell model, (b)-(d) variations in multi-fidelity topology, (e) a full solid model, as shown in Fig. 3.37. The mesh for models corresponding to each of the five fidelity levels is shown in Fig. 3.38. Displacement and maximum principal stress contours are compared in Fig. 3.39 and Fig. 3.40 respectively. It can be observed qualitatively, that the shell model varies significantly in the tank wall stress contours. On the other hand, all the three mixed models perform well in comparison with the full fidelity solid model. Since there is no one-to-one mesh correspondence between the meshes in different fidelity models, an error metric is developed to compare the peak stress values and strain energies in local user-defined domains.

Figure 3.41 shows the identified critical domains for the tank model. The domains are chosen at the crucial regions of the tank and hull intersections such that the possible singularity regions are avoided. Maximum principal stresses in six local domains (A-F) on the tank top and tank side walls are compared for different models. Additionally, the local volumetric strain energy values in critical components, namely, the chocks, tank side walls and tank top walls are also compared. The solid model is considered as the full fidelity

model and all the values are compared relative to the full solid model. The comparison of results from different multi-fidelity models with the full solid model is also automated using python scripting. The local domain regions were parameterized using the geometric information and saved as node sets in the models.

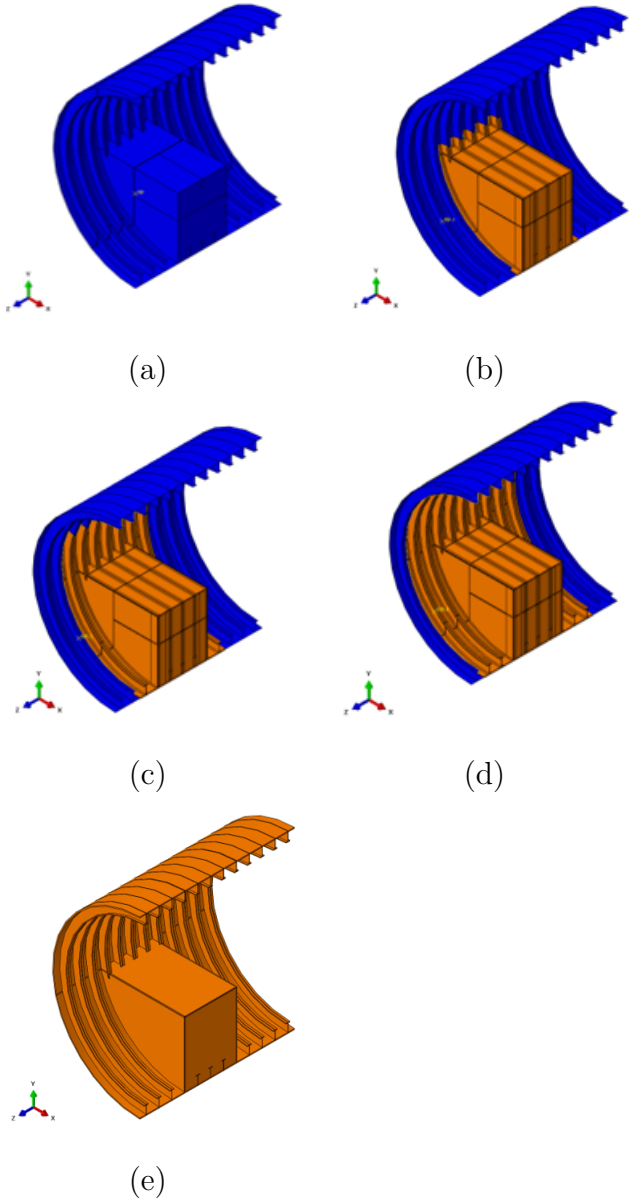



Figure 3.37: Baseline model with different fidelities compared (a) full shell; (b) multi-fidelity configuration 1 (NbTank = 0.5, AngT = $5^0$); (c) multi-fidelity configuration 2 (NbTank = 1.5, AngT = $30^0$); (d) multi-fidelity configuration 3 (NbTank = 2.5, AngT = $60^0$); (e) full solid.

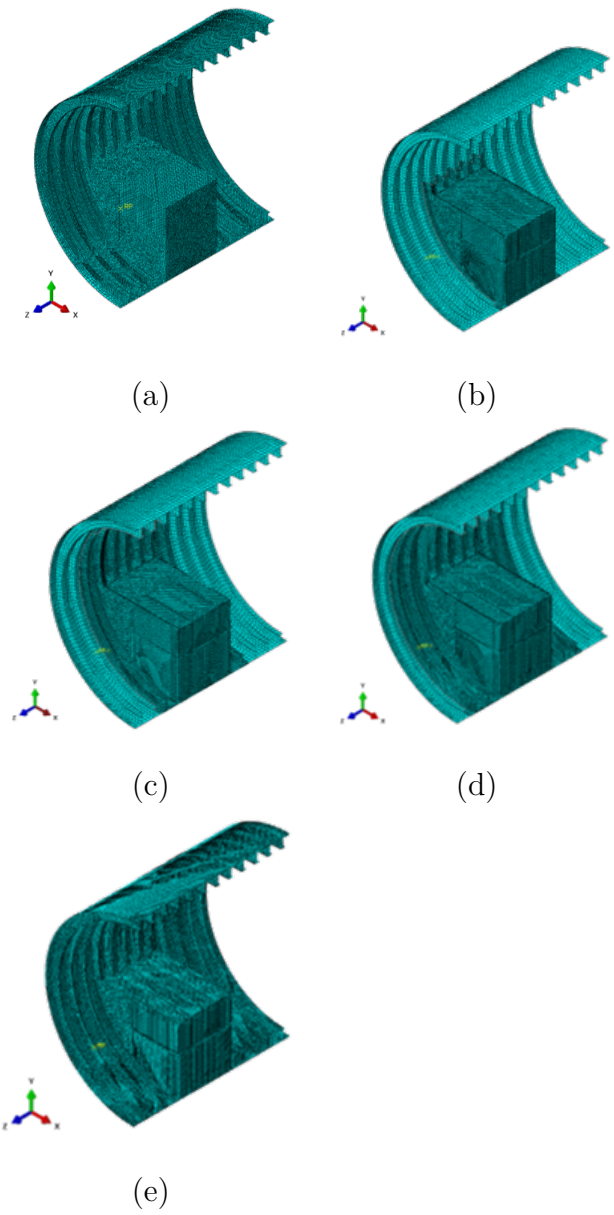



Figure 3.38: Baseline model meshes (a) full shell; (b) multi-fidelity configuration 1 (NbTank = 0.5, AngT = $5^0$); (c) multi-fidelity configuration 2 (NbTank = 1.5, AngT = $40^0$); (d) multi-fidelity configuration 3 (NbTank = 2.5, AngT = $60^0$); (e) full solid.

The relative percentage errors at the local domains for the different models is shown in Fig. 3.42. The local errors in the full shell model are high as it fails to capture the 3-D stress components at the tank wall joints. In particular, the chock-frame-tank top

intersection region is critical and the full shell model shows a significant error for the strain energy of the chock. This also verifies the need for a 3-D region near the hull-tank walls connection. All the mixed models have been modeled with 3-D elements in the tank walls and internal tank stiffeners with a varying topology in the hull. The use of a multi-fidelity model drastically reduces the errors for all the local domains and components.

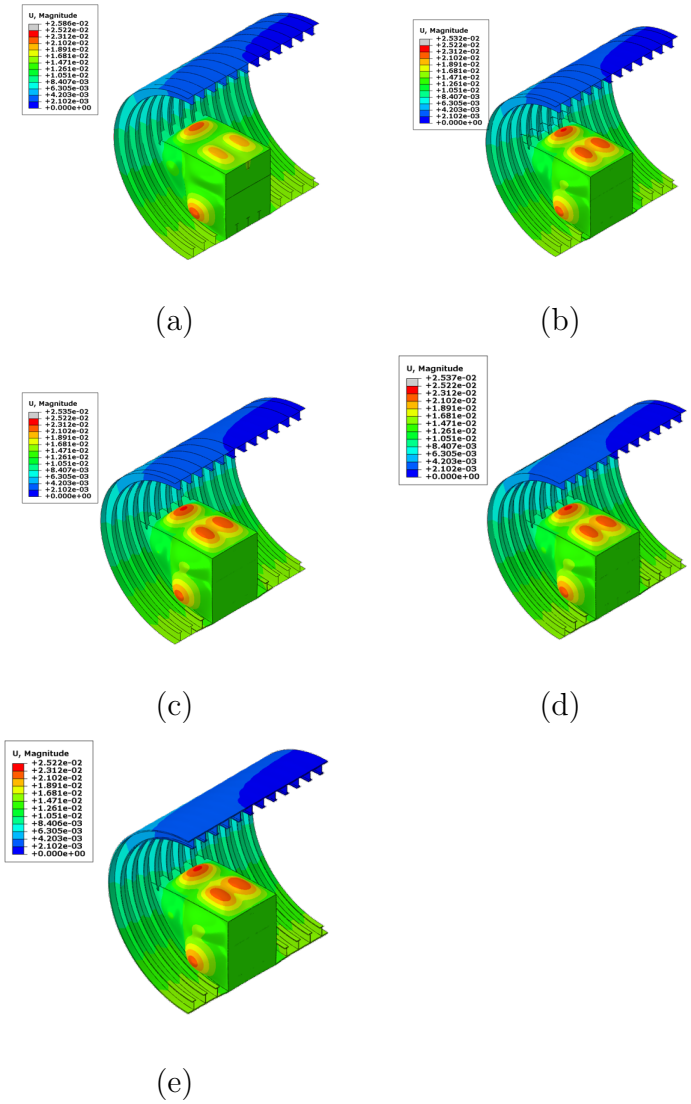



Figure 3.39: Baseline model displacement contours (a) full shell; (b) multi-fidelity configuration 1 (NbTank = 0.5, AngT = $5^0$); (c) multi-fidelity configuration 2 (NbTank = 1.5, AngT = $40^0$); (d) multi-fidelity configuration 3 (NbTank = 2.5, AngT = $60^0$); (e) full solid.

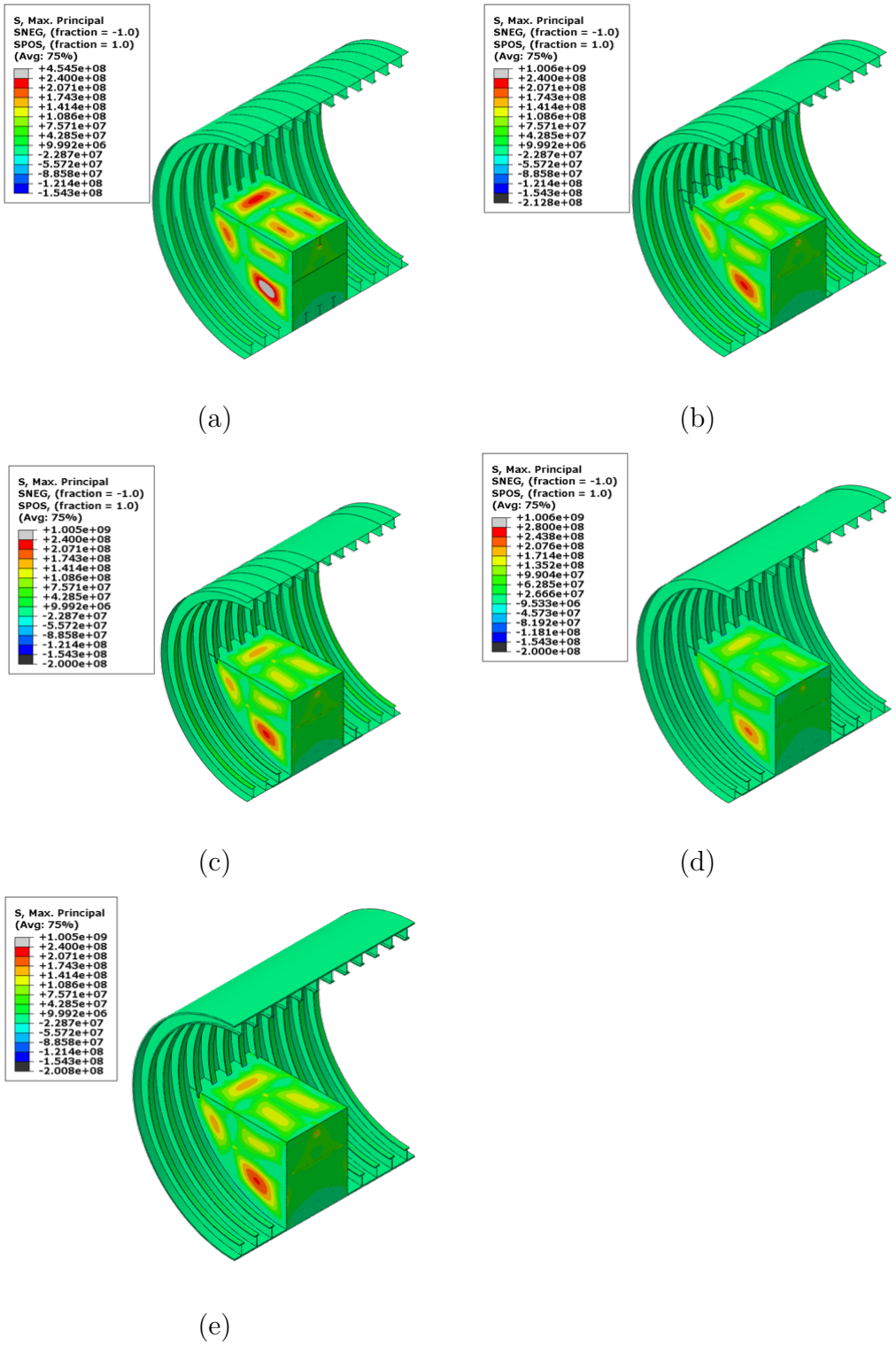



Figure 3.40: Baseline model maximum principal stress contours (a) full shell; (b) multi-fidelity configuration 1 (NbTank = 0.5, AngT = $5^0$); (c) multi-fidelity configuration 2 (NbTank = 1.5, AngT = $40^0$); (d) multi-fidelity configuration 3 (NbTank = 2.5, AngT = $60^0$); (e) full solid.

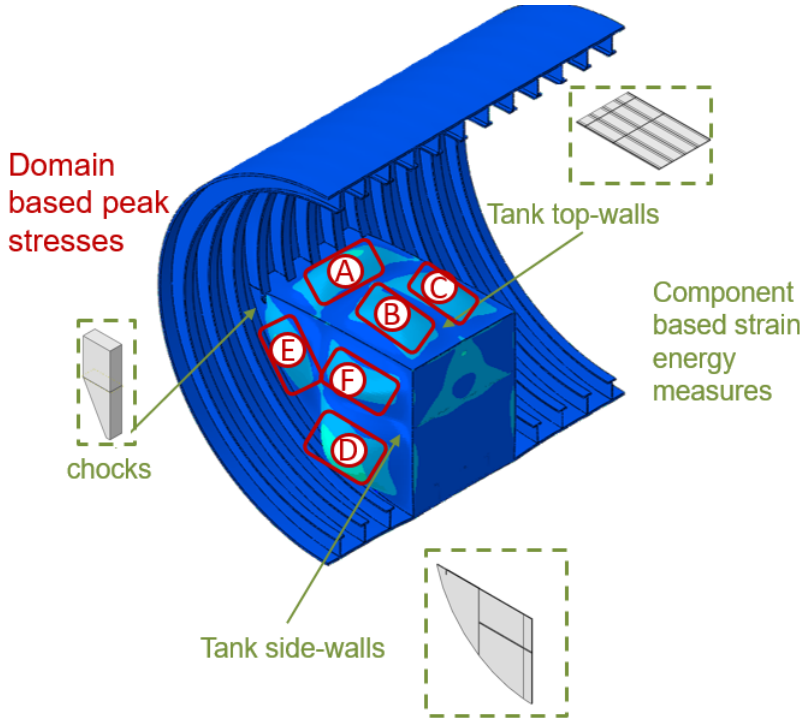



Figure 3.41: Error metric defined as a combination of peak stresses in user-defined local domains and component based strain energy values for comparison between models of different fidelities.

Additionally, an average of the local errors is combined into a single value and is plotted as shown in Fig. 3.43. A post-processing subroutine is developed that would open the output files in Abaqus/Standard and extract the needed values and also derive the relative error as compared to the corresponding full solid model. A comparison of the degrees of freedom relative to the average errors demonstrates that a multi-fidelity model with a minimal solid topology is sufficient to achieve accurate critical stresses with much less computational resources, as compared to the full solid model. Hence a multi-fidelity model can be considered useful in such large and complex structures. Of course, this specific multi-fidelity model is only sufficient for the geometric parameters of the baseline model. A variation in the tank assembly can lead to a different optimal mixed modeling topology and this is where an automated process for the development of such models could be

greatly advantageous.

In this work, similar mesh sizes are chosen for the solid regions in the mixed models and the full fidelity models. If different mesh sizes are used for the solid regions in the full solid and mixed models, there could be a coupled effect of the mesh size and element topology in the models on the optimal mixed modeling topology. It should be noted that in this work, the effect of element topology, in particular, shell-solid topology is studied. In future, a combined effect of the mesh size and element type would need to be considered to evaluate the overall effect of modeling fidelity variations.

Table 3.2: Comparison of computational efficiency of different fidelities.

| Model(NbTank, AngT) | Solid Elements | Shell Elements | Total | Nodes | DOFs |
|---|---|---|---|---|---|
| Full Shell | 0 | 77,598 | 77,598 | 77,397 | 464,377 |
| Mixed Fidelity ($5^o$, 0.5) | 184,013 | 9,357 | 193,370 | 234,386 | 732,611 |
| Mixed Fidelity ($40^o$, 1.5) | 235,872 | 7,314 | 243,186 | 304,188 | 936,309 |
| Mixed Fidelity ($60^o$, 2.5) | 282,203 | 5,313 | 287,516 | 367,084 | 1,119,279 |
| Full Solid | 302,539 | 0 | 302,539 | 422,122 | 1,266,367 |

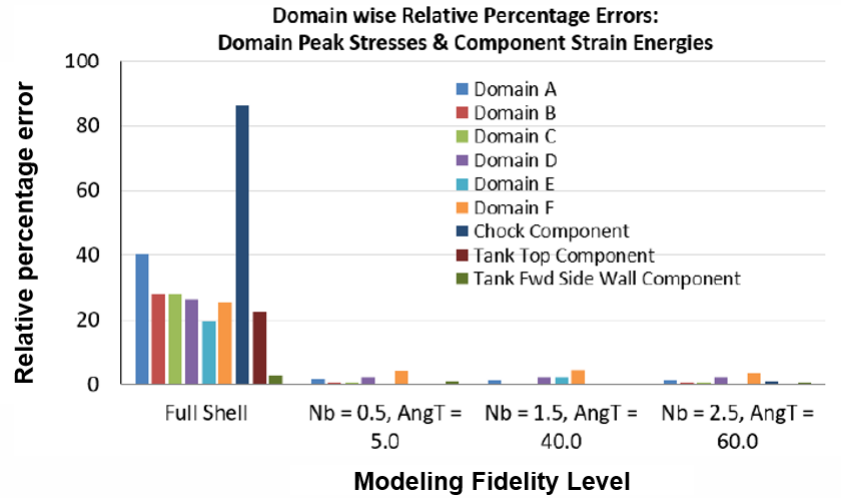



Figure 3.42: Domain-wise relative percentage errors for varying modeling fidelities.

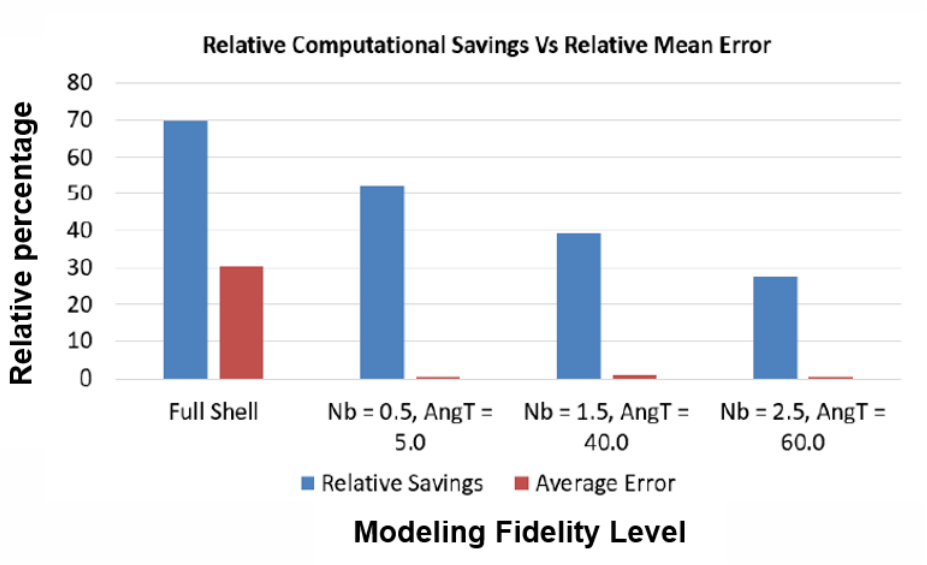



Figure 3.43: Relative mean error from modeling fidelities versus degrees of freedom (DOFs) of the models.

## 3.6 Conclusion

A framework for automated development of multi-fidelity models with variable elemental topology from full fidelity and structural fidelity models using parameterization was detailed. Additionally, insights gained from the automation of the tank model were discussed. An effective volume decomposition blueprint for obtaining robust hex-dominant meshes for complex multi-fidelity (coupled shell-solid) and full fidelity assembly models was presented. The advantage of using explicit shell-solid coupling to assemble shell and solid parts of the assembly, to allow for non-congruent meshing (for more computational savings) at the boundary was also presented. Moreover, automated post-processing and comparison techniques for evaluating models with different fidelities was also demonstrated. As industries are moving towards systems level multi-disciplinary optimizations [59], there is a need for sufficiently accurate and adaptable multi-fidelity models. The techniques

presented in the paper can be extended to be used for developing mixed models of other complex stiffened pressure vessel problems including, but not limited to, optimization of aircraft fuselage, stiffened launch vehicles with cryogenic propellant tank structures, multi-stage launch vehicles with thick stiffeners, and optimization of ring-stiffened ocean structures with composite laminates.

An application submarine tank structure with multi-fidelity elements was generated and analyzed to demonstrate the capabilities of the framework presented. It was shown that a multi-fidelity model can provide results with sufficient accuracy with substantial cost savings as compared to full fidelity and structural fidelity models. These models show that the integrated automation of the entire FE modeling and analysis have the potential to provide faster and accurate design concepts. It can be concluded that once developed, the generation of mixed models can be automated and utilized by industry for their design or optimization phases for quick and accurate analyses. The need for robust automation of mixed dimensional models is increasing and this work is a step towards achieving faster and accurate design decisions.

## Acknowledgments

The authors are grateful to the US NAVY for sponsoring this research under contract number: N68335-19-C-0117 and Topic Number: N171-059 with Nathan Leong as the Navy's technical monitor. The authors wish to extend special thanks to Mohcine Harrach for his work towards the parameterization and scripting of partitioning the complex tank assembly model. The authors would also like to thank Dr. Mohamed Jrad, Steven Doyle and Junhyeon Seo for their contributions and technical discussions during the project work.

# Chapter 4

# Parametric Study of the Submarine Tank Multi-Fidelity Model

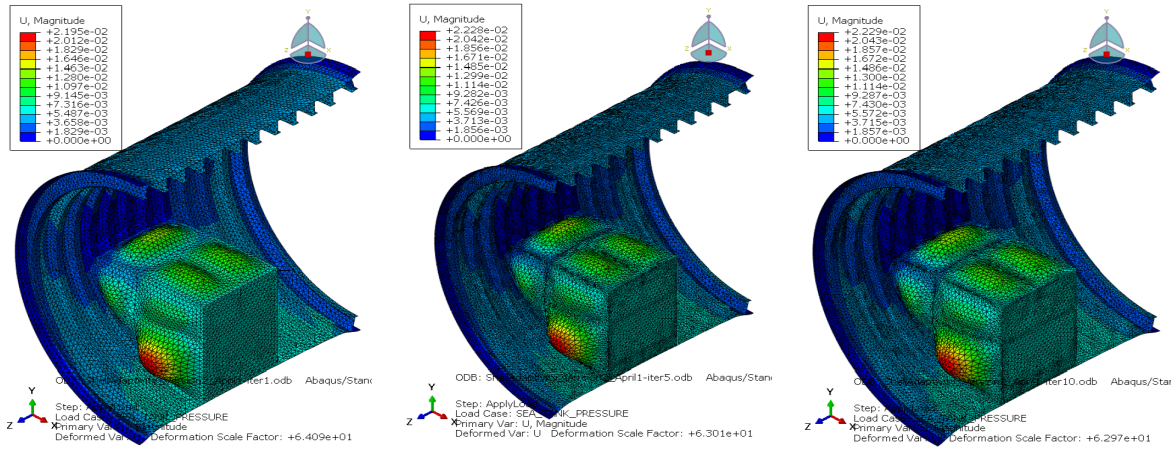
## 4.1 Identification of Critical Regions using Adaptive Meshing

Adaptive re-meshing is a procedure that is utilized to generate refined or coarser meshes in a model based on the distribution of discretization error estimators. Although the goal of an adaptive resmeshing procedure is to improve the solution accuracy by increasing fidelity levels in regions with a higher discretization error indicator, this process could also be employed to identify locations of interest *a priori* in a finite element model.

The submarine tank structure is a complex structural configuration with several connecting components and internal tank stiffeners subjected to external as well as internal pressure loads. The regions near the tank walls; especially, the region of the radial connecting hull with the rectangular tank-top wall, and the connection between the tank walls are critical regions which need careful consideration for design purposes. Based on these insights, the multi-fidelity modeling of the submarine tank structure utilized solid regions around the tank and shell elements everywhere else as shown in Fig. 3.21. In order to confirm these identified critical regions, a mesh adaptivity study using Abaqus/Standard is performed for the full solid and full shell models. For this study, the default C3D10, a 10-node tetrahedral element with quadratic interpolating functions is used for the solid model and the default S3R, 3-node triangular element is utilized for the shell model. A simpler model without periodic boundary conditions is utilized since the mesh keeps changing in every iteration of Abaqus/Standard. A combined loading of sea and tank pressure is applied with a fixed boundary conditions on the two hull ends. Adaptivity re-meshing rules based on a combination of strain energy and von Mises stress are utilized.

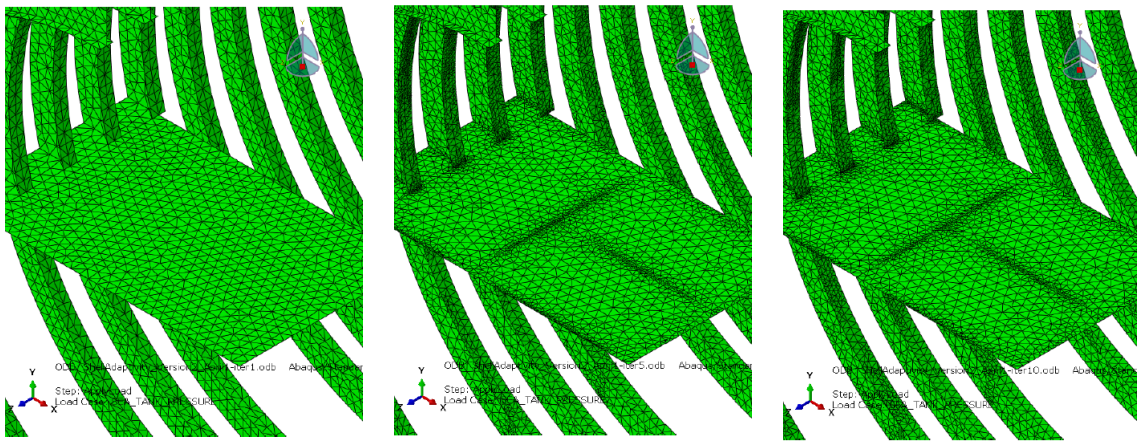
A bug in Abaqus/CAE was found where it incorrectly switches shell normal directions between adaptivity iterations. As such, the application of pressure loads cannot be applied

correctly to the full shell model. In order to mitigate this issue, surface traction loads with explicit direction definition as available in Abaqus/Standard are utilized to perform adaptivity of the full shell model. The displacement contours from adaptivity re-meshing results for the full shell model are shown in Fig. 4.1.




(a) (b) (c)

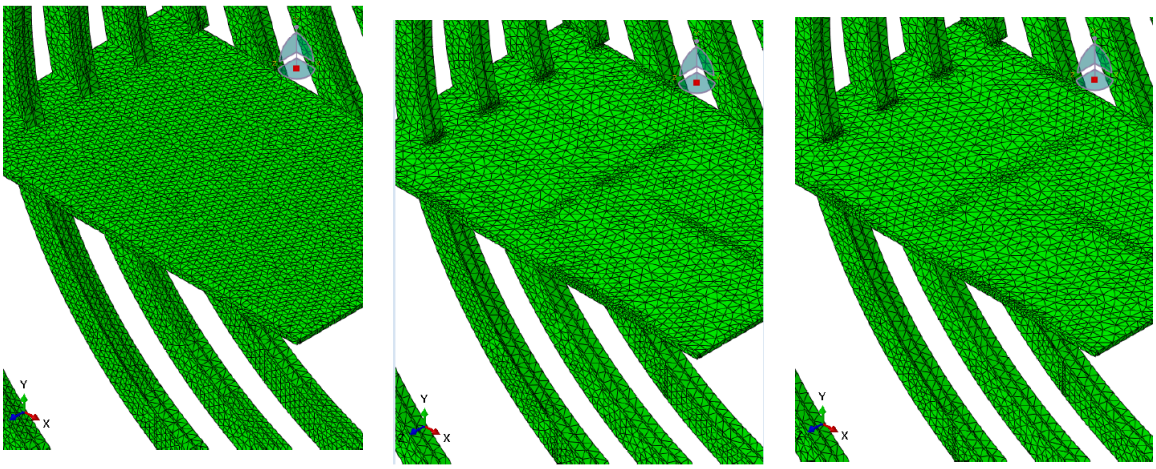
Figure 4.1: Displacement contours using adaptive re-meshing performed in Abaqus/Standard for a full shell model: (a) iteration 1; (b) iteration 5; (c) iteration 10.



(a) (b) (c)

Figure 4.2: Full shell model adaptive mesh refinement at the frames, tank top, and internal stiffener connections (a) iteration 1; (b) iteration 5; (c) iteration 10.

The connections at the tank top wall and frames are zoomed in Fig. 4.2 for clarity. The adaptivity results for the full solid model are shown in Fig. 4.3. It can be observed that the mesh adaptivity procedure for both models re-meshes the connection regions near the frames, tank top and the internal stiffeners as well as other connection regions during every iteration. This signifies a need for a higher mesh fidelity in these regions, and thus, a need for a higher modeling fidelity using solid elements.

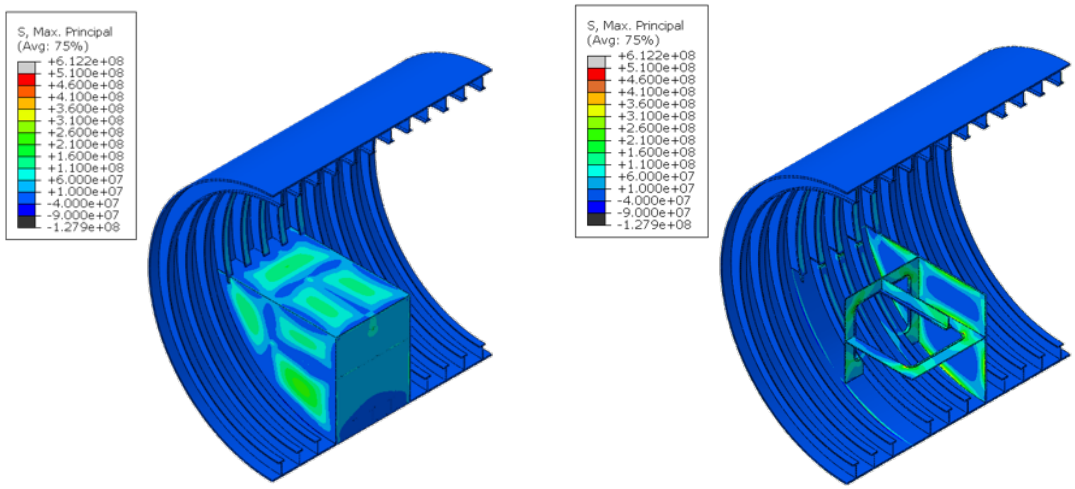


(a) (b) (c)

Figure 4.3: Full solid model adaptive mesh refinement at the frames, tank top, and internal stiffener connections (a) iteration 1; (b) iteration 4; (c) iteration 7.

This, in turn, also indicates that the tank walls intersection and the internal stiffener locations are one of the critical regions for the tank model. Thus, the previously defined multi-fidelity topology choices for the tank, *i.e.*, solid elements near the tank connections and internal stiffeners, and shell elements away from the tank are confirmed.

## 4.2 Effect of Variation in External Loading Parameters



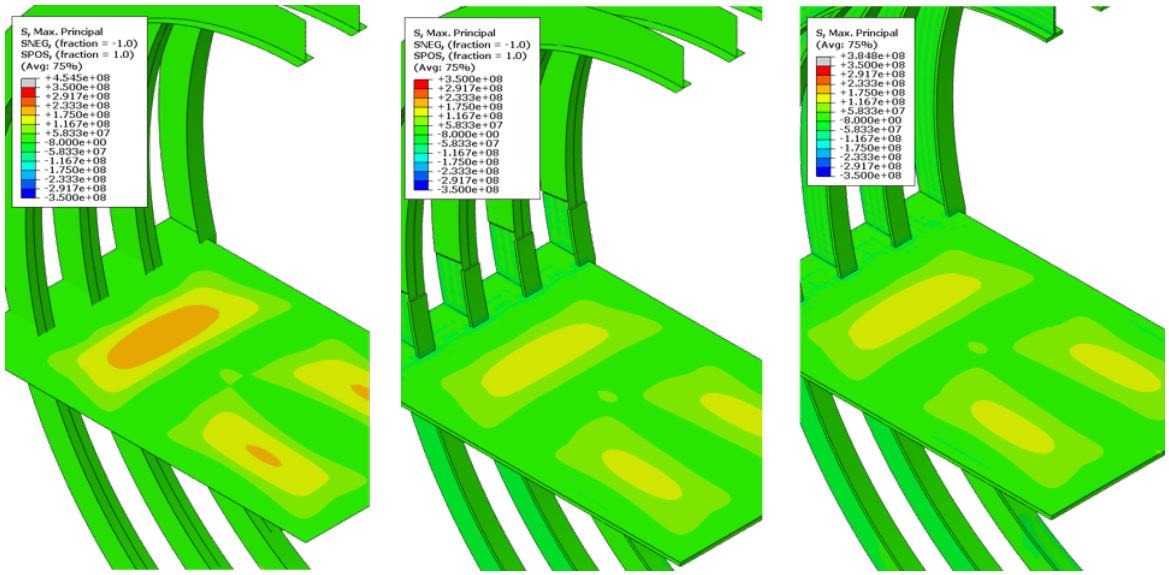

(a) (b)

Figure 4.4: Computational model of a submarine hull structure under a combined load of hydrostatic pressure at 240 m below sea level (2.35 MPa) and an internal tank pressure of 689.5 kPa: (a) with tank walls; (b) some tank walls not shown.

The considered submarine tank model is a deep-diving structure that typically experiences underwater depth range of 240 m to 540 m. Hence, in order to understand the effect of the external hydrostatic loads on the mechanical response of the tank structure, a parametric investigation is performed. The maximum principal stress contour plots for the baseline tank structure at a depth of 240 m (an external hydrostatic pressure of 2.35 MPa) and an internal tank pressure of 689.5 kPa is shown in Fig. 4.4.

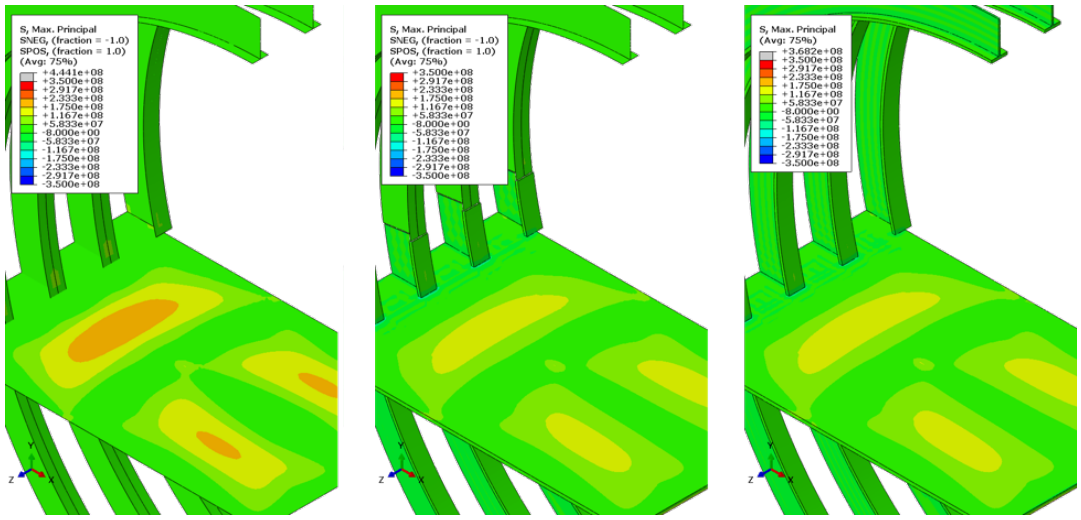
The model is subjected to a range of external pressure loads for three different depth levels, namely, 240 m, 440 m and 540 m. Figures 4.5, 4.6, 4.7 show the comparison of the contour plots of different fidelity models near the intersection of the tank-top wall with the middle

frames (other components are not shown for clarity). It can be observed that there is no significant change in the patterns of the stress contours for different underwater depth loads. Because the goal is to understand the effect of the multi-fidelity parameters which is dependent on the contour pattern variations at identified critical locations, the tank models are subjected to a constant load case of water pressure at 240 m below sea level and an internal tank pressure of 689.5 kPa for the further geometric trade studies presented.



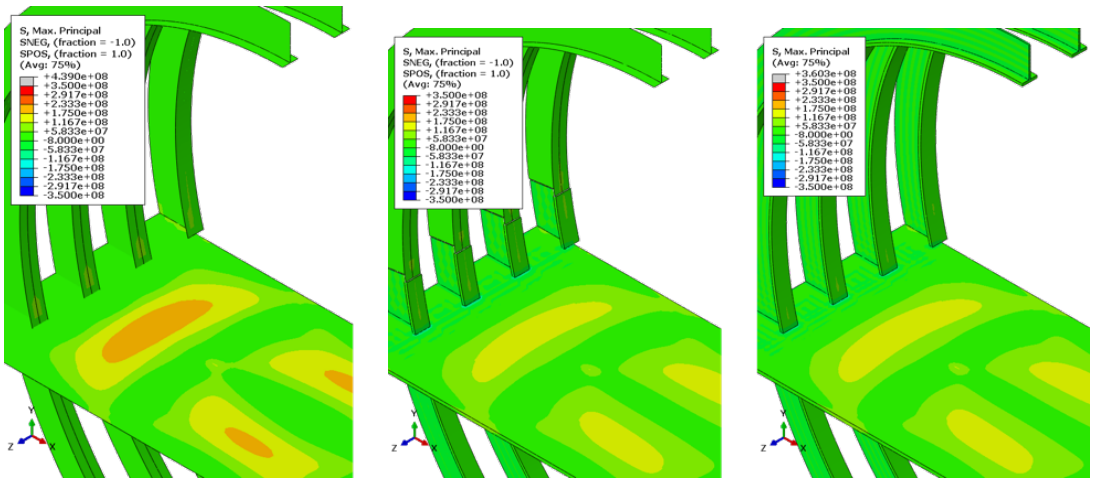

(a) (b) (c)

Figure 4.5: Computational models (parts removed for clarity) of a submarine hull structure under a combined load of hydrostatic pressure at 240m below sea level and an internal tank pressure of 689.5 kPa: (a) shell Model; (b) multi-Fidelity model; (c) solid Model.




(a) (b) (c)

Figure 4.6: Computational models of a submarine hull structure under a combined load of hydrostatic pressure at 440m below sea level (4.32 MPa) and an internal tank pressure of 689.5 kPa: (a) shell model; (b) ; multi-Fidelity model(c) solid Model.




(a) (b) (c)

Figure 4.7: Computational models of a submarine hull structure under a combined load of hydrostatic pressure at 540 m below sea level (5.30 MPa) and an internal tank pressure of 689.5 kPa: (a) shell model; (b) multi-Fidelity model; (c) solid Model.

## 4.3 Tank Model Trade Studies using DAKOTA

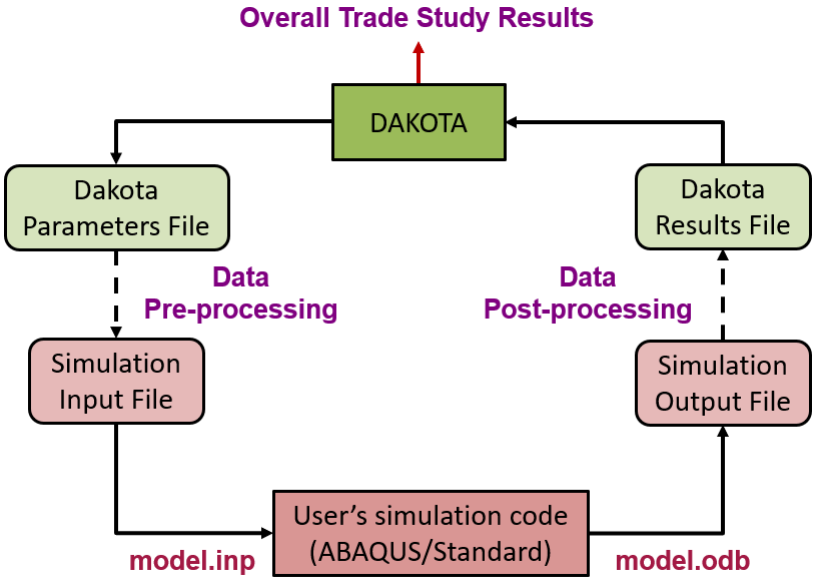
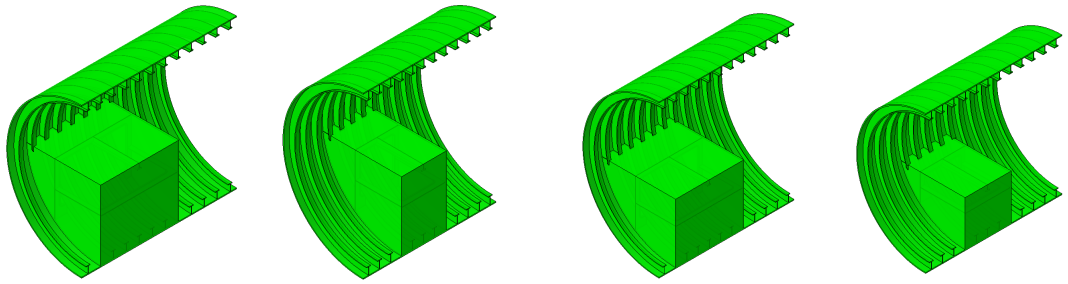



Figure 4.8: Schematic of the information flow between Dakota toolkit and Abaqus/Standard for parameter studies.[60].

The Dakota toolkit [60] developed by Sandia National Labs is utilized to perform a number of trade studies. Figure 4.8 demonstrates the flow of information between Dakota and the user-simulation code (Abaqus/Standard). The dotted-lines indicate the exchange/modification of information that must be scripted by the user. Once the data pre-processing and post-processing scripts are defined by the user, Dakota starts running the list of defined trade studies by submitting the parameter combinations to the simulation code in order to generate the simulation input file (in our case, 'model.inp'). It then reads the response from the simulation results file (in our case, 'model.odb') and adds it to a file that contains all the trade study results. A wrapper script is utilized to connect the pre-processing and post-processing codes between Dakota and Abaqus/Standard.

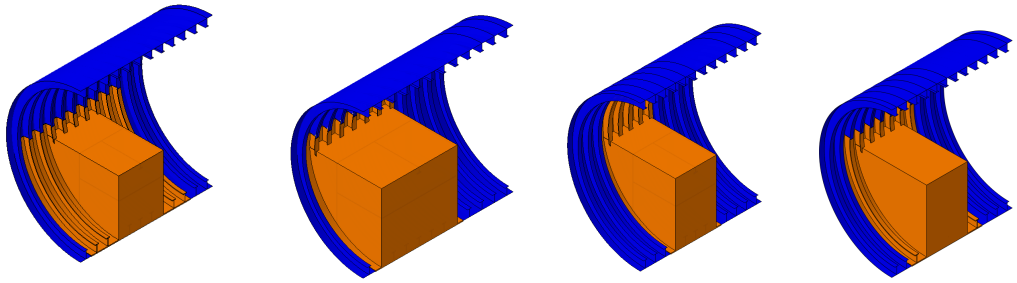
Table 4.1: Tank geometric and multi-fidelity modeling parameter bounds.

| Bound | $R$ | $l$ | $Nb$ | $w_2$ | $f_2$ | $NbTank$ | $AngT$ |
|---|---|---|---|---|---|---|---|
| Lower | 5.7 | 0.80 | 4 | $0.5 * R$ | 0.5 | 0.5 | 5.0 |
| Upper | 7.0 | 1.10 | 6 | $1.0 * R$ | 0.6 | 2.5 | 60.0 |



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

Figure 4.9: Representative geometries: (a) geometry A ; (b) geometry B; (c) geometry C; (d) geometry D.



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

Figure 4.10: Representative multi-fidelity topologies for different geometries: (a) Geometry A ; (b) Geometry B; (c) Geometry C; (d) Geometry D.

The number of potential parameters for the submarine tank problem is very large and only the most important variables are emphasized to develop the Design-of-Experiments (DOE)

for this parametric study. The critical parameters and their bounds are shown in Table 4.1. Dependent parameters, like the width of the longitudinal stiffener, $s_3$ and location of the longitudinal stiffener, $x_{loc}$, are determined based on the parametric values of the bay spacing and radius using the geometry parametric equations. The other non-critical parameters, namely, $t$, $t_1$, $m_1$, $c_1$, $c_2$, $c_5$, $s_7$, $s_8$, $b_1$, $b_2$, $f_1$, $f_3$ and $f_4$ are kept constant. A range of multi-fidelity topology parameters, $NbTank$ and $AngT$, as shown in Table 4.1 are studied for every each structural configuration of the tank. Instances of representative multi-fidelity models investigated are shown in Fig. 4.10. Fifteen geometries with a total of 300 multi-fidelity combinations are investigated.

## 4.4 Component-wise Strain Energy Error Measures

For this study, the aforementioned domain-wise volumetric strain energy and global strain energy relative error values are studied. These values are automatically computed using the python-based post-processing scripts developed to extract the values of interest from the component-wise 'element sets' defined for full-fidelity and multi-fidelity models. The critical components identified include chock, tank-top and tank side walls.

A comparison of all the local and global error measures individually for models with different fidelities was previously shown in Fig.3.42. These error error metrics show a drastic decrease in error levels from the use of full shell model versus the use of multi-fidelity models. However, once a sufficient practical solid region is used in the multi-fidelity models, errors become insignificant. Figure 4.11 shows the trends for all the geometries considered with respect to the change in the solid angular region, $NbTank$, for different number of solid side bays, $AngT$.

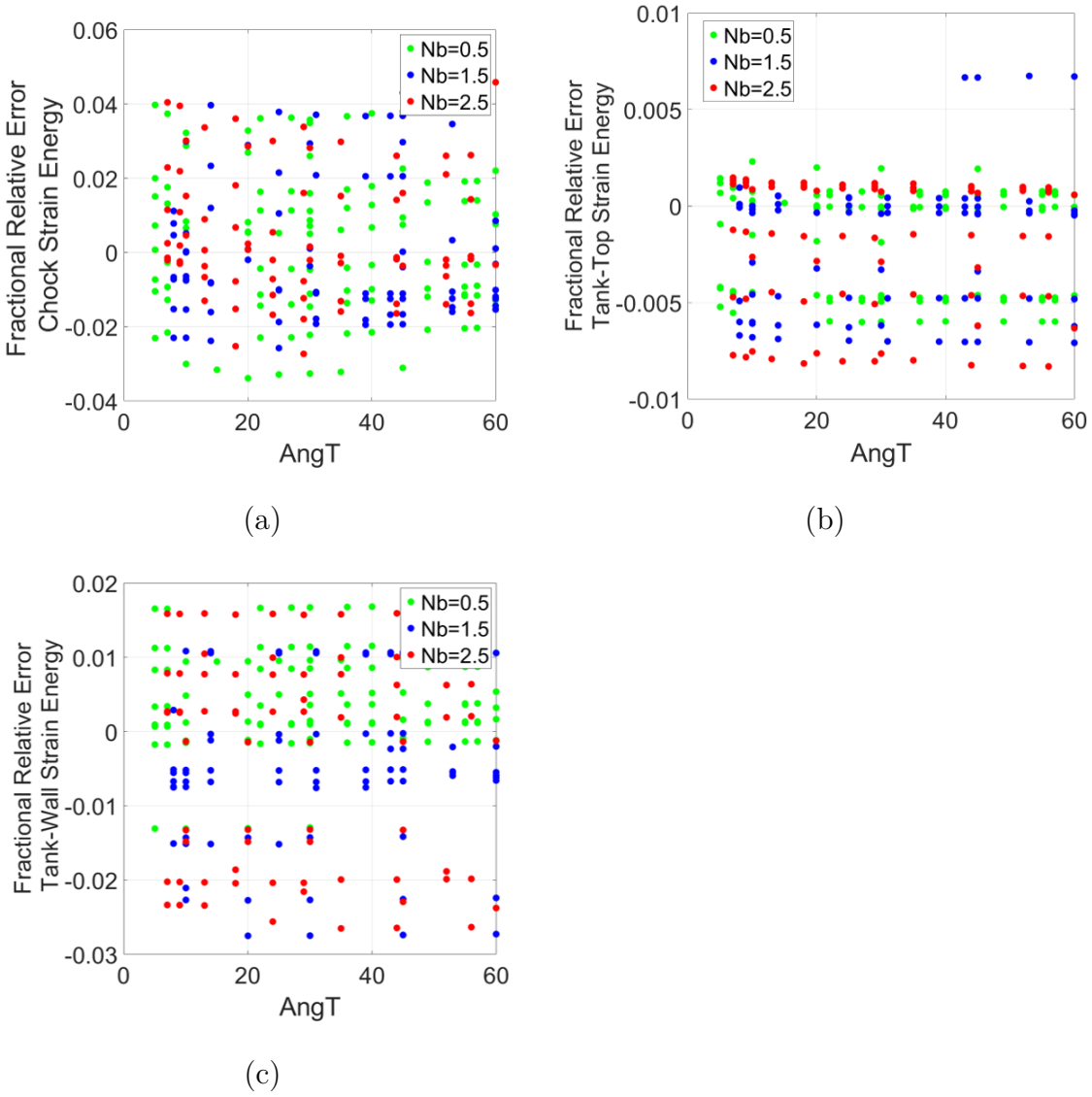



Figure 4.11: Domain-wise fractional relative errors in strain energy for all data-points: (a) chock; (b) tank-top component; (c) tank-side wall component.

It can be observed that all the different parametric multi-fidelity modeling topologies result in error values as small as close to 5% and there is no significant trend in the component-based strain energy errors with the rise in the fidelity levels. All the errors, however, are in the same levels of magnitude, indicating the confidence in using

multi-fidelity models to predict local strain energy values. The absence of significant trends in the local strain energy domains can be attributed to inherent scatter in the models, including mesh discretization and statistical variability due to presence of small values. On the other hand, the global strain energy absolute relative error measure, does show meaningful trends, albeit with small variations.

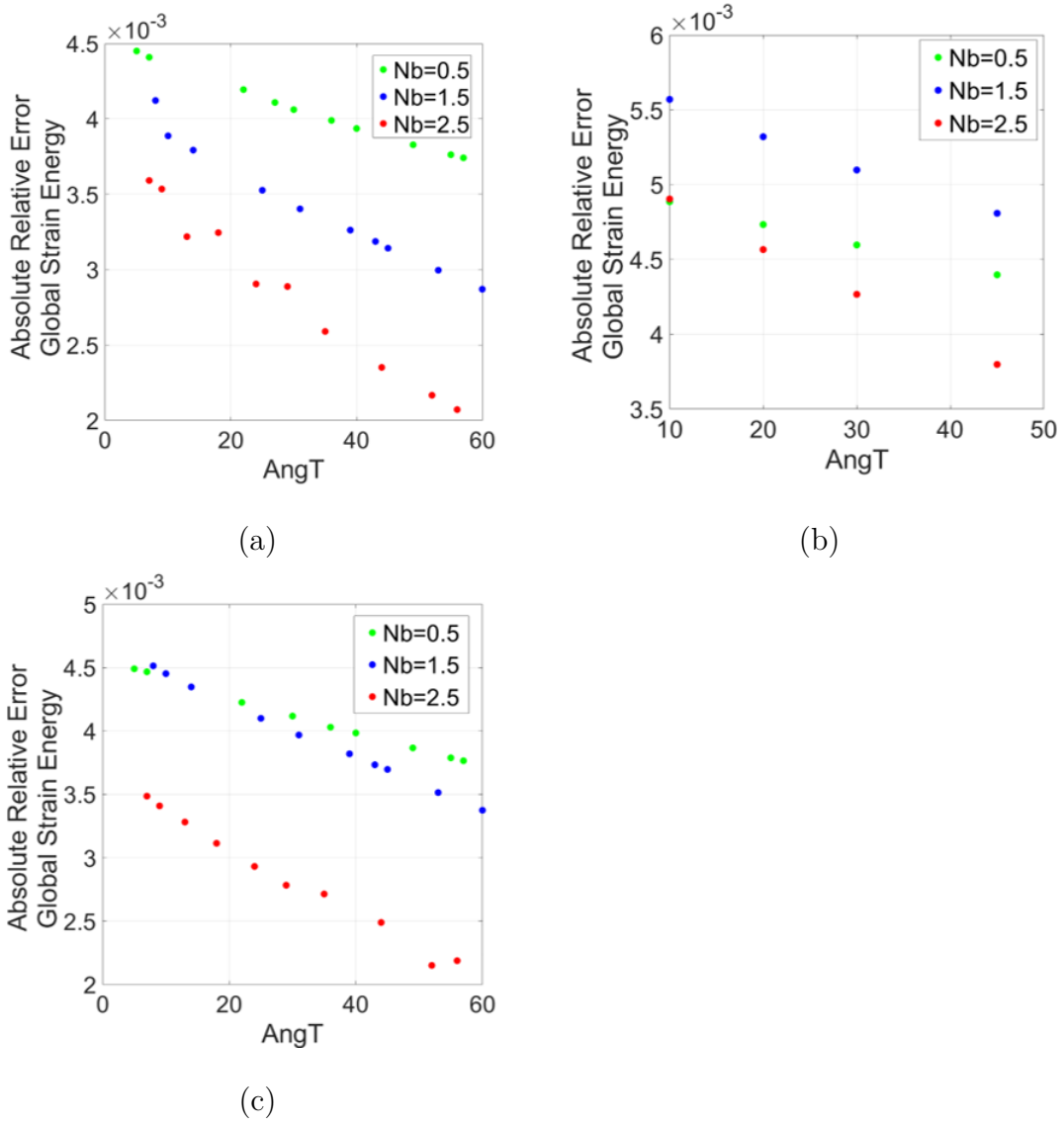



Figure 4.12: Representative multi-fidelity geometries: (a) multi-fidelity combination 1 ; (b) multi-fidelity combination 2; (c) multi-fidelity combination 3.

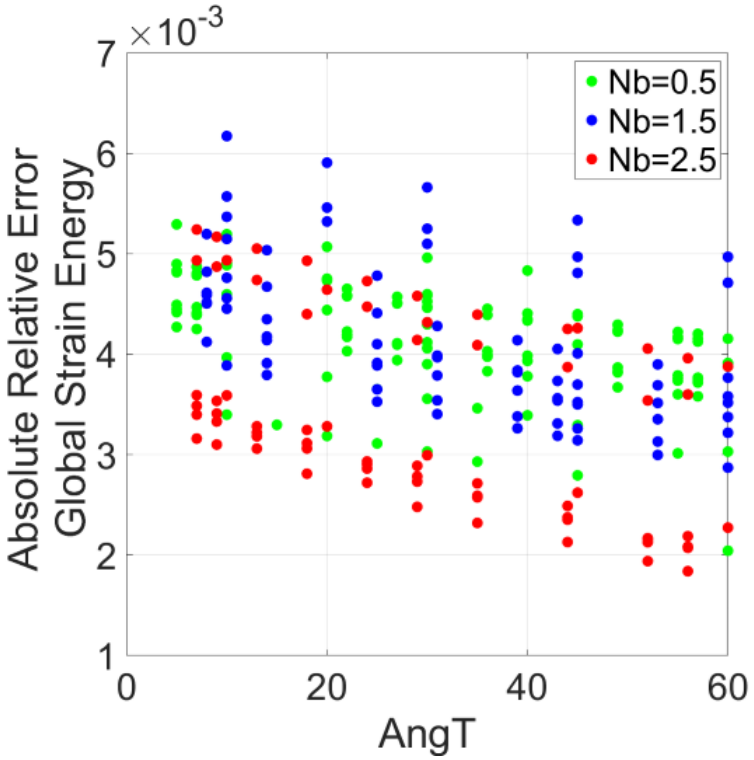


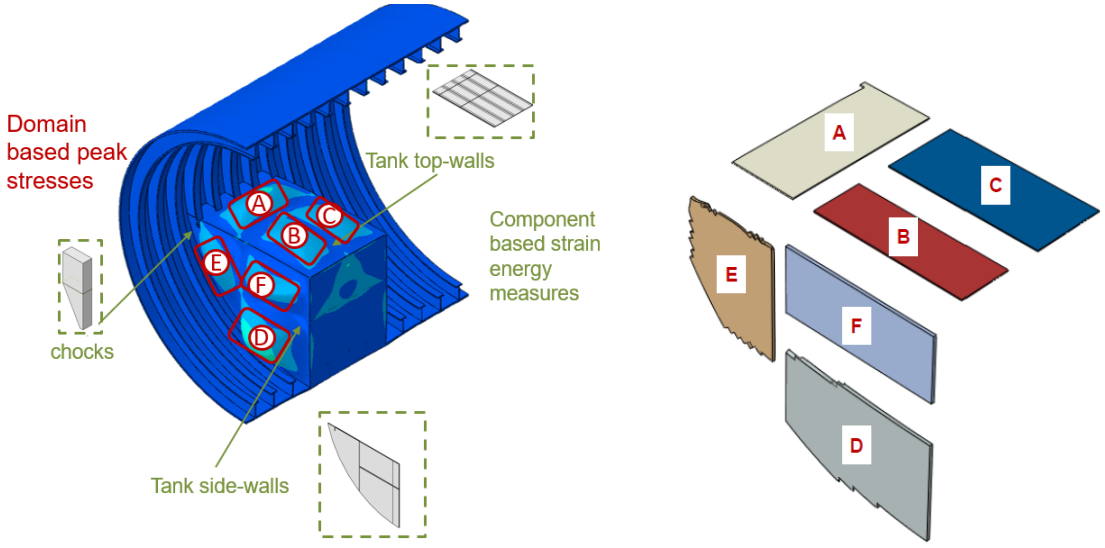

Figure 4.13: Effect of $AngT$ on the absolute relative error in the global strain energy value for all data points.

As shown in Fig. 4.12, the relative errors in the global strain energy for different tank structural and multi-fidelity configurations tend to decrease with an increase in the multi-fidelity parameter $AngT$, which represents the solid radial region above tank. This is understandable since the critical regions are the frames and tank top connections, and the size of solid region above the tank directly impacts the response. This trend is also captured in the overall graph as shown in Fig.4.13. This could be attributed to the ability of the overall strain energy values to take into account the complete variations in the element-wise (shell versus solid) topology in the multi-fidelity models.

The component-based domains are selected based on the pre-identified critical regions like the chock, tank-top and tank walls which are defined as solid regions irrespective of the modeling fidelity level. Once a sufficient level of fidelity is reached, the difference in the

component strain energies between the full solid and multi-fidelity models is almost negligible. This is because the chosen components are modeled as solid elements in the full-fidelity model as well as mixed-fidelity models, with similar mesh discretization. On the other hand, the global strain energy measure takes into account the whole model which includes the variations in the coupled shell-solid topologies and thus shows a meaningful trend, although with small numerical variations.

## 4.5 Domain-wise Maximum Principal Stress Error Measures




(a) (b)

Figure 4.14: Comparison of maximum principal stresses in different models using local domain-based comparison: (a) proposed domains along with the entire tank structure; (b) only proposed domains shown with other parts removed for clarity.

Finite element models with different dimensional fidelities often do not have a one-to-one correspondence with respect to meshes. In order to compare stress values, local domain-based stress comparison is proposed as shown in Fig. 4.14. These domains, as shown in Fig. 4.14(a), are selected based on the previously identified critical regions. Figure 4.14(b) shows the domains with other parts of the tank structure removed for clarity. Location A, B and C are defined at the tank top wall and location D, E and F are defined at the tank-side wall. The comparison is performed by obtaining nodal stresses that are extrapolated and averaged from the integration points in the models. As mentioned in Chapter 3, the linear C3D8R elements with reduced integration and hourglass control are utilized for the solid regions and the S4R elements are utilized for the shell regions. These regions are parameterized using geometry information such that the automation scripts can identify these local regions and define them as element and node sets. Once the element sets and node sets are defined, the maximum stresses in a the localized domains between different models can be compared. The parameterization bounds for all the different domains, A, B, C, D, E, and F are shown in Tables. 4.2, 4.3 and 4.4 respectively. The constant terms in the bounding equations are chosen such that the potential singularity-causing regions are avoided.

Table 4.2: Parameterized bounding boxes for selection of domains A and B.

| Bounds | A | B |
|---|---|---|
| Xmin | $-R + 2.2 * (f_2 + f_4)$ | $x_{loc} + 6 * t_1$ |
| Xmax | $x_{loc} - 6 * t_1$ | $-16 * m_1$ |
| Ymin | $-R + w_2$ | $-R + w_2$ |
| Ymax | $-R + w_2 + t_1$ | $-R + w_2 + t_1$ |
| Zmin | $4.5 * l + 6 * t_1$ | $4.5 * l + 0.5 * Nb * l + 5 * t_1$ |
| Zmax | $4.5 * l + Nb * l - 7 * t_1$ | $4.5 * l + Nb * l - 8 * t_1$ |

Table 4.3: Parametric bounding boxes for selection of domains C and D.

| Bounds | C | D |
|---|---|---|
| Xmin | $x_{loc} + 6 * t_1$ | $x_{loc} + 6 * t_1$ |
| Xmax | $-16 * m_1$ | $-22 * m1$ |
| Ymin | $-R + w_2$ | $-R + 0.2 * R$ |
| Ymax | $-R + w_2 + t_1$ | $-R + 0.5 * w_2$ |
| Zmin | $4.5 * l + 6 * t_1$ | $4.5 * l + Nb * l - 0.5 * t_1$ |
| Zmax | $4.5 * l + 0.5 * Nb * l - 5 * t_1$ | $4.5 * l + Nb * l + 0.5 * t_1$ |

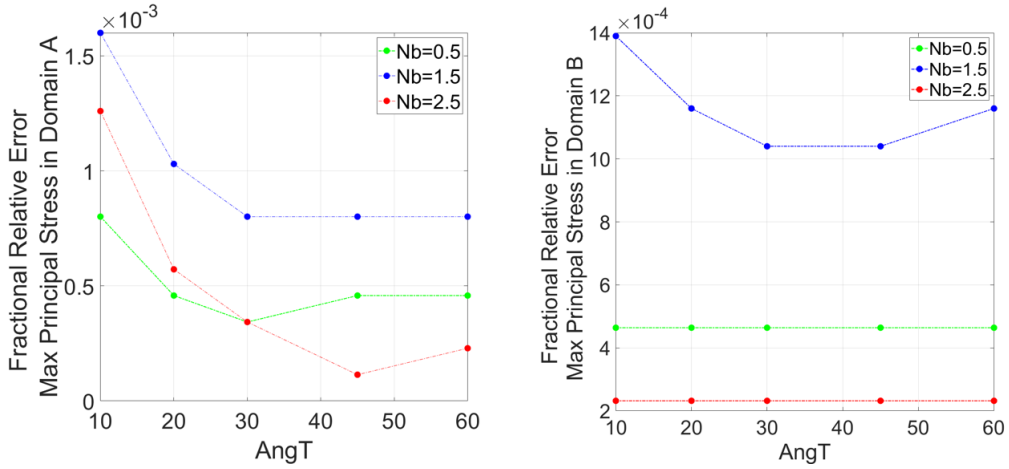
Table 4.4: Parametric bounding boxes for selection of domains E and F.

| Bounds | E | F |
|---|---|---|
| Xmin | $-R + 2.2 * (f_2 + f_4)$ | $x_{loc} + 7 * t_1$ |
| Xmax | $x_{loc} - 6 * t_1$ | $-22 * m_1$ |
| Ymin | $-R + 0.45 * R$ | $-R + 0.65 * w_2$ |
| Ymax | $-R + 0.9 * w_2$ | $-R + 0.95 * w_2$ |
| Zmin | $4.5 * l + Nb * l - 0.5 * t_1$ | $4.5 * l + Nb * l - 0.5 * t_1$ |
| Zmax | $4.5 * l + Nb * l + 0.5 * t_1$ | $4.5 * l + Nb * l + 0.5 * t_1$ |

Figure 4.15(a) shows the trends in fractional relative error values of maximum stresses in domain A, as the angular solid region ($AngT$) in the tank model is increased for the baseline geometry for different solid side bay regions ($NbTank$). It can be observed that although there seems to be no significant trend in the errors due to variations in the number of solid side bays, $NbTank$, there is a tendency for errors to decrease as the angular solid region ($AngT$) is increased. This pattern is consistent with the trend seen in

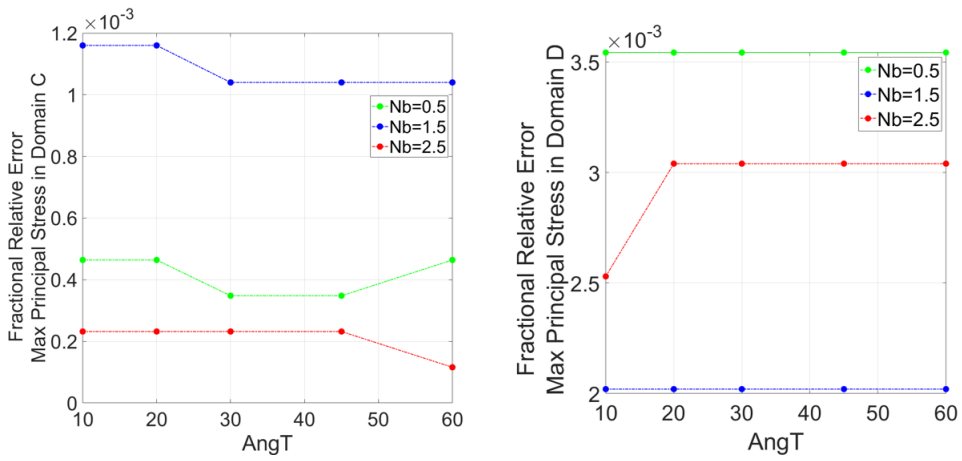
the global strain energy error metrics discussed in the previous section.

For the other local domain-based stress error measures as shown in Figs. 4.16 and 4.17, the error values remain more or less constant with slight ups and downs (which could be due to different underlying reasons including mesh discretization, inherent numerical extrapolation effect etc. and fuzziness in the system). Thus, the errors in maximum principal stresses in the local domains do not significantly vary and are scattered with noise and fuzziness due to previously mentioned reasons. This could be due to the fact that these domains are all solid regions with similar mesh discretizations in both the full-fidelity models and the multi-fidelity models. Moreover, as soon as a minimum required solid regions are used away from the tank walls, *i.e.* minimum practical values of $NbTank$ and $AngT$, there is very negligible difference between the corresponding domains. This is further coupled with the inherent scatter and noise in the system.



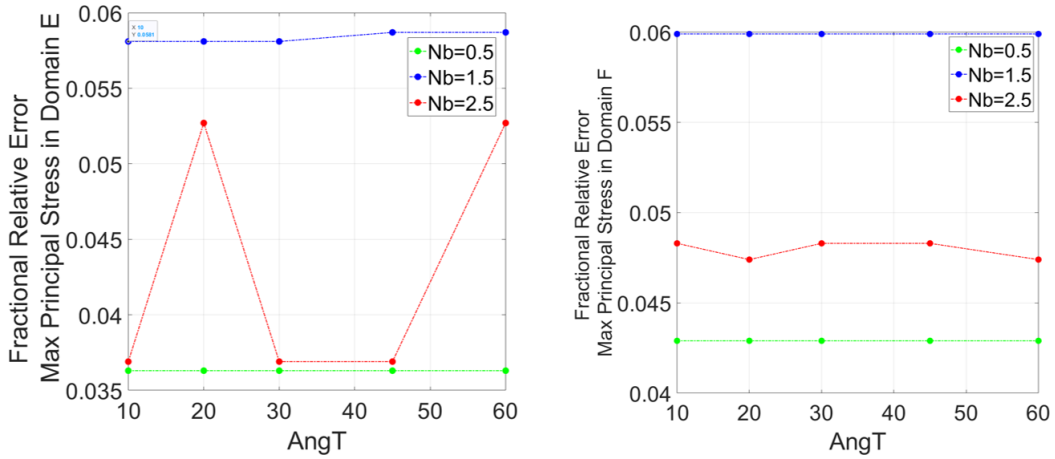

(a) (b)

Figure 4.15: Effect of multi-fidelity parameters on the domain-wise maximum principal stresses of baseline geometry: (a) local domain A ; (b) local domain B.




(a) (b)

Figure 4.16: Effect of multi-fidelity parameters on the domain-wise maximum principal stresses of baseline geometry: (a) local domain C ; (b) local domain D.




(a) (b)

Figure 4.17: Effect of multi-fidelity parameters on the domain-wise maximum principal stresses of baseline geometry: (a) local domain E ; (b) local domain F.

The domain-wise and global volumetric strain energy relative errors are between the full-fidelity and multi-fidelity models are bounded around 7% for all the multi-fidelity

combinations investigated. Thus, there is an excellent agreement between the full-fidelity models and multi-fidelity models, and the presented coupled shell-solid models can be utilized with confidence for a range of large submarine tank structures. A comparison of the computational costs of different modeling fidelities was presented in Fig. 3.43 for the baseline tank model, which demonstrated the efficacy of utilizing multi-fidelity models for the complicated tank structure.

## 4.6 Multi-Fidelity Modeling Recommendations for the Tank Structure

Table 4.5: Modeling recommendations for the complex submarine tank structure.

| Bound | $NbTank$ | $AngT$ |
|---|---|---|
| Practical | 0.5 | 5.0 |
| Conservative | 1.5 | 30.0 |

Based on the parametric investigation performed, a minimum practical solid region in the critical areas is sufficient to provide excellent accuracy with significant computational savings. It was also observed that the effect of increasing solid region above the tank-top has a higher effect on improving the accuracy in the regions of interest. Hence, it would be useful to fix the number of solid side bays ($NbTank$) to a minimum practical value and increase the amount of angular solid region ($AngT$) above the tank-top wall. Table.4.5 provides some practical and prudent bounds that can be applied when developing multi-fidelity coupled shell-solid tank problems as discussed in this work.

## 4.7 Discussion: Effect of Multi-fidelity Parameters on Proposed Error Measures

Based on the parametric investigation performed, it was found that for tank geometry ranges considered and for the critical regions of interest, a solid tank region and a minimum solid region above and sideways of the tank is sufficient to obtain very high accuracy. Thus, the applicability of coupled shell-solid finite element models for a realistic and global submarine tank structure by performing parametric studies of different geometries, loads and modeling topolgies is confirmed.

With regards to the proposed error measures, it was found that because all the component and local domains were modeled using solid elements in all the different modeling fidelities involved and a minimum solid region away from the tank was enough, the errors were negligible without any meaningful trends. However, the trend and the effect of variation of the multi-fidelity parameters is clearly visible on the global strain energy measure, which could be attributed to the fact that this value is affected by both the solid and shell regions of the model.

Application to more complex problems would confirm the benefits of using localized error metrics for providing reliable comparison between different modeling fidelities. Additionally, there will always be slight uncertainties involved in FE modeling of complex structures due to fuzziness in mesh discretization, inherent numerical uncertainties etc. and thus, a tool that takes into account the fuzziness and provides optimal modeling recommendations is highly desirable.

Chapter 5 presents a framework for decision tree based fuzzy inference system, to provide reliable modeling recommendations regarding the optimal modeling fidelity choices for a

coupled shell-solid models. The proof-of-concept for this expert system is demonstrated with a component level T-joint structure.

# Chapter 5

# Making Finite Element Modeling Choices using Decision Tree based Fuzzy Inference System

This chapter presents a manuscript that is submitted to the AIAA Journal and is currently under review.

[1]

[1] **Palwankar, M.**, Kapania, R.K., Hammerand, D., "Making Finite Element Modeling Choices using Decision Trees based Fuzzy Inference Systems," AIAA Journal (submitted and under review).

## 5.1 Abstract

This work presents a Decision Tree (DT) based Fuzzy Inference System (FIS) for making optimal choices in the development of reduced-order Finite Element (FE) models; in our case, coupled shell-solid multi-fidelity models. FE analysis is widely used to simulate the real-world response of complex engineering structures and requires a high-level of expertise for making *a-priori* modeling decisions. Many times, these decisions are quite subjective in nature and lead to significant analyst-to-analyst variability, which in turn leads to considerable differences in engineering solutions. An expert system that recommends optimal modeling choices would notably reduce such variability. Expert systems utilize a knowledge base, developed by a subject matter expert, which is not always easy for complex structures. This work assesses the potential of interpretable machine learning (DTs) to create data-driven rules that could be utilized by a FIS to make modeling choices for a multi-fidelity T-joint model. Specifically, the FIS takes the structural geometry and desired accuracy as inputs, and infers the optimal 2D-3D topology distribution. Once developed, the FIS is able to provide real-time optimal choices along with interpretability that fosters analyst's confidence. The utilized framework can be generalized to complex non-linear finite element modeling and adaptable mesh generation.

## 5.2 Introduction

Finite element analysis (FEA) is widely used to numerically simulate the real-world response of complex engineering structures. It attempts to approximate the solution of a partial differential equation by discretizing complex continuum domains into finite elements and by using these elements to develop a set of linear or non-linear algebraic

equations [4]. There are various element types available that can be utilized, and a structural analyst usually makes choices based on their experience and expertise to select the most appropriate element configuration.

In general, 3-D solid finite elements have no *a priori* assumptions about the structural response and hence provide higher accuracy with adequate discretization. However, using them quickly becomes computationally expensive. Shell elements, on the other hand, are relatively economical for thin structures, but have certain restrictions in terms of structural response in the shell's transverse direction. In aerospace and automotive industry, thin-walled structures are commonly employed which can be analyzed using shell finite elements to represent the mid-surface mesh of the structure. Shell elements have multiple integration points through the thickness along with rotational degrees of freedom to include bending effects, whereas solid elements only have translational degrees of freedom. Hence, under bending loads, multiple solid elements are needed through the thickness of the structure to capture the bending response. In situations like these, it is more convenient to utilize 2-D surface elements like shells, which neglect both normal and transverse shear stresses in the thickness direction. However, utilization of only shell elements in the presence of complex structures (having connections, joints etc.) is problematic [8].

As a way to combine the benefits of different element types, mixed solid/shell models are utilized [10, 13, 14, 16, 58] to capture the critical features of a complicated structure with significant reduction in the global degrees of freedom as compared to a full 3-D solid model. Note that careful consideration of the 3D-2D coupling is required to avoid errors in the complex regions of interest. Such multi-fidelity models are common in the initial design stages rather than during the final steps, wherein full fidelity models are employed for highest accuracy. As compared to a sub-model, which requires a two step analysis, including an initial global analysis followed by a sub-model analysis, the coupled shell-solid

model uses a single step analysis. Krueger *et al.* [13] presented a shell/solid modeling technique for analyzing delaminated composite laminates. Other works by Krueger *et al.* [9, 14], used shell/3D modeling technique for analyzing skin-stiffener debond specimens and performing debonding. Figure 5.1 shows a typical multi-fidelity shell/3D model developed by Krueger and Minguet [9] to analyze a debond-panel specimen. Peric *et al.* [58] utilized a shell/3D model to perform a numerical and experimental investigation of residual stresses and distortions due to a welded T-joint using a shell/solid modeling.

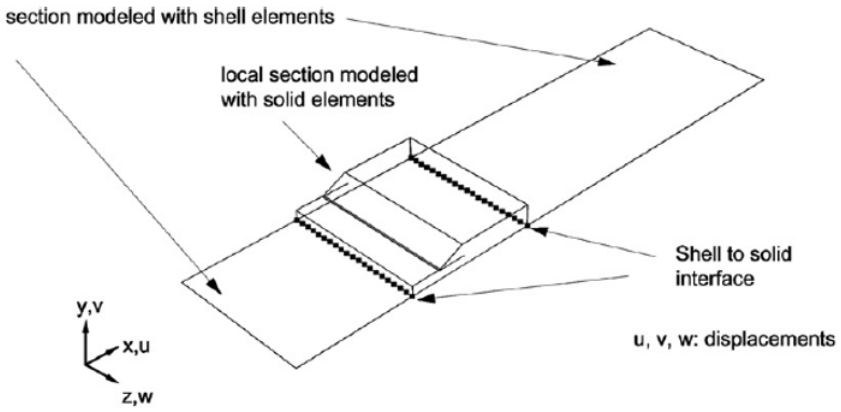



Figure 5.1: Example of a shell/3D model utilized by Krueger *et al.* [9] for debond specimens.

Complex computational simulations with adaptive optimizations are clearly on the rise and multi-fidelity models are being increasingly sought. Development of such a multi-fidelity model, however, requires crucial *a-priori* modeling choices to be made regarding its geometric fidelity. Many times, these modeling decisions can be quite subjective in nature and lead to significant analyst-to-analyst variability in critical modeling choices, which in turn leads to considerable differences in obtaining reliable engineering solutions. Glickman and Romero [37] demonstrated the significant effect of analyst-to-analyst variations in the development of computational models, and the need for efficient ways to eliminate them. This is especially important in complex structures with multiple connections, where a shell

model would be insufficient but a full solid model is computationally expensive. The ideal procedure is to only use 3-D solid elements where it is critical to capture the details of the transverse structural response and to model the remaining thin-walled areas (often the largest portion) with shell elements [38]. However, the problem of determining element topologies in a multi-fidelity model is itself fuzzy in nature and depends strongly on the context of the problem [39]. Moreover, this choice becomes additionally vague during the preliminary design of complex engineering structures or analysis of aging structures where the effect of geometry, loading and environment conditions is not precisely known [40]. This leads to significant on-going adjustment of model development in industry and is still a major bottleneck. In the review work by Naganarayana and Prathap [7], the need for an expert system in modeling complex structures is emphasized. A decision support system that recommends optimal modeling choices can aid in removing this subjectivity and thereby help analysts make confident choices.

Many researchers have focused on developing an intelligent system for structural analysis. Novak *et al.* [41] summarized the state-of-the-art of structural analysis based design improvements into a collective advisory intelligent support system. This work demonstrated the usefulness of such expert systems in the design of new products or as an educational tool. Breitfeld *et al.* [42] presented an expert system using a knowledge base and object oriented programming for the verification of finite element analysis. In another work, Dolsak *et al.* [43] proposed an intelligent decision support for structural design analysis and emphasised the useful of such an intelligent advisory tool. Another advisory system was developed by Hetey *et al.* [44] for reliable FEM modeling in aerospace. This tool involved global as well as specific knowledge rules in the form of dialogue boxes for selecting element types, boundary conditions etc. These works, however, utilized knowledge bases which were prepared by subject matter experts. For complex structural models, there

is hardly a binary decision regarding different modeling choices and hence, the development of a rule base is not straightforward.

Some researchers have focused on building expert systems using machine learning techniques like Artificial Neural Networks (ANNs). Yeh *et al.* [45] presented an expert system for debugging FEM input data using ANNs and showed that ANNs can work sufficiently as a knowledge acquisition tool. However, a major limitation of Yeh et al.'s work was that the ANNs that they employed lack the ability to explain their reasoning, which is critical for establishing user confidence. Another work by Li *et al.* [46] presented a hybrid expert system for finite element modeling of fuselage frames. Although they utilized ANNs for classifying different frame structure types, the knowledge base was developed manually using expert knowledge.

There is a significant demand for an expert system framework which can learn rules automatically and also be consistent with physics of the problem to provide intelligent support for complex FE multi-fidelity modeling decision making. This work presents an expert system using fuzzy logic and Decision Trees (DTs) for recommending the optimal modeling choices for developing multi-fidelity models. Fuzzy logic, first introduced by Zadeh [47], has gained popularity due to its successful implementation in application areas of automatic control, expert systems, fault diagnostics, decision analysis etc. Fuzzy systems have shown success with treating imprecise and subjective information into the decision making of engineering systems[40, 48]. Work by Manetviz *et al.* [49] on utilizing soft computing for automating FEM is worth mentioning. The authors argued that the overall intelligent FEM package is a 'test-bed' for many fuzzy computing techniques.

Mamdani Fuzzy Inference System (FIS) is a control system that can make conclusions blending together a set of linguistic control rules obtained from human experts. It has been widely utilized to develop expert systems for a variety of problems where fuzzy diagnostics

is useful. An important input to the fuzzy systems are the system rules, which have been developed using experience of human experts. Rao *et al.* [51] developed a fuzzy logic-based expert system to predict the results of finite element analysis by using element size and shape ratio as inputs. Their work showed that fuzzy logic can predict FE results within reasonable accuracy. Their approaches utilized manual knowledge base construction as well as Genetic Algorithm (GA) based rule extraction. The GA knowledge base although data-driven, is both dependent on the encoding of the rules and is computationally expensive. Another work by Sangiovanni *et al.* [52] presents a fuzzy logic based finite element mesh design. The authors emphasized the benefit of learning using learning algorithms such that the rules can be adapted to different problems, as compared to the classical mesh adaptive methods which are sensitive to changes in the geometry.

This is where decision trees prove can be especially attractive, since they automatically learn decision-making rules from the data and can be extracted by simply traversing paths from the root to nodes in a trained model. This concept has been proposed and effectively used in the past [54, 55, 56, 57] for different applications. Decision trees have been widely used for extracting rules and performing classification for different applications. As compared to black-box classification and prediction models like ANNs and Support Vector Machines (SVMs), decision trees have the advantage of being easy to interpret and also visualize in a tree structure. More importantly, decision trees directly learn rules from data in the form IF-THEN blocks which can then be easily utilized to develop a fuzzy inference system. The learning ability of decision trees to provide rules for fuzzy systems, which utilize high-level human-like reasoning has been proved to be effective and easier to interpret. Until now, significant work has not been performed to develop an expert system that provides optimal element topology recommendations for multi-fidelity models. This paper focuses on the implementation of an expert system framework using decision trees

and fuzzy logic for making modeling recommendations for developing optimal multi-fidelity models (coupled shell-solid).

Although the capability of the proposed framework is demonstrated using a T-joint structure, this methodology can be readily transferred to more complex structural configurations. The paper makes the following contributions:
1) Implementation of a decision support system for modeling choices for the development of multi-fidelity models.
2) Consideration for imprecise data for modeling recommendations using fuzzy logic.
3) Utilization of decision trees to extract fuzzy rules for modeling recommendations which lead to an automated knowledge base related to structural behavior.

The paper is structured as follows: Section II contains the methodology used for designing numerical experiments and generating data for the static analysis of a T-joint. Section III contains a theoretical background on fuzzy logic-based Mamdani inference system and decision trees, along with the developed expert system framework. Finally, conclusions and future work have been discussed in the section IV.

## 5.3 Methodology

### 5.3.1 Hierarchical Qualification Approach for Multi-Fidelity FE Modeling

A hierarchical qualification/validation approach is implemented as a preliminary step towards comparing models with different fidelities in Fig. 5.2. Note that Fig. 5.2 is adapted from the ASME guide on verification and validation [61] and enhanced with the notion of hierarchical qualification which will be explained subsequently. Although, a final

assessment of computational models must include experimental results, it is not always possible to perform physical experiments or often, such data is not available until later years in an overall development program. In such cases, a hierarchical qualification scheme can be utilized to compare lower and multi-fidelity models against the ones with higher fidelity and gain confidence in modeling results. Note that a higher fidelity model is not intended to be restricted to verification refinements (*e.g.*, refined meshes) but rather, they can involve additional physics in their modeling. Here, a higher fidelity model involved detailed geometric representation with higher element fidelity (*e.g.*, 3-D continuum solid versus 2-D shell). This paper is solely focused on multi-fidelity models involving solid versus shell modeling. In particular, the C3D8 and the shell S4R elements of Abaqus/Standard are utilized as representatives of three-dimensional and shell elements respectively. However, our approach can be applied to any set of solid and shell elements. A multi-fidelity model leads to situations where the solid element's surface needs to be numerically compatible with the shell edge as shown in Fig. 5.3. In this work, Abaqus/Standard is utilized to employ coupling constraint equations to ensure that the bending rotations and moments are correctly transferred between the two domains in a self-equilibrating manner [11]. Note that shell-solid coupling is a source of error in the immediate vicinity of the coupling region and care is taken to place the shell-solid coupling regions away from the regions of interest.

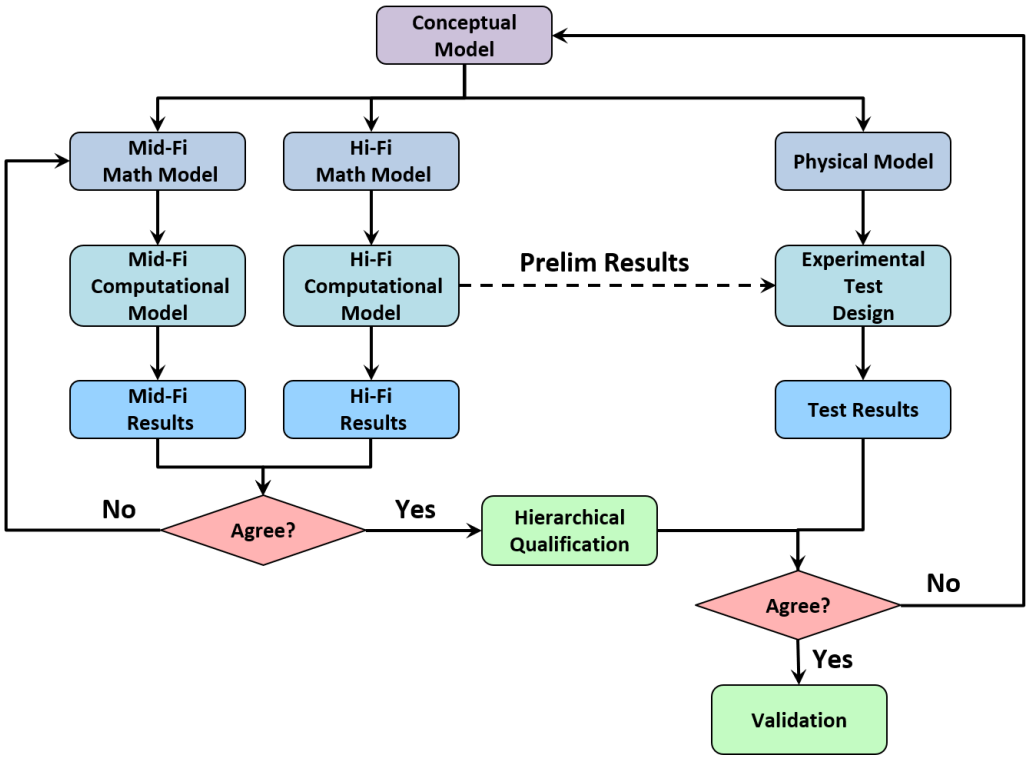



Figure 5.2: A hierarchical qualification approach used for comparing different fidelities.

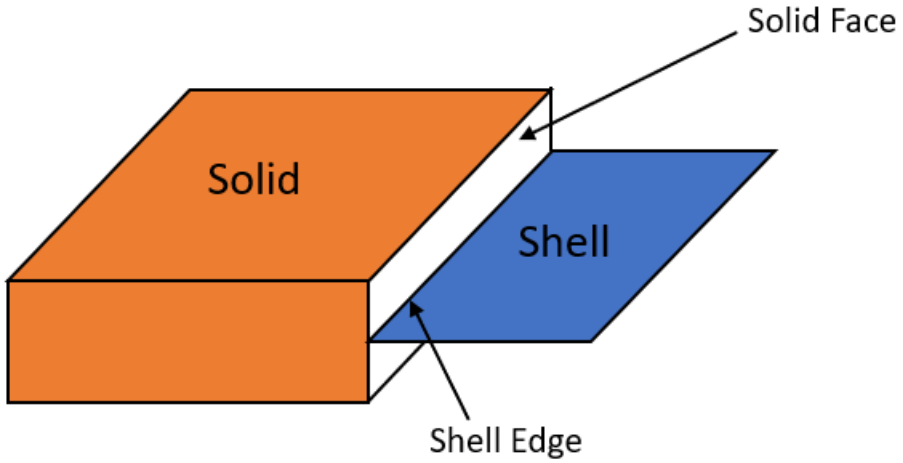



Figure 5.3: Shell regions (shown in blue) are connected to solid regions (shown in orange) at the shell-to-solid interface using a set of internal coupling constraints.

### 5.3.2 Design of Experiments: T-joint Structure

A T-Joint component is representative of many structural connections in engineering applications. Although using a multi-fidelity model for a T-joint might not result in significant savings for static analyses, studies [58] have shown the importance of multi-fidelity T-joint models for nonlinear analyses. The T-joint structure needs solid elements near the joints to capture local 3-D stresses, but for regions relatively far away from the joint, shell elements can be used. Thus, utilizing a combination of both approaches is ideal for achieving efficiency and accuracy. Hence, for this work, a T-joint static analysis was chosen as a demonstration structure to assess the potential of using decision trees rules-based fuzzy inference system for providing modeling recommendations.

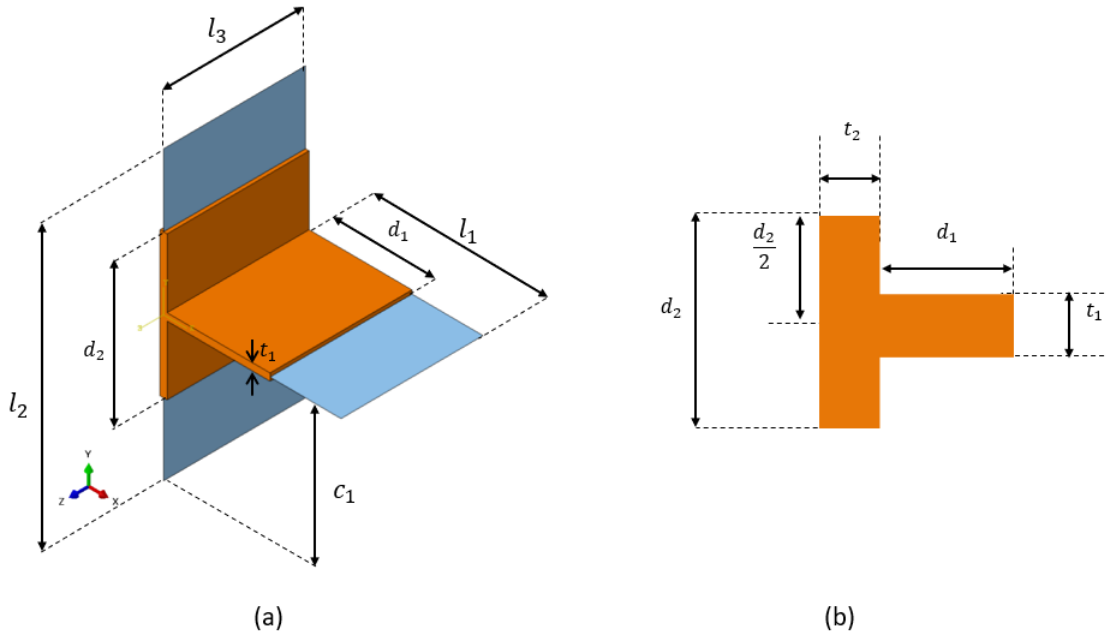



Figure 5.4: T-joint multi-fidelity model: (a) geometry parameterization; (b) side view of the solid section's geometric parameters.

### Geometry Parameters

The geometry of the T-joint is parameterized for comprehensive study as shown in Fig.5.4. The overbar denotes the non-dimensional quantities which are expressed in terms of the length of the horizontal and vertical members $l_1$ and $l_2$, respectively, as given below.

$$\bar{l}_1 = \frac{l_1}{l_2},\ \bar{l}_3 = \frac{l_3}{l_2},\ \bar{c}_1 = \frac{c_1}{l_2},\ \bar{t}_1 = \frac{t_1}{l_1},\ \bar{t}_2 = \frac{t_2}{l_2},\ \bar{d}_1 = \frac{d_1}{l_1},\ \bar{d}_2 = \frac{d_2}{l_2}$$

Here, $\bar{l}_1$, $\bar{l}_3$, $\bar{c}_1$, $\bar{t}_1$, and $\bar{t}_2$ are the non-dimensional geometric input parameters defining the structure, whereas, $\bar{d}_1$ and $\bar{d}_2$ are the relative 3-D region sizes near the T-corner. The following constraints have been applied on the parameters for the multi-fidelity geometry to be valid:

$$\bar{l}_1,\ \bar{l}_3,\ \bar{c}_1,\ \bar{t}_1,\ \bar{t}_2,\ \bar{d}_1,\ \bar{d}_2 > 0$$

$$0 < \bar{c}_1 < 1,\ 0 < \bar{d}_1 \leq 1$$

$$0 < \bar{d}_2\ \leq \min[1,\ 2\bar{c}_1,\ 2 - 2\bar{c}_1]$$

The bounds for the geometric input parameters are chosen such that no dimension becomes too large as compared to the other dimensions. For the mid-fidelity parameters, a practical value of 0.2 (20 percent) is considered to be minimum. Figure 5.5 shows some representative geometries created using the sampling process with the geometry upper and lower bounds given in Table 5.1.

Table 5.1: T-joint parameter upper and lower bounds.

| **Bounds** | $\bar{l}_1$ | $\bar{l}_3$ | $\bar{t}_1$ | $\bar{t}_2$ | $\bar{d}_1$ | $\bar{d}_2$ |
|---|---|---|---|---|---|---|
| **Lower Bound** | 0.50 | 0.50 | 0.02 | 0.05 | 0.2 | 0.2 |
| **Upper Bound** | 1.50 | 1.00 | 0.04 | 0.10 | 0.95 | 0.95 |

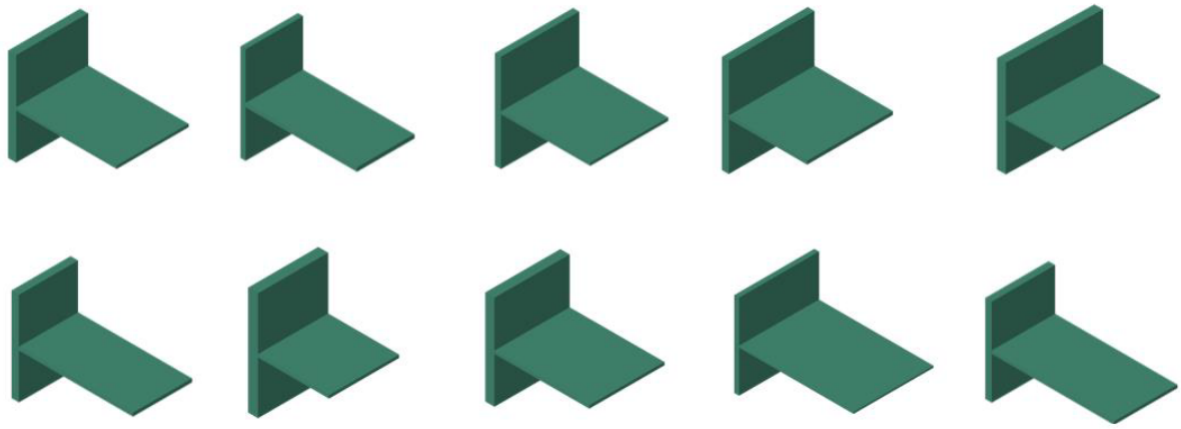


Figure 5.5: Representative geometries created using LHS.

### Load Combinations

The methodology utilized for generating Latin Hypercube Sampling (LHS) data points for multi-fidelity models with several geometric realizations and various load combinations is shown in Fig. 5.6. Three LHS samplings are used to select each aspect of the model. A unit-load approach is adopted where the baseline results for each type of load with a unit magnitude are obtained and then are linearly combined together to give the outcome for any arbitrary load combination as shown in Eq.5.1. Such a superposition technique can be applied for linear elastic material models with linear boundary conditions and small deformations. The individual displacement, stress and strain components add linearly, whereas combined results like max and min principal stresses, von Mises stress, and strain energy density do not. These values can be calculated after linear superposition has been performed on the corresponding stress and strain components.

$$f(ax + by) = af(x) + bf(y) \tag{5.1}$$

Two unit-load cases corresponding to edge loadings of the horizontal member have been

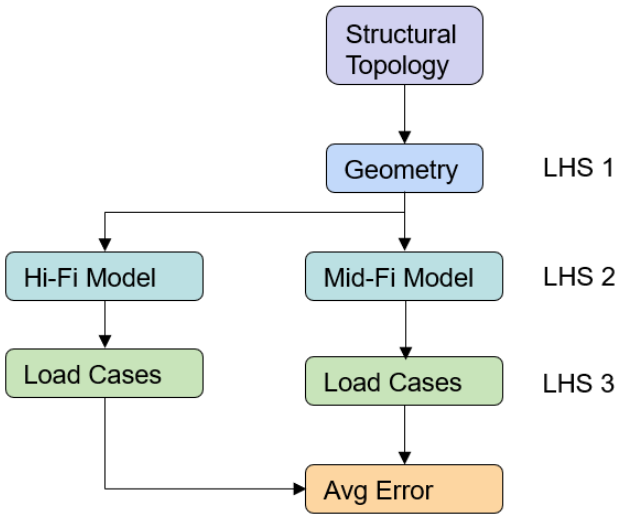



Figure 5.6: Multi-level DOE scheme employing LHS sampling of geometry, modeling fidelity, and loads.

utilized (Eq. 5.2). Because the modeling recommendations for linear analysis will only depend on the ratio of the various load components and not their individual magnitudes, the LHS selection is performed using polar or spherical coordinates to express the ratio in terms of the angular measures. Using such angular measures to define load ratios is illustrated in Fig. 5.7.

$$\bar{F}_{combo} = \frac{1}{w_{all}}(w_1 F_y + w_2 M_y) \tag{5.2}$$

Principal stresses and strains are calculated using the eigenvalues of the stress and strain tensors, respectively. Comparison between different fidelities was performed using the critical max principal stress over the entire structure. Finally, an overall load scaling by choosing $w_{all}$ such that it gives a unit maximum principal stress for the high fidelity model and comparisons are made between models of different fidelities. All model representations

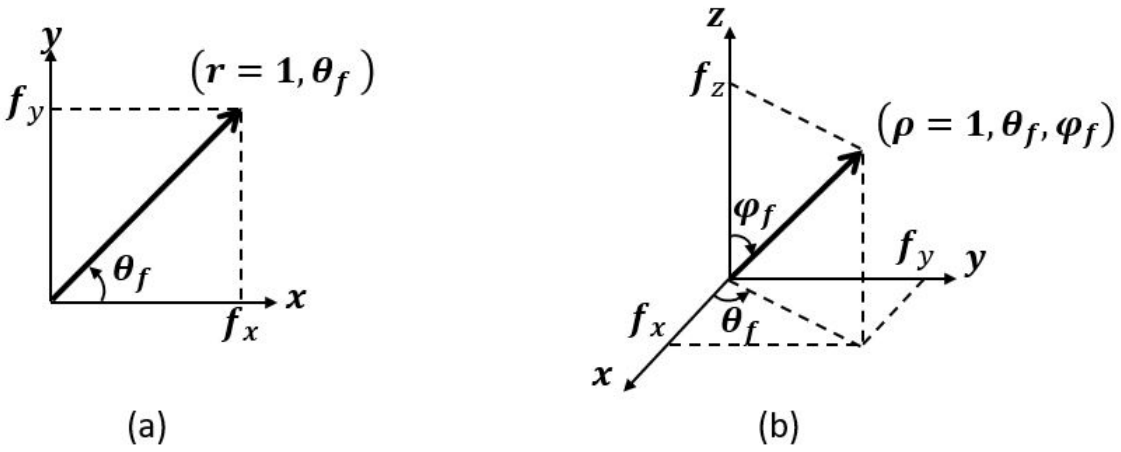



Figure 5.7: Load combinations using Latin Hyper Cube sampling: (a) design space for loads in 2 dimensional loading; (b) design space for 3 dimensional loading.

of a single geometry realization are analyzed under the same load combinations. For any given model, the results post-processing involved the following steps:

1. Extract results from unit load analyses
2. Scale and superimpose unit load results to give combined loading results
3. Extract quantities of interest from combined loading results
4. Determine differences in quantities of interest between hi-fidelity and multi-fidelity models

Figure 5.6 shows the multilevel design of experiments scheme for data generation for predicting modeling errors. The average error in the results for the range of load cases considered is calculated for each multi-fidelity realization of a particularly sized structure.

For each data point in the design space, discrete values of $\bar{d}_1$ and $\bar{d}_2$ are used to create multi-fidelity models using Abaqus/Standard [62]. The unit load results include a

transverse load ($F_y$) and a bending moment about the transverse-axis ($M_y$), as shown in Figure 5.8. The unit load cases are run as independent linear steps in a single Abaqus/Standard simulation for a given geometry realization. Loads are applied to the right horizontal surface using a reference point definition. Although, the use of a reference point makes the load application region slightly rigid, it ensures appropriate equivalent load application on the solid surface in the full fidelity model and shell edge in the multi-fidelity model. The T-joint structure is studied under a fixed boundary condition as shown in Fig. 5.8. Note that as shown in Fig. 5.9, results from regions near the applied boundary conditions (5% of $l_3$) and near the free edges (10% of $d_1$), are not considered to avoid errors due to boundary effects.

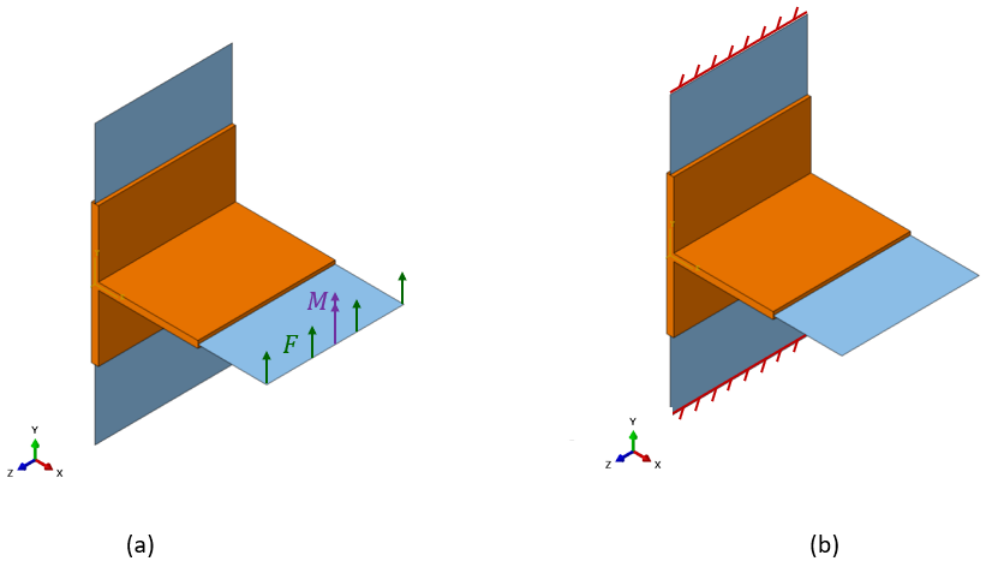



Figure 5.8: T-joint multi-fidelity model: (a) loads applied at the free end; (b) boundary conditions applied at the ends.

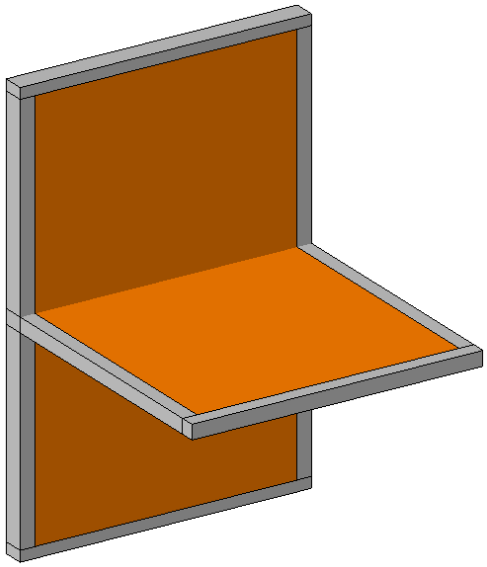


Figure 5.9: Results region of interest shown in orange and discarded regions in gray color.

### Mesh Parameters

The solid models are meshed using 3-D continuum elements C3D8 (full integration) while the multi-fidelity models utilize a combination of C3D8 full integration and S4R shell elements, where s4R is a 4-node shell element with reduced integration and hourglass control. The solid meshes are created by fixing a minimum of 3 elements through the thickness and increasing it depending on the thickness of the horizontal and vertical members. In order to make sure that there is minimal effect of noise due to the mesh size used, a very refined global mesh size was employed by using the $h$-convergence scheme for all the geometries. Figure 5.10 shows a typical mesh for a sample geometry. The mesh resolution used in the full solid and the mixed-fidelity models was kept the same for consistent comparison. A database is developed for several T-Joint configurations and their corresponding mixed-fidelity model combinations. A total dataset of 3000 data points with 30 geometric parameter sets, each with set 100 multi-fidelity model representations, is constructed. The average error for each mid-fi model is calculated using 100 load

combinations. The entire data generation process including parametric geometry creation, model definitions and results extraction is performed using Abaqus/CAE and Abaqus/Standard scripting in Python.

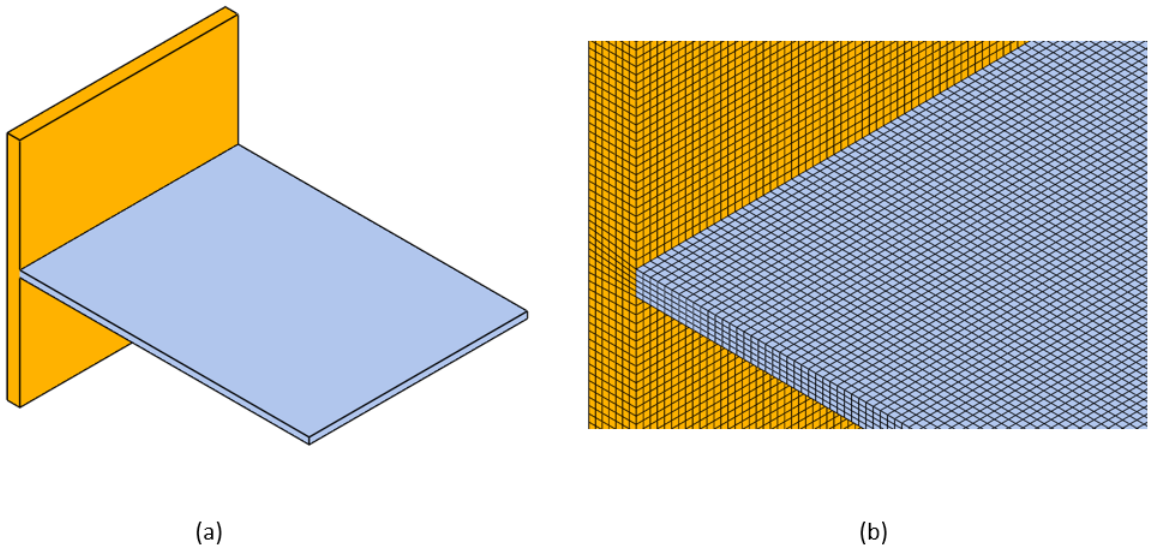



Figure 5.10: Typical refined mesh for a sample geometry: (a) color-coded T-joint members; (b) zoomed-in view of the mesh.

Figure 5.11 shows the maximum principal stress contours for a sample full solid and a multi-fidelity model for a particular load combination of $F_y$ and $M_y$ with a ratio of 1:2 ($\theta = 63.4°$). As can be observed from Fig. 5.11, the maximum principal stress occurs near the T-joint transition and occurs nearly at the same location for the full solid and multi-fidelity model. Hence, an error metric based on the maximum principal stress in the structure is utilized to compare the models with different fidelities for various load combinations and then averaged to provide an average error for a particular mixed fidelity topology.

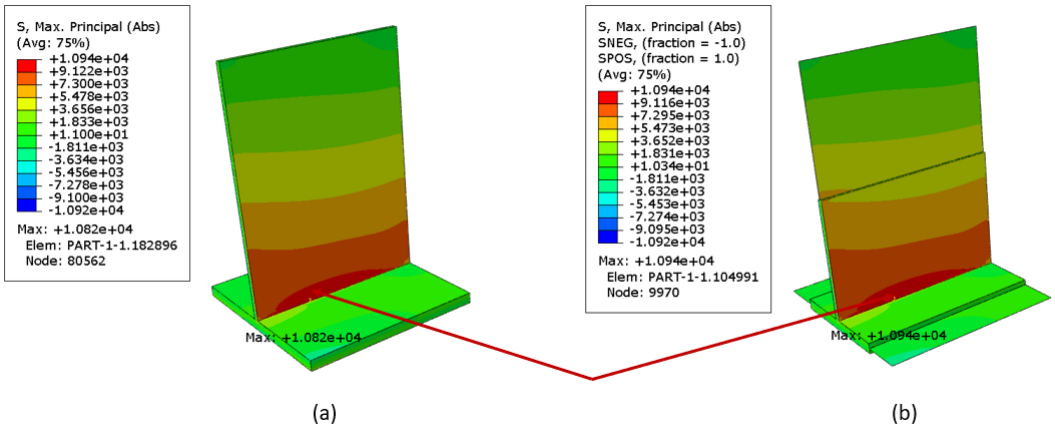



Figure 5.11: Maximum principal stress contours for a combined $F_y$ and $M_y$ loading (ratio: 1:2) (a) full solid, (b) multi-fidelity model.

### 5.3.3 Data Pre-processing

During the data generation process, data points with $\bar{d}_1$ and $\bar{d}_2$ less than 0.2 are generated. However, it is decided that for practical purposes, it is appropriate to consider a relative minimum 3-D region of 20 percent. Hence, these data points are eliminated from training dataset utilized for the machine learning model development. A comparison of the full solid and the several multi-fidelity T-joint models results in error values below 6 percent. Although the errors were not extremely critical, the error trends still highlight the effect of increasing 3-D topology in the critical regions to reduce errors, and thus are still useful in assessing and demonstrating the proposed approach. Of course, higher errors are expected in more complex problems.

Figures 5.12, 5.13, and 5.14 show the effect of $\bar{d}_1$ and $\bar{d}_2$ on the error values of maximum principal stress for different sample geometries. It is observed that for some geometries, the effect of an increase in $\bar{d}_2$ shows a monotonic decrease in error levels, while for other geometries, the effect of $\bar{d}_1$ is dominant, or their interaction is important. A bi-linear

regression was performed to confirm the interaction effect of multi-fidelity parameters on the error values and the $p$-values of this interaction effect were observed to be significant for some geometries, while for others, that of individual $\bar{d}_1$ and $\bar{d}_2$ were dominant. However, it was also observed that for most geometries, increase in $\bar{d}_2$ leads to consistent downward trend in error values.

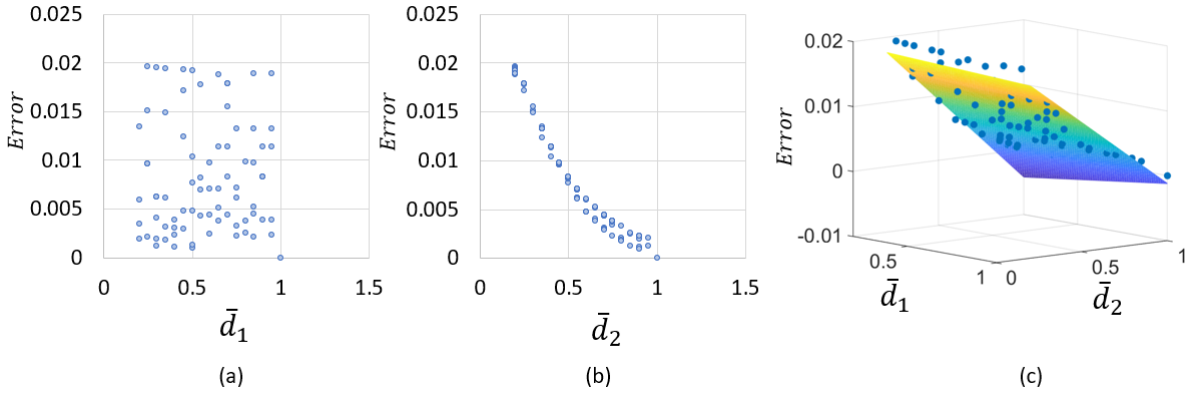



Figure 5.12: Effect of mixed-fidelity parameters on error values for sample geometry 1: (a) effect of $\bar{d}_1$; (b) effect of $\bar{d}_2$; (c) interaction effect of $\bar{d}_1$ and $\bar{d}_2$.

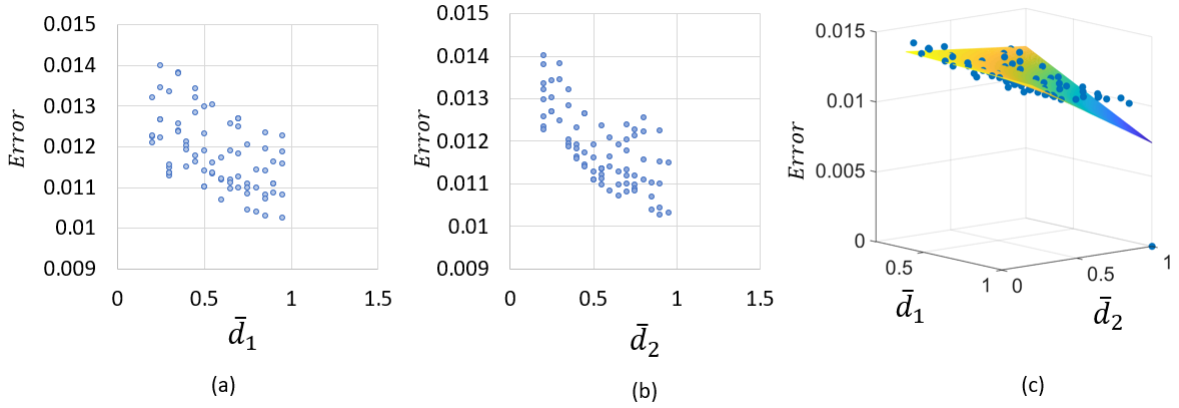



Figure 5.13: Effect of mixed-fidelity parameters on error values for sample geometry 2: (a) effect of $\bar{d}_1$; (b) effect of $\bar{d}_2$; (c) interaction effect of $\bar{d}_1$ and $\bar{d}_2$.

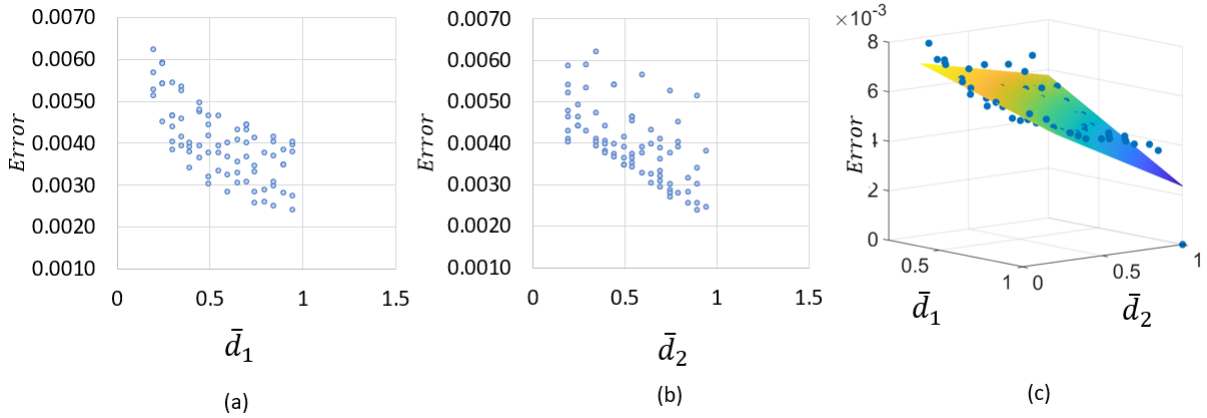



Figure 5.14: Effect of mixed-fidelity parameters on error values for sample geometry 3: (a) effect of $\bar{d}_1$; (b) effect of $\bar{d}_2$; (c) (c) interaction effect of $\bar{d}_1$ and $\bar{d}_2$.

### 5.3.4 Conversion of Numerical Values into Categories using Fuzzy C-means clustering

The error values from the data collection are converted into five linguistic categories using fuzzy c-means clustering. This clustering technique starts with an initial guess for the number of clusters intended and then iteratively updates the cluster centers based on an optimization function. This function is based on the distance of a data point from its cluster center weighted by its membership grade (incorporating fuzziness). Figure 5.15 shows the five linguistic labels and the scales at which the labels were segregated. The numerical error ranges for the linguistic labels are presented in Table 5.2. Note that data imbalance is observed in the dataset as a relatively high number of data points are categorized into 0 to 0.7% (class A) and 0.7 to 1.4% (class B) error categories. As such, in addition to ensuring high accuracy of the prediction model, the correct classification of individual classes also needs to be verified. This can be done by computing the confusion matrix of the classifier which is demonstrated in Section III.B.

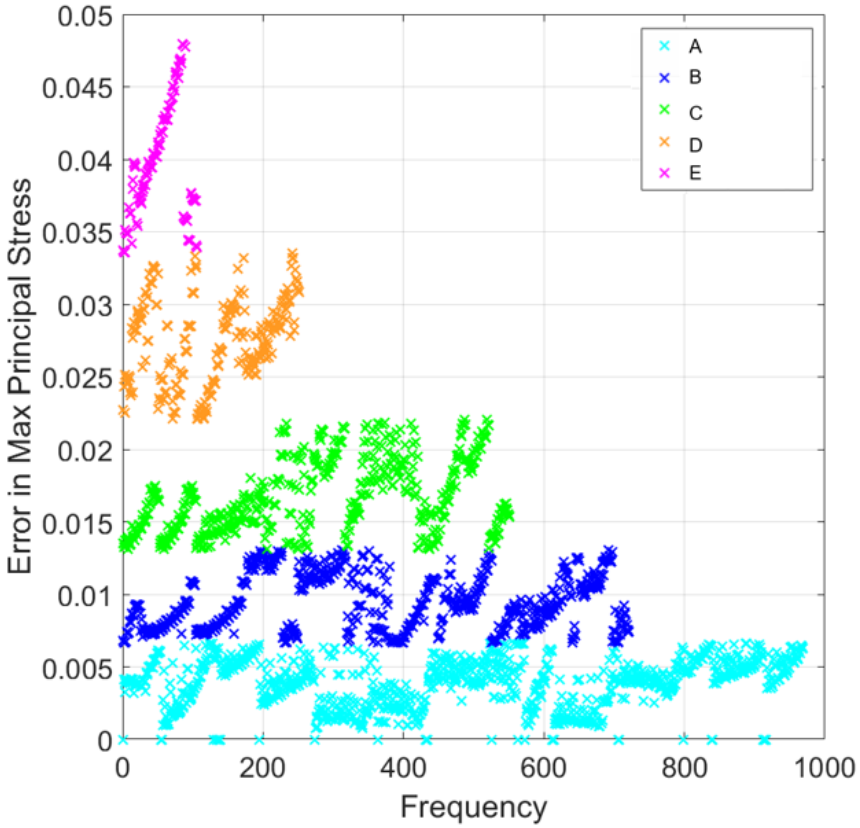



Figure 5.15: Categorizing error values using fuzzy C-means clustering (FCM).

Table 5.2: Linguistic error terms in numerical error ranges.

| Linguistic Term | Error Range |
|---|---|
| A | $0 - 0.7\%$ |
| B | $0.7 - 1.4\%$ |
| C | $1.4 - 2.2\%$ |
| D | $2.2 - 3.4\%$ |
| E | $3.4 - 5\%$ |

## 5.4 Decision Tree Rules based Fuzzy Logic Classifier

### 5.4.1 Fuzzy Logic

Fuzzy logic was first introduced by Zadeh [47] in 1973. Since then it has become very popular in dealing with uncertain and imprecise data which is commonly encountered in real-life engineering applications. The applications range from industrial processes to consumer products to decision support systems. A fuzzy logic based inference system makes decisions based on a set of rules. Development of fuzzy inference system has three main features: (a) input/output membership functions and (b) rule-base and (c) fuzzy logic operations. A fuzzy membership function is a mathematical function which defines an element's degree of membership in a set. An element can belong to multiple sets with varying degrees of membership as compared to a Boolean set. Thus, the standard logical set operations like A AND B, A OR B , NOT B etc. are replaced with min $(\tilde{A}, \tilde{B})$, $\max(\tilde{A}, \tilde{B})$, and $1 - \tilde{A}$ where $\tilde{A}$ and $\tilde{B}$ are fuzzy sets [56]. Fuzzy logic converts the fuzzy sets into linguistic variables such that the rule base can be represented in the form of 'IF..THEN..' rules.

Real world data does not have crisp defined values (*e.g.*, statistical variation of material properties) and hence obtaining crisp error assessment may not be critical. Instead, an approximate error estimate of a multi-fidelity model could be more helpful. Hence, this problem can be modeled appropriately using fuzzy logic. In this work, the Fuzzy Inference System (FIS) toolbox as available in MATLAB based on Mamdani type is utilized. A rule-based is developed using decision trees. Such an approach makes use of the benefits offered by decision trees as well as fuzzy logic. The goal of this framework is not to make perfect guesses of the error values or precise topology distributions, but to make predictions with the correct trends. Hence, in this regard, using the fuzzy inference system

is appropriate for developing an expert system for modeling predictions. The following section provides a brief introduction to decision trees.

### 5.4.2 Decision Trees

A simple example of a decision tree (DT) for a design space with two features $x_1$ and $x_2$, is presented in Fig. 5.16(a). This design space needs to be classified into red and blue labels. Figure 5.16(b) demonstrates the color-coded design space as predicted by DT. The structure of the tree is shown in Fig.5.17. A decision tree has a three types of nodes: (a) the root node at which an attribute split is made, (b) the internal node which are derived from the splits made at the top nodes, (c) leaf nodes which perform the final prediction. Since, the input parameters, $x_1$ and $x_2$ are numerical, the tree performs splits at the numerical values of the input attributes, namely at $x_1 = 3.35$ and then at $x_2 = 3.2$. It can be observed that the tree makes predictions by dividing the design space and using if-block conditions from the root node to a terminal node. Also, it can be noted that in this example, the design space is slightly mixed and that the decision tree predicts the most dominant class in a region and ignores the less frequent values. The choice of the best criterion to split on is made by using the concept of impurity measures, which assesses the goodness of a split. There are a number of splitting rules available including the Gini index, Class Probability, Twoing rule and Entropy. The standard Classification and Regression Tree (CART) algorithm uses the Gini index and is sometimes biased in a way that it could make multiple splits on an input variable with continuous values as compared to an input variable with categories,i.e., having discrete values. This could mask more important predictors and miss significant interactions between inputs and response. MATLAB [53] recommends using a 'Interaction-Curvature' based split predictor selection technique when a combination of continuous valued and categorical valued input features

exist. This has the capability to reduce the bias and enhance the tree interpretation. This is because it performs Chi-square tests of independence between each predictor and the response. If the $p$-value of this test is less than a threshold, the null hypothesis is rejected and the feature is selected to be split on. Thus, the most important feature interacting with the response variable will be split first, thereby also providing the level of predictor importance related to the problem. Moreover, it also considers the interaction between a pair of predictors and response. In this work, the 'Interaction-Curvature' based split criterion is used, as there are continuous-valued and categorical features to be used and tree interpretation is important. The main features that make decision trees an attractive approach are that they can model complex relationships without *a priori* assumptions and can handle numerical, as well as categorical attributes. Along with this, DTs offer easy interpretation and visualization that mimics human decision making process.

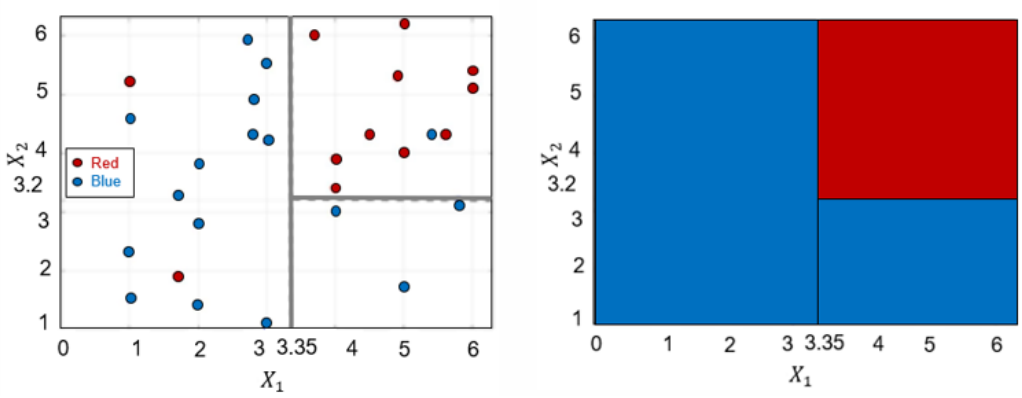



Figure 5.16: A simple classifier for numerical input parameters:(a) design space; (b) visualization of the design space.

### 5.4.3 Decision Tree for predicting T-joint Mixed Modeling Error

Figure 5.18 shows the crisp decision tree developed for the prediction of error in the mixed FE model. The splits made by decision tree also demonstrate the features that contribute

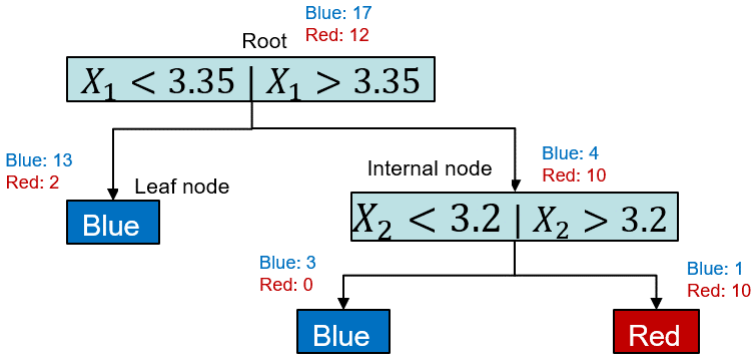



Figure 5.17: Trained decision tree from the simple design space.

the most for classification (based on the appearance of the features from top to bottom). This property of decision tree is used to make feature selection and all the input variables are used for the initial tree development. It can be observed that $\bar{l}_1$ is the most dominant feature that is used by the decision tree for grouping different geometries. It is important, when training a decision tree, to keep it simple to make the most of its interpretability. Also, as the tree depth increases, the tree tends to over-fit the data. Hence, a study of the variation in the cross-validation error with an increase in the number of splits is performed, and a tree depth of around 20 splits is considered sufficient. The confusion matrix of the tree is shown in Fig. 5.19. This matrix allows the visualization of the performance of an algorithm, where the rows labels correspond to the actual class label and the column labels denote the predictions made by the trained model and the numerical values give the DT classification. As can be observed in Fig. 5.19, the diagonal values show significantly higher values indicating the adequate performance of the classifier. A modified accuracy of the classifier can be calculated using the confusion matrix to include the number of correct predictions to the total number of predictions for each class label.

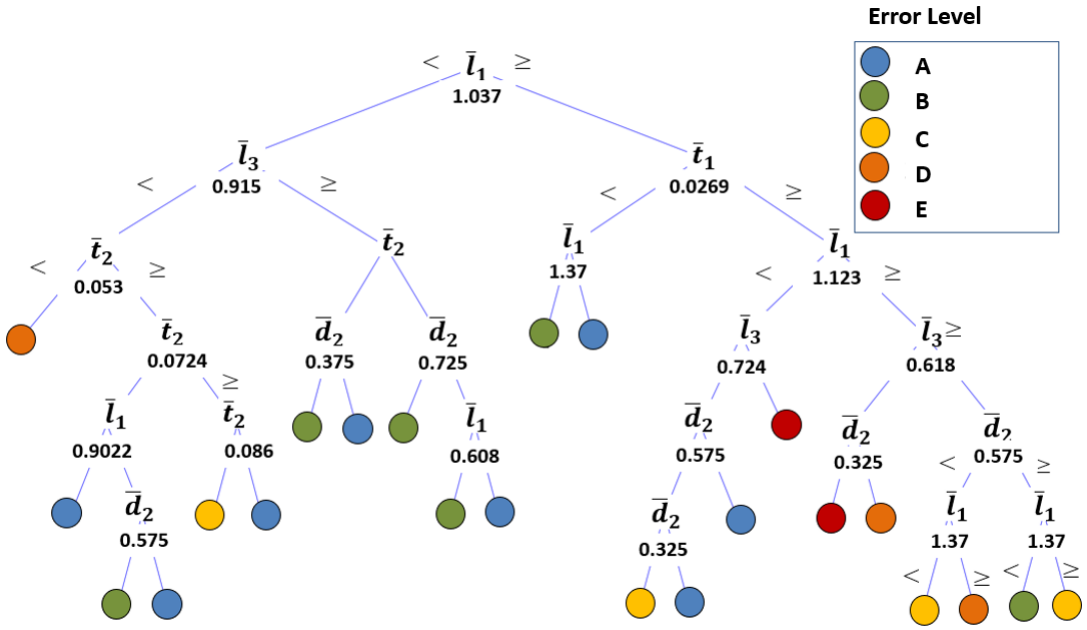



Figure 5.18: Decision Tree predicting error levels arising from multi-fidelity T-joint models.

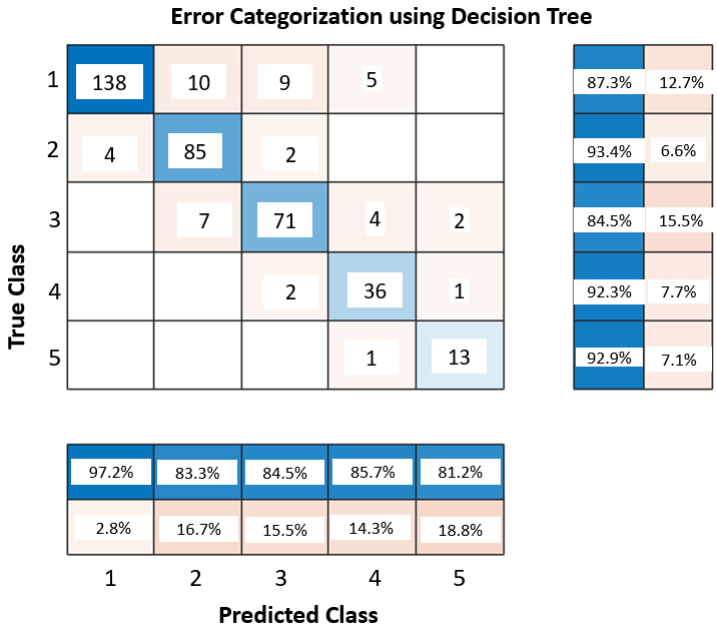



Figure 5.19: Confusion matrix of the decision tree predicting error levels (DT-I).

### 5.4.4 Decision Tree for predicting optimal modeling choices

#### Generation of Optimization Data

The first decision tree predicting errors arising from the variation of multi-fidelity parameters could be used by an analyst to make modeling decisions that lead to sufficient accuracy. However, in order to consider both accuracy and efficiency, it is important to make nearly optimal choices. Thus, rather than utilizing the first decision tree to predict error, it would be beneficial to utilize this decision tree in an optimization loop to generate a new dataset that involves the optimal choices for different geometry and error combinations. This dataset can then be employed to develop another decision tree (DT-II) that predicts optimal modeling recommendations. This procedure is also known as 'predict-then-optimize' approach which is widely used for combining optimization using machine learning approaches. The objective function for this problem can be stated as maximization of the 2-D topology in the multi-fidelity model such that sufficient accuracy is achieved based on the acceptable error level. This can be written as a minimization problem of the form,

$$f(x) = \frac{1}{\Omega_2} \tag{5.3}$$

The 2-D region can be defined as the volume of the 2-D part in the T-joint structure as shown here 5.4.

$$\Omega_2 = (l_1 - d_1)t_1 l_3 + (l_2 - d_2)t_2 l_3 \tag{5.4}$$

The error levels can be added as constraints to the objective function. Each constraint can

be written as

$$g_i = DT_i/U_i^{max} - 1 \leq 0 \tag{5.5}$$

where $DT_i$ is the error from the multi-fidelity model as predicted by DT-I and $U_i$ is the accepted error level. This constrained optimization problem can then be converted into a unconstrained one using the penalty approach.

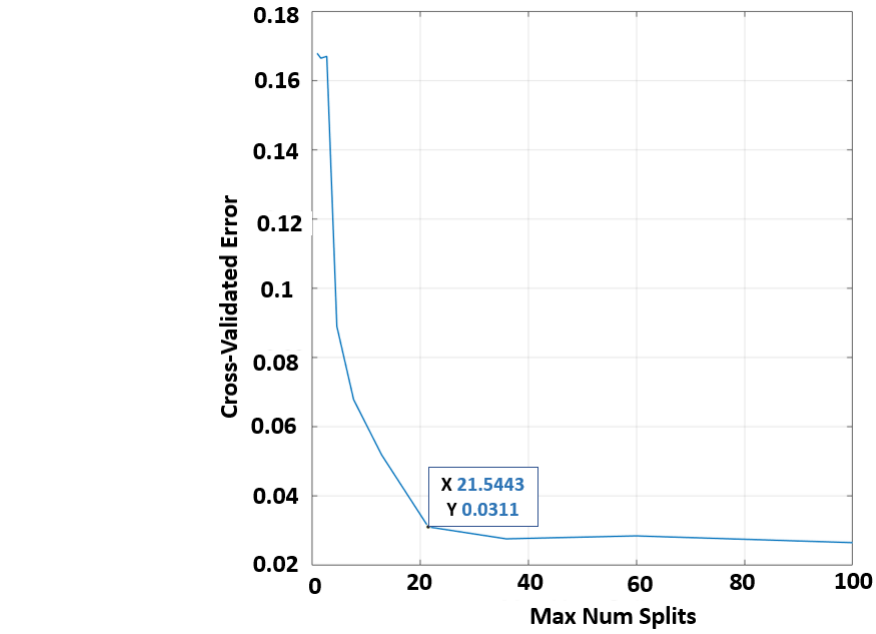
$$h(x) = f(x) + K\left(\sum \max(g_i, 0)^2\right) \tag{5.6}$$

where $K$ is $1,000$, a very large number.

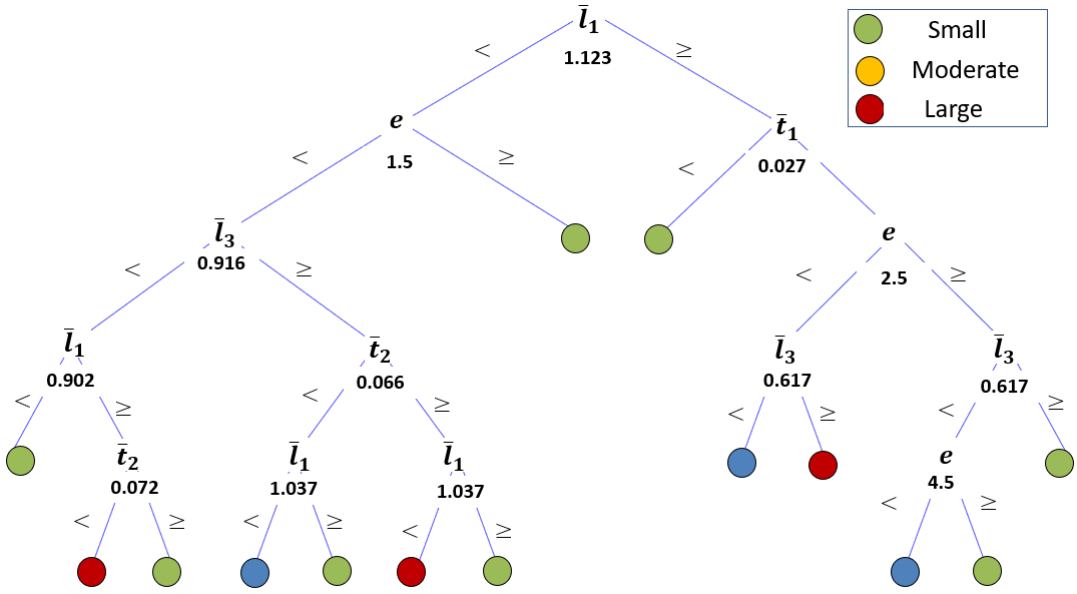
Using this optimization formulation, a new dataset with $20,000$ points is generated which includes optimal discrete values of $\bar{d_1}$ and $\bar{d_2}$, for different error levels and geometric configurations. In particular, five different error levels ranging from below A,B,C,D and E were considered. For each acceptable error level, the optimal modeling recommendations ($\bar{d_1}$) and ($\bar{d_2}$) are selected using grid-based optimization for a particular structural configuration. A pre-processing step of the new data revealed that a minimal $\bar{d_1}$ value of 20% is always sufficient. Hence, only the effect of increasing $d_2$ on accuracy can be considered significant. The optimal values of $d_2$ are found to be varying from 0.2 to 0.7 and are clustered into three categories for simplicity and easier interpretability. The optimal values ranging from 0 to 0.3 represent 'Small' category, values from 0.3 to 0.5, 'Moderate' category, and values greater than 0.5, 'Large' category.

### Decision Tree Predicting Optimal Multi-fidelity Levels

Utilizing the optimization dataset developed, a second decision tree (DT-II) is trained that can predict the optimal values of $\bar{d_2}$. The decision tree is trained with a maximum of 22




(a)




(b)

Figure 5.20: DT-II parameter selection; (a) maximum number of 22 splits gives the best cross-validation error; (b) decision Tree predicting optimal modeling choices for developing multi-fidelity T-joint models.

node splits based on the trend of the cross-validation error performance as shown in Fig. 5.20. The obtained decision tree is further pruned by 3 levels (Fig. 5.21) to avoid over-fitting and with a motivation to obtain easier rules that make sense with respect to the physics of the structure. The levels pruned involved node splits of the input variable $\bar{l}_1$ at very close values (0.902, 1.037) which signify over-fitting in the design space and can be merged under a single split value of the input variable $\bar{l}_1$ at the root node. Moreover, pruning these levels do not significantly affect the performance of the model based on the confusion matrices. In fact, pruning these levels makes the interpretation based on $l_1$ easier as this input variable is now only split into two membership functions, 'Small' and 'Large'. The confusion matrix of the pruned tree given in Fig. 5.22 shows that it can be reliably used to make predictions for different class labels. The goal of this framework is not to make perfect guesses of the error values or precise topology distributions, but to make predictions with the correct trends. As compared to a trained Artificial Neural Network (ANN) which generally is black-box model, decision trees combined with fuzzy logic provides reliable predictions with interpretability that increases the confidence in the model predictions.

### 5.4.5 Fuzzy Inference System

Once the decision tree for optimal modeling recommendations is trained, the crisp decision rules are utilized to develop membership functions for the inputs and output of the fuzzy inference system. Figure 5.23 shows the process of a Mamdani-type fuzzy inference system. Real valued geometry inputs and mid-fi modeling parameters are fuzzified and fed to the inference system. This system then utilizes the stored rule-base to infer a fuzzified output which is then defuzzified to give out a crisp output. This work uses the fuzzy logic application as available in MATLAB®[53].

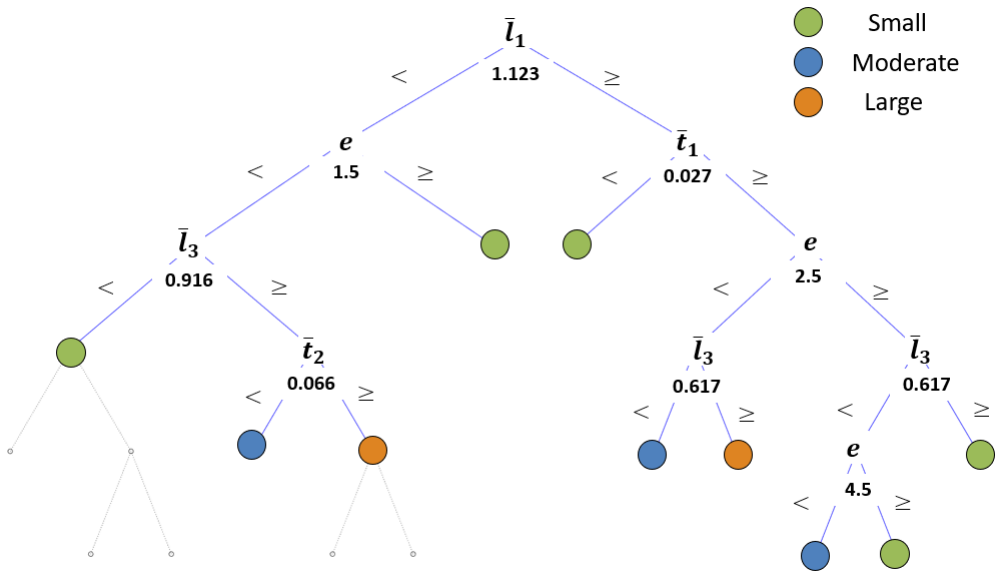



Figure 5.21: Further pruning of the decision tree by 3 levels to avoid over-fitting and achieve easier interpretability.

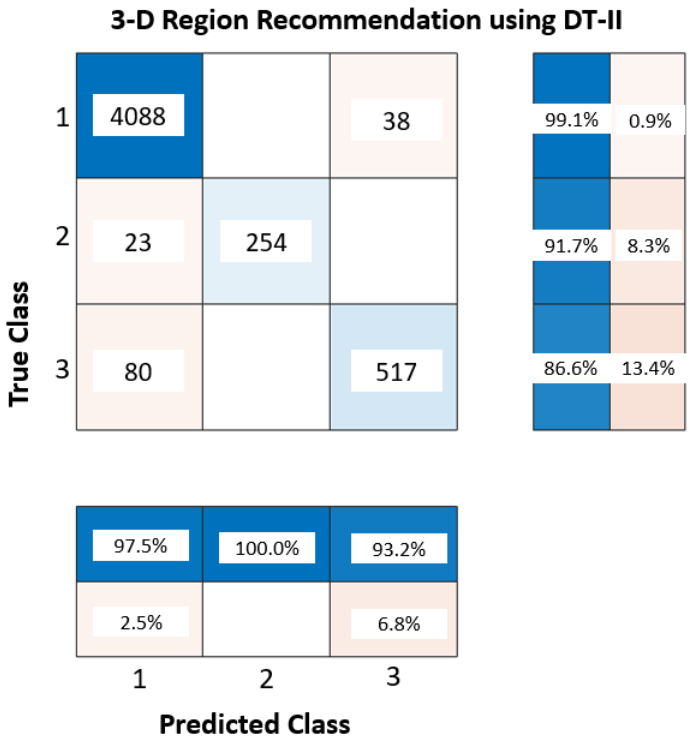



Figure 5.22: Confusion matrix for DT-II.

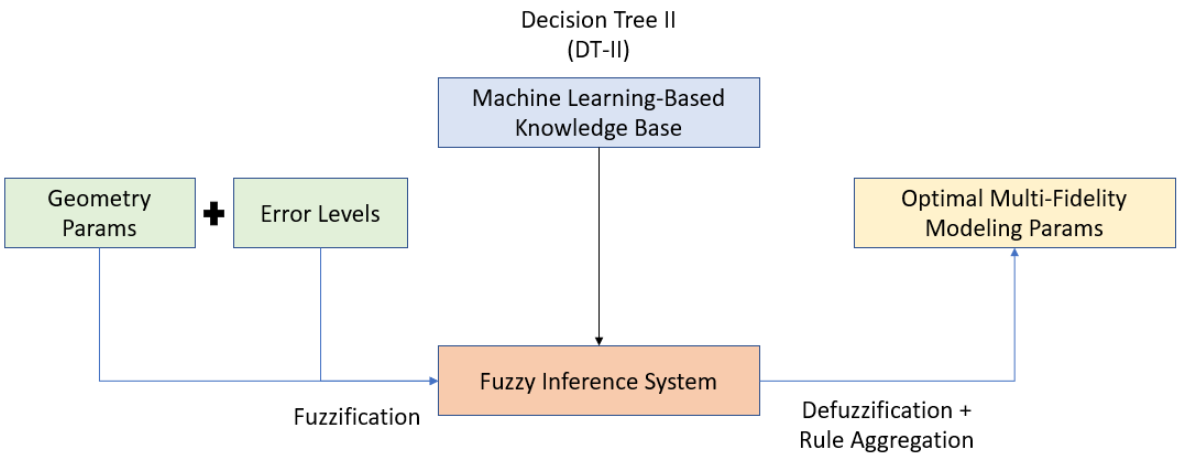



Figure 5.23: Mamdani type fuzzy logic based inference using data-driven rules from the decision tree.

### Input and Output Membership functions

The membership functions for the fuzzy system variables are defined according to the node split thresholds on the branches of the decision tree. The input variables, $\bar{l}_2$, $\bar{l}_3$, $\bar{t}_1$, $\bar{t}_2$ and $\bar{d}_2$ are utilized to develop the antecedent part of the fuzzy inference system. The membership functions are selected to be of the form of trapezoidal functions because of these functions' simplicity and thus speed. From the structure of the decision tree, it can be observed that $\bar{l}_1$ is branched once at the root node in the tree structure. Hence, for this variable, two membership functions are defined. Similarly, membership functions for the other variables are defined as shown in Figs. 5.24, 5.25, 5.26, 5.27, 5.28 and 5.29.

## 5.4.6 Rule-base from Traversing Decision Trees

Once the input and output parameters are fuzzified using the membership functions and the fuzzy boundaries based on the information from the decision tree, the rules can be written in the form of IF-THEN statements. For simplicity, all the rules are given equal

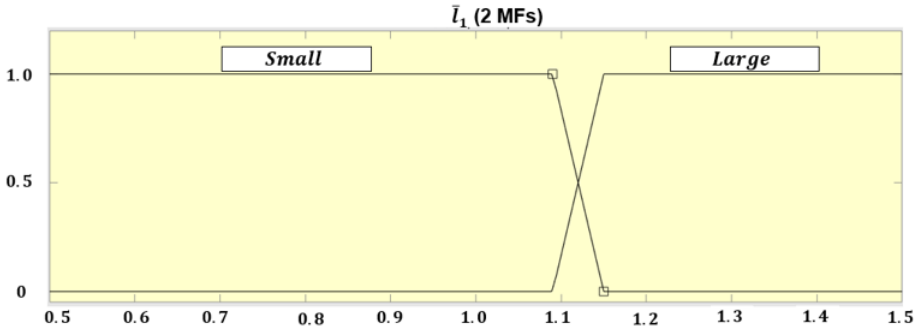



Figure 5.24: Fuzzy membership function for the input variable $\bar{l}_1$.

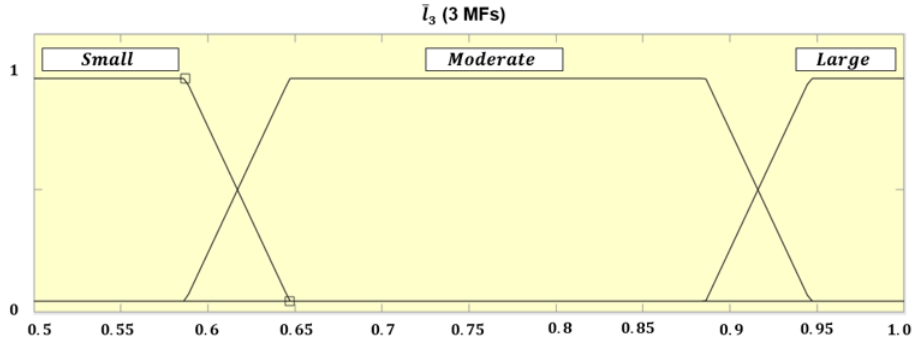



Figure 5.25: Fuzzy membership function for the input variable $\bar{l}_3$.

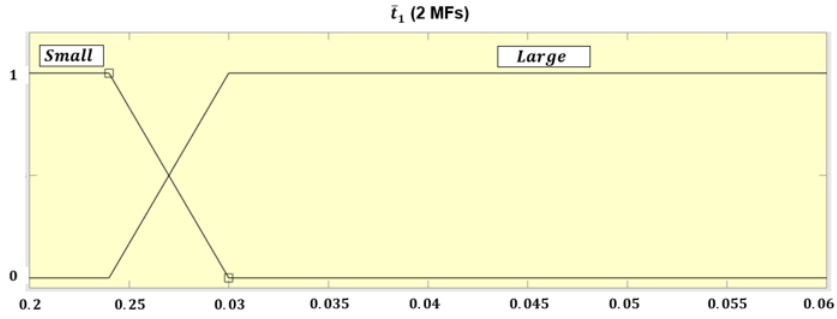



Figure 5.26: Fuzzy membership function for the input variable $\bar{t}_1$.

weights in the fuzzy inference system, and are evaluated in parallel in the system. Rules obtained from a decision tree can be verified using domain knowledge as a sanity check.

Rule 1: If $\bar{l}_1$ is (Small) & Error is (NOT A), then $\bar{d}_2$ is (Small).

Rule 2: If $\bar{l}_1$ is (Small) & $\bar{l}_3$ is (NOT Large) & Error is (A), then $\bar{d}_2$ is (Small).

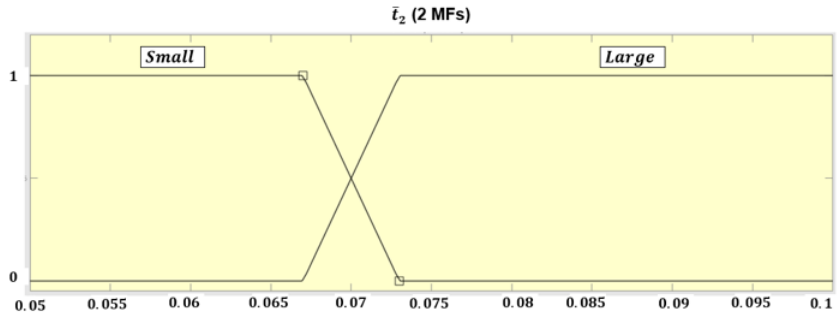



Figure 5.27: Fuzzy membership function for the input variable $\bar{t}_2$.

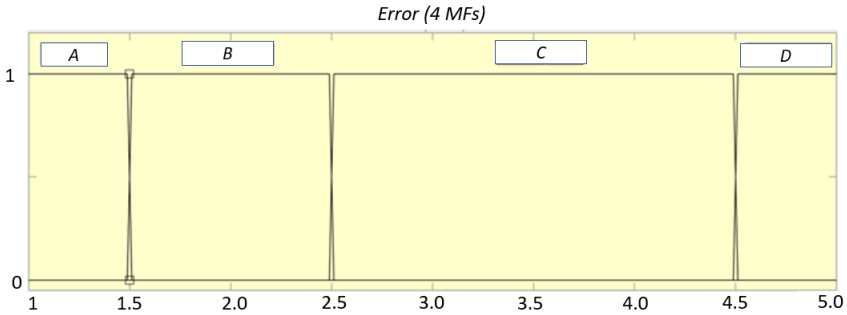



Figure 5.28: Fuzzy membership function for the input variable error.

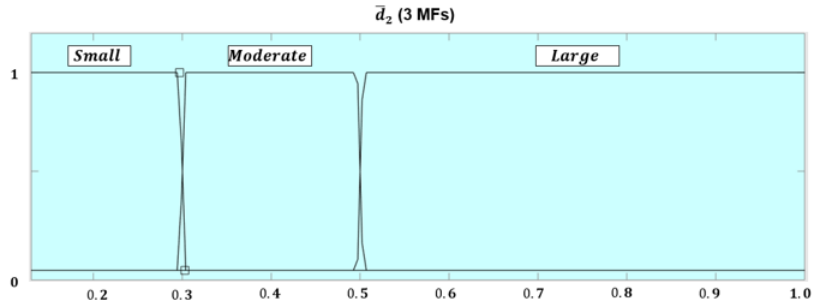



Figure 5.29: Fuzzy membership function for the output variable $\bar{d}_2$.

Rule 3: If $\bar{l}_1$ is (Small) & $\bar{l}_3$ is (Large) & $\bar{t}_2$ is (Small) & Error is (A), then $\bar{d}_2$ is (Moderate).

Rule 4: If $\bar{l}_1$ is (Small) & $\bar{l}_3$ is (Large) & $\bar{t}_2$ is (Large) & Error is (A), then $\bar{d}_2$ is (Large).

Rule 5: If $\bar{l}_1$ is (Large) & $\bar{t}_1$ is (Small), then $\bar{d}_2$ is (Small).

Rule 6: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (Small) & $\bar{t}_1$ is (Large) and Error is (A), then $\bar{d}_2$ is (Moderate).

Rule 7: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (Small) & $\bar{t}_1$ is (Large) and Error is (B), then $\bar{d}_2$ is

(Moderate).

Rule 8:If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (NOT Small) & $\bar{t}_1$ is (Large) and Error is (A), then $\bar{d}_2$ is (Large).

Rule 9: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (NOT Small) & $\bar{t}_1$ is (Large) and Error is (B), then $\bar{d}_2$ is (Large).

Rule 10: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (NOT Small) & $\bar{t}_1$ is (Large) and Error is (C), then $\bar{d}_2$ is (Small).

Rule 11: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (NOT Small) & $\bar{t}_1$ is (Large) and Error is (C), then $\bar{d}_2$ is (Small).

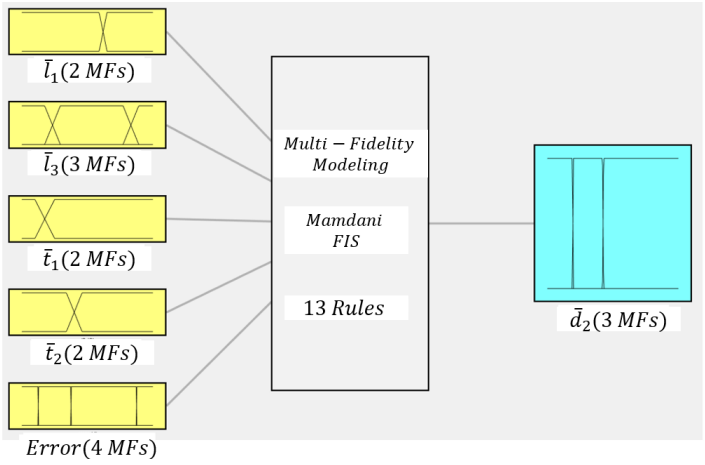
Rule 12: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (Small) & $\bar{t}_1$ is (Large) and Error is (D), then $\bar{d}_2$ is (Moderate).

Rule 13: If $\bar{l}_1$ is (Large) & $\bar{l}_3$ is (Small) & $\bar{t}_1$ is (Large) and Error is (C), then $\bar{d}_2$ is (Small).

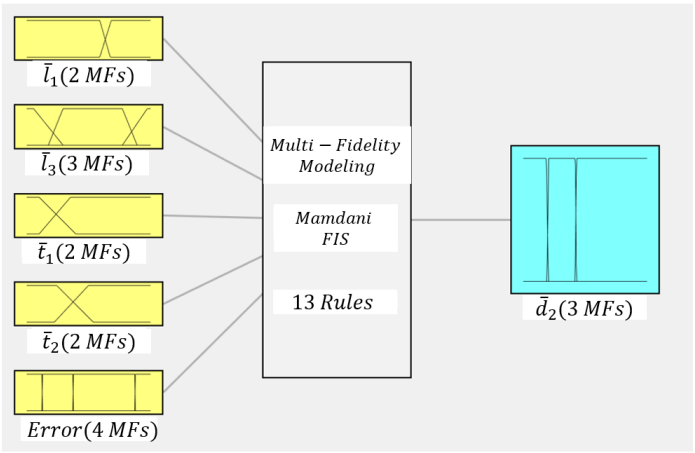
It can be observed that the decision tree divides the dataset into two major clusters; (a) geometries having longer horizontal plate relative to the vertical plate; (b) geometries that have shorter horizontal plate relative to the vertical plate. This makes sense as the loads are applied on the free end of the horizontal plate. Moreover, because transverse loads and a moments about the y-axis in varying combinations are applied on the free end of the horizontal plate, the overall bending moment from $F_y$ increases with a larger $\bar{l}_1$, and hence the effect of the thickness of the horizontal member, $\bar{t}_1$, becomes important in this case. On the other hand, for a shorter $\bar{l}_1$, the overall effect of the moment applied about the Y-axis dominates, which tries to twist the vertical member and hence the thickness, $\bar{t}_2$ is important for this case. This effect is also captured by the decision tree and the rules derived from the right-hand side of the tree (larger $\bar{l}_1$) have node splits at $\bar{t}_1$, whereas the rules on the left-hand side have node splits at $\bar{t}_2$. Additionally, one of the rules indicates that for a very long horizontal plate with small thickness, the small 3-D region in the

horizontal and vertical members is sufficient to get small to very small errors. This is in agreement with the general terms of using shell elements in long and thin plates. The effect of width, $\bar{l}_3$, interacting with other variables is also captured. It can be seen that for lower errors, a longer and wider vertical plate needs slightly larger 3-D region in the vertical member, whereas a moderate 3-D region is sufficient for shorter widths. For shorter $\bar{l}_1$ (length of horizontal plate), a wider ($\bar{l}_3$), and thicker ($\bar{t}_2$) vertical plate needs relatively higher 3-D region in the vertical member. Also, it can be observed that with the increase in the 3-D region in the vertical members, the error is reduced. Note that $\bar{d}_1$ has been kept to a minimum value of 20 percent. The extracted rule base was then fed into the MATLAB based interactive fuzzy inference system to make optimal modeling recommendations for a mixed-fidelity T-joint model. The inference system predicts continuous $\bar{d}_2$ values within the recommendation categories. The range of output membership function is defined from 0 to 1. The fuzzy membership values from 0 to 0.3 represent 'Small' category, values from 0.3 to 0.5 'Moderate', and values greater than 0.5, 'Large' category. A benefit of including fuzziness is that the FIS employs all the rules that could affect the inference with varying degrees at once to result in a final decision. The analyst can utilized the final defuzzified value in an approximate fashion and also make sense of how much this value lies within a category.

### 5.4.7 Tuning of Fuzzy Inference System's Membership Functions using Pattern Search Algorithm




(a)




(b)

Figure 5.30: Fuzzy inference system architecture: (a) before tuning input & output membership functions; (b) after tuning input & output membership functions.

Once the basic fuzzy inference system is defined using the membership functions for inputs and outputs, it is optimized for improved performance. The original membership functions are tuned using pattern search algorithm, a local optimization method, for faster

convergence of the parameter values using MATLAB. This optimization technique searches a set of points around the current point by utilizing an adaptive mesh size to search directions where the value of the objective function is lower than the current point. The parameters of the trapezoidal membership functions of the input and outputs were constrained to be within reasonable ranges such that the cross-over point between adjacent membership functions as derived from the decision tree stays more or less the same. In other words, the variation of the membership functions is constrained to be within a small range. A custom cost function was defined for the tuning process, which post-processed the crisp defuzzified values from the fuzzy inference system to categories. A root mean squared value (RMSE) was calculated for the difference in the predicted categories and true categories. Figure 5.30 shows the architecture of the original FIS (Fig. 5.30(a)) and the tuned FIS (Fig. 5.30(b)). It can be observed that the membership functions of the inputs has been slightly changed with the cost function decreasing from 0.6 to 0.51. The process of generating an output fuzzy set with the rule base with tuned membership functions is shown in Fig. 5.31. The first five columns indicate rules and the activated membership functions for the geometry inputs and the accepted error limit. The last column indicates the decision leading to the prediction for optimal d2bar. Based on the geometry and error inputs, the corresponding membership functions, and hence the consequent rules are activated. The vertical red lines show the location of the respective input values in the membership functions. All the relevant rules are initiated based on the membership function location result in a fuzzy output which then gets defuzzified by using the 'centroid' method. This method takes an aggregate of all the output membership functions in varying degrees and outputs the centroid value of the total area of all the membership functions.

The time taken by the FIS to make predictions is only a few seconds and can be utilized readily by the analysts to make optimal modeling choices in real time during the FE

modeling process.

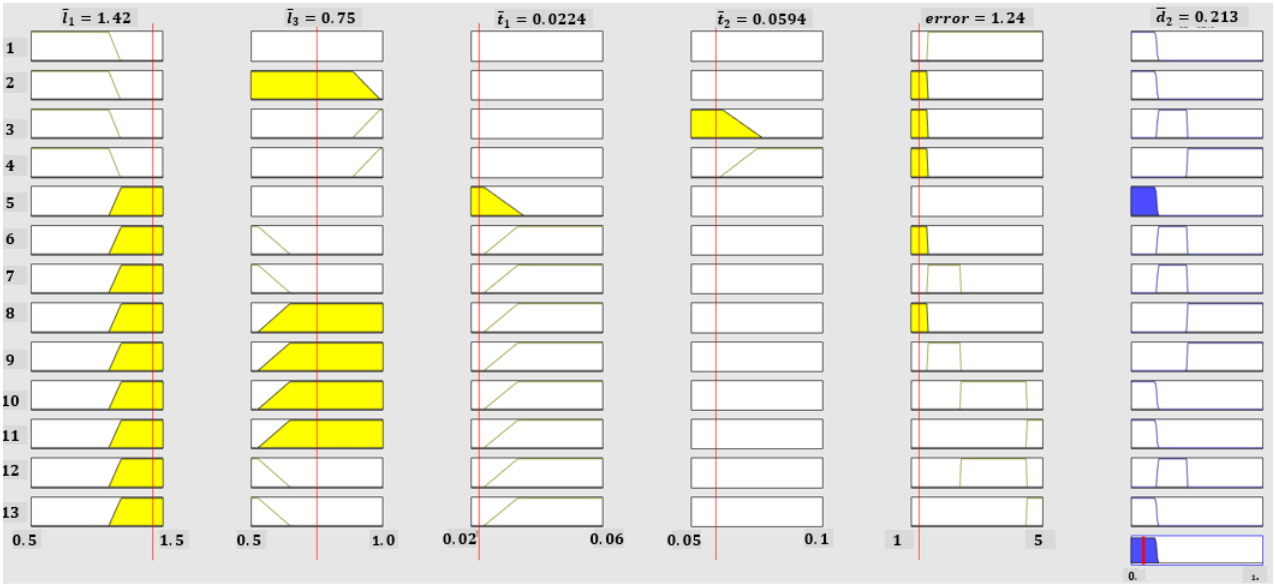



Figure 5.31: Interactive rule viewer (available in MATLAB) to view the inference process based on the input values.

## 5.5 Conclusions

The paper contributes a novel application of decision trees (DT) and fuzzy inference system (FIS) in the field of expert systems for providing optimal modeling recommendations related to multi-fidelity finite element models. A predict-then-optimize approach was utilized where a decision tree (DT-I) is first trained to predict the errors resulting from the implementation of various multi-fidelity parameters. The error predictions are then utilized within an optimization loop to generate a dataset that contains optimal multi-fidelity choices for various geometry and error configurations. A second decision tree (DT-II) is trained on the generated optimization dataset to recommend modeling choices given a geometry configuration and acceptable error level. The extracted fuzzy rules from the

second decision tree (DT-II) are fed to a fuzzy inference system. The results indicate that such a system has the ability to extract useful and interpretable rules for decision making which follow a human-like thinking process. Additionally, fuzzification using a FIS provides robustness to the system. This, in turn, has the potential for significant savings, as the time required for making recommendations is very low and is suitable for online implementation. Moreover, it has the potential to remove the subjective nature of such modeling decisions which leads to differences in critical modeling decisions.

A limitation of this approach is that not all data-driven rules can be useful. This can be attributed to multiple factors such as the presence of noise in the data and algorithm used for training the model. However, the interpretability of these rules makes it possible for the analyst to make a choice regarding whether to accept or discard a particular rule in the final expert system. On the other hand, it is also possible to add manual rules to the FIS and combine extracted rules from multiple sources to improve the capabilities of FIS. This, thus, provides control as well as guidance to the analyst.

Another limitation is that this approach may not be appropriate for cases where designs with high variations are expected, such that cases in which a fuzzy logic based model would not make sense. An example of such high-variance situations could be determination of spacecraft trajectories which needs to be very precise. Another limitation is that with increasing variables and the number of adjectives used for their description, the number of fuzzy rules can become numerous and difficult to handle and may require computer implementation. Hence, the number of features and their sets should be reduced as much as possible. Alternatively, it is possible to obtain reduction of redundant fuzzy rules by using optimization algorithms [63] if the learned rules are large. This could be done by tuning the FIS for rule parameters along with membership functions. In this work, the number of rules after pruning the decision tree were found to be reasonable and also made

sense from physical aspects. Another approach would be to utilize a connected small tree of fuzzy inference systems to break down a complex problem into small parts. This approach can be further improved with an error metric that provides a combined global-local error information from the interest regions and global regions between different models. Additionally, the effect of $h$-refinement was not considered in this study. Future studies could consider the coupled effect of element topology and mesh refinement on the response from mixed fidelity models for an accurate error estimate.

The presented approach has been demonstrated using the multi-fidelity finite element model development of a common engineering component, T-joint structure. This approach could be further generalized for developing interpretable expert systems for other complex problems to reduce the variability in computational modeling decisions present in such problems.

## Acknowledgments

The authors are grateful to the US NAVY SBIR for sponsoring this research under contract number: N00178-17-C-1310 and Topic Number: N171-059 with Nathan Leong as the Technical Point of Contact (TPOC). A special thanks to Junhyeon Seo for his contributions towards the post-processing code development. The authors would also like to thank to Dr. Karanpreet Singh and Dr. Mohammed Jrad for their contributions to the early stages of this work.

# Chapter 6

# Conclusions

The need for robust automation of mixed-dimensional models along with optimal modeling decisions that reduce the dependency on the expertise and experience of the analyst is notably increasing for achieving faster and optimal designs. This dissertation made novel contributions in accelerating the FE modeling process by developing techniques for eliminating bottlenecks and subjectivity in the pre-processing and post-processing of mixed-dimensional FE models of complex structures.

A framework for automated development of mixed-dimensional models with variable elemental topology from full fidelity and structural fidelity models using parameterization, cutting tools and scripting is detailed. Insights gained from the automation of a complex mixed-dimensional models are discussed. Techniques like using explicit shell-solid coupling to assemble shell and solid parts of the assembly at the boundary are proposed for additional computational savings. A blueprint for strategic volume decomposition for achieving robust hexahedral meshes in complicated assemblies is proposed.

An application submarine tank structure with multi-fidelity element topology was generated and analyzed to demonstrate the capabilities of the framework presented. It was shown that a multi-fidelity model can provide results with sufficient accuracy with substantial cost savings as compared to full fidelity and structural fidelity models. These models show that the integrated automation of the entire FE modeling and analysis has the potential to provide faster and accurate solutions with significant computational savings. It can be concluded that once scripted, the generation of mixed models can be automated and utilized by industry for their design or optimization phases for quick and accurate designs and analyses. Moreover, domain-based error metrics for robust comparisons between models of different fidelities is proposed. These domains are selected based on the critical regions away from potential singularity-causing areas. The maximum principal stresses and strain energy totals in the local and global regions proved to be

useful metrics for quantitative comparison between models with different fidelities.

A parametric study was performed using the automation scripts to study the volumetric component-wise and global strain energy errors. The domains are selected based on critical components in the model, namely, chock, tank-top and tank-walls; and are defined as element sets in the FE models. The creation of these element sets is scripted for automated extraction and comparison of post-processing results. It was observed that the global strain energy error metric tends towards decreasing values as the multi-fidelity levels are increased. It is anticipated that the domain-wise stress metrics and component-based strain error metrics will be more useful for other complicated problems that show larger errors between full-fidelity and multi-fidelity models. A further study on a complex model that shows larger differences between strain and stress values for different levels of fidelities is needed confirm the benefit of using such domain-wise error metrics.

Additionally, this dissertation contributes a novel application of interpretable machine learning (decision trees) and fuzzy inference system (FIS) in the field of expert systems for providing optimal modeling recommendations for developing multi-fidelity FE models. The automation scripts developed for multi-fidelity models are utilized to generate datasets for training this software tool. The results indicate that such a system has the ability to extract useful and interpretable rules for decision making which follow a human-like thinking process. Moreover, fuzzification using a FIS provides robustness to small parameter uncertainties. The transparency and the fuzzy form of the extracted rules produces interpretable rules that could be selected/discarded based on prior knowledge or physics-based sanity checks by the analyst. This feature provides confidence to the analyst in terms of utilizing the predictions of the data-driven tool along with excellent computational efficiency. This, in turn, has the potential for significant time, effort and cost savings.

Parametric studies on the effect of multi-fidelity parameters on the modeling accuracy confirmed that a multi-fidelity model can predict structural response with sufficient accuracy along with substantial cost savings as compared to full fidelity and structural fidelity models. The excellent accuracy of mixed-fidelity models with minimum practical solid regions provides confidence to the industry about the benefit of utilizing coupled shell-solid models for computational efficiency. Moreover, it was confirmed that a minimum practical solid region is sufficient for static analyses of different T-joint and stiffened pressure vessels with tank configurations.

Due to the excellent agreement between the full-fidelity models and multi-fidelity models with minimum practical solid regions, making further error reductions in the already insignificant errors does not seem useful for the presented application geometries (submarine tank and T-joint structure) under static loading conditions. Nonetheless, the proposed expert system framework was assessed using the T-joint structure which exhibited noteworthy capability in terms of useful rule-base extraction, fuzzy consideration and computational efficiency along with and benefits for implementation to more complex problems.

Chapter 7 identifies some potential directions for future research as well as more complex application problems that could be more fitting for the proposed expert system implementation.

# Chapter 7

# Directions for Future Work

The proposed techniques and frameworks can be improved to further boost the speed of finite element modeling and analysis process. This chapter discusses some limitations of the proposed techniques in the dissertation and provides future research directions.

## 7.1 Automated Identification of Strategic Volume Partitions

Even though the proposed blueprint of the volume partitioning procedure proved to be a robust methodology for making strategic partitions to a wide range of complex tank assembly structural configurations, the partitions to be made were pre-identified manually. The needed cells and partitioning edges were parameterized and scripted in order to achieve robust hex-meshable models. This blueprint of the volume partitioning procedure can be further developed into robust algorithm for decomposing complicated assemblies into hex meshable regions. This would mean automatically detecting the assembly interface lines (trace features) and decomposing the volume based on the assembly topology and curvature without needing any human intervention. Potential techniques could include use of image-based machine learning techniques. Additionally, this blueprint can be augmented with existing volume decomposition methods to develop a robust methodology. This would greatly accelerate the FE modeling process and expand the design exploration space.

## 7.2 Improvements in the Expert System Framework

The fuzzy knowledge rules' interpretability makes it possible for the analyst to make a choice regarding whether to accept or discard a particular rule in the final inference

system. The architecture of the fuzzy inference system facilitates manual addition of experience based rules along with data-driven rules. Thus, knowledge from multiple sources can be combined to develop a hybrid knowledge which can further improve the performance of the decision-making tool.

One limitation of the proposed fuzzy inference system is that with increasing variables and the corresponding number of membership functions, the number of fuzzy rules can become enormous and difficult to handle. Hence, the number of features and their sets should be reduced as much as possible. This could be done by utilizing a connected small tree of fuzzy inference systems to break down a complex problem into small parts. Additionally, a parametric study is needed to understand the coupled effect of mesh size as well as mesh elemental topology on the accuracy of multi-fidelity models. The expert system can then be augmented with the optimal recommendations for both, mesh size and mesh type with optimal combination. Finally, the fuzzy inference system can be expanded to include a feedback loop for making decisions regarding elemental and mesh topology in an adaptive sense for time-dependent and non-linear structural problems.

## 7.3 Improved Sampling Methods for Reduced Data-sets

One important drawback of implementation of any machine learning is the need for a large dataset. For engineering problems, especially, one with non-linearity or composite laminates, the cost of developing large number of data points for a reliable trade study can quickly become exponentially high. In these situations, it would be beneficial to utilize active learning (also known as 'query learning' techniques) to achieve high accuracy using

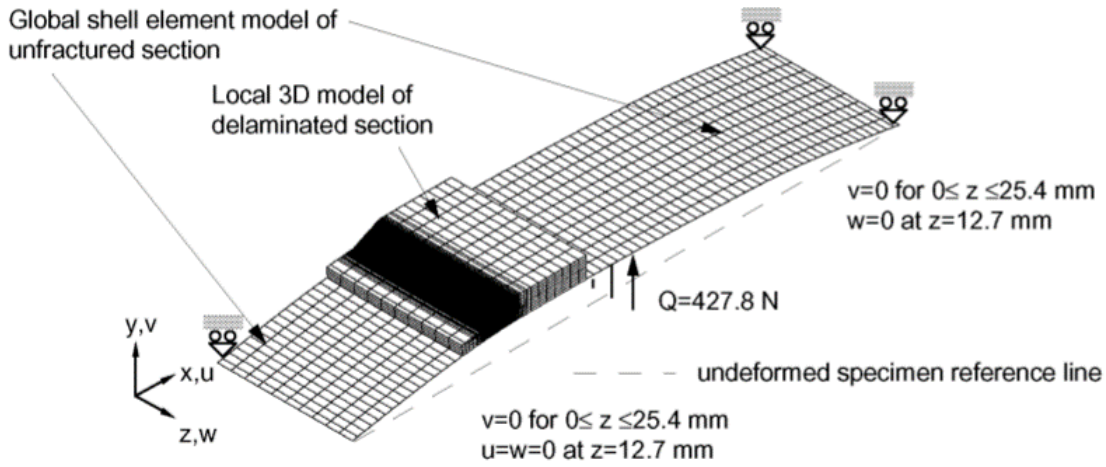



Figure 7.1: Example of a composite debond specimen modeled as a multi-fidelity model for computational savings [13].

as few data points as possible, thereby minimizing the cost of acquiring training data. The idea is to train the machine learning model using a small dataset. Later, the model is allowed to interactively query selected data points which seem to have high uncertainty. This way, only data points which improve the performance are queried and greatly reduce the number of redundant data points.

## 7.4 Potential Applications of Proposed Techniques for Substantial Benefits

Applications for developing automated multi-fidelity models of other complex stiffened pressure vessel problems including, but not limited to, optimization of aircraft fuselage, stiffened launch vehicles with cryogenic propellant tank structures, multi-stage launch vehicles with thick stiffeners, and optimization of ring-stiffened ocean structures with composite laminates. Additionally, the expert system framework can be applied to more complicated problems where the effect of 3-D stresses is pronounced. For instance,

composite components in aerospace structures are typically made of curved or flat panels with co-cured or adhesively-bonded frames and stiffeners. Such structures are traditionally analyzed using plate and shell elements. However, shell elements do not account for the correct section stiffness, kinematics, and strain energy release rates in the vicinity of delaminations and debonds. This can be modeled with sufficient accuracy using solid elements. However, several solid elements are needed through the thickness to model the individual composite plies, which can make the FE model computationally expensive [9]. An example of a shell/3D multi-fidelity model developed to analyze such a problem is shown in Fig. 7.1.

Another example would be a thick composite laminate with a bolted hole joint where 3-D stresses are prominent depending on the loading, laminate sequence, bolt pre-load and boundary conditions. Figure 7.2 shows an example of a full-fidelity model of a composite bolted joint with a countersunk hole under bearing failure to due contact with the bolt.

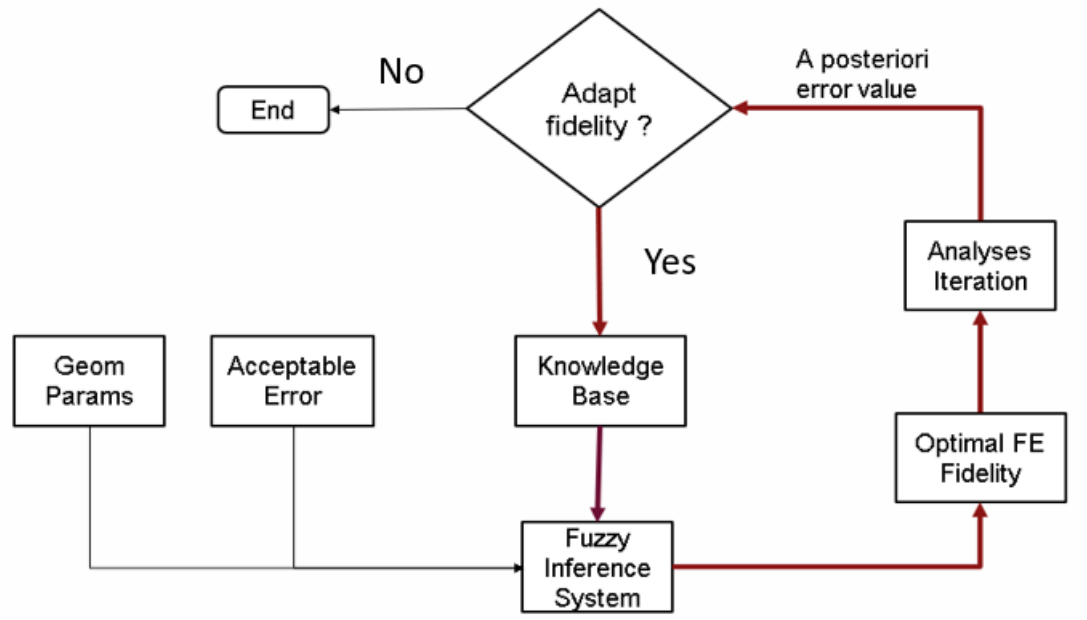



Figure 7.4: Schematic representing utilization of the proposed expert system in an adaptive loop for time-dependent problems.

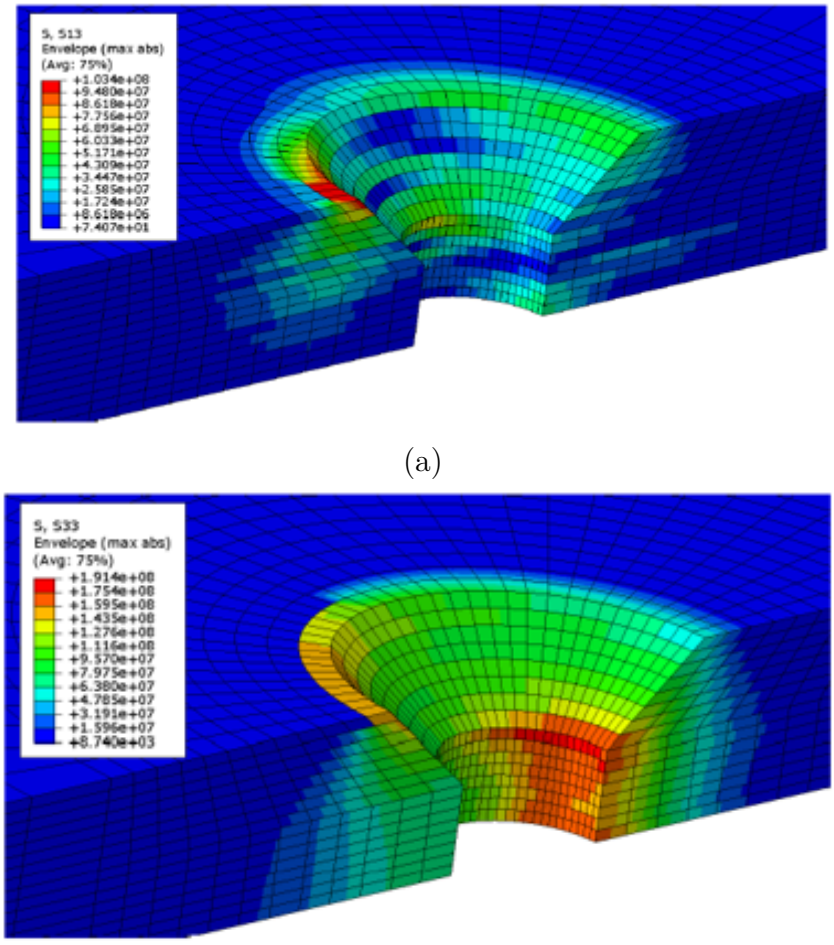



Figure 7.2: : Composite bolted joint under a single shear test; (a) high transverse shear stress near the hole; (b) high transverse normal stresses near the hole [64].

Aforementioned numerical investigation work by Salerno *et al.* [15] to assess the accuracy and efficiency of thin composite laminates under a uniform distributed loads found the coupled models (Fig. 7.3) results within 10−15% of the reference solutions with a significant speed up. The authors stated that coupled shell-solid models can offer excellent reduction in computational costs provided the 3-D regions are sufficiently large to cover the possible delaminations. Since the locations and size of delamination depend on the layup, loading and other geometry parameters, the authors suggest the use of a mesh updating

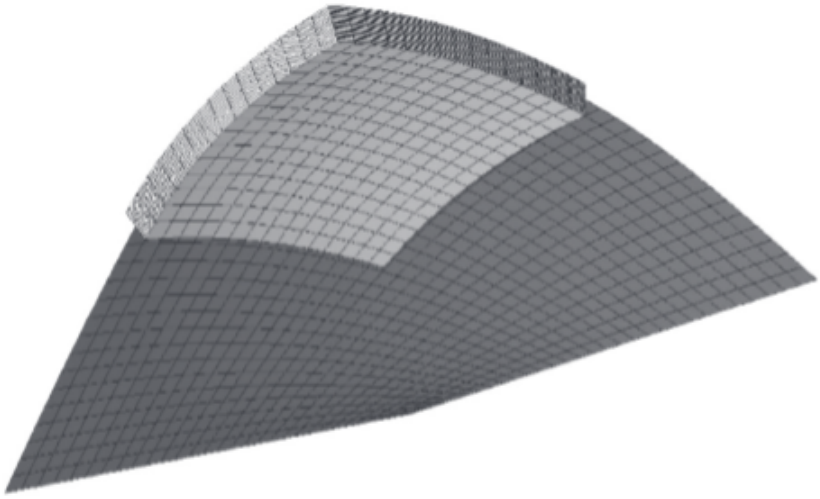


Figure 7.3: Example of a bent composite plate modeled using a coupled shell-solid approach [15].

procedure that switches from a 2-D shell topology to 3-D solid every time a delamination stress threshold is reached. This is where the proposed expert system framework would be especially useful to predict the optimal 3-D regions as an open-loop or within an adaptive mesh updating framework as shown in Fig. 7.4. Additionally, highly nonlinear finite element models, particularly with composites can greatly benefit with the utilization of this framework.

Finally, this framework could also be adopted by other domain-discretization numerical techniques like computational fluid dynamics analysts for deciding the different meshing and boundary layer parameters for complex flows. Thus, the proposed framework of data-driven expert system has broad applications in highly complex computational modeling applications.

# Bibliography


[1] JM Voth and GH Sturtevant. Digital engineering: Expanding the advantage. *Journal of Marine Engineering & Technology*, pages 1–9, 2022.

[2] Wikipedia contributors. Columbia-class submarine — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/Columbia-class_submarine. [Online; accessed 13-January-2022].

[3] Phil Zimmerman, Tracee Gilbert, and Frank Salvatore. Digital engineering transformation across the department of defense. *The Journal of Defense Modeling and Simulation*, 16(4):325–338, 2019.

[4] RK Rajan, RV Nambiar, KL Lawrence, and TJ Lawley. A description of an expert system for finite element modeling and analysis. *Computing Systems in Engineering*, 4 (2-3):169–177, 1993.

[5] First a350 xwb fuselage takes shape. URL https://ambitiontofly.wordpress.com/2011/10/02/first-a350-xwb-fuselage-takes-shape/.

[6] H Hibbitt, B Karlsson, and P Sorensen. Abaqus user subroutine reference manual version 6.10. *Dassault Systemes Simulia Corp.: USA*, 2011.

[7] BP Naganarayana and Gangan Prathap. Expert systems and finite element structural analysis—a review. *Sadhana*, 17(2):275–298, 1992.

[8] Liang Sun, Christopher M Tierney, Cecil G Armstrong, and Trevor T Robinson.

Decomposing complex thin-walled cad models for hexahedral-dominant meshing. *Computer-Aided Design*, 103:118–131, 2018.

[9] Ronald Krueger and Pierre J Minguet. Analysis of composite skin–stiffener debond specimens using a shell/3d modeling technique. *Composite Structures*, 81(1):41–59, 2007.

[10] Kar Wei Shim, Dermot J Monaghan, and Cecil G Armstrong. Mixed dimensional coupling in finite element stress analysis. *Engineering with Computers*, 18(3):241–252, 2002.

[11] Michael Smith. *ABAQUS/Standard User's Manual, Version 6.9*. Dassault Systèmes Simulia Corp, United States, 2009.

[12] N Osawa, K Hashimoto, J Sawamura, T Nakai, and S Suzuki. Study on shell–solid coupling fe analysis for fatigue assessment of ship structure. *Marine Structures*, 20(3): 143–163, 2007.

[13] Ronald Krueger and T Kevin O'Brien. A shell/3d modeling technique for the analysis of delaminated composite laminates. *Composites Part A: Applied Science and Manufacturing*, 32(1):25–44, 2001.

[14] Ronald Krueger, James G Ratcliffe, and Pierre J Minguet. Panel stiffener debonding analysis using a shell/3d modeling technique. *Composites Science and Technology*, 69 (14):2352–2362, 2009.

[15] Gigliola Salerno, Stefano Mariani, and Alberto Corigliano. Reduced order modeling of composite laminates through solid-shell coupling. *Journal of Aerospace Technology and Management*, 9:397–403, 2017.

[16] Kazuki Ikushima. Nonlinear mechanical FE analysis of thin-plate complex structures using the shell-solid mixed method. In *International Conference on Offshore Mechanics and Arctic Engineering*, volume 84348, page V003T03A006. American Society of Mechanical Engineers, 2020.

[17] J Reinoso, A Blázquez, A Estefani, F París, J Cañas, Edward Arévalo, and F Cruz. Experimental and three-dimensional global-local finite element analysis of a composite component including degradation process at the interfaces. *Composites Part B: Engineering*, 43(4):1929–1942, 2012.

[18] Xiuli Shen, Wentong Hu, and Jiang Fan. Automatic blade blend modeling and hexahedral mesh regeneration for aircraft engine optimization. *Structural and Multidisciplinary Optimization*, 57(3):1345–1355, 2018.

[19] Darcy L Allison, Craig C Morris, Joseph A Schetz, Rakesh K Kapania, and Layne T Watson. Reevaluating conceptual design fidelity: An efficient supersonic air vehicle design case. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 230(3):581–598, 2016.

[20] Jiacheng Xie, Darshan Sarojini, Yu Cai, Jason A Corman, and Dimitri N Mavris. Certification-driven platform for multidisciplinary design space exploration in airframe preliminary design. *Journal of Aircraft*, pages 1–21, 2021.

[21] Kok Yong Lee and Yea Dat Chuah. An automated approach to generate mixed-dimensional finite element analysis model. In *2010 2nd International Conference on Computer Engineering and Technology*, volume 5, pages V5–121. IEEE, 2010.

[22] TT Robinson, CG Armstrong, and R Fairey. Automated mixed dimensional modelling

from 2D and 3D CAD models. *Finite Elements in Analysis and Design*, 47(2): 151–165, 2011.

[23] Song-Hua Ma and Ling Tian. Analysis feature recognition and mixed-dimensional model reconstruction from finite alement analysis mesh model. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 228(7):1197–1207, 2014.

[24] Muhammad Sohaib. Parameterized automated generic model for aircraft wing structural design and mesh feneration for finite element analysis. Master's thesis, Linköping University, Linköping, Sweden, 2011.

[25] Changliang Lai, Junbiao Wang, and Chuang Liu. Parameterized finite element modeling and buckling analysis of six typical composite grid cylindrical shells. *Applied Composite Materials*, 21(5):739–758, 2014.

[26] Chao Ying Zhang, Hu Liu, and Yun Peng Ma. Optimization of aircraft mid-fuselage structure design based on parametric modeling. In *Advanced Materials Research*, volume 933, pages 423–427. Trans Tech Publ, 2014.

[27] Abdelkader Benaouali and Stanislaw Kachel. An automated CAD/CAE integration system for the parametric design of aircraft wing structures. *Journal of Theoretical and Applied Mechanics*, 55(2):447–459, 2017.

[28] Nathan J Love, Rakesh K Kapania, and Darcy L Allison. Integrated aircraft parametric structural & outer mold line geometry modeling software. In *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 0800, 2017.

[29] Yanping Liu, Yongqiang Zhao, Ming Liu, and Xiaoyu Sun. Parameterized

high-precision finite element modelling method of 3d helical gears with contact zone refinement. *Shock and Vibration*, 2019, 2019.

[30] Srinivas C Tadepalli, Ahmet Erdemir, and Peter R Cavanagh. Comparison of hexahedral and tetrahedral elements in finite element analysis of the foot and footwear. *Journal of Biomechanics*, 44(12):2337–2343, 2011.

[31] Steven E Benzley, Ernest Perry, Karl Merkley, Brett Clark, and Greg Sjaardama. A comparison of all hexagonal and all tetrahedral finite element meshes for elastic and elasto-plastic analysis. In *Proceedings, 4th international meshing roundtable*, volume 17, pages 179–191. Citeseer, 1995.

[32] T Blacker. Automated conformal hexahedral meshing constraints, challenges and opportunities. *Engineering with Computers*, 17(3):201–210, 2001.

[33] Timothy J Tautges. The generation of hexahedral meshes for assembly geometry: Survey and progress. *International Journal for Numerical Methods in Engineering*, 50 (12):2617–2642, 2001.

[34] H Böhm, A Hornig, A Langkamp, M Gude, and A Keskin. On the development of strategies for an efficient semi-automated hex-meshing process of complex jet engine component assemblies. In *ASME Turbo Expo 2015: Turbine Technical Conference and Exposition*. American Society of Mechanical Engineers Digital Collection, 2015.

[35] Yong Lu, Rajit Gadh, and Timothy J Tautges. Feature based hex meshing methodology: Feature recognition and volume decomposition. *Computer-Aided Design*, 33(3):221–232, 2001.

[36] TKH Tam and Cecil G Armstrong. Finite element mesh control by integer

programming. *International Journal for Numerical Methods in Engineering*, 36(15): 2581–2605, 1993.

[37] Matthew R Glickman and Vicente J Romero. Analyst-to-analyst variability in simulation-based prediction. Technical report, Sandia National Lab., SAND2017-1263, Albuquerque, NM (United States), 2017.

[38] Carlos G Dávila. Solid-to-shell transition elements for the computation of interlaminar stresses. *Computing Systems in Engineering*, 5(2):193–202, 1994.

[39] Adam J Sadowski and J Michael Rotter. Solid or shell finite elements to model thick cylindrical tubes and shells under global bending. *International Journal of Mechanical Sciences*, 74:143–153, 2013.

[40] UO Akpan, TS Koko, IR Orisamolu, and BK Gallant. Practical fuzzy finite element analysis of structures. *Finite Elements in analysis and Design*, 38(2):93–111, 2001.

[41] Marina Novak and Bojan Dolšak. An intelligent system for structural analysis-based design improvements. *Acta Polytechnica*, 48(6), 2008.

[42] Th Breitfeld and B Kroplin. An expert system for the verification of finite-element calculations. In *Proceedings of the Fourth International Symposium on Assessment of Software Tools*, pages 18–23. IEEE, 1996.

[43] Bojan Dolšak and Marina Novak. Intelligent decision support for structural design analysis. *Advanced Engineering Informatics*, 25(2):330–340, 2011.

[44] Laszlo Hetey, James Campbell, and Rade Vignjevic. Advisory system development for reliable FEM modelling in aerospace. *Aircraft Engineering and Aerospace Technology: An International Journal*, 87:11–8, 2015.

[45] Yi-Cherng Yeh, Yau-Hwaug Kuo, and Deh-Shiu Hsu. Building an expert system for debugging FEM input data with artificial neural networks. *Expert Systems with Applications*, 5(1-2):59–70, 1992.

[46] Shuixiang Li and Minggao Qiao. A hybrid expert system for finite element modeling of fuselage frames. *Expert Systems with Applications*, 24(1):87–93, 2003.

[47] Lotfi A Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on systems, Man, and Cybernetics*, (1):28–44, 1973.

[48] Singiresu S Rao and James P Sawyer. Fuzzy finite element approach for analysis of imprecisely defined systems. *AIAA Journal*, 33(12):2364–2370, 1995.

[49] Larry M Manevitz and Dan Givoli. Towards automating the finite element method: a test-bed for soft computing. *Applied Soft Computing*, 3(1):37–51, 2003.

[50] Hans-Jürgen Zimmermann. *Fuzzy set theory—and its applications*. Springer Science & Business Media, 2011.

[51] Amara Venkata Subba Rao and Dilip Kumar Pratihar. Fuzzy logic-based expert system to predict the results of finite element analysis. *Knowledge-Based Systems*, 20 (1):37–50, 2007.

[52] Guido Sangiovanni. Using fuzzy logic to generate the mesh for the finite element method. In *International Workshop on Fuzzy Logic and Applications*, pages 410–419. Springer, 2005.

[53] Statistics Toolbox Release. Matlab and statistics toolbox release 2018a, 2018.

[54] Lawrence O Hall and Petter Lande. Generation of fuzzy rules from decision trees. *JACIII*, 2(4):128–133, 1998.

[55] Asry Faidhul Ashaari Pinem and Erwin Budi Setiawan. Implementation of classification and regression tree (cart) and fuzzy logic algorithm for intrusion detection system. In *2015 3rd International Conference on Information and Communication Technology (ICoICT)*, pages 266–271. IEEE, 2015.

[56] Mahmoud Omid. Design of an expert system for sorting pistachio nuts through decision tree and fuzzy logic classifier. *Expert Systems with Applications*, 38(4): 4339–4347, 2011.

[57] Adel Bakhshipour, Hemad Zareiforoush, and Iraj Bagheri. Application of decision trees and fuzzy inference system for quality classification and modeling of black and green tea based on visual features. *Journal of Food Measurement and Characterization*, pages 1–15, 2020.

[58] Mato Perić, Zdenko Tonković, Alan Rodić, Martin Surjak, Ivica Garašić, Ivanka Boras, and Srećko Švaić. Numerical analysis and experimental investigation of welding residual stresses and distortions in a T-joint fillet weld. *Materials & Design*, 53: 1052–1063, 2014.

[59] Philip S Beran, Dean Bryson, Andrew S Thelen, Matteo Diez, and Andrea Serani. Comparison of multi-fidelity approaches for military vehicle design. In *AIAA Aviation 2020 Forum*, page 3158, 2020.

[60] Brian Adams, William Bohnhoff, Keith Dalbey, Mohamed Ebeida, John Eddy, Michael Eldred, Russell Hooper, Patricia Hough, Kenneth Hu, John Jakeman, et al. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.13 user's manual. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2020.

[61] American Society of Mechanical Engineers. Guide for verification and validation in computational solid mechanics. ASME, 2006.

[62] Simulia (2017) Abaqus 2017, Documentation. *Dassault Systemes, Rhode Island*, 2017.

[63] Pintu Chandra Shill, MAH Akhand, MD Asaduzzaman, and Kazuyuki Murase. Optimization of fuzzy logic controllers with rule base size reduction using genetic algorithms. *International Journal of Information Technology & Decision Making*, 14 (05):1063–1092, 2015.

[64] Manasi Prafulla Palwankar. Evaluation of a modified fixture for testing composite bolted joints with countersunk fasteners under bearing loads. Master's thesis, San Diego State University, 2016.